aws

Guide du développeur

Amazon OpenSearch Service



Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon OpenSearch Service: Guide du développeur

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Les marques commerciales et la présentation commerciale d'Amazon ne peuvent pas être utilisées en relation avec un produit ou un service extérieur à Amazon, d'une manière susceptible d'entraîner une confusion chez les clients, ou d'une manière qui dénigre ou discrédite Amazon. Toutes les autres marques commerciales qui ne sont pas la propriété d'Amazon appartiennent à leurs propriétaires respectifs, qui peuvent ou non être affiliés ou connectés à Amazon, ou sponsorisés par Amazon.

Table of Contents

Qu'est-ce qu'Amazon OpenSearch Service ? ... 1
Fonctions d'Amazon OpenSearch Service ... 2
Utilisation ... 3
Versions prises en charge d'Elasticsearch et OpenSearch ... 4
Support standard et étendu ... 4
Versions standard prises en charge et étendues ... 7
Calcul des frais de support étendu ... 7
Tarification ... 9
Services connexes ... 9
Configuration ... 12
Accorder des autorisations ... 12
Octroi d’un accès par programmation ... 12
Configurez le AWS CLI ... 14
Ouvrez la console 15
Premiers pas ... 16
Création d'un domaine ... 17
Charger des données pour les indexer ... 18
Option 1 : Charger un seul document ... 19
Option 2 : Charger plusieurs documents ... 19
Rechercher des documents ... 20
Rechercher des documents via la ligne de commande ... 20
Rechercher des documents à l'aide de OpenSearch tableaux de bord ... 21
Supprimer un domaine ... 22
OpenSearch Ingestion d'Amazon ... 24
Concepts clés ... 24
Avantages ... 26
Limites ... 27
Versions de Data Prepper prises en charge ... 27
Dimensionnement des pipelines ... 28
Tarification ... 30
Soutenu Régions AWS ... 30
Configuration des rôles et des utilisateurs ... 30
Rôle du pipeline ... 32
Rôle d'ingestion ... 35

Accorder aux pipelines l'accès aux domaines ... 37
Autoriser les pipelines à accéder aux collections ... 41
Commencer à utiliser OpenSearch Ingestion ... 47
Tutoriel : Ingérer des données dans un domaine ... 47
Tutoriel : Ingérer des données dans une collection ... 56
Caractéristiques du pipeline ... 63
Mise en mémoire tampon persistante ... 64
Fractionnement ... 66
Création de chaînes ... 67
Files d’attente de lettres mortes ... 68
Gestion des indices ... 70
End-to-end accusé de réception ... 73
Contre-pression à la source ... 74
Création de pipelines ... 75
Conditions préalables et rôle IAM requis ... 76
Autorisations IAM requises ... 76
Spécification de la version du pipeline ... 78
Spécification du chemin d'ingestion ... 79
Création de pipelines ... 79
Suivi de l'état de la création du pipeline ... 85
Travailler avec des plans ... 86
Visualisation des pipelines ... 88
Mise à jour des pipelines ... 90
Considérations ... 91
Autorisations nécessaires ... 91
Mise à jour des pipelines ... 92
Déploiements bleu/vert pour les mises à jour du pipeline ... 93
Gestion des coûts du pipeline ... 94
Arrêter un pipeline ... 95
Démarrage d'un pipeline ... 96
Supprime des pipelines. ... 97
Plugins et options pris en charge ... 98
Plugins pris en charge ... 98
Processeurs apatrides et processeurs dynamiques ... 101
Exigences et contraintes de configuration ... 101
Intégrer les pipelines ... 106

Construction du paramètre d'ingestion ... 107
Création d'un rôle d'ingestion ... 107
Services Atlassian ... 110
Amazon DynamoDB ... 127
Amazon DocumentDB ... 144
Confluent Cloud Kafka ... 163
Amazon MSK ... 173
Amazon S3 ... 181
Amazon Security Lake ... 191
Fluent Bit ... 196
Fluentd ... 198
OpenTelemetry Collectionneur ... 200
Kafka autogéré ... 202
Clusters autogérés OpenSearch ... 209
Amazon Kinesis Data Streams ... 218
Étapes suivantes ... 223
AWS Lambda ... 224
Migration des données entre les domaines et les collections ... 227
Limites ... 228
OpenSearch Le service en tant que source ... 228
Spécification de plusieurs OpenSearch récepteurs de domaine de service ... 231
Migration des données vers une collection OpenSearch VPC sans serveur ... 231
Gérer les pipelines à l'aide du AWS SDKs ... 232
Python ... 232
Sécurité lors de OpenSearch l'ingestion ... 236
Configuration de l'accès VPC pour les pipelines ... 237
Gestion de l’identité et des accès ... 242
Surveillance avec CloudTrail ... 252
Pipelines de balisage ... 256
Autorisations nécessaires ... 257
Utilisation des balises (console) ... 257
Utilisation des balises (AWS CLI) ... 257
Journalisation et surveillance ... 258
Surveillance des journaux du pipeline ... 258
Surveillance des métriques du pipeline ... 260
Bonnes pratiques ... 293

Bonnes pratiques d'ordre général 294
CloudWatch Alarmes recommandées 294
Amazon OpenSearch sans serveur 301
Avantages 301
Qu'est-ce qu'Amazon OpenSearch Serverless ? 302
Cas d'utilisation du mode OpenSearch Serverless 303
Comment ça marche 303
Choix d'un type de collection 305
Tarification 306
Soutenu Régions AWS 307
Limites 307
Comparaison entre le OpenSearch service et le mode OpenSearch sans serveur 308
Tutoriel : Débuter avec OpenSearch Serverless 312
Étape 1 : configurer des autorisations 313
Étape 2 : créer une collection 314
Étape 3 : charger et rechercher des données 315
Étape 4 : supprimer la collection 317
Étapes suivantes 317
Créer et gérer les collections 317
Gestion des collections 318
Utilisation de collections de recherche vectorielle 327
Utilisation des politiques de cycle de vie des données 336
Gérer les collections à l'aide du AWS SDKs 345
Création de collections avec CloudFormation 357
Gérer les limites de capacité 358
Configurer les paramètres de capacité 361
Limites de capacité maximale 361
Surveiller l'utilisation de la capacité 362
Ingérer des données dans les collections 362
Autorisations minimales requises 363
OpenSearch Ingestion 364
Fluent Bit 364
Amazon Data Firehose 365
Go 365
Java 368
JavaScript 369

Logstash 372
Python 374
Ruby 376
Autres clients 377
Sécurité en mode OpenSearch Serverless 378
Stratégies de chiffrement 380
Stratégies réseau 381
Stratégies d'accès aux données 382
Authentification IAM et SAML 383
Sécurité de l’infrastructure 383
Démarrer avec la sécurité 384
Gestion de l’identité et des accès 399
Chiffrement 418
Accès réseau 429
Contrôle d'accès aux données 441
Points de terminaison d’un VPC 452
Authentification SAML 464
Validation de conformité 474
Baliser des collections 475
Autorisations nécessaires 476
Baliser des collections (console) 476
Balisage de collections ()AWS CLI 477
Opérations et plugins pris en charge 477
Opérations et autorisations d' OpenSearch API prises en charge 477
OpenSearch Plugins pris en charge 484
Surveillance OpenSearch sans serveur 485
Surveillance avec CloudWatch 486
Surveillance avec CloudTrail 491
Surveillance avec EventBridge 495
Création et gestion des domaines 498
Création de domaines OpenSearch de service 498
Création OpenSearch de domaines de service (console) 498
Création OpenSearch de domaines de service (AWS CLI) 505
Création OpenSearch de domaines de service (AWS SDKs) 507
Création OpenSearch de domaines de service (AWS CloudFormation) 507
Configuration des politiques d'accès 507

Paramètres avancés du cluster 508
Configuration changes 509
Changements qui entraînent généralement des blue/green déploiements 510
Changements qui n'entraînent généralement pas de blue/green déploiements 511
Déterminer si une modification entraînera un déploiement bleu/vert 512
Suivi d'une modification de configuration 516
Étapes d'un changement de configuration 518
Impact sur les performances des déploiements bleu/vert 521
Frais associés aux modifications de la configuration 522
Résolution des erreurs de validation 522
Mises à jour du logiciel de service 528
Mises à jour facultatives ou obligatoires 528
mises à jour des correctifs 529
Considérations 530
Lancement d'une mise à jour 530
Fenêtres creuses 534
Mises à jour de 535
Lorsque les domaines ne sont pas éligibles à une mise à jour 535
Fenêtres creuses 536
Mises à jour du logiciel de service 537
Optimisations Auto-Tune en dehors des heures de pointe 538
Activation de la fenêtre hors pointe 539
Configuration d'une fenêtre personnalisée en dehors des heures de pointe 539
Affichage des actions planifiées 540
Planifier à nouveau les actions 542
Migration depuis les fenêtres de maintenance Auto-Tune 544
Notifications 545
Prise en main des notifications 545
Niveaux de gravité des notifications 546
Exemple d' EventBridge événement 547
Configuration d'un domaine Multi-AZ 548
Multi-AZ avec mode veille 548
Multi-AZ sans mode veille 550
Interruptions des zones de disponibilité 554
Prise en charge de VPC 556
VPC contre domaines publics 556

Limites ... 557
Architecture ... 557
Création d'instantanés d'index ... 565
Prérequis ... 566
Inscription d'un référentiel d'instantanés manuels ... 571
Prise d'instantanés manuels ... 576
Restauration des instantanés ... 577
Suppression d’instantanés manuels ... 580
Automatisation des instantanés grâce à la gestion des instantanés ... 580
Automatisation des instantanés avec Index State Management ... 582
Utilisation de Curator pour les instantanés ... 583
Mise à niveau de domaines ... 583
Chemins de mise à niveau pris en charge ... 584
Mise à niveau d'un domaine (console) ... 588
Mise à niveau d'un domaine (CLI) ... 588
Mise à niveau d'un domaine (SDK) ... 589
Résolution des problèmes liés aux échecs de validation ... 591
Dépannage d'une mise à niveau ... 591
Utilisation d'un instantané pour migrer des données ... 594
Création d'un point de terminaison personnalisé ... 602
Points de terminaison personnalisés pour les nouveaux domaines ... 603
Points de terminaison personnalisés pour les domaines existants ... 604
Cartographie CNAME ... 604
Auto-Tune ... 605
Types de modifications ... 605
Activation ou désactivation d'Auto-Tune ... 607
Optimisation ... 608
Auto-Tune ... 609
Balisage des domaines ... 609
Exemples de balisage ... 610
Balisage des domaines (console) ... 611
Marquage de domaines ()AWS CLI ... 611
Marquage de domaines ()AWS SDKs ... 613
Exécution d'actions en bloc ... 614
Redémarrer le OpenSearch processus sur un nœud de données ... 615
Redémarrage d'un nœud de données ... 615

Redémarrer le processus des tableaux de bord 616
Limites 617
Travailler avec des requêtes directes 618
Tarification 618
Limites 619
Limitations générales 619
Limitations pour Amazon S3 619
Limitations pour Amazon CloudWatch Logs 620
Limitations d'Amazon Security Lake 620
Recommandations 621
Recommandations générales 621
Recommandations d'Amazon S3 622
CloudWatch Recommandations relatives aux journaux 623
Recommandations de Security Lake 623
Quotas 624
Quotas pour Amazon S3 624
Quotas pour les CloudWatch journaux 625
Quotas pour Security Lake 626
Soutenu Régions AWS 628
Disponible Régions AWS pour Amazon S3 628
Disponible Régions AWS pour les CloudWatch journaux 628
Disponible Régions AWS pour Security Lake 629
Requêtes directes dans S3 629
Création d'une source de données S3 630
Configuration d'une source de données S3 638
Requêtes directes dans les CloudWatch journaux 641
Création d'une source de données CloudWatch Logs 642
Configuration d'une source de données CloudWatch Logs 648
Requêtes directes dans Security Lake 650
Création d'une source de données Security Lake 651
Configuration d'une source de données Security Lake 657
Gestion d'une source de données 660
Surveillance à l'aide CloudWatch de sources de données métriques 661
Activation et désactivation des sources de données 664
Surveillance avec le AWS budget 665
Suppression d'une source de données 665

Optimisation des performances des requêtes 666
Ignorer les index 667
Vues matérialisées 667
Index de couverture 668
Commandes SQL et PPL prises en charge 668
Commandes SQL prises en charge 669
Commandes PPL prises en charge 879
Surveillance des domaines 1064
Surveillance des métriques d'un cluster 1065
Afficher les métriques dans CloudWatch 1066
Interprétation des graphiques d'état en OpenSearch service 1067
Métriques du cluster 1067
Métriques du nœud principal dédié 1076
Mesures dédiées concernant le nœud de coordinateur 1077
Métriques du volume EBS 1078
Métriques des instances 1081
UltraWarm métriques 1095
Métriques de stockage à froid 1101
OR1 métriques 1102
Métriques d'alerte 1103
Métriques de détection d'anomalies 1104
Métriques de recherche asynchrone 1106
Auto-Tune 1108
Multi-AZ avec métriques 1109
Métriques ponctuelles 1112
Métriques SQL 1112
Métriques k-NN 1113
Métriques de recherche inter-clusters 1117
Métriques de réplication inter-clusters (CCR) 1117
Métriques Learning to Rank 1119
Métriques du langage de traitement PPL (Piped Processing Language) 1120
Surveillance des journaux 1121
Activation de la publication des journaux (console) 1122
Activation de la publication des journaux (AWS CLI) 1125
Activation de la publication des journaux (AWS SDKs) 1127
Activation de la publication des journaux (CloudFormation) 1127

Définition des seuils de lenteur de journalisation des demandes de recherche 1129
Définition des seuils de lenteur de journalisation des partitions 1130
Tester les journaux lents 1130
Affichage des journaux 1131
Surveillance des journaux d'audit 1131
Limites 1132
Activation des journaux d'audit 1133
Activez la journalisation des audits à l'aide du AWS CLI 1135
Activation du journal d'audit à l'aide de l'API de configuration 1135
Couches et catégories de journaux d'audit 1135
Paramètres des journaux d'audit 1138
Exemples de journaux d'audit 1142
Configuration des journaux d'audit à l'aide de l'API REST 1145
Surveillance des événements 1146
Événements de mise à jour du logiciel de service 1147
Événements Auto-Tune 1154
Événements relatifs à l'état du cluster 1159
Événements de point de terminaison d'un VPC 1172
Événements de retrait de nœuds 1175
Événements de retrait de nœuds dégradés 1177
Événements d'erreur de domaine 1179
Didacticiel : Écoute des événements OpenSearch de service 1181
Tutoriel : Envoi d'alertes SNS pour les mises à jour disponibles 1183
Surveillance avec CloudTrail 1185
Informations sur OpenSearch le service Amazon dans CloudTrail 492
Comprendre les entrées du fichier journal Amazon OpenSearch Service 493
Sécurité 1190
Protection des données 1191
Chiffrement au repos 1192
Node-to-node chiffrement 1196
Gestion de l’identité et des accès 1197
Types de stratégies 1197
Formulation et signature de demandes OpenSearch de service 1207
En cas de conflit entre plusieurs stratégies 1208
Références des éléments de stratégie 1209
Options avancées et considérations relatives aux API 1215

Configuration des politiques d'accès ... 1219
Exemples de stratégies supplémentaires ... 1219
Référence des autorisations d'API ... 1219
AWS politiques gérées ... 1220
Prévention du problème de l’adjoint confus entre services ... 1230
Contrôle précis des accès ... 1232
Vue d'ensemble : contrôle d'accès précis et OpenSearch sécurité des services ... 1233
Concepts clés ... 1237
À propos de l'utilisateur principal ... 1237
Activation du contrôle précis des accès ... 1239
Accès aux OpenSearch tableaux de bord en tant qu'utilisateur principal ... 1243
Gestion des autorisations ... 1245
Configurations recommandées ... 1251
Limites ... 1254
Modification de l'utilisateur maître ... 1255
Utilisateurs principaux supplémentaires ... 1256
Instantanés manuels ... 1258
Intégrations ... 1258
Différences d'API REST ... 1259
Didacticiel : contrôle précis des accès avec l'authentification Cognito ... 1261
Didacticiel : base de données utilisateur interne et authentification de base ... 1266
Validation de conformité ... 1269
Résilience ... 1271
Jetons Web JSON ... 1271
Considérations ... 1272
Modification de la stratégie d'accès au domaine ... 1272
Configuration de l'authentification et de l'autorisation JWT ... 1273
Utiliser un JWT pour envoyer une demande de test ... 1273
Sécurité de l’infrastructure ... 1275
Utilisation des points de terminaison d'un OpenSearch VPC gérés par le Service ... 1276
Authentification SAML pour les tableaux de bord OpenSearch ... 1281
Présentation de la configuration SAML ... 1282
Considérations ... 1282
Authentification SAML pour les domaines VPC ... 1283
Modification de la stratégie d'accès au domaine ... 1283

Configuration de l'authentification initiée par le fournisseur de services ou le fournisseur d'identité ... 1285
Configuration de l'authentification initiée à la fois par le SP et l'IdP ... 1292
Configuration de l'authentification SAML (AWS CLI) ... 1292
Configuration de l'authentification SAML (API de configuration) ... 1293
Résolution des problèmes SAML ... 1293
Désactivation de l'authentification SAML ... 1297
Support du centre d'identité IAM pour Amazon Service OpenSearch ... 1298
Authentification Amazon Cognito pour les tableaux de bord OpenSearch ... 1301
Prérequis ... 1302
Configurer un domaine pour utiliser l'authentification Amazon Cognito ... 1306
Autorisation du rôle authentifié ... 1309
Configuration des fournisseurs d'identité ... 1311
(Facultatif) Configuration du contrôle précis des accès ... 1311
(Facultatif) Personnalisation de la page de connexion ... 1312
(Facultatif) Configuration de la sécurité avancée ... 1313
Test ... 1313
Quotas ... 1313
Problèmes de configuration courants ... 1314
Désactivation de l'authentification Amazon Cognito pour les tableaux de bord OpenSearch ... 1319
Suppression de domaines utilisant l'authentification Amazon Cognito pour les tableaux de bord OpenSearch ... 1319
Utilisation des rôles liés à un service ... 1319
Rôle de création de domaine VPC et de source de données ... 1320
Rôle de création d'une collection ... 1324
Rôle de création d'un pipeline ... 1326
Exemple de code ... 1330
Compatibilité des clients Elasticsearch ... 1330
Compression des requêtes HTTP ... 1331
Activation de la compression gzip ... 1331
En-têtes obligatoires ... 1332
Exemple de code (Python 3) ... 1332
À l'aide du AWS SDKs ... 1334
Java ... 1334
Python ... 1345

Nœud 1348
Indexation des données 1351
Restrictions de dénomination des index 1351
Réduction de la taille des réponses 1352
Codecs d'index 1354
Chargement de données de streaming dans le OpenSearch service 1354
Chargement de données de streaming depuis OpenSearch Ingestion 1355
Chargement de données de streaming à partir d'Amazon S3 1355
Chargement de données de streaming à partir d'Amazon Kinesis Data Streams 1361
Chargement de données de streaming à partir d'Amazon DynamoDB 1366
Chargement de données de streaming depuis Amazon Data Firehose 1370
Chargement de données de streaming depuis Amazon CloudWatch 1371
Chargement de données de streaming depuis AWS IoT 1371
Chargement de données avec Logstash 1371
Configuration 1372
Recherche de données 1375
Recherches d'URI 1376
Recherches dans le corps de la demande 1377
Optimisation des champs 1379
Mise en évidence des résultats de recherche 1379
API Count 1381
Pagination des résultats de recherche 1382
Point dans le temps 1382
Les `size` paramètres `from` et 1382
Langage de requête Dashboards 1383
Packages 1385
Autorisations requises 1386
Chargement des packages dans Amazon S3 1386
Importation et association de packages 1387
Utilisation de packages avec OpenSearch 1388
Mise à jour de packages 1392
Mise à jour manuelle des index avec un nouveau dictionnaire 1396
Dissociation et suppression de packages 1398
Plug-ins personnalisés 1399
Plug-ins tiers 1417
Prise en charge de SQL 1421

Exemple d'appel ... 1423
Remarques et différences ... 1423
SQL Workbench ... 1424
CLI SQL ... 1274
Pilote JDBC ... 1424
Pilote ODBC ... 1424
Recherche k-NN ... 1425
Prise en main de k-NN ... 1425
Différences, réglage et limitations de k-NN ... 1428
Recherche croisée entre clusters ... 1428
Limites ... 1429
Conditions préalables à la recherche inter-clusters ... 1430
Tarification de la recherche inter-clusters ... 1430
Configuration d'une connexion ... 1430
Suppression d'une connexion ... 1432
Configuration de la procédure de sécurité et d'exemples ... 1432
OpenSearch Tableaux de bord ... 1439
Learning to Rank ... 1439
Prise en main de Learning to Rank ... 1440
API Learning to Rank ... 1461
Recherche asynchrone ... 1468
Exemple d'appel de recherche ... 1468
Autorisations relatives à la recherche asynchrone ... 1470
Paramètres de recherche asynchrone ... 1471
Recherche croisée entre clusters ... 1471
UltraWarm ... 1473
Point dans le temps ... 1473
Considérations ... 1474
Créer un PIT ... 1474
Permissions ponctuelles ... 1476
Réglages PIT ... 1477
Recherche croisée entre clusters ... 1477
UltraWarm ... 1478
Recherche sémantique ... 1478
Recherche par segments simultanés ... 1478
Génération de requêtes en langage naturel ... 1479

Prérequis 1480
Premiers pas 1480
Configurer des autorisations 1480
Automatisation des configurations 1481
OpenSearch Tableaux de bord 1482
Contrôle de l'accès aux tableaux de bord 1483
Utilisation d'un proxy pour accéder au OpenSearch service à partir de tableaux de bord ... 1483
Configuration des tableaux de bord pour utiliser un serveur de carte WMS 1487
Connexion d'un serveur de tableaux de bord local au service OpenSearch 1488
Gestion des index dans les tableaux de bord 1490
Fonctionnalités supplémentaires 1490
OpenSearch UI 1492
Historique de versions 1493
Premiers pas 1495
Autorisations requises pour créer des applications Amazon OpenSearch Service 1495
Création d'une application 1499
Gestion des administrateurs d'applications 1507
Activation de la fédération SAML avec IAM 1511
Étape 1 : configurer l'application du fournisseur d'identité (Okta) 1512
Étape 2 : Configuration de AWS la configuration pour Okta 1516
Étape 3 : Création de la politique d'accès à Amazon OpenSearch Service dans IAM 1518
Étape 4 : vérifier l'expérience d'authentification unique initiée par le fournisseur d'identité avec SAML 1521
Gestion des associations de sources de données et des autorisations d'accès aux VPC 1525
Associer une source de données à une application d' OpenSearch interface utilisateur 1525
Gestion de l'accès aux domaines dans un VPC 1526
Configuration de l'accès aux collections OpenSearch sans serveur dans un VPC 1528
Utilisation des espaces OpenSearch de travail Amazon Service 1531
Création d'espaces de travail d'applications d' OpenSearch interface utilisateur 1531
Confidentialité de l'espace de travail et collaborateurs 1532
Types d'espaces de travail 1533
Accès aux données entre régions et entre comptes grâce à la recherche entre clusters 1534
Configuration des autorisations d'accès pour l'accès aux données entre régions et entre comptes grâce à la recherche entre clusters 1536
Création d'une connexion entre les domaines 1539

Test de votre configuration de sécurité pour l'accès aux données entre régions et entre comptes grâce à la recherche entre clusters 1541
Suppression d’une connexion 1544
Gestion de l'accès à l' OpenSearch interface utilisateur depuis un point de terminaison VPC . 1545
Création d'une connexion privée entre un VPC et une interface utilisateur OpenSearch 1545
Mise à jour de la politique de point de terminaison du VPC pour autoriser l'accès à l' OpenSearch application d'interface utilisateur 1547
Révocation de l'accès à l' OpenSearchinterface utilisateur dans une politique de point de terminaison VPC 1547
Gestion des index 1549
UltraWarm rangement 1549
Prérequis 1550
UltraWarm exigences relatives aux performances et exigences relatives aux performances 1552
UltraWarm tarification 1553
Activant UltraWarm 1553
Migration des index vers le stockage UltraWarm 1556
Automatisation des migrations 1559
Réglage des migrations 1559
Annulation des migrations 1560
Liste des index hot et warm 1560
Rebasculement d'index à chaud vers le stockage hot 1560
Restauration des index à chaud à partir d'instantanés 1561
Instantanés manuels des index warm 1562
Migration d'index à chaud vers le stockage à froid 1563
Bonnes pratiques relatives aux index KNN 1564
Désactivation UltraWarm 1565
Stockage à froid 1565
Prérequis 1566
Stockage à froid : exigences et considérations relatives aux performances 1567
Tarification du stockage à froid 1568
Activation du stockage à froid 1568
Gestion des index froids dans Dashboards OpenSearch 1570
Migration des index vers le stockage à froid 1570
Automatisation des migrations vers le stockage à froid 1572
Annulation des migrations vers le stockage à froid 1572

Liste des index froids ... 1573
Migration d'index à froid vers le stockage à chaud ... 1577
Restauration des index à froid à partir d'instantanés ... 1578
Annulation des migrations du stockage à froid vers le stockage à chaud ... 1578
Mise à jour des métadonnées des index froids ... 1579
Suppression d'index froids ... 1579
Désactivation du stockage à froid ... 1580
OpenSearch stockage optimisé ... 1580
Limites ... 1581
Réglage pour un meilleur débit d'ingestion ... 1581
En quoi les instances OpenSearch optimisées diffèrent-elles des autres instances ... 1581
En quoi OR1 diffère-t-il du UltraWarm stockage ... 1582
Approvisionnement d'un domaine avec des instances OR1 ... 1583
Gestion d'états des index ... 1584
Créer une politique ISM ... 1585
Exemples de politiques ... 1586
Modèles ISM ... 1590
Différences ... 1590
Didacticiel : Automatisation des processus ISM ... 1592
Cumulatifs d'index ... 1597
Création d'une tâche de cumulatif d'index ... 1597
Transformations d'index ... 1599
Création d'une tâche de transformation d'index ... 1599
Réplication inter-clusters (CCR) ... 1601
Limites ... 1602
Prérequis ... 1602
Conditions d'autorisation ... 1603
Configurer une connexion inter-clusters ... 1604
Démarrer la réplication ... 1605
Confirmer la réplication ... 1606
Mettre en pause et reprendre la réplication ... 1607
Arrêter la réplication ... 1608
Suivi automatique ... 1608
Mise à niveau des domaines connectés ... 1610
Réindexation à distance ... 1610
Prérequis ... 1611

Réindexer les données entre les domaines Internet OpenSearch du service 1611
Réindexer les données lorsque le domaine distant se trouve dans un VPC 1613
Réindexer les données entre les domaines non liés OpenSearch aux services 1618
Réindexer des jeux de données volumineux 1618
Paramètres de réindexation à distance 1620
Flux de données 1621
Premiers pas avec les flux de données 1621
Surveillance des données 1625
Alerte 1625
Autorisations relatives aux alertes 1626
Démarrer avec les alertes 1626
Notifications 1627
Différences 1628
Détection des anomalies 1629
..... 1630
Didacticiel : Détection d'une utilisation élevée de l'UC avec un détecteur d'anomalies 1634
Prise en charge d’Amazon Q 1638
Soutenu Régions AWS 1639
Configuration d'Amazon Q OpenSearch 1639
Génération de visualisations en langage naturel 1640
Afficher les résumés et les informations sur les alertes 1641
Avant de commencer 1644
Affichage des résumés et des informations sur les alertes 1645
Consultez les résumés des résultats de requêtes générés par Amazon Q sur la page Discover 1646
Décrire les détecteurs d'anomalies 1647
Accédez au chat Amazon Q pour les questions OpenSearch de service 1648
Machine learning 1652
Connecteurs pour Services AWS 1652
Prérequis 1653
Création d'un connecteur OpenSearch de service 1656
Connecteurs pour plateformes externes 1658
Prérequis 1659
Création d'un connecteur OpenSearch de service 1662
CloudFormation intégrations de modèles 1664
Prérequis 1665

Amazon SageMaker AI modèles ... 1666
Modèles Amazon Bedrock ... 1667
Paramètres ML Commons non pris en charge ... 1668
Plugin Flow Framework ... 1669
Création de connecteurs ML dans OpenSearch Service ... 1669
Configurer des autorisations ... 1677
Analyses de sécurité ... 1679
Composants et concepts d'analyse de sécurité ... 1679
Types de journal ... 1680
Détecteurs ... 1680
Règles ... 1680
Conclusions ... 1681
Alerts (Alertes) ... 1681
Exploration des données de sécurité ... 1681
Configurer des autorisations ... 1683
Résolution des problèmes ... 1685
Aucune erreur d'index de ce type ... 1685
Observabilité ... 1686
Explorez vos données grâce à l'analytique des événements ... 1686
Créer des visualisations ... 1689
Plongez plus profondément avec Trace Analytics ... 1690
Trace Analytics ... 1690
Prérequis ... 1691
OpenTelemetry Configuration des échantillons de collecteur ... 1692
OpenSearch Configuration de l'échantillon d'ingestion ... 1693
Exploration des données de suivi ... 1694
Langage PPL (Piped Processing Language) ... 1695
... 1696
Bonnes pratiques ... 1698
Surveillance et alertes ... 1698
Configuration des CloudWatch alarmes ... 1698
Activer la publication des journaux ... 1699
Stratégie de partition ... 1699
Déterminer le nombre de partitions et de nœuds de données ... 1700
Éviter l'asymétrie de stockage ... 1701
Stabilité ... 1701

Tenez-vous au courant de OpenSearch 1701
Améliorez les performances des instantanés 1702
Activer les nœuds principaux dédiés 1703
Déployer sur plusieurs zones de disponibilité 1703
Contrôler le flux d'ingestion et la mise en mémoire tampon 1703
Créer des mappages pour les charges de travail de recherche 1704
Utiliser des modèles d'index 1705
Gérer les index avec Index State Management 1706
Supprimez les index inutilisés 1706
Utiliser plusieurs domaines pour une haute disponibilité 1706
Performances 1707
Optimiser la taille et la compression des demandes groupées 1707
Réduire la taille des réponses aux demandes groupées 1707
Régler les intervalles d'actualisation 1708
Activer Auto-Tune 1708
Sécurité 1708
Activer le contrôle précis des accès 1708
Déployer des domaines dans un VPC 1709
Appliquer une stratégie d'accès restrictive 1709
Activer le chiffrement au repos 1709
Activer node-to-node le chiffrement 1710
Moniteur avec AWS Security Hub 1710
Optimisation des coûts 1710
Utiliser les types d'instance de dernière génération 1710
Utilisation des derniers volumes Amazon EBS gp3 1711
Utilisation UltraWarm et stockage à froid pour les données des journaux de séries chronologiques 1711
Examiner les recommandations pour les instances réservées 1712
Dimensionnement des domaines 1712
Calcul des exigences de stockage 1712
Choix du nombre de partitions 1714
Choix des types d'instances et test 1716
Mise à l'échelle d'une capacité de plusieurs péta-octets 1718
Nœuds de coordination dédiés 1720
Quand utiliser des nœuds de coordination dédiés 1720
Architecture et comportement 1721

Exigences et limitations 1721
Provisionnement de nœuds de coordination dédiés 1721
Bonnes pratiques 1722
Nœuds maîtres dédiés 1724
Choix du nombre de nœuds principaux dédiés 1725
Choix des types d'instance pour les nœuds principaux dédiés 1727
CloudWatch Alarmes recommandées 1728
Référence générale 1737
Types d’instance pris en charge 1737
Types d'instance de la génération actuelle 1737
Types d’instance d’ancienne génération 1757
Fonctionnalités par version 1761
Plugins par version du moteur 1767
Plug-ins optionnels 1771
Opérations prises en charge 1772
Différences notables entre API 1772
Quotas 1828
UltraWarm quotas de stockage 1829
Nombre de nœuds de données par AZ 1727
Limite totale de nœuds par famille d'instances 1829
Quotas de taille du volume EBS 1830
Quotas de réseau 1840
Quotas de taille de partition 1848
Quotas de nombre d'unités 1848
Quota de processus Java 1849
Quota de stratégie de domaine 1849
Instances réservées 1849
Achat d'instances réservées (console) 1850
Achat d'instances réservées (AWS CLI) 1851
Achat d'instances réservées (AWS SDKs) 1853
Examen des coûts 1855
Autres ressource prises en charge 1855
Didacticiels 1857
Création et recherche de documents 1857
Prérequis 1857
Ajout d'un document à un index 1858

Création d'ID IDs 1859
Mise à jour d'un document avec une commande POST 1860
Exécution d'actions en bloc 1861
Recherche de documents 1862
Ressources connexes 1864
Migration vers le service OpenSearch 1864
Création et chargement de l'instantané 1864
Création d'un domaine 1866
Accordez des autorisations d'accès au compartiment S3. 1867
Restaurer l'instantané. 1869
Création d'une application de recherche 1872
Prérequis 1873
Étape 1 : Indexer des exemples de données 1873
Étape 2 : Création et déploiement de la fonction Lambda 1874
Étape 3 : Création de l'API dans API Gateway 1877
Étape 4 : (Facultatif) modifier la stratégie d'accès au domaine 1879
Mapper le rôle Lambda (si vous utilisez le contrôle d'accès précis) 1882
Étape 5 : Tester l'application web 1883
Étapes suivantes 1885
Visualisation des appels au support 1886
Étape 1 : Configurer les prérequis 1887
Étape 2 : Copier un exemple de code 1888
(Facultatif) Étape 3 : Indexer des exemples de données 1892
Étape 4 : Analyser et visualiser vos données 1894
Étape 5 : Nettoyage des ressources et étapes suivantes 1898
OpenSearch Renommage d'Amazon Service 1900
Nouvelle version d'API 1900
Types d'instances renommés 1901
Modifications des stratégies d'accès 1901
Politiques IAM 1901
Politiques SCP 1901
Nouveaux types de ressources 1902
Kibana renommé Tableaux de bord OpenSearch 1903
CloudWatch Métriques renommées 1904
Modifications apportées à la console Billing and Cost Management 1905
Nouveau format d'événement 1906

Qu'est-ce qui demeure identique ? ... 1906
Mise en route : mettez à niveau vos domaines vers OpenSearch 1.x ... 1906
Résolution des problèmes ... 1908
Impossible d'accéder aux OpenSearch tableaux de bord ... 1908
Impossible d'accéder au domaine VPC ... 1908
Cluster en lecture seule ... 1908
Statut de cluster rouge ... 1910
Correction automatique des clusters rouges ... 1911
Récupération après une importante charge de traitement continue ... 1912
Statut de cluster jaune ... 1914
ClusterBlockException ... 1915
Manque d'espace de stockage disponible ... 1915
Pression mémoire élevée de la JVM ... 1915
Erreur lors de la migration vers le mode Multi-AZ avec mode veille ... 1916
Création d'un index, d'un modèle d'index ou d'une politique ISM lors de la migration de domaines sans mode veille vers des domaines en mode veille ... 1685
Nombre de copies de données incorrect ... 1917
JVM OutOfMemoryError ... 1917
Nœuds de cluster en échec ... 1918
Limite maximale de partitions dépassée ... 1918
Domaine bloqué dans l'état de traitement ... 1919
Solde de débordement EBS faible ... 1919
La métrique EBS augmente lors du redimensionnement du volume ... 1920
Impossible d'activer les journaux d'audit ... 1920
Impossible de fermer l'index ... 1921
Vérifications des licences des clients ... 1921
Limitation des demandes ... 1921
Impossible d'accéder au nœud via SSH ... 1921
Erreur d'instantané « Non valide pour la classe de stockage de l'objet » ... 1921
En-tête d'hôte non valide ... 1922
Type d'instance M3 non valide ... 1922
Les hot queries cessent de fonctionner après l'activation UltraWarm ... 1922
Impossible de revenir à une version plus ancienne après la mise à niveau ... 1923
Résumé nécessaire des domaines pour toutes les Régions AWS ... 1923
Erreur du navigateur lors de l'utilisation des OpenSearch tableaux de bord ... 1924
Asymétrie des partitions et de stockage des nœuds ... 1924

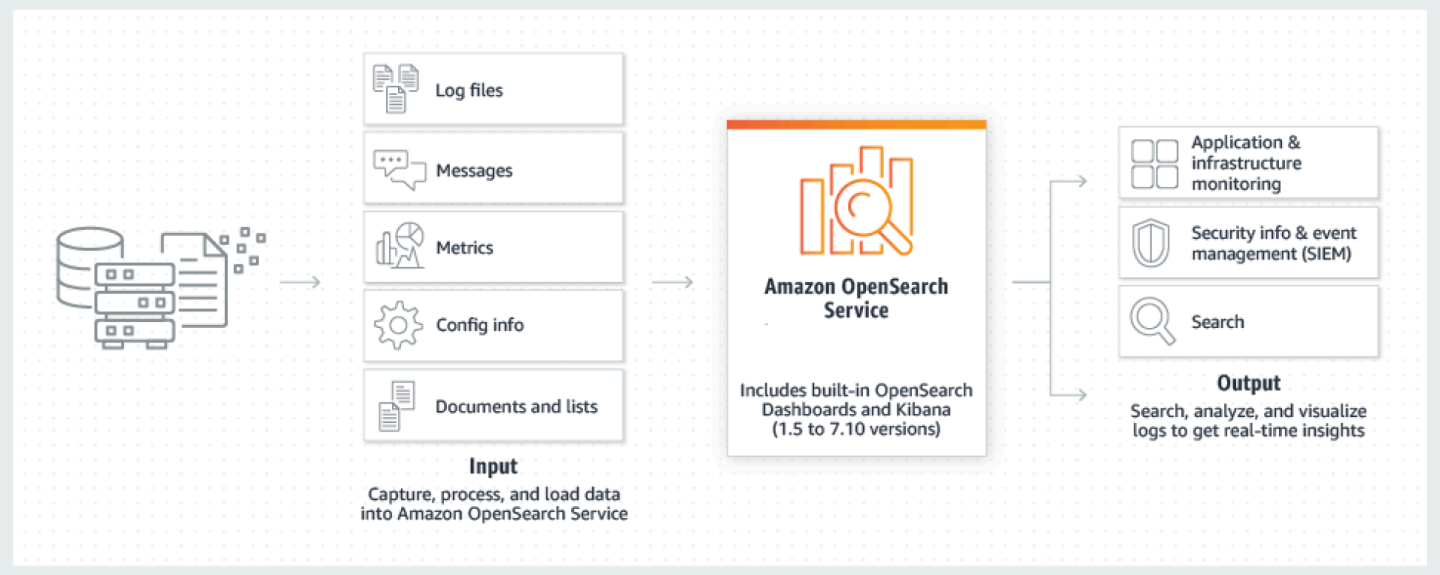
Asymétrie des partitions et du stockage des index 1925
Opération non autorisée après la sélection de l'accès VPC 1926
Blocage du chargement suite à la création d'un domaine VPC 1926
Demandes refusées à l' OpenSearch API 1926
Impossible de se connecter à partir d'Alpine Linux 1927
Trop de demandes pour Search Backpressure 1928
Erreur de certificat lors de l'utilisation du kit SDK 1928
L'installation du plugin personnalisé échoue en raison de la compatibilité des versions 1929
Historique de la documentation 1932
Mises à jour antérieures 1989
AWS Glossaire 1993
..... mcmxciv

Qu'est-ce qu'Amazon OpenSearch Service ?

Amazon OpenSearch Service est un service géré qui facilite le déploiement, l'utilisation et la mise à l'échelle de OpenSearch clusters dans le AWS cloud. Un domaine de OpenSearch service est synonyme de OpenSearch cluster. Les domaines sont des clusters qui contiennent les paramètres, les types d'instances, le nombre d'instances et les ressources de stockage que vous spécifiez. Amazon OpenSearch Service prend en charge OpenSearch Elasticsearch OSS hérité (jusqu'à 7.10, la dernière version open source du logiciel). Lorsque vous créez un domaine, vous avez la possibilité de choisir le moteur de recherche que vous voulez utiliser.

OpenSearchest un moteur de recherche et d'analyse entièrement open source qui peut être utilisé, entre autres, pour l'analyse de journaux, la surveillance en temps réel des applications et l'analyse des flux de clics. Pour plus d'informations, consultez la documentation OpenSearch .

Amazon OpenSearch Service fournit toutes les ressources pour votre OpenSearch cluster et le lance. Il détecte et remplace aussi automatiquement les nœuds de OpenSearch service défaillants, ce qui réduit le traitement associé aux infrastructures autogérées. Vous pouvez facilement faire évoluer votre cluster avec un seul appel de l'API ou en quelques clics sur la console.



Pour commencer à utiliser OpenSearch Service, vous devez créer un domaine de OpenSearch service, équivalent à un OpenSearch cluster. Chaque EC2 instance du cluster agit comme un nœud OpenSearch de service.

Vous pouvez utiliser la console OpenSearch de service pour installer et configurer un domaine en quelques minutes. Si vous préférez un accès programmatique, vous pouvez utiliser le AWS CLIAWS SDKs, le ou Terraform.

Fonctions d'Amazon OpenSearch Service

OpenSearch Service inclut les fonctionnalités suivantes :

Évolutivité

- Plusieurs configurations d'UC, de mémoire et de capacité de stockage, appelées types d'instances, y compris les instances économiques Graviton
- Supporte jusqu'à 1002 nœuds de données
- Jusqu'à 25 Po de stockage connecté
- Stockage UltraWarmà froid et économique pour les données en lecture seule

Sécurité

- AWS Identity and Access Management Contrôle d'accès (IAM)
- Intégration aisée avec Amazon VPC et les groupes de sécurité VPC
- Chiffrement des données au repos et node-to-node chiffrement
- Authentification Amazon Cognito, HTTP basique ou SAML pour les tableaux de bord OpenSearch
- Sécurité au niveau de l'index, du document et du champ
- Journaux d'audit
- Multi-location de Dashboards

Stabilité

- Plusieurs emplacements géographiques pour vos ressources, appelés régions et zones de disponibilité
- Allocation du nœud sur deux ou trois zones de disponibilité dans la même région AWS , appelée Multi-AZ
- Les nœuds maîtres dédiés à décharger des tâches de gestion de cluster
- Les instantanés automatiques pour sauvegarder et restaurer des domaines OpenSearch de service

Flexibilité

- Prise en charge de SQL pour l'intégration à des applications de business intelligence (BI)
- Packages personnalisés pour améliorer les résultats de recherche

Intégration aux services les plus connus

- Visualisation des données à l'aide de OpenSearch tableaux de bord
- Intégration à Amazon CloudWatch pour surveiller les métriques OpenSearch de domaine Service et configurer les alarmes
- Intégration à AWS CloudTrail pour vérifier les appels d'API de configuration vers les domaines OpenSearch de service
- Intégration à Amazon S3, Amazon Kinesis et Amazon DynamoDB pour le chargement de données de streaming dans Service OpenSearch
- Alertes émises par Amazon SNS lorsque vos données dépassent certains seuils

Quand l'utiliser OpenSearch par rapport à Amazon OpenSearch Service

Utilisez le tableau suivant pour vous aider à déterminer si le OpenSearch service Amazon fourni ou autogéré OpenSearch est le bon choix pour vous.

OpenSearch	Amazon OpenSearch Service
• Votre organisation est disposée à surveiller et à gérer manuellement des clusters auto-provisionnés et dispose de personnes possédant les compétences nécessaires pour le faire. • Vous souhaitez un contrôle complet de votre code au niveau de la compilation. • Votre entreprise préfère, ou utilise uniquement, les logiciels open source.	• Vous ne souhaitez pas gérer, surveiller et entreteni r manuellement votre infrastructure. • Vous recherchez des moyens simples de gérer les coûts d'analyse croissants en répartissant vos données sur différents niveaux de stockage, en tirant parti de la durabilité et du faible coût d'Amazon S3. • Vous souhaitez tirer parti des intégrations avec d'autres sites Services AWS tels que DynamoDB, Amazon DocumentDB (avec compatibilité MongoDB), IAM et. CloudWatch CloudFormation

OpenSearch	Amazon OpenSearch Service
• Vous avez une stratégie multicloud qui nécessite des technologies qui ne sont pas spécifiques à un fournisseur. • Votre équipe est capable de résoudre tous les problèmes de production critiques. • Vous voulez avoir la flexibilité d'utiliser, de modifier et d'étendre le produit comme vous le souhaitez. • Vous souhaitez accéder immédiatement aux nouvelles fonctionnalités dès leur sortie.	• Vous souhaitez accéder facilement à un formulaire d'assistance Support pour la maintenance préventive et en cas de problème de production. • Vous souhaitez tirer parti de fonctionnalités telles que l'autoréparation, la maintenance proactive, la résilience et les sauvegardes.

Versions prises en charge d'Elasticsearch et OpenSearch

OpenSearch Le service prend en charge les versions suivantes de OpenSearch:

- 2,19, 2,17, 2,15, 2,13, 2,11, 2,9, 2,7, 2,5, 2,3, 1,3, 1,2, 1,1 et 1,0

OpenSearch Le service prend en charge les versions suivantes de l'ancienne version d'Elasticsearch :

- 7,10, 7,9, 7,8, 7,4, 7,1, 6,8, 6,7, 6,5, 6,4, 6,3, 6,2, 6,0, 5,6, 5,5, 5,3, 5,1, 2,3 et 1,5

Nous vous recommandons de passer à la dernière OpenSearch version disponible pour tirer le meilleur parti du OpenSearch Service, en termes de rapport qualité-prix, de richesse des fonctionnalités et d'amélioration de la sécurité.

Support standard et étendu

AWS fournit des corrections de bogues et des mises à jour de sécurité pour les versions bénéficiant d'un support standard. Pour les versions bénéficiant d'un support étendu, AWS propose des correctifs de sécurité critiques pendant au moins 12 mois après la fin du support standard, moyennant un

montant forfaitaire par heure d'instance normalisée (NIH). Le NIH est basé sur la taille de l'instance et les heures d'utilisation.

Les frais de support étendu s'appliquent automatiquement lorsqu'un domaine exécute une version qui n'est plus couverte par le support standard. Pour éviter ces frais, passez à une version prise en charge.

Les tableaux suivants indiquent le calendrier de fin de support pour les anciennes versions d'Elasticsearch OpenSearch et les anciennes versions.

OpenSearch Le service prend en charge plusieurs versions OpenSearch et les anciennes versions open source d'Elasticsearch. Pour certaines versions, nous avons déjà publié les dates de fin du support standard et les dates de support étendues. Nous vous recommandons de passer à la dernière OpenSearch version disponible pour tirer le meilleur parti du OpenSearch Service en termes de rapport qualité-prix, de richesse des fonctionnalités et d'amélioration de la sécurité. Les tableaux suivants fournissent des listes d'Elasticsearch et de ses OpenSearch versions, ainsi que leurs calendriers de support.

Le calendrier de fin de support pour les versions d'Elasticsearch est le suivant :

Version du logiciel	Support standard	Support étendu
Elasticsearch versions 1.5 et 2.3	7 novembre 2025	7 novembre 2026
Versions 5.1 à 5.5 d'Elasticsearch	7 novembre 2025	7 novembre 2026
Elasticsearch versions 5.6	7 novembre 2025	7 novembre 2028
Versions d'Elastic	7 novembre 2025	7 novembre 2026

Version du logiciel	Support standard	Support étendu
search 6.0 à 6.7		
Elasticsearch versions 6.8	Non annoncé	Non annoncé
Versions 7.1 à 7.8 d'Elasticsearch	7 novembre 2025	7 novembre 2026
Elasticsearch versions 7.9	Non annoncé	Non annoncé
Elasticsearch versions 7.10	Non annoncé	Non annoncé

Le calendrier de fin du support pour OpenSearch les versions est le suivant :

Version du logiciel	Support standard	Support étendu
OpenSearch versions 1.0 à 1.2	7 novembre 2025	7 novembre 2026
OpenSearch versions 1.3	Non annoncé	Non annoncé

Version du logiciel	Support standard	Support étendu
OpenSearch versions 2.3 à 2.9	7 novembre 2025	7 novembre 2026
OpenSearch versions 2.11 et ultérieures	Non annoncé	Non annoncé

Support standard et support étendu d' OpenSearch Elasticsearch

AWS fournit régulièrement des corrections de bogues et des mises à jour de sécurité pour les versions couvertes par le Support Standard. Pour les versions bénéficiant d'un support étendu, AWS fournit des correctifs de sécurité critiques pendant une période d'au moins 12 mois après la fin du support standard, moyennant un montant forfaitaire supplémentaire pour chaque heure d'instance normalisée (NIH). Le NIH est calculé en fonction de la taille de l'instance (par exemple, moyenne, grande) et du nombre d'heures de l'instance (voir la section sur le calcul des frais de support étendu ci-dessous pour un exemple). Les frais de support étendu sont appliqués automatiquement lorsqu'un domaine exécute une version pour laquelle le support standard a pris fin. Vous pouvez effectuer une mise à niveau vers une version récente qui est toujours couverte par le support standard afin d'éviter des frais de support prolongés. Pour plus d'informations sur les frais d'assistance prolongée, consultez la page de tarification. Pour obtenir des informations générales sur le support étendu, consultez la FAQ sur le support étendu.

Calcul des frais de support étendu

Les domaines exécutant des versions bénéficiant d'un support étendu seront facturés un supplément forfaitairefee/Normalized Instance Hour (NIH), for example, $0.0065 in the US East (North Virginia) Region. NIH is computed as a factor of the instance size (e.g., medium, large), and the number of instance hours. For example, if you are running an m7g.medium.search instance for 24 hours in the US East (North Virginia) Region, which is priced at $0.068/Instance hour (on-demand), you will typically pay $1.632 ($0.068x24). If you are running a version that is in extended support, you will pay an additional $0.0065/NIH, calculé comme suit : 0,0065$ x 24 (nombre d'heures d'instance) x 2

(facteur de normalisation de la taille ; 2 pour les instances de taille moyenne), soit 0,312$ pour un support prolongé de 24 heures. Le montant total que vous paierez pendant 24 heures sera la somme du coût d'utilisation standard de l'instance et du coût du support étendu, soit 1,944$ (1,632 $+0,312 $). Le tableau ci-dessous indique le facteur de normalisation pour différentes tailles d'instance dans OpenSearch Service.

Taille d’instance	Facteur de normalisation
nano	0,25
micro	0,5
Xsmall	1
medium	2
large	4
xlarge	8
2xlarge	16
4xlarge	32
8xlarge	64
9xlarge	72
10xlarge	80
12xlarge	96
16xlarge	128
18xlarge	144
24xlarge	192
32xlarge	256

Tarification Amazon OpenSearch Service

Pour le OpenSearch Service, vous payez chaque heure d'utilisation d'une EC2 instance et selon la taille cumulée des volumes de stockage EBS associés à vos instances. Les frais AWS de transfert de données standard s'appliquent également.

Toutefois, il existe des exceptions notables en matière de transfert de données. Si un domaine utilise des zones de disponibilité multiples, OpenSearch Service ne facture pas le trafic entre elles. Une quantité significative de données sont transférées au sein d'un domaine lors des opérations d'allocation d'une partition et de rééquilibrage. OpenSearch Service ne mesure ni ne facture ce trafic. De même, OpenSearch Service ne facture pas le transfert de données entre UltraWarm/à froid nœuds et Amazon S3.

Pour obtenir des informations complètes sur la tarification, consultez Tarification Amazon OpenSearch Service. Pour plus d'informations sur les frais encourus durant les changements de configuration, reportez-vous à la section the section called “Frais associés aux modifications de la configuration”.

Services connexes

OpenSearch Service est généralement utilisé avec les services suivants :

Amazon CloudWatch

OpenSearch Les domaines de service envoient automatiquement des métriques pour que vous CloudWatch puissiez surveiller l'état et les performances du domaine. Pour de plus amples informations, veuillez consulter Surveillance des métriques OpenSearch du cluster avec Amazon CloudWatch.

CloudWatch Les journaux peuvent également opérer en sens inverse. Vous pouvez configurer CloudWatch Logs pour diffuser des données vers OpenSearch Service à des fins d'analyse. Pour en savoir plus, veuillez consulter la section the section called “Chargement de données de streaming depuis Amazon CloudWatch”.

AWS CloudTrail

Utilisez AWS CloudTrail pour obtenir un historique des appels d'API de configuration du OpenSearch service et des événements qui y sont liés pour votre compte. Pour de plus amples informations, veuillez consulter Surveillance des appels OpenSearch d'API Amazon Service avec AWS CloudTrail.

Amazon Kinesis

Kinesis est un service géré permettant de traiter en temps réel des données de streaming à très grande échelle. Pour plus d'informations, consultez the section called “Chargement de données de streaming à partir d'Amazon Kinesis Data Streams” et the section called “Chargement de données de streaming depuis Amazon Data Firehose”.

Amazon S3

Amazon Simple Storage Service (Amazon S3) fournit un stockage sur Internet. Ce guide fournit un exemple de code Lambda pour l'intégration à Amazon S3. Pour de plus amples informations, veuillez consulter the section called “Chargement de données de streaming à partir d'Amazon S3”.

AWS IAM

AWS Identity and Access Management (IAM) est un service web que vous pouvez utiliser pour gérer l'accès à vos domaines de OpenSearch service. Pour de plus amples informations, veuillez consulter the section called “Gestion de l'identité et des accès”.

AWS Lambda

AWS Lambda est un service informatique qui vous permet d'exécuter du code sans nécessiter le provisionnement ou la gestion des serveurs. Ce guide fournit un exemple de code Lambda pour diffuser des données à partir de DynamoDB, Amazon S3 et Kinesis. Pour de plus amples informations, veuillez consulter the section called “Chargement de données de streaming dans le OpenSearch service”.

Amazon DynamoDB

Amazon DynamoDB est un service de base de données NoSQL entièrement géré, offrant des performances exceptionnelles et prévisibles en termes de rapidité et d'évolutivité. Pour en savoir plus sur la diffusion de données vers OpenSearch Service, consultezthe section called “Chargement de données de streaming à partir d'Amazon DynamoDB”.

Amazon QuickSight

Vous pouvez visualiser les données de OpenSearch Service à l'aide de QuickSight tableaux de bord. Pour plus d'informations, consultez la section Utilisation d'Amazon OpenSearch Service with QuickSight dans le guide de QuickSight l'utilisateur.

> **Note**
>
> OpenSearch inclut certains codes Elasticsearch sous licence Apache d'Elasticsearch B.V. et d'autres codes sources. Elasticsearch B.V. n'est pas la source de cet autre code source. ELASTICSEARCH est une marque déposée d'Elasticsearch B.V.

Configuration d'Amazon OpenSearch Service

Accorder des autorisations

Dans les environnements de production, nous vous recommandons d'utiliser des stratégies plus précises. Pour en savoir plus sur la gestion des accès, consultez Gestion des accès pour les AWS ressources dans le Guide IAM de l'utilisateur.

Pour activer l’accès, ajoutez des autorisations à vos utilisateurs, groupes ou rôles :

- Utilisateurs et groupes dans AWS IAM Identity Center :

  Créez un jeu d’autorisations. Suivez les instructions de la rubrique Création d’un jeu d’autorisations du Guide de l’utilisateur AWS IAM Identity Center .
- Utilisateurs gérés dans IAM par un fournisseur d’identité :

  Créez un rôle pour la fédération d’identité. Suivez les instructions de la rubrique Création d’un rôle pour un fournisseur d’identité tiers (fédération) dans le Guide de l’utilisateur IAM.
- Utilisateurs IAM :
  - Créez un rôle que votre utilisateur peut assumer. Suivez les instructions de la rubrique Création d’un rôle pour un utilisateur IAM dans le Guide de l’utilisateur IAM.
  - (Non recommandé) Attachez une politique directement à un utilisateur ou ajoutez un utilisateur à un groupe d’utilisateurs. Suivez les instructions de la rubrique Ajout d’autorisations à un utilisateur (console) du Guide de l’utilisateur IAM.

Octroi d’un accès par programmation

Les utilisateurs ont besoin d'un accès programmatique s'ils souhaitent interagir avec AWS en dehors de la AWS Management Console. La manière d'accorder un accès programmatique dépend du type d'utilisateur qui accède à AWS.

Pour accorder aux utilisateurs un accès programmatique, choisissez l’une des options suivantes.

Quel utilisateur a besoin d'un accès programmatique ?	Pour	Par
Identité de la main-d'œuvre (Utilisateurs gérés dans IAM Identity Center)	Utilisez des informations d'identification temporaires pour signer les demandes programmatiques adressées au AWS CLI AWS SDKs, ou AWS APIs.	Suivez les instructions de l'interface que vous souhaitez utiliser. • Pour le AWS CLI, voir Configuration du AWS CLI à utiliser AWS IAM Identity Center dans le guide de AWS Command Line Interface l'utilisateur. • Pour AWS SDKs, outils, et AWS APIs, voir Authentification IAM Identity Center dans le guide de référence AWS SDKs et Tools.
IAM	Utilisez des informations d'identification temporaires pour signer les demandes programmatiques adressées au AWS CLI AWS SDKs, ou AWS APIs.	Suivez les instructions de la section Utilisation d'informations d'identification temporaires avec AWS les ressources du Guide de l'utilisateur IAM.
IAM	(Non recommandé) Utilisez des informations d'identification à long terme pour signer des demandes programmatiques adressées au AWS CLI AWS SDKs, ou AWS APIs.	Suivez les instructions de l'interface que vous souhaitez utiliser. • Pour le AWS CLI, voir Authentification à l'aide des informations d'identification utilisateur IAM dans le Guide de l'AWS Command Line Interface utilisateur.

Quel utilisateur a besoin d'un accès programmatique ?	Pour	Par
		• Pour les outils AWS SDKs et, voir Authentifier à l'aide d'informations d'identification à long terme dans le guide de référence des outils AWS SDKs et. • Pour AWS APIs, voir Gestion des clés d'accès pour les utilisateurs IAM dans le Guide de l'utilisateur IAM.

Installez et configurez AWS CLI

Si vous souhaitez utiliser le OpenSearch Service APIs, vous devez installer la dernière version du AWS Command Line Interface (AWS CLI). Vous n'avez pas AWS CLI besoin du OpenSearch service depuis la console et vous pouvez démarrer sans la CLI en suivant les étapes décrites dansMise en route avec Amazon OpenSearch Service.

Pour configurer le AWS CLI

1. Pour installer la dernière version de l' AWS CLI pour macOS, Linux ou Windows, consultez Installation ou mise à jour de la dernière version de la AWS CLI.
2. Pour configurer AWS CLI et sécuriser votre accès à Services AWS, y compris au OpenSearch service, voir Configuration rapide avec `aws configure`.
3. Pour vérifier la configuration de l', tapez la DataBrew commande suivante à l'invite de commande.

   ```
   aws opensearch help
   ```

   AWS CLI les commandes utilisent la valeur par défaut Région AWS de votre configuration, sauf si vous la définissez avec un paramètre ou un profil. Pour définir votre Région AWS paramètre, vous pouvez ajouter le `--region` paramètre à chaque commande.

Pour définir votre Région AWS profil, ajoutez d'abord un profil nommé dans le `~/.aws/config` fichier ou le `%UserProfile%/.aws/config` fichier (pour Microsoft Windows). Suivez les étapes décrites dans la section Profils nommés pour le AWS CLI. Ensuite, définissez vos paramètres Région AWS et les autres à l'aide d'une commande similaire à celle de l'exemple suivant.

```
[profile opensearch]
aws_access_key_id = ACCESS-KEY-ID-OF-IAM-USER
aws_secret_access_key = SECRET-ACCESS-KEY-ID-OF-IAM-USER
region = us-east-1
output = text
```

Ouvrez la console .

La plupart des rubriques de cette section consacrées à la console commencent par la console OpenSearch Service. Si vous n'êtes pas encore connecté à votre Compte AWS, connectez-vous, puis ouvrez la console OpenSearch Service et passez à la section suivante pour continuer avec OpenSearch Service.

Mise en route avec Amazon OpenSearch Service

Pour commencer, souscrivez à un Compte AWS si vous n'en n'avez pas déjà un. Après avoir configuré un compte, suivez le didacticiel Mise en route pour Amazon OpenSearch Service. Consultez les rubriques de présentation suivantes si vous avez besoin de plus d'informations quand vous apprenez à utiliser le service :

- Créez un domaine.
- Dimensionner le domaine en fonction de la charge de travail.
- Contrôlez l'accès à votre domaine à l'aide d'une politique d'accès au domaine ou d'un contrôle d'accès précis.
- Indexez les données manuellement ou à partir d'autres AWS services.
- Utilisez OpenSearch les tableaux de bord pour rechercher vos données et créer des visualisations.
- Découvrez des options de création de domaine plus avancées. Pour de plus amples informations, veuillez consulter Création et gestion des domaines.
- Apprenez à gérer les index de votre domaine. Pour de plus amples informations, veuillez consulter Gestion des index.
- Essayez un des tutoriels consacrés à l'utilisation d'Amazon OpenSearch Service. Pour de plus amples informations, veuillez consulter Didacticiels.

Pour plus d'informations sur la migration vers OpenSearch Service à partir d'un OpenSearch cluster autogéré, consultez. the section called “Migration vers le service OpenSearch ”

Pour des informations plus détaillées, consultez Création et gestion des domaines et les autres rubriques de ce guide. Pour en savoir plus amples sur la migration vers OpenSearch Service depuis un OpenSearch cluster autogéré, consultez la section. the section called “Migration vers le service OpenSearch ”

Vous pouvez compléter les étapes suivantes à l'aide de la console OpenSearch Service AWS CLI, de la ou des AWS kits SDK. Pour en savoir plus amples sur l'installation et la configuration de l'interface AWS CLI, consultez le Guide de l'AWS Command Line Interface utilisateur de l'.

Création d'un domaine Amazon OpenSearch Service

> **Important**
>
> Ce tutoriel concis explique comment configurer un domaine de test Amazon OpenSearch Service. N'utilisez pas ce processus pour créer des domaines de production. Pour une version complète du même processus, consultez Création et gestion des domaines.

Un domaine OpenSearch Service est synonyme de OpenSearch cluster. Les domaines sont des clusters qui contiennent les paramètres, les types d'instances, le nombre d'instances et les ressources de stockage que vous spécifiez. Vous pouvez créer un domaine OpenSearch de service à l'aide de la console, du AWS CLI, ou du AWS SDKs.

Créer un domaine de OpenSearch service à l'aide de la console

1. Accédez à la console https://aws.amazon.comet choisissez Se connecter à la console.
2. Sous Analytics, sélectionnez Amazon OpenSearch Service.
3. Choisissez Create domain (Créer un domaine).
4. Donnez un nom au domaine. Les exemples de ce tutoriel utilisent le nom movies.
5. Pour la méthode de création de domaine, choisissez Standard create.

   > **Note**
   >
   > Pour configurer rapidement un domaine de production conformément aux meilleures pratiques, vous pouvez choisir Easy create. Pour les besoins de développement et de test de ce didacticiel, nous utiliserons Standard Create.

6. Pour les modèles, choisissez Dev/Test.
7. Pour l'option de déploiement, choisissez Domain with standby.
8. Pour Version, choisissez la dernière version.
9. Pour le moment, ignorez les sections Nœuds de données, Stockage des données à chaud et à froid, Nœuds principaux dédiés, Configuration des instantanés et Point de terminaison personnalisé.
10. Par souci de simplicité dans le cadre de ce tutoriel, utilisez un domaine d'accès public. Sous Network (Réseau), choisissez Public Access (Accès public).

11. Dans les paramètres de contrôle précis des accès, maintenez la case Activer le contrôle précis des accès avec le contrôle précis des accès pour les paramètres de contrôle précis des accès. Sélectionnez Créer un utilisateur principal et entrez un nom d'utilisateur et un mot de passe.
12. Pour le moment, ignorez les sections Authentification SAML et Authentification Amazon Cognito.
13. Pour Access policy (Stratégie d'accès), choisissez Only use fine-grained access control (Utiliser uniquement le contrôle précis des accès). Dans ce didacticiel, le contrôle précis des accès gère l'authentification, pas la stratégie d'accès au domaine.
14. Ignorez les autres paramètres et choisissez Create (Créer). L'initialisation des nouveaux domaines prend généralement 15 à 30 minutes, mais cela peut prendre plus de temps en fonction de la configuration. Après l'initialisation de votre domaine, sélectionnez-le pour ouvrir son panneau de configuration. Notez le point de terminaison du domaine sous General information (Informations générales) (par exemple, `https://search-`*`my-domain`*`.us-east-1.es.amazonaws.com`), que vous utiliserez à l'étape suivante.

Suivant : télécharger des données vers un domaine de OpenSearch service à des fins d'indexation

Chargement de données vers Amazon OpenSearch Service pour les indexer

> **Important**
>
> Ce tutoriel concis explique comment charger une petite quantité de données de test dans Amazon OpenSearch Service. Pour plus d'informations sur le chargement de données dans un domaine de production, consultez Indexation des données.

Vous pouvez charger des données dans un domaine OpenSearch Service à l'aide de la ligne de commande ou de la plupart des langages de programmation.

Par commodité, les exemples de demande suivants utilisent curl (client HTTP courant). Les clients comme curl ne peuvent pas effectuer la signature de demande qui est exigée si vos stratégies d'accès spécifient des rôles ou utilisateurs IAM. Pour mener à bien ce processus, vous devez utiliser le contrôle précis des accès avec le nom d'utilisateur principal et le mot de passe que vous avez configurés à l'Étape 1.

Vous pouvez installer curl sous Windows et l'utiliser à partir de l'invite de commande. Toutefois, nous recommandons l'utilisation d'un outil comme Cygwin ou Windows Subsystem for Linux. curl est déjà préinstallé sur macOS et sur la plupart des distributions Linux.

Option 1 : Charger un seul document

Exécutez la commande suivante pour ajouter un document unique au domaine movies :

```
curl -XPUT -u 'master-user:master-user-password' 'domain-endpoint/movies/_doc/1' -d
 '{"director": "Burton, Tim", "genre": ["Comedy","Sci-Fi"], "year": 1996, "actor":
 ["Jack Nicholson","Pierce Brosnan","Sarah Jessica Parker"], "title": "Mars Attacks!"}'
 -H 'Content-Type: application/json'
```

Dans la commande, fournissez le nom utilisateur et le mot de passe que vous avez créé à l'Étape 1.

Pour obtenir une explication détaillée de cette commande et savoir comment envoyer des demandes signées au OpenSearch Service, consultezIndexation des données.

Option 2 : Charger plusieurs documents

Charger un fichier JSON qui contient plusieurs documents dans un domaine OpenSearch Service

1. Créez un fichier local nommé `bulk_movies.json`. Collez le contenu suivant dans le fichier et ajoutez une nouvelle ligne de fin :

   ```
   { "index" : { "_index": "movies", "_id" : "2" } }
   {"director": "Frankenheimer, John", "genre": ["Drama", "Mystery", "Thriller",
    "Crime"], "year": 1962, "actor": ["Lansbury, Angela", "Sinatra, Frank", "Leigh,
    Janet", "Harvey, Laurence", "Silva, Henry", "Frees, Paul", "Gregory, James",
    "Bissell, Whit", "McGiver, John", "Parrish, Leslie", "Edwards, James", "Flowers,
    Bess", "Dhiegh, Khigh", "Payne, Julie", "Kleeb, Helen", "Gray, Joe", "Nalder,
    Reggie", "Stevens, Bert", "Masters, Michael", "Lowell, Tom"], "title": "The
    Manchurian Candidate"}
   { "index" : { "_index": "movies", "_id" : "3" } }
   {"director": "Baird, Stuart", "genre": ["Action", "Crime", "Thriller"], "year":
    1998, "actor": ["Downey Jr., Robert", "Jones, Tommy Lee", "Snipes, Wesley",
    "Pantoliano, Joe", "Jacob, Ir\u00e8ne", "Nelligan, Kate", "Roebuck, Daniel",
    "Malahide, Patrick", "Richardson, LaTanya", "Wood, Tom", "Kosik, Thomas",
    "Stellate, Nick", "Minkoff, Robert", "Brown, Spitfire", "Foster, Reese",
    "Spielbauer, Bruce", "Mukherji, Kevin", "Cray, Ed", "Fordham, David", "Jett,
    Charlie"], "title": "U.S. Marshals"}
   ```

```
{ "index" : { "_index": "movies", "_id" : "4" } }
{"director": "Ray, Nicholas", "genre": ["Drama", "Romance"], "year": 1955, "actor":
 ["Hopper, Dennis", "Wood, Natalie", "Dean, James", "Mineo, Sal", "Backus, Jim",
 "Platt, Edward", "Ray, Nicholas", "Hopper, William", "Allen, Corey", "Birch,
 Paul", "Hudson, Rochelle", "Doran, Ann", "Hicks, Chuck", "Leigh, Nelson",
 "Williams, Robert", "Wessel, Dick", "Bryar, Paul", "Sessions, Almira", "McMahon,
 David", "Peters Jr., House"], "title": "Rebel Without a Cause"}
```

2. Exécutez la commande suivante dans le répertoire local où le fichier est stocké pour le charger dans le domaine movies :

```
curl -XPOST -u 'master-user:master-user-password' 'domain-endpoint/movies/_bulk' --
data-binary @bulk_movies.json -H 'Content-Type: application/x-ndjson'
```

Pour plus d'informations sur le format de fichier en bloc, consultez Indexation des données.

À suivre : Rechercher des documents

Rechercher des documents dans Amazon OpenSearch Service

Pour rechercher des documents dans un domaine Amazon OpenSearch Service, utilisez l'API OpenSearch de recherche. Sinon, vous pouvez utiliser les OpenSearch tableaux de bord pour rechercher des documents dans le domaine.

Rechercher des documents via la ligne de commande

Exécutez la commande suivante pour rechercher le mot mars dans le domaine movies :

```
curl -XGET -u 'master-user:master-user-password' 'domain-endpoint/movies/_search?
q=mars&pretty=true'
```

Si vous avez utilisé les données en bloc sur la page précédente, essayez de rechercher rebelle à la place.

La réponse devrait être similaire à ce qui suit :

```
{
  "took" : 5,
```

```
  "timed_out" : false,
  "_shards" : {
    "total" : 5,
    "successful" : 5,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 1,
      "relation" : "eq"
    },
    "max_score" : 0.2876821,
    "hits" : [
      {
        "_index" : "movies",
        "_type" : "_doc",
        "_id" : "1",
        "_score" : 0.2876821,
        "_source" : {
          "director" : "Burton, Tim",
          "genre" : [
            "Comedy",
            "Sci-Fi"
          ],
          "year" : 1996,
          "actor" : [
            "Jack Nicholson",
            "Pierce Brosnan",
            "Sarah Jessica Parker"
          ],
          "title" : "Mars Attacks!"
        }
      }
    ]
  }
}
```

Rechercher des documents à l'aide de OpenSearch tableaux de bord

OpenSearch Les tableaux de bord sont un outil de visualisation open source populaire conçu pour fonctionner avec OpenSearch. Ils fournissent une interface utilisateur utile pour la recherche et la surveillance de vos indices.

Rechercher des documents depuis un domaine OpenSearch Service à l'aide de tableaux de bord

1. Accédez à l'URL OpenSearch des tableaux de bord correspondant à votre domaine. Vous pouvez trouver l'URL dans le tableau de bord du domaine dans la console OpenSearch Service. Le format de l'URL est le suivant :

   ```
   domain-endpoint/_dashboards/
   ```
2. Connectez-vous à l'aide de votre nom d'utilisateur et de votre mot de passe principaux.
3. Pour utiliser Dashboards, vous devez créer au moins un modèle d'index. Dashboards utilise ces modèles pour identifier les index que vous voulez analyser. Ouvrez le panneau de navigation de gauche, choisissez Stack Management (Gestion des piles), choisissez Index Patterns (Modèles d'index), puis choisissez Create index pattern (Créer un modèle d'index). Dans le cadre de ce tutoriel, saisissez movies.
4. Choisissez Next step (Étape suivante), puis Create index pattern (Créer un modèle d'index). Une fois le modèle créé, vous pouvez consulter les différents champs du document, comme `actor` et `director`.
5. Retournez à la page Index Patterns (Modèles d'index) et vérifiez que `movies` est défini comme modèle par défaut. Si ce n'est pas le cas, sélectionnez le modèle et choisissez l'icône en forme d'étoile pour en faire le modèle par défaut.
6. Pour commencer à rechercher vos données, ouvrez à nouveau le panneau de navigation de gauche et choisissez Discover (Découvrir).
7. Sur la barre de recherche, saisissez mars si vous avez chargé un seul document, ou rebelle si vous avez chargé plusieurs documents, puis appuyez sur Entrée. Vous pouvez essayer de rechercher d'autres termes, tels que des noms d'acteurs ou de réalisateurs.

À suivre : Supprimer un domaine

Suppression d'un domaine Amazon OpenSearch Service

Étant donné que le domaine movies est utilisé dans ce tutoriel à des fins de test, veillez à le supprimer lorsque vous aurez fini de l'utiliser afin de ne pas payer de frais.

Supprimer un domaine OpenSearch Service de la console

1. Connectez-vous à la console Amazon OpenSearch Service.
2. Sous Domains (Domaines), sélectionnez le domaine movies.

3. Choisissez Delete (Supprimer) et confirmez la suppression.

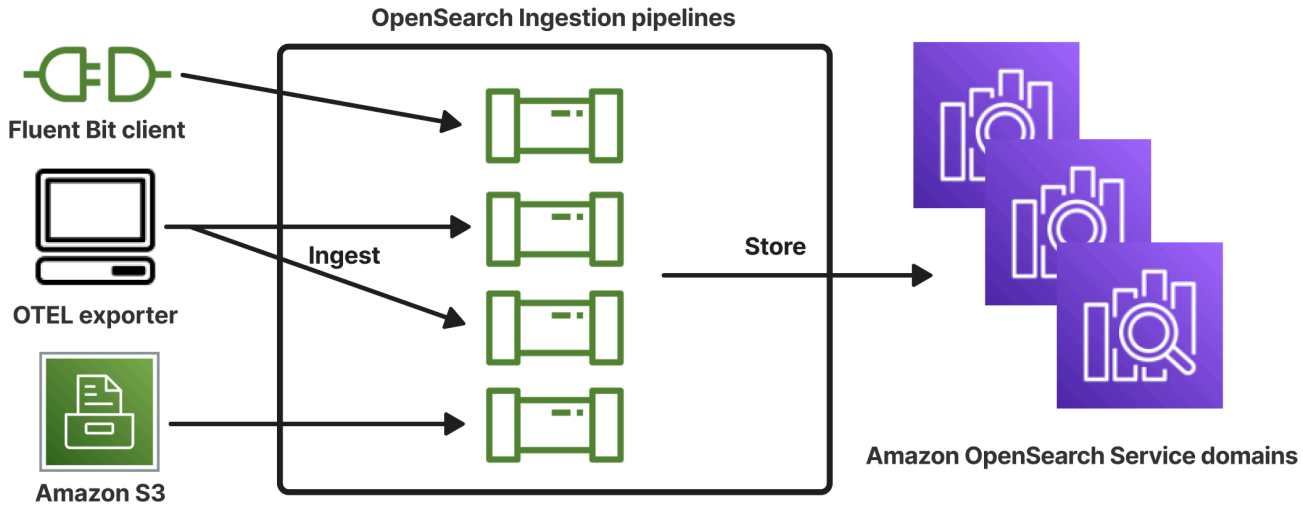
Présentation d'Amazon OpenSearch Ingestion

Amazon OpenSearch Ingestion est un collecteur de données sans serveur entièrement géré qui diffuse des journaux, des métriques et des données de suivi en temps réel vers les domaines Amazon OpenSearch Service et les collections OpenSearch sans serveur.

Avec OpenSearch Ingestion, vous n'avez plus besoin d'outils tiers tels que Logstash ou Jaeger pour ingérer des données. Vous configurez vos producteurs de données pour qu'ils envoient des données OpenSearch à Ingestion, qui les envoie automatiquement au domaine ou à la collection que vous avez spécifiés. Vous pouvez également transformer les données avant leur livraison.

Comme OpenSearch Ingestion fonctionne sans serveur, vous n'avez pas à gérer l'infrastructure, à appliquer des correctifs logiciels ou à dimensionner les clusters manuellement. Vous pouvez configurer des pipelines d'ingestion directement dans le AWS Management Console, et OpenSearch Ingestion s'occupe du reste.

En tant que composant d'Amazon OpenSearch Service, OpenSearch Ingestion repose sur Data Prepper, un collecteur de données open source qui filtre, enrichit, transforme, normalise et agrège les données pour les analyser et les visualiser en aval.



Concepts clés d'Amazon OpenSearch Ingestion

Avant de commencer à utiliser OpenSearch Ingestion, il est utile de comprendre ces concepts clés.

Pipeline

Du point de vue de OpenSearch l'ingestion, un pipeline fait référence à un seul collecteur de données provisionné que vous créez dans OpenSearch Service. Vous pouvez le considérer comme le fichier de configuration YAML complet, qui inclut un ou plusieurs sous-pipelines. Pour connaître les étapes de création d'un pipeline d'ingestion, voirthe section called “Création de pipelines”.

Sous-pipeline

Vous définissez des sous-pipelines dans un fichier de configuration YAML. Chaque sous-pipeline est une combinaison d'une source, d'une mémoire tampon, de zéro ou plusieurs processeurs et d'un ou plusieurs récepteurs. Vous pouvez définir plusieurs sous-pipelines dans un seul fichier YAML, chacun avec des sources, des processeurs et des récepteurs uniques. Pour faciliter la surveillance avec CloudWatch les autres services, nous vous recommandons de spécifier un nom de pipeline distinct de tous ses sous-pipelines.

Vous pouvez enchaîner plusieurs sous-pipelines dans un même fichier YAML, de telle sorte que la source d'un sous-pipeline soit un autre sous-pipeline et que son récepteur soit un troisième sous-pipeline. Pour obtenir un exemple, consultez the section called “OpenTelemetry Collectionneur”.

Source

Composant d'entrée d'un sous-pipeline. Il définit le mécanisme par lequel un pipeline consomme des enregistrements. La source peut consommer des événements soit en les recevant via HTTPS, soit en les lisant depuis des points de terminaison externes tels qu'Amazon S3. Il existe deux types de sources : celles basées sur le push et celles basées sur le pull. Les sources basées sur le push, telles que HTTP et les OTel journaux, transmettent les enregistrements aux points de terminaison d'ingestion. Les sources basées sur le pull, telles que OTel trace et S3, extraient les données de la source.

Processors

Unités de traitement intermédiaires capables de filtrer, transformer et enrichir les enregistrements dans le format souhaité avant de les publier dans le récepteur. Le processeur est un composant optionnel d'un pipeline. Si vous ne définissez pas de processeur, les enregistrements sont publiés dans le format défini dans la source. Vous pouvez avoir plusieurs processeurs. Un pipeline exécute les processeurs dans l'ordre dans lequel vous les définissez.

Sink

Composant de sortie d'un sous-pipeline. Il définit une ou plusieurs destinations vers lesquelles un sous-pipeline publie des enregistrements. OpenSearch L'ingestion prend en charge les domaines de OpenSearch service en tant que récepteurs. Il prend également en charge les sous-pipelines en tant que puits. Cela signifie que vous pouvez enchaîner plusieurs sous-pipelines au sein d'un même pipeline d' OpenSearch ingestion (fichier YAML). OpenSearch Les clusters autogérés ne sont pas pris en charge en tant que récepteurs.

Buffer

Partie d'un processeur qui fait office de couche entre la source et le récepteur. Vous ne pouvez pas configurer manuellement une mémoire tampon dans votre pipeline. OpenSearch L'ingestion utilise une configuration de tampon par défaut.

Acheminement

Partie d'un processeur qui permet aux auteurs de pipelines d'envoyer uniquement des événements répondant à certaines conditions à différents récepteurs.

Une définition de sous-pipeline valide doit contenir une source et un récepteur. Pour plus d'informations sur chacun de ces éléments de pipeline, consultez la référence de configuration.

Avantages d'Amazon OpenSearch Ingestion

OpenSearch L'ingestion présente les principaux avantages suivants :

- Vous n'avez plus besoin de gérer manuellement un pipeline auto-provisionné.
- Adapte automatiquement vos pipelines en fonction des limites de capacité que vous définissez.
- Maintient votre pipeline à jour grâce à des correctifs de sécurité et à des correctifs de bogues.
- Permet de connecter des pipelines à votre cloud privé virtuel (VPC) pour une couche de sécurité supplémentaire.
- Vous permet d'arrêter et de démarrer des pipelines afin de contrôler les coûts.
- Fournit des plans de configuration de pipeline pour les cas d'utilisation les plus courants afin de vous aider à être opérationnel plus rapidement.
- Vous permet d'interagir par programmation avec vos pipelines via les différents AWS SDKs et l'API d' OpenSearch ingestion.

- Prend en charge le suivi des performances sur Amazon CloudWatch et la journalisation des erreurs dans CloudWatch Logs.

Limites d'Amazon OpenSearch Ingestion

OpenSearch L'ingestion présente les limites suivantes :

- Vous ne pouvez ingérer des données que dans des domaines exécutant la OpenSearch version 1.0 ou ultérieure, ou Elasticsearch 6.8 ou version ultérieure. Si vous utilisez la source de OTel suivi, nous vous recommandons d'utiliser Elasticsearch 7.9 ou version ultérieure afin de pouvoir utiliser le plug-in OpenSearch Dashboards.
- Si un pipeline écrit dans un domaine de OpenSearch service situé au sein d'un VPC, le pipeline doit être créé au même endroit Région AWS que le domaine.
- Vous ne pouvez configurer qu'une seule source de données dans une définition de pipeline.
- Vous ne pouvez pas spécifier de OpenSearch clusters autogérés comme récepteurs.
- Vous ne pouvez pas spécifier un point de terminaison personnalisé en tant que récepteur. Vous pouvez toujours écrire sur un domaine pour lequel les points de terminaison personnalisés sont activés, mais vous devez spécifier son point de terminaison standard.
- Vous ne pouvez pas spécifier de ressources dans les régions optionnelles en tant que sources ou récepteurs.
- Certaines contraintes s'appliquent aux paramètres que vous pouvez inclure dans une configuration de pipeline. Pour de plus amples informations, veuillez consulter the section called “Exigences et contraintes de configuration”.

Versions de Data Prepper prises en charge

OpenSearch Ingestion prend actuellement en charge les versions principales suivantes de Data Prepper :

- 2.x

Lorsque vous créez un pipeline à l'aide de l'éditeur de code, utilisez l'`version`option requise pour spécifier la version principale de Data Prepper à utiliser. Par exemple,`version: "2"`. OpenSearch Ingestion récupère la dernière version mineure prise en charge de cette version majeure et approvisionne le pipeline avec cette version.

Si vous n'utilisez pas l'éditeur de code pour créer votre pipeline, OpenSearch Ingestion approvisionne automatiquement votre pipeline avec la dernière version prise en charge.

Actuellement, OpenSearch Ingestion approvisionne les pipelines avec la version 2.7 de Data Prepper. Pour plus d'informations, consultez les notes de mise à jour de la version 2.7. Toutes les versions mineures d'une version majeure particulière ne sont pas prises en charge par OpenSearch Ingestion.

Lorsque vous mettez à jour la configuration d'un pipeline, si une nouvelle version mineure de Data Prepper est prise en charge, OpenSearch Ingestion met automatiquement à niveau le pipeline vers la dernière version mineure prise en charge de la version majeure spécifiée dans la configuration du pipeline. Par exemple, vous avez `version: "2"` peut-être intégré la configuration de votre pipeline et OpenSearch Ingestion a initialement provisionné le pipeline avec la version 2.6.0. Lorsque la prise en charge de la version 2.7.0 est ajoutée et que vous modifiez la configuration du pipeline, OpenSearch Ingestion met à niveau le pipeline vers la version 2.7.0. Ce processus permet de maintenir votre pipeline à jour avec les dernières corrections de bogues et améliorations de performances. OpenSearch Ingestion ne peut pas mettre à jour la version principale de votre pipeline à moins que vous ne changiez manuellement l'`version`option dans la configuration du pipeline. Pour de plus amples informations, veuillez consulter the section called “Mise à jour des pipelines”.

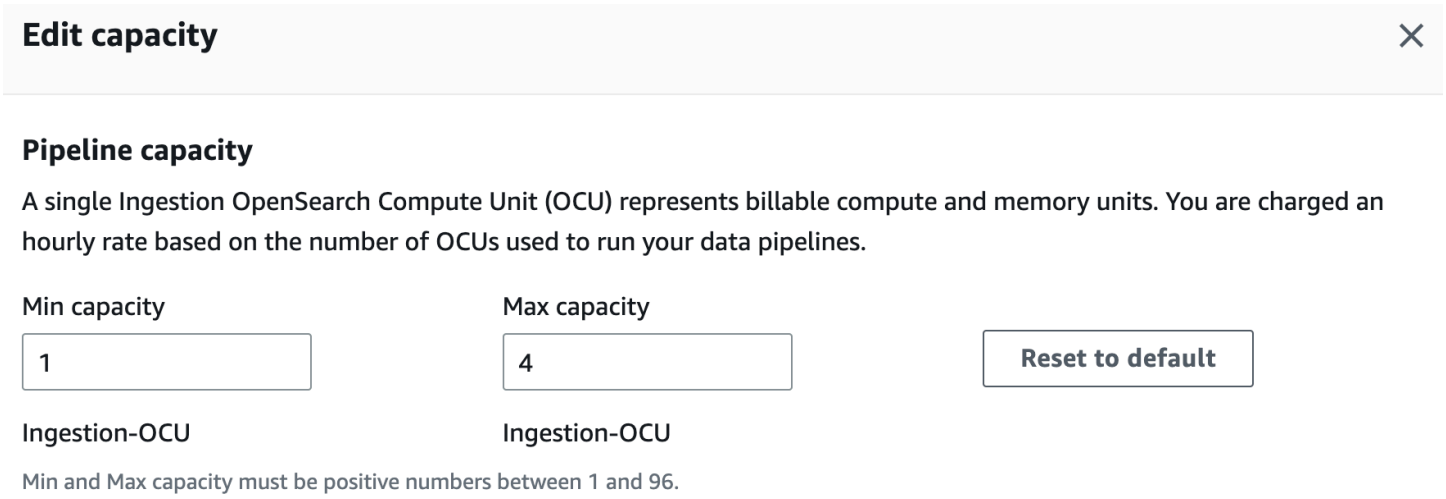
Dimensionnement des pipelines dans Amazon OpenSearch Ingestion

OpenSearch L'ingestion adapte automatiquement la capacité du pipeline en fonction des unités de OpenSearch calcul d'ingestion minimales et maximales spécifiées (ingestion OCUs). Cela élimine le besoin de provisionnement et de gestion manuels.

Chaque OCU d'ingestion est une combinaison d'environ 15 GiB de mémoire et de 2 V. CPUs Vous pouvez spécifier les valeurs OCU minimales et maximales pour un pipeline, et OpenSearch Ingestion adapte automatiquement la capacité de votre pipeline en fonction de ces limites.

Vous spécifiez les valeurs suivantes lorsque vous créez un pipeline :

- Capacité minimale — Le pipeline peut réduire sa capacité jusqu'à ce nombre d'ingestion OCUs. La capacité minimale spécifiée est également la capacité de départ d'un pipeline.
- Capacité maximale — Le pipeline peut augmenter sa capacité jusqu'à ce nombre d'ingestion OCUs.



Assurez-vous que la capacité maximale d'un pipeline est suffisamment élevée pour faire face aux pics de charge de travail, et que la capacité minimale est suffisamment faible pour minimiser les coûts lorsque le pipeline n'est pas occupé. En fonction de vos paramètres, OpenSearch Ingestion ajuste automatiquement le nombre d'ingestion OCUs pour votre pipeline afin de traiter la charge de travail d'ingestion. À un moment donné, vous n'êtes facturé OCUs que pour l'ingestion activement utilisée par votre pipeline.

La capacité allouée à votre pipeline d' OpenSearch ingestion augmente ou diminue en fonction des exigences de traitement de votre pipeline et de la charge générée par votre application client. Lorsque la capacité est limitée, OpenSearch Ingestion augmente en allouant davantage d'unités de calcul (GiB de mémoire). Lorsque votre pipeline traite de petites charges de travail ou ne traite pas de données du tout, il peut être réduit au minimum configuré pour l'ingestion OCUs.

Vous pouvez spécifier un minimum de 1 OCU d'ingestion, un maximum de 96 ingestion OCUs pour les pipelines apatrides et un maximum de 48 ingestion OCUs pour les pipelines statiques. Nous recommandons un minimum de 2 ingérations OCUs pour les sources basées sur le push. Lorsque la mise en mémoire tampon persistante est activée, vous pouvez spécifier un minimum de 2 et un maximum de 384 ingestions. OCUs

Avec un pipeline de log standard avec une source unique, un modèle de grok simple et un récepteur, chaque unité de calcul peut supporter jusqu'à 2 MiB par seconde. Pour les pipelines de journaux plus complexes comportant plusieurs processeurs, chaque unité de calcul peut supporter une charge d'ingestion moindre. Sur la base de la capacité du pipeline et de l'utilisation des ressources, le processus de mise à OpenSearch l'échelle de l'ingestion démarre.

Pour garantir une haute disponibilité, OCUs les ingestions sont réparties entre les zones de disponibilité (AZs). Le nombre de AZs dépend de la capacité minimale que vous spécifiez.

Par exemple, si vous spécifiez un minimum de 2 unités de calcul, OCUs l'ingestion utilisée à un moment donné est répartie uniformément sur 2 AZs. Si vous spécifiez un minimum de 3 unités de calcul ou plus, l'ingestion OCUs est répartie uniformément sur 3 AZs. Nous vous recommandons de configurer au moins deux processus d'ingestion OCUs afin de garantir une disponibilité de 99,9 % pour vos pipelines d'ingestion.

L'ingestion ne vous est pas facturée OCUs lorsqu'un pipeline se trouve dans les `Stopped` états `Create failed`,`Creating`,`Deleting`, et.

Pour obtenir des instructions sur la configuration et la récupération des paramètres de capacité d'un pipeline, consultezthe section called “Création de pipelines”.

OpenSearch Prix d'ingestion

À un moment donné, vous ne payez OCUs que pour le nombre d'ingestion allouées à un pipeline, que des données circulent ou non dans le pipeline. OpenSearch L'ingestion s'adapte immédiatement à vos charges de travail en augmentant ou en diminuant la capacité du pipeline en fonction de l'utilisation.

Pour en savoir plus sur les tarifs, consultez les tarifs d'Amazon OpenSearch Service.

Soutenu Régions AWS

OpenSearch L'ingestion est disponible dans un sous-ensemble de Régions AWS ce OpenSearch service disponible dans. Pour obtenir la liste des régions prises en charge, consultez la section Points OpenSearch de terminaison et quotas Amazon Service dans le Références générales AWS.

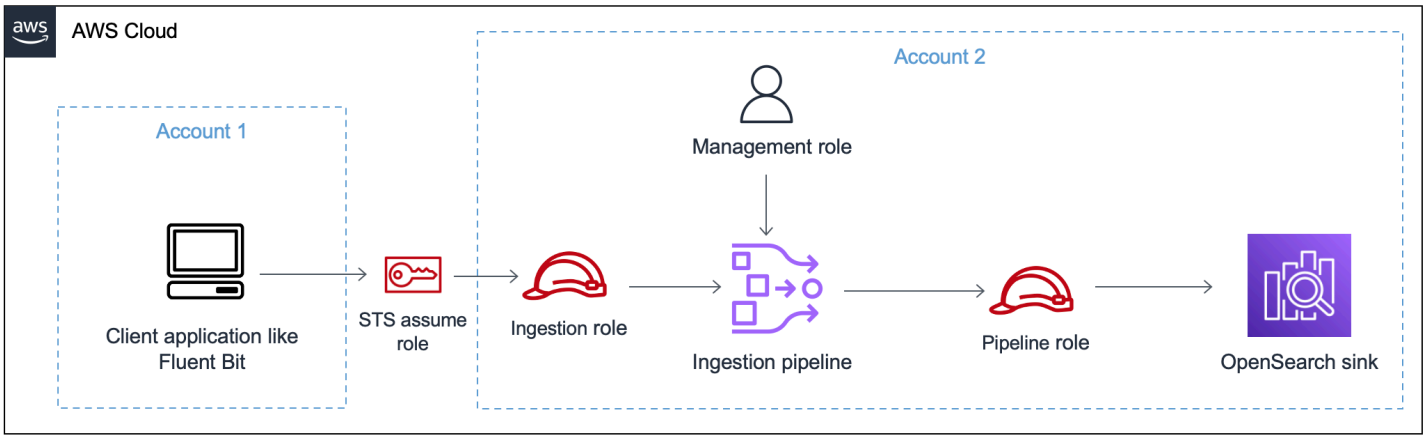
Configuration des rôles et des utilisateurs dans Amazon OpenSearch Ingestion

Amazon OpenSearch Ingestion utilise une variété de modèles d'autorisations et de rôles IAM afin de permettre aux applications source d'écrire dans des pipelines et aux pipelines d'écrire dans des récepteurs. Avant de commencer à ingérer des données, vous devez créer un ou plusieurs rôles IAM dotés d'autorisations spécifiques en fonction de votre cas d'utilisation.

Au minimum, les rôles suivants sont requis pour configurer un pipeline réussi.

Name (Nom)	Description
Rôle du pipeline	Le rôle de pipeline fournit les autorisations requises pour qu'un pipeline puisse lire à partir de la source et écrire dans le domaine ou le récepteur de collection. Vous pouvez créer manuellement le rôle de pipeline ou demander à OpenSearch Ingestion de le créer pour vous.
Rôle d'ingestion	Le rôle d'ingestion contient l'`osis:Ingest` autorisation pour la ressource du pipeline. Cette autorisation permet aux sources basées sur le push d'ingérer des données dans un pipeline.

L'image suivante illustre une configuration de pipeline typique, dans laquelle une source de données telle qu'Amazon S3 ou Fluent Bit écrit dans un pipeline d'un autre compte. Dans ce cas, le client doit assumer le rôle d'ingestion pour accéder au pipeline. Pour de plus amples informations, veuillez consulter the section called “Ingestion entre comptes”.



Pour un guide de configuration simple, voirthe section called “Tutoriel : Ingérer des données dans un domaine”.

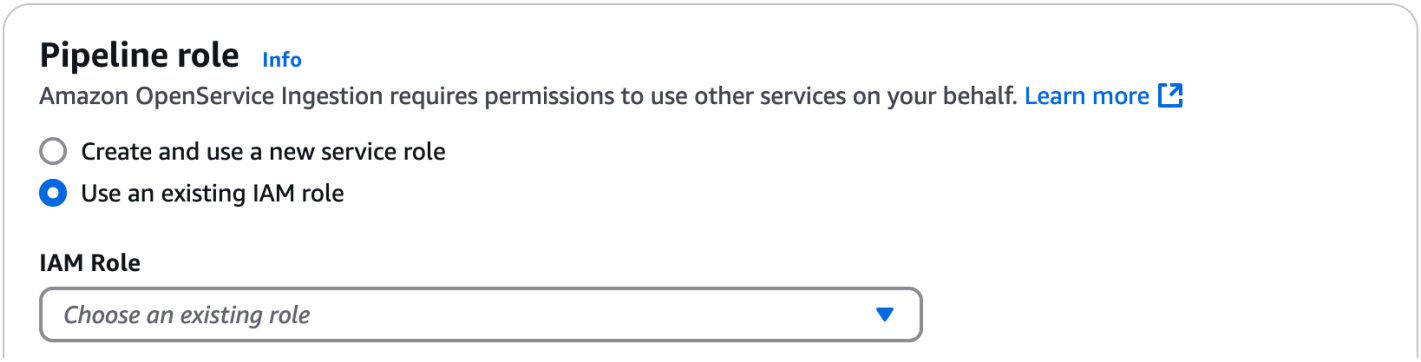
Rubriques

- the section called “Rôle du pipeline”
- the section called “Rôle d'ingestion”
- the section called “Ingestion entre comptes”

Rôle du pipeline

Un pipeline a besoin de certaines autorisations pour lire à partir de sa source et écrire dans son récepteur. Ces autorisations dépendent de l'application cliente ou de l'application cliente Service AWS qui écrit dans le pipeline, et du fait que le récepteur est un domaine de OpenSearch service, une collection OpenSearch sans serveur ou Amazon S3. En outre, un pipeline peut avoir besoin d'autorisations pour extraire physiquement les données de l'application source (si la source est un plugin basé sur le pull) et d'autorisations pour écrire dans une file d'attente de lettres mortes S3, si cette option est activée.

Lorsque vous créez un pipeline, vous avez la possibilité de spécifier un rôle IAM existant que vous avez créé manuellement, ou de demander à Ingestion de OpenSearch créer automatiquement le rôle de pipeline en fonction de la source et du récepteur que vous avez sélectionnés. L'image suivante montre comment spécifier le rôle du pipeline dans le AWS Management Console.



Rubriques

- Automatisation de la création de rôles dans le pipeline
- Création manuelle du rôle de pipeline

Automatisation de la création de rôles dans le pipeline

Vous pouvez demander à OpenSearch Ingestion de créer le rôle de pipeline pour vous. Il identifie automatiquement les autorisations requises par le rôle en fonction de la source et des récepteurs configurés. Il crée un rôle IAM avec le préfixe et `OpenSearchIngestion-` le suffixe que vous entrez. Par exemple, si vous entrez `PipelineRole` le suffixe, OpenSearch Ingestion crée un rôle nommé`OpenSearchIngestion-PipelineRole`.

La création automatique du rôle de pipeline simplifie le processus de configuration et réduit le risque d'erreurs de configuration. En automatisant la création des rôles, vous pouvez éviter d'attribuer

manuellement des autorisations, en veillant à ce que les bonnes politiques soient appliquées sans risquer de créer des erreurs de configuration en matière de sécurité. Cela permet également de gagner du temps et d'améliorer la conformité en matière de sécurité en appliquant les meilleures pratiques, tout en garantissant la cohérence entre les multiples déploiements de pipelines.

OpenSearch Ingestion ne peut créer automatiquement le rôle de pipeline que dans le AWS Management Console. Si vous utilisez l'API AWS CLI d' OpenSearch ingestion ou l'une des API SDKs, vous devez spécifier un rôle de pipeline créé manuellement.

Pour qu' OpenSearch Ingestion crée le rôle pour vous, sélectionnez Créer et utilisez un nouveau rôle de service.

> **Important**
>
> Vous devez toujours modifier manuellement la politique d'accès au domaine ou à la collection pour accorder l'accès au rôle de pipeline. Pour les domaines qui utilisent un contrôle d'accès précis, vous devez également associer le rôle de pipeline à un rôle de backend. Vous pouvez effectuer ces étapes avant ou après avoir créé le pipeline.
> Pour obtenir des instructions, consultez les rubriques suivantes :
>
> - Configuration de l'accès aux données pour le domaine
> - Configuration des données et de l'accès au réseau pour la collecte

Création manuelle du rôle de pipeline

Vous préférerez peut-être créer manuellement le rôle de pipeline si vous avez besoin de mieux contrôler les autorisations pour répondre à des exigences de sécurité ou de conformité spécifiques. La création manuelle vous permet d'adapter les rôles en fonction de l'infrastructure existante ou des stratégies de gestion des accès. Vous pouvez également choisir une configuration manuelle pour intégrer le rôle à d'autres Services AWS ou pour vous assurer qu'il correspond à vos besoins opérationnels uniques.

Pour choisir un rôle de pipeline créé manuellement, sélectionnez Utiliser un rôle IAM existant, puis choisissez un rôle existant. Le rôle doit disposer de toutes les autorisations nécessaires pour recevoir les données de la source sélectionnée et écrire sur le récepteur sélectionné. Les sections suivantes expliquent comment créer manuellement un rôle de pipeline.

Rubriques

- Autorisations de lecture depuis une source
- Autorisations d'écriture dans un récepteur de domaine
- Autorisations d'écriture dans un réservoir de collection
- Autorisations d'écriture sur Amazon S3 ou dans une file d'attente de lettres mortes

Autorisations de lecture depuis une source

Un pipeline d' OpenSearch ingestion doit être autorisé pour lire et recevoir des données provenant de la source spécifiée. Par exemple, pour une source Amazon DynamoDB, elle a besoin d'autorisations telles que et. `dynamodb:DescribeTable` `dynamodb:DescribeStream` Pour des exemples de politiques d'accès aux rôles du pipeline pour des sources courantes, telles qu'Amazon S3, Fluent Bit et le OpenTelemetry Collector, consultezthe section called “Intégrer les pipelines”.

Autorisations d'écriture dans un récepteur de domaine

Un pipeline d' OpenSearch ingestion a besoin d'une autorisation pour écrire dans un domaine de OpenSearch service configuré comme récepteur. Ces autorisations incluent la possibilité de décrire le domaine et de lui envoyer des requêtes HTTP. Ces autorisations sont les mêmes pour les domaines publics et VPC. Pour obtenir des instructions permettant de créer un rôle de pipeline et de le spécifier dans la politique d'accès au domaine, voir Autoriser les pipelines à accéder aux domaines.

Autorisations d'écriture dans un réservoir de collection

Un pipeline d' OpenSearch ingestion a besoin d'une autorisation pour écrire dans une collection OpenSearch sans serveur configurée comme récepteur. Ces autorisations incluent la possibilité de décrire la collection et de lui envoyer des requêtes HTTP.

Tout d'abord, assurez-vous que votre politique d'accès aux rôles dans le pipeline accorde les autorisations requises. Incluez ensuite ce rôle dans une politique d'accès aux données et accordez-lui les autorisations nécessaires pour créer des index, mettre à jour des index, décrire des index et rédiger des documents au sein de la collection. Pour obtenir des instructions pour effectuer chacune de ces étapes, voir Autoriser les pipelines à accéder aux collections.

Autorisations d'écriture sur Amazon S3 ou dans une file d'attente de lettres mortes

Si vous spécifiez Amazon S3 comme destination réceptrice pour votre pipeline, ou si vous activez une file d'attente de lettres mortes (DLQ), le rôle de pipeline doit lui permettre d'accéder au compartiment S3 que vous spécifiez comme destination.

Associez une politique d'autorisation distincte au rôle de pipeline qui fournit l'accès au DLQ. Au minimum, le rôle doit se voir attribuer l'`S3:PutObject`action sur la ressource du bucket :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "WriteToS3DLQ",
      "Effect": "Allow",
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::my-dlq-bucket/*"
    }
  ]
}
```

Rôle d'ingestion

Le rôle d'ingestion est un rôle IAM qui permet aux services externes d'interagir et d'envoyer des données en toute sécurité à un pipeline d' OpenSearch ingestion. Pour les sources basées sur le push, telles qu'Amazon Security Lake, ce rôle doit accorder des autorisations pour transférer des données dans le pipeline, notammento`sis:Ingest`. Pour les sources basées sur le pull, comme Amazon S3, le rôle doit permettre à OpenSearch Ingestion de l'assumer et d'accéder aux données avec les autorisations nécessaires.

Rubriques

- Rôle d'ingestion pour les sources basées sur le push
- Rôle de l'ingestion pour les sources basées sur l'extraction
- Ingestion entre comptes

Rôle d'ingestion pour les sources basées sur le push

Pour les sources basées sur le push, les données sont envoyées ou poussées vers le pipeline d'ingestion depuis un autre service, tel qu'Amazon Security Lake ou Amazon DynamoDB. Dans ce scénario, le rôle d'ingestion doit au minimum être `osis:Ingest` autorisé à interagir avec le pipeline.

La politique d'accès IAM suivante explique comment accorder cette autorisation au rôle d'ingestion :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "osis:Ingest"
      ],
      "Resource": "arn:aws:osis:region:account-id:pipeline/pipeline-name/*"
    }
  ]
}
```

Rôle de l'ingestion pour les sources basées sur l'extraction

Pour les sources basées sur l'extraction, le pipeline OpenSearch d'ingestion extrait ou récupère activement les données d'une source externe, telle qu'Amazon S3. Dans ce cas, le pipeline doit assumer un rôle de pipeline IAM qui accorde les autorisations nécessaires pour accéder à la source de données. Dans ces scénarios, le rôle d'ingestion est synonyme de rôle de pipeline.

Le rôle doit inclure une relation de confiance permettant à OpenSearch Ingestion de l'assumer, ainsi que des autorisations spécifiques à la source de données. Pour de plus amples informations, veuillez consulter the section called “Autorisations de lecture depuis une source”.

Ingestion entre comptes

Il se peut que vous deviez ingérer des données dans un pipeline à partir d'un autre Compte AWS système, tel qu'un compte d'application. Pour configurer l'ingestion entre comptes, définissez un rôle d'ingestion au sein du même compte que le pipeline et établissez une relation de confiance entre le rôle d'ingestion et le compte d'application :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
     "Effect": "Allow",
```

```
      "Principal": {
        "AWS": "arn:aws:iam::external-account-id:root"
       },
      "Action": "sts:AssumeRole"
  }]
}
```

Configurez ensuite votre application pour qu'elle assume le rôle d'ingestion. Le compte d'application doit accorder au rôle d'application AssumeRoledes autorisations pour le rôle d'ingestion dans le compte de pipeline.

Pour obtenir des étapes détaillées et des exemples de politiques IAM, consultezthe section called “Fournir un accès à l'ingestion entre comptes”.

Accorder aux OpenSearch pipelines Amazon Ingestion l'accès aux domaines

Un pipeline OpenSearch Amazon Ingestion a besoin d'une autorisation pour écrire dans le domaine de OpenSearch service configuré comme récepteur. Pour fournir un accès, vous configurez un rôle AWS Identity and Access Management (IAM) avec une politique d'autorisations restrictive qui limite l'accès au domaine auquel un pipeline envoie des données. Par exemple, vous souhaiterez peut-être limiter un pipeline d'ingestion au seul domaine et aux index nécessaires pour prendre en charge son cas d'utilisation.

> **Important**
>
> Vous pouvez choisir de créer manuellement le rôle de pipeline, ou vous pouvez demander à Ingestion de OpenSearch le créer pour vous lors de la création du pipeline. Si vous choisissez la création automatique des rôles, OpenSearch Ingestion ajoute toutes les autorisations requises à la politique d'accès aux rôles du pipeline en fonction de la source et du récepteur que vous choisissez. Il crée un rôle de pipeline dans IAM avec le préfixe `OpenSearchIngestion-` et le suffixe que vous entrez. Pour de plus amples informations, veuillez consulter the section called “Rôle du pipeline”.
>
> Si OpenSearch Ingestion crée le rôle de pipeline pour vous, vous devez tout de même inclure le rôle dans la politique d'accès au domaine et le mapper à un rôle principal (si le domaine utilise un contrôle d'accès précis), avant ou après la création du pipeline. Consultez l'étape 2 pour obtenir des instructions.

Rubriques

- Étape 1 : Création du rôle de pipeline
- Étape 2 : configurer l'accès aux données pour le domaine

Étape 1 : Création du rôle de pipeline

Le rôle de pipeline doit être associé à une politique d'autorisation lui permettant d'envoyer des données au récepteur de domaine. Il doit également avoir une relation de confiance permettant à OpenSearch Ingestion d'assumer le rôle. Pour savoir comment associer une politique à un rôle, consultez la section Ajout d'autorisations d'identité IAM dans le Guide de l'utilisateur IAM.

L'exemple de politique suivant illustre le privilège minimal que vous pouvez accorder à un rôle de pipeline pour qu'il puisse écrire sur un seul domaine :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "es:DescribeDomain",
            "Resource": "arn:aws:es:*:account-id:domain/*"
        },
        {
            "Effect": "Allow",
            "Action": "es:ESHttp*",
            "Resource": "arn:aws:es:*:account-id:domain/domain-name/*"
        }
    ]
}
```

Si vous envisagez de réutiliser le rôle pour écrire dans plusieurs domaines, vous pouvez élargir la politique en remplaçant le nom de domaine par un caractère générique (*).

Le rôle doit avoir la relation de confiance suivante, ce qui permet à OpenSearch Ingestion d'assumer le rôle de pipeline :

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"osis-pipelines.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

Étape 2 : configurer l'accès aux données pour le domaine

Pour qu'un pipeline puisse écrire des données dans un domaine, celui-ci doit disposer d'une politique d'accès au niveau du domaine qui autorise le rôle du pipeline à y accéder.

L'exemple de politique d'accès au domaine suivant permet au rôle de pipeline nommé `pipeline-role` d'écrire des données dans le domaine nommé `ingestion-domain` :

JSON

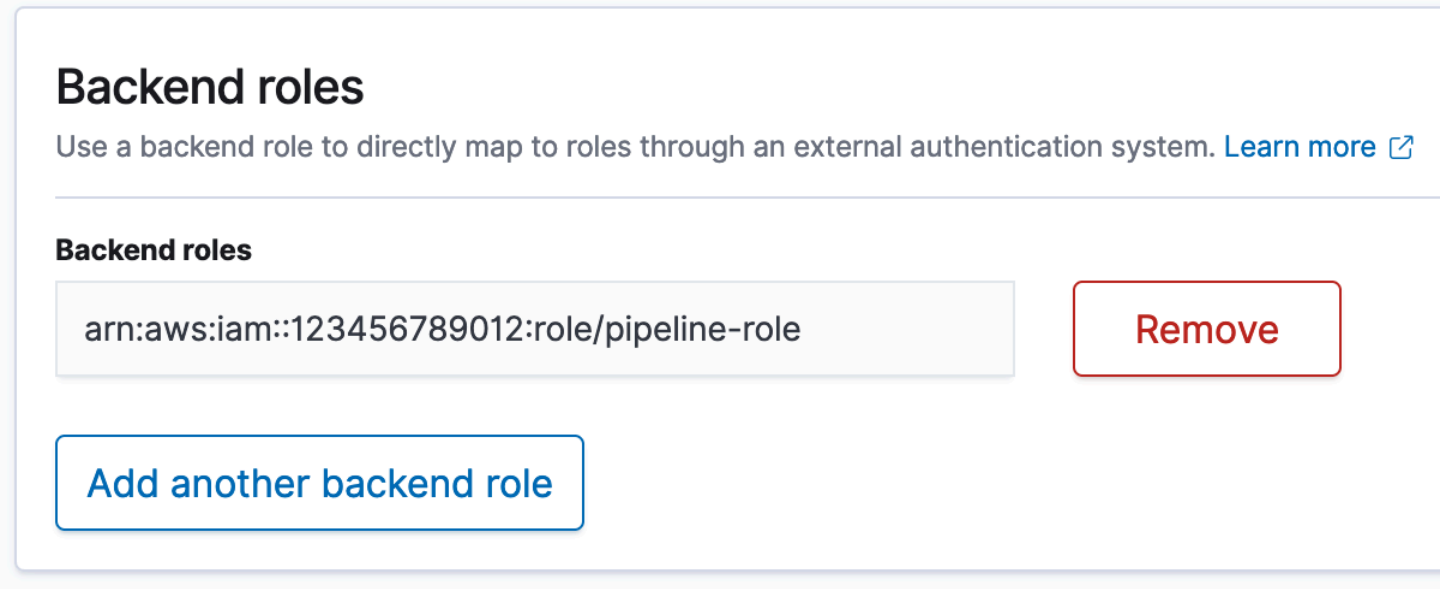
```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::account-id:role/pipeline-role"
      },
      "Action": ["es:DescribeDomain", "es:ESHttp*"],
      "Resource": "arn:aws:es:region:account-id:domain/domain-name/*"
    }
  ]
}
```

Cartographier le rôle du pipeline (uniquement pour les domaines qui utilisent un contrôle d'accès précis)

Si votre domaine utilise un contrôle d'accès précis pour l'authentification, vous devez suivre des étapes supplémentaires pour fournir à votre pipeline l'accès à un domaine. Les étapes varient en fonction de la configuration de votre domaine :

- Scénario 1 : rôle principal et rôle de pipeline différents — Si vous utilisez un nom de ressource IAM Amazon (ARN) comme utilisateur principal et que celui-ci est différent du rôle de pipeline, vous devez mapper le rôle de pipeline au rôle OpenSearch `all_access` principal. Cela ajoute le rôle de pipeline en tant qu'utilisateur principal supplémentaire. Pour plus d'informations, consultez la section Utilisateurs principaux supplémentaires.
- Scénario 2 : utilisateur principal dans la base de données utilisateur interne — Si votre domaine utilise un utilisateur principal dans la base de données utilisateur interne et une authentification HTTP de base pour les OpenSearch tableaux de bord, vous ne pouvez pas transmettre le nom d'utilisateur et le mot de passe principaux directement dans la configuration du pipeline. Mappez plutôt le rôle de pipeline au rôle de OpenSearch `all_access` backend. Cela ajoute le rôle de pipeline en tant qu'utilisateur principal supplémentaire. Pour plus d'informations, consultez la section Utilisateurs principaux supplémentaires.
- Scénario 3 : même rôle principal et rôle de pipeline (peu fréquent) — Si vous utilisez un ARN IAM en tant qu'utilisateur principal et que c'est le même ARN que vous utilisez comme rôle de pipeline, vous n'avez aucune autre action à effectuer. Le pipeline dispose des autorisations requises pour écrire sur le domaine. Ce scénario est rare car la plupart des environnements utilisent un rôle d'administrateur ou un autre rôle en tant que rôle principal.

L'image suivante montre comment mapper le rôle de pipeline à un rôle de backend :



Autoriser les pipelines OpenSearch Amazon Ingestion à accéder aux collections

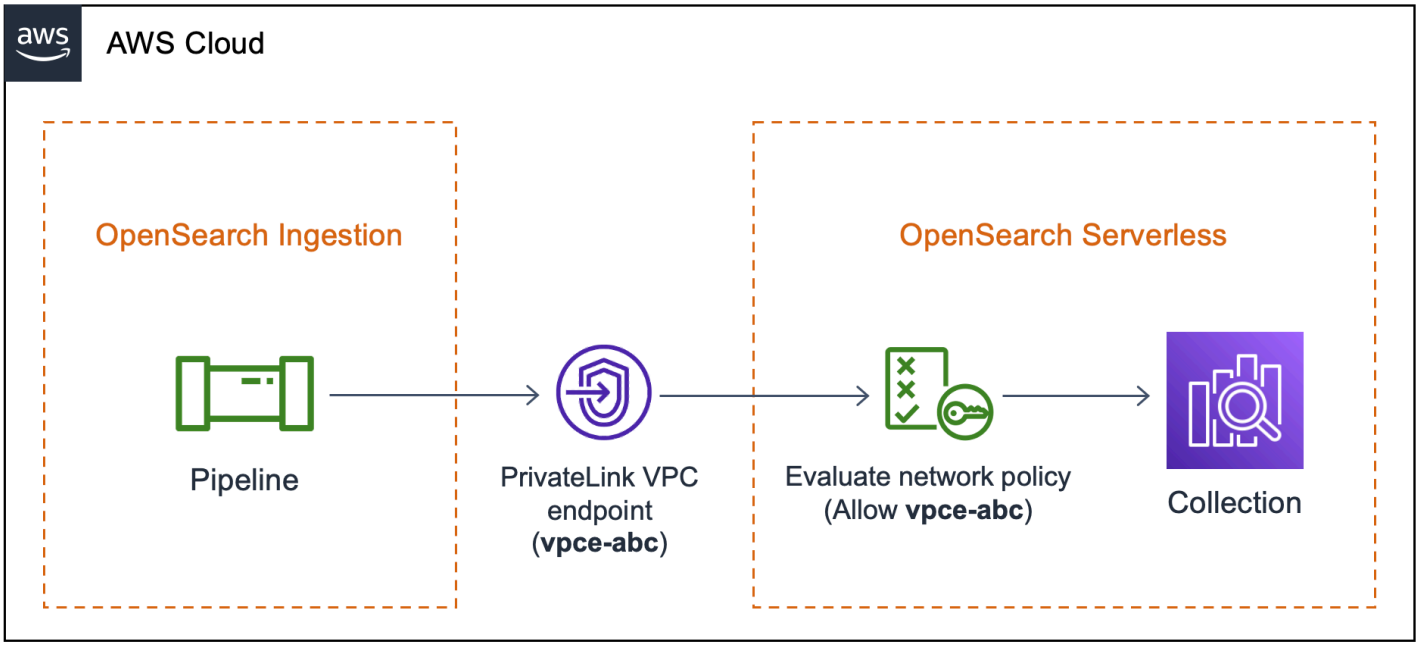
Un pipeline Amazon OpenSearch Ingestion peut écrire dans une collection publique OpenSearch sans serveur ou une collection VPC. Pour donner accès à la collection, vous configurez un rôle de pipeline AWS Identity and Access Management (IAM) avec une politique d'autorisation qui accorde l'accès à la collection. Le pipeline assume ce rôle afin de signer les demandes destinées au récepteur de collecte OpenSearch sans serveur.

Important

Vous pouvez choisir de créer manuellement le rôle de pipeline, ou vous pouvez demander à Ingestion de OpenSearch le créer pour vous lors de la création du pipeline. Si vous choisissez la création automatique des rôles, OpenSearch Ingestion ajoute toutes les autorisations requises à la politique d'accès aux rôles du pipeline en fonction de la source et du récepteur que vous choisissez. Il crée un rôle de pipeline dans IAM avec le préfixe `OpenSearchIngestion-` et le suffixe que vous entrez. Pour de plus amples informations, veuillez consulter the section called “Rôle du pipeline”.

Si OpenSearch Ingestion crée le rôle de pipeline pour vous, vous devez tout de même inclure le rôle dans la politique d'accès aux données de la collection, avant ou après avoir créé le pipeline. Consultez l'étape 2 pour obtenir des instructions.

Lors de la création du pipeline OpenSearch , Ingestion crée une AWS PrivateLink connexion entre le pipeline et la collection OpenSearch Serverless. Tout le trafic provenant du pipeline passe par ce point de terminaison VPC et est acheminé vers la collection. Pour accéder à la collection, le point de terminaison doit être autorisé à accéder à la collection par le biais d'une politique d'accès au réseau.



Rubriques

- Étape 1 : Création du rôle de pipeline
- Étape 2 : Configuration des données et de l'accès au réseau pour la collecte

Étape 1 : Création du rôle de pipeline

Le rôle de pipeline doit être associé à une politique d'autorisation lui permettant d'envoyer des données au récepteur de collecte. Il doit également avoir une relation de confiance permettant à OpenSearch Ingestion d'assumer le rôle. Pour savoir comment associer une politique à un rôle, consultez la section Ajout d'autorisations d'identité IAM dans le Guide de l'utilisateur IAM.

L'exemple de politique suivant illustre le privilège minimal que vous pouvez accorder dans le cadre d'une politique d'accès aux rôles de pipeline pour qu'elle puisse écrire dans des collections :

JSON

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Statement1",
            "Effect": "Allow",
            "Action": [
                "aoss:APIAccessAll",
                "aoss:BatchGetCollection",
                "aoss:CreateSecurityPolicy",
                "aoss:GetSecurityPolicy",
                "aoss:UpdateSecurityPolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

Le rôle doit avoir la relation de confiance suivante, ce qui permet à OpenSearch Ingestion de l'assumer :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "osis-pipelines.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Étape 2 : Configuration des données et de l'accès au réseau pour la collecte

Créez une collection OpenSearch sans serveur avec les paramètres suivants. Pour obtenir des instructions relatives à la création d'une collection, consultezthe section called “Créer des collections”.

Politique d'accès aux données

Créez une politique d'accès aux données pour la collection qui accorde les autorisations requises au rôle de pipeline. Par exemple :

```
[
  {
    "Rules": [
      {
        "Resource": [
          "index/collection-name/*"
        ],
        "Permission": [
          "aoss:CreateIndex",
          "aoss:UpdateIndex",
          "aoss:DescribeIndex",
          "aoss:WriteDocument"
        ],
        "ResourceType": "index"
      }
    ],
    "Principal": [
      "arn:aws:iam::account-id:role/pipeline-role"
    ],
    "Description": "Pipeline role access"
  }
]
```

> **Note**
>
> Dans l'`Principal`élément, spécifiez l'Amazon Resource Name (ARN) du rôle de pipeline.

Politique d'accès au réseau

Chaque collection que vous créez dans OpenSearch Serverless est associée à au moins une politique d'accès réseau. Les politiques d'accès au réseau déterminent si la collection est accessible via Internet à partir de réseaux publics ou si elle doit être consultée de manière privée. Pour plus d'informations sur les politiques réseau, consultezthe section called “Accès réseau”.

Dans le cadre d'une politique d'accès réseau, vous ne pouvez spécifier que des points de terminaison OpenSearch VPC gérés sans serveur. Pour de plus amples informations, veuillez consulter the

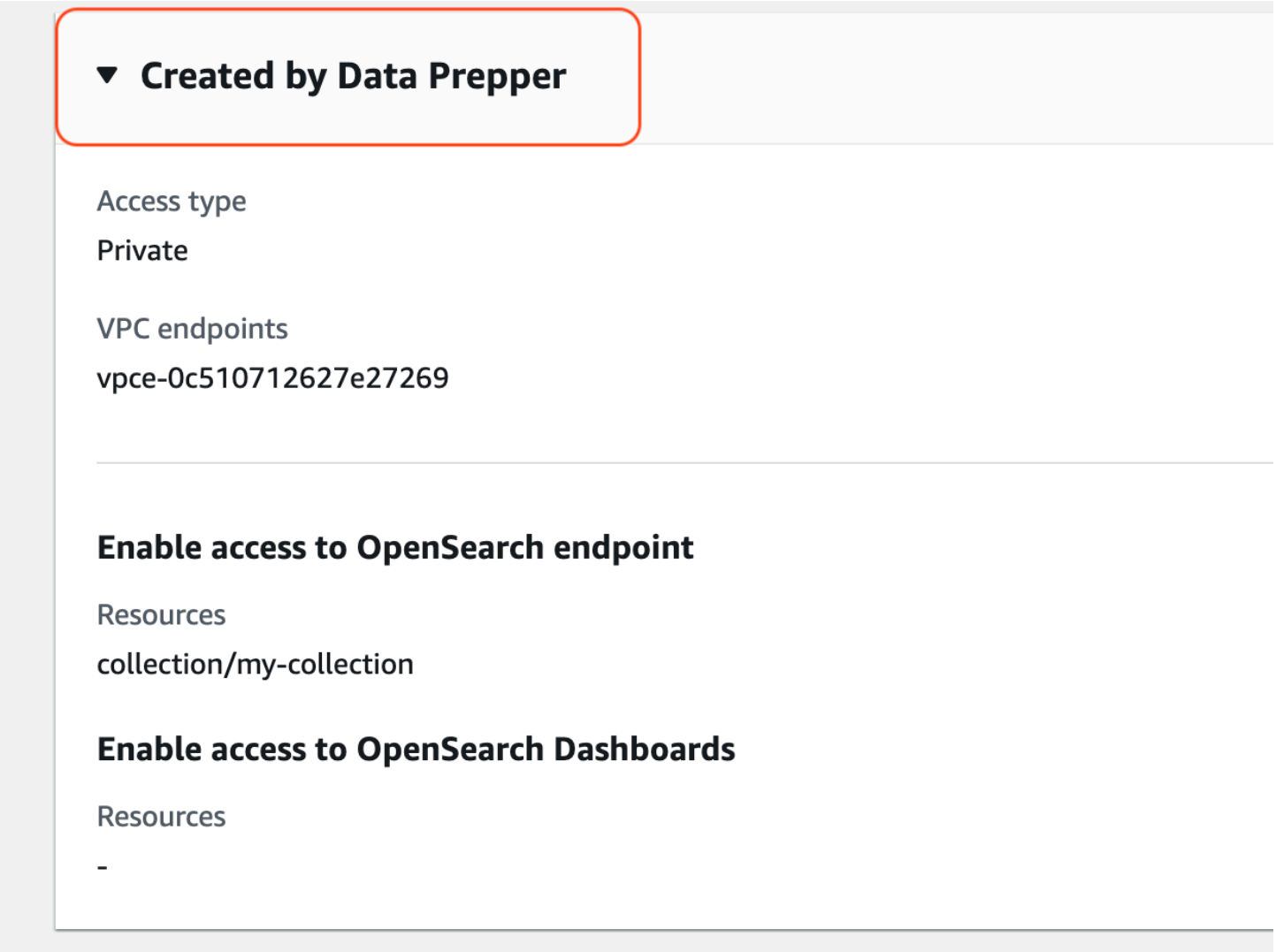
section called “Points de terminaison d’un VPC”. Toutefois, pour que le pipeline puisse écrire dans la collection, la politique doit également accorder l'accès au point de terminaison VPC qu' OpenSearch Ingestion crée automatiquement entre le pipeline et la collection. Par conséquent, si vous choisissez une collection OpenSearch Serverless comme récepteur de destination pour un pipeline, vous devez entrer le nom de la politique réseau associée dans le champ Nom de la stratégie réseau.

Lors de la création du OpenSearch pipeline, Ingestion vérifie l'existence de la politique réseau spécifiée. S'il n'existe pas, OpenSearch Ingestion le crée. Si elle existe, OpenSearch Ingestion la met à jour en y ajoutant une nouvelle règle. La règle accorde l'accès au point de terminaison VPC qui connecte le pipeline et la collection.

Par exemple :

```
{
   "Rules":[
      {
         "Resource":[
            "collection/my-collection"
         ],
         "ResourceType":"collection"
      }
   ],
   "SourceVPCEs":[
      "vpce-0c510712627e27269" # The ID of the VPC endpoint that OpenSearch Ingestion
 creates between the pipeline and collection
   ],
   "Description":"Created by Data Prepper"
}
```

Dans la console, toutes les règles qu' OpenSearch Ingestion ajoute à vos politiques réseau sont nommées Created by Data Prepper :



> **Note**
>
> En général, une règle qui spécifie l'accès public pour une collection remplace une règle qui spécifie un accès privé. Par conséquent, si l'accès public était déjà configuré à la politique, cette nouvelle règle OpenSearch ajoutée par Ingestion ne modifie pas réellement le comportement de la politique. Pour de plus amples informations, veuillez consulter the section called “Priorité des stratégies”.

Si vous arrêtez ou supprimez le pipeline, OpenSearch Ingestion supprime le point de terminaison VPC situé entre le pipeline et la collection. Il modifie également la politique réseau pour supprimer le point de terminaison VPC de la liste des points de terminaison autorisés. Si vous redémarrez le

pipeline, il recrée le point de terminaison VPC et met à jour à nouveau la politique réseau avec l'ID du point de terminaison.

Commencer à utiliser Amazon OpenSearch Ingestion

Amazon OpenSearch Ingestion prend en charge l'ingestion de données dans des domaines OpenSearch de services gérés et des collections OpenSearch sans serveur. Les didacticiels suivants vous expliquent les étapes de base nécessaires à la mise en place et au fonctionnement d'un pipeline.

Le premier didacticiel explique comment utiliser Amazon OpenSearch Ingestion pour configurer un pipeline simple et ingérer des données dans un domaine Amazon OpenSearch Service.

Le deuxième didacticiel explique comment utiliser Amazon OpenSearch Ingestion pour configurer un pipeline simple et intégrer des données dans une collection Amazon OpenSearch Serverless.

> **Note**
>
> La création du pipeline échouera si vous ne configurez pas les autorisations appropriées. Consultez the section called “Configuration des rôles et des utilisateurs” pour mieux comprendre les rôles requis avant de créer un pipeline.

Rubriques

- Tutoriel : Ingestion de données dans un domaine à l'aide d'Amazon OpenSearch Ingestion
- Tutoriel : Ingestion de données dans une collection à l'aide d'Amazon OpenSearch Ingestion

Tutoriel : Ingestion de données dans un domaine à l'aide d'Amazon OpenSearch Ingestion

Ce didacticiel explique comment utiliser Amazon OpenSearch Ingestion pour configurer un pipeline simple et ingérer des données dans un domaine Amazon OpenSearch Service. Un pipeline est une ressource qu' OpenSearch Ingestion approvisionne et gère. Vous pouvez utiliser un pipeline pour filtrer, enrichir, transformer, normaliser et agréger les données à des fins d'analyse et de visualisation en aval dans OpenSearch Service.

Ce didacticiel vous explique les étapes de base nécessaires à la mise en service rapide d'un pipeline. Pour des instructions plus complètes, voirthe section called “Création de pipelines”.

Dans le cadre de ce didacticiel, vous suivrez les étapes suivantes :

1. Créez un domaine.
2. Créez un pipeline.
3. Ingérez des exemples de données.

Dans le didacticiel, vous allez créer les ressources suivantes :

- Un domaine nommé dans `ingestion-domain` lequel le pipeline écrit
- Un pipeline nommé `ingestion-pipeline`

Autorisations requises

Pour terminer ce didacticiel, votre utilisateur ou votre rôle doit être associé à une politique basée sur l'identité avec les autorisations minimales suivantes. Ces autorisations vous permettent de créer un rôle de pipeline et d'associer une politique (`iam:Create*`et`iam:Attach*`), de créer ou de modifier un domaine (`es:*`) et d'utiliser des pipelines (`osis:*`).

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Resource":"*",
         "Action":[
            "osis:*",
            "iam:Create*",
            "iam:Attach*",
            "es:*"
         ]
      },
      {
         "Resource":[
            "arn:aws:iam::your-account-id:role/OpenSearchIngestion-PipelineRole"
         ],
         "Effect":"Allow",
         "Action":[
```

```
            "iam:CreateRole",
            "iam:AttachPolicy",
            "iam:PassRole"
        ]
    }
  ]
}
```

Étape 1 : Création du rôle de pipeline

Tout d'abord, créez un rôle que le pipeline assumera afin d'accéder au récepteur du domaine de OpenSearch service. Vous allez inclure ce rôle dans la configuration du pipeline plus loin dans ce didacticiel.

Pour créer le rôle de pipeline

1. Ouvrez la AWS Identity and Access Management console à l'adresse https://console.aws.amazon.com/iamv2/.
2. Choisissez Politiques, puis choisissez Créer une politique.
3. Dans ce didacticiel, vous allez ingérer des données dans un domaine appelé`ingestion-domain`, que vous allez créer à l'étape suivante. Sélectionnez JSON et collez la politique suivante dans l'éditeur. *`your-account-id`*Remplacez-le par votre identifiant de compte et modifiez la région si nécessaire.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "es:DescribeDomain",
            "Resource": "arn:aws:es:us-east-1:your-account-id:domain/
ingestion-domain"
        },
        {
            "Effect": "Allow",
            "Action": "es:ESHttp*",
```

```
            "Resource": "arn:aws:es:us-east-1:your-account-id:domain/
ingestion-domain/*"
        }
    ]
}
```

Si vous souhaitez écrire des données dans un domaine existant, remplacez-les `ingestion-domain` par le nom de votre domaine.

> **Note**
>
> Pour simplifier ce didacticiel, nous utilisons une politique d'accès étendue. Dans les environnements de production, nous vous recommandons toutefois d'appliquer une politique d'accès plus restrictive à votre rôle de pipeline. Pour un exemple de politique fournissant les autorisations minimales requises, voirthe section called “Accorder aux pipelines l'accès aux domaines”.

4. Choisissez Next, puis Next et nommez votre pipeline de politiques.
5. Choisissez Create Policy (Créer une politique).
6. Créez ensuite un rôle et associez-y la politique. Cliquez sur Rôles, puis sur Créer un rôle.
7. Choisissez Politique de confiance personnalisée et collez la politique suivante dans l'éditeur :

   JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"osis-pipelines.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

8. Choisissez Suivant. Recherchez et sélectionnez ensuite pipeline-policy (que vous venez de créer).
9. Choisissez Next et nommez le rôle PipelineRole.
10. Choisissez Créer un rôle.

N'oubliez pas le nom de ressource Amazon (ARN) du rôle (par exemple,`arn:aws:iam::`*`your-account-id`*`:role/PipelineRole`). Vous en aurez besoin pour créer votre pipeline.

Étape 2 : créer un domaine

Créez d'abord un nom de domaine dans `ingestion-domain` lequel les données seront ingérées.

Accédez à la console Amazon OpenSearch Service à l'https://console.aws.amazon.com/aos/accueil et créez un domaine répondant aux exigences suivantes :

- Exécute la OpenSearch version 1.0 ou ultérieure, ou Elasticsearch 7.4 ou version ultérieure
- Utilise l'accès public
- N'utilise pas de contrôle d'accès détaillé

> **Note**
>
> Ces exigences visent à garantir la simplicité de ce didacticiel. Dans les environnements de production, vous pouvez configurer un domaine avec un accès VPC à l'aide d'un contrôle and/or d'accès précis. Pour utiliser un contrôle d'accès précis, voir Cartographier le rôle du pipeline.

Le domaine doit disposer d'une politique d'accès qui accorde l'autorisation au rôle `OpenSearchIngestion-PipelineRole` IAM, que le OpenSearch service créera pour vous à l'étape suivante. Le pipeline assumera ce rôle afin d'envoyer des données au récepteur de domaine.

Assurez-vous que le domaine applique la politique d'accès au niveau du domaine suivante, qui accorde au rôle de pipeline l'accès au domaine. Remplacez la région et le numéro de compte par les vôtres :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::your-account-id:role/OpenSearchIngestion-
PipelineRole"
      },
      "Action": "es:*",
      "Resource": "arn:aws:es:us-east-1:your-account-id:domain/ingestion-domain/
*"
    }
  ]
}
```

Pour plus d'informations sur la création de politiques d'accès au niveau du domaine, consultez. the section called “Stratégies basées sur les ressources”

Si vous avez déjà créé un domaine, modifiez sa politique d'accès existante pour accorder les autorisations ci-dessus à`OpenSearchIngestion-PipelineRole`.

Étape 3 : Création d'un pipeline

Maintenant que vous avez un domaine, vous pouvez créer un pipeline.

Pour créer un pipeline

1. Dans la console Amazon OpenSearch Service, choisissez Pipelines dans le volet de navigation de gauche.
2. Choisissez Créer un pipeline.
3. Sélectionnez le pipeline vide, puis sélectionnez Select Blueprint.
4. Dans ce didacticiel, nous allons créer un pipeline simple qui utilise le plugin source HTTP. Le plugin accepte les données du journal dans un format de tableau JSON. Nous allons spécifier un seul domaine de OpenSearch service comme récepteur et intégrer toutes les données dans l'`application_logs`index.

Dans le menu Source, choisissez HTTP. Pour le chemin, entrez /logs.

5. Pour simplifier ce didacticiel, nous allons configurer l'accès public pour le pipeline. Pour les options de réseau source, choisissez Accès public. Pour plus d'informations sur la configuration de l'accès au VPC, consultez. the section called “Configuration de l'accès VPC pour les pipelines”
6. Choisissez Suivant.
7. Pour Processeur, entrez la date et choisissez Ajouter.
8. Activez À partir de l'heure de réception. Conservez tous les autres paramètres par défaut.
9. Choisissez Suivant.
10. Configurez les détails du récepteur. Pour le type de OpenSearch ressource, choisissez Cluster géré. Choisissez ensuite le domaine OpenSearch de service que vous avez créé dans la section précédente.

    Dans le champ Nom de l'index, entrez application_logs. OpenSearch L'ingestion crée automatiquement cet index dans le domaine s'il n'existe pas déjà.
11. Choisissez Suivant.
12. Nommez le pipeline d'ingestion. Conservez les paramètres de capacité par défaut.
13. Pour le rôle de pipeline, sélectionnez Créer et utiliser un nouveau rôle de service. Le rôle de pipeline fournit les autorisations requises pour qu'un pipeline puisse écrire sur le récepteur de domaine et lire à partir de sources basées sur le pull. En sélectionnant cette option, vous autorisez OpenSearch Ingestion à créer le rôle pour vous, plutôt que de le créer manuellement dans IAM. Pour de plus amples informations, veuillez consulter the section called “Configuration des rôles et des utilisateurs”.
14. Pour le suffixe du nom du rôle de service, entrez PipelineRole. Dans IAM, le rôle aura le format`arn:aws:iam::`*`your-account-id`*`:role/OpenSearchIngestion-`**`PipelineRole`**.
15. Choisissez Suivant. Vérifiez la configuration de votre pipeline et choisissez Create pipeline. Le pipeline prend 5 à 10 minutes pour devenir actif.

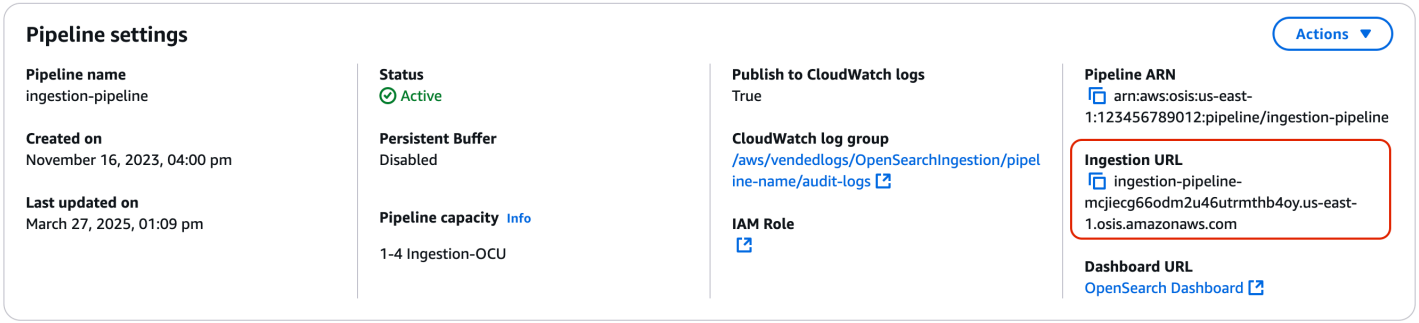
Étape 4 : Ingérer des exemples de données

Lorsque l'état du pipeline est atteint`Active`, vous pouvez commencer à y ingérer des données. Vous devez signer toutes les requêtes HTTP adressées au pipeline à l'aide de Signature Version 4. Utilisez un outil HTTP tel que Postman ou awscurl pour envoyer des données au pipeline. Comme

pour l'indexation de données directement dans un domaine, l'ingestion de données dans un pipeline nécessite toujours soit un rôle IAM, soit une clé d'accès IAM et une clé secrète.

> **Note**
>
> Le signataire principal de la demande doit disposer de l'autorisation `osis:Ingest` IAM.

Tout d'abord, récupérez l'URL d'ingestion sur la page des paramètres du pipeline :



Ensuite, ingérez des exemples de données. La requête suivante utilise awscurl pour envoyer un seul fichier journal au pipeline :

```
awscurl --service osis --region us-east-1 \
    -X POST \
    -H "Content-Type: application/json" \
    -d
 '[{"time":"2014-08-11T11:40:13+00:00","remote_addr":"122.226.223.69","status":"404","request":
 http://www.k2proxy.com//hello.html HTTP/1.1","http_user_agent":"Mozilla/4.0
 (compatible; WOW64; SLCC2;)"}]' \
    https://pipeline-endpoint.us-east-1.osis.amazonaws.com/logs
```

Vous devriez voir une `200 OK` réponse. Si vous recevez une erreur d'authentification, cela peut être dû au fait que vous ingérez des données provenant d'un compte distinct de celui dans lequel se trouve le pipeline. Consultez the section called “Résolution des problèmes d'autorisations”.

Maintenant, interrogez l'`application_logs`index pour vous assurer que votre entrée de journal a été correctement ingérée :

```
awscurl --service es --region us-east-1 \
     -X GET \
```

```
     https://search-ingestion-domain.us-east-1.es.amazonaws.com/application_logs/
_search | json_pp
```

Exemple de réponse:


```
{
   "took":984,
   "timed_out":false,
   "_shards":{
      "total":1,
      "successful":5,
      "skipped":0,
      "failed":0
   },
   "hits":{
      "total":{
         "value":1,
         "relation":"eq"
      },
      "max_score":1.0,
      "hits":[
         {
            "_index":"application_logs",
            "_type":"_doc",
            "_id":"z6VY_IMBRpceX-DU6V4O",
            "_score":1.0,
            "_source":{
               "time":"2014-08-11T11:40:13+00:00",
               "remote_addr":"122.226.223.69",
               "status":"404",
               "request":"GET http://www.k2proxy.com//hello.html HTTP/1.1",
               "http_user_agent":"Mozilla/4.0 (compatible; WOW64; SLCC2;)",
               "@timestamp":"2022-10-21T21:00:25.502Z"
            }
         }
      ]
   }
}
```


Résolution des problèmes d'autorisations

Si vous avez suivi les étapes du didacticiel et que des erreurs d'authentification persistent lorsque vous essayez d'ingérer des données, cela peut être dû au fait que le rôle qui écrit dans un pipeline

se trouve dans un pipeline différent de Compte AWS celui du pipeline lui-même. Dans ce cas, vous devez créer et assumer un rôle qui vous permet spécifiquement d'ingérer des données. Pour obtenir des instructions, veuillez consulter the section called “Fournir un accès à l'ingestion entre comptes”.

Ressources connexes

Ce didacticiel a présenté un cas d'utilisation simple d'ingestion d'un seul document via HTTP. Dans les scénarios de production, vous allez configurer vos applications clientes (telles que Fluent Bit, Kubernetes ou le OpenTelemetry Collector) pour envoyer des données vers un ou plusieurs pipelines. Vos pipelines seront probablement plus complexes que le simple exemple de ce didacticiel.

Pour commencer à configurer vos clients et à ingérer des données, consultez les ressources suivantes :

- Création et gestion de pipelines
- Configuration de vos clients pour envoyer des données à OpenSearch Ingestion
- Documentation Data Prepper

Tutoriel : Ingestion de données dans une collection à l'aide d'Amazon OpenSearch Ingestion

Ce didacticiel explique comment utiliser Amazon OpenSearch Ingestion pour configurer un pipeline simple et intégrer des données dans une collection Amazon OpenSearch Serverless. Un pipeline est une ressource qu' OpenSearch Ingestion approvisionne et gère. Vous pouvez utiliser un pipeline pour filtrer, enrichir, transformer, normaliser et agréger les données à des fins d'analyse et de visualisation en aval dans OpenSearch Service.

Pour un didacticiel expliquant comment ingérer des données dans un domaine de OpenSearch service provisionné, consultez. the section called “Tutoriel : Ingérer des données dans un domaine”

Vous allez effectuer les étapes suivantes dans ce didacticiel :.

1. Créez une collection.
2. Créez un pipeline.
3. Ingérez des exemples de données.

Dans le didacticiel, vous allez créer les ressources suivantes :

- Une collection nommée dans `ingestion-collection` laquelle le pipeline va écrire
- Un pipeline nommé `ingestion-pipeline-serverless`

Autorisations requises

Pour terminer ce didacticiel, votre utilisateur ou votre rôle doit être associé à une politique basée sur l'identité avec les autorisations minimales suivantes. Ces autorisations vous permettent de créer un rôle de pipeline et d'associer une politique (`iam:Create*`et`iam:Attach*`), de créer ou de modifier une collection (`aoss:*`) et de travailler avec des pipelines (`osis:*`).

En outre, plusieurs autorisations IAM sont requises pour créer automatiquement le rôle de pipeline et le transmettre à OpenSearch Ingestion afin qu'elle puisse écrire des données dans la collection.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Resource":"*",
         "Action":[
            "osis:*",
            "iam:Create*",
            "iam:Attach*",
            "aoss:*"
         ]
      },
      {
         "Resource":[
            "arn:aws:iam::your-account-id:role/OpenSearchIngestion-PipelineRole"
         ],
         "Effect":"Allow",
         "Action":[
            "iam:CreateRole",
            "iam:AttachPolicy",
            "iam:PassRole"
         ]
      }
   ]
```

```
}
```

Étape 1 : créer une collection

Créez d'abord une collection dans laquelle ingérer des données. Nous nommerons la collection`ingestion-collection`.

1. Accédez à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Choisissez Collections dans le menu de navigation de gauche, puis choisissez Créer une collection.
3. Nommez la collection ingestion-collection.
4. Pour Sécurité, choisissez Création standard.
5. Sous Paramètres d'accès au réseau, définissez le type d'accès sur Public.
6. Conservez tous les autres paramètres par défaut et choisissez Next (Suivant).
7. Configurez maintenant une politique d'accès aux données pour la collection. Désélectionnez Faire correspondre automatiquement les paramètres de politique d'accès.
8. Pour la méthode de définition, choisissez JSON et collez la politique suivante dans l'éditeur. Cette politique fait deux choses :

   - Permet au rôle de pipeline d'écrire dans la collection.
   - Permet de lire des extraits de la collection. Plus tard, après avoir ingéré des exemples de données dans le pipeline, vous interrogerez la collection pour vous assurer que les données ont été correctement ingérées et écrites dans l'index.

```
[
  {
    "Rules": [
      {
        "Resource": [
          "index/ingestion-collection/*"
        ],
        "Permission": [
          "aoss:CreateIndex",
          "aoss:UpdateIndex",
          "aoss:DescribeIndex",
```

```
          "aoss:ReadDocument",
          "aoss:WriteDocument"
        ],
        "ResourceType": "index"
      }
    ],
    "Principal": [
      "arn:aws:iam::your-account-id:role/OpenSearchIngestion-PipelineRole",
      "arn:aws:iam::your-account-id:role/Admin"
    ],
    "Description": "Rule 1"
  }
]
```

9. Modifiez les `Principal` éléments pour inclure votre Compte AWS identifiant. Pour le second principal, spécifiez un utilisateur ou un rôle que vous pourrez utiliser pour interroger la collection ultérieurement.
10. Choisissez Suivant. Donnez un nom à la politique d'accès pipeline-collection-accesset sélectionnez à nouveau Next.
11. Vérifiez la configuration de votre collection et choisissez Submit (Soumettre).

Étape 2 : Création d'un pipeline

Maintenant que vous disposez d'une collection, vous pouvez créer un pipeline.

Pour créer un pipeline

1. Dans la console Amazon OpenSearch Service, choisissez Pipelines dans le volet de navigation de gauche.
2. Choisissez Créer un pipeline.
3. Sélectionnez le pipeline vide, puis sélectionnez Select Blueprint.
4. Dans ce didacticiel, nous allons créer un pipeline simple qui utilise le plugin source HTTP. Le plugin accepte les données du journal dans un format de tableau JSON. Nous allons spécifier une seule collection OpenSearch Serverless comme récepteur et intégrer toutes les données dans l'`my_logs`index.

   Dans le menu Source, choisissez HTTP. Pour le chemin, entrez /logs.
5. Pour simplifier ce didacticiel, nous allons configurer l'accès public pour le pipeline. Pour les options de réseau source, choisissez Accès public. Pour plus d'informations sur la configuration

de l'accès au VPC, consultez. the section called “Configuration de l'accès VPC pour les pipelines”

6. Choisissez Suivant.
7. Pour Processeur, entrez la date et choisissez Ajouter.
8. Activez À partir de l'heure de réception. Conservez tous les autres paramètres par défaut.
9. Choisissez Suivant.
10. Configurez les détails du récepteur. Pour le type de OpenSearch ressource, choisissez Collection (sans serveur). Choisissez ensuite la collection OpenSearch de services que vous avez créée dans la section précédente.

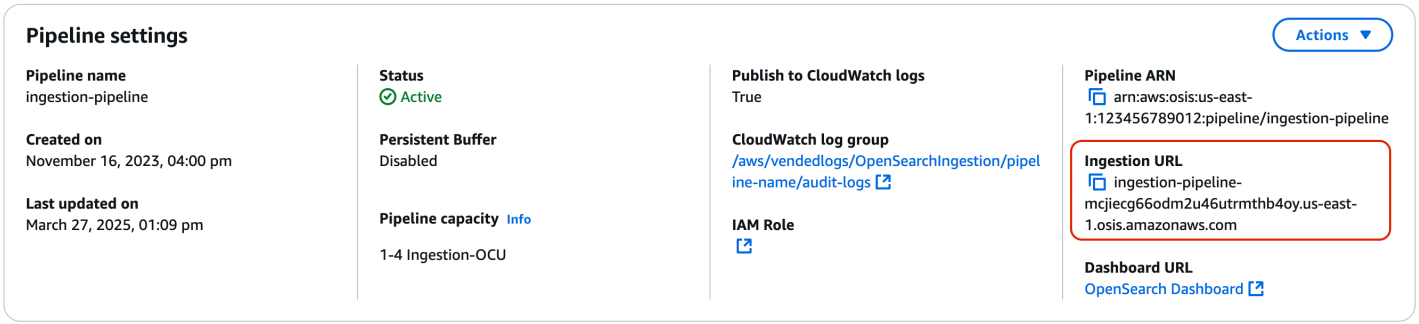
    Laissez le nom de la politique réseau par défaut. Dans le champ Nom de l'index, entrez my_logs. OpenSearch L'ingestion crée automatiquement cet index dans la collection s'il n'existe pas déjà.
11. Choisissez Suivant.
12. Nommez le pipeline ingestion-pipeline-serverless. Conservez les paramètres de capacité par défaut.
13. Pour le rôle de pipeline, sélectionnez Créer et utiliser un nouveau rôle de service. Le rôle de pipeline fournit les autorisations requises pour qu'un pipeline puisse écrire dans le récepteur de collection et lire à partir de sources basées sur le pull. En sélectionnant cette option, vous autorisez OpenSearch Ingestion à créer le rôle pour vous, plutôt que de le créer manuellement dans IAM. Pour de plus amples informations, veuillez consulter the section called “Configuration des rôles et des utilisateurs”.
14. Pour le suffixe du nom du rôle de service, entrez PipelineRole. Dans IAM, le rôle aura le format`arn:aws:iam::`*`your-account-id`*`:role/OpenSearchIngestion-`**`PipelineRole`**.
15. Choisissez Suivant. Vérifiez la configuration de votre pipeline et choisissez Create pipeline. Le pipeline prend 5 à 10 minutes pour devenir actif.

Étape 3 : Ingérer des exemples de données

Lorsque l'état du pipeline est atteint`Active`, vous pouvez commencer à y ingérer des données. Vous devez signer toutes les requêtes HTTP adressées au pipeline à l'aide de Signature Version 4. Utilisez un outil HTTP tel que Postman ou awscurl pour envoyer des données au pipeline. Comme pour l'indexation de données directement dans une collection, l'ingestion de données dans un pipeline nécessite toujours soit un rôle IAM, soit une clé d'accès IAM et une clé secrète.

> **Note**
>
> Le signataire principal de la demande doit disposer de l'autorisation `osis:Ingest` IAM.

Tout d'abord, récupérez l'URL d'ingestion sur la page des paramètres du pipeline :



Envoyez ensuite des échantillons de données vers le chemin d'ingestion. L'exemple de demande suivant utilise awscurl pour envoyer un seul fichier journal au pipeline :

```
awscurl --service osis --region us-east-1 \
    -X POST \
    -H "Content-Type: application/json" \
    -d
 '[{"time":"2014-08-11T11:40:13+00:00","remote_addr":"122.226.223.69","status":"404","request":
 http://www.k2proxy.com//hello.html HTTP/1.1","http_user_agent":"Mozilla/4.0
 (compatible; WOW64; SLCC2;)"}]' \
    https://pipeline-endpoint.us-east-1.osis.amazonaws.com/logs
```

Vous devriez voir une `200 OK` réponse.

Maintenant, interrogez l'`my_logs`index pour vous assurer que l'entrée du journal a été correctement ingérée :

```
awscurl --service aoss --region us-east-1 \
     -X GET \
     https://collection-id.us-east-1.aoss.amazonaws.com/my_logs/_search | json_pp
```

Exemple de réponse:

```
{
   "took":348,
```

```
   "timed_out":false,
   "_shards":{
      "total":0,
      "successful":0,
      "skipped":0,
      "failed":0
   },
   "hits":{
      "total":{
         "value":1,
         "relation":"eq"
      },
      "max_score":1.0,
      "hits":[
         {
            "_index":"my_logs",
            "_id":"1%3A0%3ARJgDvIcBTy5m12xrKE-y",
            "_score":1.0,
            "_source":{
               "time":"2014-08-11T11:40:13+00:00",
               "remote_addr":"122.226.223.69",
               "status":"404",
               "request":"GET http://www.k2proxy.com//hello.html HTTP/1.1",
               "http_user_agent":"Mozilla/4.0 (compatible; WOW64; SLCC2;)",
               "@timestamp":"2023-04-26T05:22:16.204Z"
            }
         }
      ]
   }
}
```


Ressources connexes

Ce didacticiel a présenté un cas d'utilisation simple d'ingestion d'un seul document via HTTP. Dans les scénarios de production, vous allez configurer vos applications clientes (telles que Fluent Bit, Kubernetes ou le OpenTelemetry Collector) pour envoyer des données vers un ou plusieurs pipelines. Vos pipelines seront probablement plus complexes que le simple exemple de ce didacticiel.

Pour commencer à configurer vos clients et à ingérer des données, consultez les ressources suivantes :

- Création et gestion de pipelines

- Configuration de vos clients pour envoyer des données à OpenSearch Ingestion
- Documentation Data Prepper

Présentation des fonctionnalités du pipeline dans Amazon OpenSearch Ingestion

Amazon OpenSearch Ingestion approvisionne des pipelines, qui se composent d'une source, d'une mémoire tampon, de zéro ou plusieurs processeurs et d'un ou plusieurs récepteurs. Les pipelines d'ingestion sont alimentés par Data Prepper en tant que moteur de données. Pour une vue d'ensemble des différents composants d'un pipeline, voirthe section called “Concepts clés”.

Les sections suivantes fournissent un aperçu de certaines des fonctionnalités les plus couramment utilisées dans Amazon OpenSearch Ingestion.

> **Note**
>
> Il ne s'agit pas d'une liste exhaustive des fonctionnalités disponibles pour les pipelines. Pour une documentation complète de toutes les fonctionnalités de pipeline disponibles, consultez la documentation Data Prepper. Notez que OpenSearch l'ingestion impose certaines contraintes sur les plugins et les options que vous pouvez utiliser. Pour de plus amples informations, veuillez consulter the section called “Plugins et options pris en charge”.

Rubriques

- Mise en mémoire tampon persistante
- Fractionnement
- Création de chaînes
- Files d’attente de lettres mortes
- Gestion des indices
- End-to-end accusé de réception
- Contre-pression à la source

Mise en mémoire tampon persistante

Une mémoire tampon persistante stocke vos données dans une mémoire tampon sur disque dans plusieurs zones de disponibilité afin d'améliorer la durabilité des données. Vous pouvez utiliser la mise en mémoire tampon persistante pour ingérer des données provenant de toutes les sources push prises en charge sans configurer de mémoire tampon autonome. Ces sources incluent le HTTP et OpenTelemetry pour les journaux, les traces et les métriques. Pour activer la mise en mémoire tampon persistante, choisissez Activer la mémoire tampon persistante lorsque vous créez ou mettez à jour un pipeline. Pour de plus amples informations, veuillez consulter the section called “Création de pipelines”.

OpenSearch L'ingestion détermine dynamiquement le nombre de OCUs à utiliser pour la mise en mémoire tampon persistante, en tenant compte de la source de données, des transformations en continu et de la destination du récepteur. Comme il en alloue une partie OCUs à la mise en mémoire tampon, vous devrez peut-être augmenter le minimum et le maximum OCUs pour maintenir le même débit d'ingestion. Les pipelines conservent les données dans la mémoire tampon pendant 72 heures au maximum.

Si vous activez la mise en mémoire tampon persistante pour un pipeline, les tailles de charge utile maximales des demandes par défaut sont les suivantes :

- Sources HTTP — 10 Mo
- OpenTelemetry sources — 4 Mo

Pour les sources HTTP, vous pouvez augmenter la taille maximale de la charge utile à 20 Mo. La taille de la charge utile de la demande inclut l'intégralité de la requête HTTP, qui contient généralement plusieurs événements. Chaque événement ne peut pas dépasser 3,5 Mo.

Les pipelines dotés d'une mise en mémoire tampon persistante divisent les unités de pipeline configurées entre les unités de calcul et les unités de mémoire tampon. Si un pipeline utilise un processeur gourmand en CPU tel que grok, key-value ou split string, il alloue les unités dans un rapport de 1:1. buffer-to-compute Sinon, il les alloue selon un ratio de 3:1, en privilégiant toujours les unités de calcul.

Par exemple :

- Pipeline avec grok et 2 unités maximum : 1 unité de calcul et 1 unité tampon
- Pipeline avec grok et 5 unités maximum — 3 unités de calcul et 2 unités de mémoire tampon

- Pipeline sans processeurs et 2 unités maximum : 1 unité de calcul et 1 unité de mémoire tampon
- Pipeline sans processeurs et 4 unités maximum : 1 unité de calcul et 3 unités de mémoire tampon
- Pipeline avec grok et 5 unités maximum — 2 unités de calcul et 3 unités de mémoire tampon

Par défaut, les pipelines utilisent un Clé détenue par AWS pour chiffrer les données de la mémoire tampon. Ces pipelines ne nécessitent aucune autorisation supplémentaire pour le rôle de pipeline.

Vous pouvez également spécifier une clé gérée par le client et ajouter les autorisations IAM suivantes au rôle de pipeline :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "KeyAccess",
            "Effect": "Allow",
            "Action": [
              "kms:Decrypt",
              "kms:GenerateDataKeyWithoutPlaintext"
            ],
            "Resource": "arn:aws:kms:{region}:{aws-account-
id}:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ]
}
```

Pour plus d'informations, consultez Clés gérées par le client dans le Guide du développeur AWS Key Management Service (langue française non garantie).

> **Note**
>
> Si vous désactivez la mise en mémoire tampon persistante, votre pipeline commence à fonctionner entièrement sur la mémoire tampon en mémoire.

Fractionnement

Vous pouvez configurer un pipeline d' OpenSearch ingestion pour diviser les événements entrants en un sous-pipeline, ce qui vous permet d'effectuer différents types de traitement sur le même événement entrant.

L'exemple de pipeline suivant divise les événements entrants en deux sous-pipelines. Chaque sous-pipeline utilise son propre processeur pour enrichir et manipuler les données, puis envoie les données vers différents OpenSearch index.

```
version: "2"
log-pipeline:
  source:
    http:
    ...
  sink:
    - pipeline:
        name: "logs_enriched_one_pipeline"
    - pipeline:
        name: "logs_enriched_two_pipeline"

logs_enriched_one_pipeline:
  source:
    log-pipeline
  processor:
   ...
  sink:
    - opensearch:
        # Provide a domain or collection endpoint
        # Enable the 'serverless' flag if the sink is an OpenSearch Serverless
 collection
        aws:
          ...
        index: "enriched_one_logs"

logs_enriched_two_pipeline:
  source:
    log-pipeline
  processor:
   ...
  sink:
    - opensearch:
        # Provide a domain or collection endpoint
```

```
        # Enable the 'serverless' flag if the sink is an OpenSearch Serverless
 collection
        aws:
          ...
          index: "enriched_two_logs"
```

Création de chaînes

Vous pouvez enchaîner plusieurs sous-pipelines afin d'effectuer le traitement et l'enrichissement des données par morceaux. En d'autres termes, vous pouvez enrichir un événement entrant avec certaines capacités de traitement dans un sous-pipeline, puis l'envoyer vers un autre sous-pipeline pour un enrichissement supplémentaire avec un processeur différent, et enfin l'envoyer vers son OpenSearch récepteur.

Dans l'exemple suivant, le `log_pipeline` sous-pipeline enrichit un événement de journal entrant avec un ensemble de processeurs, puis envoie l'événement à un OpenSearch index nommé. `enriched_logs` Le pipeline envoie le même événement au `log_advanced_pipeline` sous-pipeline, qui le traite et l'envoie à un OpenSearch index différent nomméé`enriched_advanced_logs`.

```
version: "2"
log-pipeline:
  source:
    http:
    ...
  processor:
    ...
  sink:
    - opensearch:
        # Provide a domain or collection endpoint
        # Enable the 'serverless' flag if the sink is an OpenSearch Serverless
 collection
        aws:
          ...
          index: "enriched_logs"
    - pipeline:
        name: "log_advanced_pipeline"

log_advanced_pipeline:
  source:
    log-pipeline
```

```
  processor:
    ...
  sink:
    - opensearch:
        # Provide a domain or collection endpoint
        # Enable the 'serverless' flag if the sink is an OpenSearch Serverless
collection
        aws:
          ...
          index: "enriched_advanced_logs"
```

Files d’attente de lettres mortes

Les files d'attente contenant des lettres mortes (DLQs) sont des destinations pour les événements qu'un pipeline ne parvient pas à écrire dans un récepteur. Dans OpenSearch Ingestion, vous devez spécifier un compartiment Amazon S3 doté des autorisations d'écriture appropriées à utiliser comme DLQ. Vous pouvez ajouter une configuration DLQ à chaque récepteur d'un pipeline. Lorsqu'un pipeline rencontre des erreurs d'écriture, il crée des objets DLQ dans le compartiment S3 configuré. Les objets DLQ existent dans un fichier JSON sous la forme d'un tableau d'événements ayant échoué.

Un pipeline écrit des événements dans le DLQ lorsque l'une des conditions suivantes est remplie :

- Le nombre maximum de tentatives pour le OpenSearch lavabo est épuisé. OpenSearch L'ingestion nécessite un minimum de 16 pour ce réglage.
- Le récepteur rejette des événements en raison d'une erreur.

Configuration

Pour configurer une file d'attente en lettres mortes pour un sous-pipeline, choisissez Enable S3 DLQ lorsque vous configurez la destination de votre récepteur. Spécifiez ensuite les paramètres requis pour la file d'attente. Pour plus d'informations, consultez la section Configuration dans la documentation Data Prepper DLQ.

Les fichiers écrits dans ce DLQ S3 ont le modèle de dénomination suivant :

```
dlq-v${version}-${pipelineName}-${pluginId}-${timestampIso8601}-${uniqueId}
```

Pour obtenir des instructions sur la configuration manuelle du rôle de pipeline afin d'autoriser l'accès au compartiment S3 dans lequel le DLQ écrit, consultez. the section called “Autorisations d'écriture sur Amazon S3 ou dans une file d'attente de lettres mortes”

exemple

Prenons l'exemple de fichier DLQ suivant :

```
dlq-v2-apache-log-pipeline-opensearch-2023-04-05T15:26:19.152938Z-e7eb675a-
f558-4048-8566-dac15a4f8343
```

Voici un exemple de données qui n'ont pas pu être écrites dans le récepteur et qui sont envoyées au compartiment DLQ S3 pour une analyse plus approfondie :

```
Record_0
pluginId            "opensearch"
pluginName          "opensearch"
pipelineName        "apache-log-pipeline"
failedData
index    "logs"
indexId   null
status    0
message  "Number of retries reached the limit of max retries (configured value 15)"
document
log      "sample log"
timestamp     "2023-04-14T10:36:01.070Z"

Record_1
pluginId            "opensearch"
pluginName          "opensearch"
pipelineName        "apache-log-pipeline"
failedData
index               "logs"
indexId   null
status    0
message  "Number of retries reached the limit of max retries (configured value 15)"
document
log                 "another sample log"
timestamp           "2023-04-14T10:36:01.071Z"
```

Gestion des indices

Amazon OpenSearch Ingestion possède de nombreuses fonctionnalités de gestion d'index, notamment les suivantes.

Création d'index

Vous pouvez spécifier un nom d'index dans un puits de pipeline et OpenSearch Ingestion crée l'index lorsqu'elle approvisionne le pipeline. Si un index existe déjà, le pipeline l'utilise pour indexer les événements entrants. Si vous arrêtez et redémarrez un pipeline, ou si vous mettez à jour sa configuration YAML, le pipeline tente de créer de nouveaux index s'ils n'existent pas déjà. Un pipeline ne peut jamais supprimer un index.

Les exemples de cuvettes suivants créent deux index lorsque le pipeline est approvisionné :

```
sink:
  - opensearch:
      index: apache_logs
  - opensearch:
      index: nginx_logs
```

Génération de noms et de modèles d'index

Vous pouvez générer des noms d'index dynamiques en utilisant des variables issues des champs des événements entrants. Dans la configuration du récepteur, utilisez le format `string${}` pour signaler l'interpolation des chaînes et utilisez un pointeur JSON pour extraire les champs des événements. Les options pour `index_type` sont `custom` ou`management_disabled`. Comme la `index_type` valeur par défaut est `custom` pour OpenSearch les domaines et `management_disabled` pour les collections OpenSearch sans serveur, elle peut être désactivée.

Par exemple, le pipeline suivant sélectionne le `metadataType` champ parmi les événements entrants pour générer des noms d'index.

```
pipeline:
  ...
  sink:
    opensearch:
      index: "metadata-${metadataType}"
```

La configuration suivante continue de générer un nouvel index tous les jours ou toutes les heures.

```
pipeline:
  ...
  sink:
    opensearch:
      index: "metadata-${metadataType}-%{yyyy.MM.dd}"

pipeline:
  ...
  sink:
    opensearch:
      index: "metadata-${metadataType}-%{yyyy.MM.dd.HH}"
```

Le nom de l'index peut également être une chaîne simple avec un modèle date-heure comme suffixe, tel que. `my-index-%{yyyy.MM.dd}` Lorsque le récepteur envoie des données à OpenSearch, il remplace le modèle date-heure par l'heure UTC et crée un nouvel index pour chaque jour, tel que. `my-index-2022.01.25` Pour plus d'informations, consultez le DateTimeFormattercours.

Ce nom d'index peut également être une chaîne formatée (avec ou sans suffixe de modèle date-heure), telle que. `my-${index}-name` Lorsque le récepteur envoie des données à OpenSearch, il remplace la `"${index}"` partie par la valeur de l'événement en cours de traitement. Si le format est le cas`"${index1/index2/index3}"`, il remplace le champ `index1/index2/index3` par sa valeur dans l'événement.

Génération d'un document IDs

Un pipeline peut générer un identifiant de document lors de l'indexation de documents vers OpenSearch. Il peut déduire ces documents IDs à partir des champs des événements entrants.

Cet exemple utilise le `uuid` champ d'un événement entrant pour générer un identifiant de document.

```
pipeline:
  ...
  sink:
    opensearch:
      index_type: custom
      index: "metadata-${metadataType}-%{yyyy.MM.dd}"
      "document_id": "uuid"
```

Dans l'exemple suivant, le processeur d'ajout d'entrées fusionne les champs `uuid` et ceux `other_field` de l'événement entrant pour générer un identifiant de document.

Cette `create` action garantit que les documents portant la mention « identique » IDs ne sont pas remplacés. Le pipeline supprime les documents dupliqués sans aucune nouvelle tentative ni événement DLQ. C'est une attente raisonnable pour les auteurs de pipeline qui utilisent cette action, car l'objectif est d'éviter de mettre à jour les documents existants.

```
pipeline:
  ...
  processor:
   - add_entries:
      entries:
        - key: "my_doc_id_field"
          format: "${uuid}-${other_field}"
  sink:
    - opensearch:
       ...
       action: "create"
       document_id: "my_doc_id"
```

Vous souhaiterez peut-être définir l'ID du document d'un événement sur un champ d'un sous-objet. Dans l'exemple suivant, le plugin OpenSearch sink utilise le sous-objet `info/id` pour générer un identifiant de document.

```
sink:
  - opensearch:
       ...
       document_id: info/id
```

Compte tenu de l'événement suivant, le pipeline générera un document dont le `_id` champ est défini sur `json001` :

```
{
   "fieldA":"arbitrary value",
   "info":{
      "id":"json001",
      "fieldA":"xyz",
      "fieldB":"def"
   }
}
```

Génération du routage IDs

Vous pouvez utiliser l'`routing_field`option du plugin OpenSearch sink pour définir la valeur d'une propriété de routage de documents (`_routing`) sur une valeur provenant d'un événement entrant.

Le routage prend en charge la syntaxe des pointeurs JSON, de sorte que les champs imbriqués sont également disponibles, et pas seulement les champs de niveau supérieur.

```
sink:
  - opensearch:
       ...
       routing_field: metadata/id
       document_id: id
```

Compte tenu de l'événement suivant, le plugin génère un document dont le `_routing` champ est défini sur `abcd` :

```
{
   "id":"123",
   "metadata":{
      "id":"abcd",
      "fieldA":"valueA"
   },
   "fieldB":"valueB"
}
```

Pour obtenir des instructions sur la création de modèles d'index que les pipelines peuvent utiliser lors de la création d'index, voir Modèles d'index.

End-to-end accusé de réception

OpenSearch L'ingestion garantit la durabilité et la fiabilité des données en suivant leur transmission de la source aux récepteurs dans des pipelines apatrides à l'aide d'un end-to-endaccusé de réception.

> **Note**
>
> Actuellement, seul le plugin source S3 prend en charge l' end-to-endaccusé de réception.

Avec un end-to-end accusé de réception, le plugin source du pipeline crée un ensemble d'accusés de réception pour surveiller un lot d'événements. Il reçoit un accusé de réception positif lorsque ces événements sont envoyés avec succès à leurs récepteurs, ou un accusé de réception négatif lorsqu'aucun des événements n'a pu être envoyé à leurs récepteurs.

En cas de panne ou de crash d'un composant du pipeline, ou si une source ne reçoit pas d'accusé de réception, la source expire et prend les mesures nécessaires, telles qu'une nouvelle tentative ou l'enregistrement de la panne. Si le pipeline possède plusieurs collecteurs ou plusieurs sous-pipelines configurés, les accusés de réception au niveau de l'événement ne sont envoyés qu'une fois que l'événement a été envoyé à tous les récepteurs de tous les sous-pipelines. Si une DLQ est configurée sur un récepteur, les accusés de end-to-end réception suivent également les événements écrits sur la DLQ.

Pour activer l' end-to-endaccusé de réception, développez les options supplémentaires dans la configuration de la source Amazon S3 et choisissez Activer l'accusé de réception des end-to-end messages.

Contre-pression à la source

Un pipeline peut subir une contre-pression lorsqu'il est occupé à traiter des données, ou si ses récepteurs sont temporairement inactifs ou lents à ingérer les données. OpenSearch L'ingestion permet de gérer la contre-pression de différentes manières en fonction du plugin source utilisé par un pipeline.

Source HTTP

Les pipelines qui utilisent le plug-in source HTTP gèrent la contre-pression différemment selon le composant du pipeline congestionné :

- Tampons — Lorsque les tampons sont pleins, le pipeline commence à renvoyer l'état HTTP `REQUEST_TIMEOUT` avec le code d'erreur 408 au point de terminaison source. Au fur et à mesure que les tampons sont libérés, le pipeline recommence à traiter les événements HTTP.
- Threads source — Lorsque tous les threads source HTTP sont occupés à exécuter des requêtes et que la taille de la file d'attente de demandes non traitées dépasse le nombre maximum autorisé de demandes, le pipeline commence à renvoyer l'état HTTP `TOO_MANY_REQUESTS` avec le code d'erreur 429 au point de terminaison source. Lorsque la file d'attente de demandes tombe en dessous de la taille de file d'attente maximale autorisée, le pipeline recommence à traiter les demandes.

OTel source

Lorsque les tampons sont pleins pour les pipelines qui utilisent des OpenTelemetry sources (OTel journaux, OTel métriques et OTel traces), le pipeline commence à renvoyer l'état HTTP `REQUEST_TIMEOUT` avec le code d'erreur 408 au point de terminaison source. Au fur et à mesure que les tampons sont libérés, le pipeline recommence à traiter les événements.

Source S3

Lorsque les tampons sont pleins pour les pipelines avec une source S3, les pipelines arrêtent de traiter les notifications SQS. Au fur et à mesure que les tampons sont libérés, les pipelines recommencent à traiter les notifications.

Si un récepteur est en panne ou ne parvient pas à ingérer les données et qu'un end-to-end accusé de réception est activé pour la source, le pipeline arrête de traiter les notifications SQS jusqu'à ce qu'il reçoive un accusé de réception de la part de tous les récepteurs.

Création de pipelines OpenSearch Amazon Ingestion

Un pipeline est le mécanisme utilisé OpenSearch par Amazon Ingestion pour déplacer les données de leur source (d'où proviennent les données) vers leur récepteur (où les données sont acheminées). Dans OpenSearch Ingestion, le récepteur sera toujours un domaine Amazon OpenSearch Service unique, tandis que la source de vos données peut être des clients tels qu'Amazon S3, Fluent Bit ou le OpenTelemetry Collector.

Pour plus d'informations, consultez la section Pipelines dans la OpenSearch documentation.

Rubriques

- Conditions préalables et rôle IAM requis
- Autorisations IAM requises
- Spécification de la version du pipeline
- Spécification du chemin d'ingestion
- Création de pipelines
- Suivi de l'état de la création du pipeline
- Travailler avec des plans

Conditions préalables et rôle IAM requis

Pour créer un pipeline d' OpenSearch ingestion, vous devez disposer des ressources suivantes :

- Rôle IAM, appelé rôle de pipeline, assumé OpenSearch par Ingestion pour écrire dans le récepteur. Vous pouvez créer ce rôle à l'avance ou demander à OpenSearch Ingestion de le créer automatiquement pendant que vous créez le pipeline.
- Un domaine de OpenSearch service ou une collection OpenSearch sans serveur servant de récepteur. Si vous écrivez sur un domaine, celui-ci doit exécuter la OpenSearch version 1.0 ou une version ultérieure, ou Elasticsearch 7.4 ou une version ultérieure. Le récepteur doit disposer d'une politique d'accès qui accorde les autorisations appropriées à votre rôle de pipeline IAM.

Pour obtenir des instructions sur la création de ces ressources, consultez les rubriques suivantes :

- the section called “Accorder aux pipelines l'accès aux domaines”
- the section called “Autoriser les pipelines à accéder aux collections”

> **Note**
>
> Si vous écrivez sur un domaine qui utilise un contrôle d'accès précis, vous devez effectuer des étapes supplémentaires. Consultez the section called “Cartographier le rôle du pipeline (uniquement pour les domaines qui utilisent un contrôle d'accès précis)”.

Autorisations IAM requises

OpenSearch L'ingestion utilise les autorisations IAM suivantes pour créer des pipelines :

- `osis:CreatePipeline`— Créez un pipeline.
- `osis:ValidatePipeline`— Vérifiez si une configuration de pipeline est valide.
- `iam:CreateRole`et `iam:AttachPolicy` — Demandez à OpenSearch Ingestion de créer automatiquement le rôle de pipeline pour vous.
- `iam:PassRole`— Transmettez le rôle de pipeline à OpenSearch Ingestion afin qu'elle puisse écrire des données dans le domaine. Cette autorisation doit porter sur la ressource de rôle du pipeline, ou simplement * si vous prévoyez d'utiliser des rôles différents dans chaque pipeline.

Par exemple, la politique suivante autorise la création d'un pipeline :

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Resource":"*",
         "Action":[
            "osis:CreatePipeline",
            "osis:ListPipelineBlueprints",
            "osis:ValidatePipeline"
         ]
      },
      {
         "Resource":[
            "arn:aws:iam::your-account-id:role/pipeline-role"
         ],
         "Effect":"Allow",
         "Action":[
            "iam:CreateRole",
            "iam:AttachPolicy",
            "iam:PassRole"
         ]
      }
   ]
}
```

OpenSearch L'ingestion inclut également une autorisation appeléeosis:Ingest, qui est requise pour envoyer des demandes signées au pipeline à l'aide de Signature Version 4. Pour de plus amples informations, veuillez consulter the section called “Création d'un rôle d'ingestion”.

> **Note**
>
> En outre, le premier utilisateur à créer un pipeline dans un compte doit disposer des autorisations nécessaires pour effectuer cette iam:CreateServiceLinkedRole action. Pour plus d'informations, consultez la section ressource du rôle du pipeline.

Pour plus d'informations sur chaque autorisation, consultez la section Actions, ressources et clés de condition pour OpenSearch l'ingestion dans la référence d'autorisation de service.

Spécification de la version du pipeline

Lorsque vous créez un pipeline à l'aide de l'éditeur de configuration, vous devez spécifier la version principale de Data Prepper que le pipeline exécutera. Pour spécifier la version, incluez l'`version`option dans la configuration de votre pipeline :

```
version: "2"
log-pipeline:
  source:
    ...
```

Lorsque vous choisissez Create, OpenSearch Ingestion détermine la dernière version mineure disponible de la version principale que vous spécifiez et approvisionne le pipeline avec cette version. Par exemple, si vous spécifiez `version: "2"` que la dernière version prise en charge de Data Prepper est 2.1.1, OpenSearch Ingestion approvisionne votre pipeline avec la version 2.1.1. Nous n'affichons pas publiquement la version mineure que votre pipeline exécute.

Afin de mettre à niveau votre pipeline lorsqu'une nouvelle version majeure de Data Prepper est disponible, modifiez la configuration du pipeline et spécifiez la nouvelle version. Vous ne pouvez pas rétrograder un pipeline vers une version antérieure.

> **Note**
>
> OpenSearch Ingestion ne prend pas immédiatement en charge les nouvelles versions de Data Prepper dès leur sortie. Il y aura un certain décalage entre le moment où une nouvelle version sera accessible au public et le moment où elle sera prise en charge dans OpenSearch Ingestion. En outre, OpenSearch Ingestion peut explicitement ne pas prendre en charge complètement certaines versions majeures ou mineures. Pour obtenir la liste complète, consultez the section called “Versions de Data Prepper prises en charge”.

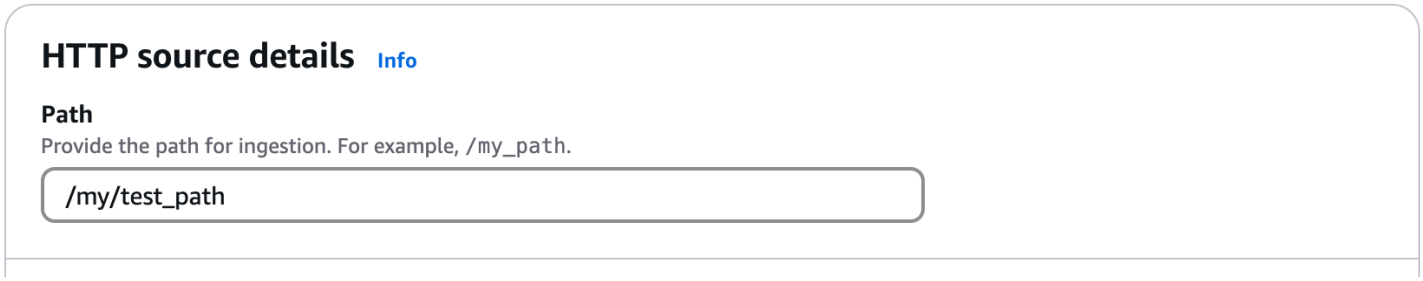
Chaque fois que vous apportez une modification à votre pipeline qui lance un blue/green déploiement, OpenSearch Ingestion peut le mettre à niveau vers la dernière version mineure de la version majeure actuellement configurée pour le pipeline. Pour plus d'informations, consultezthe section called “Déploiements bleu/vert pour les mises à jour du pipeline”. OpenSearch L'ingestion ne

peut pas modifier la version principale de votre pipeline à moins que vous ne mettiez explicitement à jour l'`version`option dans la configuration du pipeline.

Spécification du chemin d'ingestion

Pour les sources basées sur le pull, telles que le OTel traçage et OTel les métriques, OpenSearch l'ingestion nécessite l'`path`option supplémentaire dans votre configuration source. Le chemin est une chaîne telle que`/log/ingest`, qui représente le chemin de l'URI pour l'ingestion. Ce chemin définit l'URI que vous utilisez pour envoyer des données au pipeline.

Supposons, par exemple, que vous spécifiez le chemin suivant pour un pipeline avec une source HTTP :



Lorsque vous ingérez des données dans le pipeline, vous devez spécifier le point de terminaison suivant dans la configuration de votre client :`https://`*`pipeline-name-abc123`*`.us-west-2.osis.amazonaws.com/my/`*`test_path`*.

Le chemin doit commencer par une barre oblique (/) et peut contenir les caractères spéciaux « - », « _ », « . ', et'/', ainsi que l'`${pipelineName}`espace réservé. Si vous utilisez `${pipelineName}` (tel que`/${pipelineName}/test_path`), OpenSearch Ingestion remplace la variable par le nom du sous-pipeline associé.

Création de pipelines

Cette section décrit comment créer des pipelines d' OpenSearch ingestion à l'aide de la console de OpenSearch service et du AWS CLI.

console

Pour créer un pipeline, connectez-vous à la console Amazon OpenSearch Service depuis votre https://console.aws.amazon.com/aos/domicile et choisissez Create pipeline.

Sélectionnez un pipeline vide ou choisissez un plan de configuration. Les plans incluent un pipeline préconfiguré pour une variété de cas d'utilisation courants. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Choisissez Select Blueprint.

Configuration de la source

1. Si vous partez d'un pipeline vide, sélectionnez une source dans le menu déroulant. Les sources disponibles peuvent inclure Services AWS d' OpenTelemetryautres sources ou HTTP. Pour de plus amples informations, veuillez consulter the section called “Intégrer les pipelines”.
2. En fonction de la source que vous choisissez, configurez des paramètres supplémentaires pour la source. Par exemple, pour utiliser Amazon S3 comme source, vous devez spécifier l'URL de la file d'attente Amazon SQS à partir du pipeline de réception des messages. Pour obtenir la liste des plug-ins sources pris en charge et des liens vers leur documentation, consultezthe section called “Plugins et options pris en charge”.
3. Pour certaines sources, vous devez spécifier les options du réseau source. Choisissez l'accès VPC ou l'accès public. Si vous choisissez Public access (Accès public), passez à l'étape suivante. Si vous choisissez l'accès VPC, configurez les paramètres suivants :

Paramètre	Description
Gestion des terminaux	Choisissez si vous souhaitez créer vous-même vos points de terminais on de cloud privé virtuel (VPC) ou laisser Ingestion les créer OpenSearch pour vous. La gestion des points de terminaison est par défaut celle des points de terminaison gérés par OpenSearch Ingestion.
VPC	Choisissez l'ID du VPC que vous souhaitez utiliser. Le VPC et le pipeline doivent être identiques. Région AWS
Sous-réseaux	Choisissez un ou plusieurs sous-réseaux. OpenSearch Le service placera un point de terminaison VPC et des interfaces réseau élastiques dans les sous-réseaux.
Groupes de sécurité	Choisissez un ou plusieurs groupes de sécurité VPC qui permettent à l'application requise d'atteindre le pipeline d' OpenSearch ingestion sur les ports (80 ou 443) et les protocoles (HTTP ou HTTPs) exposés par le pipeline.

Paramètre	Description
Options de fixation en VPC	Si votre source est un point de terminaison autogéré, attachez votre pipeline à un VPC. Choisissez l'une des options CIDR par défaut fournies ou utilisez un CIDR personnalisé.

Pour de plus amples informations, veuillez consulter the section called “Configuration de l'accès VPC pour les pipelines”.

4. Choisissez Suivant.

Configuration du processeur

Ajoutez un ou plusieurs processeurs à votre pipeline. Les processeurs sont des composants d'un sous-pipeline qui vous permettent de filtrer, de transformer et d'enrichir les événements avant de publier des enregistrements dans le domaine ou le collecteur de collection. Pour obtenir la liste des processeurs pris en charge et les liens vers leur documentation, consultezthe section called “Plugins et options pris en charge”.

Vous pouvez sélectionner Actions et ajouter les éléments suivants :

- Routage conditionnel — Achemine les événements vers différents puits en fonction de conditions spécifiques. Pour plus d'informations, consultez la section Routage conditionnel.
- Sous-pipeline : chaque sous-pipeline est une combinaison d'une source unique, de zéro ou plusieurs processeurs et d'un seul récepteur. Un seul sous-pipeline peut avoir une source externe. Tous les autres doivent avoir des sources qui sont d'autres sous-pipelines dans la configuration globale du pipeline. Une configuration de pipeline unique peut contenir de 1 à 10 sous-pipelines.

Choisissez Suivant.

Configurer le lavabo

Sélectionnez la destination où le pipeline publie les enregistrements. Chaque sous-pipeline doit contenir au moins un puits. Vous pouvez ajouter un maximum de 10 cuves à un pipeline.

Pour les OpenSearch éviers, configurez les champs suivants :

Paramètre	Description
Nom de la politique réseau (récepteurs sans serveur uniquement)	Si vous avez sélectionné une collection OpenSearch sans serveur, entrez un nom de politique réseau. OpenSearch L'ingestion crée la politique si elle n'existe pas ou la met à jour avec une règle qui accorde l'accès au point de terminaison VPC connectant le pipeline et la collection. Pour de plus amples informations, veuillez consulter the section called “Autoriser les pipelines à accéder aux collections”.
Nom de l'index	Nom de l'index dans lequel le pipeline envoie les données. OpenSearch L'ingestion crée cet index s'il n'existe pas déjà.
Options de mappage d'index	Choisissez la manière dont le pipeline stocke et indexe les documents et leurs champs dans le OpenSearch récepteur. Si vous sélectionnez le mappage dynamique, OpenSearch des champs sont automatiquement ajoutés lorsque vous indexez un document. Si vous sélectionnez Personnal iser le mappage, entrez un modèle de mappage d'index. Pour plus d'informa tions, consultez la section Modèles d'index.
Activer DLQ	Configurez une file d'attente de lettres mortes (DLQ) Amazon S3 pour le pipeline. Pour de plus amples informations, veuillez consulter the section called “Files d’attente de lettres mortes”.
Réglages supplémentaires	Configurez les options avancées pour le OpenSearch lavabo. Pour plus d'informations, consultez la section Options de configuration dans la documentation de Data Prepper.

Pour ajouter un récepteur Amazon S3, choisissez Ajouter un récepteur et Amazon S3. Pour de plus amples informations, veuillez consulter the section called “Amazon S3 en tant que destination”.

Choisissez Suivant.

Configurer le pipeline

Configurez les paramètres de pipeline supplémentaires suivants :

Paramètre	Description
Nom du pipeline	Nom unique pour le pipeline.
Tampon persistant	Une mémoire tampon persistante stocke vos données dans une mémoire tampon sur disque dans plusieurs zones de disponibilité. Pour de plus amples informations, veuillez consulter the section called “Mise en mémoire tampon persistante”. Si vous activez la mise en mémoire tampon persistante, sélectionnez la AWS Key Management Service clé pour chiffrer les données de la mémoire tampon.
Capacité du pipeline	La capacité minimale et maximale du pipeline, en unités de OpenSearch calcul d'ingestion (OCUs). Pour de plus amples informations, veuillez consulter the section called “Dimensionnement des pipelines”.
Rôle du pipeline	Rôle IAM qui fournit les autorisations requises pour que le pipeline puisse écrire sur le récepteur et lire à partir de sources basées sur le pull. Vous pouvez créer le rôle vous-même ou demander à Ingestion de OpenSearch le créer pour vous en fonction du cas d'utilisation que vous avez sélectionné. Pour de plus amples informations, veuillez consulter the section called “Configuration des rôles et des utilisateurs”.
Balises	Ajoutez une ou plusieurs balises à votre pipeline. Pour de plus amples informations, veuillez consulter the section called “Pipelines de balisage”.
Options de publication des journaux	Activez la publication des journaux du pipeline sur Amazon CloudWatch Logs. Nous vous recommandons d'activer la publication des journaux afin de pouvoir résoudre plus facilement les problèmes liés au pipeline. Pour de plus amples informations, veuillez consulter the section called “Surveillance des journaux du pipeline”.

Choisissez Next., puis passez en revue la configuration de votre pipeline et choisissez Create pipeline.

OpenSearch Ingestion exécute un processus asynchrone pour créer le pipeline. Une fois que l'état du pipeline est `Active` atteint, vous pouvez commencer à ingérer des données.

AWS CLI

La commande create-pipeline accepte la configuration du pipeline sous forme de chaîne ou dans un fichier .yaml ou .json. Si vous fournissez la configuration sous forme de chaîne, chaque nouvelle ligne doit être supprimée avec`\n`. Par exemple, `"log-pipeline:\n source:\n http:\n processor:\n - grok:\n ..."`

L'exemple de commande suivant crée un pipeline avec la configuration suivante :

- Minimum de 4 OCUs ingestions, maximum de 10 ingestions OCUs
- Provisionné dans un cloud privé virtuel (VPC)
- Publication de journaux activée

```
aws osis create-pipeline \
  --pipeline-name my-pipeline \
  --min-units 4 \
  --max-units 10 \
  --log-publishing-options
 IsLoggingEnabled=true,CloudWatchLogDestination={LogGroup="MyLogGroup"} \
  --vpc-options
 SecurityGroupIds={sg-12345678,sg-9012345},SubnetIds=subnet-1212234567834asdf \
  --pipeline-configuration-body "file://pipeline-config.yaml" \
  --pipeline-role-arn  arn:aws:iam::1234456789012:role/pipeline-role
```

OpenSearch Ingestion exécute un processus asynchrone pour créer le pipeline. Une fois que l'état du pipeline est `Active` atteint, vous pouvez commencer à ingérer des données. Pour vérifier l'état du pipeline, utilisez la GetPipelinecommande.

OpenSearch API d'ingestion

Pour créer un pipeline d' OpenSearch ingestion à l'aide de OpenSearch l'API d'ingestion, appelez l'CreatePipelineopération.

Une fois votre pipeline créé avec succès, vous pouvez configurer votre client et commencer à ingérer des données dans votre domaine OpenSearch de service. Pour de plus amples informations, veuillez consulter the section called “Intégrer les pipelines”.

Suivi de l'état de la création du pipeline

Vous pouvez suivre l'état d'un pipeline au fur et à mesure qu' OpenSearch Ingestion le provisionne et le prépare à ingérer des données.

console

Une fois que vous avez initialement créé un pipeline, celui-ci passe par plusieurs étapes car OpenSearch Ingestion le prépare à ingérer des données. Pour visualiser les différentes étapes de création du pipeline, choisissez le nom du pipeline pour voir sa page de paramètres du pipeline. Sous État, choisissez Afficher les détails.

Un pipeline passe par les étapes suivantes avant d'être disponible pour l'ingestion de données :

- Validation — Validation de la configuration du pipeline. Lorsque cette étape est terminée, toutes les validations sont réussies.
- Création d'un environnement : préparation et approvisionnement des ressources. Lorsque cette étape est terminée, le nouvel environnement de pipeline a été créé.
- Déployer le pipeline : déploiement du pipeline. Lorsque cette étape est terminée, le pipeline a été déployé avec succès.
- Vérifier l'état du pipeline : vérification de l'état du pipeline. Lorsque cette étape est terminée, tous les bilans de santé sont passés avec succès.
- Activer le trafic — Permettre au pipeline d'ingérer des données. Lorsque cette étape est terminée, vous pouvez commencer à ingérer des données dans le pipeline.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Utilisez la get-pipeline-change-progresscommande pour vérifier l'état d'un pipeline. La AWS CLI demande suivante vérifie l'état d'un pipeline nommé `my-pipeline` :

```
aws osis get-pipeline-change-progress \
    --pipeline-name my-pipeline
```

Réponse :

```
{
   "ChangeProgressStatuses": {
```

```
      "ChangeProgressStages": [
         {
            "Description": "Validating pipeline configuration",
            "LastUpdated": 1.671055851E9,
            "Name": "VALIDATION",
            "Status": "PENDING"
         }
      ],
      "StartTime": 1.671055851E9,
      "Status": "PROCESSING",
      "TotalNumberOfStages": 5
   }
}
```

OpenSearch API d'ingestion

Pour suivre l'état de la création du pipeline à l'aide de OpenSearch l'API Ingestion, appelez l'GetPipelineChangeProgressopération.

Travailler avec des plans

Plutôt que de créer une définition de pipeline à partir de zéro, vous pouvez utiliser des plans de configuration, qui sont des modèles préconfigurés pour des scénarios d'ingestion courants tels que Trace Analytics ou les journaux Apache. Les plans de configuration vous aident à approvisionner facilement des pipelines sans avoir à créer une configuration à partir de zéro.

console

Pour utiliser un plan de pipeline

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Choisissez Pipelines dans le volet de navigation de gauche, puis choisissez Créer un pipeline.
3. Sélectionnez un plan dans la liste des cas d'utilisation, puis choisissez Sélectionner un plan. La configuration du pipeline est renseignée avec un sous-pipeline correspondant au cas d'utilisation que vous avez sélectionné.

   Le plan du pipeline n'est pas valide tel quel. Vous devez définir des paramètres supplémentaires en fonction de la source sélectionnée.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour obtenir une liste de tous les plans disponibles à l'aide du AWS CLI, envoyez une list-pipeline-blueprintsdemande.

```
aws osis list-pipeline-blueprints
```

La demande renvoie une liste de tous les plans disponibles.

Pour obtenir des informations plus détaillées sur un plan spécifique, utilisez la get-pipeline-blueprintcommande :

```
aws osis get-pipeline-blueprint --blueprint-name AWS-ApacheLogPipeline
```

Cette requête renvoie le contenu du plan du pipeline de logs Apache :

```
{
   "Blueprint":{
      "PipelineConfigurationBody":"###\n  # Limitations: https://docs.aws.amazon.com/
opensearch-service/latest/ingestion/ingestion.html#ingestion-limitations\n###\n###\n
 # apache-log-pipeline:\n    # This pipeline receives logs via http (e.g. FluentBit),
 extracts important values from the logs by matching\n    # the value in the 'log' key
 against the grok common Apache log pattern. The grokked logs are then sent\n    # to
 OpenSearch to an index named 'logs'\n###\n\nversion: \"2\"\napache-log-pipeline:\n
 source:\n    http:\n      # Provide the path for ingestion. ${pipelineName} will be
 replaced with pipeline name configured for this pipeline.\n      # In this case it
 would be \"/apache-log-pipeline/logs\". This will be the FluentBit output URI value.
\n      path: \"/${pipelineName}/logs\"\n  processor:\n    - grok:\n        match:\n
         log: [ \"%{COMMONAPACHELOG_DATATYPED}\" ]\n  sink:\n    - opensearch:\n
   # Provide an AWS OpenSearch Service domain endpoint\n        # hosts: [ \"https://
search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-east-1.es.amazonaws.com\" ]\n        aws:
\n          # Provide the region of the domain.\n          # region: \"us-east-1\"\n
        # Enable the 'serverless' flag if the sink is an Amazon OpenSearch Serverless
 collection\n          # serverless: true\n        index: \"logs\"\n        # Enable
 the S3 DLQ to capture any failed requests in an S3 bucket\n        # dlq:\n
  # s3:\n            # Provide an S3 bucket\n            # bucket: \"your-dlq-bucket-
name\"\n            # Provide a key path prefix for the failed requests\n
  # key_path_prefix: \"${pipelineName}/logs/dlq\"\n            # Provide the region
 of the bucket.\n            # region: \"us-east-1\"\n            # Provide a Role
 ARN with access to the bucket. This role should have a trust relationship with osis-
pipelines.amazonaws.com\n"
      "BlueprintName":"AWS-ApacheLogPipeline"
```

```
    }
}
```

OpenSearch API d'ingestion

Pour obtenir des informations sur les plans de pipeline à l'aide de OpenSearch l'API d'ingestion, utilisez les GetPipelineBlueprintopérations ListPipelineBlueprintset.

Afficher les pipelines d' OpenSearch ingestion d'Amazon

Vous pouvez consulter les détails d'un pipeline Amazon OpenSearch Ingestion à l'aide de l'API AWS Management Console, de ou de l'API OpenSearch d'ingestion. AWS CLI

console

Pour afficher un pipeline

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, choisissez pipelines.
3. (Facultatif) Pour afficher les pipelines ayant un statut particulier, choisissez N'importe quel statut et sélectionnez un statut à filtrer.

   Un pipeline peut avoir les états suivants :

   - `Active`— Le pipeline est actif et prêt à ingérer des données.
   - `Creating`— Le pipeline est en cours de création.
   - `Updating`— Le pipeline est en cours de mise à jour.
   - `Deleting`— Le pipeline est en cours de suppression.
   - `Create failed`— Le pipeline n'a pas pu être créé.
   - `Update failed`— Le pipeline n'a pas pu être mis à jour.
   - `Stop failed`— Le pipeline n'a pas pu être arrêté.
   - `Start failed`— Le pipeline n'a pas pu être démarré.
   - `Stopping`— Le pipeline est en cours d'arrêt.
   - `Stopped`— Le pipeline est arrêté et peut être redémarré à tout moment.
   - `Starting`— Le pipeline démarre.

L'ingestion ne vous est pas facturée OCUs lorsqu'un pipeline se trouve dans les `Stopped` états `Create failed`,`Creating`,`Deleting`, et.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour afficher les pipelines à l'aide de AWS CLI, envoyez une demande list-pipelines :

```
aws osis list-pipelines
```

La requête renvoie une liste de tous les pipelines existants :

```
{
    "NextToken": null,
    "Pipelines": [
        {
            "CreatedAt": 1.671055851E9,
            "LastUpdatedAt": 1.671055851E9,
            "MaxUnits": 4,
            "MinUnits": 2,
            "PipelineArn": "arn:aws:osis:us-west-2:123456789012:pipeline/log-pipeline",
            "PipelineName": "log-pipeline",
            "Status": "ACTIVE",
            "StatusReason": {
                "Description": "The pipeline is ready to ingest data."
            }
        },
            "CreatedAt": 1.671055851E9,
            "LastUpdatedAt": 1.671055851E9,
            "MaxUnits": 2,
            "MinUnits": 8,
            "PipelineArn": "arn:aws:osis:us-west-2:123456789012:pipeline/another-
pipeline",
            "PipelineName": "another-pipeline",
            "Status": "CREATING",
            "StatusReason": {
                "Description": "The pipeline is being created. It is not able to ingest
 data."
            }
        }
    ]
}
```

Pour obtenir des informations sur un pipeline unique, utilisez la commande get-pipeline :

```
aws osis get-pipeline --pipeline-name "my-pipeline"
```

La demande renvoie des informations de configuration pour le pipeline spécifié :

```
{
    "Pipeline": {
        "PipelineName": "my-pipeline",
        "PipelineArn": "arn:aws:osis:us-east-1:123456789012:pipeline/my-pipeline",
        "MinUnits": 9,
        "MaxUnits": 10,
        "Status": "ACTIVE",
        "StatusReason": {
            "Description": "The pipeline is ready to ingest data."
        },
        "PipelineConfigurationBody": "log-pipeline:\n source:\n http:\n processor:\n
 - grok:\n match:\nlog: [ '%{COMMONAPACHELOG}' ]\n - date:\n from_time_received: true
\n destination: \"@timestamp\"\n  sink:\n - opensearch:\n hosts: [ \"https://search-
mdp-performance-test-duxkb4qnycd63rpy6svmvyvfpi.us-east-1.es.amazonaws.com\" ]\n index:
 \"apache_logs\"\n aws_sts_role_arn: \"arn:aws:iam::123456789012:role/my-domain-role
\"\n  aws_region: \"us-east-1\"\n  aws_sigv4: true",,
        "CreatedAt": "2022-10-01T15:28:05+00:00",
        "LastUpdatedAt": "2022-10-21T21:41:08+00:00",
        "IngestEndpointUrls": [
            "my-pipeline-123456789012.us-east-1.osis.amazonaws.com"
        ]
    }
}
```

OpenSearch API d'ingestion

Pour afficher les pipelines OpenSearch d'ingestion à l'aide de OpenSearch l'API d'ingestion, appelez les GetPipelineopérations ListPipelinesand.

Mise à jour des pipelines OpenSearch Amazon Ingestion

Vous pouvez mettre à jour les pipelines Amazon OpenSearch Ingestion à l' AWS Management Console aide de l'API AWS CLI, de ou de l'API d' OpenSearch ingestion. OpenSearch L'ingestion lance un blue/green déploiement lorsque vous mettez à jour la configuration d'un pipeline. Pour de plus amples informations, veuillez consulter the section called “Déploiements bleu/vert pour les mises à jour du pipeline”.

Rubriques

- Considérations
- Autorisations nécessaires
- Mise à jour des pipelines
- Déploiements bleu/vert pour les mises à jour du pipeline

Considérations

Lorsque vous mettez à jour un pipeline, tenez compte des points suivants :

- Vous ne pouvez pas mettre à jour le nom ou les paramètres réseau d'un pipeline.
- Si votre pipeline écrit sur un récepteur de domaine VPC, vous ne pouvez pas revenir en arrière et remplacer le récepteur par un autre domaine VPC une fois le pipeline créé. Vous devez supprimer et recréer le pipeline avec le nouveau récepteur. Vous pouvez toujours faire passer le récepteur d'un domaine VPC à un domaine public, d'un domaine public à un domaine VPC ou d'un domaine public à un autre domaine public.
- Vous pouvez à tout moment faire basculer le récepteur du pipeline entre un domaine OpenSearch de service public et une collection OpenSearch sans serveur.
- Lorsque vous mettez à jour la configuration de la source, du processeur ou du récepteur d'un pipeline, OpenSearch Ingestion lance un blue/green déploiement. Pour de plus amples informations, veuillez consulter the section called “Déploiements bleu/vert pour les mises à jour du pipeline”.
- Lorsque vous mettez à jour la configuration de la source, du processeur ou du récepteur d'un pipeline OpenSearch , Ingestion met automatiquement à niveau votre pipeline vers la dernière version mineure prise en charge de la version principale de Data Prepper exécutée par le pipeline. Ce processus permet de maintenir votre pipeline à jour avec les dernières corrections de bogues et améliorations de performances.
- Vous pouvez toujours apporter des mises à jour à votre pipeline lorsqu'il est arrêté.

Autorisations nécessaires

OpenSearch L'ingestion utilise les autorisations IAM suivantes pour mettre à jour les pipelines :

- `osis:UpdatePipeline`— Mettez à jour un pipeline.

- `osis:ValidatePipeline`— Vérifiez si une configuration de pipeline est valide.
- `iam:PassRole`— Transmettez le rôle de pipeline à OpenSearch Ingestion afin qu'elle puisse écrire des données dans le domaine. Cette autorisation n'est requise que si vous mettez à jour la configuration du pipeline, et non si vous modifiez d'autres paramètres tels que la publication des journaux ou les limites de capacité.

Par exemple, la politique suivante autorise la mise à jour d'un pipeline :

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Resource":"*",
         "Action":[
            "osis:UpdatePipeline",
            "osis:ValidatePipeline"
         ]
      },
      {
         "Resource":[
            "arn:aws:iam::your-account-id:role/pipeline-role"
         ],
         "Effect":"Allow",
         "Action":[
            "iam:PassRole"
         ]
      }
   ]
}
```

Mise à jour des pipelines

Vous pouvez mettre à jour les pipelines Amazon OpenSearch Ingestion à l' AWS Management Console aide de l'API AWS CLI, de ou de l'API d' OpenSearch ingestion.

console

Pour mettre à jour un pipeline

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Choisissez Pipelines dans le volet de navigation de gauche.
3. Choisissez un pipeline pour ouvrir ses paramètres. Choisissez ensuite l'une des options d'édition.
4. Une fois les modifications terminées, choisissez Save (Enregistrer).

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour mettre à jour un pipeline à l'aide du AWS CLI, envoyez une demande de mise à jour du pipeline. L'exemple de demande suivant télécharge un nouveau fichier de configuration et met à jour les valeurs de capacité minimale et maximale :

```
aws osis update-pipeline \
  --pipeline-name "my-pipeline" \
  --pipline-configuration-body "file://new-pipeline-config.yaml" \
  --min-units 11 \
  --max-units 18
```

OpenSearch API d'ingestion

Pour mettre à jour un pipeline d' OpenSearch ingestion à l'aide de OpenSearch l'API d'ingestion, appelez l'UpdatePipelineopération.

Déploiements bleu/vert pour les mises à jour du pipeline

OpenSearch L'ingestion lance un processus de déploiement bleu/vert lorsque vous mettez à jour la configuration d'un pipeline.

Blue/green refers to the practice of creating a new environment for pipeline updates and routing traffic to the new environment after those updates are complete. The practice minimizes downtime and maintains the original environment in the event that deployment to the new environment is unsuccessful. Blue/greenles déploiements eux-mêmes n'ont aucun impact sur les performances, mais les performances peuvent changer si la configuration de votre pipeline change d'une manière qui altère les performances.

OpenSearch L'ingestion bloque l'auto-scaling lors blue/green des déploiements. Vous continuez à être facturé uniquement pour le trafic vers l'ancien pipeline jusqu'à ce qu'il soit redirigé vers le nouveau pipeline. Une fois le trafic redirigé, vous n'êtes facturé que pour le nouveau pipeline. Vous n'êtes jamais facturé pour deux pipelines simultanément.

Lorsque vous mettez à jour la configuration de la source, du processeur ou du récepteur d'un OpenSearch pipeline, Ingestion peut automatiquement mettre à niveau votre pipeline vers la dernière version mineure prise en charge de la version principale exécutée par le pipeline. Par exemple, vous avez `version: "2"` peut-être intégré la configuration de votre pipeline et OpenSearch Ingestion a initialement provisionné le pipeline avec la version 2.1.0. Lorsque la prise en charge de la version 2.1.1 est ajoutée et que vous modifiez la configuration de votre pipeline, OpenSearch Ingestion met à niveau votre pipeline vers la version 2.1.1.

Ce processus permet de maintenir votre pipeline à jour avec les dernières corrections de bogues et améliorations de performances. OpenSearch Ingestion ne peut pas mettre à jour la version principale de votre pipeline à moins que vous ne changiez manuellement l'`version`option dans la configuration du pipeline.

Gestion des coûts du OpenSearch pipeline Amazon Ingestion

Vous pouvez démarrer et arrêter des pipelines d'ingestion dans Amazon OpenSearch Ingestion pour contrôler le flux de données en fonction de vos besoins. L'arrêt d'un pipeline interrompt le traitement des données tout en préservant les configurations. Vous pouvez donc le redémarrer sans le reconfigurer. Cela peut aider à optimiser les coûts, à gérer l'utilisation des ressources ou à résoudre les problèmes. Lorsque vous arrêtez un pipeline, OpenSearch Ingestion ne traite pas les données entrantes, mais les données précédemment ingérées restent disponibles dans OpenSearch.

Le démarrage et l'arrêt simplifie les processus de configuration et de destruction des pipelines que vous utilisez à des fins de développement, de test ou d'activités similaires qui ne nécessitent pas une disponibilité continue. Lorsque votre pipeline est arrêté, vous ne serez pas facturé pour des heures d'ingestion EC2. Vous pouvez toujours mettre à jour les pipelines arrêtés, et ils reçoivent automatiquement des mises à jour des versions mineures et des correctifs de sécurité. Le redémarrage d'un pipeline permet de reprendre le traitement des nouvelles données entrantes.

> **Note**
>
> Si votre pipeline a une capacité excédentaire mais doit rester opérationnel, envisagez d'ajuster ses limites de capacité maximale plutôt que de l'arrêter et de le redémarrer.

Cela peut aider à gérer les coûts tout en garantissant que le pipeline continue de traiter les données de manière efficace. Pour en savoir plus, consultez the section called “Dimensionnement des pipelines”.

Les rubriques suivantes expliquent comment démarrer et arrêter des pipelines à l'aide des API AWS Management Console AWS CLI,, et OpenSearch Ingestion.

Rubriques

- Arrêt d'un pipeline Amazon OpenSearch Ingestion
- Démarrage d'un pipeline OpenSearch Amazon Ingestion

Arrêt d'un pipeline Amazon OpenSearch Ingestion

Pour utiliser un pipeline d' OpenSearch ingestion ou effectuer une administration, vous devez toujours commencer par un pipeline actif, puis arrêter le pipeline, puis le redémarrer. Lorsque votre pipeline est arrêté, vous ne serez pas facturé pour des heures d'ingestion EC2.

console

Arrêter un pipeline

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le panneau de navigation, choisissez Pipelines, choisissez un pipeline. Vous pouvez effectuer l'opération d'arrêt soit à partir de cette page, soit en accédant à la page de détails du pipeline que vous voulez arrêter.
3. Pour Actions, choisissez Arrêter le pipeline.

   Si un pipeline ne peut pas être arrêté et démarré, l'action Arrêter le pipeline n'est pas disponible.

AWS CLI

Pour arrêter un pipeline à partir de l' AWS CLI, appelez la commande stop-pipeline avec les paramètres suivants :

- `--pipeline-name`— nom du pipeline.

Example

```
aws osis stop-pipeline --pipeline-name my-pipeline
```

OpenSearch API d'ingestion

Pour arrêter un pipeline à partir de OpenSearch l'API Ingestion, appelez l'StopPipelineopération avec le paramètre suivant :

- `PipelineName`— nom du pipeline.

Démarrage d'un pipeline OpenSearch Amazon Ingestion

Le pipeline d' OpenSearch ingestion que vous démarrez est toujours un pipeline qui est déjà à l'état arrêté (ou « stopped »). Le pipeline conserve ses paramètres de configuration, notamment les limites de capacité, les paramètres réseau et les options de publication des journaux.

Le redémarrage d'un pipeline prend généralement plusieurs minutes.

console

Démarrer un pipeline

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le panneau de navigation, choisissez Pipelines, choisissez un pipeline. Vous pouvez effectuer l'opération de démarrage soit à partir de cette page, soit en accédant à la page de détails du pipeline que vous voulez démarrer.
3. Pour Actions, choisissez Démarrer le pipeline.

AWS CLI

Pour démarrer un pipeline à partir de l' AWS CLI, appelez la commande start-pipeline avec les paramètres suivants :

- `--pipeline-name`— nom du pipeline.

Example

```
aws osis start-pipeline --pipeline-name my-pipeline
```

OpenSearch API d'ingestion

Pour démarrer un pipeline d' OpenSearch ingestion à l'aide de OpenSearch l'API d'ingestion, appelez l'StartPipelineopération avec le paramètre suivant :

- `PipelineName`— nom du pipeline.

Suppression des pipelines OpenSearch Amazon Ingestion

Vous pouvez supprimer un pipeline Amazon OpenSearch Ingestion à l'aide de l' AWS Management Console API AWS CLI, de ou de l'API OpenSearch d'ingestion. Vous ne pouvez pas supprimer un pipeline dont le statut est `Creating` ou`Updating`.

console

Pour supprimer un pipeline

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, choisissez Pipelines.
3. Sélectionnez le pipeline que vous souhaitez supprimer, puis choisissez Actions, Supprimer.
4. Pour confirmer la suppression, choisissez Supprimer.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour supprimer un pipeline à l'aide du AWS CLI, envoyez une demande de suppression du pipeline :

```
aws osis delete-pipeline --pipeline-name "my-pipeline"
```

OpenSearch API d'ingestion

Pour supprimer un pipeline d' OpenSearch ingestion à l'aide de OpenSearch l'API d'ingestion, appelez l'DeletePipelineopération avec le paramètre suivant :

- `PipelineName`— nom du pipeline.

Plug-ins et options pris en charge pour les pipelines OpenSearch Amazon Ingestion

Amazon OpenSearch Ingestion prend en charge un sous-ensemble de sources, de processeurs et de récepteurs au sein de OpenSearch Data Prepper open source. En outre, OpenSearch Ingestion impose certaines contraintes aux options disponibles pour chaque plugin pris en charge. Les sections suivantes décrivent les plugins et les options associées pris en charge OpenSearch par Ingestion.

> **Note**
>
> OpenSearch Ingestion ne prend en charge aucun plug-in de mémoire tampon car elle configure automatiquement une mémoire tampon par défaut. Vous recevez une erreur de validation si vous incluez une mémoire tampon dans la configuration de votre pipeline.

Rubriques

- Plugins pris en charge
- Processeurs apatrides et processeurs dynamiques
- Exigences et contraintes de configuration

Plugins pris en charge

OpenSearch Ingestion prend en charge les plug-ins Data Prepper suivants :

Les sources :

- DocumentDB
- DynamoDB

- HTTP
- Kafka
- Kinesis
- OpenSearch

- OTel journaux
- OTel métriques
- OTel trace
- S3

Processeurs :

- Ajouter des entrées
- Regrouper
- Détecteur d'anomalies
- AWS Lambda
- Convertir le type d'entrée
- Copier les valeurs
- CSV
- Date
- Retard
- Décompresser
- Supprimer des entrées
- Disséquer
- Supprimer des événements
- Aplatir
- Géo-IP
- Grok
- Valeur clé
- Liste à cartographier
- Chaîne en minuscules
- De la carte à la liste
- Événement de mutation (série de processeurs)
- Chaîne mutante (série de processeurs)
- Masquer

- OTel métriques
- OTel groupe de traces
- OTel trace
- Analyser l'ion
- Analyser le JSON
- Analyser le XML
- Renommer les clés
- itinéraires
- Sélectionnez les entrées
- Cartographie des services
- Événement partagé
- Chaîne fendue
- Convertisseur de chaînes
- Chaîne de substitution
- Redirecteur Trace Peer
- Translate
- Corde à découper
- Tronquer
- Chaîne majuscule
- Agent utilisateur
- Écrire du JSON

Éviers :

- OpenSearch(prend en charge OpenSearch Service, OpenSearch Serverless et Elasticsearch 6.8 ou version ultérieure)
- S3

Codecs Sink :

- Avro
- DJSON

- JSON
- Parquet

Processeurs apatrides et processeurs dynamiques

Les processeurs sans état exécutent des opérations telles que les transformations et le filtrage, tandis que les processeurs statiques exécutent des opérations telles que des agrégations, qui mémorisent le résultat de l'exécution précédente. OpenSearch L'ingestion prend en charge les processeurs dynamiques Aggregate et Service-MAP. Tous les autres processeurs pris en charge sont apatrides.

Pour les pipelines contenant uniquement des processeurs apatrides, la limite de capacité maximale est de 96 ingestions. OCUs Si un pipeline contient des processeurs dynamiques, la limite de capacité maximale est de 48 OCUs ingestions. Toutefois, si la mise en mémoire tampon persistante est activée sur un pipeline, il peut avoir un maximum de 384 ingestions OCUs avec uniquement des processeurs sans état, ou de 192 ingestions OCUs s'il contient des processeurs statiques. Pour de plus amples informations, veuillez consulter the section called “Dimensionnement des pipelines”.

End-to-end l'accusé de réception n'est pris en charge que pour les processeurs sans état. Pour de plus amples informations, veuillez consulter the section called “End-to-end accusé de réception”.

Exigences et contraintes de configuration

Sauf indication contraire ci-dessous, toutes les options décrites dans la référence de configuration de Data Prepper pour les plug-ins pris en charge répertoriés ci-dessus sont autorisées dans les pipelines OpenSearch d'ingestion. Les sections suivantes expliquent les contraintes imposées OpenSearch par Ingestion à certaines options du plugin.

> **Note**
>
> OpenSearch Ingestion ne prend en charge aucun plug-in de mémoire tampon car elle configure automatiquement une mémoire tampon par défaut. Vous recevez une erreur de validation si vous incluez une mémoire tampon dans la configuration de votre pipeline.

De nombreuses options sont configurées et gérées en interne par OpenSearch Ingestion, telles que `authentication` et`acm_certificate_arn`. D'autres options, telles que `thread_count` et`request_timeout`, ont un impact sur les performances si elles sont modifiées manuellement. Par

conséquent, ces valeurs sont définies en interne afin de garantir des performances optimales de vos pipelines.

Enfin, certaines options ne peuvent pas être transmises à OpenSearch Ingestion, comme `ism_policy_file` et`sink_template`, car il s'agit de fichiers locaux lorsqu'ils sont exécutés dans Data Prepper open source. Ces valeurs ne sont pas prises en charge.

Rubriques

- Options générales en matière de pipeline
- Processeur Grok
- Source HTTP
- OpenSearch Sink
- OTel source des métriques, source de OTel trace et source OTel des journaux
- OTel processeur de groupe de traces
- OTel processeur de traçage
- Processeur Service-Map
- Source S3

Options générales en matière de pipeline

Les options générales de pipeline suivantes sont définies par OpenSearch Ingestion et ne sont pas prises en charge dans les configurations de pipeline :

- `workers`
- `delay`

Processeur Grok

Les options de processeur Grok suivantes ne sont pas prises en charge :

- `patterns_directories`
- `patterns_files_glob`

Source HTTP

Le plugin source HTTP présente les exigences et contraintes suivantes :

- L'`path`option est obligatoire. Le chemin est une chaîne telle que`/log/ingest`, qui représente le chemin de l'URI pour l'ingestion du journal. Ce chemin définit l'URI que vous utilisez pour envoyer des données au pipeline. Par exemple, `https://log-pipeline.us-west-2.osis.amazonaws.com/log/ingest`. Le chemin doit commencer par une barre oblique (/) et peut contenir les caractères spéciaux « - », « _ », « » . ', et'/', ainsi que l'`${pipelineName}`espace réservé.
- Les options de source HTTP suivantes sont définies par OpenSearch Ingestion et ne sont pas prises en charge dans les configurations de pipeline :
  - `port`
  - `ssl`
  - `ssl_key_file`
  - `ssl_certificate_file`
  - `aws_region`
  - `authentication`
  - `unauthenticated_health_check`
  - `use_acm_certificate_for_ssl`
  - `thread_count`
  - `request_timeout`
  - `max_connection_count`
  - `max_pending_requests`
  - `health_check_service`
  - `acm_private_key_password`
  - `acm_certificate_timeout_millis`
  - `acm_certificate_arn`

OpenSearch Sink

Le plugin OpenSearchsink présente les exigences et limites suivantes.

- L'`aws`option est obligatoire et doit contenir les options suivantes :
  - `sts_role_arn`
  - `region`
  - `hosts`

  - `serverless`(si le récepteur est une collection OpenSearch sans serveur)
- L'`sts_role_arn`option doit pointer vers le même rôle pour chaque récepteur dans un fichier de définition YAML.
- L'`hosts`option doit spécifier un point de terminaison OpenSearch de domaine de service ou un point de terminaison de collecte OpenSearch sans serveur. Vous ne pouvez pas spécifier de point de terminaison personnalisé pour un domaine ; il doit s'agir du point de terminaison standard.
- Si l'`hosts`option est un point de terminaison de collecte sans serveur, vous devez définir l'`serverless`option sur`true`. De plus, si votre fichier de définition YAML contient l'`index_type`option, elle doit être définie sur`management_disabled`, sinon la validation échoue.
- Les options suivantes ne sont pas prises en charge :
  - `username`
  - `password`
  - `cert`
  - `proxy`
  - `dlq_file`- Si vous souhaitez transférer les événements ayant échoué vers une file d'attente morte (DLQ), vous devez utiliser l'`dlq`option et spécifier un compartiment S3.
  - `ism_policy_file`
  - `socket_timeout`
  - `template_file`
  - `insecure`

OTel source des métriques, source de OTel trace et source OTel des journaux

Les plug-ins source de OTel métriques, source de OTel trace et source de OTel journaux présentent les exigences et limites suivantes :

- L'`path`option est obligatoire. Le chemin est une chaîne telle que`/log/ingest`, qui représente le chemin de l'URI pour l'ingestion du journal. Ce chemin définit l'URI que vous utilisez pour envoyer des données au pipeline. Par exemple, `https://log-pipeline.us-west-2.osis.amazonaws.com`*`/log/ingest`*. Le chemin doit commencer par une barre oblique (/) et peut contenir les caractères spéciaux « - », « _ », « » . ', et'/', ainsi que l'`${pipelineName}`espace réservé.
- Les options suivantes sont définies par OpenSearch Ingestion et ne sont pas prises en charge dans les configurations de pipeline :

- `port`
- `ssl`
- `sslKeyFile`
- `sslKeyCertChainFile`
- `authentication`
- `unauthenticated_health_check`
- `useAcmCertForSSL`
- `unframed_requests`
- `proto_reflection_service`
- `thread_count`
- `request_timeout`
- `max_connection_count`
- `acmPrivateKeyPassword`
- `acmCertIssueTimeOutMillis`
- `health_check_service`
- `acmCertificateArn`
- `awsRegion`

OTel processeur de groupe de traces

Le processeur de groupe de OTel traces présente les exigences et limites suivantes :

- L'`aws`option est obligatoire et doit contenir les options suivantes :
  - `sts_role_arn`
  - `region`
  - `hosts`
- L'`sts_role_arn`option spécifie le même rôle que le rôle de pipeline que vous spécifiez dans la configuration du OpenSearch récepteur.
- Les `insecure` options`username`, `passwordcert`, et ne sont pas prises en charge.
- L'`aws_sigv4`option est obligatoire et doit être définie sur true.

- L'`serverless`option du plugin Récepteur (OpenSearch Sink Plugin) n'est pas prise en charge. Le processeur de groupe de traces Otel ne fonctionne pas actuellement avec les collections OpenSearch sans serveur.
- Le nombre de `otel_trace_group` processeurs dans le corps de configuration du pipeline ne peut pas dépasser 8.

OTel processeur de traçage

Le processeur de OTel suivi présente les exigences et limites suivantes :

- La valeur de l'`trace_flush_interval`option ne peut pas dépasser 300 secondes.

Processeur Service-Map

Le processeur Service-Map est soumis aux exigences et limitations suivantes :

- La valeur de l'`window_duration`option ne peut pas dépasser 300 secondes.

Source S3

Le plugin source S3 est soumis aux exigences et limitations suivantes :

- L'`aws`option est obligatoire et doit contenir `region` des `sts_role_arn` options.
- La valeur de l'`records_to_accumulate`option ne peut pas dépasser 200.
- La valeur de l'`maximum_messages`option ne peut pas dépasser 10.
- Si est spécifiée, l'`disable_bucket_ownership_validation`option doit être définie sur false.
- Si elle est spécifiée, l'`input_serialization`option doit être définie sur`parquet`.

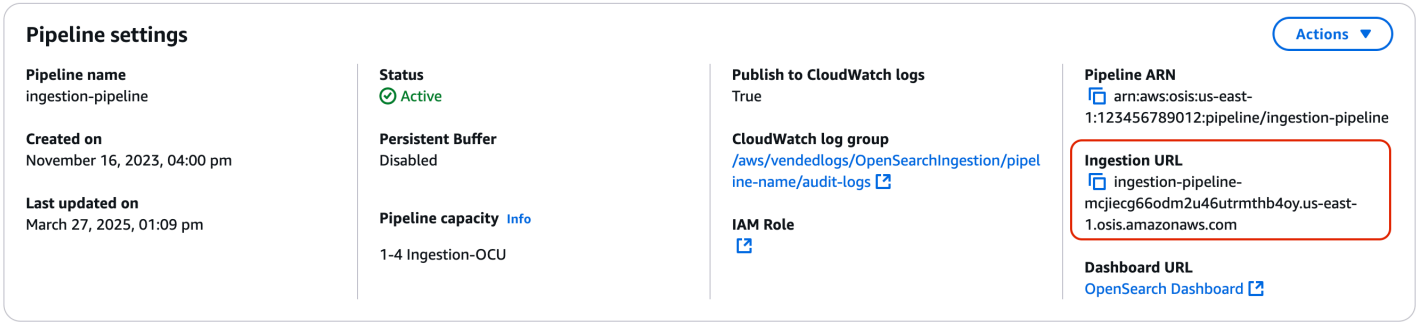
Intégration des pipelines OpenSearch Amazon Ingestion à d'autres services et applications

Pour réussir à ingérer des données dans un pipeline Amazon OpenSearch Ingestion, vous devez configurer votre application cliente (la source) pour envoyer les données au point de terminaison du pipeline. Votre source peut être des clients tels que les journaux Fluent Bit, le OpenTelemetry Collector ou un simple compartiment S3. La configuration exacte varie d'un client à l'autre.

Les différences importantes lors de la configuration de la source (par rapport à l'envoi de données directement à un domaine de OpenSearch service ou à une collection OpenSearch sans serveur) concernent le nom du AWS service (`osis`) et le point de terminaison hôte, qui doit être le point de terminaison du pipeline.

Construction du paramètre d'ingestion

Pour ingérer des données dans un pipeline, envoyez-les au point de terminaison d'ingestion. Pour localiser l'URL d'ingestion, accédez à la page des paramètres du pipeline et copiez l'URL d'ingestion.



Pour créer le point de terminaison d'ingestion complet pour les sources basées sur le pull, telles que le OTel traçage et OTel les métriques, ajoutez le chemin d'ingestion depuis la configuration de votre pipeline à l'URL d'ingestion.

Supposons, par exemple, que la configuration de votre pipeline comporte le chemin d'ingestion suivant :



Le point de terminaison d'ingestion complet, que vous spécifiez dans la configuration de votre client, prendra le format suivant :`https://`*`ingestion-pipeline-abcdefg`*`.us-east-1.osis.amazonaws.com`*`/my/test_path`*.

Création d'un rôle d'ingestion

Toutes les demandes adressées OpenSearch à Ingestion doivent être signées avec Signature Version 4. Au minimum, le rôle qui signe la demande doit être autorisé à effectuer

l'`osis:Ingest`action, ce qui lui permet d'envoyer des données à un pipeline d' OpenSearch ingestion.

Par exemple, la politique AWS Identity and Access Management (IAM) suivante permet au rôle correspondant d'envoyer des données vers un seul pipeline :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "osis:Ingest",
      "Resource": "arn:aws:osis:region:account-id:pipeline/pipeline-name"
    }
  ]
}
```

> **Note**
>
> Pour utiliser le rôle pour tous les pipelines, remplacez l'ARN de l'`Resource`élément par un caractère générique (*).

Fournir un accès à l'ingestion entre comptes

> **Note**
>
> Vous ne pouvez fournir un accès d'ingestion entre comptes que pour les pipelines publics, et non pour les pipelines VPC.

Il se peut que vous deviez ingérer des données dans un pipeline à partir d'un autre compte Compte AWS, tel qu'un compte hébergeant votre application source. Si le principal qui écrit dans un pipeline se trouve dans un compte différent de celui du pipeline lui-même, vous devez configurer le principal pour qu'il fasse confiance à un autre rôle IAM pour ingérer des données dans le pipeline.

Pour configurer les autorisations d'ingestion entre comptes

1. Créez le rôle d'ingestion avec `osis:Ingest` autorisation (décrit dans la section précédente) au même Compte AWS titre que le pipeline. Pour obtenir des instructions, consultez la section Création de rôles IAM.
2. Associez une politique de confiance au rôle d'ingestion qui permet au principal d'un autre compte de l'assumer :

   JSON

   ```
   {
     "Version": "2012-10-17",
     "Statement": [{
        "Effect": "Allow",
        "Principal": {
          "AWS": "arn:aws:iam::external-account-id:root"
         },
        "Action": "sts:AssumeRole"
     }]
   }
   ```

3. Dans l'autre compte, configurez votre application cliente (par exemple, Fluent Bit) pour qu'elle assume le rôle d'ingestion. Pour que cela fonctionne, le compte de l'application doit autoriser l'utilisateur ou le rôle de l'application à assumer le rôle d'ingestion.

   L'exemple de politique basée sur l'identité suivant permet au principal rattaché d'assumer à `ingestion-role` partir du compte de pipeline :

   JSON

   ```
   {
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Action": "sts:AssumeRole",
         "Resource": "arn:aws:iam::account-id:role/ingestion-role"
       }
     ]
   ```

```
}
```

L'application client peut ensuite utiliser l'AssumeRoleopération pour assumer `ingestion-role` et ingérer des données dans le pipeline associé.

Utilisation d'un pipeline OpenSearch d'ingestion avec Atlassian Services

Vous pouvez utiliser les plugins source Atlassian Jira et Confluence pour intégrer les données des services Atlassian dans votre pipeline d'ingestion. OpenSearch Ces intégrations vous permettent de créer une base de connaissances consultable unifiée en synchronisant des projets Jira complets et des espaces Confluence, tout en préservant la pertinence en temps réel grâce à une surveillance continue et à une synchronisation automatique des mises à jour.

Integrating with Jira

Transformez votre expérience Jira grâce à de puissantes fonctionnalités de recherche contextuelle en intégrant votre contenu Jira dans. OpenSearch Le plugin source Atlassian Jira de Data Prepper vous permet de créer une base de connaissances unifiée consultable en synchronisant des projets Jira complets, tout en préservant leur pertinence en temps réel grâce à une surveillance continue et à une synchronisation automatique des mises à jour. Cette intégration permet la synchronisation des données avec des options de filtrage flexibles pour des projets, des types de problèmes et des statuts spécifiques, garantissant ainsi que seules les informations dont vous avez besoin sont importées.

Pour garantir une connectivité sûre et fiable, le plugin prend en charge plusieurs méthodes d'authentification, y compris l'authentification de base par clé d'API, avec la sécurité supplémentaire de la gestion des informations d'identification à l'aide d'un secret stocké dans AWS Secrets Manager. OAuth2 Il propose également le renouvellement automatique des jetons pour un accès ininterrompu, garantissant ainsi un fonctionnement continu. Basée sur la version 2 de l'API d'Atlassian, cette intégration permet aux équipes de tirer des enseignements précieux de leurs données Jira grâce aux fonctionnalités OpenSearch de recherche avancées.

Integrating with Confluence

Améliorez la gestion des connaissances et les capacités de collaboration de votre équipe en intégrant le contenu Atlassian Confluence OpenSearch via le plugin source Confluence de Data Prepper. Cette intégration vous permet de créer un référentiel centralisé et consultable de connaissances collectives, améliorant ainsi la découverte d'informations et la productivité des

équipes. En synchronisant le contenu de Confluence et en surveillant en permanence les mises à jour, le plugin garantit que votre OpenSearch index reste complet up-to-date.

L'intégration offre des options de filtrage flexibles, vous permettant d'importer du contenu de manière sélective à partir d'espaces ou de types de pages spécifiques, en adaptant le contenu synchronisé aux besoins de votre organisation. Le plugin prend en charge à la fois les clés API de base et les méthodes OAuth2 d'authentification, avec la possibilité de gérer en toute sécurité les informations d'identification via AWS Secrets Manager. La fonction de renouvellement automatique des jetons du plugin garantit un accès ininterrompu et un fonctionnement sans faille. Basée sur l'API Confluence d'Atlassian, cette intégration permet aux équipes de tirer parti des fonctionnalités de recherche avancées OpenSearch de leur contenu Confluence, améliorant ainsi l'accessibilité et l'utilisation des informations au sein de l'organisation.

Rubriques

- Prérequis
- Configuration d'un rôle de pipeline
- Configuration du pipeline de connecteurs Jira
- Configuration du pipeline du connecteur Confluence
- Cohérence des données
- Limites
- Informations relatives CloudWatch aux connecteurs Atlassian
- Connexion d'un pipeline Amazon OpenSearch Ingestion à Atlassian Jira ou Confluence à l'aide de la version 2.0 OAuth

Prérequis

Avant de créer votre pipeline OpenSearch d'ingestion, procédez comme suit :

1. Préparez les informations d'identification de votre site Jira en choisissant l'une des options suivantes. OpenSearch L'ingestion ne nécessite qu'une `ReadOnly` autorisation du contenu.
   a. Option 1 : clé API — Connectez-vous à votre compte Atlassian et utilisez les informations de la rubrique suivante pour générer votre clé API :
      - Gérez les jetons d'API pour votre compte Atlassian

b. Option 2 : OAuth2 — Connectez-vous à votre compte Atlassian et utilisez les informations qu'il contient. the section called “Connexion d'un pipeline Amazon OpenSearch Ingestion à Atlassian Jira ou Confluence à l'aide de la version 2.0 OAuth ”

2. Créez un secret AWS Secrets Manager pour stocker les informations d'identification créées à l'étape précédente. Effectuez les choix suivants en suivant la procédure :

   - Pour Secret type (Type de secret), choisissez Other type of secret (Autre type de secret).
   - Pour les paires clé/valeur, créez les paires suivantes, en fonction du type d'autorisation sélectionné :

   API key

   ```
   {
      "username": user-name-usualy-email-id,
      "password": api-key
   }
   ```

   OAuth 2.0

   ```
   {
      "clientId": client-id
      "clientSecret": client-secret
      "accessKey": access-key
      "refreshKey": refresh-key
   }
   ```

   Après avoir créé le secret, copiez le nom de ressource Amazon (ARN) du secret. Vous allez l'inclure dans la politique d'autorisation des rôles du pipeline.

Configuration d'un rôle de pipeline

Le rôle transmis dans le pipeline doit être associé à la politique suivante pour pouvoir lire et écrire sur le secret créé dans la section des prérequis.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Sid": "SecretReadWrite",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetResourcePolicy",
                "secretsmanager:GetSecretValue",
                "secretsmanager:DescribeSecret",
                "secretsmanager:PutSecretValue",
                "secretsmanager:ListSecretVersionIds"
            ],
            "Resource": "secret-arn-created-in-step-2"
        }
    ]
}
```

Le rôle doit également être associé à une politique d'accès et d'écriture sur le récepteur que vous avez choisi. Par exemple, si vous OpenSearch le choisissez comme récepteur, la politique est similaire à la suivante :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "OpenSearchWritePolicy",
            "Effect": "Allow",
            "Action": "aoss:*",
            "Resource": "aoss-collection-arn"
        }
    ]
}
```

Configuration du pipeline de connecteurs Jira

Vous pouvez utiliser un plan Atlassian Jira préconfiguré pour créer ce pipeline. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Remplacez *placeholder values* par vos propres informations.

```
version: "2"
extension:
  aws:
    secrets:
      jira-account-credentials:
        secret_id: "secret-arn"
        region: "secret-region"
        sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
atlassian-jira-pipeline:
  source:
    jira:
      # We only support one host url for now
      hosts: ["jira-host-url"]
      acknowledgments: true
      authentication:
        # Provide one of the authentication method to use. Supported methods are
 'basic' and 'oauth2'.
        # For basic authentication, password is the API key that you generate using
 your jira account
        basic:
          username: ${{aws_secrets:jira-account-credentials:username}}
          password: ${{aws_secrets:jira-account-credentials:password}}
        # For OAuth2 based authentication, we require the following 4 key values stored
 in the secret
        # Follow atlassian instructions at the below link to generate these keys.
        # https://developer.atlassian.com/cloud/confluence/oauth-2-3lo-apps/
        # If you are using OAuth2 authentication, we also require, write permission to
 your AWS secret to
        # be able to write the renewed tokens back into the secret.
        # oauth2:
          # client_id: ${{aws_secrets:jira-account-credentials:clientId}}
          # client_secret: ${{aws_secrets:jira-account-credentials:clientSecret}}
          # access_token: ${{aws_secrets:jira-account-credentials:accessToken}}
          # refresh_token: ${{aws_secrets:jira-account-credentials:refreshToken}}
      filter:
        project:
          key:
            include:
              # This is not project name.
              # It is an alphanumeric project key that you can find under project
 details in Jira.
              - "project-key"
              - "project-key"
```

```
            # exclude:
              # - "project-key"
              # - "project-key"
        issue_type:
          include:
            - "issue-type"
            # - "Story"
            # - "Bug"
            # - "Task"
         # exclude:
             # - "Epic"
        status:
          include:
            - "ticket-status"
            # - "To Do"
            # - "In Progress"
            # - "Done"
         # exclude:
           # - "Backlog"

  sink:
    - opensearch:
        # Provide an Amazon OpenSearch Service domain endpoint
        hosts: [ "https://search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-
east-1.es.amazonaws.com" ]
        index: "index_${getMetadata(\"project\")}"
        # Ensure adding unique document id which is the unique ticket id in this case
        document_id: '${/id}'
        aws:
          # Provide a Role ARN with access to the domain. This role should have a trust
 relationship with osis-pipelines.amazonaws.com
          sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
          # Provide the region of the domain.
          region: "us-east-1"
          # Enable the 'serverless' flag if the sink is an Amazon OpenSearch Serverless
 collection
          serverless: false
          # serverless_options:
            # Specify a name here to create or update network policy for the serverless
 collection
            # network_policy_name: "network-policy-name"
        # Enable the 'distribution_version' setting if the Amazon OpenSearch Service
 domain is of version Elasticsearch 6.x
        # distribution_version: "es6"
```

```
        # Enable and switch the 'enable_request_compression' flag if the default
compression setting is changed in the domain.
        # See the section called “Compression des requêtes HTTP”
        # enable_request_compression: true/false
        # Optional: Enable the S3 DLQ to capture any failed requests in an S3 bucket.
Delete this entire block if you don't want a DLQ.
        dlq:
          s3:
            # Provide an S3 bucket
            bucket: "your-dlq-bucket-name"
            # Provide a key path prefix for the failed requests
            # key_path_prefix: "kinesis-pipeline/logs/dlq"
            # Provide the region of the bucket.
            region: "us-east-1"
            # Provide a Role ARN with access to the bucket. This role should have a
trust relationship with osis-pipelines.amazonaws.com
            sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
```

Clé des attributs de la source Jira :

1. hosts : votre URL Jira dans le cloud ou sur site. En général, cela ressemble à`https://your-domain-name.atlassian.net/`.
2. accusés de réception : pour garantir la livraison des données jusqu'au récepteur.
3. authentification : décrit comment vous souhaitez que le pipeline accède à votre instance Jira. Choisissez `Basic` ou `OAuth2` et spécifiez les attributs clés correspondants faisant référence aux clés de votre AWS secret.
4. filtre : cette section vous permet de sélectionner la partie de vos données Jira à extraire et à synchroniser.
   a. projet : Répertoriez les clés de projet que vous souhaitez synchroniser dans la `include` section. Sinon, listez les projets que vous souhaitez exclure dans la `exclude` section. Fournissez une seule des options d'inclusion ou d'exclusion à un moment donné.
   b. issue_type : types de problèmes spécifiques que vous souhaitez synchroniser. Suivez le modèle similaire `include` ou le `exclude` modèle qui répond à vos besoins. Notez que les pièces jointes apparaîtront sous forme de liens d'ancrage vers la pièce jointe d'origine, mais que le contenu de la pièce jointe ne sera pas extrait.
   c. status : filtre d'état spécifique que vous souhaitez appliquer à la requête d'extraction de données. Si vous le spécifie`include`, seuls les tickets présentant ces statuts seront

synchronisés. Si vous le spécifiez`exclude`, tous les tickets, à l'exception de ceux dont le statut d'exclusion est répertorié, seront synchronisés.

Configuration du pipeline du connecteur Confluence

Vous pouvez utiliser un plan Atlassian Confluence préconfiguré pour créer ce pipeline. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

```
version: "2"
extension:
  aws:
    secrets:
      confluence-account-credentials:
        secret_id: "secret-arn"
        region: "secret-region"
        sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
atlassian-confluence-pipeline:
  source:
    confluence:
      # We currently support only one host URL.
      hosts: ["confluence-host-url"]
      acknowledgments: true
      authentication:
        # Provide one of the authentication method to use. Supported methods are
 'basic' and 'oauth2'.
        # For basic authentication, password is the API key that you generate using
 your Confluence account
        basic:
          username: ${{aws_secrets:confluence-account-credentials:confluenceId}}
          password: ${{aws_secrets:confluence-account-
credentials:confluenceCredential}}
        # For OAuth2 based authentication, we require the following 4 key values stored
 in the secret
        # Follow atlassian instructions at the following link to generate these keys:
        # https://developer.atlassian.com/cloud/confluence/oauth-2-3lo-apps/
        # If you are using OAuth2 authentication, we also require write permission to
 your AWS secret to
        # be able to write the renewed tokens back into the secret.
        # oauth2:
          # client_id: ${{aws_secrets:confluence-account-credentials:clientId}}
          # client_secret: ${{aws_secrets:confluence-account-credentials:clientSecret}}
          # access_token: ${{aws_secrets:confluence-account-credentials:accessToken}}
```

```
          # refresh_token: ${{aws_secrets:confluence-account-credentials:refreshToken}}
      filter:
        space:
          key:
            include:
              # This is not space name.
              # It is a space key that you can find under space details in Confluence.
              - "space key"
              - "space key"
           # exclude:
             #  - "space key"
             #  - "space key"
        page_type:
          include:
            - "content type"
            # - "page"
            # - "blogpost"
            # - "comment"
         # exclude:
            # - "attachment"

  sink:
    - opensearch:
        # Provide an Amazon OpenSearch Service domain endpoint
        hosts: [ "https://search-mydomain-1a2a3a4a5a6a7a8a9a0a9a8a7a.us-
east-1.es.amazonaws.com" ]
         index: "index_${getMetadata(\"space\")}"
        # Ensure adding unique document id which is the unique ticket ID in this case.
        document_id: '${/id}'
        aws:
          # Provide the Amazon Resource Name (ARN) for a role with access to the
 domain. This role should have a trust relationship with osis-pipelines.amazonaws.com.
          sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
          # Provide the Region of the domain.
          region: "us-east-1"
          # Enable the 'serverless' flag if the sink is an Amazon OpenSearch Serverless
 collection
          serverless: false
          # serverless_options:
            # Specify a name here to create or update network policy for the serverless
 collection.
            # network_policy_name: "network-policy-name"
        # Enable the 'distribution_version' setting if the Amazon OpenSearch Service
 domain is of version Elasticsearch 6.x
```

```
        # distribution_version: "es6"
        # Enable and switch the 'enable_request_compression' flag if the default
compression setting is changed in the domain.
        # For more information, see the section called “Compression des requêtes HTTP”.
        # enable_request_compression: true/false
        # Optional: Enable the S3 DLQ to capture any failed requests in an S3 bucket.
Delete this entire block if you don't want a DLQ.
        dlq:
          s3:
            # Provide an S3 bucket
            bucket: "your-dlq-bucket-name"
            # Provide a key path prefix for the failed requests
            # key_path_prefix: "kinesis-pipeline/logs/dlq"
            # Provide the Rregion of the bucket.
            region: "us-east-1"
            # Provide the Amazon Resource Name (ARN) for a role with access to the
bucket. This role should have a trust relationship with osis-pipelines.amazonaws.com
            sts_role_arn: "arn:aws:iam::123456789012:role/Example-Role"
```

Attributs clés de la source Confluence :

1. hosts : votre URL Confluence dans le cloud ou sur site. En général, cela ressemble à `https://your-domain-name.atlassian.net/`
2. accusés de réception : pour garantir la livraison des données jusqu'au récepteur.
3. authentification : décrit comment vous souhaitez que le pipeline accède à votre instance Confluence. Choisissez `Basic` ou `OAuth2` et spécifiez les attributs clés correspondants faisant référence aux clés de votre AWS secret.
4. filtre : cette section vous permet de sélectionner la partie de vos données Confluence à extraire et à synchroniser.
   a. espace : Répertoriez les touches d'espace que vous souhaitez synchroniser dans la `include` section. Sinon, listez les espaces que vous souhaitez exclure dans la `exclude` section. Fournissez une seule des options d'inclusion ou d'exclusion à un moment donné.
   b. page_type : types de pages spécifiques (comme une page, un article de blog ou des pièces jointes) que vous souhaitez synchroniser. Suivez le modèle similaire `include` ou le `exclude` modèle qui répond à vos besoins. Notez que les pièces jointes apparaîtront sous forme de liens d'ancrage vers la pièce jointe d'origine, mais que le contenu de la pièce jointe ne sera pas extrait.

Cohérence des données

Sur la base des filtres spécifiés dans le pipeline YAML, les projets (ou espaces) sélectionnés seront extraits une fois et entièrement synchronisés avec le récepteur cible. Ensuite, la surveillance continue des modifications capturera les modifications au fur et à mesure qu'elles se produisent et mettra à jour les données dans le récepteur. Une exception est que le suivi des modifications synchronise uniquement les `update` actions, `create` et non les `delete` actions.

Limites

- Les actions de suppression effectuées par l'utilisateur ne seront pas synchronisées. Une fois enregistrées dans le récepteur, les données y resteront. Les mises à jour remplaceront le contenu existant par de nouvelles modifications si le mappage des identifiants est spécifié dans les paramètres du récepteur.
- Les instances locales utilisant d'anciennes versions du logiciel Atlassian qui ne prennent pas en charge les éléments suivants ne APIs sont pas compatibles avec cette source :
  - API de recherche Jira version 3
    - `rest/api/3/search`
    - `rest/api/3/issue`
  - Confluence
    - `wiki/rest/api/content/search`
    - `wiki/rest/api/content`
    - `wiki/rest/api/settings/systemInfo`

Informations relatives CloudWatch aux connecteurs Atlassian

Type : métriques du connecteur Jira

Source	Métrique	Type : Metric
acknowledgementSetSuccesses.nombre	Compteur	Si les accusés de réception sont activés, cette métrique indique le nombre de tickets synchronisés avec succès.

Source	Métrique	Type : Metric
acknowledgementSetFailures.nombre	Compteur	Si les accusés de réception sont activés, cette métrique indique le nombre de tickets qui n'ont pas pu être synchronisés.
CrawlingTime.AVG	Horloge	Le temps qu'il a fallu pour analyser tous les nouveaux changements.
ticketFetchLatency.avg	Horloge	La latence moyenne de l'API de récupération des tickets.
ticketFetchLatency.max	Horloge	La latence maximale de l'API de récupération des tickets.
Tickets demandés.Nombre	Compteur	Nombre de demandes de récupération de tickets effectuées.
ticketRequestedFailed.nombre	Compteur	Nombre de demandes de récupération de tickets ayant échoué.
ticketRequestedSuccess.nombre	Compteur	Nombre de demandes d'extraction de tickets réussies.
searchCallLatency.avg	Horloge	Latence moyenne des appels de l'API de recherche.
searchCallLatency.max	Horloge	Latence maximale des appels de l'API de recherche.

Source	Métrique	Type : Metric
searchResultsFound.nombre	Compteur	Nombre d'éléments trouvés lors d'un appel de recherche donné.
searchRequestFailed.nombre	Compteur	Nombre d'échecs d'appels de l'API de recherche.
Défaillances d'authentification. Nombre	Compteur	Nombre d'échecs d'authentification.

Type : métriques du connecteur Confluence

Source	Métrique	Type : Metric
acknowledgementSetSuccesses.nombre	Compteur	Si les accusés de réception sont activés, cette métrique indique le nombre de pages synchronisées avec succès.
acknowledgementSetFailures.nombre	Compteur	Si les accusés de réception sont activés, cette métrique indique le nombre de pages qui n'ont pas pu être synchronisées.
CrawlingTime.AVG	Horloge	Le temps qu'il a fallu pour analyser tous les nouveaux changements.
pageFetchLatency.avg	Horloge	Latence de l'API de récupération de contenu (moyenne).

Source	Métrique	Type : Metric
pageFetchLatency.max	Horloge	Latence de l'API de récupération de contenu (maximale).
Nombre de pages demandées	Compteur	Nombre d'appels de l'API de récupération de contenu.
pageRequestFailed.nombre	Compteur	Nombre de demandes infructueuses de l'API de récupération de contenu.
pageRequestedSuccess.nombre	Compteur	Nombre de demandes d'API de récupération de contenu réussies.
searchCallLatency.avg	Horloge	Latence moyenne des appels de l'API de recherche.
searchCallLatency.max	Horloge	Latence maximale des appels de l'API de recherche
searchResultsFound.nombre	Compteur	Nombre d'éléments trouvés lors d'un appel de recherche donné.
searchRequestsFailed.nombre	Compteur	Nombre d'échecs d'appels de l'API de recherche.
Défaillances d'authentification. Nombre	Compteur	Nombre d'échecs d'authentification.

Connexion d'un pipeline Amazon OpenSearch Ingestion à Atlassian Jira ou Confluence à l'aide de la version 2.0 OAuth

Utilisez les informations de cette rubrique pour vous aider à configurer et à connecter un pipeline Amazon OpenSearch Ingestion à un compte Jira ou Confluence à l'aide de l'authentification OAuth 2.0. Effectuez cette tâche lorsque vous terminez la the section called “Prérequis” procédure d'utilisation d'un pipeline d' OpenSearch ingestion avec Atlassian Services, mais que vous choisissez de ne pas utiliser les informations d'identification des clés d'API.

Rubriques

- Création d'une application d'intégration OAuth 2.0
- Génération et actualisation d'un jeton d'accès Atlassian Developer

Création d'une application d'intégration OAuth 2.0

Utilisez la procédure suivante pour créer une application d'intégration OAuth 2.0 sur le site Web d'Atlassian Developer.

Pour créer une application d'intégration OAuth 2.0

1. Connectez-vous à votre compte Atlassian Developer sur myapps/. https://developer.atlassian.com/console/
2. Choisissez Créer, intégration OAuth 2.0.
3. Dans Nom, entrez un nom pour identifier l'objectif de l'application.
4. Cochez la case J'accepte d'être lié par les conditions d'utilisation d'Atlassian destinées aux développeurs, puis choisissez Créer.
5. Dans le volet de navigation de gauche, choisissez Autorisation, puis Ajouter.
6. Pour l'URL de rappel, entrez n'importe quelle URL, telle que **`https://www.amazon.com`** ou**`https://www.example.com`**, puis choisissez Enregistrer les modifications.
7. Dans le volet de navigation de gauche, choisissez la page Permissions, puis dans la ligne correspondant à l'API Jira, choisissez Ajouter, puis choisissez Configurer. Sélectionnez toutes les autorisations de lecture de Classic Scopes (liste ci-dessous), puis sélectionnez Enregistrer
8. Choisissez l'onglet Étendues granulaires, puis sélectionnez Modifier les étendues pour ouvrir la boîte de dialogue Modifier l'API Jira.
9. Sélectionnez les autorisations pour le plugin source que vous utilisez :

Jira

```
read:audit-log:jira
read:issue:jira
read:issue-meta:jira
read:attachment:jira
read:comment:jira
read:comment.property:jira
read:field:jira
read:field.default-value:jira
read:field.option:jira
read:field-configuration-scheme:jira
read:field-configuration:jira
read:issue-link:jira
read:issue-link-type:jira
read:issue-link-type:jira
read:issue.remote-link:jira
read:issue.property:jira
read:resolution:jira
read:issue-details:jira
read:issue-type:jira
read:issue-worklog:jira
read:issue-field-values:jira
read:issue.changelog:jira
read:issue.transition:jira
read:issue.vote:jira
read:jira-expressions:jira
```

Confluence

```
read:content:confluence
read:content-details:confluence
read:space-details:confluence
read:audit-log:confluence
read:page:confluence
read:blogpost:confluence
read:custom-content:confluence
read:comment:confluence
read:space:confluence
read:space.property:confluence
read:space.setting:confluence
read:content.property:confluence
```

```
read:content.metadata:confluence
read:task:confluence
read:whiteboard:confluence
read:app-data:confluence
manage:confluence-configuration
```

10. Choisissez Enregistrer.

Pour des informations connexes, consultez Implémentation de la OAuth version 2.0 (3LO) et Détermination des étendues requises pour une opération sur le site Web d'Atlassian Developer.

Génération et actualisation d'un jeton d'accès Atlassian Developer

Utilisez la procédure suivante pour générer et actualiser un jeton d'accès Atlassian Developer sur le site Web d'Atlassian Developer.

Pour générer et actualiser un jeton d'accès Jira

1. Connectez-vous à votre compte Atlassian Developer sur myapps/. https://developer.atlassian.com/console/
2. Choisissez l'application dans laquelle vous l'avez crééethe section called “Création d'une application d'intégration OAuth 2.0”.
3. Dans le volet de navigation de gauche, sélectionnez Autorisation.
4. Copiez la valeur granulaire de l'URL d'autorisation de l'API Atlassian en bas de page et collez-la dans l'éditeur de texte de votre choix.

   Le format de l'URL est le suivant :

   ```
   https://auth.atlassian.com/authorize?
   audience=api.atlassian.com
   &client_id=YOUR_CLIENT_ID
   &scope=REQUESTED_SCOPE%20REQUESTED_SCOPE_TWO
   &redirect_uri=https://YOUR_APP_CALLBACK_URL
   &state=YOUR_USER_BOUND_VALUE
   &response_type=code
   &prompt=consent
   ```

5. En `state=YOUR_USER_BOUND_VALUE` effet, remplacez la valeur du paramètre par celle de votre choix, telle que state= » **`sample_text`** ».

Pour plus d'informations, voir À quoi sert le paramètre d'état ? sur le site Web d'Atlassian Developer.

6. Notez que la `scope` section répertorie les étendues granulaires que vous avez sélectionnées lors d'une tâche précédente. Par exemple : `scope=read%3Ajira-work%20read%3Ajira-user%20offline_access`

   `offline_access`indique que vous souhaitez générer un`refresh_token`.

7. Ouvrez une fenêtre de navigateur Web et entrez l'URL d'autorisation que vous avez copiée dans la barre d'adresse de la fenêtre du navigateur.
8. Lorsque la page cible s'ouvre, vérifiez que les informations sont correctes, puis choisissez Accepter pour être redirigé vers votre page d'accueil Jira ou Confluence.
9. Une fois la page d'accueil chargée, copiez l'URL de cette page. Il contient le code d'autorisation de votre application. Vous utilisez ce code pour générer votre jeton d'accès. La section complète qui suit `code=` est le code d'autorisation.
10. Utilisez la commande cURL suivante pour générer le jeton d'accès. Remplacez *`placeholder values`* par vos propres informations.

> **Tip**
>
> Vous pouvez également utiliser un service tiers tel que Postman.

```
curl --request POST --url 'https://auth.atlassian.com/oauth/token' \
--header 'Content-Type: application/json' \
--data '{"grant_type": "authorization_code",
"client_id": "YOUR_CLIENT_ID",
"client_secret": "YOUR_CLIENT_SECRET",
"code": "AUTHORIZATION_CODE",
"redirect_uri": "YOUR_CALLBACK_URL"}'
```

La réponse à cette commande inclut les valeurs pour `access_code` et`refresh_token`.

Utilisation d'un pipeline OpenSearch d'ingestion avec Amazon DynamoDB

Vous pouvez utiliser le plug-in DynamoDB pour diffuser des événements de table, tels que les créations, les mises à jour et les suppressions, vers des domaines OpenSearch Amazon Service

et des collections Amazon Serverless. OpenSearch Le pipeline utilise la capture des données de modification (CDC) pour un streaming à grande échelle et à faible latence.

Vous pouvez traiter les données DynamoDB avec ou sans capture initiale complète.

- Avec un instantané complet, DynamoDB point-in-time utilise la restauration (PITR) pour créer une sauvegarde et la télécharger sur Amazon S3. OpenSearch L'ingestion indexe ensuite l'instantané dans un ou plusieurs OpenSearch index. Pour garantir la cohérence, le pipeline synchronise toutes les modifications DynamoDB avec. OpenSearch Cette option nécessite que vous activiez à la fois les flux PITR et DynamoDB Streams.
- Sans capture instantanée : OpenSearch l'ingestion diffuse uniquement les nouveaux événements DynamoDB. Choisissez cette option si vous avez déjà un instantané ou si vous avez besoin d'un streaming en temps réel sans données historiques. Cette option nécessite que vous activiez uniquement DynamoDB Streams.

Pour plus d'informations, consultez la section Intégration de DynamoDB Zero-ETL à OpenSearch Amazon Service Amazon DynamoDB dans le manuel du développeur.

Rubriques

- Prérequis
- Étape 1 : configurer le rôle du pipeline
- Étape 2 : Création du pipeline
- Cohérence des données
- Types de données de mappage
- Limites
- CloudWatch Alarmes recommandées pour DynamoDB

Prérequis

Pour configurer votre pipeline, vous devez disposer d'une table DynamoDB sur laquelle DynamoDB Streams est activé. Votre stream doit utiliser le type de `NEW_IMAGE` stream view. Cependant, les pipelines OpenSearch d'ingestion peuvent également diffuser des événements `NEW_AND_OLD_IMAGES` si ce type de vue de flux correspond à votre cas d'utilisation.

Si vous utilisez des instantanés, vous devez également activer la point-in-time restauration sur votre table. Pour plus d'informations, consultez les sections Création d'une table, point-in-timeActivation de la restauration et Activation d'un flux dans le guide du développeur Amazon DynamoDB.

Étape 1 : configurer le rôle du pipeline

Une fois votre table DynamoDB configurée, configurez le rôle de pipeline que vous souhaitez utiliser dans la configuration de votre pipeline et ajoutez les autorisations DynamoDB suivantes dans le rôle :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "allowRunExportJob",
            "Effect": "Allow",
            "Action": [
                "dynamodb:DescribeTable",
                "dynamodb:DescribeContinuousBackups",
                "dynamodb:ExportTableToPointInTime"
            ],
            "Resource": [
                "arn:aws:dynamodb:region:account-id:table/my-table"
            ]
        },
        {
            "Sid": "allowCheckExportjob",
            "Effect": "Allow",
            "Action": [
                "dynamodb:DescribeExport"
            ],
            "Resource": [
                "arn:aws:dynamodb:region:account-id:table/my-table/export/*"
            ]
        },
        {
            "Sid": "allowReadFromStream",
            "Effect": "Allow",
            "Action": [
                "dynamodb:DescribeStream",
                "dynamodb:GetRecords",
```

```
                "dynamodb:GetShardIterator"
            ],
            "Resource": [
                "arn:aws:dynamodb:region:account-id:table/my-table/stream/*"
            ]
        },
        {
            "Sid": "allowReadAndWriteToS3ForExport",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:AbortMultipartUpload",
                "s3:PutObject",
                "s3:PutObjectAcl"
            ],
            "Resource": [
                "arn:aws:s3:::my-bucket/export-folder/*"
            ]
        }
    ]
}
```

Vous pouvez également utiliser une clé gérée par AWS KMS le client pour chiffrer les fichiers de données d'exportation. Pour déchiffrer les objets exportés, spécifiez `s3_sse_kms_key_id` l'ID de clé dans la configuration d'exportation du pipeline au format suivant :`arn:aws:kms:region:account-id:key/my-key-id`. La politique suivante inclut les autorisations requises pour utiliser une clé gérée par le client :

```
{
    "Sid": "allowUseOfCustomManagedKey",
    "Effect": "Allow",
    "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
    ],
    "Resource": arn:aws:kms:region:account-id:key/my-key-id
}
```

Étape 2 : Création du pipeline

Vous pouvez ensuite configurer un pipeline d' OpenSearch ingestion comme le suivant, qui spécifie DynamoDB comme source. Cet exemple de pipeline ingère des données provenant de `table-a` l'instantané PITR, suivies d'événements provenant de DynamoDB Streams. Une position de départ de `LATEST` indique que le pipeline doit lire les dernières données de DynamoDB Streams.

```
version: "2"
cdc-pipeline:
  source:
    dynamodb:
      tables:
      - table_arn: "arn:aws:dynamodb:region:account-id:table/table-a"
        export:
          s3_bucket: "my-bucket"
          s3_prefix: "export/"
        stream:
          start_position: "LATEST"
      aws:
        region: "us-east-1"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.region.es.amazonaws.com"]
      index: "${getMetadata(\"table-name\")}"
      index_type: custom
      normalize_index: true
      document_id: "${getMetadata(\"primary_key\")}"
      action: "${getMetadata(\"opensearch_action\")}"
      document_version: "${getMetadata(\"document_version\")}"
      document_version_type: "external"
```

Vous pouvez utiliser un plan DynamoDB préconfiguré pour créer ce pipeline. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Cohérence des données

OpenSearch L'ingestion prend en charge end-to-end la reconnaissance afin de garantir la durabilité des données. Lorsqu'un pipeline lit des instantanés ou des flux, il crée dynamiquement des partitions pour un traitement parallèle. Le pipeline marque une partition comme terminée lorsqu'il reçoit un accusé de réception après avoir ingéré tous les enregistrements du OpenSearch domaine ou de la collection.

Si vous souhaitez intégrer des données dans une collection de recherche OpenSearch sans serveur, vous pouvez générer un identifiant de document dans le pipeline. Si vous souhaitez intégrer des données dans une collection de séries chronologiques OpenSearch sans serveur, notez que le pipeline ne génère pas d'identifiant de document.

Un pipeline d' OpenSearch ingestion fait également correspondre les actions des événements entrants aux actions d'indexation groupées correspondantes pour faciliter l'ingestion de documents. Cela garantit la cohérence des données, de sorte que chaque modification de données dans DynamoDB est conciliée avec les modifications de document correspondantes. OpenSearch

Types de données de mappage

OpenSearch Le service fait correspondre dynamiquement les types de données de chaque document entrant au type de données correspondant dans DynamoDB. Le tableau suivant montre comment OpenSearch Service mappe automatiquement les différents types de données.

Type de données	OpenSearch	DynamoDB
Nombre	OpenSearch mappe automatiquement les données numériques. S'il s'agit d'un nombre entier, OpenSearch mappez-le sous forme de valeur longue. Si le nombre est fractionnaire, il est OpenSearch mappé sous forme de valeur flottante. OpenSearch mappe dynamique ment divers attributs en fonction du premier document envoyé. Si vous avez plusieurs types de données pour le même attribut dans DynamoDB, par exemple un nombre entier et un nombre fractionnaire, le mappage risque d'échouer. Par exemple, si votre premier document possède un attribut qui est un nombre entier et qu'un document ultérieur	DynamoDB prend en charge les nombres.

Type de données	OpenSearch	DynamoDB
	possède le même attribut sous forme de nombre fractionnaire, le second document OpenSearch ne parvient pas à ingérer. Dans ces cas, vous devez fournir un modèle de mappage explicite, tel que le suivant : `{ "template": { "mappings": { "properties": { "MixedNumberAttribute": { "type": "float" } } } } }` Si vous avez besoin d'une double précision, utilisez le mappage de champs de type chaîne. Il n'existe aucun type numérique équivalent prenant en charge 38 chiffres de précision dans OpenSearch.	

Type de données	OpenSearch	DynamoDB
Ensemble de numéros	OpenSearch mappe automatiquement un ensemble de nombres dans un tableau de valeurs longues ou de valeurs flottantes. Comme pour les nombres scalaires, cela dépend du fait que le premier nombre ingéré est un nombre entier ou un nombre fractionnaire. Vous pouvez fournir des mappages pour des ensembles de nombres de la même manière que vous mappez des chaînes scalaires.	DynamoDB prend en charge les types qui représentent des ensembles de nombres.
Chaîne	OpenSearch mappe automatiquement les valeurs des chaînes sous forme de texte. Dans certaines situations, telles que les valeurs énumérées, vous pouvez mapper le type de mot clé. L'exemple suivant montre comment mapper un attribut DynamoDB `PartType` nommé à un mot clé. OpenSearch `{ "template": { "mappings": { "properties": { "PartType": { "type": "keyword" } } } } }`	DynamoDB prend en charge les chaînes de caractères.

Type de données	OpenSearch	DynamoDB
Set de cordes	OpenSearch mappe automatiquement un ensemble de chaînes dans un tableau de chaînes. Vous pouvez fournir des mappages pour des ensembles de chaînes de la même manière que vous mappez des chaînes scalaires.	DynamoDB prend en charge les types qui représentent des ensembles de chaînes.
Binaire	OpenSearch mappe automatiquement les données binaires sous forme de texte. Vous pouvez fournir un mappage pour les écrire sous forme de champs binaires OpenSearch. L'exemple suivant montre comment mapper un attribut DynamoDB `ImageData` nommé à OpenSearch un champ binaire. `{ "template": { "mappings": { "properties": { "ImageData": { "type": "binary" } } } } }`	DynamoDB prend en charge les attributs de type binaire.

Type de données	OpenSearch	DynamoDB
Ensemble binaire	OpenSearch mappe automatiquement un ensemble binaire dans un tableau de données binaires sous forme de texte. Vous pouvez fournir des mappages pour des ensembles de nombres de la même manière que vous mappez un binaire scalaire.	DynamoDB prend en charge les types qui représentent des ensembles de valeurs binaires.
Booléen	OpenSearch mappe un type booléen DynamoDB en un type booléen. OpenSearch	DynamoDB prend en charge les attributs de type booléen.
Null	OpenSearch peut ingérer des documents dont le type DynamoDB est nul. Elle enregistre la valeur sous forme de valeur nulle dans le document. Il n'existe aucun mappage pour ce type, et ce champ n'est ni indexé ni consultable. Si le même nom d'attribut est utilisé pour un type nul, puis change ultérieur ement pour un type différent, tel qu'une chaîne, OpenSearch crée un mappage dynamique pour la première valeur non nulle. Les valeurs suivantes peuvent toujours être des valeurs nulles DynamoDB.	DynamoDB prend en charge les attributs de type nul.

Type de données	OpenSearch	DynamoDB
Map	OpenSearch mappe les attributs de mappage DynamoDB aux champs imbriqués. Les mêmes mappages s'appliquent dans un champ imbriqué. L'exemple suivant fait correspondre une chaîne d'un champ imbriqué à un mot clé saisi dans OpenSearch : `{ "template": { "mappings": { "properties": { "AdditionalDescriptions": { "properties": { "PartType": { "type": "keyword" } } } } } } }`	DynamoDB prend en charge les attributs de type de carte.

Type de données	OpenSearch	DynamoDB
Liste	OpenSearch fournit des résultats différents pour les listes DynamoDB, en fonction du contenu de la liste. Lorsqu'une liste contient tous le même type de types scalaires (par exemple, une liste de toutes les chaînes), elle OpenSearch ingère la liste sous forme de tableau de ce type. Cela fonctionn e pour les types chaîne, numérique, booléen et nul. Les restrictions pour chacun de ces types sont les mêmes que celles pour un scalaire de ce type. Vous pouvez également fournir des mappages pour des listes de cartes en utilisant le même mappage que celui que vous utiliseriez pour une carte. Vous ne pouvez pas fournir de liste de types mixtes.	DynamoDB prend en charge les attributs de type liste.

Type de données	OpenSearch	DynamoDB
Définir	OpenSearch fournit des résultats différents pour les ensembles DynamoDB en fonction du contenu de l'ensemble. Lorsqu'un ensemble contient tous le même type de types scalaires (par exemple, un ensemble de toutes les chaînes), il OpenSearch ingère l'ensembl e sous forme de tableau de ce type. Cela fonctionne pour les types chaîne, numérique, booléen et nul. Les restricti ons pour chacun de ces types sont les mêmes que celles pour un scalaire de ce type. Vous pouvez également fournir des mappages pour des ensembles de cartes en utilisant le même mappage que celui que vous utiliseriez pour une carte. Vous ne pouvez pas fournir un ensemble de types mixtes.	DynamoDB prend en charge les types qui représentent des ensembles.

Nous vous recommandons de configurer la file d'attente des lettres mortes (DLQ) dans votre OpenSearch pipeline d'ingestion. Si vous avez configuré la file d'attente, le OpenSearch service envoie tous les documents défaillants qui ne peuvent pas être ingérés en raison d'échecs de mappage dynamique vers la file d'attente.

En cas d'échec des mappages automatiques, vous pouvez utiliser `template_type` et `template_content` dans la configuration de votre pipeline pour définir des règles de mappage explicites. Vous pouvez également créer des modèles de mappage directement dans votre domaine de recherche ou votre collection avant de démarrer le pipeline.

Limites

Tenez compte des limites suivantes lorsque vous configurez un pipeline d' OpenSearch ingestion pour DynamoDB :

- L'intégration d' OpenSearch ingestion avec DynamoDB ne prend actuellement pas en charge l'ingestion entre régions. Votre table DynamoDB OpenSearch et votre pipeline d'ingestion doivent être identiques. Région AWS
- Votre table DynamoDB OpenSearch et votre pipeline d'ingestion doivent être identiques. Compte AWS
- Un pipeline d' OpenSearch ingestion ne prend en charge qu'une seule table DynamoDB comme source.
- DynamoDB Streams ne stocke les données dans un journal que pendant 24 heures maximum. Si l'ingestion à partir d'un instantané initial d'une grande table prend 24 heures ou plus, il y aura une perte de données initiale. Pour atténuer cette perte de données, estimez la taille de la table et configurez les unités de calcul appropriées pour les pipelines d' OpenSearch ingestion.

CloudWatch Alarmes recommandées pour DynamoDB

Les CloudWatch mesures suivantes sont recommandées pour surveiller les performances de votre pipeline d'ingestion. Ces indicateurs peuvent vous aider à identifier la quantité de données traitées à partir des exportations, le nombre d'événements traités à partir des flux, les erreurs lors du traitement des exportations et des événements des flux, ainsi que le nombre de documents écrits vers la destination. Vous pouvez configurer des CloudWatch alarmes pour effectuer une action lorsque l'une de ces mesures dépasse une valeur spécifiée pendant une durée spécifiée.

Métrique	Description
`dynamodb-pipeline.BlockingBuffer.bufferUsage.value`	Indique la quantité de mémoire tampon utilisée.
`dynamodb-pipeline.dynamodb.activeExportS3ObjectConsumers.value`	Indique le nombre total de OCUs ceux qui traitent activement des objets Amazon S3 pour l'exportation.
`dynamodb-pipeline.dynamodb.bytesProcessed.count`	Nombre d'octets traités à partir de la source DynamoDB.

Métrique	Description
dynamodb-pipeline.dynamodb.changeEventsProcessed.count	Nombre d'événements de modification traités à partir du flux DynamoDB.
dynamodb-pipeline.dynamodb.changeEventsProcessingErrors.count	Nombre d'erreurs liées aux événements de modification traités par DynamoDB.
dynamodb-pipeline.dynamodb.exportJobFailure.count	Nombre de tentatives de soumission de tâches d'exportation qui ont échoué.
dynamodb-pipeline.dynamodb.exportJobSuccess.count	Nombre de tâches d'exportation soumises avec succès.
dynamodb-pipeline.dynamodb.exportRecordsProcessed.count	Nombre total d'enregistrements traités à partir de l'exportation.
dynamodb-pipeline.dynamodb.exportRecordsTotal.count	Nombre total d'enregistrements exportés depuis DynamoDB, essentiel pour le suivi des volumes d'exportation de données.
dynamodb-pipeline.dynamodb.exportS3ObjectsProcessed.count	Nombre total de fichiers de données d'exportation qui ont été traités avec succès depuis Amazon S3.
dynamodb-pipeline.opensearch.bulkBadRequestErrors.count	Nombre d'erreurs lors de demandes groupées dues à une demande mal formée.
dynamodb-pipeline.opensearch.bulkRequestLatency.avg	Latence moyenne pour les demandes d'écriture en masse adressées à OpenSearch.
dynamodb-pipeline.opensearch.bulkRequestNotFoundErrors.count	Nombre de demandes groupées qui ont échoué car les données cibles sont introuvables.
dynamodb-pipeline.opensearch.bulkRequestNumberOfRetries.count	Nombre de tentatives effectuées par les pipelines OpenSearch d'ingestion pour écrire un OpenSearch cluster.

Métrique	Description
`dynamodb-pipeline.opensearch.bulkRequestSizeBytes.sum`	Taille totale en octets de toutes les demandes groupées adressées à OpenSearch.
`dynamodb-pipeline.opensearch.documentErrors.count`	Nombre d'erreurs lors de l'envoi de documents à OpenSearch. Les documents à l'origine des erreurs seront envoyés à DLQ.
`dynamodb-pipeline.opensearch.documentsSuccess.count`	Nombre de documents écrits avec succès dans un OpenSearch cluster ou une collection.
`dynamodb-pipeline.opensearch.documentsSuccessFirstAttempt.count`	Nombre de documents indexés avec succès OpenSearch lors de la première tentative.
`dynamodb-pipeline.opensearch.documentsVersionConflictErrors.count`	Nombre d'erreurs dues à des conflits de versions dans les documents pendant le traitement.
`dynamodb-pipeline.opensearch.PipelineLatency.avg`	Latence moyenne du pipeline d' OpenSearch ingestion pour traiter les données en lisant depuis la source et en écrivant vers la destination.
`dynamodb-pipeline.opensearch.PipelineLatency.max`	Latence maximale du pipeline d' OpenSearch ingestion pour traiter les données en lisant depuis la source jusqu'à l'écriture de la destination.
`dynamodb-pipeline.opensearch.recordsIn.count`	Nombre d'enregistrements ingérés avec succès. OpenSearch Cette métrique est essentielle pour suivre le volume de données traitées et stockées.
`dynamodb-pipeline.opensearch.s3.dlqS3RecordsFailed.count`	Nombre d'enregistrements qui n'ont pas pu être écrits dans DLQ.

Métrique	Description
`dynamodb-pipeline.opensearch.s3.dlqS3RecordsSuccess.count`	Nombre d'enregistrements écrits dans DLQ.
`dynamodb-pipeline.opensearch.s3.dlqS3RequestLatency.count`	Nombre de mesures de latence pour les demandes adressées à la file d'attente des lettres mortes Amazon S3.
`dynamodb-pipeline.opensearch.s3.dlqS3RequestLatency.sum`	Latence totale pour toutes les demandes adressées à la file d'attente des lettres mortes Amazon S3
`dynamodb-pipeline.opensearch.s3.dlqS3RequestSizeBytes.sum`	Taille totale en octets de toutes les demandes envoyées à la file d'attente de lettres mortes Amazon S3.
`dynamodb-pipeline.recordsProcessed.count`	Nombre total d'enregistrements traités dans le pipeline, indicateur clé du débit global.
`dynamodb.changeEventsProcessed.count`	Aucun enregistrement n'est collecté à partir des flux DynamoDB. Cela peut être dû à l'absence d'activité sur la table, à une exportation en cours ou à un problème d'accès aux flux DynamoDB.
`dynamodb.exportJobFailure.count`	La tentative de déclenchement d'une exportation vers S3 a échoué.
`dynamodb-pipeline.opensearch.bulkRequestInvalidInputErrors.count`	Nombre d'erreurs liées aux demandes groupées OpenSearch dues à une saisie non valide, crucial pour le suivi de la qualité des données et des problèmes opérationnels.

Métrique	Description
`opensearch.EndToEndLatency.avg`	La latence de bout en bout est supérieure à celle souhaitée pour la lecture à partir de flux DynamoDB. Cela peut être dû à un OpenSearch cluster sous-dimensionné ou à une capacité OCU maximale du pipeline trop faible pour le débit WCU de la table DynamoDB. Cette latence de bout en bout sera élevée après une exportation et devrait diminuer au fil du temps à mesure qu'elle rattrape les derniers flux DynamoDB.

Utilisation d'un pipeline d' OpenSearch ingestion avec Amazon DocumentDB

Vous pouvez utiliser le plug-in DocumentDB pour diffuser les modifications apportées aux documents, telles que les créations, les mises à jour et les suppressions, vers Amazon Service. OpenSearch Le pipeline prend en charge la capture des données de modification (CDC), si disponible, ou le sondage d'API pour un streaming à grande échelle et à faible latence.

Vous pouvez traiter les données avec ou sans capture initiale complète. Un instantané complet capture l'intégralité d'une collection Amazon DocumentDB et la télécharge sur Amazon S3. Le pipeline envoie ensuite les données à un ou plusieurs OpenSearch index. Après avoir ingéré l'instantané, le pipeline synchronise les modifications en cours pour maintenir la cohérence et finit par rattraper les mises à jour en temps quasi réel.

Si vous disposez déjà d'un instantané complet provenant d'une autre source, ou si vous devez uniquement traiter de nouveaux événements, vous pouvez diffuser sans instantané. Dans ce cas, le pipeline lit directement les flux de modifications d'Amazon DocumentDB sans chargement groupé initial.

Si vous activez le streaming, vous devez activer un flux de modifications sur votre collection Amazon DocumentDB. Toutefois, si vous effectuez uniquement un chargement complet ou une exportation, vous n'avez pas besoin d'un flux de modifications.

Prérequis

Avant de créer votre pipeline OpenSearch d'ingestion, effectuez les étapes suivantes :

1. Créez un cluster Amazon DocumentDB autorisé à lire les données en suivant les étapes décrites dans Créer un cluster Amazon DocumentDB dans le guide du développeur Amazon DocumentDB. Si vous utilisez l'infrastructure CDC, configurez votre cluster Amazon DocumentDB pour publier des flux de modifications.
2. Activez le protocole TLS sur votre cluster Amazon DocumentDB.
3. Configurez un VPC CIDR d'un espace d'adressage privé à utiliser avec Ingestion. OpenSearch
4. Configurez l'authentification sur votre cluster Amazon DocumentDB avec. AWS Secrets Manager Activez la rotation des secrets en suivant les étapes décrites dans Rotation automatique des mots de passe pour Amazon DocumentDB. Pour plus d'informations, consultez Accès aux bases de données à l'aide du contrôle d'accès et de la sécurité basés sur les rôles dans Amazon DocumentDB.
5. Si vous utilisez un flux de modifications pour vous abonner aux modifications de données de votre collection Amazon DocumentDB, évitez les pertes de données en prolongeant la période de conservation jusqu'à 7 jours à l'aide du `change_stream_log_retention_duration` paramètre. Les événements des flux de modifications sont stockés pendant 3 heures, par défaut, après l'enregistrement de l'événement, ce qui n'est pas suffisant pour les collections volumineuses. Pour modifier la période de rétention du flux de modifications, consultez la section Modification de la durée de conservation du journal du flux de modifications.
6. Créez un domaine OpenSearch de service ou une collection OpenSearch Serverless. Pour plus d'informations, consultez the section called “ Création de domaines OpenSearch de service” et the section called “Créer des collections”.
7. Associez une politique basée sur les ressources à votre domaine ou une politique d'accès aux données à votre collection. Ces politiques d'accès permettent à OpenSearch Ingestion d'écrire des données depuis votre cluster Amazon DocumentDB vers votre domaine ou votre collection.

   L'exemple de politique d'accès au domaine suivant permet au rôle de pipeline, que vous créez à l'étape suivante, d'écrire des données dans un domaine. Assurez-vous de le mettre à jour `resource` avec votre propre ARN.

   JSON

```
{
  "Version": "2012-10-17",
```

```
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::pipeline-account-id:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:region:account-id:domain/domain-name"
      ]
    }
  ]
}
```

Pour créer un rôle IAM doté des autorisations appropriées pour accéder aux données d'écriture de la collection ou du domaine, consultezthe section called “Configuration des rôles et des utilisateurs”.

Étape 1 : configurer le rôle du pipeline

Une fois les prérequis de votre pipeline Amazon DocumentDB définis, configurez le rôle de pipeline que vous souhaitez utiliser dans la configuration de votre pipeline et ajoutez les autorisations Amazon DocumentDB suivantes dans le rôle :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "allowS3ListObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::s3-bucket"
            ],
```

```
            "Condition": {
                "StringLike": {
                    "s3:prefix": "s3-prefix/*"
                }
            }
        },
        {
            "Sid": "allowReadAndWriteToS3ForExportStream",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObject"
            ],
            "Resource": [
                "arn:aws:s3:::s3-bucket/s3-prefix/*"
            ]
        },
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": ["arn:aws:secretsmanager:region:account-id:secret:secret-
name"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:*:account-id:network-interface/*",
                "arn:aws:ec2:*:account-id:subnet/*",
                "arn:aws:ec2:*:account-id:security-group/*"
            ]
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
               "StringEquals":
                    {
                        "aws:RequestTag/OSISManaged": "true"
                    }
            }
        }
    ]
}
```

Vous devez fournir les EC2 autorisations Amazon ci-dessus sur le rôle IAM que vous utilisez pour créer le pipeline d' OpenSearch ingestion, car le pipeline utilise ces autorisations pour créer et supprimer une interface réseau dans votre VPC. Le pipeline ne peut accéder au cluster Amazon DocumentDB que via cette interface réseau.

Étape 2 : Création du pipeline

Vous pouvez ensuite configurer un pipeline d' OpenSearch ingestion comme le suivant, qui spécifie Amazon DocumentDB comme source. Notez que pour renseigner le nom de l'index, la `getMetadata` fonction l'utilise *`documentdb_collection`* comme clé de métadonnées. Si vous souhaitez utiliser un autre nom d'index sans la `getMetadata` méthode, vous pouvez utiliser la configuration`index: "`*`my_index_name`*`"`.

```
version: "2"
documentdb-pipeline:
  source:
    documentdb:
      acknowledgments: true
      host: "https://docdb-cluster-id.us-east-1.docdb.amazonaws.com"
      port: 27017
      authentication:
        username: ${aws_secrets:secret:username}
        password: ${aws_secrets:secret:password}
      aws:
      s3_bucket: "bucket-name"
      s3_region: "bucket-region"
      s3_prefix: "path" #optional path for storing the temporary data
      collections:
        - collection: "dbname.collection"
          export: true
          stream: true
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      index: "${getMetadata(\"documentdb_collection\")}"
      index_type: custom
      document_id: "${getMetadata(\"primary_key\")}"
      action: "${getMetadata(\"opensearch_action\")}"
      document_version: "${getMetadata(\"document_version\")}"
      document_version_type: "external"
extension:
  aws:
    secrets:
      secret:
        secret_id: "my-docdb-secret"
        region: "us-east-1"
        refresh_interval: PT1H
```

Vous pouvez utiliser un plan Amazon DocumentDB préconfiguré pour créer ce pipeline. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Si vous utilisez le AWS Management Console pour créer votre pipeline, vous devez également l'attacher à votre VPC afin d'utiliser Amazon DocumentDB comme source. Pour ce faire, recherchez la section Options du réseau source, cochez la case Attacher au VPC et choisissez votre CIDR parmi

l'une des options par défaut fournies. Vous pouvez utiliser n'importe quel CIDR à partir d'un espace d'adressage privé tel que défini dans la RFC 1918 Best Current Practice.

Pour fournir un CIDR personnalisé, sélectionnez Autre dans le menu déroulant. Pour éviter toute collision d'adresses IP entre OpenSearch Ingestion et Amazon DocumentDB, assurez-vous que le CIDR VPC Amazon DocumentDB est différent du CIDR pour l'ingestion. OpenSearch

Pour plus d'informations, consultez Configuration de l'accès VPC pour un pipeline.

Cohérence des données

Le pipeline garantit la cohérence des données en interrogeant ou en recevant en permanence les modifications du cluster Amazon DocumentDB et en mettant à jour les documents correspondants dans l' OpenSearchindex.

OpenSearch L'ingestion prend en charge end-to-end la reconnaissance afin de garantir la durabilité des données. Lorsqu'un pipeline lit des instantanés ou des flux, il crée dynamiquement des partitions pour un traitement parallèle. Le pipeline marque une partition comme terminée lorsqu'il reçoit un accusé de réception après avoir ingéré tous les enregistrements du OpenSearch domaine ou de la collection.

Si vous souhaitez intégrer des données dans une collection de recherche OpenSearch sans serveur, vous pouvez générer un identifiant de document dans le pipeline. Si vous souhaitez intégrer des données dans une collection de séries chronologiques OpenSearch sans serveur, notez que le pipeline ne génère pas d'identifiant de document. Vous devez donc l'omettre `document_id: "${getMetadata(\"primary_key\")}"` dans la configuration de votre récepteur de pipeline.

Un pipeline d' OpenSearch ingestion fait également correspondre les actions des événements entrants aux actions d'indexation groupées correspondantes pour faciliter l'ingestion de documents. Cela permet de maintenir la cohérence des données, de sorte que chaque modification de données dans Amazon DocumentDB soit conciliée avec les modifications de document correspondantes dans. OpenSearch

Types de données de mappage

OpenSearch Le service mappe dynamiquement les types de données de chaque document entrant au type de données correspondant dans Amazon DocumentDB. Le tableau suivant montre comment OpenSearch Service mappe automatiquement les différents types de données.

Type de données	OpenSearch	Amazon DocumentDB
Entier	OpenSearch mappe automatiq uement les valeurs entières d'Amazon DocumentDB en nombres entiers. OpenSearch OpenSearch mappe dynamiquement le champ en fonction du premier document envoyé. Si vous avez plusieurs types de données pour le même attribut dans Amazon DocumentDB, le mappage automatique risque d'échouer. Par exemple, si votre premier document possède un attribut long et qu'un document ultérieur possède le même attribut sous forme de nombre entier, le second document OpenSearch ne parvient pas à être ingéré. Dans ces cas, vous devez fournir un modèle de mappage explicite qui choisit le type de numéro le plus flexible, tel que le suivant : `{ "template": { "mappings": { "properties": { "MixedNumberField": { "type": "float" } } } } }`	Amazon DocumentDB prend en charge les nombres entiers.

Type de données	OpenSearch	Amazon DocumentDB
Long	OpenSearch mappe automatiq uement les valeurs longues d'Amazon DocumentDB en valeurs longues. OpenSearch OpenSearch mappe dynamiquement le champ en fonction du premier document envoyé. Si vous avez plusieurs types de données pour le même attribut dans Amazon DocumentDB, le mappage automatique risque d'échouer. Par exemple, si votre premier document possède un attribut long et qu'un document ultérieur possède le même attribut sous forme de nombre entier, le second document OpenSearch ne parvient pas à être ingéré. Dans ces cas, vous devez fournir un modèle de mappage explicite qui choisit le type de numéro le plus flexible, tel que le suivant : `{ "template": { "mappings": { "properties": { "MixedNumberField": { "type": "float" } } } } }`	Amazon DocumentDB prend en charge les fichiers longs.

Type de données	OpenSearch	Amazon DocumentDB
Chaîne	OpenSearch mappe automatiquement les valeurs des chaînes sous forme de texte. Dans certaines situations, telles que les valeurs énumérées, vous pouvez mapper le type de mot clé. L'exemple suivant montre comment mapper un attribut Amazon DocumentDB nommé `PartType` à un OpenSearch mot clé. `{ "template": { "mappings": { "properties": { "PartType": { "type": "keyword" } } } } }`	Amazon DocumentDB prend en charge les chaînes de caractères.

Type de données	OpenSearch	Amazon DocumentDB
Double	OpenSearch mappe automatiquement les valeurs doubles d'Amazon DocumentDB en OpenSearch doubles. OpenSearch mappe dynamiquement le champ en fonction du premier document envoyé. Si vous avez plusieurs types de données pour le même attribut dans Amazon DocumentDB, le mappage automatique risque d'échouer. Par exemple, si votre premier document possède un attribut long et qu'un document ultérieur possède le même attribut sous forme de nombre entier, le second document OpenSearch ne parvient pas à être ingéré. Dans ces cas, vous devez fournir un modèle de mappage explicite qui choisit le type de numéro le plus flexible, tel que le suivant : `{ "template": { "mappings": { "properties": { "MixedNumberField": { "type": "float" } } } } }`	Amazon DocumentDB prend en charge les doublons.

Type de données	OpenSearch	Amazon DocumentDB
Date	Par défaut, la date correspond à un entier dans OpenSearch. Vous pouvez définir un modèle de mappage personnalisé pour associer une date à une OpenSearch date. `{ "template": { "mappings": { "properties": { "myDateField": { "type": "date", "format": "epoch_second" } } } } }`	Amazon DocumentDB prend en charge les dates.

Type de données	OpenSearch	Amazon DocumentDB
Horodatage	Par défaut, l'horodatage correspond à un entier dans. OpenSearch Vous pouvez définir un modèle de mappage personnalisé pour associer une date à une OpenSearch date. `{ "template": { "mappings": { "properties": { "myTimestampField": { "type": "date", "format": "epoch_second" } } } } }`	Amazon DocumentDB prend en charge les horodatages.
Booléen	OpenSearch mappe un type booléen Amazon DocumentDB en un type booléen. OpenSearch	Amazon DocumentDB prend en charge les attributs de type booléen.

Type de données	OpenSearch	Amazon DocumentDB
Décimal	OpenSearch mappe les attributs de mappage Amazon DocumentDB aux champs imbriqués. Les mêmes mappages s'appliquent dans un champ imbriqué. L'exemple suivant fait correspondre une chaîne d'un champ imbriqué à un mot clé saisi dans OpenSearch : `{ "template": { "mappings": { "properties": { "myDecimalField": { "type": "double" } } } } }` Grâce à ce mappage personnalisé, vous pouvez interroger et agréger le champ avec une précision à deux niveaux. La valeur d'origine conserve toute la précision des `_source` propriétés du OpenSearch document. Sans ce mappage, OpenSearch utilise le texte par défaut.	Amazon DocumentDB prend en charge les nombres décimaux.
Expression régulière	Le type regex crée des champs imbriqués. Il s'agit notamment *<myFieldName>* `.pattern` de et*<myFieldName>* `.options`.	Amazon DocumentDB prend en charge les expressions régulières.

Type de données	OpenSearch	Amazon DocumentDB
Données binaires	OpenSearch mappe automatiquement les données binaires Amazon DocumentDB en OpenSearch texte. Vous pouvez fournir un mappage pour les écrire sous forme de champs binaires OpenSearch. L'exemple suivant montre comment mapper un champ Amazon DocumentDB nommé `imageData` à un champ OpenSearch binaire. `{ "template": { "mappings": { "properties": { "imageData": { "type": "binary" } } } } }`	Amazon DocumentDB prend en charge les champs de données binaires.
ObjectId	Les champs dotés d'un type d'ObjectID correspondent aux champs de OpenSearch texte. La valeur sera la représentation sous forme de chaîne de l'ObjectID.	Amazon DocumentDB prend en charge les ObjectID.

Type de données	OpenSearch	Amazon DocumentDB
Null	OpenSearch peut ingérer des documents avec le type nul Amazon DocumentDB. Elle enregistre la valeur sous forme de valeur nulle dans le document. Il n'existe aucun mappage pour ce type, et ce champ n'est ni indexé ni consultable. Si le même nom d'attribut est utilisé pour un type nul, puis change ultérieur ement pour un type différent, tel qu'une chaîne, OpenSearch crée un mappage dynamique pour la première valeur non nulle. Les valeurs suivantes peuvent toujours être des valeurs nulles Amazon DocumentDB.	Amazon DocumentDB prend en charge les champs de type nul.
Non défini	OpenSearch peut ingérer des documents dont le type est indéfini Amazon DocumentDB. Elle enregistre la valeur sous forme de valeur nulle dans le document. Il n'existe aucun mappage pour ce type, et ce champ n'est ni indexé ni consultable. Si le même nom de champ est utilisé pour un type non défini, puis change ultérieurement pour un type différent, tel qu'une chaîne, OpenSearch crée un mappage dynamique pour la première valeur non définie. Les valeurs suivantes peuvent toujours être des valeurs non définies d'Amazon DocumentDB.	Amazon DocumentDB prend en charge les champs de type non définis.

Type de données	OpenSearch	Amazon DocumentDB
MinKey	OpenSearch peut ingérer des documents de type Amazon DocumentDB MinKey. Elle enregistre la valeur sous forme de valeur nulle dans le document. Il n'existe aucun mappage pour ce type, et ce champ n'est ni indexé ni consultable. Si le même nom de champ est utilisé pour un type MinKey puis change ultérieurement en un autre type, tel qu'une chaîne, OpenSearch crée un mappage dynamique pour la première valeur autre que MinKey. Les valeurs suivantes peuvent toujours être des valeurs MinKey d'Amazon DocumentDB.	Amazon DocumentDB prend en charge les champs de type MinKey.
MaxKey	OpenSearch peut ingérer des documents avec le type Amazon DocumentDB MaxKey. Elle enregistre la valeur sous forme de valeur nulle dans le document. Il n'existe aucun mappage pour ce type, et ce champ n'est ni indexé ni consultable. Si le même nom de champ est utilisé pour un type MaxKey puis change ultérieurement en un autre type, tel qu'une chaîne, OpenSearch crée un mappage dynamique pour la première valeur autre que MaxKey. Les valeurs suivantes peuvent toujours être des valeurs MaxKey d'Amazon DocumentDB.	Amazon DocumentDB prend en charge les champs de type MaxKey.

Nous vous recommandons de configurer la file d'attente des lettres mortes (DLQ) dans votre OpenSearch pipeline d'ingestion. Si vous avez configuré la file d'attente, le OpenSearch service envoie tous les documents défaillants qui ne peuvent pas être ingérés en raison d'échecs de mappage dynamique vers la file d'attente.

En cas d'échec des mappages automatiques, vous pouvez utiliser `template_type` et `template_content` dans la configuration de votre pipeline pour définir des règles de mappage explicites. Vous pouvez également créer des modèles de mappage directement dans votre domaine de recherche ou votre collection avant de démarrer le pipeline.

Limites

Tenez compte des limites suivantes lorsque vous configurez un pipeline d' OpenSearch ingestion pour Amazon DocumentDB :

- L'intégration d' OpenSearch ingestion avec Amazon DocumentDB ne prend actuellement pas en charge l'ingestion entre régions. Votre cluster Amazon DocumentDB et votre pipeline OpenSearch d'ingestion doivent être identiques. Région AWS
- L'intégration d' OpenSearch ingestion avec Amazon DocumentDB ne prend actuellement pas en charge l'ingestion entre comptes. Votre cluster Amazon DocumentDB et votre pipeline OpenSearch d'ingestion doivent être identiques. Compte AWS
- Un pipeline d' OpenSearch ingestion ne prend en charge qu'un seul cluster Amazon DocumentDB comme source.
- L'intégration d' OpenSearch ingestion avec Amazon DocumentDB prend spécifiquement en charge les clusters basés sur des instances Amazon DocumentDB. Il ne prend pas en charge les clusters élastiques Amazon DocumentDB.
- L'intégration d' OpenSearch ingestion est uniquement prise en charge en AWS Secrets Manager tant que mécanisme d'authentification pour votre cluster Amazon DocumentDB.
- Vous ne pouvez pas mettre à jour la configuration du pipeline existante pour ingérer des données provenant d'une autre base de données ou d'une autre collection. Vous devez plutôt créer un nouveau pipeline.

CloudWatch Alarmes recommandées

Pour de meilleures performances, nous vous recommandons d'utiliser les CloudWatch alarmes suivantes lorsque vous créez un pipeline d' OpenSearch ingestion pour accéder à un cluster Amazon DocumentDB en tant que source.

CloudWatch Alarme	Description
<pipeline-name> .DocumentDB.Informations d'identification modifiées	Cette métrique indique la fréquence à laquelle AWS les secrets font l'objet d'une rotation.
<pipeline-name> .documentdb. executorRefreshErrors	Cette métrique indique les échecs d'actualisation AWS des secrets.
<pipeline-name> .documentdb. exportRecordsTotal	Cette métrique indique le nombre d'enregistrements exportés depuis Amazon DocumentDB.
<pipeline-name> .documentdb. exportRecordsProcessed	Cette métrique indique le nombre d'enregistrements traités par le pipeline OpenSearch d'ingestion.
<pipeline-name> .documentdb. exportRecordProcessingErreurs	Cette métrique indique le nombre d'erreurs de traitement dans un pipeline d' OpenSearch ingestion lors de la lecture des données d'un cluster Amazon DocumentDB.
<pipeline-name> .documentdb. exportRecordsSuccessTotal	Cette métrique indique le nombre total d'enregistrements d'exportation traités avec succès.
<pipeline-name> .documentdb. exportRecordsFailedTotal	Cette métrique indique le nombre total d'enregistrements d'exportation qui n'ont pas pu être traités.
<pipeline-name> .Document DB. Octets reçus	Cette métrique indique le nombre total d'octets reçus par un pipeline d' OpenSearch ingestion.
<pipeline-name> .DocumentDB.Octets traités	Cette métrique indique le nombre total d'octets traités par un pipeline d' OpenSearch ingestion.
<pipeline-name> .documentdb. exportPartitionQueryTotal	Cette métrique indique le total de la partition d'exportation.

CloudWatch Alarme	Description
<pipeline-name> .documentdb. streamRecordsSuccessTotal	Cette métrique indique le nombre d'enregistrements traités avec succès à partir du flux.
<pipeline-name> .documentdb. streamRecordsFailedTotal	Cette métrique indique le nombre total d'enregistrements n'ayant pas pu être traités à partir du flux.

Utilisation d'un pipeline d' OpenSearch ingestion avec Confluent Cloud Kafka

Vous pouvez utiliser un pipeline d' OpenSearch ingestion pour diffuser des données depuis des clusters Confluent Cloud Kafka vers des domaines Amazon OpenSearch Service et des collections OpenSearch Serverless. OpenSearch Ingestion prend en charge les configurations de réseau public et privé pour le streaming de données depuis les clusters Confluent Cloud Kafka vers des domaines ou des collections gérés par OpenSearch Service ou OpenSearch Serverless.

Connectivité aux clusters Kafka publics de Confluent Cloud

Vous pouvez utiliser des pipelines d' OpenSearch ingestion pour migrer les données d'un cluster Confluent Cloud Kafka avec une configuration publique, ce qui signifie que le nom DNS du domaine peut être résolu publiquement. Pour ce faire, configurez un pipeline d' OpenSearch ingestion avec le cluster Kafka public Confluent Cloud comme source et OpenSearch Service ou OpenSearch Serverless comme destination. Cela traite vos données de streaming depuis un cluster source autogéré vers un domaine ou une collection de destination AWS géré.

Prérequis

Avant de créer votre pipeline OpenSearch d'ingestion, effectuez les étapes suivantes :

1. Créez un cluster de clusters Confluent Cloud Kafka faisant office de source. Le cluster doit contenir les données que vous souhaitez ingérer dans OpenSearch Service.
2. Créez un domaine OpenSearch de service ou une collection OpenSearch sans serveur vers lequel vous souhaitez migrer les données. Pour plus d’informations, consultez the section called “ Création de domaines OpenSearch de service” et the section called “Créer des collections”.

3. Configurez l'authentification sur votre cluster Confluent Cloud Kafka avec. AWS Secrets Manager Activez la rotation des secrets en suivant les étapes décrites dans Rotation AWS Secrets Manager des secrets.
4. Associez une politique basée sur les ressources à votre domaine ou une politique d'accès aux données à votre collection. Ces politiques d'accès permettent à OpenSearch Ingestion d'écrire des données de votre cluster autogéré vers votre domaine ou votre collection.

   L'exemple de politique d'accès au domaine suivant permet au rôle de pipeline, que vous créez à l'étape suivante, d'écrire des données dans un domaine. Assurez-vous de le mettre à jour `resource` avec votre propre ARN.

   JSON

   ```
   {
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
           "AWS": "arn:aws:iam::pipeline-account-id:role/pipeline-role"
         },
         "Action": [
           "es:DescribeDomain",
           "es:ESHttp*"
         ],
         "Resource": [
           "arn:aws:es:region:account-id:domain/domain-name"
         ]
       }
     ]
   }
   ```

   Pour créer un rôle IAM doté des autorisations appropriées pour accéder aux données d'écriture de la collection ou du domaine, consultezthe section called “Configuration des rôles et des utilisateurs”.

Étape 1 : configurer le rôle du pipeline

Après avoir défini les prérequis de votre pipeline de cluster Confluent Cloud Kafka, configurez le rôle de pipeline que vous souhaitez utiliser dans la configuration de votre pipeline et ajoutez l'autorisation

d'écrire dans un domaine de OpenSearch service ou une collection OpenSearch sans serveur, ainsi que l'autorisation de lire les secrets de Secrets Manager.

Les autorisations suivantes sont nécessaires pour gérer l'interface réseau :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:*:account-id:network-interface/*",
                "arn:aws:ec2:*:account-id:subnet/*",
                "arn:aws:ec2:*:account-id:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [ "ec2:CreateTags" ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
```

```
            "Condition": {
               "StringEquals": { "aws:RequestTag/OSISManaged": "true" }
            }
        }
    ]
}
```

L'autorisation requise pour lire les secrets du AWS Secrets Manager service est la suivante :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": ["secretsmanager:GetSecretValue"],
            "Resource": ["arn:aws:secretsmanager:region:account-
id:secret:,secret-name"]
        }
    ]
}
```

Les autorisations suivantes sont nécessaires pour écrire sur un domaine Amazon OpenSearch Service :

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::account-id:role/pipeline-role"
      },
      "Action": ["es:DescribeDomain", "es:ESHttp*"],
      "Resource": "arn:aws:es:region:account-id:domain/domain-name/*"
    }
  ]
}
```

Étape 2 : Création du pipeline

Vous pouvez ensuite configurer un pipeline d' OpenSearch ingestion comme celui-ci, qui spécifie votre Confluent Cloud Kafka comme source.

Vous pouvez spécifier plusieurs domaines OpenSearch de service comme destinations pour vos données. Cette fonctionnalité permet le routage conditionnel ou la réplication des données entrantes dans plusieurs domaines OpenSearch de service.

Vous pouvez également migrer les données d'un cluster Confluent Kafka source vers une collection VPC OpenSearch sans serveur. Assurez-vous de fournir une politique d'accès au réseau dans la configuration du pipeline. Vous pouvez utiliser un registre de schémas Confluent pour définir un schéma Confluent.

```
version: "2"
kafka-pipeline:
  source:
    kafka:
      encryption:
        type: "ssl"
      topics:
        - name: "topic-name"
          group_id: "group-id"
      bootstrap_servers:
        - "bootstrap-server.us-east-1.aws.private.confluent.cloud:9092"
      authentication:
        sasl:
          plain:
            username: ${aws_secrets:confluent-kafka-secret:username}
            password: ${aws_secrets:confluent-kafka-secret:password}
      schema:
        type: confluent
        registry_url: https://my-registry.us-east-1.aws.confluent.cloud
        api_key: "${{aws_secrets:schema-secret:schema_registry_api_key}}"
        api_secret: "${{aws_secrets:schema-secret:schema_registry_api_secret}}"
        basic_auth_credentials_source: "USER_INFO"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
  aws:
    secrets:
```

```
      confluent-kafka-secret:
        secret_id: "my-kafka-secret"
        region: "us-east-1"
      schema-secret:
        secret_id: "my-self-managed-kafka-schema"
        region: "us-east-1"
```

Vous pouvez utiliser un plan préconfiguré pour créer ce pipeline. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Connectivité aux clusters Confluent Cloud Kafka dans un VPC

Vous pouvez également utiliser des pipelines OpenSearch d'ingestion pour migrer les données d'un cluster Confluent Cloud Kafka exécuté dans un VPC. Pour ce faire, configurez un pipeline d' OpenSearch ingestion avec un cluster Confluent Cloud Kafka comme source et OpenSearch Service ou OpenSearch Serverless comme destination. Cela traite vos données de streaming depuis un cluster source Confluent Cloud Kafka vers un domaine ou une AWS collection de destination géré.

OpenSearch Ingestion prend en charge les clusters Confluent Cloud Kafka configurés dans tous les modes réseau pris en charge dans Confluent. Les modes de configuration réseau suivants sont pris en charge en tant que source dans OpenSearch Ingestion :

- AWS Appairage de VPC
- AWS PrivateLink pour les clusters dédiés
- AWS PrivateLink pour les clusters d'entreprise
- AWS Transit Gateway

Prérequis

Avant de créer votre pipeline OpenSearch d'ingestion, effectuez les étapes suivantes :

1. Créez un cluster Confluent Cloud Kafka avec une configuration réseau VPC contenant les données que vous souhaitez ingérer dans Service. OpenSearch
2. Créez un domaine OpenSearch de service ou une collection OpenSearch sans serveur vers lequel vous souhaitez migrer les données. Pour plus d'informations, voir Pour plus d'informations, voir the section called “ Création de domaines OpenSearch de service” etthe section called “Créer des collections”.

3. Configurez l'authentification sur votre cluster Confluent Cloud Kafka avec. AWS Secrets Manager Activez la rotation des secrets en suivant les étapes décrites dans Rotation AWS Secrets Manager des secrets.
4. Obtenez l'ID du VPC qui a accès au cluster Confluent Cloud Kafka. Choisissez le VPC CIDR à utiliser par Ingestion. OpenSearch

> **Note**
>
> Si vous utilisez le AWS Management Console pour créer votre pipeline, vous devez également attacher votre pipeline d' OpenSearch ingestion à votre VPC afin d'utiliser le cluster Confluent Cloud Kafka. Pour ce faire, recherchez la section Configuration réseau, cochez la case Attacher au VPC et choisissez votre CIDR parmi l'une des options par défaut fournies, ou sélectionnez le vôtre. Vous pouvez utiliser n'importe quel CIDR à partir d'un espace d'adressage privé tel que défini dans la RFC 1918 Best Current Practice. Pour fournir un CIDR personnalisé, sélectionnez Autre dans le menu déroulant. Pour éviter une collision d'adresses IP entre OpenSearch l'ingestion et l'adresse autogérée OpenSearch, assurez-vous que le CIDR OpenSearch VPC autogéré est différent du CIDR pour l'ingestion. OpenSearch

5. Associez une politique basée sur les ressources à votre domaine ou une politique d'accès aux données à votre collection. Ces politiques d'accès permettent à OpenSearch Ingestion d'écrire des données de votre cluster autogéré vers votre domaine ou votre collection.

> **Note**
>
> Si vous utilisez Confluent Cloud Kafka AWS PrivateLink pour connecter votre Confluent, vous devez configurer les options DHCP du VPC. Les noms d'hôte DNS et la résolution DNS doivent être activés.
> Plus précisément, utilisez les valeurs du jeu d'options suivantes :
>
> ```
> domain-name: aws.private.confluent.cloud
> domain-name-servers: AmazonProvidedDNS
> ```
>
> Cette modification garantit que la résolution DNS pour le point de PrivateLink terminaison Confluent fonctionne correctement au sein du VPC.

L'exemple de politique d'accès au domaine suivant permet au rôle de pipeline, que vous créez à l'étape suivante, d'écrire des données dans un domaine. Assurez-vous de le mettre à jour `resource` avec votre propre ARN.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::pipeline-account-id:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:region:account-id:domain/domain-name"
      ]
    }
  ]
}
```

Pour créer un rôle IAM doté des autorisations appropriées pour accéder aux données d'écriture de la collection ou du domaine, consultezthe section called “Configuration des rôles et des utilisateurs”.

Étape 1 : configurer le rôle du pipeline

Une fois les prérequis de votre pipeline configurés, configurez le rôle de pipeline que vous souhaitez utiliser dans la configuration de votre pipeline et ajoutez-y les autorisations suivantes :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": ["arn:aws:secretsmanager:region:account-id:secret:secret-
name"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:*:account-id:network-interface/*",
                "arn:aws:ec2:*:account-id:subnet/*",
                "arn:aws:ec2:*:account-id:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
```

```
            "Condition": {
               "StringEquals":
                    {
                        "aws:RequestTag/OSISManaged": "true"
                    }
            }
        }
    ]
}
```

Vous devez fournir les EC2 autorisations Amazon ci-dessus sur le rôle IAM que vous utilisez pour créer le pipeline d' OpenSearch ingestion, car le pipeline utilise ces autorisations pour créer et supprimer une interface réseau dans votre VPC. Le pipeline ne peut accéder au cluster Kafka que via cette interface réseau.

Étape 2 : Création du pipeline

Vous pouvez ensuite configurer un pipeline d' OpenSearch ingestion comme le suivant, qui spécifie Kafka comme source.

Vous pouvez spécifier plusieurs domaines OpenSearch de service comme destinations pour vos données. Cette fonctionnalité permet le routage conditionnel ou la réplication des données entrantes dans plusieurs domaines OpenSearch de service.

Vous pouvez également migrer les données d'un cluster Confluent Kafka source vers une collection VPC OpenSearch sans serveur. Assurez-vous de fournir une politique d'accès au réseau dans la configuration du pipeline. Vous pouvez utiliser un registre de schémas Confluent pour définir un schéma Confluent.

```
 version: "2"
kafka-pipeline:
  source:
    kafka:
      encryption:
        type: "ssl"
      topics:
        - name: "topic-name"
          group_id: "group-id"
      bootstrap_servers:
        - "bootstrap-server.us-east-1.aws.private.confluent.cloud:9092"
      authentication:
```

```
        sasl:
          plain:
            username: ${aws_secrets:confluent-kafka-secret:username}
            password: ${aws_secrets:confluent-kafka-secret:password}
      schema:
        type: confluent
        registry_url: https://my-registry.us-east-1.aws.confluent.cloud
        api_key: "${{aws_secrets:schema-secret:schema_registry_api_key}}"
        api_secret: "${{aws_secrets:schema-secret:schema_registry_api_secret}}"
        basic_auth_credentials_source: "USER_INFO"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
      index: "confluent-index"
extension:
  aws:
    secrets:
      confluent-kafka-secret:
        secret_id: "my-kafka-secret"
        region: "us-east-1"
      schema-secret:
        secret_id: "my-self-managed-kafka-schema"
        region: "us-east-2"
```

Utilisation d'un pipeline OpenSearch d'ingestion avec Amazon Managed Streaming for Apache Kafka

Vous pouvez utiliser le plugin Kafka pour intégrer les données d'Amazon Managed Streaming for Apache Kafka (Amazon MSK) OpenSearch dans votre pipeline d'ingestion. Avec Amazon MSK, vous pouvez créer et exécuter des applications qui utilisent Apache Kafka pour traiter les données de streaming. OpenSearch Ingestion utilise AWS PrivateLink pour se connecter à Amazon MSK. Vous pouvez ingérer des données provenant de clusters Amazon MSK et Amazon MSK Serverless. La seule différence entre les deux processus réside dans les étapes préalables que vous devez suivre avant de configurer votre pipeline.

Rubriques

- Prérequis Amazon MSK provisionnés
- Conditions préalables requises pour Amazon MSK Serverless

- Étape 1 : Configuration d'un rôle de pipeline
- Étape 2 : Création du pipeline
- Étape 3 : (Facultatif) Utiliser le registre des AWS Glue schémas
- Étape 4 : (Facultatif) Configurer les unités de calcul recommandées (OCUs) pour le pipeline Amazon MSK

Prérequis Amazon MSK provisionnés

Avant de créer votre pipeline OpenSearch d'ingestion, effectuez les étapes suivantes :

1. Créez un cluster provisionné par Amazon MSK en suivant les étapes décrites dans la section Création d'un cluster dans le guide du développeur Amazon Managed Streaming for Apache Kafka. Pour le type de courtier, choisissez n'importe quelle option à l'exception des `t3` types, car ceux-ci ne sont pas pris en charge par OpenSearch Ingestion.
2. Une fois que le cluster a atteint le statut actif, suivez les étapes décrites dans Activer la connectivité multi-VPC.
3. Suivez les étapes décrites dans Attacher une politique de cluster au cluster MSK pour associer l'une des politiques suivantes, selon que votre cluster et votre pipeline sont identiques Compte AWS ou non. Cette politique permet à OpenSearch Ingestion de créer une AWS PrivateLink connexion à votre cluster Amazon MSK et de lire les données des rubriques Kafka. Assurez-vous de le mettre à jour `resource` avec votre propre ARN.

   Les règles suivantes s'appliquent lorsque votre cluster et votre pipeline se trouvent dans le même environnement Compte AWS :

   JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:DescribeClusterV2"
      ],
```

```
      "Resource": "arn:aws:kafka:region:account-id:cluster/cluster-
name/cluster-id"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis-pipelines.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:GetBootstrapBrokers",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:region:account-id:cluster/cluster-
name/cluster-id"
    }
  ]
}
```

Si votre cluster Amazon MSK se trouve dans un pipeline différent Compte AWS de celui de votre pipeline, joignez plutôt la politique suivante. Notez que l'accès entre comptes n'est possible qu'avec les clusters Amazon MSK provisionnés et non avec les clusters Amazon MSK Serverless. L'ARN du AWS `principal` doit être l'ARN du même rôle de pipeline que celui que vous avez fourni à la configuration de votre pipeline :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:region:msk-account-id:cluster/cluster-
name/cluster-id"
    },
    {
```

```
      "Effect": "Allow",
      "Principal": {
        "Service": "osis-pipelines.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:GetBootstrapBrokers",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:region:msk-account-id:cluster/cluster-
name/cluster-id"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::pipeline-account-id:role/pipeline-role"
      },
      "Action": [
        "kafka-cluster:*",
        "kafka:*"
      ],
      "Resource": [
        "arn:aws:kafka:us-east-1:msk-account-id:cluster/cluster-name/cluster-
id",
        "arn:aws:kafka:us-east-1:msk-account-id:topic/cluster-name/cluster-id/
*",
        "arn:aws:kafka:us-east-1:msk-account-id:group/cluster-name/*"
      ]
    }
  ]
}
```

4. Créez un sujet Kafka en suivant les étapes décrites dans Créer un sujet. Assurez-vous qu'il s'*BootstrapServerString*agit de l'un des bootstrap du point de terminaison privé (VPC unique). URLs La valeur de `--replication-factor` doit être 2 ou3, en fonction du nombre de zones de votre cluster Amazon MSK. La valeur pour `--partitions` doit être au moins égale à`10`.
5. Produisez et consommez des données en suivant les étapes décrites dans la section Produire et consommer des données. Encore une fois, assurez-vous qu'il s'*BootstrapServerString*agit de l'un des bootstrap de votre point de terminaison privé (Single-VPC). URLs

Conditions préalables requises pour Amazon MSK Serverless

Avant de créer votre pipeline OpenSearch d'ingestion, effectuez les étapes suivantes :

1. Créez un cluster Amazon MSK Serverless en suivant les étapes décrites dans Create an MSK Serverless Cluster du manuel Amazon Managed Streaming for Apache Kafka Developer Guide.
2. Une fois que le cluster a atteint le statut Actif, suivez les étapes décrites dans Attacher une politique de cluster au cluster MSK pour associer la stratégie suivante. Assurez-vous de le mettre à jour `resource` avec votre propre ARN.

   JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:region:account-id:cluster/cluster-
name/cluster-id"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "osis-pipelines.amazonaws.com"
      },
      "Action": [
        "kafka:CreateVpcConnection",
        "kafka:GetBootstrapBrokers",
        "kafka:DescribeClusterV2"
      ],
      "Resource": "arn:aws:kafka:region:account-id:cluster/cluster-
name/cluster-id"
    }
  ]
}
```

Cette politique permet à OpenSearch Ingestion de créer une AWS PrivateLink connexion à votre cluster Amazon MSK Serverless et de lire les données des rubriques Kafka. Cette politique s'applique lorsque votre cluster et votre pipeline sont identiques Compte AWS, ce qui doit être vrai car Amazon MSK Serverless ne prend pas en charge l'accès entre comptes.

3. Créez un sujet Kafka en suivant les étapes décrites dans Créer un sujet. Assurez-vous qu'il s'*BootstrapServerString*agit de l'un de vos bootstrap IAM SASL (Simple Authentication and Security Layer). URLs La valeur de `--replication-factor` doit être `2` ou`3`, en fonction du nombre de zones de votre cluster Amazon MSK Serverless. La valeur pour `--partitions` doit être au moins égale à`10`.
4. Produisez et consommez des données en suivant les étapes décrites dans la section Produire et consommer des données. Encore une fois, assurez-vous qu'il s'*BootstrapServerString*agit de l'un de vos bootstrap IAM SASL (Simple Authentication and Security Layer). URLs

Étape 1 : Configuration d'un rôle de pipeline

Après avoir configuré votre cluster Amazon MSK provisionné ou sans serveur, ajoutez les autorisations Kafka suivantes dans le rôle de pipeline que vous souhaitez utiliser dans la configuration de votre pipeline :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:Connect",
                "kafka-cluster:AlterCluster",
                "kafka-cluster:DescribeCluster",
                "kafka:DescribeClusterV2",
                "kafka:GetBootstrapBrokers"
            ],
            "Resource": [
                "arn:aws:kafka:region:account-id:cluster/cluster-name/cluster-id"
            ]
        },
        {
```

```
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:*Topic*",
                "kafka-cluster:ReadData"
            ],
            "Resource": [
                "arn:aws:kafka:region:account-id:topic/cluster-name/cluster-
id/topic-name"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:AlterGroup",
                "kafka-cluster:DescribeGroup"
            ],
            "Resource": [
                "arn:aws:kafka:region:account-id:group/cluster-name/*"
            ]
        }
    ]
}
```

Étape 2 : Création du pipeline

Vous pouvez ensuite configurer un pipeline d' OpenSearch ingestion comme le suivant, qui spécifie Kafka comme source :

```
version: "2"
log-pipeline:
  source:
    kafka:
      acknowledgements: true
      topics:
      - name: "topic-name"
        group_id: "grouplambd-id"
      aws:
        msk:
          arn: "arn:aws:kafka:region:account-id:cluster/cluster-name/cluster-id"
        region: "us-west-2"
  processor:
  - grok:
```

```
      match:
        message:
        - "%{COMMONAPACHELOG}"
  - date:
      destination: "@timestamp"
      from_time_received: true
  sink:
  - opensearch:
      hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
      index: "index_name"
      aws_region: "region"
      aws_sigv4: true
```

Vous pouvez utiliser un plan Amazon MSK préconfiguré pour créer ce pipeline. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Étape 3 : (Facultatif) Utiliser le registre des AWS Glue schémas

Lorsque vous utilisez OpenSearch Ingestion avec Amazon MSK, vous pouvez utiliser le format de données AVRO pour les schémas hébergés dans le AWS Glue registre des schémas. Avec le registre des AWS Glue schémas, vous pouvez découvrir, contrôler et faire évoluer les schémas de flux de données de manière centralisée.

Pour utiliser cette option, activez le schéma `type` dans la configuration de votre pipeline :

```
schema:
  type: "aws_glue"
```

Vous devez également fournir des AWS Glue autorisations d'accès en lecture dans votre rôle de pipeline. Vous pouvez utiliser la politique AWS gérée appelée AWSGlueSchemaRegistryReadonlyAccess. De plus, votre registre doit se trouver dans la même Compte AWS région que votre pipeline OpenSearch d'ingestion.

Étape 4 : (Facultatif) Configurer les unités de calcul recommandées (OCUs) pour le pipeline Amazon MSK

Chaque unité de calcul a un consommateur par sujet. Les courtiers équilibrent les différences entre ces consommateurs pour un sujet donné. Toutefois, lorsque le nombre de partitions est supérieur au nombre de consommateurs, Amazon MSK héberge plusieurs partitions pour chaque consommateur. OpenSearch Ingestion intègre une mise à l'échelle automatique pour augmenter ou diminuer en fonction de l'utilisation du processeur ou du nombre d'enregistrements en attente dans le pipeline.

Pour des performances optimales, répartissez vos partitions sur de nombreuses unités de calcul pour un traitement parallèle. Si les rubriques comportent un grand nombre de partitions (par exemple, plus de 96, ce qui est le maximum OCUs par pipeline), nous vous recommandons de configurer un pipeline de 1 à 96 OCUs. Cela est dû au fait qu'il sera automatiquement redimensionné selon les besoins. Si un sujet comporte un faible nombre de partitions (par exemple, moins de 96), maintenez l'unité de calcul maximale identique au nombre de partitions.

Lorsqu'un pipeline comporte plusieurs sujets, choisissez le sujet contenant le plus grand nombre de partitions comme référence pour configurer le maximum d'unités de calcul. En ajoutant un autre pipeline avec un nouvel ensemble de produits OCUs au même sujet et au même groupe de consommateurs, vous pouvez augmenter le débit de manière presque linéaire.

Utilisation d'un pipeline OpenSearch d'ingestion avec Amazon S3

Avec OpenSearch Ingestion, vous pouvez utiliser Amazon S3 comme source ou comme destination. Lorsque vous utilisez Amazon S3 comme source, vous envoyez des données vers un pipeline d' OpenSearch ingestion. Lorsque vous utilisez Amazon S3 comme destination, vous écrivez les données d'un pipeline d' OpenSearch ingestion dans un ou plusieurs compartiments S3.

Rubriques

- Amazon S3 en tant que source
- Amazon S3 en tant que destination
- Compte croisé Amazon S3 en tant que source

Amazon S3 en tant que source

Vous pouvez utiliser Amazon S3 comme source pour traiter les données de deux manières : avec le traitement S3-SQS et avec les scans planifiés.

Utilisez le traitement S3-SQS lorsque vous avez besoin d'analyser des fichiers en temps quasi réel après leur écriture dans S3. Vous pouvez configurer les compartiments Amazon S3 pour déclencher un événement chaque fois qu'un objet est stocké ou modifié dans le compartiment. Utilisez une analyse planifiée ponctuelle ou récurrente pour traiter par lots les données d'un compartiment S3.

Rubriques

- Prérequis
- Étape 1 : configurer le rôle du pipeline

- Étape 2 : Création du pipeline

Prérequis

Pour utiliser Amazon S3 comme source d'un pipeline d' OpenSearch ingestion à la fois pour un scan planifié ou un traitement S3-SQS, créez d'abord un compartiment S3.

> **Note**
>
> Si le compartiment S3 utilisé comme source dans le pipeline d' OpenSearch ingestion se trouve dans un autre compartiment Compte AWS, vous devez également activer les autorisations de lecture entre comptes sur le compartiment. Cela permet au pipeline de lire et de traiter les données. Pour activer les autorisations entre comptes, consultez la section Octroi par le propriétaire du bucket des autorisations de bucket entre comptes dans le guide de l'utilisateur Amazon S3.
> Si vos compartiments S3 se trouvent dans plusieurs comptes, utilisez une `bucket_owners` carte. Pour un exemple, consultez la section Accès S3 entre comptes dans la OpenSearch documentation.

Pour configurer le traitement S3-SQS, vous devez également effectuer les étapes suivantes :

1. Créez une file d'attente Amazon SQS.
2. Activez les notifications d'événements sur le compartiment S3 avec la file d'attente SQS comme destination.

Étape 1 : configurer le rôle du pipeline

Contrairement aux autres plugins source qui envoient des données vers un pipeline, le plug-in source S3 possède une architecture basée sur la lecture dans laquelle le pipeline extrait les données de la source.

Par conséquent, pour qu'un pipeline puisse lire depuis S3, vous devez spécifier un rôle dans la configuration source S3 du pipeline qui a accès à la fois au compartiment S3 et à la file d'attente Amazon SQS. Le pipeline assumera ce rôle afin de lire les données de la file d'attente.

> **Note**
>
> Le rôle que vous spécifiez dans la configuration source S3 doit être le rôle de pipeline. Par conséquent, votre rôle de pipeline doit contenir deux politiques d'autorisation distinctes : l'une pour écrire dans un récepteur et l'autre pour extraire de la source S3. Vous devez utiliser le même principe `sts_role_arn` dans tous les composants du pipeline.

L'exemple de politique suivant indique les autorisations requises pour utiliser S3 en tant que source :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action":[
          "s3:ListBucket",
          "s3:GetBucketLocation",
          "s3:GetObject"
       ],
      "Resource": "arn:aws:s3:::bucket-name/*"
    },
    {
       "Effect":"Allow",
       "Action":"s3:ListAllMyBuckets",
       "Resource":"arn:aws:s3:::*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "sqs:DeleteMessage",
        "sqs:ReceiveMessage",
        "sqs:ChangeMessageVisibility"
      ],
      "Resource": "arn:aws:sqs:us-west-2:account-id:MyS3EventSqsQueue"
    }
  ]
}
```

Vous devez associer ces autorisations au rôle IAM que vous spécifiez dans l'`sts_role_arn`option de configuration du plugin source S3 :

```
version: "2"
source:
  s3:
    ...
    aws:
      ...
processor:
  ...
sink:
  - opensearch:
      ...
```

Étape 2 : Création du pipeline

Après avoir configuré vos autorisations, vous pouvez configurer un pipeline d' OpenSearch ingestion en fonction de votre cas d'utilisation d'Amazon S3.

Traitement S3-SQS

Pour configurer le traitement S3-SQS, configurez votre pipeline pour spécifier S3 comme source et configurez les notifications Amazon SQS :

```
version: "2"
s3-pipeline:
  source:
    s3:
      notification_type: "sqs"
      codec:
        newline: null
      sqs:
        queue_url: "https://sqs.us-east-1amazonaws.com/account-id/ingestion-queue"
      compression: "none"
      aws:
        region: "region"
  processor:
  - grok:
      match:
        message:
        - "%{COMMONAPACHELOG}"
  - date:
```

```
      destination: "@timestamp"
      from_time_received: true
  sink:
  - opensearch:
      hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
      index: "index-name"
      aws:
        region: "region"
```

Si vous observez une faible utilisation du processeur lors du traitement de petits fichiers sur Amazon S3, envisagez d'augmenter le débit en modifiant la valeur de l'`workers`option. Pour plus d'informations, consultez les options de configuration du plugin S3.

Scan programmé

Pour configurer une analyse planifiée, configurez votre pipeline avec une planification au niveau de l'analyse qui s'applique à tous vos compartiments S3, ou au niveau du compartiment. Une planification au niveau du compartiment ou une configuration à intervalles de numérisation remplace toujours une configuration au niveau du scan.

Vous pouvez configurer des scans planifiés avec un scan unique, idéal pour la migration des données, ou un scan récurrent, idéal pour le traitement par lots.

Pour configurer votre pipeline afin qu'il puisse lire depuis Amazon S3, utilisez les plans Amazon S3 préconfigurés. Vous pouvez modifier la `scan` partie de la configuration de votre pipeline pour répondre à vos besoins de planification. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Scan unique

Un scan programmé ne s'exécute qu'une seule fois. Dans la configuration de votre pipeline, vous pouvez utiliser un `start_time` et `end_time` pour spécifier à quel moment vous souhaitez que les objets du compartiment soient scannés. Vous pouvez également l'utiliser `range` pour spécifier l'intervalle de temps par rapport à l'heure actuelle pendant laquelle vous souhaitez que les objets du compartiment soient scannés.

Par exemple, une plage définie pour analyser PT4H tous les fichiers créés au cours des quatre dernières heures. Pour configurer une analyse unique afin qu'elle soit exécutée une deuxième fois, vous devez arrêter et redémarrer le pipeline. Si aucune plage n'est configurée, vous devez également mettre à jour les heures de début et de fin.

La configuration suivante permet de configurer une analyse unique de tous les compartiments et de tous les objets qu'ils contiennent :

```
version: "2"
log-pipeline:
  source:
    s3:
      codec:
        csv:
      compression: "none"
      aws:
        region: "region"
      acknowledgments: true
      scan:
        buckets:
          - bucket:
              name: my-bucket
              filter:
                include_prefix:
                  - Objects1/
                exclude_suffix:
                  - .jpeg
                  - .png
          - bucket:
              name: my-bucket-2
              key_prefix:
                include:
                  - Objects2/
                exclude_suffix:
                  - .jpeg
                  - .png
      delete_s3_objects_on_read: false
  processor:
    - date:
        destination: "@timestamp"
        from_time_received: true
  sink:
    - opensearch:
        hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
        index: "index-name"
        aws:
          region: "region"
        dlq:
```

```
          s3:
            bucket: "dlq-bucket"
            region: "us-east-1"
```

La configuration suivante met en place une analyse unique de tous les compartiments pendant une période spécifiée. Cela signifie que S3 traite uniquement les objets dont l'heure de création se situe dans cette fenêtre.

```
scan:
  start_time: 2023-01-21T18:00:00.000Z
  end_time: 2023-04-21T18:00:00.000Z
  buckets:
    - bucket:
        name: my-bucket-1
        filter:
          include:
            - Objects1/
          exclude_suffix:
            - .jpeg
            - .png
    - bucket:
        name: my-bucket-2
        filter:
          include:
            - Objects2/
          exclude_suffix:
            - .jpeg
            - .png
```

La configuration suivante configure une analyse unique à la fois au niveau du scan et au niveau du bucket. Les heures de début et de fin au niveau du bucket remplacent les heures de début et de fin au niveau du scan.

```
scan:
  start_time: 2023-01-21T18:00:00.000Z
  end_time: 2023-04-21T18:00:00.000Z
  buckets:
    - bucket:
        start_time: 2023-01-21T18:00:00.000Z
        end_time: 2023-04-21T18:00:00.000Z
        name: my-bucket-1
        filter:
```

```
          include:
            - Objects1/
          exclude_suffix:
            - .jpeg
            - .png
    - bucket:
        start_time: 2023-01-21T18:00:00.000Z
        end_time: 2023-04-21T18:00:00.000Z
        name: my-bucket-2
        filter:
          include:
            - Objects2/
          exclude_suffix:
            - .jpeg
            - .png
```

L'arrêt d'un pipeline supprime toute référence préexistante aux objets scannés par le pipeline avant l'arrêt. Si un seul pipeline de numérisation est arrêté, il scanne à nouveau tous les objets après son démarrage, même s'ils ont déjà été numérisés. Si vous devez arrêter un seul pipeline de scan, il est recommandé de modifier votre créneau horaire avant de recommencer le pipeline.

Si vous devez filtrer les objets par heure de début et de fin, l'arrêt et le démarrage de votre pipeline sont la seule option. S'il n'est pas nécessaire de filtrer par heure de début et heure de fin, vous pouvez filtrer les objets par nom. Filtrer par nom ne vous oblige pas à arrêter et à démarrer votre pipeline. Pour ce faire, utilisez `include_prefix` et`exclude_suffix`.

Scan récurrent

Une analyse planifiée récurrente exécute une analyse des compartiments S3 que vous avez spécifiés à intervalles réguliers et planifiés. Vous ne pouvez configurer ces intervalles qu'au niveau du scan, car les configurations individuelles au niveau du bucket ne sont pas prises en charge.

Dans la configuration de votre pipeline`interval`, la fréquence du scan récurrent peut être comprise entre 30 secondes et 365 jours. Le premier de ces scans a toujours lieu lorsque vous créez le pipeline. `count`Définit le nombre total d'instances de scan.

La configuration suivante permet de configurer un scan récurrent, avec un délai de 12 heures entre les scans :

```
scan:
  scheduling:
```

```
    interval: PT12H
    count: 4
  buckets:
    - bucket:
        name: my-bucket-1
        filter:
          include:
            - Objects1/
          exclude_suffix:
            - .jpeg
            - .png
    - bucket:
        name: my-bucket-2
        filter:
          include:
            - Objects2/
          exclude_suffix:
            - .jpeg
            - .png
```

Amazon S3 en tant que destination

Pour écrire des données d'un pipeline d' OpenSearch ingestion dans un compartiment S3, utilisez le plan S3 préconfiguré pour créer un pipeline avec un récepteur S3. Ce pipeline achemine des données sélectives vers un OpenSearch récepteur et envoie simultanément toutes les données pour archivage dans S3. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Lorsque vous créez votre récepteur S3, vous pouvez spécifier votre formatage préféré à partir de différents codecs récepteurs. Par exemple, si vous souhaitez écrire des données sous forme de colonnes, choisissez le codec Parquet ou Avro. Si vous préférez un format basé sur des lignes, choisissez JSON ou NDJSON. Pour écrire des données dans S3 dans un schéma spécifique, vous pouvez également définir un schéma en ligne dans les codecs récepteurs à l'aide du format Avro.

L'exemple suivant définit un schéma en ligne dans un récepteur S3 :

```
- s3:
  codec:
    parquet:
      schema: >
        {
           "type" : "record",
```

```
    "namespace" : "org.vpcFlowLog.examples",
    "name" : "VpcFlowLog",
    "fields" : [
      { "name" : "version", "type" : "string"},
      { "name" : "srcport", "type": "int"},
      { "name" : "dstport", "type": "int"},
      { "name" : "start", "type": "int"},
      { "name" : "end", "type": "int"},
      { "name" : "protocol", "type": "int"},
      { "name" : "packets", "type": "int"},
      { "name" : "bytes", "type": "int"},
      { "name" : "action", "type": "string"},
      { "name" : "logStatus", "type" : "string"}
    ]
  }
```

Lorsque vous définissez ce schéma, spécifiez un sur-ensemble de toutes les clés susceptibles d'être présentes dans les différents types d'événements que votre pipeline envoie à un récepteur.

Par exemple, s'il est possible qu'une clé soit manquante lors d'un événement, ajoutez cette clé dans votre schéma avec une `null` valeur. Les déclarations de valeur nulle permettent au schéma de traiter des données non uniformes (certains événements possèdent ces clés, d'autres non). Lorsque ces clés sont présentes dans des événements entrants, leurs valeurs sont écrites dans des récepteurs.

Cette définition de schéma agit comme un filtre qui permet uniquement d'envoyer des clés définies aux récepteurs et supprime les clés non définies des événements entrants.

Vous pouvez également utiliser `include_keys` et `exclude_keys` dans votre récepteur pour filtrer les données acheminées vers d'autres récepteurs. Ces deux filtres s'excluent mutuellement, vous ne pouvez donc en utiliser qu'un à la fois dans votre schéma. En outre, vous ne pouvez pas les utiliser dans des schémas définis par l'utilisateur.

Pour créer des pipelines avec de tels filtres, utilisez le plan de filtre récepteur préconfiguré. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Compte croisé Amazon S3 en tant que source

Vous pouvez accorder l'accès à plusieurs comptes avec Amazon S3 afin que les pipelines OpenSearch d'ingestion puissent accéder aux compartiments S3 d'un autre compte en tant que source. Pour activer l'accès entre comptes, consultez la section Octroi par le propriétaire du bucket

des autorisations de bucket entre comptes dans le guide de l'utilisateur Amazon S3. Une fois que vous avez accordé l'accès, assurez-vous que votre rôle de pipeline dispose des autorisations requises.

Vous pouvez ensuite créer un pipeline en utilisant `bucket_owners` pour activer l'accès entre comptes à un compartiment Amazon S3 en tant que source :

```
s3-pipeline:
 source:
  s3:
   notification_type: "sqs"
   codec:
    csv:
     delimiter: ","
     quote_character: "\""
     detect_header: True
   sqs:
    queue_url: "https://sqs.ap-northeast-1.amazonaws.com/401447383613/test-s3-queue"
   bucket_owners:
    my-bucket-01: 123456789012
    my-bucket-02: 999999999999
   compression: "gzip"
```

Utilisation d'un pipeline d' OpenSearch ingestion avec Amazon Security Lake

Vous pouvez utiliser le plug-in source S3 pour intégrer les données d'Amazon Security Lake dans votre pipeline OpenSearch d'ingestion. Security Lake centralise automatiquement les données de sécurité provenant des AWS environnements, des environnements sur site et des fournisseurs de SaaS dans un lac de données spécialement conçu à cet effet. Vous pouvez créer un abonnement qui réplique les données de Security Lake vers votre pipeline d' OpenSearch ingestion, qui les écrit ensuite dans votre domaine de OpenSearch service ou votre collection OpenSearch Serverless.

Pour configurer votre pipeline afin qu'il lise depuis Security Lake, utilisez le plan préconfiguré de Security Lake. Le plan inclut une configuration par défaut pour l'ingestion de fichiers parquet OCSF (Open Cybersecurity Schema Framework) à partir de Security Lake. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Rubriques

- Utilisation d'un pipeline d' OpenSearch ingestion avec Amazon Security Lake comme source

- Utilisation d'un pipeline d' OpenSearch ingestion avec Amazon Security Lake comme récepteur

Utilisation d'un pipeline d' OpenSearch ingestion avec Amazon Security Lake comme source

Vous pouvez utiliser le plug-in source Amazon S3 dans votre pipeline OpenSearch d'ingestion pour ingérer des données depuis Amazon Security Lake. Security Lake centralise automatiquement les données de sécurité provenant des AWS environnements, des systèmes sur site et des fournisseurs de SaaS dans un lac de données spécialement conçu à cet effet.

Amazon Security Lake possède les attributs de métadonnées suivants au sein d'un pipeline :

- `bucket_name`: nom du compartiment Amazon S3 créé par Security Lake pour stocker les données de sécurité.
- `path_prefix`: nom de source personnalisé défini dans la politique de rôle IAM de Security Lake.
- `region`: l' Région AWS endroit où se trouve le bucket Security Lake S3.
- `accountID`: Compte AWS ID sous lequel Security Lake est activé.
- `sts_role_arn`: ARN du rôle IAM destiné à être utilisé avec Security Lake.

Prérequis

Avant de créer votre pipeline OpenSearch d'ingestion, effectuez les étapes suivantes :

- Activez Security Lake.
- Créez un abonné dans Security Lake.
  - Choisissez les sources que vous souhaitez intégrer à votre pipeline.
  - Pour les informations d'identification de l'abonné, ajoutez l'ID de l' Compte AWS endroit où vous souhaitez créer le pipeline. Pour l'ID externe, spécifiez`OpenSearchIngestion-{accountid}`.
  - Pour la méthode d'accès aux données, choisissez S3.
  - Pour les détails des notifications, choisissez la file d'attente SQS.

Lorsque vous créez un abonné, Security Lake crée automatiquement deux politiques d'autorisation intégrées, l'une pour S3 et l'autre pour SQS. Les politiques prennent le format suivant : `AmazonSecurityLake-{12345}-S3` et`AmazonSecurityLake-{12345}-SQS`. Pour permettre

à votre pipeline d'accéder aux sources d'abonnés, vous devez associer les autorisations requises à votre rôle de pipeline.

Configurer le rôle du pipeline

Créez une nouvelle politique d'autorisations dans IAM qui combine uniquement les autorisations requises issues des deux politiques créées automatiquement par Security Lake. L'exemple de politique suivant montre le minimum de privilèges requis pour qu'un pipeline d' OpenSearch ingestion puisse lire des données provenant de plusieurs sources de Security Lake :

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "s3:GetObject"
         ],
         "Resource":[
            "arn:aws:s3:::aws-security-data-lake-region-abcde/aws/
LAMBDA_EXECUTION/1.0/*",
            "arn:aws:s3:::aws-security-data-lake-region-abcde/aws/S3_DATA/1.0/*",
            "arn:aws:s3:::aws-security-data-lake-region-abcde/aws/VPC_FLOW/1.0/
*",
            "arn:aws:s3:::aws-security-data-lake-region-abcde/aws/ROUTE53/1.0/*",
            "arn:aws:s3:::aws-security-data-lake-region-abcde/aws/
SH_FINDINGS/1.0/*"
         ]
      },
      {
         "Effect":"Allow",
         "Action":[
            "sqs:ReceiveMessage",
            "sqs:DeleteMessage"
         ],
         "Resource":[
            "arn:aws:sqs:region:account-id:AmazonSecurityLake-abcde-Main-Queue"
         ]
      }
   ]
```

```
}
```

> **Important**
>
> Security Lake ne gère pas la politique des rôles du pipeline à votre place. Si vous ajoutez ou supprimez des sources de votre abonnement Security Lake, vous devez mettre à jour la politique manuellement. Security Lake crée des partitions pour chaque source de journal. Vous devez donc ajouter ou supprimer manuellement des autorisations dans le rôle de pipeline.

Vous devez associer ces autorisations au rôle IAM que vous spécifiez dans l'`sts_role_arn`option de configuration du plugin source S3, sous`sqs`.

```
version: "2"
source:
  s3:
    ...
    sqs:
      queue_url: "https://sqs.us-east-1amazonaws.com/account-id/
AmazonSecurityLake-abcde-Main-Queue"
    aws:
      ...
processor:
  ...
sink:
  - opensearch:
      ...
```

Création du pipeline

Après avoir ajouté les autorisations au rôle de pipeline, utilisez le plan préconfiguré de Security Lake pour créer le pipeline. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Vous devez spécifier l'`queue_url`option dans la configuration `s3` source, à savoir l'URL de la file d'attente Amazon SQS à partir de laquelle effectuer la lecture. Pour formater l'URL, recherchez le point de terminaison d'abonnement dans la configuration de l'abonné et `arn:aws:` remplacez-

le parhttps://. Par exemple, https://sqs.*us-east-1*amazonaws.com/*account-id*/AmazonSecurityLake-*abdcef*-Main-Queue.

L'ARN sts_role_arn que vous spécifiez dans la configuration de la source S3 doit être l'ARN du rôle de pipeline.

Utilisation d'un pipeline d' OpenSearch ingestion avec Amazon Security Lake comme récepteur

Utilisez le plug-in Amazon S3 sink dans OpenSearch Ingestion pour envoyer des données depuis n'importe quelle source prise en charge vers Amazon Security Lake. Security Lake collecte et stocke les données de sécurité provenant AWS des environnements sur site et des fournisseurs de SaaS dans un lac de données dédié.

Pour configurer votre pipeline afin d'écrire les données des journaux dans Security Lake, utilisez le modèle préconfiguré des journaux de trafic du pare-feu. Le plan inclut une configuration par défaut pour récupérer les journaux de sécurité bruts ou d'autres données stockées dans un compartiment Amazon S3, traiter les enregistrements et les normaliser. Il mappe ensuite les données à l'Open Cybersecurity Schema Framework (OCSF) et envoie les données conformes à l'OCSF transformées à Security Lake.

Le pipeline possède les attributs de métadonnées suivants :

- bucket_name: nom du compartiment Amazon S3 créé par Security Lake pour stocker les données de sécurité.
- path_prefix: nom de source personnalisé défini dans la politique de rôle IAM de Security Lake.
- region: l' Région AWS endroit où se trouve le bucket Security Lake S3.
- accountID: Compte AWS ID sous lequel Security Lake est activé.
- sts_role_arn: ARN du rôle IAM destiné à être utilisé avec Security Lake.

Prérequis

Avant de créer un pipeline pour envoyer des données à Security Lake, effectuez les étapes suivantes :

- Activer et configurer Amazon Security Lake : configurez Amazon Security Lake pour centraliser les données de sécurité provenant de différentes sources. Pour obtenir des instructions, consultez la section Activation de Security Lake à l'aide de la console.

Lorsque vous sélectionnez une source, choisissez Ingérer des AWS sources spécifiques et sélectionnez une ou plusieurs sources de journaux et d'événements que vous souhaitez ingérer.

- Configurer les autorisations : configurez le rôle de pipeline avec les autorisations requises pour écrire des données dans Security Lake. Pour plus d'informations, consultez la section Rôle du pipeline.

Création du pipeline

Utilisez le plan préconfiguré de Security Lake pour créer le pipeline. Pour plus d'informations, consultez la section Utilisation de plans pour créer un pipeline.

Utilisation d'un pipeline OpenSearch d'ingestion avec Fluent Bit

Cet exemple de fichier de configuration Fluent Bit envoie les données de journal de Fluent Bit à un pipeline d' OpenSearch ingestion. Pour plus d'informations sur l'ingestion des données de journal, consultez Log Analytics dans la documentation de Data Prepper.

Remarques :

- La `host` valeur doit être le point de terminaison de votre pipeline. Par exemple, *`pipeline-endpoint`*`.`*`us-east-1`*`osis.amazonaws.com`.
- La valeur `aws_service` doit être `osis`.
- La `aws_role_arn` valeur est l'ARN du rôle AWS IAM que le client doit assumer et utiliser pour l'authentification Signature version 4.

```
[INPUT]
  name                  tail
  refresh_interval      5
  path                  /var/log/test.log
  read_from_head        true

[OUTPUT]
  Name http
  Match *
  Host pipeline-endpoint.us-east-1osis.amazonaws.com
  Port 443
  URI /log/ingest
  Format json
```

```
  aws_auth true
  aws_region region
  aws_service osis
  aws_role_arn arn:aws:iam::account-id:role/ingestion-role
  Log_Level trace
  tls On
```

Vous pouvez ensuite configurer un pipeline d' OpenSearch ingestion comme celui-ci, dont la source est HTTP :

```
version: "2"
unaggregated-log-pipeline:
  source:
    http:
      path: "/log/ingest"
  processor:
    - grok:
        match:
          log:
            - "%{TIMESTAMP_ISO8601:timestamp} %{NOTSPACE:network_node}
 %{NOTSPACE:network_host} %{IPORHOST:source_ip}:%{NUMBER:source_port:int} ->
 %{IPORHOST:destination_ip}:%{NUMBER:destination_port:int} %{GREEDYDATA:details}"
    - grok:
        match:
          details:
            - "'%{NOTSPACE:http_method} %{NOTSPACE:http_uri}' %{NOTSPACE:protocol}"
            - "TLS%{NOTSPACE:tls_version} %{GREEDYDATA:encryption}"
            - "%{NUMBER:status_code:int} %{NUMBER:response_size:int}"
    - delete_entries:
        with_keys: ["details", "log"]

  sink:
    - opensearch:
        hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
        index: "index_name"
        index_type: custom
        bulk_size: 20
        aws:
          region: "region"
```

Utilisation d'un pipeline OpenSearch d'ingestion avec Fluentd

Fluentd est un écosystème de collecte de données open source qui fournit SDKs différents langages et sous-projets tels que Fluent Bit. Cet exemple de fichier de configuration Fluentd envoie les données de journal de Fluentd à un pipeline d'ingestion. OpenSearch Pour plus d'informations sur l'ingestion des données de journal, consultez Log Analytics dans la documentation de Data Prepper.

Remarques :

- La `endpoint` valeur doit être le point de terminaison de votre pipeline. Par exemple, *`pipeline-endpoint`*`.`*`us-east-1`*`osis.amazonaws.com/apache-log-pipeline/logs`.
- La valeur `aws_service` doit être `osis`.
- La `aws_role_arn` valeur est l'ARN du rôle AWS IAM que le client doit assumer et utiliser pour l'authentification Signature version 4.

```
<source>
  @type tail
  path logs/sample.log
  path_key log
  tag apache
  <parse>
    @type none
  </parse>
</source>

<filter apache>
  @type record_transformer
  <record>
    log ${record["message"]}
  </record>
</filter>

<filter apache>
  @type record_transformer
  remove_keys message
</filter>

<match apache>
  @type http
  endpoint pipeline-endpoint.us-east-1osis.amazonaws.com/apache-log-pipeline/logs
  json_array true
```

```
  <auth>
    method aws_sigv4
    aws_service osis
    aws_region region
    aws_role_arn arn:aws:iam::account-id:role/ingestion-role
  </auth>

  <format>
    @type json
  </format>

  <buffer>
    flush_interval 1s
  </buffer>
</match>
```

Vous pouvez ensuite configurer un pipeline d' OpenSearch ingestion comme celui-ci, dont la source est HTTP :

```
version: "2"
apache-log-pipeline:
  source:
    http:
      path: "/${pipelineName}/logs"
  processor:
    - grok:
        match:
          log:
            - "%{TIMESTAMP_ISO8601:timestamp} %{NOTSPACE:network_node}
 %{NOTSPACE:network_host} %{IPORHOST:source_ip}:%{NUMBER:source_port:int} ->
 %{IPORHOST:destination_ip}:%{NUMBER:destination_port:int} %{GREEDYDATA:details}"
  sink:
    - opensearch:
        hosts: ["https://search-domain-endpoint.us-east-1es.amazonaws.com"]
        index: "index_name"
        aws_region: "region"
        aws_sigv4: true
```

Utilisation d'un pipeline OpenSearch d'ingestion avec OpenTelemetry Collector

Cet exemple OpenTelemetry de fichier de configuration exporte les données de trace depuis le OpenTelemetry collecteur et les envoie vers un pipeline d' OpenSearch ingestion. Pour plus d'informations sur l'ingestion de données de trace, consultez Trace Analytics dans la documentation de Data Prepper.

Remarques :

- La `endpoint` valeur doit inclure le point de terminaison de votre pipeline. Par exemple, `https://`*`pipeline-endpoint`*`.`*`us-east-1`*`osis.amazonaws.com`.
- La valeur `service` doit être `osis`.
- L'`compression`option pour l' OTLP/HTTP exportateur doit correspondre à l'`compression`option sur la OpenTelemetry source du pipeline.

```
extensions:
  sigv4auth:
    region: "region"
    service: "osis"

receivers:
  jaeger:
    protocols:
      grpc:

exporters:
  otlphttp:
    traces_endpoint: "https://pipeline-endpoint.us-east-1.osis.amazonaws.com/v1/traces"
    auth:
      authenticator: sigv4auth
    compression: none

service:
  extensions: [sigv4auth]
  pipelines:
    traces:
      receivers: [jaeger]
      exporters: [otlphttp]
```

Vous pouvez ensuite configurer un pipeline d' OpenSearch ingestion comme le suivant, qui spécifie le plugin de OTel trace comme source :

```
version: "2"
otel-trace-pipeline:
  source:
    otel_trace_source:
      path: "/v1/traces"
  processor:
    - trace_peer_forwarder:
  sink:
    - pipeline:
        name: "trace-pipeline"
    - pipeline:
        name: "service-map-pipeline"
trace-pipeline:
  source:
    pipeline:
      name: "otel-trace-pipeline"
  processor:
    - otel_traces:
  sink:
    - opensearch:
        hosts: ["https://search-domain-endpoint.us-east-1.es.amazonaws.com"]
        index_type: trace-analytics-raw
        aws:
          region: "region"
service-map-pipeline:
  source:
    pipeline:
      name: "otel-trace-pipeline"
  processor:
    - service_map:
  sink:
    - opensearch:
        hosts: ["https://search-domain-endpoint.us-east-1.es.amazonaws.com"]
        index_type: trace-analytics-service-map
        aws:
          region: "region"
```

Pour un autre exemple de pipeline, consultez le plan d'analyse de trace préconfiguré. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Utiliser un pipeline d' OpenSearch ingestion avec Kafka

Vous pouvez utiliser le plugin Kafka pour diffuser des données depuis des clusters Kafka autogérés vers des domaines Amazon OpenSearch Service et OpenSearch des collections Serverless. OpenSearch L'ingestion prend en charge les connexions à partir de clusters Kafka configurés avec un réseau public ou privé (VPC). Cette rubrique décrit les conditions préalables et les étapes à suivre pour configurer un pipeline d'ingestion, notamment la configuration des paramètres réseau et des méthodes d'authentification telles que le protocole TLS mutuel (MTL), le SASL/SCRAM ou l'IAM.

Migration de données depuis des clusters Kafka publics

Vous pouvez utiliser des pipelines d' OpenSearch ingestion pour migrer des données depuis un cluster Kafka public autogéré, ce qui signifie que le nom DNS du domaine peut être résolu publiquement. Pour ce faire, configurez un pipeline d' OpenSearch ingestion avec Kafka autogéré comme source et OpenSearch Service ou OpenSearch Serverless comme destination. Cela traite vos données de streaming depuis un cluster source autogéré vers un domaine ou une collection de destination AWS géré.

Prérequis

Avant de créer votre pipeline OpenSearch d'ingestion, effectuez les étapes suivantes :

1. Créez un cluster Kafka autogéré avec une configuration de réseau public. Le cluster doit contenir les données que vous souhaitez ingérer dans OpenSearch Service.
2. Créez un domaine OpenSearch de service ou une collection OpenSearch sans serveur vers lequel vous souhaitez migrer les données. Pour plus d’informations, consultez the section called “ Création de domaines OpenSearch de service” et the section called “Créer des collections”.
3. Configurez l'authentification sur votre cluster autogéré avec AWS Secrets Manager. Activez la rotation des secrets en suivant les étapes décrites dans Rotation AWS Secrets Manager des secrets.
4. Associez une politique basée sur les ressources à votre domaine ou une politique d'accès aux données à votre collection. Ces politiques d'accès permettent à OpenSearch Ingestion d'écrire des données de votre cluster autogéré vers votre domaine ou votre collection.

   L'exemple de politique d'accès au domaine suivant permet au rôle de pipeline, que vous créez à l'étape suivante, d'écrire des données dans un domaine. Assurez-vous de le mettre à jour `resource` avec votre propre ARN.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::pipeline-account-id:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:region:account-id:domain/domain-name"
      ]
    }
  ]
}
```

Pour créer un rôle IAM doté des autorisations appropriées pour accéder aux données d'écriture de la collection ou du domaine, consultezthe section called “Configuration des rôles et des utilisateurs”.

Étape 1 : configurer le rôle du pipeline

Une fois les prérequis de votre pipeline Kafka configurés, configurez le rôle de pipeline que vous souhaitez utiliser dans la configuration de votre pipeline et ajoutez l'autorisation d'écrire dans un domaine de OpenSearch service ou une collection OpenSearch sans serveur, ainsi que l'autorisation de lire les secrets depuis Secrets Manager.

Étape 2 : Création du pipeline

Vous pouvez ensuite configurer un pipeline d' OpenSearch ingestion comme le suivant, qui spécifie Kafka comme source.

Vous pouvez spécifier plusieurs domaines OpenSearch de service comme destinations pour vos données. Cette fonctionnalité permet le routage conditionnel ou la réplication des données entrantes dans plusieurs domaines OpenSearch de service.

Vous pouvez également migrer les données d'un cluster Confluent Kafka source vers une collection VPC OpenSearch sans serveur. Assurez-vous de fournir une politique d'accès au réseau dans la configuration du pipeline. Vous pouvez utiliser un registre de schémas Confluent pour définir un schéma Confluent.

```
version: "2"
kafka-pipeline:
  source:
    kafka:
      encryption:
        type: "ssl"
      topics:
        - name: "topic-name"
          group_id: "group-id"
      bootstrap_servers:
        - "bootstrap-server.us-east-1.aws.private.confluent.cloud:9092"
      authentication:
        sasl:
          plain:
            username: ${aws_secrets:confluent-kafka-secret:username}
            password: ${aws_secrets:confluent-kafka-secret:password}
      schema:
        type: confluent
        registry_url: https://my-registry.us-east-1.aws.confluent.cloud
        api_key: "${{aws_secrets:schema-secret:schema_registry_api_key}}"
        api_secret: "${{aws_secrets:schema-secret:schema_registry_api_secret}}"
        basic_auth_credentials_source: "USER_INFO"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
      index: "confluent-index"
extension:
  aws:
    secrets:
      confluent-kafka-secret:
        secret_id: "my-kafka-secret"
        region: "us-east-1"
      schema-secret:
        secret_id: "my-self-managed-kafka-schema"
        region: "us-east-1"
```

Vous pouvez utiliser un plan préconfiguré pour créer ce pipeline. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Migration de données depuis des clusters Kafka dans un VPC

Vous pouvez également utiliser des pipelines OpenSearch d'ingestion pour migrer les données d'un cluster Kafka autogéré exécuté dans un VPC. Pour ce faire, configurez un pipeline d' OpenSearch ingestion avec Kafka autogéré comme source et OpenSearch Service ou OpenSearch Serverless comme destination. Cela traite vos données de streaming depuis un cluster source autogéré vers un domaine ou une collection de destination AWS géré.

Prérequis

Avant de créer votre pipeline OpenSearch d'ingestion, effectuez les étapes suivantes :

1. Créez un cluster Kafka autogéré avec une configuration réseau VPC contenant les données que vous souhaitez ingérer dans Service. OpenSearch
2. Créez un domaine OpenSearch de service ou une collection OpenSearch sans serveur vers lequel vous souhaitez migrer les données. Pour plus d'informations, consultez les sections Création OpenSearch de domaines de service et Création de collections.
3. Configurez l'authentification sur votre cluster autogéré avec AWS Secrets Manager. Activez la rotation des secrets en suivant les étapes décrites dans Rotation AWS Secrets Manager des secrets.
4. Obtenez l'ID du VPC qui a accès à Kafka autogéré. Choisissez le VPC CIDR à utiliser par Ingestion. OpenSearch

> **Note**
>
> Si vous utilisez le AWS Management Console pour créer votre pipeline, vous devez également attacher votre pipeline d' OpenSearch ingestion à votre VPC afin d'utiliser Kafka autogéré. Pour ce faire, recherchez la section Configuration réseau, cochez la case Attacher au VPC et choisissez votre CIDR parmi l'une des options par défaut fournies, ou sélectionnez le vôtre. Vous pouvez utiliser n'importe quel CIDR à partir d'un espace d'adressage privé tel que défini dans la RFC 1918 Best Current Practice.
> Pour fournir un CIDR personnalisé, sélectionnez Autre dans le menu déroulant. Pour éviter une collision d'adresses IP entre OpenSearch l'ingestion et l'adresse autogérée

OpenSearch, assurez-vous que le CIDR OpenSearch VPC autogéré est différent du CIDR pour l'ingestion. OpenSearch

5. Associez une politique basée sur les ressources à votre domaine ou une politique d'accès aux données à votre collection. Ces politiques d'accès permettent à OpenSearch Ingestion d'écrire des données de votre cluster autogéré vers votre domaine ou votre collection.

   L'exemple de politique d'accès au domaine suivant permet au rôle de pipeline, que vous créez à l'étape suivante, d'écrire des données dans un domaine. Assurez-vous de le mettre à jour `resource` avec votre propre ARN.

   JSON

   ```
   {
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
           "AWS": "arn:aws:iam::pipeline-account-id:role/pipeline-role"
         },
         "Action": [
           "es:DescribeDomain",
           "es:ESHttp*"
         ],
         "Resource": [
           "arn:aws:es:region:account-id:domain/domain-name"
         ]
       }
     ]
   }
   ```

   Pour créer un rôle IAM doté des autorisations appropriées pour accéder aux données d'écriture de la collection ou du domaine, consultezthe section called “Configuration des rôles et des utilisateurs”.

Étape 1 : configurer le rôle du pipeline

Une fois les prérequis de votre pipeline configurés, configurez le rôle de pipeline que vous souhaitez utiliser dans la configuration de votre pipeline et ajoutez-y les autorisations suivantes :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": ["arn:aws:secretsmanager:region:account-id:secret:secret-
name"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:*:account-id:network-interface/*",
                "arn:aws:ec2:*:account-id:subnet/*",
                "arn:aws:ec2:*:account-id:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "*"
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
               "StringEquals":
                    {
                        "aws:RequestTag/OSISManaged": "true"
                    }
            }
        }
    ]
}
```

Vous devez fournir les EC2 autorisations Amazon ci-dessus sur le rôle IAM que vous utilisez pour créer le pipeline d' OpenSearch ingestion, car le pipeline utilise ces autorisations pour créer et supprimer une interface réseau dans votre VPC. Le pipeline ne peut accéder au cluster Kafka que via cette interface réseau.

Étape 2 : Création du pipeline

Vous pouvez ensuite configurer un pipeline d' OpenSearch ingestion comme le suivant, qui spécifie Kafka comme source.

Vous pouvez spécifier plusieurs domaines OpenSearch de service comme destinations pour vos données. Cette fonctionnalité permet le routage conditionnel ou la réplication des données entrantes dans plusieurs domaines OpenSearch de service.

Vous pouvez également migrer les données d'un cluster Confluent Kafka source vers une collection VPC OpenSearch sans serveur. Assurez-vous de fournir une politique d'accès au réseau dans la configuration du pipeline. Vous pouvez utiliser un registre de schémas Confluent pour définir un schéma Confluent.

```
 version: "2"
kafka-pipeline:
  source:
    kafka:
      encryption:
        type: "ssl"
```

```
      topics:
        - name: "topic-name"
          group_id: "group-id"
      bootstrap_servers:
        - "bootstrap-server.us-east-1.aws.private.confluent.cloud:9092"
      authentication:
        sasl:
          plain:
            username: ${aws_secrets:confluent-kafka-secret:username}
            password: ${aws_secrets:confluent-kafka-secret:password}
      schema:
        type: confluent
        registry_url: https://my-registry.us-east-1.aws.confluent.cloud
        api_key: "${{aws_secrets:schema-secret:schema_registry_api_key}}"
        api_secret: "${{aws_secrets:schema-secret:schema_registry_api_secret}}"
        basic_auth_credentials_source: "USER_INFO"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
      index: "confluent-index"
extension:
  aws:
    secrets:
      confluent-kafka-secret:
        secret_id: "my-kafka-secret"
        region: "us-east-1"
      schema-secret:
        secret_id: "my-self-managed-kafka-schema"
        region: "us-east-1"
```

Vous pouvez utiliser un plan préconfiguré pour créer ce pipeline. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Migration de données à partir de OpenSearch clusters autogérés à l'aide d'Amazon Ingestion OpenSearch

Vous pouvez utiliser un pipeline Amazon OpenSearch Ingestion autogéré OpenSearch ou Elasticsearch pour migrer des données vers des domaines Amazon OpenSearch Service et OpenSearch des collections sans serveur. OpenSearch Ingestion prend en charge les configurations

de réseau public et privé pour la migration des données depuis Elasticsearch OpenSearch et autogéré.

Migration depuis des clusters publics OpenSearch

Vous pouvez utiliser des pipelines d' OpenSearch ingestion pour migrer des données depuis un cluster autogéré OpenSearch ou Elasticsearch avec une configuration publique, ce qui signifie que le nom DNS du domaine peut être résolu publiquement. Pour ce faire, configurez un pipeline d' OpenSearch ingestion avec Elasticsearch OpenSearch ou autogéré comme source et OpenSearch Service ou OpenSearch Serverless comme destination. Cela permet de migrer efficacement vos données d'un cluster source autogéré vers un domaine ou une AWS collection de destination géré.

Prérequis

Avant de créer votre pipeline OpenSearch d'ingestion, effectuez les étapes suivantes :

1. Créez un cluster autogéré OpenSearch ou Elastisearch contenant les données que vous souhaitez migrer et configurez un nom DNS public. Pour obtenir des instructions, consultez la section Création d'un cluster dans la OpenSearch documentation.
2. Créez un domaine OpenSearch de service ou une collection OpenSearch sans serveur vers lequel vous souhaitez migrer les données. Pour plus d’informations, consultez the section called “ Création de domaines OpenSearch de service” et the section called “Créer des collections”.
3. Configurez l'authentification sur votre cluster autogéré avec AWS Secrets Manager. Activez la rotation des secrets en suivant les étapes décrites dans Rotation AWS Secrets Manager des secrets.
4. Associez une politique basée sur les ressources à votre domaine ou une politique d'accès aux données à votre collection. Ces politiques d'accès permettent à OpenSearch Ingestion d'écrire des données de votre cluster autogéré vers votre domaine ou votre collection.

   L'exemple de politique d'accès au domaine suivant permet au rôle de pipeline, que vous créez à l'étape suivante, d'écrire des données dans un domaine. Assurez-vous de le mettre à jour `resource` avec votre propre ARN.

   JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

```
      "Principal": {
        "AWS": "arn:aws:iam::pipeline-account-id:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:region:account-id:domain/domain-name"
      ]
    }
  ]
}
```

Pour créer un rôle IAM doté des autorisations appropriées pour accéder aux données d'écriture de la collection ou du domaine, consultezthe section called “Configuration des rôles et des utilisateurs”.

Étape 1 : configurer le rôle du pipeline

Une fois les prérequis de votre OpenSearch pipeline définis, configurez le rôle de pipeline que vous souhaitez utiliser dans la configuration de votre pipeline et ajoutez l'autorisation d'écrire dans un domaine de OpenSearch service ou une collection OpenSearch sans serveur, ainsi que l'autorisation de lire les secrets depuis Secrets Manager.

Étape 2 : Création du pipeline

Vous pouvez ensuite configurer un pipeline d' OpenSearch ingestion comme le suivant, qui est OpenSearch spécifié comme source.

Vous pouvez spécifier plusieurs domaines OpenSearch de service comme destinations pour vos données. Cette fonctionnalité permet le routage conditionnel ou la réplication des données entrantes dans plusieurs domaines OpenSearch de service.

Vous pouvez également migrer les données d'une source OpenSearch ou d'un cluster Elasticsearch vers une collection VPC OpenSearch sans serveur. Assurez-vous de fournir une politique d'accès au réseau dans la configuration du pipeline.

```
version: "2"
opensearch-migration-pipeline:
  source:
```

```
    opensearch:
      acknowledgments: true
      host: [ "https://my-self-managed-cluster-name:9200" ]
      indices:
        include:
          - index_name_regex: "include-.*"
        exclude:
          - index_name_regex: '\..*'
      authentication:
        username: ${aws_secrets:secret:username}
        password: ${aws_secrets:secret:password}
        scheduling:
           interval: "PT2H"
           index_read_count: 3
           start_time: "2023-06-02T22:01:30.00Z"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
          #Uncomment the following lines if your destination is an OpenSearch
 Serverless collection
          #serverless: true
          # serverless_options:
          #     network_policy_name: "network-policy-name"
      index: "${getMetadata(\"opensearch-index\")}"
      document_id: "${getMetadata(\"opensearch-document_id\")}"
      enable_request_compression: true
      dlq:
        s3:
          bucket: "bucket-name"
          key_path_prefix: "apache-log-pipeline/logs/dlq"
          region: "us-east-1"
extension:
  aws:
    secrets:
      secret:
        secret_id: "my-opensearch-secret"
        region: "us-east-1"
        refresh_interval: PT1H
```

Vous pouvez utiliser un plan préconfiguré pour créer ce pipeline. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Migration de données depuis des OpenSearch clusters dans un VPC

Vous pouvez également utiliser des pipelines d' OpenSearch ingestion pour migrer des données depuis un cluster autogéré OpenSearch ou Elasticsearch exécuté dans un VPC. Pour ce faire, configurez un pipeline d' OpenSearch ingestion avec Elasticsearch OpenSearch ou autogéré comme source et OpenSearch Service ou OpenSearch Serverless comme destination. Cela permet de migrer efficacement vos données d'un cluster source autogéré vers un domaine ou une AWS collection de destination géré.

Prérequis

Avant de créer votre pipeline OpenSearch d'ingestion, effectuez les étapes suivantes :

1. Créez un cluster autogéré OpenSearch ou Elastisearch avec une configuration réseau VPC contenant les données que vous souhaitez migrer.
2. Créez un domaine OpenSearch de service ou une collection OpenSearch sans serveur vers lequel vous souhaitez migrer les données. Pour plus d'informations, consultez les sections Création OpenSearch de domaines de service et Création de collections.
3. Configurez l'authentification sur votre cluster autogéré avec AWS Secrets Manager. Activez la rotation des secrets en suivant les étapes décrites dans Rotation AWS Secrets Manager des secrets.
4. Obtenez l'ID du VPC ayant accès à Elasticsearch ou autogéré OpenSearch . Choisissez le VPC CIDR à utiliser par Ingestion. OpenSearch

> **Note**
>
> Si vous utilisez le AWS Management Console pour créer votre pipeline, vous devez également associer votre pipeline d' OpenSearch ingestion à votre VPC afin d'utiliser Elasticsearch ou autogéré OpenSearch . Pour ce faire, recherchez la section Options du réseau source, cochez la case Attacher au VPC et choisissez votre CIDR parmi l'une des options par défaut fournies. Vous pouvez utiliser n'importe quel CIDR à partir d'un espace d'adressage privé tel que défini dans la RFC 1918 Best Current Practice.
> Pour fournir un CIDR personnalisé, sélectionnez Autre dans le menu déroulant. Pour éviter une collision d'adresses IP entre OpenSearch l'ingestion et l'adresse autogérée OpenSearch, assurez-vous que le CIDR OpenSearch VPC autogéré est différent du CIDR pour l'ingestion. OpenSearch

5. Associez une politique basée sur les ressources à votre domaine ou une politique d'accès aux données à votre collection. Ces politiques d'accès permettent à OpenSearch Ingestion d'écrire des données de votre cluster autogéré vers votre domaine ou votre collection.

   L'exemple de politique d'accès au domaine suivant permet au rôle de pipeline, que vous créez à l'étape suivante, d'écrire des données dans un domaine. Assurez-vous de le mettre à jour `resource` avec votre propre ARN.

   JSON

   ```
   {
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
           "AWS": "arn:aws:iam::pipeline-account-id:role/pipeline-role"
         },
         "Action": [
           "es:DescribeDomain",
           "es:ESHttp*"
         ],
         "Resource": [
           "arn:aws:es:region:account-id:domain/domain-name"
         ]
       }
     ]
   }
   ```

   Pour créer un rôle IAM doté des autorisations appropriées pour accéder aux données d'écriture de la collection ou du domaine, consultezthe section called “Configuration des rôles et des utilisateurs”.

Étape 1 : configurer le rôle du pipeline

Une fois les prérequis de votre pipeline configurés, configurez le rôle de pipeline que vous souhaitez utiliser dans la configuration de votre pipeline et ajoutez-y les autorisations suivantes :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "SecretsManagerReadAccess",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": ["arn:aws:secretsmanager:region:account-id:secret:secret-
name"]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachNetworkInterface",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeNetworkInterfaces"
            ],
            "Resource": [
                "arn:aws:ec2:*:account-id:network-interface/*",
                "arn:aws:ec2:*:account-id:subnet/*",
                "arn:aws:ec2:*:account-id:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:Describe*"
            ],
            "Resource": "*"
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
               "StringEquals":
                    {
                        "aws:RequestTag/OSISManaged": "true"
                    }
            }
        }
    ]
}
```

Vous devez fournir les EC2 autorisations Amazon ci-dessus sur le rôle IAM que vous utilisez pour créer le pipeline d' OpenSearch ingestion, car le pipeline utilise ces autorisations pour créer et supprimer une interface réseau dans votre VPC. Le pipeline ne peut accéder au OpenSearch cluster que via cette interface réseau.

Étape 2 : Création du pipeline

Vous pouvez ensuite configurer un pipeline d' OpenSearch ingestion comme le suivant, qui est OpenSearch spécifié comme source.

Vous pouvez spécifier plusieurs domaines OpenSearch de service comme destinations pour vos données. Cette fonctionnalité permet le routage conditionnel ou la réplication des données entrantes dans plusieurs domaines OpenSearch de service.

Vous pouvez également migrer les données d'une source OpenSearch ou d'un cluster Elasticsearch vers une collection VPC OpenSearch sans serveur. Assurez-vous de fournir une politique d'accès au réseau dans la configuration du pipeline.

```
version: "2"
opensearch-migration-pipeline:
  source:
    opensearch:
      acknowledgments: true
      host: [ "https://my-self-managed-cluster-name:9200" ]
      indices:
```

```
        include:
          - index_name_regex: "include-.*"
        exclude:
          - index_name_regex: '\..*'
      authentication:
        username: ${aws_secrets:secret:username}
        password: ${aws_secrets:secret:password}
        scheduling:
           interval: "PT2H"
           index_read_count: 3
           start_time: "2023-06-02T22:01:30.00Z"
  sink:
  - opensearch:
      hosts: ["https://search-mydomain.us-east-1.es.amazonaws.com"]
      aws:
          region: "us-east-1"
          #Uncomment the following lines if your destination is an OpenSearch
 Serverless collection
          #serverless: true
          # serverless_options:
          #     network_policy_name: "network-policy-name"
      index: "${getMetadata(\"opensearch-index\")}"
      document_id: "${getMetadata(\"opensearch-document_id\")}"
      enable_request_compression: true
      dlq:
        s3:
          bucket: "bucket-name"
          key_path_prefix: "apache-log-pipeline/logs/dlq"
          region: "us-east-1"
extension:
  aws:
    secrets:
      secret:
        secret_id: "my-opensearch-secret"
        region: "us-east-1"
        refresh_interval: PT1H
```

Vous pouvez utiliser un plan préconfiguré pour créer ce pipeline. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

Utilisation d'un pipeline d' OpenSearch ingestion avec Amazon Kinesis Data Streams

Vous pouvez utiliser le plug-in Kinesis pour diffuser des données depuis Amazon Kinesis Data Streams vers des domaines Amazon OpenSearch Service et des collections sans serveur. OpenSearch Le pipeline extrait les enregistrements d'Amazon Kinesis OpenSearch, les envoie à Amazon Kinesis et génère automatiquement des index en fonction du nom du flux et de la date actuelle.

Connectivité à Amazon Kinesis Data Streams

Vous pouvez utiliser des pipelines d' OpenSearch ingestion pour migrer des données depuis Amazon Kinesis Data Streams avec une configuration publique, ce qui signifie que le nom DNS du domaine peut être résolu publiquement. Pour ce faire, configurez un pipeline d' OpenSearch ingestion avec Amazon Kinesis Data Streams comme source OpenSearch et Service OpenSearch ou Serverless comme destination. Cela traite vos données de streaming depuis un cluster source autogéré vers un domaine ou une collection de destination AWS géré.

Prérequis

Avant de créer votre pipeline OpenSearch d'ingestion, effectuez les étapes suivantes :

1. Créez un flux de données Amazon Kinesis faisant office de source. Le flux doit contenir les données que vous souhaitez ingérer dans le OpenSearch Service.
2. Créez un domaine OpenSearch de service ou une collection OpenSearch sans serveur vers lequel vous souhaitez migrer les données. Pour plus d'informations, consultez les sections Création OpenSearch de domaines de service et Création de collections.
3. Configurez l'authentification sur votre flux de données Amazon Kinesis avec. AWS Secrets Manager Activez la rotation des secrets en suivant les étapes décrites dans Rotation AWS Secrets Manager des secrets.
4. Associez une politique basée sur les ressources à votre domaine ou une politique d'accès aux données à votre collection. Ces politiques d'accès permettent à OpenSearch Ingestion d'écrire des données de votre cluster autogéré vers votre domaine ou votre collection.

   L'exemple de politique d'accès au domaine suivant permet au rôle de pipeline, que vous créez à l'étape suivante, d'écrire des données dans un domaine. Assurez-vous de le mettre à jour `resource` avec votre propre ARN.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::account-id:role/pipeline-role"
      },
      "Action": [
        "es:DescribeDomain",
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:region:account-id:domain/domain-name"
      ]
    }
  ]
}
```

Pour créer un rôle IAM doté des autorisations appropriées pour accéder aux données d'écriture de la collection ou du domaine, consultezthe section called “Configuration des rôles et des utilisateurs”.

Étape 1 : configurer le rôle du pipeline

Une fois les prérequis de votre pipeline Amazon Kinesis Data Streams définis, configurez le rôle de pipeline que vous souhaitez utiliser dans la configuration de votre pipeline et ajoutez l'autorisation d'écrire dans un domaine de service OpenSearch ou OpenSearch une collection sans serveur, ainsi que l'autorisation de lire les secrets de Secrets Manager.

L'autorisation suivante est nécessaire pour écrire dans un compartiment, un domaine et une collection Amazon S3 :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Sid": "allowReadFromStream",
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStream",
                "kinesis:DescribeStreamConsumer",
                "kinesis:DescribeStreamSummary",
                "kinesis:GetRecords",
                "kinesis:GetShardIterator",
                "kinesis:ListShards",
                "kinesis:ListStreams",
                "kinesis:ListStreamConsumers",
                "kinesis:RegisterStreamConsumer",
                "kinesis:SubscribeToShard"
            ],
            "Resource": [
                "arn:aws:kinesis:region:account-id:stream/stream-name"
            ]
        }
    ]
}
```

Si le chiffrement côté serveur est activé pour les flux, la politique KMS suivante déchiffrera les enregistrements des flux :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "allowDecryptionOfCustomManagedKey",
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey"
            ],
            "Resource": "arn:aws:kms:region:account-id:key/key-id"
        }
    ]
}
```

Pour qu'un pipeline puisse écrire des données dans un domaine, celui-ci doit disposer d'une politique d'accès au niveau du domaine qui autorise le rôle de pipeline sts_role_arn à y accéder. L'exemple de politique d'accès au domaine suivant permet au rôle de pipeline nommé`pipeline-role`, que vous avez créé à l'étape précédente, d'écrire des données dans le domaine nommé `ingestion-domain` :

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::account-id:role/pipeline-role"
      },
      "Action": ["es:DescribeDomain", "es:ESHttp*"],
      "Resource": "arn:aws:es:region:account-id:domain/domain-name/*"
    }
  ]
}
```

Étape 2 : Création du pipeline

Vous pouvez ensuite configurer un pipeline d' OpenSearch ingestion en spécifiant Amazon Kinesis comme source. Les attributs de métadonnées disponibles sont les suivants :

- `stream_name`: nom du flux de données Kinesis à partir duquel l'enregistrement est ingéré.
- `partition_key`: clé de partition de l'enregistrement du flux de données Kinesis en cours d'ingestion.
- `sequence_number`: numéro de séquence de l'enregistrement du flux de données Kinesis en cours d'ingestion.
- `sub_sequence_number`: numéro de sous-séquence de l'enregistrement du flux de données Kinesis en cours d'ingestion.

Vous pouvez spécifier plusieurs domaines OpenSearch de service comme destinations pour vos données. Cette fonctionnalité permet le routage conditionnel ou la réplication des données entrantes dans plusieurs domaines OpenSearch de service.

Vous pouvez également migrer les données d'Amazon Kinesis vers une collection VPC OpenSearch sans serveur. Un plan est disponible sur la console OpenSearch d'ingestion pour créer un pipeline. Pour créer un pipeline, vous pouvez utiliser le `AWS-KinesisDataStreamsPipeline` plan suivant.

```
version: "2"
     kinesis_data_streams_pipeline:
       source:
         kinesis_data_streams:
           acknowledgments: true
           codec:
             newline:
           streams:
             - stream_name: "<stream name>"
             - stream_name: "<stream name>"
           aws:
             region: "region"
       sink:
         - opensearch:
             hosts: [ "https://search-mydomain.region.es.amazonaws.com" ]
             index: "index_${getMetadata(\"stream-name\")}"
             document_id: "${getMetadata(\"partition_key\")}"
             aws:
               sts_role_arn: "<<arn:aws:iam::123456789012:role/Example-Role>>"
               region: "region"

               s3:
                 bucket: "dlq-bucket-name"
                 region: "region"
```

Vous pouvez utiliser un plan préconfiguré pour créer ce pipeline. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”. Vous pouvez également consulter la documentation Opensearch open source pour des options de configuration supplémentaires. Pour en savoir plus, consultez la section Options de configuration.

Cohérence des données

OpenSearch L'ingestion prend en charge end-to-end la reconnaissance afin de garantir la durabilité des données. Lorsque le pipeline lit des enregistrements de flux depuis Kinesis, il répartit de manière dynamique le travail de lecture des enregistrements de flux en fonction des partitions associées au flux. Le pipeline vérifie automatiquement les flux lorsqu'il reçoit un accusé de réception après avoir ingéré tous les enregistrements du OpenSearch domaine ou de la collection. Cela permettra d'éviter le double traitement des enregistrements de flux.

> **Note**
>
> Si vous souhaitez créer l'index en fonction du nom du flux, vous pouvez définir l'index dans la section du récepteur d'opensearch comme « index_$ {getMetadata (\ » ***stream_name*** \ »)} ».

(Facultatif) Configurer les unités de calcul recommandées (OCUs) pour le pipeline Kinesis Data Streams

Un minimum de 2 unités de calcul (OCU) est recommandé lors de la création d'un pipeline source Kinesis. Cela permettra de répartir uniformément les enregistrements du flux de données Kinesis par partition traité entre les unités de calcul, garantissant ainsi un mécanisme à faible latence pour l'ingestion des enregistrements de flux.

Un pipeline de sources de flux de OpenSearchKinesis données peut également être configuré pour ingérer des enregistrements de flux provenant de plusieurs flux. Il est recommandé d'ajouter une unité de calcul supplémentaire par nouveau flux.

> **Note**
>
> Si votre pipeline comporte plus d'unités de calcul (OCU) qu'il n'y a de partitions dans l'ensemble de flux configuré dans le pipeline, certaines unités de calcul peuvent rester inactives sans traiter aucun enregistrement de flux par partition.

Étapes suivantes

Après avoir exporté vos données vers un pipeline, vous pouvez les interroger depuis le domaine de OpenSearch service configuré comme récepteur pour le pipeline. Les ressources suivantes peuvent vous aider à démarrer :

- Observabilité
- the section called “Trace Analytics”
- the section called “Langage PPL (Piped Processing Language)”

Utilisation d'un pipeline OpenSearch d'ingestion avec AWS Lambda

Utilisez le AWS Lambda processeur pour enrichir les données provenant de n'importe quelle source ou destination prise en charge par OpenSearch Ingestion à l'aide d'un code personnalisé. Avec le processeur Lambda, vous pouvez appliquer vos propres transformations ou enrichissements de données, puis renvoyer les événements traités dans votre pipeline pour un traitement ultérieur. Ce processeur permet un traitement personnalisé des données et vous donne un contrôle total sur la manière dont les données sont manipulées avant leur transfert dans le pipeline.

> **Note**
>
> La limite de charge utile pour un seul événement traité par un processeur Lambda est de 5 Mo. En outre, le processeur Lambda ne prend en charge que les réponses au format de tableau JSON.

Prérequis

Avant de créer un pipeline avec un processeur Lambda, créez les ressources suivantes :

- Une AWS Lambda fonction qui enrichit et transforme vos données sources. Pour obtenir des instructions, voir Création de votre première fonction Lambda.
- Un domaine OpenSearch de service ou une collection OpenSearch sans serveur qui sera le récepteur du pipeline. Pour plus d'informations, consultez the section called " Création de domaines OpenSearch de service" et the section called "Créer des collections".
- Rôle de pipeline qui inclut les autorisations d'écriture dans le domaine ou le récepteur de collection. Pour de plus amples informations, veuillez consulter the section called "Rôle du pipeline".

  Le rôle de pipeline nécessite également une politique d'autorisation attachée qui lui permet d'appeler la fonction Lambda spécifiée dans la configuration du pipeline. Par exemple :

  JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "allowinvokeFunction",
```

```
            "Effect": "Allow",
            "Action": [
                "lambda:invokeFunction",
                "lambda:InvokeAsync",
                "lambda:ListFunctions"
            ],
            "Resource": "arn:aws:lambda:region:account-id:function:function-
name"

        }
    ]
}
```

Crée un pipeline.

Pour l'utiliser AWS Lambda en tant que processeur, configurez un pipeline d' OpenSearch ingestion et `aws_lambda` spécifiez-le en tant que processeur. Vous pouvez également utiliser le plan d'enrichissement AWS Lambda personnalisé pour créer le pipeline. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

L'exemple de pipeline suivant reçoit des données d'une source HTTP, les enrichit à l'aide d'un processeur de données et du AWS Lambda processeur, et ingère les données traitées dans un OpenSearch domaine.

```
version: "2"
lambda-processor-pipeline:
  source:
    http:
      path: "/${pipelineName}/logs"
  processor:
      - date:
        destination: "@timestamp"
        from_time_received: true
    - aws_lambda:
        function_name: "my-lambda-function"

        tags_on_failure: ["lambda_failure"]
        batch:
            key_name: "events"
        aws:
          region: region
  sink:
```

```
    - opensearch:
        hosts: [ "https://search-mydomain.us-east-1es.amazonaws.com" ]
        index: "table-index"
        aws:
          region: "region"
          serverless: false
```

L'exemple de AWS Lambda fonction suivant transforme les données entrantes en ajoutant une nouvelle paire clé-valeur (`"transformed": "true"`) à chaque élément du tableau d'événements fourni, puis en renvoyant la version modifiée.

```
import json

def lambda_handler(event, context):
    input_array = event.get('events', [])
    output = []
    for input in input_array:
        input["transformed"] = "true";
        output.append(input)

    return output
```

Traitement par lot

Les pipelines envoient des événements par lots au processeur Lambda et ajustent dynamiquement la taille du lot pour s'assurer qu'elle reste inférieure à la limite de 5 Mo.

Voici un exemple de lot de pipeline :

```
batch:
    key_name: "events"

input_arrary = event.get('events', [])
```

> **Note**
>
> Lorsque vous créez un pipeline, assurez-vous que l'`key_name`option de la configuration du processeur Lambda correspond à la clé d'événement du gestionnaire Lambda.

Filtrage conditionnel

Le filtrage conditionnel vous permet de contrôler le moment où votre AWS Lambda processeur appelle la fonction Lambda en fonction de conditions spécifiques dans les données d'événements. Cela est particulièrement utile lorsque vous souhaitez traiter certains types d'événements de manière sélective tout en ignorant d'autres.

L'exemple de configuration suivant utilise le filtrage conditionnel :

```
processors:
  - aws_lambda:
      function_name: "my-lambda-function"
      aws:
        region: "region"
      lambda_when: "/sourceIp == 10.10.10.10"
```

Migration de données entre domaines et collections à l'aide d'Amazon Ingestion OpenSearch

Vous pouvez utiliser des pipelines OpenSearch d'ingestion pour migrer des données entre des domaines Amazon OpenSearch Service ou des collections VPC OpenSearch sans serveur. Pour ce faire, vous configurez un pipeline dans lequel vous configurez un domaine ou une collection comme source, et un autre domaine ou collection comme récepteur. Cela permet de migrer efficacement vos données d'un domaine ou d'une collection à l'autre.

Pour migrer des données, vous devez disposer des ressources suivantes :

- Un domaine de OpenSearch service source ou une collection de VPC OpenSearch sans serveur. Ce domaine ou cette collection contient les données que vous souhaitez migrer. Si vous utilisez un domaine, il doit exécuter la OpenSearch version 1.0 ou ultérieure, ou Elasticsearch version 7.4 ou ultérieure. Le domaine doit également disposer d'une politique d'accès qui accorde les autorisations appropriées à votre rôle de pipeline.
- Domaine ou collection VPC distinct vers lequel vous souhaitez migrer vos données. Ce domaine ou cette collection agira en tant que récepteur du pipeline.
- Rôle de pipeline qu' OpenSearch Ingestion utilisera pour lire et écrire dans votre collection ou votre domaine. Vous incluez l'Amazon Resource Name (ARN) de ce rôle dans la configuration de votre pipeline. Pour plus d’informations, consultez les ressources suivantes :

- the section called “Accorder aux pipelines l'accès aux domaines”
- the section called “Autoriser les pipelines à accéder aux collections”

Rubriques

- Limites
- OpenSearch Le service en tant que source
- Spécification de plusieurs OpenSearch récepteurs de domaine de service
- Migration des données vers une collection OpenSearch VPC sans serveur

Limites

Les limites suivantes s'appliquent lorsque vous désignez des domaines de OpenSearch service ou des collections OpenSearch sans serveur comme récepteurs :

- Un pipeline ne peut pas écrire dans plusieurs domaines VPC.
- Vous pouvez uniquement migrer des données vers ou depuis des collections OpenSearch sans serveur qui utilisent un accès VPC. Les collections publiques ne sont pas prises en charge.
- Vous ne pouvez pas spécifier une combinaison de VPC et de domaines publics dans une configuration de pipeline unique.
- Vous pouvez avoir un maximum de 20 cuvettes hors pipeline dans une seule configuration de pipeline.
- Vous pouvez spécifier des cuvettes parmi un maximum de trois différentes Régions AWS dans une configuration de pipeline unique.
- Un pipeline comportant plusieurs récepteurs peut connaître une réduction de la vitesse de traitement au fil du temps si l'un des récepteurs est indisponible pendant trop longtemps ou s'il n'est pas doté d'une capacité suffisante pour recevoir les données entrantes.

OpenSearch Le service en tant que source

Le domaine ou la collection que vous spécifiez comme source est celui à partir duquel les données sont migrées.

Création d'un rôle de pipeline dans IAM

Pour créer votre pipeline OpenSearch d'ingestion, vous devez d'abord créer un rôle de pipeline pour accorder un accès en lecture et en écriture entre les domaines ou les collections. Pour ce faire, effectuez les opérations suivantes :

1. Créez une nouvelle politique d'autorisation dans IAM à associer au rôle de pipeline. Assurez-vous d'autoriser la lecture à partir de la source et l'écriture dans le récepteur. Pour plus d'informations sur la définition des autorisations de pipeline IAM pour les domaines OpenSearch de service, consultez the section called “Accorder aux pipelines l'accès aux domaines” etthe section called “Autoriser les pipelines à accéder aux collections”.
2. Spécifiez les autorisations suivantes dans le rôle de pipeline pour lire à partir de la source :

   JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"es:ESHttpGet",
         "Resource":[
            "arn:aws:es:region:account-id:domain/domain-name/",
            "arn:aws:es:region:account-id:domain/domain-name/_cat/indices",
            "arn:aws:es:region:account-id:domain/domain-name/_search",
            "arn:aws:es:region:account-id:domain/domain-name/_search/scroll",
            "arn:aws:es:region:account-id:domain/domain-name/*/_search"
         ]
      },
      {
         "Effect":"Allow",
         "Action":"es:ESHttpPost",
         "Resource":[
            "arn:aws:es:region:account-id:domain/domain-name/*/_search/
point_in_time",
            "arn:aws:es:region:account-id:domain/domain-name/*/_search/scroll"
         ]
      },
      {
         "Effect":"Allow",
         "Action":"es:ESHttpDelete",
         "Resource":[
```

```
            "arn:aws:es:region:account-id:domain/domain-name/_search/
point_in_time",
            "arn:aws:es:region:account-id:domain/domain-name/_search/scroll"
         ]
      }
   ]
}
```

Création d'un pipeline

Après avoir attaché la politique au rôle de pipeline, utilisez le plan de AWSOpenSearchDataMigrationPipelinemigration pour créer le pipeline. Ce plan inclut une configuration par défaut pour la migration des données entre les domaines de OpenSearch service ou les collections. Pour de plus amples informations, veuillez consulter the section called “Travailler avec des plans”.

> **Note**
>
> OpenSearch L'ingestion utilise la version et la distribution de votre domaine source pour déterminer le mécanisme à utiliser pour la migration. Certaines versions prennent en charge `point_in_time` cette option. OpenSearch Serverless utilise `search_after` cette option car elle ne prend pas en charge `point_in_time` ou`scroll`.

De nouveaux index sont peut-être en cours de création pendant le processus de migration, ou des documents peuvent être mis à jour pendant la migration. Pour cette raison, vous devrez peut-être effectuer une ou plusieurs analyses des données d'index de votre domaine pour récupérer des données nouvelles ou mises à jour.

Spécifiez le nombre de scans à exécuter en configurant le `index_read_count` et `interval` dans la configuration du pipeline. L'exemple suivant montre comment effectuer plusieurs scans :

```
scheduling:
    interval: "PT2H"
    index_read_count: 3
    start_time: "2023-06-02T22:01:30.00Z"
```

OpenSearch L'ingestion utilise la configuration suivante pour garantir que vos données sont écrites dans le même index et conservent le même identifiant de document :

```
index: "${getMetadata(\"opensearch-index\")}"
document_id: "${getMetadata(\"opensearch-document_id\")}"
```

Spécification de plusieurs OpenSearch récepteurs de domaine de service

Vous pouvez spécifier plusieurs domaines OpenSearch de service public comme destinations pour vos données. Vous pouvez utiliser cette fonctionnalité pour effectuer un routage conditionnel ou répliquer les données entrantes dans plusieurs domaines OpenSearch de service. Vous pouvez spécifier jusqu'à 10 domaines de OpenSearch service public différents en tant que récepteurs.

Dans l'exemple suivant, les données entrantes sont acheminées de manière conditionnelle vers différents domaines de OpenSearch service :

```
...
  route:
    - 2xx_status: "/response >= 200 and /response < 300"
    - 5xx_status: "/response >= 500 and /response < 600"
  sink:
    - opensearch:
        hosts: [ "https://search-response-2xx.region.es.amazonaws.com" ]
        aws:
          region: "us-east-1"
        index: "response-2xx"
        routes:
          - 2xx_status
    - opensearch:
        hosts: [ "https://search-response-5xx.region.es.amazonaws.com" ]
        aws:
          region: "us-east-1"
        index: "response-5xx"
        routes:
          - 5xx_status
```

Migration des données vers une collection OpenSearch VPC sans serveur

Vous pouvez utiliser OpenSearch Ingestion pour migrer les données d'un domaine de OpenSearch service source ou d'une collection OpenSearch sans serveur vers un récepteur de collecte VPC. Vous devez fournir une politique d'accès au réseau dans la configuration du pipeline. Pour plus d'informations sur l'ingestion de données dans des collections VPC OpenSearch sans serveur, consultez. the section called “Tutoriel : Ingérer des données dans une collection”

Pour migrer des données vers une collection VPC

1. Créez une collection OpenSearch sans serveur. Pour obtenir des instructions, veuillez consulter the section called “Tutoriel : Ingérer des données dans une collection”.
2. Créez une politique réseau pour la collection qui spécifie l'accès VPC à la fois au point de terminaison de la collection et au point de terminaison des tableaux de bord. Pour obtenir des instructions, veuillez consulter the section called “Accès réseau”.
3. Créez le rôle de pipeline si vous n'en avez pas déjà un. Pour obtenir des instructions, veuillez consulter the section called “Rôle du pipeline”.
4. Créez le pipeline. Pour obtenir des instructions, veuillez consulter the section called “Travailler avec des plans”.

Utilisation du AWS SDKs pour interagir avec Amazon OpenSearch Ingestion

Cette section inclut un exemple d'utilisation du AWS SDKs pour interagir avec Amazon OpenSearch Ingestion. L'exemple de code montre comment créer un domaine et un pipeline, puis comment intégrer des données dans le pipeline.

Rubriques

- Python

Python

L'exemple de script suivant utilise le AWS SDK pour Python (Boto3)pour créer un rôle de pipeline IAM, un domaine dans lequel écrire des données et un pipeline pour ingérer des données. Il ingère ensuite un exemple de fichier journal dans le pipeline à l'aide de la bibliothèque `requests` HTTP.

Pour installer les dépendances requises, exécutez les commandes suivantes :

```
pip install boto3
pip install botocore
pip install requests
pip install requests-auth-aws-sigv4
```

Dans le script, remplacez toutes les instances de *`account-id`* par votre Compte AWS identifiant.

```
import boto3
import botocore
from botocore.config import Config
import requests
from requests_auth_aws_sigv4 import AWSSigV4
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

opensearch = boto3.client('opensearch', config=my_config)
iam = boto3.client('iam', config=my_config)
osis = boto3.client('osis', config=my_config)

domainName = 'test-domain'  # The name of the domain
pipelineName = 'test-pipeline' # The name of the pipeline

def createPipelineRole(iam, domainName):
    """Creates the pipeline role"""
    response = iam.create_policy(
        PolicyName='pipeline-policy',
        PolicyDocument=f'{{\"Version\":\"2012-10-17\",\"Statement\":[{{\"Effect\":
\"Allow\",\"Action\":\"es:DescribeDomain\",\"Resource\":\"arn:aws:es:us-east-1:account-
id:domain\/{domainName}\"}},{{\"Effect\":\"Allow\",\"Action\":\"es:ESHttp*\",\"Resource
\":\"arn:aws:es:us-east-1:account-id:domain\/{domainName}\/*\"}}]}}'
    )
    policyarn = response['Policy']['Arn']

    response = iam.create_role(
        RoleName='PipelineRole',
        AssumeRolePolicyDocument='{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Principal\":{\"Service\":\"osis-pipelines.amazonaws.com\"},\"Action\":
\"sts:AssumeRole\"}]}'
    )
    rolename=response['Role']['RoleName']

    response = iam.attach_role_policy(
        RoleName=rolename,
        PolicyArn=policyarn
    )

    print('Creating pipeline role...')
```

```
    time.sleep(10)
    print('Role created: ' + rolename)

def createDomain(opensearch, domainName):
    """Creates a domain to ingest data into"""
    response = opensearch.create_domain(
        DomainName=domainName,
        EngineVersion='OpenSearch_2.3',
        ClusterConfig={
            'InstanceType': 't2.small.search',
            'InstanceCount': 5,
            'DedicatedMasterEnabled': True,
            'DedicatedMasterType': 't2.small.search',
            'DedicatedMasterCount': 3
        },
        # Many instance types require EBS storage.
        EBSOptions={
            'EBSEnabled': True,
            'VolumeType': 'gp2',
            'VolumeSize': 10
        },
        AccessPolicies=f'{{\"Version\":\"2012-10-17\",\"Statement\":[{{\"Effect
\":\"Allow\",\"Principal\":{{\"AWS\":\"arn:aws:iam::account-id:role\/PipelineRole
\"}},\"Action\":\"es:*\",\"Resource\":\"arn:aws:es:us-east-1:account-id:domain\/
{domainName}\/*\"}}]}}',
        NodeToNodeEncryptionOptions={
            'Enabled': True
        }
    )
    return(response)

def waitForDomainProcessing(opensearch, domainName):
    """Waits for the domain to be active"""
    try:
        response = opensearch.describe_domain(
            DomainName=domainName
        )
        # Every 30 seconds, check whether the domain is processing.
        while 'Endpoint' not in response['DomainStatus']:
            print('Creating domain...')
            time.sleep(60)
            response = opensearch.describe_domain(
                DomainName=domainName)
```

```
        # Once we exit the loop, the domain is ready for ingestion.
        endpoint = response['DomainStatus']['Endpoint']
        print('Domain endpoint ready to receive data: ' + endpoint)
        createPipeline(osis, endpoint)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceNotFoundException':
            print('Domain not found.')
        else:
            raise error

def createPipeline(osis, endpoint):
    """Creates a pipeline using the domain and pipeline role"""
    try:
        definition = f'version: \"2\"\nlog-pipeline:\n  source:\n    http:\n      path:
 \"/${{pipelineName}}/logs\"\n  processor:\n    - date:\n        from_time_received:
 true\n        destination: \"@timestamp\"\n  sink:\n    - opensearch:\n        hosts:
 [ \"https://{endpoint}\" ]\n        index: \"application_logs\"\n        aws:\n
   region: \"us-east-1\"'
        response = osis.create_pipeline(
            PipelineName=pipelineName,
            MinUnits=4,
            MaxUnits=9,
            PipelineConfigurationBody=definition,
            PipelineRoleArn="arn:aws:iam::account-id:role/PipelineRole"
        )

        response = osis.get_pipeline(
                PipelineName=pipelineName
        )

        # Every 30 seconds, check whether the pipeline is active.
        while response['Pipeline']['Status'] == 'CREATING':
            print('Creating pipeline...')
            time.sleep(30)
            response = osis.get_pipeline(
                PipelineName=pipelineName)

        # Once we exit the loop, the pipeline is ready for ingestion.
        ingestionEndpoint = response['Pipeline']['IngestEndpointUrls'][0]
        print('Pipeline ready to ingest data at endpoint: ' + ingestionEndpoint)
        ingestData(ingestionEndpoint)

    except botocore.exceptions.ClientError as error:
```

```
        if error.response['Error']['Code'] == 'ResourceAlreadyExistsException':
            print('Pipeline already exists.')
            response = osis.get_pipeline(
                PipelineName=pipelineName
            )
            ingestionEndpoint = response['Pipeline']['IngestEndpointUrls'][0]
            ingestData(ingestionEndpoint)
        else:
            raise error


def ingestData(ingestionEndpoint):
    """Ingests a sample log file into the pipeline"""
    endpoint = 'https://' + ingestionEndpoint
    r = requests.request('POST', f'{endpoint}/log-pipeline/logs',

 data='[{"time":"2014-08-11T11:40:13+00:00","remote_addr":"122.226.223.69","status":"404","requ
 http://www.k2proxy.com//hello.html HTTP/1.1","http_user_agent":"Mozilla/4.0
 (compatible; WOW64; SLCC2;)"}]',
    auth=AWSSigV4('osis'))
    print('Ingesting sample log file into pipeline')
    print('Response: ' + r.text)

def main():
    createPipelineRole(iam, domainName)
    createDomain(opensearch, domainName)
    waitForDomainProcessing(opensearch, domainName)

if __name__ == "__main__":
    main()
```

Sécurité dans Amazon OpenSearch Ingestion

La sécurité du cloud AWS est la priorité absolue. En tant que AWS client, vous bénéficiez d'un centre de données et d'une architecture réseau conçus pour répondre aux exigences des entreprises les plus sensibles en matière de sécurité.

La sécurité est une responsabilité partagée entre vous AWS et vous. Le modèle de responsabilité partagée décrit cette notion par les termes sécurité du cloud et sécurité dans le cloud :

- Sécurité du cloud : AWS est chargée de protéger l'infrastructure qui exécute les AWS services dans le AWS cloud. AWS vous fournit également des services que vous pouvez utiliser en toute

sécurité. Des auditeurs tiers testent et vérifient régulièrement l'efficacité de notre sécurité dans le cadre des programmes de conformitéAWS.

- Sécurité dans le cloud — Votre responsabilité est déterminée par le AWS service que vous utilisez. Vous êtes également responsable d'autres facteurs, y compris de la sensibilité de vos données, des exigences de votre entreprise, ainsi que de la législation et de la réglementation applicables.

Cette documentation vous aide à comprendre comment appliquer le modèle de responsabilité partagée lors de l'utilisation d' OpenSearch Ingestion. Les rubriques suivantes expliquent comment configurer OpenSearch Ingestion pour atteindre vos objectifs de sécurité et de conformité. Vous apprendrez également à utiliser d'autres AWS services qui vous aident à surveiller et à sécuriser vos ressources OpenSearch d'ingestion.

Rubriques

- Configuration de l'accès VPC pour les pipelines Amazon Ingestion OpenSearch
- Identity and Access Management pour Amazon OpenSearch Ingestion
- Journalisation des appels d' OpenSearch API Amazon Ingestion à l'aide de AWS CloudTrail

Configuration de l'accès VPC pour les pipelines Amazon Ingestion OpenSearch

Vous pouvez accéder à vos pipelines Amazon OpenSearch Ingestion à l'aide d'un point de terminaison d'un VPC d'interface. Un VPC est un réseau virtuel dédié à votre. Compte AWS Il est logiquement isolé des autres réseaux virtuels dans le AWS cloud. L'accès à un pipeline via un point de terminaison de VPC permet une communication sécurisée entre OpenSearch Ingestion et les autres services au sein du VPC, sans avoir besoin d'une passerelle Internet, d'un périphérique NAT ou d'une connexion VPN. Tout le trafic reste en sécurité dans le AWS Cloud.

OpenSearch L'ingestion établit cette connexion privée en créant un point de terminaison d'interface optimisé par AWS PrivateLink. Nous créons une interface réseau du point de terminaison dans chaque sous-réseau que vous spécifiez lors de la création du pipeline. Il s'agit d'interfaces réseau gérées par le demandeur qui servent de point d'entrée pour le trafic destiné au pipeline d' OpenSearch ingestion. Vous pouvez également choisir de créer et de gérer les points de terminaison d'interface.

L'utilisation d'un VPC vous permet d'appliquer le flux de données à travers vos pipelines d' OpenSearch ingestion dans les limites du VPC, plutôt que sur Internet public. Les pipelines qui ne

font pas partie d'un VPC envoient et reçoivent des données via des points de terminaison publics et Internet.

Un pipeline avec accès VPC peut écrire dans des domaines publics ou de OpenSearch service VPC, ainsi que dans des collections publiques ou VPC sans serveur. OpenSearch

Rubriques

- Considérations
- Limites
- Prérequis
- Configuration de l'accès VPC pour un pipeline
- Points de terminaison de VPC autogérés
- Rôle lié à un service pour l'accès VPC

Considérations

Tenez compte des éléments suivants lorsque vous configurez l'accès VPC pour un pipeline.

- Un pipeline n'a pas besoin de se trouver dans le même VPC que son récepteur. Vous n'avez pas non plus besoin d'établir une connexion entre les deux VPCs. OpenSearch Ingestion se charge de les connecter pour vous.
- Vous ne pouvez spécifier qu'un seul VPC pour votre pipeline.
- Contrairement aux pipelines publics, un pipeline VPC doit se trouver dans le même emplacement Région AWS que le domaine ou le récepteur de collection dans lequel il écrit.
- Vous pouvez choisir de déployer un pipeline dans un, deux ou trois sous-réseaux de votre VPC. Les sous-réseaux sont répartis dans les mêmes zones de disponibilité que celles dans lesquelles vos unités de OpenSearch calcul d'ingestion (OCUs) sont déployées.
- Si vous déployez un pipeline uniquement dans un sous-réseau et que la zone de disponibilité tombe en panne, vous ne pourrez pas ingérer de données. Pour garantir une haute disponibilité, nous vous recommandons de configurer des pipelines avec deux ou trois sous-réseaux.
- La spécification d'un groupe de sécurité est facultative. Si vous ne fournissez pas de groupe de sécurité, OpenSearch Ingestion utilise le groupe de sécurité par défaut spécifié dans le VPC.

Limites

Les pipelines avec accès VPC présentent les limitations suivantes.

- Vous ne pouvez pas modifier la configuration réseau d'un pipeline après l'avoir créé. Si vous lancez un pipeline au sein d'un VPC, vous ne pourrez pas le transformer ultérieurement en point de terminaison public, et vice versa.
- Vous pouvez lancer votre pipeline avec un point de terminaison VPC d'interface ou un point de terminaison public, mais vous ne pouvez pas faire les deux. Vous devez choisir l'une ou l'autre lorsque vous créez un pipeline.
- Après avoir configuré un pipeline avec accès VPC, vous ne pouvez plus le déplacer vers un autre VPC, et vous ne pouvez pas modifier ses paramètres de sous-réseaux ou de groupes de sécurité.
- Si votre pipeline écrit vers un récepteur de domaine ou de collection qui utilise un accès VPC, vous ne pouvez pas revenir en arrière plus tard et modifier le récepteur (VPC ou public) une fois le pipeline créé. Vous devez supprimer et recréer le pipeline avec un nouveau récepteur. Vous pouvez toujours passer d'un récepteur public à un récepteur avec accès VPC.
- Vous ne pouvez pas fournir un accès d'ingestion entre comptes aux pipelines VPC.

Prérequis

Avant de pouvoir provisionner un pipeline avec accès VPC, vous devez effectuer les opérations suivantes :

- Créer un VPC

  Pour créer le VPC, vous pouvez utiliser la console Amazon VPC, AWS CLI ou un des. AWS SDKs Pour plus d'informations, consultez la section Travailler avec VPCs dans le guide de l'utilisateur Amazon VPC. Si vous avez déjà un VPC, vous pouvez ignorer cette étape.

- Réserver des adresses IP

  OpenSearch Ingestion place une interface Elastic Network dans chaque sous-réseau que vous spécifiez lors de la création du pipeline. Chaque interface réseau est associée à une adresse IP. Vous devez réserver une adresse IP par sous-réseau pour les interfaces réseau.

Configuration de l'accès VPC pour un pipeline

Vous pouvez activer l'accès VPC pour un pipeline dans la console de OpenSearch service ou à l'aide du. AWS CLI

console

Vous configurez l'accès au VPC lors de la création du pipeline. Sous Options du réseau source, choisissez l'accès VPC et configurez les paramètres suivants :

Paramètre	Description
Gestion des terminaux	Choisissez si vous souhaitez créer vous-même vos points de terminaison VPC ou laisser Ingestion les créer OpenSearch pour vous.
VPC	Choisissez le cloud privé virtuel (VPC) que vous souhaitez utiliser. Le VPC et le pipeline doivent se trouver dans le même. Région AWS
Sous-réseaux	Sélectionnez un ou plusieurs sous-réseaux. OpenSearch Service place un point de terminaison de VPC et des interfaces réseau Elastic dans les sous-réseaux.
Groupes de sécurité	Choisissez un ou plusieurs groupes de sécurité VPC qui permettent à l'application requise d'atteindre le pipeline d' OpenSearch ingestion sur les ports (80 ou 443) et les protocoles (HTTP ou HTTPs) exposés par le pipeline.
Options de fixation en VPC	Si votre source nécessite une communication inter-VPC, telle qu'Amazon DocumentDB OpenSearch autogérée ou Confluent Kafka OpenSearch , Ingestion crée des interfaces réseau élastiques ENIs () dans les sous-rése aux que vous spécifiez afin de vous connecter à ces sources. OpenSearch Utilisations d'ingestion ENIs dans chaque zone de disponibilité pour atteindre les sources spécifiées. L'option Attacher au VPC connecte le VPC du plan de données OpenSearch d'ingestion au VPC que vous avez spécifié. Sélectionnez une réservation CIDR pour le VPC géré afin de déployer l'interface réseau.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour configurer l'accès au VPC à l'aide du AWS CLI, spécifiez le `--vpc-options` paramètre :

```
aws osis create-pipeline \
  --pipeline-name vpc-pipeline \
  --min-units 4 \
  --max-units 10 \
  --vpc-options
 SecurityGroupIds={sg-12345678,sg-9012345},SubnetIds=subnet-1212234567834asdf \
  --pipeline-configuration-body "file://pipeline-config.yaml"
```

Points de terminaison de VPC autogérés

Lorsque vous créez un pipeline, vous pouvez utiliser la gestion des terminaux pour créer un pipeline avec des points de terminaison autogérés ou des points de terminaison gérés par des services. La gestion des terminaux est facultative et par défaut, ce sont les points de terminaison gérés par OpenSearch Ingestion.

Pour créer un pipeline avec un point de terminaison VPC autogéré dans AWS Management Console le, consultez la section Création de pipelines avec OpenSearch la console de service. Pour créer un pipeline avec un point de terminaison VPC autogéré dans AWS CLI le, vous pouvez utiliser le paramètre de `--vpc-options` la commande create-pipeline :

```
--vpc-options SubnetIds=subnet-abcdef01234567890,VpcEndpointManagement=CUSTOMER
```

Vous pouvez créer vous-même un point de terminaison pour votre pipeline lorsque vous spécifiez le service du point de terminaison. Pour trouver votre service de point de terminaison, utilisez la commande get-pipeline, qui renvoie une réponse similaire à la suivante :

```
"vpcEndpointService" : "com.amazonaws.osis.us-east-1.pipeline-
id-1234567890abcdef1234567890",
"vpcEndpoints" : [
  {
    "vpcId" : "vpc-1234567890abcdef0",
    "vpcOptions" : {
      "subnetIds" : [ "subnet-abcdef01234567890", "subnet-021345abcdef6789" ],
      "vpcEndpointManagement" : "CUSTOMER"
    }
  }
```

Utilisez le `vpcEndpointService` from de la réponse pour créer un point de terminaison VPC avec le AWS Management Console ou. AWS CLI

Si vous utilisez des points de terminaison VPC autogérés, vous devez activer les `enableDnsSupport` attributs DNS `enableDnsHostnames` dans votre VPC. Notez que si vous avez un pipeline avec un point de terminaison autogéré que vous arrêtez et redémarrez, vous devez recréer le point de terminaison VPC dans votre compte.

Rôle lié à un service pour l'accès VPC

Un rôle lié à un service est un type unique de rôle IAM qui délègue des autorisations à un service afin qu'il puisse créer et gérer des ressources en votre nom. Si vous choisissez un point de terminaison de VPC géré par un service, OpenSearch Ingestion nécessite un rôle lié à un service appelé pour AWSServiceRoleForAmazonOpenSearchIngestionServiceaccéder à votre VPC, créer le point de terminaison du pipeline et placer des interfaces réseau dans un sous-réseau de votre VPC.

Si vous choisissez un point de terminaison VPC autogéré OpenSearch , Ingestion nécessite un rôle lié à un service appelé. AWSServiceRoleForOpensearchIngestionSelfManagedVpce Pour plus d'informations sur ces rôles, leurs autorisations et la procédure à suivre pour les supprimer, consultezthe section called “Rôle de création d'un pipeline”.

OpenSearch L'ingestion crée automatiquement le rôle lorsque vous créez un pipeline d'ingestion. Pour que cette création automatique réussisse, l'utilisateur qui crée le premier pipeline dans un compte doit disposer des autorisations nécessaires à l'`iam:CreateServiceLinkedRole`action. Pour en savoir plus, consultez Autorisations de rôles liés à un service dans le Guide de l'utilisateur IAM. Vous pouvez consulter le rôle dans la console AWS Identity and Access Management (IAM) une fois qu'il a été créé.

Identity and Access Management pour Amazon OpenSearch Ingestion

AWS Identity and Access Management (IAM) est un outil Service AWS qui permet à un administrateur de contrôler en toute sécurité l'accès aux AWS ressources. Les administrateurs IAM contrôlent qui peut être authentifié (connecté) et autorisé (autorisé) à utiliser les ressources d' OpenSearch ingestion. IAM est un Service AWS outil que vous pouvez utiliser sans frais supplémentaires.

Rubriques

- Politiques basées sur l'identité pour l'ingestion OpenSearch

- Actions politiques relatives à OpenSearch l'ingestion
- Ressources relatives aux politiques relatives à OpenSearch l'ingestion
- Clés de conditions de politique pour Amazon OpenSearch Ingestion
- ABAC avec ingestion OpenSearch
- Utilisation d'informations d'identification temporaires avec OpenSearch Ingestion
- Rôles liés à un service pour l'ingestion OpenSearch
- Exemples de politiques basées sur l'identité pour l'ingestion OpenSearch

Politiques basées sur l'identité pour l'ingestion OpenSearch

Prend en charge les politiques basées sur l’identité : oui

Les politiques basées sur l’identité sont des documents de politique d’autorisations JSON que vous pouvez attacher à une identité telle qu’un utilisateur, un groupe d’utilisateurs ou un rôle IAM. Ces politiques contrôlent quel type d’actions des utilisateurs et des rôles peuvent exécuter, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l’identité, consultez Définition d’autorisations IAM personnalisées avec des politiques gérées par le client dans le Guide de l’utilisateur IAM.

Avec les politiques IAM basées sur l’identité, vous pouvez spécifier des actions et ressources autorisées ou refusées, ainsi que les conditions dans lesquelles les actions sont autorisées ou refusées. Vous ne pouvez pas spécifier le principal dans une politique basée sur une identité, car celle-ci s’applique à l’utilisateur ou au rôle auquel elle est attachée. Pour découvrir tous les éléments que vous utilisez dans une politique JSON, consultez Références des éléments de politique JSON IAM dans le Guide de l’utilisateur IAM.

Exemples de politiques basées sur l'identité pour l'ingestion OpenSearch

Pour consulter des exemples de politiques basées sur OpenSearch l'identité d'ingestion, consultez. the section called “Exemples de politiques basées sur l’identité”

Actions politiques relatives à OpenSearch l'ingestion

Prend en charge les actions de politique : oui

L’élément `Action` d’une politique JSON décrit les actions que vous pouvez utiliser pour autoriser ou refuser l’accès à une politique. Les actions de stratégie portent généralement le même nom

que l'opération AWS d'API associée. Il existe quelques exceptions, telles que les actions avec autorisations uniquement qui n'ont pas d'opération API correspondante. Certaines opérations nécessitent également plusieurs actions dans une politique. Ces actions supplémentaires sont nommées actions dépendantes.

Intégration d'actions dans une politique afin d'accorder l'autorisation d'exécuter les opérations associées.

Dans OpenSearch Ingestion, les actions stratégiques utilisent le préfixe suivant avant l'action :

```
osis
```

Pour indiquer plusieurs actions dans une seule déclaration, séparez-les par des virgules.

```
"Action": [
      "osis:action1",
      "osis:action2"
         ]
```

Vous pouvez préciser plusieurs actions à l'aide de caractères génériques (*). Par exemple, pour spécifier toutes les actions qui commencent par le mot `List`, incluez l'action suivante :

```
"Action": "osis:List*"
```

Pour consulter des exemples de politiques basées sur OpenSearch l'identité d'ingestion, consultez. Exemples de politiques basées sur l'identité pour Serverless OpenSearch

Ressources relatives aux politiques relatives à OpenSearch l'ingestion

Prend en charge les ressources de politique : oui

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément de politique JSON `Resource` indique le ou les objets auxquels l'action s'applique. Les instructions doivent inclure un élément `Resource` ou `NotResource`. Il est recommandé de définir une ressource à l'aide de son Amazon Resource Name (ARN). Vous pouvez le faire pour des actions

qui prennent en charge un type de ressource spécifique, connu sous la dénomination autorisations de niveau ressource.

Pour les actions qui ne sont pas compatibles avec les autorisations de niveau ressource, telles que les opérations de liste, utilisez un caractère générique (*) afin d'indiquer que l'instruction s'applique à toutes les ressources.

```
"Resource": "*"
```

Clés de conditions de politique pour Amazon OpenSearch Ingestion

Prend en charge les clés de condition de politique spécifiques au service : Non

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Condition` (ou le bloc `Condition`) vous permet de spécifier des conditions lorsqu'une instruction est appliquée. L'élément `Condition` est facultatif. Vous pouvez créer des expressions conditionnelles qui utilisent des opérateurs de condition, tels que les signes égal ou inférieur à, pour faire correspondre la condition de la politique aux valeurs de la demande.

Si vous spécifiez plusieurs éléments `Condition` dans une instruction, ou plusieurs clés dans un seul élément `Condition`, AWS les évalue à l'aide d'une opération `AND` logique. Si vous spécifiez plusieurs valeurs pour une seule clé de condition, AWS évalue la condition à l'aide d'une `OR` opération logique. Toutes les conditions doivent être remplies avant que les autorisations associées à l'instruction ne soient accordées.

Vous pouvez aussi utiliser des variables d'espace réservé quand vous spécifiez des conditions. Par exemple, vous pouvez accorder à un utilisateur IAM l'autorisation d'accéder à une ressource uniquement si elle est balisée avec son nom d'utilisateur IAM. Pour plus d'informations, consultez Éléments d'une politique IAM : variables et identifications dans le Guide de l'utilisateur IAM.

AWS prend en charge les clés de condition globales et les clés de condition spécifiques au service. Pour voir toutes les clés de condition AWS globales, voir les clés de contexte de condition AWS globales dans le guide de l'utilisateur IAM.

Pour consulter la liste des clés de condition d' OpenSearch ingestion, consultez la section Clés de condition pour Amazon OpenSearch Ingestion dans la référence d'autorisation de service. Pour

savoir avec quelles actions et ressources vous pouvez utiliser une clé de condition, consultez Actions définies par Amazon OpenSearch Ingestion.

ABAC avec ingestion OpenSearch

Prise en charge d'ABAC (balises dans les politiques) : Oui

Le contrôle d'accès par attributs (ABAC) est une stratégie d'autorisation qui définit des autorisations en fonction des attributs. Dans AWS, ces attributs sont appelés balises. Vous pouvez associer des balises aux entités IAM (utilisateurs ou rôles) et à de nombreuses AWS ressources. L'étiquetage des entités et des ressources est la première étape d'ABAC. Vous concevez ensuite des politiques ABAC pour autoriser des opérations quand l'identification du principal correspond à celle de la ressource à laquelle il tente d'accéder.

L'ABAC est utile dans les environnements qui connaissent une croissance rapide et pour les cas où la gestion des politiques devient fastidieuse.

Pour contrôler l'accès basé sur des étiquettes, vous devez fournir les informations d'étiquette dans l'élément de condition d'une politique utilisant les clés de condition `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` ou `aws:TagKeys`.

Si un service prend en charge les trois clés de condition pour tous les types de ressources, alors la valeur pour ce service est Oui. Si un service prend en charge les trois clés de condition pour certains types de ressources uniquement, la valeur est Partielle.

Pour plus d'informations sur ABAC, consultez Définition d'autorisations avec l'autorisation ABAC dans le Guide de l'utilisateur IAM. Pour accéder à un didacticiel décrivant les étapes de configuration de l'ABAC, consultez Utilisation du contrôle d'accès par attributs (ABAC) dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur le balisage des ressources OpenSearch d'ingestion, consultezthe section called "Pipelines de balisage".

Utilisation d'informations d'identification temporaires avec OpenSearch Ingestion

Prend en charge les informations d'identification temporaires : oui

Certains Services AWS ne fonctionnent pas lorsque vous vous connectez à l'aide d'informations d'identification temporaires. Pour plus d'informations, y compris celles qui Services AWS fonctionnent avec des informations d'identification temporaires, consultez Services AWS la section relative à l'utilisation d'IAM dans le guide de l'utilisateur d'IAM.

Vous utilisez des informations d'identification temporaires si vous vous connectez à l' AWS Management Console aide d'une méthode autre qu'un nom d'utilisateur et un mot de passe. Par exemple, lorsque vous accédez à AWS l'aide du lien d'authentification unique (SSO) de votre entreprise, ce processus crée automatiquement des informations d'identification temporaires. Vous créez également automatiquement des informations d'identification temporaires lorsque vous vous connectez à la console en tant qu'utilisateur, puis changez de rôle. Pour plus d'informations sur le changement de rôle, consultez Passage d'un rôle utilisateur à un rôle IAM (console) dans le Guide de l'utilisateur IAM.

Vous pouvez créer manuellement des informations d'identification temporaires à l'aide de l' AWS API AWS CLI or. Vous pouvez ensuite utiliser ces informations d'identification temporaires pour y accéder AWS. AWS recommande de générer dynamiquement des informations d'identification temporaires au lieu d'utiliser des clés d'accès à long terme. Pour plus d'informations, consultez Informations d'identification de sécurité temporaires dans IAM.

Rôles liés à un service pour l'ingestion OpenSearch

Prend en charge les rôles liés aux services : Oui

Un rôle lié à un service est un type de rôle de service lié à un. Service AWS Le service peut endosser le rôle afin d'effectuer une action en votre nom. Les rôles liés au service apparaissent dans votre Compte AWS fichier et appartiennent au service. Un administrateur IAM peut consulter, mais ne peut pas modifier, les autorisations concernant les rôles liés à un service.

OpenSearch L'ingestion utilise un rôle lié à un service appelé. `AWSServiceRoleForAmazonOpenSearchIngestionService` Le rôle lié à un service nommé `AWSServiceRoleForOpensearchIngestionSelfManagedVpce` est également disponible pour les pipelines dotés de points de terminaison VPC autogérés. Pour plus de détails sur la création et la gestion des OpenSearch rôles liés au service Ingestion, consultez. the section called "Rôle de création d'un pipeline"

Exemples de politiques basées sur l'identité pour l'ingestion OpenSearch

Par défaut, les utilisateurs et les rôles ne sont pas autorisés à créer ou à modifier des ressources OpenSearch d'ingestion. Ils ne peuvent pas non plus effectuer de tâches à l'aide de l'API AWS Management Console, AWS Command Line Interface (AWS CLI) ou de AWS l'API. Pour octroyer aux utilisateurs des autorisations d'effectuer des actions sur les ressources dont ils ont besoin, un administrateur IAM peut créer des politiques IAM. L'administrateur peut ensuite ajouter les politiques IAM aux rôles et les utilisateurs peuvent assumer les rôles.

Pour apprendre à créer une politique basée sur l'identité IAM à l'aide de ces exemples de documents de politique JSON, consultez Création de politiques IAM (console) dans le Guide de l'utilisateur IAM.

Pour plus de détails sur les actions et les types de ressources définis par Amazon OpenSearch Ingestion, y compris le format ARNs de chaque type de ressource, consultez la section Actions, ressources et clés de condition pour Amazon OpenSearch Ingestion dans la référence d'autorisation de service.

Rubriques

- Bonnes pratiques en matière de stratégies
- Utilisation d' OpenSearch Ingestion dans la console
- Administration des pipelines OpenSearch d'ingestion
- Ingestion de données dans un pipeline d' OpenSearch ingestion

Bonnes pratiques en matière de stratégies

Les politiques basées sur l'identité sont très puissantes. Ils déterminent si quelqu'un peut créer, accéder ou supprimer des ressources OpenSearch d'ingestion dans votre compte. Ces actions peuvent entraîner des frais pour votre Compte AWS. Lorsque vous créez ou modifiez des politiques basées sur l'identité, suivez ces instructions et recommandations :

Les politiques basées sur l'identité déterminent si quelqu'un peut créer, accéder ou supprimer des ressources OpenSearch d'ingestion dans votre compte. Ces actions peuvent entraîner des frais pour votre Compte AWS. Lorsque vous créez ou modifiez des politiques basées sur l'identité, suivez ces instructions et recommandations :

- Commencez AWS par les politiques gérées et passez aux autorisations du moindre privilège : pour commencer à accorder des autorisations à vos utilisateurs et à vos charges de travail, utilisez les politiques AWS gérées qui accordent des autorisations pour de nombreux cas d'utilisation courants. Ils sont disponibles dans votre Compte AWS. Nous vous recommandons de réduire davantage les autorisations en définissant des politiques gérées par les AWS clients spécifiques à vos cas d'utilisation. Pour plus d'informations, consultez politiques gérées par AWS ou politiques gérées par AWS pour les activités professionnelles dans le Guide de l'utilisateur IAM.
- Accordez les autorisations de moindre privilège : lorsque vous définissez des autorisations avec des politiques IAM, accordez uniquement les autorisations nécessaires à l'exécution d'une seule tâche. Pour ce faire, vous définissez les actions qui peuvent être entreprises sur des ressources spécifiques dans des conditions spécifiques, également appelées autorisations de moindre

privilège. Pour plus d'informations sur l'utilisation d'IAM pour appliquer des autorisations, consultez politiques et autorisations dans IAM dans le Guide de l'utilisateur IAM.

- Utilisez des conditions dans les politiques IAM pour restreindre davantage l'accès : vous pouvez ajouter une condition à vos politiques afin de limiter l'accès aux actions et aux ressources. Par exemple, vous pouvez écrire une condition de politique pour spécifier que toutes les demandes doivent être envoyées via SSL. Vous pouvez également utiliser des conditions pour accorder l'accès aux actions de service si elles sont utilisées par le biais d'un service spécifique Service AWS, tel que AWS CloudFormation. Pour plus d'informations, consultez Conditions pour éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.
- Utilisez l'Analyseur d'accès IAM pour valider vos politiques IAM afin de garantir des autorisations sécurisées et fonctionnelles : l'Analyseur d'accès IAM valide les politiques nouvelles et existantes de manière à ce que les politiques IAM respectent le langage de politique IAM (JSON) et les bonnes pratiques IAM. IAM Access Analyzer fournit plus de 100 vérifications de politiques et des recommandations exploitables pour vous aider à créer des politiques sécurisées et fonctionnelles. Pour plus d'informations, consultez Validation de politiques avec IAM Access Analyzer dans le Guide de l'utilisateur IAM.
- Exiger l'authentification multifactorielle (MFA) : si vous avez un scénario qui nécessite des utilisateurs IAM ou un utilisateur root, activez l'authentification MFA pour une sécurité accrue. Compte AWS Pour exiger la MFA lorsque des opérations d'API sont appelées, ajoutez des conditions MFA à vos politiques. Pour plus d'informations, consultez Sécurisation de l'accès aux API avec MFA dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur les bonnes pratiques dans IAM, consultez Bonnes pratiques de sécurité dans IAM dans le Guide de l'utilisateur IAM.

Utilisation d' OpenSearch Ingestion dans la console

Pour accéder à OpenSearch Ingestion depuis la console de OpenSearch service, vous devez disposer d'un ensemble minimal d'autorisations. Ces autorisations doivent vous permettre de répertorier et d'afficher les détails relatifs aux ressources OpenSearch d'ingestion de votre AWS compte. Si vous créez une stratégie basée sur l'identité qui est plus restrictive que les autorisations minimales requises, la console ne fonctionnera pas comme prévu pour les entités (telles que les rôles IAM) de cette stratégie.

Il n'est pas nécessaire d'accorder des autorisations de console minimales aux utilisateurs qui appellent uniquement l'API AWS CLI ou l' AWS API. Autorisez plutôt l'accès à uniquement aux actions qui correspondent à l'opération d'API que vous tentez d'effectuer.

La politique suivante permet à un utilisateur d'accéder à OpenSearch Ingestion depuis la console OpenSearch de service :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Resource": "*",
            "Effect": "Allow",
            "Action": [
                "osis:ListPipelines",
                "osis:GetPipeline",
                "osis:ListPipelineBlueprints",
                "osis:GetPipelineBlueprint",
                "osis:GetPipelineChangeProgress"
            ]
        }
    ]
}
```

Vous pouvez également utiliser la politique the section called “AmazonOpenSearchIngestionReadOnlyAccess” AWS gérée, qui accorde un accès en lecture seule à toutes les ressources d' OpenSearch ingestion pour un. Compte AWS

Administration des pipelines OpenSearch d'ingestion

Cette politique est un exemple de politique « administrateur de pipeline » qui permet à un utilisateur de gérer et d'administrer les pipelines Amazon OpenSearch Ingestion. L'utilisateur peut créer, afficher et supprimer des pipelines.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Resource": "arn:aws:osis:region:123456789012:pipeline/*",
            "Action": [
```

```
                "osis:CreatePipeline",
                "osis:DeletePipeline",
                "osis:UpdatePipeline",
                "osis:ValidatePipeline",
                "osis:StartPipeline",
                "osis:StopPipeline"
            ],
            "Effect": "Allow"
        },
        {
            "Resource": "*",
            "Action": [
                "osis:ListPipelines",
                "osis:GetPipeline",
                "osis:ListPipelineBlueprints",
                "osis:GetPipelineBlueprint",
                "osis:GetPipelineChangeProgress"
            ],
            "Effect": "Allow"
        }
    ]
}
```

Ingestion de données dans un pipeline d' OpenSearch ingestion

Cet exemple de politique permet à un utilisateur ou à une autre entité d'ingérer des données dans un pipeline Amazon OpenSearch Ingestion depuis son compte. L'utilisateur ne peut pas modifier les pipelines.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Resource": "arn:aws:osis:region:123456789012:pipeline/*",
            "Action": [
                "osis:Ingest"
            ],
            "Effect": "Allow"
        }
```

```
    ]
}
```

Journalisation des appels d' OpenSearch API Amazon Ingestion à l'aide de AWS CloudTrail

Amazon OpenSearch Ingestion est intégré à AWS CloudTrail, service qui enregistre les actions effectuées par un utilisateur, un rôle ou un AWS service dans OpenSearch Ingestion.

CloudTrail capture les appels d'API pour OpenSearch Ingestion en tant qu'événements. Les appels capturés incluent les appels de la section OpenSearch Ingestion de la console de OpenSearch service et les appels de code aux opérations de OpenSearch l'API d'ingestion.

Si vous créez un journal d'activité, vous pouvez activer la livraison continue d' CloudTrail événements à un compartiment Amazon S3, y compris des événements pour OpenSearch l'ingestion. Si vous ne configurez pas de journal de suivi, vous pouvez toujours afficher les événements les plus récents dans la CloudTrail console dans Event history (Historique des événements).

À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été envoyée à OpenSearch Ingestion, l'adresse IP à partir de laquelle la demande a été effectuée, l'auteur de la demande, la date de la demande, ainsi que d'autres informations.

Pour en savoir plus CloudTrail, consultez le guide de AWS CloudTrail l'utilisateur.

OpenSearch Informations sur l'ingestion dans CloudTrail

CloudTrail est activé sur votre compte au Compte AWS moment de la création de ce compte. Lorsqu'une activité a lieu dans OpenSearch Ingestion, cette activité est enregistrée dans un CloudTrail événement avec d'autres événements de AWS service dans Event history (Historique des événements). Vous pouvez afficher, rechercher et télécharger les événements récents dans votre Compte AWS. Pour de plus amples informations, veuillez consulter Affichage des événements avec CloudTrail Event history (Affichage des événements).

Pour un enregistrement continu des événements dans votre Compte AWS, y compris les événements d' OpenSearch Ingestion, créez un journal de suivi. Un journal de suivi CloudTrail permet de livrer des fichiers journaux dans un compartiment Amazon S3. Par défaut, lorsque vous créez un journal d’activité dans la console, il s’applique à toutes les régions Régions AWS.

Le journal de suivi consigne les événements de toutes les Régions dans la AWS partition et livre les fichiers journaux dans le compartiment Amazon S3 de votre choix. En outre, vous pouvez configurer d'autres AWS services pour analyser plus en profondeur les données d'événement collectées dans les CloudTrail journaux et agir sur celles-ci. Pour plus d'informations, consultez les ressources suivantes :

- Présentation de la création d'un journal de suivi
- CloudTrail services et intégrations pris en charge
- Configuration des notifications Amazon SNS pour CloudTrail
- Réception de fichiers CloudTrail journaux de plusieurs régions et Réception de fichiers CloudTrail journaux de plusieurs comptes

Toutes les actions OpenSearch d'ingestion sont enregistrées CloudTrail et documentées dans la référence de OpenSearch l'API d'ingestion. Par exemple, les appels adressés aux actions `CreateCollection` `ListCollections`, `DeleteCollection` génèrent des entrées dans les fichiers journaux CloudTrail .

Chaque événement ou entrée de journal contient des informations sur la personne ayant initié la demande. Les informations relatives à l'identité vous permettent de déterminer :

- Si la requête a été effectuée avec les informations d'identification utilisateur racine ou AWS Identity and Access Management (IAM).
- Si la demande a été effectuée avec les informations d'identification de sécurité temporaires d'un rôle ou d'un utilisateur fédéré.
- Si la requête a été effectuée par un autre AWS service.

Pour de plus amples informations, veuillez consulter l'élément userIdentity CloudTrail .

Présentation OpenSearch des entrées des fichiers journaux

Un journal d'activité est une configuration qui permet d'envoyer des événements sous forme de fichiers journaux à un compartiment Simple Storage Service (Amazon S3) que vous spécifiez. CloudTrail les fichiers journaux peuvent contenir une ou plusieurs entrées de journal.

Un événement représente une demande individuelle d'une source quelconque. Il comprend les informations sur l'action demandée, la date et l'heure de l'action, les paramètres de la demande,

etc. CloudTrail les fichiers journaux ne constituent pas une pile ordonnée retraçant les appels d'API publics. Ils ne suivent aucun ordre précis.

L'exemple suivant montre une entrée de CloudTrail journal qui illustre l'`DeletePipeline`action.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn":"arn:aws:iam::123456789012:user/test-user",
        "accountId": "123456789012",
        "accessKeyId": "access-key",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/Admin",
                "accountId": "123456789012",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-04-21T16:48:33Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-04-21T16:49:22Z",
    "eventSource": "osis.amazonaws.com",
    "eventName": "UpdatePipeline",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "123.456.789.012",
    "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/112.0.0.0 Safari/537.36",
    "requestParameters": {
        "pipelineName": "my-pipeline",
        "pipelineConfigurationBody": "version: \"2\"\nlog-pipeline:\n  source:\n
 http:\n        path: \"/test/logs\"\n  processor:\n    - grok:\n        match:\n
        log: [ '%{COMMONAPACHELOG}' ]\n    - date:\n        from_time_received: true
\n        destination: \"@timestamp\"\n  sink:\n    - opensearch:\n        hosts:
 [ \"https://search-b5zd22mwxhggheqpj5ftslgyle.us-west-2.es.amazonaws.com\" ]\n
  index: \"apache_logs2\"\n        aws_sts_role_arn: \"arn:aws:iam::709387180454:role/
```

```
canary-bootstrap-OsisRole-J1BARLD26QKN\"\n        aws_region: \"us-west-2\"\n
 aws_sigv4: true\n"
    },
    "responseElements": {
        "pipeline": {
            "pipelineName": "my-pipeline",sourceIPAddress
            "pipelineArn": "arn:aws:osis:us-west-2:123456789012:pipeline/my-pipeline",
            "minUnits": 1,
            "maxUnits": 1,
            "status": "UPDATING",
            "statusReason": {
                "description": "An update was triggered for the pipeline. It is still
 available to ingest data."
            },
            "pipelineConfigurationBody": "version: \"2\"\nlog-pipeline:\n  source:\n
    http:\n        path: \"/test/logs\"\n  processor:\n    - grok:\n        match:
\n          log: [ '%{COMMONAPACHELOG}' ]\n    - date:\n        from_time_received:
 true\n        destination: \"@timestamp\"\n  sink:\n    - opensearch:\n        hosts:
 [ \"https://search-b5zd22mwxhggheqpj5ftslgyle.us-west-2.es.amazonaws.com\" ]\n
  index: \"apache_logs2\"\n        aws_sts_role_arn: \"arn:aws:iam::709387180454:role/
canary-bootstrap-OsisRole-J1BARLD26QKN\"\n        aws_region: \"us-west-2\"\n
 aws_sigv4: true\n",
            "createdAt": "Mar 29, 2023 1:03:44 PM",
            "lastUpdatedAt": "Apr 21, 2023 9:49:21 AM",
            "ingestEndpointUrls": [
                "my-pipeline-tu33ldsgdltgv7x7tjqiudvf7m.us-west-2.osis.amazonaws.com"
            ]
        }
    },
    "requestID": "12345678-1234-1234-1234-987654321098",
    "eventID": "12345678-1234-1234-1234-987654321098",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "709387180454",
    "eventCategory": "Management",
    "tlsDetails": {
        "tlsVersion": "TLSv1.2",
        "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
        "clientProvidedHostHeader": "osis.us-west-2.amazonaws.com"
    },
    "sessionCredentialFromConsole": "true"
}
```

Marquage des pipelines Amazon OpenSearch Ingestion

Les balises vous permettent d'affecter des informations arbitraires à un pipeline OpenSearch Amazon Ingestion afin d'effectuer un classement et un filtrage sur ces informations. Une balise est une étiquette de métadonnées que vous attribuez ou qu' AWS attribue à une AWS ressource. Chaque balise se compose d’une clé et d’une valeur. Pour les balises que vous affectez, vous définissez la clé et la valeur. Par exemple, vous pouvez définir la clé sur `stage` et la valeur pour une ressource sur `test`.

Les balises vous permettent d’effectuer les actions suivantes :

- Identifiez et organisez vos AWS ressources. De nombreux AWS services prennent en charge le balisage. Vous pouvez donc attribuer la même balise à des ressources à partir de différents services pour indiquer que les ressources sont liées. Par exemple, vous pouvez affecter à un pipeline d' OpenSearch ingestion la même étiquette que celle que vous affectez à un domaine Amazon OpenSearch Service.
- Suivre vos AWS coûts. Vous activez ces balises sur le AWS Billing and Cost Management tableau de bord. AWS utilise les balises pour classer vos coûts par catégorie et vous fournir un rapport mensuel de répartition des coûts. Pour plus d’informations, consultez Utilisation des balises d’allocation des coûts dans le Guide de l’utilisateur AWS Billing.
- Limitez l'accès aux pipelines à l'aide d'un contrôle d'accès basé sur les attributs. Pour plus d'informations, veuillez consulter Contrôle de l'accès en fonction des clés de balises dans le Guide de l'utilisateur IAM.

Dans OpenSearch Ingestion, la ressource principale est un pipeline. Vous pouvez utiliser la console de OpenSearch service, la AWS CLI OpenSearch APIs, Ingestion ou le AWS SDKs pour ajouter, gérer et supprimer des balises d'un pipeline.

Rubriques

- Autorisations nécessaires
- Utilisation des balises (console)
- Utilisation des balises (AWS CLI)

Autorisations nécessaires

OpenSearch L'ingestion utilise les autorisations AWS Identity and Access Management Access Analyzer (IAM) suivantes pour le balisage des pipelines :

- `osis:TagResource`
- `osis:ListTagsForResource`
- `osis:UntagResource`

Pour plus d'informations sur chaque autorisation, consultez la section Actions, ressources et clés de condition pour OpenSearch l'ingestion dans la référence d'autorisation de service.

Utilisation des balises (console)

La console constitue le moyen le plus simple de baliser un pipeline.

Pour créer un tag

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, choisissez Pipelines.
3. Sélectionnez le pipeline auquel vous voulez ajouter des balises et accédez à l'onglet Tags (Balises).
4. Choisissez Manage (Gérer) et Add new tag (Ajouter une nouvelle balise).
5. Saisissez une clé de balise et une valeur de balise facultative.
6. Choisissez Save (Enregistrer).

Pour supprimer une balise, suivez les mêmes étapes et choisissez Remove (Supprimer) sur la page Manage tags (Gérer les balises).

Pour plus d'informations sur l'utilisation de la console avec des identifications, veuillez consulter Éditeur d'identification dans le Guide de démarrage de la console de gestion AWS .

Utilisation des balises (AWS CLI)

Pour baliser un pipeline à l'aide du AWS CLI, envoyez une `TagResource` demande :

```
aws osis tag-resource
  --arn arn:aws:osis:us-east-1:123456789012:pipeline/my-pipeline
  --tags Key=service,Value=osis Key=source,Value=otel
```

Supprimez les balises d'un pipeline à l'aide de la `UntagResource` commande :

```
aws osis untag-resource
  --arn arn:aws:osis:us-east-1:123456789012:pipeline/my-pipeline
  --tag-keys service
```

Affichez les balises existantes pour un pipeline avec la `ListTagsForResource` commande :

```
aws osis list-tags-for-resource
  --arn arn:aws:osis:us-east-1:123456789012:pipeline/my-pipeline
```

Journalisation et surveillance de OpenSearch l'ingestion d'Amazon avec Amazon CloudWatch

Amazon OpenSearch Ingestion publie des statistiques et des journaux sur Amazon CloudWatch.

Rubriques

- Surveillance des journaux du pipeline
- Surveillance des métriques du pipeline

Surveillance des journaux du pipeline

Vous pouvez activer la journalisation des pipelines Amazon OpenSearch Ingestion afin d'exposer les messages d'erreur et d'avertissement émis lors des opérations du pipeline et des activités d'ingestion. OpenSearch Ingestion publie tous les journaux sur Amazon CloudWatch Logs. CloudWatch Les journaux peuvent surveiller les informations contenues dans les fichiers journaux et vous avertir lorsque certains seuils sont atteints. Vous pouvez également archiver vos données de journaux dans une solution de stockage hautement durable. Pour de plus amples informations, veuillez consulter le Guide de l'utilisateur Amazon CloudWatch Logs.

Les journaux issus de OpenSearch l'ingestion peuvent indiquer un échec du traitement des demandes, des erreurs d'authentification entre la source et le récepteur et d'autres avertissements pouvant être utiles pour le dépannage. Pour ses journaux, OpenSearch Ingestion utilise les niveaux

de journalisation de `INFO`,`WARN`,`ERROR`, et`FATAL`. Nous recommandons d'activer la publication des journaux pour tous les pipelines.

Autorisations nécessaires

Pour permettre à OpenSearch Ingestion d'envoyer les journaux vers CloudWatch les journaux, vous devez être connecté en tant qu'utilisateur disposant de certaines autorisations IAM.

Vous devez disposer des autorisations de CloudWatch journalisation suivantes pour créer et mettre à jour les ressources de diffusion des journaux :

```
{a
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Resource": "*",
            "Action": [
                "logs:CreateLogDelivery",
                "logs:PutResourcePolicy",
                "logs:UpdateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:DescribeResourcePolicies",
                "logs:GetLogDelivery",
                "logs:ListLogDeliveries"
            ]
        }
    ]
}
```

Activation de la publication des journaux

Vous pouvez activer la publication de journaux sur des pipelines existants ou lors de la création d'un pipeline. Pour connaître les étapes à suivre pour activer la publication des journaux lors de la création du pipeline, consultezthe section called “Création de pipelines”.

console

Pour activer la publication des journaux sur un pipeline existant

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.

2. Dans le volet de navigation de gauche, choisissez Pipelines.
3. Ouvrez le pipeline dans lequel vous souhaitez activer les journaux, puis choisissez Actions, Modifier les options de publication des journaux.
4. Activez Publier dans CloudWatch les journaux.
5. Créez un groupe de journaux ou sélectionnez un groupe existant. Nous vous recommandons de formater le nom sous la forme d'un chemin, par exemple`/aws/vendedlogs/OpenSearchIngestion/pipeline-name/audit-logs`. Ce format facilite l'application d'une politique d' CloudWatch accès qui accorde des autorisations à tous les groupes de journaux sous un chemin spécifique tel que`/aws/vendedlogs/OpenSearchIngestion`.

   > **Important**
   >
   > Vous devez inclure le préfixe `vendedlogs` dans le nom du groupe de journaux, sinon la création échoue.

6. Choisissez Enregistrer.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour activer la publication des journaux à l'aide de AWS CLI, envoyez la requête suivante :

```
aws osis update-pipeline \
  --pipeline-name my-pipeline \
  --log-publishing-options  IsLoggingEnabled=true,CloudWatchLogDestination={LogGroup="/
aws/vendedlogs/OpenSearchIngestion/pipeline-name"}
```

Surveillance des métriques du pipeline

Vous pouvez surveiller les pipelines Amazon OpenSearch Ingestion à l'aide d'Amazon CloudWatch, qui collecte les données brutes et les transforme en métriques lisibles et disponibles en temps quasi réel. Ces statistiques sont enregistrées pour une durée de 15 mois ; par conséquent, vous pouvez accéder aux informations historiques et acquérir un meilleur point de vue de la façon dont votre service ou application web s'exécute. Vous pouvez également définir des alarmes qui surveillent certains seuils et envoient des notifications ou prennent des mesures lorsque ces seuils sont atteints. Pour de plus amples informations, veuillez consulter le Guide de CloudWatch l'utilisateur Amazon.

La OpenSearch console Ingestion affiche un ensemble de graphiques basés sur les données brutes envoyées par CloudWatch l'onglet Performances pour chaque pipeline.

OpenSearch Ingestion rapporte les statistiques de la plupart des plugins pris en charge. Si certains plugins n'ont pas leur propre tableau ci-dessous, cela signifie qu'ils ne signalent aucune métrique spécifique au plugin. Les métriques du pipeline sont publiées dans l'espace de `AWS/OSIS` noms.

Rubriques

- Métriques générales
- Métriques de mémoire tampon
- Métriques Signature V4
- Métriques de la mémoire tampon de blocage bornée
- Indicateurs relatifs aux sources de traçabilité des hôtels
- Métriques hôtelières, mesures sources
- Métriques HTTP
- Métriques S3
- Métriques Aggregate
- Métriques de date
- métriques Lambda
- Métriques Grok
- Métriques brutes de l'hôtel
- Métriques du groupe Hotel Trace
- Métriques dynamiques de la Cartographie des services
- OpenSearch métriques
- Métriques du système et de mesure

Métriques générales

Les mesures suivantes sont communes à tous les processeurs et récepteurs.

Chaque métrique est préfixée par le nom du sous-pipeline et le nom du plugin, au format **<** *`sub_pipeline_name`* **><** **><** **>***`plugin`*. *`metric_name`* Par exemple, le nom complet de la `recordsIn.count` métrique pour un sous-pipeline nommé `my-pipeline` et le processeur de date seraient`my-pipeline.date.recordsIn.count`.

Suffixe métrique	Description
`recordsIn.count`	Entrée d'enregistrements dans un composant du pipeline. Cette métrique s'applique aux processeurs et aux récepteurs. Statistiques pertinentes : somme Dimension : `PipelineName`
`recordsOut.count`	La sortie des enregistrements d'un composant du pipeline. Cette métrique s'applique aux processeurs et aux sources. Statistiques pertinentes : somme Dimension : `PipelineName`
`timeElapsed.count`	Nombre de points de données enregistrés lors de l'exécuti on d'un composant de pipeline. Cette métrique s'applique aux processeurs et aux récepteurs. Statistiques pertinentes : somme Dimension : `PipelineName`
`timeElapsed.sum`	Durée totale écoulée pendant l'exécution d'un composant de pipeline. Cette métrique s'applique aux processeurs et aux récepteurs, en millisecondes. Statistiques pertinentes : somme Dimension : `PipelineName`
`timeElapsed.max`	Durée maximale écoulée lors de l'exécution d'un composant de pipeline. Cette métrique s'applique aux processeurs et aux récepteurs, en millisecondes. Statistiques pertinentes : maximum Dimension : `PipelineName`

Métriques de mémoire tampon

Les mesures suivantes s'appliquent à la mémoire tampon de blocage bornée par défaut qu' OpenSearch Ingestion configure automatiquement pour tous les pipelines.

Chaque métrique est préfixée par le nom du sous-pipeline et le nom du tampon, au format <*sub_pipeline_name*>.<*buffer_name*>.*metric_name* Par exemple, le nom complet de la `recordsWritten.count` métrique d'un sous-pipeline nommé `my-pipeline` seraitmy-pipeline.BlockingBuffer.recordsWritten.count.

Suffixe métrique	Description
`recordsWritten.count`	Nombre d'enregistrements écrits dans une mémoire tampon. Statistiques pertinentes : somme Dimension : `PipelineName`
`recordsRead.count`	Nombre d'enregistrements lus à partir d'une mémoire tampon. Statistiques pertinentes : somme Dimension : `PipelineName`
`recordsInFlight.value`	Le nombre d'enregistrements non vérifiés lus dans une mémoire tampon. Statistiques pertinentes : moyenne Dimension : `PipelineName`
`recordsInBuffer.value`	Nombre d'enregistrements actuellement dans une mémoire tampon. Statistiques pertinentes : moyenne Dimension : `PipelineName`

Suffixe métrique	Description
`recordsProcessed.count`	Nombre d'enregistrements lus dans une mémoire tampon et traités par un pipeline. Statistiques pertinentes : somme Dimension : `PipelineName`
`recordsWriteFailed.count`	Nombre d'enregistrements que le pipeline n'a pas pu écrire sur le récepteur. Statistiques pertinentes : somme Dimension : `PipelineName`
`writeTimeElapsed.count`	Nombre de points de données enregistrés lors de l'écriture dans une mémoire tampon. Statistiques pertinentes : somme Dimension : `PipelineName`
`writeTimeElapsed.sum`	Durée totale écoulée lors de l'écriture dans une mémoire tampon, en millisecondes. Statistiques pertinentes : somme Dimension : `PipelineName`
`writeTimeElapsed.max`	Durée maximale écoulée lors de l'écriture dans une mémoire tampon, en millisecondes. Statistiques pertinentes : maximum Dimension : `PipelineName`
`writeTimeouts.count`	Le nombre de délais d'écriture dans une mémoire tampon. Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`readTimeElapsed.count`	Nombre de points de données enregistrés lors de la lecture depuis une mémoire tampon. Statistiques pertinentes : somme Dimension : `PipelineName`
`readTimeElapsed.sum`	Durée totale écoulée lors de la lecture depuis une mémoire tampon, en millisecondes. Statistiques pertinentes : somme Dimension : `PipelineName`
`readTimeElapsed.max`	Durée maximale écoulée lors de la lecture depuis une mémoire tampon, en millisecondes. Statistiques pertinentes : maximum Dimension : `PipelineName`
`checkpointTimeElapsed.count`	Nombre de points de données enregistrés lors du point de contrôle. Statistiques pertinentes : somme Dimension : `PipelineName`
`checkpointTimeElapsed.sum`	Temps total écoulé pendant le point de contrôle, en millisecondes. Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`checkpointTimeElapsed.max`	Durée maximale écoulée pendant le point de contrôle, en millisecondes. Statistiques pertinentes : maximum Dimension : `PipelineName`

Métriques Signature V4

Les métriques suivantes s'appliquent au point de terminaison d'ingestion d'un pipeline et sont associées aux plugins source (`http`,`otel_trace`, et`otel_metrics`). Toutes les demandes adressées au point de terminaison d'ingestion doivent être signées à l'aide de la version 4 de Signature. Ces indicateurs peuvent vous aider à identifier les problèmes d'autorisation lors de la connexion à votre pipeline ou à confirmer que vous vous authentifiez correctement.

Chaque métrique est préfixée par le nom du sous-pipeline et. `osis_sigv4_auth` Par exemple, *`sub_pipeline_name`*`.osis_sigv4_auth.httpAuthSuccess.count`.

Suffixe métrique	Description
`httpAuthSuccess.count`	Le nombre de demandes Signature V4 envoyées au pipeline avec succès. Statistiques pertinentes : somme Dimension : `PipelineName`
`httpAuthFailure.count`	Le nombre de demandes Signature V4 envoyées au pipeline qui ont échoué. Statistiques pertinentes : somme Dimension : `PipelineName`
`httpAuthServerError.count`	Le nombre de demandes Signature V4 adressées au pipeline qui ont renvoyé des erreurs de serveur. Statistiques pertinentes : somme

Suffixe métrique	Description
	Dimension : `PipelineName`

Métriques de la mémoire tampon de blocage bornée

Les mesures suivantes s'appliquent à la zone tampon de blocage limitée. Chaque métrique est préfixée par le nom du sous-pipeline et. `BlockingBuffer` Par exemple, *`sub_pipeline_name`*`.BlockingBuffer.bufferUsage.value`.

Suffixe métrique	Description
`bufferUsage.value`	Pourcentage d'utilisation du `buffer_size` en fonction du nombre d'enregistrements dans la mémoire tampon. `buffer_size` représente le nombre maximum d'enregistrements enregistrés dans la mémoire tampon ainsi que les enregistrements en vol qui n'ont pas été vérifiés. Statistiques pertinentes : moyenne Dimension : `PipelineName`

Indicateurs relatifs aux sources de traçabilité des hôtels

Les mesures suivantes s'appliquent à la source de OTel suivi. Chaque métrique est préfixée par le nom du sous-pipeline et. `otel_trace_source` Par exemple, *`sub_pipeline_name`*`.otel_trace_source.requestTimeouts.count`.

Suffixe métrique	Description
`requestTimeouts.count`	Nombre de demandes qui ont expiré. Statistiques pertinentes : somme Dimension : `PipelineName`
`requestsReceived.count`	Nombre de requêtes reçues par le plugin.

Suffixe métrique	Description
	Statistiques pertinentes : somme Dimension : `PipelineName`
`successRequests.count`	Nombre de requêtes qui ont été traitées avec succès par le plugin. Statistiques pertinentes : somme Dimension : `PipelineName`
`badRequests.count`	Le nombre de demandes dont le format n'est pas valide et qui ont été traitées par le plugin. Statistiques pertinentes : somme Dimension : `PipelineName`
`requestsTooLarge.count`	Le nombre de demandes dont le nombre de plages du contenu est supérieur à la capacité de la mémoire tampon. Statistiques pertinentes : somme Dimension : `PipelineName`
`internalServerError.count`	Le nombre de demandes traitées par le plugin avec un type d'exception personnalisé. Statistiques pertinentes : somme Dimension : `PipelineName`
`requestProcessDuration.count`	Nombre de points de données enregistrés lors du traitement des demandes par le plugin. Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`requestProcessDuration.sum`	Latence totale des demandes traitées par le plugin, en millisecondes. Statistiques pertinentes : somme Dimension : `PipelineName`
`requestProcessDuration.max`	Latence maximale des demandes traitées par le plugin, en millisecondes. Statistiques pertinentes : maximum Dimension : `PipelineName`
`payloadSize.count`	Nombre de la distribution des tailles de charge utile des demandes entrantes, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`
`payloadSize.sum`	Distribution totale des tailles de charge utile des demandes entrantes, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`
`payloadSize.max`	Distribution maximale des tailles de charge utile des demandes entrantes, en octets. Statistiques pertinentes : maximum Dimension : `PipelineName`

Métriques hôtelières, mesures sources

Les mesures suivantes s'appliquent à la source OTel des mesures. Chaque métrique est préfixée par le nom du sous-pipeline et. `otel_metrics_source` Par exemple, *`sub_pipeline_name`*`.otel_metrics_source.requestTimeouts.count`.

Suffixe métrique	Description
`requestTimeouts.count`	Nombre total de requêtes adressées au plugin qui ont expiré. Statistiques pertinentes : somme Dimension : `PipelineName`
`requestsReceived.count`	Nombre total de requêtes reçues par le plugin. Statistiques pertinentes : somme Dimension : `PipelineName`
`successRequests.count`	Le nombre de demandes traitées avec succès (code d'état de 200 réponses) par le plugin. Statistiques pertinentes : somme Dimension : `PipelineName`
`requestProcessDuration.count`	Un décompte de la latence des demandes traitées par le plugin, en secondes. Statistiques pertinentes : somme Dimension : `PipelineName`
`requestProcessDuration.sum`	Latence totale des demandes traitées par le plugin, en millisecondes. Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`requestProcessDuration.max`	Latence maximale des demandes traitées par le plugin, en millisecondes. Statistiques pertinentes : maximum Dimension : `PipelineName`
`payloadSize.count`	Nombre de la distribution des tailles de charge utile des demandes entrantes, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`
`payloadSize.sum`	Distribution totale des tailles de charge utile des demandes entrantes, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`
`payloadSize.max`	Distribution maximale des tailles de charge utile des demandes entrantes, en octets. Statistiques pertinentes : maximum Dimension : `PipelineName`

Métriques HTTP

Les mesures suivantes s'appliquent à la source HTTP. Chaque métrique est préfixée par le nom du sous-pipeline et. `http` Par exemple, *`sub_pipeline_name`*`.http.requestsReceived.count`.

Suffixe métrique	Description
`requestsReceived.count`	Nombre de requêtes reçues par le `/log/ingest` point de terminaison.

Suffixe métrique	Description
	Statistiques pertinentes : somme Dimension : `PipelineName`
`requestsRejected.count`	Le nombre de demandes rejetées (code d'état de réponse 429) par le plugin. Statistiques pertinentes : somme Dimension : `PipelineName`
`successRequests.count`	Le nombre de demandes traitées avec succès (code d'état de 200 réponses) par le plugin. Statistiques pertinentes : somme Dimension : `PipelineName`
`badRequests.count`	Le nombre de demandes dont le type ou le format de contenu n'est pas valide (code d'état de réponse 400) traitées par le plugin. Statistiques pertinentes : somme Dimension : `PipelineName`
`requestTimeouts.count`	Nombre de demandes dont le délai d'expiration est dépassé sur le serveur source HTTP (code d'état de réponse 415). Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`requestsTooLarge.count`	Nombre de demandes dont la taille des événements dans le contenu est supérieure à la capacité de la mémoire tampon (code d'état de réponse 413). Statistiques pertinentes : somme Dimension : `PipelineName`
`internalServerError.count`	Le nombre de demandes traitées par le plugin avec un type d'exception personnalisé (code d'état de réponse de 500). Statistiques pertinentes : somme Dimension : `PipelineName`
`requestProcessDuration.count`	Un décompte de la latence des demandes traitées par le plugin, en secondes. Statistiques pertinentes : somme Dimension : `PipelineName`
`requestProcessDuration.sum`	Latence totale des demandes traitées par le plugin, en millisecondes. Statistiques pertinentes : somme Dimension : `PipelineName`
`requestProcessDuration.max`	Latence maximale des demandes traitées par le plugin, en millisecondes. Statistiques pertinentes : maximum Dimension : `PipelineName`

Suffixe métrique	Description
`payloadSize.count`	Nombre de la distribution des tailles de charge utile des demandes entrantes, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`
`payloadSize.sum`	Distribution totale des tailles de charge utile des demandes entrantes, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`
`payloadSize.max`	Distribution maximale des tailles de charge utile des demandes entrantes, en octets. Statistiques pertinentes : maximum Dimension : `PipelineName`

Métriques S3

Les métriques suivantes s'appliquent à la source S3. Chaque métrique est préfixée par le nom du sous-pipeline et. `s3` Par exemple, *`sub_pipeline_name`*`.s3.s3ObjectsFailed.count`.

Suffixe métrique	Description
`s3ObjectsFailed.count`	Nombre total d'objets S3 que le plugin n'a pas pu lire. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3ObjectsNotFound.count`	Le nombre d'objets S3 que le plugin n'a pas pu lire en raison d'une `Not Found` erreur de S3. Ces mesures sont également prises en compte dans le calcul de la `s3ObjectsFailed` métrique.

Suffixe métrique	Description
	Statistiques pertinentes : somme Dimension : `PipelineName`
`s3ObjectsAccessDenied.count`	Le nombre d'objets S3 que le plugin n'a pas pu lire en raison `Access Denied` d'une `Forbidden` erreur de S3. Ces mesures sont également prises en compte dans le calcul de la `s3ObjectsFailed` métrique. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3ObjectReadTimeElapsed.count`	Le temps nécessaire au plugin pour exécuter une requête GET pour un objet S3, l'analyser et écrire des événements dans la mémoire tampon. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3ObjectReadTimeElapsed.sum`	Temps total nécessaire au plugin pour exécuter une requête GET pour un objet S3, l'analyser et écrire des événements dans la mémoire tampon, en millisecondes. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3ObjectReadTimeElapsed.max`	Durée maximale nécessaire au plugin pour exécuter une requête GET pour un objet S3, l'analyser et écrire des événements dans la mémoire tampon, en millisecondes. Statistiques pertinentes : maximum Dimension : `PipelineName`

Suffixe métrique	Description
`s3ObjectSizeBytes.count`	Le nombre de distributions des tailles d'objets S3, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3ObjectSizeBytes.sum`	Distribution totale de la taille des objets S3, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3ObjectSizeBytes.max`	Distribution maximale de la taille des objets S3, en octets. Statistiques pertinentes : maximum Dimension : `PipelineName`
`s3ObjectProcessedBytes.count`	Le nombre de distributions d'objets S3 traités par le plugin, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3ObjectProcessedBytes.sum`	La distribution totale des objets S3 traités par le plugin, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3ObjectProcessedBytes.max`	Distribution maximale des objets S3 traités par le plugin, en octets. Statistiques pertinentes : maximum Dimension : `PipelineName`

Suffixe métrique	Description
`s3ObjectsEvents.count`	Le nombre de distributions d'événements S3 reçus par le plugin. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3ObjectsEvents.sum`	La distribution totale des événements S3 reçus par le plugin. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3ObjectsEvents.max`	La distribution maximale des événements S3 reçus par le plugin. Statistiques pertinentes : maximum Dimension : `PipelineName`
`sqsMessageDelay.count`	Nombre de points de données enregistrés pendant que S3 enregistre l'heure d'un événement entre la création d'un objet et le moment où il est complètement analysé. Statistiques pertinentes : somme Dimension : `PipelineName`
`sqsMessageDelay.sum`	Durée totale entre le moment où S3 enregistre l'heure d'un événement pour la création d'un objet et le moment où celui-ci est complètement analysé, en millisecondes. Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`sqsMessageDelay.max`	Durée maximale entre le moment où S3 enregistre l'heure d'un événement pour la création d'un objet et le moment où celui-ci est complètement analysé, en millisecondes. Statistiques pertinentes : maximum Dimension : `PipelineName`
`s3ObjectsSucceeded.count`	Le nombre d'objets S3 que le plugin a réussi à lire. Statistiques pertinentes : somme Dimension : `PipelineName`
`sqsMessagesReceived.count`	Le nombre de messages Amazon SQS reçus de la file d'attente par le plugin. Statistiques pertinentes : somme Dimension : `PipelineName`
`sqsMessagesDeleted.count`	Nombre de messages Amazon SQS supprimés de la file d'attente par le plugin. Statistiques pertinentes : somme Dimension : `PipelineName`
`sqsMessagesFailed.count`	Le nombre de messages Amazon SQS que le plugin n'a pas réussi à analyser. Statistiques pertinentes : somme Dimension : `PipelineName`

Métriques Aggregate

Les mesures suivantes s'appliquent au processeur Aggregate. Chaque métrique est préfixée par le nom du sous-pipeline et. `aggregate` Par exemple, *`sub_pipeline_name`*`.aggregate.actionHandleEventsOut.count`.

Suffixe métrique	Description
`actionHandleEvents Out.count`	Le nombre d'événements renvoyés par l'`handleEve nt` appel à l'action configurée. Statistiques pertinentes : somme Dimension : `PipelineName`
`actionHandleEvents Dropped.count`	Le nombre d'événements renvoyés par l'`handleEve nt` appel à l'action configurée. Statistiques pertinentes : somme Dimension : `PipelineName`
`actionHandleEvents ProcessingErrors.count`	Le nombre d'appels effectués `handleEvent` pour l'action configurée qui a entraîné une erreur. Statistiques pertinentes : somme Dimension : `PipelineName`
`actionConcludeGrou pEventsOut.count`	Le nombre d'événements renvoyés par l'`concludeG roup` appel à l'action configurée. Statistiques pertinentes : somme Dimension : `PipelineName`
`actionConcludeGrou pEventsDropped.count`	Le nombre d'événements qui n'ont pas été renvoyés depuis l'`condludeGroup` appel à l'action configurée. Statistiques pertinentes : somme

Suffixe métrique	Description
	Dimension : `PipelineName`
`actionConcludeGroupEventsProcessingErrors.count`	Le nombre d'appels effectués `concludeGroup` pour l'action configurée qui a entraîné une erreur. Statistiques pertinentes : somme Dimension : `PipelineName`
`currentAggregateGroups.value`	Nombre actuel de groupes. Cette jauge diminue lorsque les groupes sont terminés et augmente lorsqu'un événement déclenche la création d'un nouveau groupe. Statistiques pertinentes : moyenne Dimension : `PipelineName`

Métriques de date

Les mesures suivantes s'appliquent au processeur de données. Chaque métrique est préfixée par le nom du sous-pipeline et. `date` Par exemple, *`sub_pipeline_name`*`.date.dateProcessingMatchSuccess.count`.

Suffixe métrique	Description
`dateProcessingMatchSuccess.count`	Le nombre d'enregistrements correspondant à au moins un des modèles spécifiés dans l'option `match` de configuration. Statistiques pertinentes : somme Dimension : `PipelineName`
`dateProcessingMatchFailure.count`	Nombre d'enregistrements qui ne correspondent à aucun des modèles spécifiés dans l'option `match` de configuration. Statistiques pertinentes : somme

Suffixe métrique	Description
	Dimension : `PipelineName`

métriques Lambda

Les mesures suivantes s'appliquent au AWS Lambdaprocesseur. Chaque métrique est préfixée par le nom du sous-pipeline et. `lambda` Par exemple, *`sub_pipeline_name`*`.lambda.recordsSuccessfullySentToLambda.count`.

Suffixe métrique	Description
`recordsSuccessfullySentToLambda.count`	Nombre d'enregistrements traités avec succès par la fonction Lambda. Statistiques pertinentes : somme Dimension : `PipelineName`
`recordsFailedToSendToLambda.count`	Nombre d'enregistrements qui n'ont pas pu être envoyés à la fonction Lambda. Statistiques pertinentes : somme Dimension : `PipelineName`

`lambdaFunctionLatency.avg`
 	
`lambdaFunctionLatency.max`
	La latence des appels de fonctions Lambda.
	
	Statistiques pertinentes : moyenne et maximale
	
	Dimension : `PipelineName`

`numberOfRequestsSucceeded.count`	Nombre total de demandes d'appel Lambda réussies. Statistiques pertinentes : somme Dimension : `PipelineName`
`numberOfRequestsFailed.count`	Nombre total de demandes d'appel Lambda ayant échoué. Statistiques pertinentes : somme

Suffixe métrique	Description
	Dimension : `PipelineName`
`requestPayloadSize.avg`	Taille des charges utiles de requêtes envoyées à Lambda. Statistiques pertinentes : moyenne Dimension : `PipelineName`
`responsePayloadSize.avg`	Taille des charges utiles de réponse reçues de Lambda. Statistiques pertinentes : moyenne Dimension : `PipelineName`

Métriques Grok

Les mesures suivantes s'appliquent au processeur Grok. Chaque métrique est préfixée par le nom du sous-pipeline et. `grok` Par exemple, *`sub_pipeline_name`*`.grok.grokProcessingMatch.count`.

Suffixe métrique	Description
`grokProcessingMatch.count`	Le nombre d'enregistrements ayant trouvé au moins un modèle correspondant à l'option de `match` configuration. Statistiques pertinentes : somme Dimension : `PipelineName`
`grokProcessingMismatch.count`	Nombre d'enregistrements qui ne correspondent à aucun des modèles spécifiés dans l'option `match` de configuration. Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`grokProcessingErrors.count`	Nombre d'erreurs de traitement d'enregistrements. Statistiques pertinentes : somme Dimension : `PipelineName`
`grokProcessingTimeouts.count`	Nombre d'enregistrements qui ont expiré lors de la mise en correspondance. Statistiques pertinentes : somme Dimension : `PipelineName`
`grokProcessingTime.count`	Nombre de points de données enregistrés alors qu'un enregistrement individuel correspondait aux modèles définis dans l'option de `match` configuration. Statistiques pertinentes : somme Dimension : `PipelineName`
`grokProcessingTime.sum`	Temps total nécessaire à chaque enregistrement individuel pour correspondre aux modèles de l'option de `match` configuration, en millisecondes. Statistiques pertinentes : somme Dimension : `PipelineName`
`grokProcessingTime.max`	Durée maximale nécessaire à chaque enregistrement individuel pour correspondre aux modèles de l'option de `match` configuration, en millisecondes. Statistiques pertinentes : maximum Dimension : `PipelineName`

Métriques brutes de l'hôtel

Les mesures suivantes s'appliquent au processeur OTel Trace Raw. Chaque métrique est préfixée par le nom du sous-pipeline et. `otel_trace_raw` Par exemple, *`sub_pipeline_name`*`.otel_trace_raw.traceGroupCacheCount.value`.

Suffixe métrique	Description
`traceGroupCacheCount.value`	Le nombre de groupes de traces dans le cache des groupes de traces. Statistiques pertinentes : somme Dimension : `PipelineName`
`spanSetCount.value`	Le nombre d'ensembles d'intervalles contenus dans la collection d'ensembles d'intervalles. Statistiques pertinentes : somme Dimension : `PipelineName`

Métriques du groupe Hotel Trace

Les mesures suivantes s'appliquent au processeur de groupe de OTel traçage. Chaque métrique est préfixée par le nom du sous-pipeline et. `otel_trace_group` Par exemple, *`sub_pipeline_name`*`.otel_trace_group.recordsInMissingTraceGroup.count`.

Suffixe métrique	Description
`recordsInMissingTraceGroup.count`	Nombre d'enregistrements d'entrée où il manque des champs de groupe de traces. Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`recordsOutFixedTraceGroup.count`	Le nombre d'enregistrements de sortie contenant des champs de groupe de traces qui ont été remplis avec succès. Statistiques pertinentes : somme Dimension : `PipelineName`
`recordsOutMissingTraceGroup.count`	Le nombre d'enregistrements de sortie dans lesquels il manque des champs de groupe de traces. Statistiques pertinentes : somme Dimension : `PipelineName`

Métriques dynamiques de la Cartographie des services

Les métriques suivantes s'appliquent au processeur dynamique Service-Map. Chaque métrique est préfixée par le nom du sous-pipeline et. `service-map-stateful` Par exemple, *`sub_pipeline_name`*`.service-map-stateful.spansDbSize.count`.

Suffixe métrique	Description
`spansDbSize.value`	La taille des octets en mémoire des intervalles dans MapDB sur les durées de fenêtre actuelles et précédentes. Statistiques pertinentes : moyenne Dimension : `PipelineName`
`traceGroupDbSize.value`	Les tailles d'octets en mémoire des groupes de traces dans MapDB pendant les durées de fenêtre actuelles et précédentes. Statistiques pertinentes : moyenne Dimension : `PipelineName`

Suffixe métrique	Description
`spansDbCount.value`	Le nombre de plages dans MapDB pendant la durée de la fenêtre actuelle et précédente. Statistiques pertinentes : somme Dimension : `PipelineName`
`traceGroupDbCount.value`	Le nombre de groupes de traces dans MapDB pendant la durée de la fenêtre actuelle et précédente. Statistiques pertinentes : somme Dimension : `PipelineName`
`relationshipCount.value`	Nombre de relations stockées pendant la durée de la fenêtre actuelle et précédente. Statistiques pertinentes : somme Dimension : `PipelineName`

OpenSearch métriques

Les mesures suivantes s'appliquent au OpenSearchrécepteur. Chaque métrique est préfixée par le nom du sous-pipeline et. `opensearch` Par exemple, *`sub_pipeline_name`*`.opensearch.bulkRequestErrors.count`.

Suffixe métrique	Description
`bulkRequestErrors.count`	Nombre total d'erreurs rencontrées lors de l'envoi de demandes groupées. Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`documentsSuccess.count`	Le nombre de documents envoyés avec succès au OpenSearch Service par demande groupée, y compris les nouvelles tentatives. Statistiques pertinentes : somme Dimension : `PipelineName`
`documentsSuccessFirstAttempt.count`	Le nombre de documents envoyés avec succès au OpenSearch service par demande groupée lors de la première tentative. Statistiques pertinentes : somme Dimension : `PipelineName`
`documentErrors.count`	Le nombre de documents qui n'ont pas pu être envoyés par le biais de demandes groupées. Statistiques pertinentes : somme Dimension : `PipelineName`
`bulkRequestFailed.count`	Nombre de demandes groupées qui ont échoué. Statistiques pertinentes : somme Dimension : `PipelineName`
`bulkRequestNumberOfRetries.count`	Le nombre de tentatives de demandes groupées ayant échoué. Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`bulkBadRequestErrors.count`	Nombre d'`Bad Request`erreurs rencontrées lors de l'envoi de demandes groupées. Statistiques pertinentes : somme Dimension : `PipelineName`
`bulkRequestNotAllowedErrors.count`	Nombre d'`Request Not Allowed`erreurs rencontrées lors de l'envoi de demandes groupées. Statistiques pertinentes : somme Dimension : `PipelineName`
`bulkRequestInvalidInputErrors.count`	Nombre d'`Invalid Input`erreurs rencontrées lors de l'envoi de demandes groupées. Statistiques pertinentes : somme Dimension : `PipelineName`
`bulkRequestNotFoundErrors.count`	Nombre d'`Request Not Found`erreurs rencontrées lors de l'envoi de demandes groupées. Statistiques pertinentes : somme Dimension : `PipelineName`
`bulkRequestTimeoutErrors.count`	Nombre d'`Request Timeout`erreurs rencontrées lors de l'envoi de demandes groupées. Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`bulkRequestServerErrors.count`	Nombre d'`Server Error`erreurs rencontrées lors de l'envoi de demandes groupées. Statistiques pertinentes : somme Dimension : `PipelineName`
`bulkRequestSizeBytes.count`	Nombre de la distribution des tailles de charge utile des demandes groupées, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`
`bulkRequestSizeBytes.sum`	Distribution totale des tailles de charge utile des demandes groupées, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`
`bulkRequestSizeBytes.max`	Distribution maximale de la taille de charge utile des demandes groupées, en octets. Statistiques pertinentes : maximum Dimension : `PipelineName`
`bulkRequestLatency.count`	Nombre de points de données enregistrés lors de l'envoi des demandes au plugin, y compris les nouvelles tentatives. Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`bulkRequestLatency.sum`	Latence totale des demandes envoyées au plugin, y compris les nouvelles tentatives, en millisecondes. Statistiques pertinentes : somme Dimension : `PipelineName`
`bulkRequestLatency.max`	Latence maximale des demandes envoyées au plugin, y compris les nouvelles tentatives, en millisecondes. Statistiques pertinentes : maximum Dimension : `PipelineName`
`s3.dlqS3RecordsSuccess.count`	Le nombre d'enregistrements envoyés avec succès à la file d'attente des lettres mortes S3. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3.dlqS3RecordsFailed.count`	Le nombre de recours qui n'ont pas été envoyés à la file d'attente des lettres mortes S3. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3.dlqS3RequestSuccess.count`	Le nombre de demandes envoyées avec succès à la file d'attente des lettres mortes S3. Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`s3.dlqS3RequestFailed.count`	Le nombre de demandes ayant échoué dans la file d'attente des lettres mortes S3. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3.dlqS3RequestLatency.count`	Nombre de points de données enregistrés lorsque les demandes sont envoyées à la file d'attente des lettres mortes S3, y compris les nouvelles tentatives. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3.dlqS3RequestLatency.sum`	Latence totale des demandes envoyées à la file d'attente des lettres mortes S3, y compris les nouvelles tentatives, en millisecondes. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3.dlqS3RequestLatency.max`	Latence maximale des demandes envoyées à la file d'attente des lettres mortes S3, y compris les nouvelles tentatives, en millisecondes. Statistiques pertinentes : maximum Dimension : `PipelineName`
`s3.dlqS3RequestSizeBytes.count`	Nombre de la distribution des tailles de charge utile des demandes vers la file d'attente de lettres mortes S3, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`

Suffixe métrique	Description
`s3.dlqS3RequestSizeBytes.sum`	Distribution totale de la taille de charge utile des demandes adressées à la file d'attente de lettres mortes S3, en octets. Statistiques pertinentes : somme Dimension : `PipelineName`
`s3.dlqS3RequestSizeBytes.max`	Distribution maximale de la taille de charge utile des demandes adressées à la file d'attente de lettres mortes S3, en octets. Statistiques pertinentes : maximum Dimension : `PipelineName`

Métriques du système et de mesure

Les métriques suivantes s'appliquent à l'ensemble du système OpenSearch d'ingestion. Ces métriques ne sont préfixées par rien.

Métrique	Description
`system.cpu.usage.value`	Pourcentage d'utilisation du processeur disponible pour tous les nœuds de données. Statistiques pertinentes : moyenne Dimension : `PipelineName area`, `id`
`system.cpu.count.value`	La quantité totale d'utilisation du processeur pour tous les nœuds de données. Statistiques pertinentes : moyenne Dimension : `PipelineName area`, `id`

Métrique	Description
`jvm.memory.max.value`	Quantité maximale de mémoire pouvant être utilisée pour la gestion de mémoire, en octets. Statistiques pertinentes : moyenne Dimension : `PipelineName area`, `id`
`jvm.memory.used.value`	Quantité totale de mémoire utilisée, en octets. Statistiques pertinentes : moyenne Dimension : `PipelineName area`, `id` signal
`jvm.memory.committed.value`	Quantité de mémoire dédiée à l'utilisation par la machine virtuelle Java (JVM), en octets. Statistiques pertinentes : moyenne Dimension : `PipelineName area`, `id`
`computeUnits`	Nombre d'unités de OpenSearch calcul d'ingestion (ingestion OCUs) utilisées par un pipeline. Statistiques pertinentes : Max, Sum, Average Dimension : `PipelineName`

Bonnes pratiques relatives à Amazon OpenSearch Ingestion

Cette rubrique fournit de bonnes pratiques relatives à la création et à la gestion des pipelines OpenSearch Amazon Ingestion et inclut des consignes générales qui s'appliquent à de nombreux cas d'utilisation. Chaque charge de travail est unique, avec des caractéristiques propres, de sorte qu'aucune recommandation générique ne convient exactement à chaque cas d'utilisation.

Rubriques

- Bonnes pratiques d'ordre général
- CloudWatch Alarmes recommandées

Bonnes pratiques d'ordre général

Les meilleures pratiques générales suivantes s'appliquent à la création et à la gestion des pipelines.

- Pour garantir une haute disponibilité, configurez des pipelines VPC avec deux ou trois sous-réseaux. Si vous déployez un pipeline uniquement dans un sous-réseau et que la zone de disponibilité tombe en panne, vous ne pourrez pas ingérer de données.
- Au sein de chaque pipeline, nous recommandons de limiter le nombre de sous-pipelines à 5 ou moins.
- Si vous utilisez le plugin source S3, utilisez des fichiers S3 de taille uniforme pour des performances optimales.
- Si vous utilisez le plug-in source S3, ajoutez 30 secondes de délai de visibilité supplémentaire pour chaque 0,25 Go de taille de fichier dans le compartiment S3 pour des performances optimales.
- Incluez une file d'attente de lettres mortes (DLQ) dans la configuration de votre pipeline afin de pouvoir décharger les événements ayant échoué et les rendre accessibles pour analyse. Si vos récepteurs rejettent des données en raison de mappages incorrects ou d'autres problèmes, vous pouvez acheminer les données vers le DLQ afin de résoudre le problème.

CloudWatch Alarmes recommandées

CloudWatch les alarmes exécutent une action lorsqu'une CloudWatch métrique dépasse une valeur spécifiée pendant un certain temps. Par exemple, vous souhaiterez peut-être vous AWS envoyer un e-mail si l'état de santé de votre cluster `red` dure plus d'une minute. Cette section comprend certaines alarmes recommandées pour Amazon OpenSearch Ingestion et explique comment répondre à celles-ci.

Pour plus d'informations sur la configuration des alarmes, consultez la section Création d' CloudWatchalarmes Amazon dans le guide de CloudWatch l'utilisateur Amazon.

alerte	Problème
`computeUnits` le maximum est = configuré `maxUnits` pendant 15 minutes, 3 fois consécutives	Le pipeline a atteint sa capacité maximale et peut nécessiter une `maxUnits` mise à jour. Augmentez la valeur maximale de votre pipeline

alerte	Problème
`opensearch.documentErrors.count` La valeur est = *`{sub_pipeline_name}`* `.opensearch.recordsIn.count` somme pendant 1 minute 1 minute, 1 fois consécutive	Le pipeline ne peut pas écrire dans le OpenSearch récepteur. Vérifiez les autorisations du pipeline et confirmez que le domaine ou la collection est sain. Vous pouvez également vérifier la liste des échecs dans la file d'attente des lettres mortes (DLQ), si elle est configurée.
`bulkRequestLatency.max` La valeur maximale est >= pendant 1 minute, 1 fois consécutive	Le pipeline connaît une latence élevée lors de l'envoi des données vers le OpenSearch récepteur. Cela est probablement dû au fait que le puits est sous-dimensionné ou à une mauvaise stratégie de sharding, qui fait que le puits prend du retard. Une latence élevée et prolongée peut avoir un impact sur les performances du pipeline et entraînera probablement une contre-pression sur les clients.
`httpAuthFailure.count` somme >= 1 pendant 1 minute, 1 fois consécutive	Les demandes d'ingestion ne sont pas authentifiées. Vérifiez que l'authentification Signature Version 4 est correctement activée sur tous les clients.
`system.cpu.usage.value` valeur >= 80 % pendant 15 minutes, 3 fois consécutives	Une utilisation élevée et soutenue de l'UC peut être problématique. Envisagez d'augmenter la valeur maximale du pipeline.

alerte	Problème
`bufferUsage.value` valeur >= 80 % pendant 15 minutes, 3 fois consécutives	Une utilisation prolongée et élevée de la mémoire tampon peut être problématique. Envisagez d'augmenter la valeur maximale du pipeline.

Autres alarmes intéressantes

Pensez à configurer les alarmes suivantes selon les fonctions Amazon OpenSearch Ingestion que vous avez l'habitude d'utiliser.

alerte	Problème
`dynamodb.exportJobFailure.count` somme 1	La tentative de déclenchement d'une exportation vers Amazon S3 a échoué.
`opensearch.EndtoEndLatency.avg` moyenne > X pendant 15 minutes, 4 fois consécutives	Le `EndtoEndLatency` est supérieur à celui souhaité pour la lecture à partir de flux DynamoDB. Cela peut être dû à un OpenSearch cluster sous-dimensionné ou à une capacité OCU maximale du pipeline trop faible pour le débit WCU de la table DynamoDB. `EndtoEndLatency` sera plus élevé après une exportation, mais devrait diminuer au fil du temps à mesure qu'il rattrape les derniers flux DynamoDB.
`dynamodb.changeEventsProcessed.count` somme = 0 pendant X minutes	Aucun enregistrement n'est collecté à partir des flux DynamoDB. Cela peut être dû à l'absence d'activité sur la table ou à un problème d'accès aux flux DynamoDB.
`opensearch.s3.dlqS3RecordsSuccess.co`	Un plus grand nombre d'enregistrements sont envoyés au DLQ qu'au OpenSearch récepteur. Passez en revue les métriques du plugin OpenSearch récepteur pour rechercher et déterminer la cause première.

alerte	Problème
`unt` somme >= `opensearc h.documen tSuccess.count` somme pendant 1 minute 1 minute, 1 fois consécutive	
`grok.grok Processin gTimeouts .count` sum = RecordsIn.Count sum pendant 1 minute, 5 fois consécutives	Toutes les données expirent pendant que le processeur Grok essaie de faire correspondre les modèles. Cela a probablement un impact sur les performances et ralentit votre pipeline. Envisagez d'ajuster vos habitudes pour réduire les délais d'attente.
`grok.grok Processin gErrors.count` La valeur est >= 1 pendant 1 minute, 1 fois consécutive	Le processeur Grok ne parvient pas à faire correspondre les modèles aux données du pipeline, ce qui entraîne des erreurs. Passez en revue vos données et les configurations du plugin Grok pour vous assurer que la correspondance des modèles est attendue.
`grok.grok Processin gMismatch .count` sum = RecordsIn.Count sum pendant 1 minute, 5 fois consécutives	Le processeur Grok n'est pas en mesure de faire correspondre les modèles aux données du pipeline. Passez en revue vos données et les configurations du plugin Grok pour vous assurer que la correspondance des modèles est attendue.

alerte	Problème
`date.dateProcessingMatchFailure.count` sum = recordsIn.Count sum pendant 1 minute, 5 fois consécutives	Le processeur de données n'est pas en mesure de faire correspondre les modèles aux données du pipeline. Passez en revue les configurations de vos plug-ins Data et Date pour vous assurer que le modèle est attendu.
`s3.s3ObjectsFailed.count` somme >= 1 pendant 1 minute, 1 fois consécutive	Ce problème se produit soit parce que l'objet S3 n'existe pas, soit parce que le pipeline ne dispose pas de privilèges suffisants. Passez en revue les `s3ObjectsAccessDenied.count` indicateurs `s3ObjectsNotFound.count` et pour en déterminer la cause première. Vérifiez que l'objet S3 existe et/ou mettez à jour les autorisations.
`s3.sqsMessagesFailed.count` somme >= 1 pendant 1 minute, 1 fois consécutive	Le plugin S3 n'a pas pu traiter un message Amazon SQS. Si une DLQ est activée dans votre file d'attente SQS, consultez le message d'échec. La file d'attente reçoit peut-être des données non valides que le pipeline tente de traiter.
`http.badRequests.count` somme >= 1 pendant 1 minute, 1 fois consécutive	Le client envoie une mauvaise demande. Vérifiez que tous les clients envoient la charge utile appropriée.
`http.requestsTooLarge.count` somme >= 1 pendant 1 minute, 1 fois consécutive	Les requêtes provenant du plugin source HTTP contiennent trop de données, ce qui dépasse la capacité de la mémoire tampon. Ajustez la taille du lot pour vos clients.

alerte	Problème
`http.internalServerError.count` somme >= 0 pendant 1 minute, 1 fois consécutive	Le plugin source HTTP ne parvient pas à recevoir les événements.
`http.requestTimeouts.count` somme >= 0 pendant 1 minute, 1 fois consécutive	Les délais d'expiration de la source sont probablement le résultat d'un sous-provisionnement du pipeline. Envisagez d'augmenter le pipeline `maxUnits` pour gérer une charge de travail supplémentaire.
`otel_trace.badRequests.count` somme >= 1 pendant 1 minute, 1 fois consécutive	Le client envoie une mauvaise demande. Vérifiez que tous les clients envoient la charge utile appropriée.
`otel_trace.requestsTooLarge.count` somme >= 1 pendant 1 minute, 1 fois consécutive	Les demandes du plugin source Otel Trace contiennent trop de données, ce qui dépasse la capacité de la mémoire tampon. Ajustez la taille du lot pour vos clients.
`otel_trace.internalServerError.count` somme >= 0 pendant 1 minute, 1 fois consécutive	Le plugin source Otel Trace ne parvient pas à recevoir les événements.

alerte	Problème
`otel_trace.request Timeouts.count` somme >= 0 pendant 1 minute, 1 fois consécutive	Les délais d'expiration de la source sont probablement le résultat d'un sous-provisionnement du pipeline. Envisagez d'augmenter le pipeline `maxUnits` pour gérer une charge de travail supplémentaire.
`otel_metrics.requestTimeouts.count` somme >= 0 pendant 1 minute, 1 fois consécutive	Les délais d'expiration de la source sont probablement le résultat d'un sous-provisionnement du pipeline. Envisagez d'augmenter le pipeline `maxUnits` pour gérer une charge de travail supplémentaire.

Amazon OpenSearch sans serveur

Amazon OpenSearch Serverless est une configuration auto-scalante à la demande pour Amazon OpenSearch Service. Contrairement aux OpenSearch domaines provisionnés, qui nécessitent une gestion manuelle de la capacité, une collection OpenSearch Serverless adapte automatiquement les ressources informatiques en fonction des besoins de votre application.

OpenSearch Serverless offre une solution rentable pour les charges de travail peu fréquentes, intermittentes ou imprévisibles. Il optimise les coûts en adaptant automatiquement la capacité de calcul en fonction de l'utilisation de votre application. Les collections sans serveur utilisent le même volume de stockage à haute capacité, distribué et hautement disponible que les domaines de service provisionnés OpenSearch .

OpenSearch Les collections sans serveur sont toujours cryptées. Vous pouvez choisir la clé de chiffrement, mais vous ne pouvez pas désactiver le chiffrement. Pour de plus amples informations, consultez the section called “Chiffrement”.

Avantages

OpenSearch Le mode Serverless présente les avantages suivants :

- Plus simple que le provisionnement : le mode OpenSearch sans serveur élimine une grande partie de la complexité liée à la gestion des OpenSearch clusters et de la capacité. Il dimensionne et ajuste automatiquement vos clusters et prend en charge la gestion du cycle de vie des partitions et des index. Il gère également les mises à jour des logiciels de service et les mises à niveau des OpenSearch versions. Toutes les mises à jour et mises à niveau se font sans interruption de service.
- Rentable — Lorsque vous utilisez OpenSearch Serverless, vous ne payez que pour les ressources que vous consommez. Il n'est donc plus nécessaire de procéder à une allocation initiale et à une surallocation pour les charges de travail de pointe.
- Haute disponibilité : OpenSearch Serverless prend en charge les charges de travail de production grâce à la redondance afin de se protéger contre les pannes de zone de disponibilité et les défaillances d'infrastructure.
- Évolutif — OpenSearch Serverless adapte automatiquement les ressources afin de maintenir des taux d'ingestion de données et des temps de réponse aux requêtes toujours rapides.

Qu'est-ce qu'Amazon OpenSearch Serverless ?

Amazon OpenSearch Serverless est une option sans serveur à la demande pour Amazon OpenSearch Service qui élimine la complexité opérationnelle liée au provisionnement, à la configuration et au réglage des clusters. OpenSearch C'est la solution idéale pour les entreprises qui préfèrent ne pas gérer elles-mêmes leurs clusters ou qui ne disposent pas des ressources et de l'expertise nécessaires pour effectuer des déploiements à grande échelle. Avec OpenSearch Serverless, vous pouvez rechercher et analyser de gros volumes de données sans gérer l'infrastructure sous-jacente.

Une collection OpenSearch sans serveur est un groupe d' OpenSearch index qui fonctionnent ensemble pour prendre en charge une charge de travail ou un cas d'utilisation spécifique. Les collections simplifient les opérations par rapport aux OpenSearch clusters autogérés, qui nécessitent un provisionnement manuel.

Les collections utilisent le même stockage à haute capacité, distribué et hautement disponible que les domaines de OpenSearch service provisionnés, mais réduisent encore la complexité en éliminant la configuration et le réglage manuels. Les données d'une collection sont cryptées pendant le transfert. OpenSearch Serverless prend également en charge les OpenSearch tableaux de bord, fournissant une interface pour l'analyse des données.

Actuellement, les collections sans serveur exécutent la OpenSearch version 2.0.x. À mesure que de nouvelles versions sont publiées, OpenSearch Serverless met automatiquement à niveau les collections pour intégrer de nouvelles fonctionnalités, des corrections de bogues et des améliorations de performances.

OpenSearch Serverless prend en charge les mêmes opérations d'API d'ingestion et de requête que la suite OpenSearch open source, ce qui vous permet de continuer à utiliser vos clients et applications existants. Vos clients doivent être compatibles avec la OpenSearch version 2.x pour fonctionner avec OpenSearch Serverless. Pour de plus amples informations, veuillez consulter the section called “Ingérer des données dans les collections”.

Rubriques

- Cas d'utilisation du mode OpenSearch Serverless
- Comment ça marche
- Choix d'un type de collection
- Tarification

- Soutenu Régions AWS
- Limites
- Comparaison entre le OpenSearch service et le mode OpenSearch sans serveur

Cas d'utilisation du mode OpenSearch Serverless

OpenSearch Serverless prend en charge deux principaux cas d'utilisation :

- Analyse des journaux : le segment d'analyse des journaux se concentre sur les grands volumes de données de séries temporelles semi-structurées et générées par des machines, afin d'obtenir des informations sur les opérations et le comportement des utilisateurs.
- Recherche en texte intégral : le segment de recherche en texte intégral alimente les applications de vos réseaux internes (systèmes de gestion de contenu, documents juridiques) et les applications accessibles sur Internet, telles que la recherche de contenu sur les sites web de commerce en ligne.

Lorsque vous créez une collection, vous choisissez l'un de ces cas d'utilisation. Pour de plus amples informations, veuillez consulter the section called “Choix d'un type de collection”.

Comment ça marche

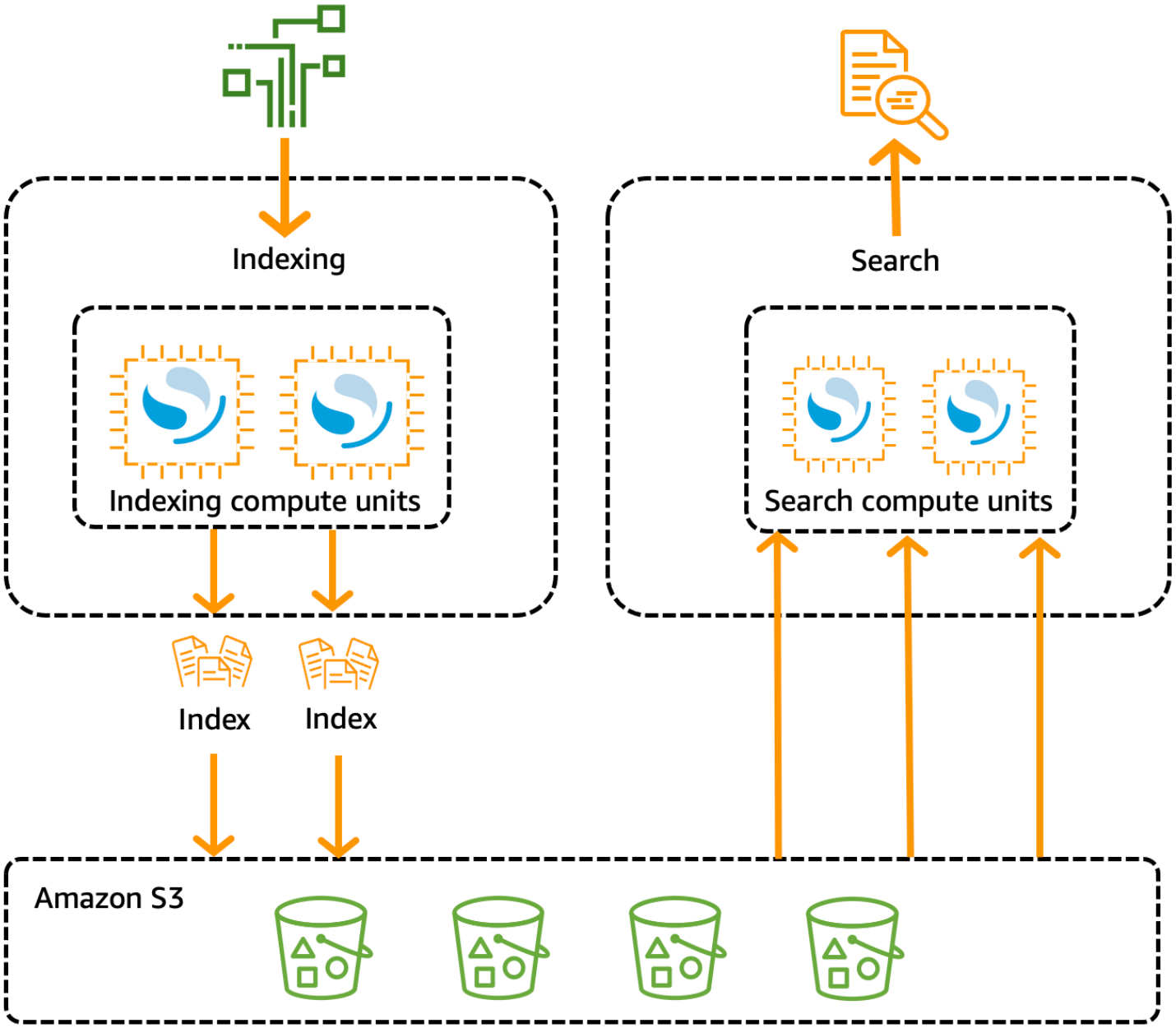
Les OpenSearch clusters traditionnels possèdent un ensemble unique d'instances qui effectuent à la fois des opérations d'indexation et de recherche, et le stockage d'index est étroitement lié à la capacité de calcul. En revanche, OpenSearch Serverless utilise une architecture native pour le cloud qui sépare les composants d'indexation (ingestion) des composants de recherche (requête), Amazon S3 étant le principal stockage de données pour les index.

Cette architecture découplée vous permet de mettre à l'échelle les fonctions de recherche et d'indexation indépendamment les unes des autres et indépendamment des données indexées dans S3. L'architecture permet également d'isoler les opérations d'ingestion et de requête afin qu'elles puissent s'exécuter simultanément sans conflit de ressources.

Lorsque vous écrivez des données dans une collection, OpenSearch Serverless les distribue aux unités de calcul d'indexation. Les unités de calcul d'indexation ingèrent les données entrantes et déplacent les index vers S3. Lorsque vous effectuez une recherche sur les données de collecte, OpenSearch Serverless achemine les demandes vers les unités de calcul de recherche qui

contiennent les données demandées. Les unités de calcul de recherche téléchargent les données indexées directement depuis S3 (si elles ne sont pas déjà mises en cache localement), exécutent des opérations de recherche et effectuent des regroupements.

L'image suivante illustre cette architecture découplée :



OpenSearch La capacité de calcul sans serveur pour l'ingestion, la recherche et l'interrogation de données est mesurée en unités de OpenSearch calcul ()OCUs. Chaque OCU est une combinaison de 6 Gio de mémoire et du processeur virtuel (vCPU) correspondant et crée un transfert de données vers Amazon S3. Chaque OCU comprend suffisamment de stockage éphémère à chaud pour 120 Gio de données d'index.

Lorsque vous créez votre première collection, OpenSearch Serverless en instancie deux, l'une pour l'indexation OCUs et l'autre pour la recherche. Afin de garantir une haute disponibilité, il lance également un ensemble de nœuds de secours dans une autre zone de disponibilité. À des fins de développement et de test, vous pouvez désactiver le paramètre Activer la redondance pour une collection, ce qui élimine les deux répliques de secours et n'en instancie que deux. OCUs Par défaut, les répliques actives redondantes sont activées, ce qui signifie qu'un total de quatre répliques OCUs sont instanciées pour la première collection d'un compte.

Ils OCUs existent même lorsqu'il n'y a aucune activité sur les points de terminaison de collecte. Toutes les collections suivantes les partagent OCUs. Lorsque vous créez des collections supplémentaires dans le même compte, OpenSearch Serverless en ajoute uniquement OCUs pour la recherche et l'ingestion si nécessaire pour prendre en charge les collections, conformément aux limites de capacité que vous spécifiez. La capacité diminue à mesure que votre utilisation des ressources informatiques diminue.

Pour plus d'informations sur la façon dont ces frais vous sont facturés OCUs, consultezthe section called “Tarification”.

Choix d'un type de collection

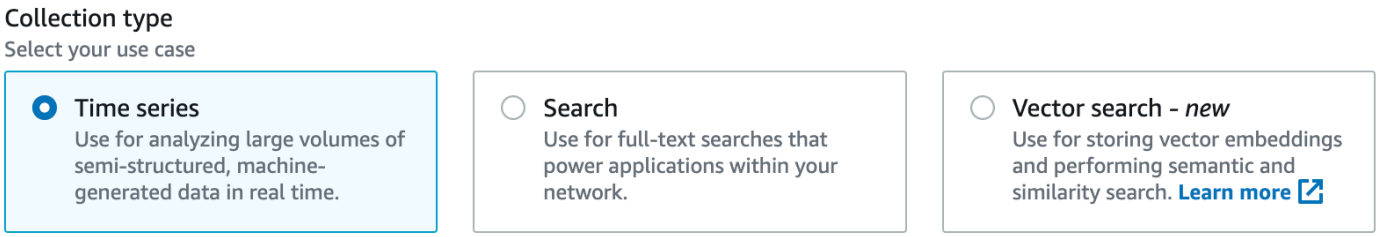
OpenSearch Serverless prend en charge trois types de collecte principaux :

Séries chronologiques : segment d'analyse des journaux qui analyse de grands volumes de données semi-structurées générées par des machines en temps réel, fournissant des informations sur les opérations, la sécurité, le comportement des utilisateurs et les performances de l'entreprise.

Recherche : recherche en texte intégral qui active les applications des réseaux internes, telles que les systèmes de gestion de contenu et les référentiels de documents juridiques, ainsi que les applications Internet telles que la recherche sur les sites de commerce électronique et la découverte de contenu.

Recherche vectorielle — La recherche sémantique sur les intégrations vectorielles simplifie la gestion des données vectorielles et permet des expériences de recherche augmentées par le machine learning (ML). Il prend en charge les applications d'IA génératives telles que les chatbots, les assistants personnels et la détection des fraudes.

Vous choisissez un type de collection lorsque vous créez une collection pour la première fois :



Le type de collection que vous choisissez dépend du type de données que vous prévoyez d'intégrer à la collection et de la manière dont vous allez les interroger. Vous ne pouvez pas modifier le type de la collection après l'avoir créée.

Les types de collection présentent les différences notables suivantes :

- Pour les collections de recherche et de recherche vectorielle, toutes les données sont stockées dans un espace de stockage à chaud afin de garantir des temps de réponse rapides aux requêtes. Les collections de séries temporelles utilisent une combinaison de stockage à chaud et tiède, les données les plus récentes étant conservées dans un stockage hot afin d'optimiser les temps de réponse aux requêtes pour les données les plus fréquemment consultées.
- Pour les séries chronologiques et les collections de recherche vectorielle, vous ne pouvez pas indexer par identifiant de document personnalisé ni mettre à jour par des requêtes upsert. Cette opération est réservée aux cas d'utilisation de recherche. Vous pouvez plutôt effectuer une mise à jour par numéro de document. Pour de plus amples informations, veuillez consulter the section called “Opérations et autorisations d' OpenSearch API prises en charge”.
- Pour les recherches et les collections de séries chronologiques, vous ne pouvez pas utiliser d'index de type K-nn.

Tarification

AWS vous facture les composants OpenSearch sans serveur suivants :

- Calcul d'ingestion de données
- Calcul de recherche et de requêtes
- Stockage conservé dans Amazon S3

Il facture l'OCU sur une base horaire, avec une granularité à la seconde. Dans votre relevé de compte, vous voyez une entrée pour le calcul en heures OCU avec une étiquette pour l'ingestion de

données et une étiquette pour la recherche. AWS vous facture également sur une base mensuelle pour les données stockées dans Amazon S3. L'utilisation des OpenSearch tableaux de bord ne vous est pas facturée.

Un minimum de 2 OCU OCUs (0,5 OCU x 2) vous est facturé pour l'ingestion et 1 OCU (0,5 OCU x 2) pour la recherche lorsque vous créez une collection et activez des répliques actives redondantes. Un minimum de 1 OCU (0,5 OCU x 2) vous est facturé pour la première collection de votre compte si vous désactivez les répliques actives redondantes. Toutes les collections suivantes peuvent les partager OCUs.

OpenSearch Serverless ajoute des unités supplémentaires par incréments de 1 OCU OCUs en fonction de la puissance de calcul et du stockage nécessaires pour prendre en charge vos collections. Vous pouvez configurer un nombre maximum de OCUs pour votre compte afin de contrôler les coûts.

> **Note**
>
> Les collections uniques ne AWS KMS keys peuvent pas être partagées OCUs avec d'autres collections.

OpenSearch Serverless tente d'utiliser les ressources minimales requises pour tenir compte de l'évolution des charges de travail. Le nombre de OCUs fournitures fournies à tout moment peut varier et n'est pas exact. Au fil du temps, l'algorithme utilisé par OpenSearch Serverless continuera de s'améliorer afin de mieux minimiser l'utilisation du système.

Pour en savoir plus sur les tarifs, consultez les tarifs d'Amazon OpenSearch Service.

Soutenu Régions AWS

OpenSearch Serverless est disponible dans un sous-ensemble de Régions AWS ce OpenSearch service disponible dans. Pour obtenir la liste des régions prises en charge, consultez la section Points OpenSearch de terminaison et quotas Amazon Service dans le Références générales AWS.

Limites

OpenSearch Le mode Serverless présente les limites suivantes :

- Certaines opérations OpenSearch d'API ne sont pas prises en charge. Consultez the section called “Opérations et autorisations d' OpenSearch API prises en charge”.

- Certains OpenSearch plugins ne sont pas pris en charge. Consultez the section called “ OpenSearch Plugins pris en charge”.
- Il n'existe actuellement aucun moyen de migrer automatiquement vos données d'un domaine de OpenSearch service géré vers une collection sans serveur. Vous devez réindexer vos données d'un domaine vers une collection.
- L'accès intercompte aux collections n'est pas pris en charge. Vous ne pouvez pas inclure les collections provenant d'autres comptes dans vos stratégies de chiffrement ou d'accès aux données.
- Les OpenSearch plugins personnalisés ne sont pas pris en charge.
- Vous ne pouvez pas prendre ou restaurer des instantanés de collections OpenSearch sans serveur.
- La recherche et la réplication entre régions ne sont pas prises en charge.
- Le nombre de ressources sans serveur que vous pouvez avoir dans un seul compte et une seule région est limité. Voir Quotas OpenSearch sans serveur.
- L'intervalle d'actualisation des index dans les collections de recherche vectorielle est d'environ 60 secondes. L'intervalle d'actualisation des index dans les recherches et les collections de séries chronologiques est d'environ 10 secondes.
- Le nombre de partitions, le nombre d'intervalles et l'intervalle d'actualisation ne sont pas modifiables et sont gérés par OpenSearch Serverless. La stratégie de partitionnement est basée sur le type de collecte et le trafic. Par exemple, une collection de séries chronologiques redimensionne les partitions primaires en fonction des goulots d'étranglement du trafic d'écriture.
- Les fonctionnalités géospatiales disponibles sur OpenSearch les versions jusqu'à 2.1 sont prises en charge.

Comparaison entre le OpenSearch service et le mode OpenSearch sans serveur

Dans OpenSearch Serverless, certains concepts et fonctionnalités sont différents de leurs fonctionnalités correspondantes pour un domaine de OpenSearch service provisionné. Par exemple, une différence importante est que OpenSearch Serverless n'a pas le concept de cluster ou de nœud.

Le tableau suivant décrit en quoi les fonctionnalités et concepts importants de OpenSearch Serverless diffèrent des fonctionnalités équivalentes dans un domaine de OpenSearch service provisionné.

Fonctionnalité	OpenSearch Service	OpenSearch Sans serveur
Domaines vs. collections	Les index sont conservés dans des domaines, qui sont des clusters pré-provisionnés OpenSearch . Pour de plus amples informations, veuillez consulter Création et gestion des domaines.	Les index sont conservés dans des collections, qui sont des regroupements logiques d'index qui représentent une charge de travail ou un cas d'utilisation spécifique. Pour de plus amples informations, veuillez consulter the section called “Gestion des collections”.
Types de nœuds et gestion de la capacité	Vous créez un cluster avec des types de nœuds qui répondent à vos spécifications en matière de coûts et de performances. Vous devez calculer vos besoins en stockage et choisir un type d'instance pour votre domaine. Pour de plus amples informations, veuillez consulter the section called “Dimensionnement des domaines”.	OpenSearch Serverless adapte et fournit automatiquement des unités de calcul supplémentaires pour votre compte en fonction de l'utilisation de votre capacité. Pour de plus amples informations, veuillez consulter the section called “Gérer les limites de capacité”.
Facturation	Vous payez pour chaque heure d'utilisation d'une EC2 instance et pour la taille cumulée de tous les volumes de stockage EBS attachés à vos instances. Pour de plus amples informations, veuillez consulter the section called “Tarification”.	Vous êtes facturé en heures d'OCU pour le calcul d'ingestion de données, le calcul de recherche et des requêtes, et le stockage conservé dans S3. Pour de plus amples informations, veuillez consulter the section called “Tarification”.
Chiffrement	Le chiffrement au repos est facultatif pour les domaines.	Le chiffrement au repos est requis pour les collections.

Fonctionnalité	OpenSearch Service	OpenSearch Sans serveur
	Pour de plus amples informations, veuillez consulter the section called “Chiffrement au repos”.	Pour de plus amples informations, veuillez consulter the section called “Chiffrement”.
Contrôle d'accès aux données	L'accès aux données au sein des domaines est déterminé par des politiques IAM et un contrôle d'accès précis.	L'accès aux données au sein des collections est déterminé par des stratégies d'accès aux données.
OpenSearchOpérations prises en charge	OpenSearch Le service prend en charge un sous-ensemble de toutes les opérations d' OpenSearch API. Pour de plus amples informations, veuillez consulter the section called “Opérations prises en charge”.	OpenSearch Serverless prend en charge un sous-ensemble différent d'opérations d' OpenSearch API. Pour de plus amples informations, veuillez consulter the section called “Opérations et plugins pris en charge”.
Connexion aux tableaux de bord	Connectez-vous à l'aide d'un nom d'utilisateur et d'un mot de passe. Pour de plus amples informations, veuillez consulter the section called “Accès aux OpenSearch tableaux de bord en tant qu'utilisateur principal”.	Si vous êtes connecté à la AWS console et que vous accédez à l'URL de votre tableau de bord, vous vous connecterez automatiquement. Pour de plus amples informations, veuillez consulter the section called “Accès aux OpenSearch tableaux de bord”.
APIs	Interagissez par programmation avec le OpenSearch service à l'aide des opérations de l'API du OpenSearch service.	Interagissez par programmation avec OpenSearch Serverless à l'aide des opérations de l'API OpenSearch Serverless.

Fonctionnalité	OpenSearch Service	OpenSearch Sans serveur
Accès réseau	Les paramètres réseau d'un domaine s'appliquent au point de terminaison du domaine ainsi qu'au point de terminaison OpenSearch des tableaux de bord. L'accès réseau pour ces deux points de terminaison est étroitement lié.	Les paramètres réseau du point de terminaison du domaine et du point de terminaison OpenSearch des tableaux de bord sont découplés. Vous pouvez choisir de ne pas configurer l'accès au réseau pour les OpenSearch tableaux de bord. Pour de plus amples informations, veuillez consulter the section called “Accès réseau”.
Signature des requêtes	Utilisez les clients REST de OpenSearch haut et de bas niveau pour signer les demandes. Spécifiez le nom du service sous la forme `es`.	À l'heure actuelle, OpenSearch Serverless prend en charge un sous-ensemble de clients pris en charge par OpenSearch Service. Lorsque vous signez des requêtes, spécifiez le nom du service sous la forme `aoss`. L'en-tête `x-amz-content-sha256` est obligatoire. Pour de plus amples informations, veuillez consulter the section called “Autres clients”.
OpenSearch mises à niveau de version	Vous mettez à niveau manuellement vos domaines au fur et à mesure que de nouvelles versions de sont OpenSearch disponibles. Vous êtes responsable de vous assurer que votre domaine répond aux exigences de mise à niveau et que vous avez pris en compte toutes les modifications majeures.	OpenSearch Serverless met automatiquement à niveau vos collections vers les nouvelles OpenSearch versions. Les mises à niveau ne se produisent pas nécessairement dès qu'une nouvelle version est disponible.

Fonctionnalité	OpenSearch Service	OpenSearch Sans serveur
Mises à jour du logiciel de service	Vous appliquez manuellement les mises à jour du logiciel de service à votre domaine dès qu'elles sont disponibles.	OpenSearch Serverless met automatiquement à jour vos collections pour utiliser les dernières corrections de bogues, fonctionnalités et améliorations de performances.
Accès VPC	Vous pouvez allouer votre domaine au sein d'un VPC. Vous pouvez également créer des points de terminaison OpenSearch VPC supplémentaires gérés par le service pour accéder au domaine.	Vous créez un ou plusieurs points de OpenSearch terminaison VPC gérés sans serveur pour votre compte. Vous incluez ensuite ces points de terminaison dans les stratégies réseau.
Authentification SAML	Vous activez l'authentification SAML par domaine. Pour de plus amples informations, veuillez consulter the section called “Authentification SAML pour les tableaux de bord OpenSearch ”.	Vous configurez un ou plusieurs fournisseurs SAML au niveau du compte, puis vous incluez l'utilisateur et le groupe associés IDs dans les politiques d'accès aux données. Pour de plus amples informations, veuillez consulter the section called “Authentification SAML”.
protocole TLS (Transport Layer Security)	OpenSearch Le service prend en charge le protocole TLS 1.2, mais il est recommandé d'utiliser le protocole TLS 1.3.	OpenSearch Serverless prend en charge le protocole TLS 1.2, mais il est recommandé d'utiliser le protocole TLS 1.3.

Tutoriel : Démarrage avec Amazon OpenSearch Serverless

Ce didacticiel vous explique les étapes de base pour mettre rapidement en place une collection de recherche Amazon OpenSearch Serverless. Une collection de recherche vous permet d'alimenter les

applications de vos réseaux internes et les applications connectées à Internet, telles que la recherche sur les sites Web de commerce électronique et la recherche de contenu.

Pour savoir comment utiliser une collection de recherche vectorielle, voirthe section called “Utilisation de collections de recherche vectorielle”. Pour des informations plus détaillées sur l'utilisation des collections, consultez the section called “Gestion des collections” et les autres rubriques de ce guide.

Dans le cadre de ce didacticiel, vous suivrez les étapes suivantes :

1. Configurer des autorisations
2. Créer une collection
3. Charger et rechercher des données
4. Supprimer la collection

> **Note**
>
> Nous vous recommandons de n'utiliser que des caractères ASCII pour votre. `IndexName` Si vous n'utilisez pas de caractères ASCII pour votre compte`IndexName`, les CloudWatch métriques entrantes seront converties `IndexName` en un format codé URL pour les caractères non ASCII.

Étape 1 : configurer des autorisations

Pour suivre ce didacticiel, et pour utiliser OpenSearch Serverless en général, vous devez disposer des autorisations IAM appropriées. Dans le cadre de ce didacticiel, vous allez créer une collection, charger et rechercher des données, puis supprimer la collection.

Votre utilisateur ou votre rôle doit être associé à une politique basée sur l'identité avec les autorisations minimales suivantes :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aoss:CreateCollection",
```

```
        "aoss:ListCollections",
        "aoss:BatchGetCollection",
        "aoss:DeleteCollection",
        "aoss:CreateAccessPolicy",
        "aoss:ListAccessPolicies",
        "aoss:UpdateAccessPolicy",
        "aoss:CreateSecurityPolicy",
        "aoss:GetSecurityPolicy",
        "aoss:UpdateSecurityPolicy",
        "iam:ListUsers",
        "iam:ListRoles"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Pour plus d'informations sur les autorisations IAM OpenSearch sans serveur, consultez. the section called “Gestion de l’identité et des accès”

Étape 2 : créer une collection

Une collection est un groupe d' OpenSearch index qui fonctionnent ensemble pour prendre en charge une charge de travail ou un cas d'utilisation spécifique.

Pour créer une collection OpenSearch sans serveur

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Choisissez Collections dans le panneau de navigation de gauche, puis choisissez Create collection (Créer une collection).
3. Nommez la collection movies (films).
4. Pour le type de collection, choisissez Search (Rechercher). Pour plus d'informations, consultez Choisir un type de collection.
5. Pour Sécurité, choisissez Création standard.
6. Sous Chiffrement, sélectionnez Utiliser Clé détenue par AWS. C'est ce AWS KMS key que OpenSearch Serverless utilisera pour chiffrer vos données.
7. Sous Network (Réseau), configurez les paramètres réseau de la collection.

- Pour le type d'accès, sélectionnez Public.
- Pour le type de ressource, sélectionnez Activer l'accès aux OpenSearch points de terminaison et Activer l'accès aux OpenSearch tableaux de bord. Comme vous téléchargerez et rechercherez des données à l'aide de OpenSearch tableaux de bord, vous devez activer les deux.

8. Choisissez Suivant.
9. Pour Configure data access (Configurer l'accès aux données), configurez les paramètres d'accès pour la collection. Les stratégies d'accès aux données permettent aux utilisateurs et aux rôles d'accéder aux données d'une collection. Dans le cadre de ce didacticiel, nous allons fournir à un seul utilisateur les autorisations requises pour indexer et rechercher des données dans la collection movies.

   Créez une règle unique qui donne accès à la collection de films. Nommez la règle Movies collection access (Accès à la collection movies).
10. Choisissez Ajouter des principaux, des utilisateurs et des rôles IAM, puis sélectionnez l'utilisateur ou le rôle que vous utiliserez pour vous connecter aux OpenSearch tableaux de bord et indexer les données. Choisissez Enregistrer.
11. Sous Index permissions (Autorisations d'index), sélectionnez toutes les autorisations.
12. Choisissez Suivant.
13. Pour les paramètres de la stratégie d'accès, choisissez Create a new data access policy (Créer une nouvelle stratégie d'accès aux données) et nommez la stratégie movies (films).
14. Choisissez Suivant.
15. Vérifiez vos paramètres de collection et choisissez Submit (Soumettre). Attendez quelques minutes pour que le statut de la collection devienne `Active`.

Étape 3 : charger et rechercher des données

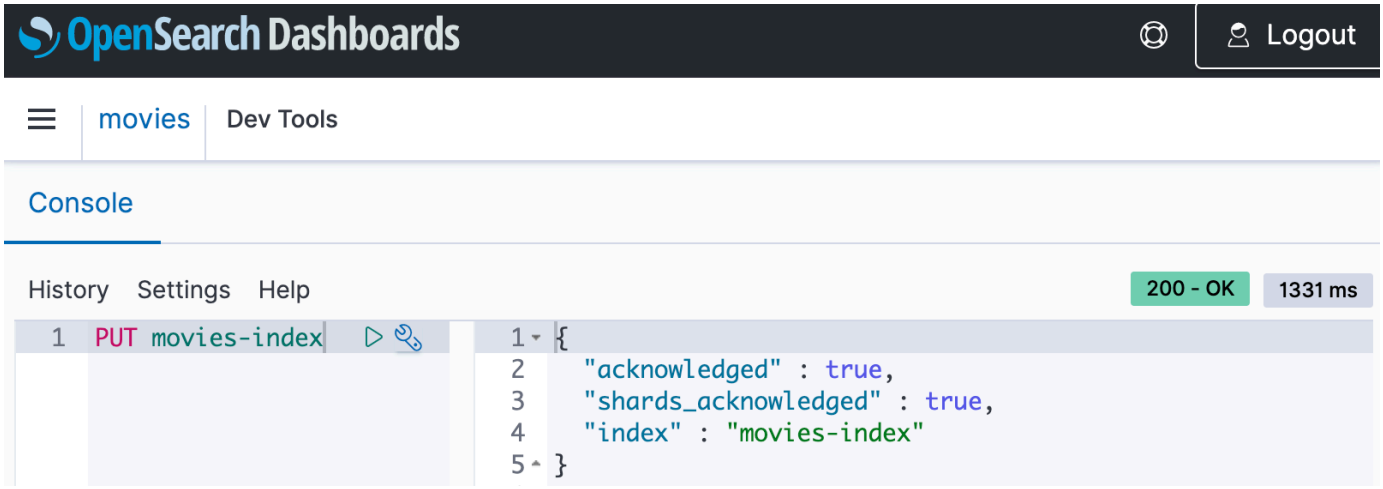
Vous pouvez télécharger des données vers une collection OpenSearch sans serveur à l'aide de Postman ou cURL. Par souci de concision, ces exemples utilisent les outils de développement de la console OpenSearch Dashboards.

Indexer et rechercher des données dans la collection movies

1. Choisissez Collections dans le panneau de navigation de gauche, puis choisissez la collection movies pour afficher sa page des détails.

2. Choisissez l'URL OpenSearch des tableaux de bord pour la collection. L'URL est au format `https://dashboards.{region}.aoss.amazonaws.com/_login/?collectionId={collection-id}`.
3. Dans OpenSearch Dashboards, ouvrez le volet de navigation de gauche et choisissez Dev Tools.
4. Pour créer un index unique appelé movies-index, envoyez la requête suivante :

```
PUT movies-index
```



5. Pour indexer un seul document dans movies-index, envoyez la requête suivante :

```
PUT movies-index/_doc/1
{
  "title": "Shawshank Redemption",
  "genre": "Drama",
  "year": 1994
}
```

6. Pour rechercher des données dans OpenSearch les tableaux de bord, vous devez configurer au moins un modèle d'index. OpenSearch utilise ces modèles pour identifier les index que vous souhaitez analyser. Ouvrez le panneau de navigation de gauche, choisissez Stack Management (Gestion des piles), choisissez Index Patterns (Modèles d'index), puis Create index pattern (Créer un modèle d'index). Dans le cadre de ce tutoriel, saisissez movies.
7. Choisissez Next step (Étape suivante), puis Create index pattern (Créer un modèle d'index). Une fois le modèle créé, vous pouvez consulter les différents champs du document, comme `title` et `genre`.

8. Pour commencer à rechercher vos données, ouvrez à nouveau le panneau de navigation de gauche et choisissez Discover (Découvrir), ou utilisez l'API de recherche dans les outils de développement.

Étape 4 : supprimer la collection

Étant donné que la collection movies est destinée aux tests, veillez à la supprimer lorsque vous aurez fini de l'utiliser.

Pour supprimer une collection OpenSearch sans serveur

1. Revenez à la console Amazon OpenSearch Service.
2. Choisissez Collections dans le panneau de navigation de gauche et sélectionnez la collection movies.
3. Choisissez Delete (Supprimer) et confirmez la suppression.

Étapes suivantes

Maintenant que vous savez comment créer une collection et indexer des données, n'hésitez pas à essayer les exercices suivants :

- Découvrez des options de création de collection plus avancées. Pour de plus amples informations, veuillez consulter the section called “Gestion des collections”.
- Découvrez comment configurer des stratégies de sécurité pour gérer la sécurité des collections à grande échelle. Pour de plus amples informations, veuillez consulter the section called “Sécurité en mode OpenSearch Serverless”.
- Découvrez d'autres moyens d'indexer les données dans des collections. Pour de plus amples informations, veuillez consulter the section called “Ingérer des données dans les collections”.

Création et gestion de collections Amazon OpenSearch Serverless

Vous pouvez créer des collections Amazon OpenSearch Serverless à l'aide de la console, de l'API AWS CLI and AWS SDKs, et AWS CloudFormation.

Rubriques

- Gestion des collections Amazon OpenSearch Serverless

- Utilisation de collections de recherche vectorielle
- Utilisation des politiques de cycle de vie des données avec Amazon OpenSearch Serverless
- Utilisation du AWS SDKs pour interagir avec Amazon OpenSearch Serverless
- Utilisation AWS CloudFormation pour créer des collections Amazon OpenSearch Serverless

Gestion des collections Amazon OpenSearch Serverless

Une collection dans Amazon OpenSearch sans serveur est un regroupement logique d'un ou de plusieurs index qui représentent une charge de travail analytique. OpenSearch sans serveur gère et ajuste automatiquement la collection, ne nécessitant qu'une saisie manuelle minimale.

Rubriques

- Autorisations nécessaires
- Créer des collections
- Accès aux OpenSearch tableaux de bord
- Afficher les collections
- Supprimer des collections

Autorisations nécessaires

OpenSearch sans serveur utilise les autorisations AWS Identity and Access Management (IAM) suivantes pour créer et gérer des collections. Vous pouvez spécifier des conditions IAM pour restreindre les utilisateurs à des collections spécifiques.

- `aoss:CreateCollection` : créer une collection.
- `aoss:ListCollections` : répertorier les collections du compte actuel.
- `aoss:BatchGetCollection` : obtenir des informations sur une ou plusieurs collections.
- `aoss:UpdateCollection` : modifier une collection.
- `aoss:DeleteCollection` : supprimer une collection.

L'exemple suivant de stratégie d'accès basée sur l'identité fournit les autorisations minimales nécessaires à un utilisateur pour gérer une collection unique nommée `Logs` :

```
[
```

```
   {
      "Sid":"Allows managing logs collections",
      "Effect":"Allow",
      "Action":[
         "aoss:CreateCollection",
         "aoss:ListCollections",
         "aoss:BatchGetCollection",
         "aoss:UpdateCollection",
         "aoss:DeleteCollection",
         "aoss:CreateAccessPolicy",
         "aoss:CreateSecurityPolicy"
      ],
      "Resource":"*",
      "Condition":{
         "StringEquals":{
            "aoss:collection":"Logs"
         }
      }
   }
]
```

`aoss:CreateAccessPolicy` et `aoss:CreateSecurityPolicy` sont incluses, car des stratégies de chiffrement, de réseau et d'accès aux données sont nécessaires au bon fonctionnement d'une collection. Pour de plus amples informations, veuillez consulter the section called “Gestion de l’identité et des accès”.

> **Note**
>
> Si vous créez la première collection de votre compte, vous devez également disposer de l'autorisation `iam:CreateServiceLinkedRole`. Pour de plus amples informations, veuillez consulter the section called “Rôle de création d'une collection”.

Créer des collections

Vous pouvez utiliser la console ou l' AWS CLI pour créer une collection sans serveur. Ces étapes expliquent comment créer une recherche ou une collection de séries chronologiques. Pour créer une collection de recherche vectorielle, voirthe section called “Utilisation de collections de recherche vectorielle”.

Créer une collection (console)

Créer une collection à l'aide de la console

1. Accédez à la console Amazon OpenSearch Service à l'adresse https://console.aws.amazon.com/aos/home/.
2. Développez Serverless (Sans serveur) dans le panneau de navigation de gauche et choisissez Collections.
3. Choisissez Create collection (Créer une collection).
4. Saisissez un nom et une description pour la collection. Le nom doit répondre aux critères suivants :
   - Est unique sur votre compte et dans votre Région AWS
   - Commence par une lettre minuscule
   - Contient entre 3 et 32 caractères
   - Contient uniquement les lettres minuscules a-z, les chiffres 0-9 et le trait d'union (-)
5. Choisissez un type de collection :
   - Séries temporelles : segment d'analyse des journaux qui se concentre sur des gros volumes de données semi-structurées et générées par des machines. Au moins 24 heures de données sont stockées sur des index chauds, le reste étant conservé dans un stockage tiède.
   - Recherche : recherche en texte intégral qui alimente les applications de vos réseaux internes et les applications disponibles sur Internet. Toutes les données de recherche sont stockées dans le stockage à chaud afin de garantir des temps de réponse rapides aux requêtes.
   - Recherche vectorielle : recherche sémantique sur les intégrations vectorielles qui simplifie la gestion des données vectorielles. Favorise les expériences de recherche augmentées par le machine learning (ML) et les applications d'IA génératives telles que les chatbots, les assistants personnels et la détection des fraudes.

   Pour de plus amples informations, veuillez consulter the section called “Choix d'un type de collection”.
6. Pour Type de déploiement, choisissez le paramètre de redondance pour votre collection. Par défaut, chaque collection est redondante, ce qui signifie que les unités de OpenSearch calcul d'indexation et de recherche (OCUs) disposent chacune de leurs propres répliques de secours dans une zone de disponibilité différente. À des fins de développement et de test, vous pouvez

choisir de désactiver la redondance, ce qui réduit à deux le nombre de OCUs membres de votre collection. Pour de plus amples informations, veuillez consulter the section called “Comment ça marche”.

7. Pour Sécurité, choisissez Création standard.
8. Dans Encryption (Chiffrement), choisissez une AWS KMS clé avec laquelle chiffrer vos données. OpenSearch sans serveur vous avertit si le nom de collection que vous avez saisi correspond à un modèle défini dans une stratégie de chiffrement. Vous pouvez choisir de conserver cette correspondance ou de la remplacer par des paramètres de chiffrement uniques. Pour de plus amples informations, veuillez consulter the section called “Chiffrement”.
9. Dans Network access settings (Paramètres d'accès au réseau), configurez l'accès au réseau pour la collection.

   - Pour le type d'accès, sélectionnez public ou privé.

     Si vous choisissez privé, spécifiez les points de terminaison VPC autorisés à Services AWS accéder à la collection.

     - Points de terminaison d'un VPC via lesquels autoriser l'accès. Pour créer un point de terminaison d'un VPC, veuillez consulter la rubrique the section called “Points de terminaison d’un VPC”.
     - Service AWS accès privé : sélectionnez un ou plusieurs services pris en charge auxquels vous souhaitez autoriser l'accès.
   - Dans Resource type (Type de ressource), choisissez si les utilisateurs peuvent accéder à la collection via son OpenSearchpoint de terminaison (pour effectuer des appels d'API via cURL, Postman, etc.), via le point de terminaison OpenSearch tableaux de bord (pour fonctionner avec des visualisations et effectuer des appels d'API via la console), ou les deux.

     > **Note**
     >
     > Service AWS l'accès privé s'applique uniquement au OpenSearch point de terminaison, pas au point de terminaison OpenSearch des tableaux de bord.

   OpenSearch sans serveur vous avertit si le nom de collection que vous avez saisi correspond à un modèle défini dans une stratégie réseau. Vous pouvez choisir de conserver cette correspondance ou de la remplacer par des paramètres réseau personnalisés. Pour de plus amples informations, veuillez consulter the section called “Accès réseau”.

10. (Facultatif) Ajoutez une ou plusieurs balises à la collection. Pour de plus amples informations, veuillez consulter the section called “Baliser des collections”.
11. Choisissez Suivant.
12. Configurez les règles d'accès aux données pour la collection, qui définissent qui peut accéder aux données de la collection. Pour chaque règle que vous créez, effectuez les opérations suivantes :

    - Choisissez Add principals (Ajouter des principaux) et sélectionnez un ou plusieurs rôles IAM ou utilisateurs et groupes SAML pour accorder l'accès aux données.
    - Sous Grant permissions (Accorder des autorisations), sélectionnez les autorisations d'alias, de modèle et d'index à accorder aux principaux associés. Pour obtenir la liste complète des autorisations et des accès qu'elles octroient, veuillez consulter la rubrique the section called “Opérations et autorisations d' OpenSearch API prises en charge”.

    OpenSearch sans serveur vous avertit si le nom de collection que vous avez saisi correspond à un modèle défini dans une stratégie d'accès aux données. Vous pouvez choisir de conserver cette correspondance ou de la remplacer par des paramètres d'accès aux données uniques. Pour de plus amples informations, veuillez consulter the section called “Contrôle d'accès aux données”.
13. Choisissez Suivant.
14. Dans Data access policy settings (Paramètres de la stratégie d'accès aux données), choisissez ce que vous voulez faire avec les règles que vous venez de créer. Vous pouvez soit les utiliser pour créer une nouvelle stratégie d'accès aux données, soit les ajouter à une stratégie existante.
15. Choisissez Suivant.
16. Vérifiez la configuration de votre collection et choisissez Submit (Soumettre).

Le statut de la collection passe à l'état `Creating` de la collection OpenSearch lorsque sans serveur crée la collection.

Créer une collection (CLI)

Avant de créer une collection à l'aide de l' AWS CLI, vous devez disposer d'une stratégie de chiffrement avec un modèle de ressource correspondant au nom prévu pour la collection. Par exemple, si vous envisagez de nommer votre collection logs-application, vous pouvez créer une stratégie de chiffrement comme suit :

```
aws opensearchserverless create-security-policy \
  --name logs-policy \
  --type encryption --policy "{\"Rules\":[{\"ResourceType\":\"collection\",\"Resource
\":[\"collection\/logs-application\"]}],\"AWSOwnedKey\":true}"
```

Si vous envisagez d'utiliser la stratégie pour des collections supplémentaires, vous pouvez élargir la règle, par exemple `collection/logs*` ou `collection/*`.

Vous devez également configurer les paramètres réseau pour la collection sous la forme d'une stratégie réseau. À l'aide de l'exemple logs-application précédent, vous pouvez créer la stratégie réseau suivante :

```
aws opensearchserverless create-security-policy \
  --name logs-policy \
  --type network --policy "[{\"Description\":\"Public access for logs collection
\",\"Rules\":[{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection\/logs-
application\"]},{\"ResourceType\":\"collection\",\"Resource\":[\"collection\/logs-
application\"]}],\"AllowFromPublic\":true}]"
```

> **Note**
>
> Vous pouvez créer des stratégies réseau après avoir créé une collection, mais nous vous recommandons de le faire au préalable.

Pour créer une collection, envoyez une CreateCollectiondemande :

```
aws opensearchserverless create-collection --name "logs-application" --type SEARCH --
description "A collection for storing log data"
```

Pour `type`, spécifiez `SEARCH` ou `TIMESERIES`. Pour de plus amples informations, veuillez consulter the section called “Choix d'un type de collection”.

Exemple de réponse

```
{
    "createCollectionDetail": {
        "id": "07tjusf2h91cunochc",
        "name": "books",
```

```
        "description":"A collection for storing log data",
        "status": "CREATING",
        "type": "SEARCH",
        "kmsKeyArn": "auto",
        "arn": "arn:aws:aoss:us-east-1:123456789012:collection/07tjusf2h91cunochc",
        "createdDate": 1665952577473
    }
}
```

Si vous ne spécifiez pas de type de collection dans la demande, il est défini par défaut à `TIMESERIES`. Si votre collection est chiffrée avec une Clé détenue par AWS, l'`kmsKeyArn` est `auto` plutôt qu'un ARN.

> **Important**
>
> Une fois la collection créée, vous ne pourrez y accéder que si elle correspond à une stratégie d'accès aux données. Pour obtenir des instructions sur la création de stratégies d'accès aux données, veuillez consulter la rubrique the section called “Contrôle d'accès aux données”.

Accès aux OpenSearch tableaux de bord

Après avoir créé une collection avec le AWS Management Console, vous pouvez naviguer vers l'URL des OpenSearch tableaux de bord de la collection. Vous pouvez trouver l'URL des tableaux de bord en choisissant Collections dans le panneau de navigation de gauche et en sélectionnant la collection pour ouvrir sa page de détails. L'URL est au format `https://dashboards.`*`us-east-1`*`.aoss.amazonaws.com/_login/?collectionId=`*`07tjusf2h91cunochc`*. Une fois que vous avez accédé à l'URL, vous vous connectez automatiquement aux tableaux de bord.

Si l'URL des OpenSearch tableaux de bord est déjà disponible mais que vous n'y êtes pas AWS Management Console, l'appel de l'URL des tableaux de bord depuis le navigateur sera redirigé vers la console. Une fois que vous aurez saisi vos AWS informations d'identification, vous vous connecterez automatiquement aux tableaux de bord. Pour plus d'informations sur l'accès aux collections pour le protocole SAML, consultez la section Accès aux OpenSearch tableaux de bord avec le protocole SAML.

Le délai d'expiration OpenSearch de la console Dashboards est d'une heure et n'est pas configurable.

> **Note**
>
> Le 10 mai 2023, OpenSearch a introduit un point de terminaison mondial commun pour les OpenSearch tableaux de bord. Vous pouvez désormais accéder aux OpenSearch tableaux de bord dans le navigateur à l'aide d'une URL au format `https://dashboards.us-east-1.aoss.amazonaws.com/_login/?collectionId=07tjusf2h91cunochc` approprié. Pour garantir la rétrocompatibilité, nous continuerons à prendre en charge les points de terminaison des OpenSearch tableaux de bord spécifiques à la collection existante avec ce format. `https://07tjusf2h91cunochc.us-east-1.aoss.amazonaws.com/_dashboards`

Afficher les collections

Vous pouvez afficher les collections existantes dans votre avec l' Compte AWS onglet Collections de la console Amazon OpenSearch Service console.

Pour répertorier les collections avec leurs IDs, envoyez une ListCollectionsrequête.

```
aws opensearchserverless list-collections
```

Exemple de réponse

```
{
   "collectionSummaries":[
      {
         "arn":"arn:aws:aoss:us-east-1:123456789012:collection/07tjusf2h91cunochc",
         "id":"07tjusf2h91cunochc",
         "name":"my-collection",
         "status":"CREATING"
      }
   ]
}
```

Pour limiter les résultats de recherche, utilisez des filtres de collection. Cette requête filtre la réponse aux collections à l'état `ACTIVE` :

```
aws opensearchserverless list-collections --collection-filters '{ "status": "ACTIVE" }'
```

Pour obtenir des informations plus détaillées sur une ou plusieurs collections, notamment le OpenSearch point de terminaison et le point de terminaison OpenSearch des tableaux de bord, envoyez une BatchGetCollectionrequête :

```
aws opensearchserverless batch-get-collection --ids ["07tjusf2h91cunochc",
 "1iu5usc4rame"]
```

> **Note**
>
> Vous pouvez inclure `--names` ou `--ids` dans la requête, mais pas les deux.

Exemple de réponse


```
{
   "collectionDetails":[
      {
         "id": "07tjusf2h91cunochc",
         "name": "my-collection",
         "status": "ACTIVE",
         "type": "SEARCH",
         "description": "",
         "arn": "arn:aws:aoss:us-east-1:123456789012:collection/07tjusf2h91cunochc",
         "kmsKeyArn": "arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
         "createdDate": 1667446262828,
         "lastModifiedDate": 1667446300769,
         "collectionEndpoint": "https://07tjusf2h91cunochc.us-
east-1.aoss.amazonaws.com",
         "dashboardEndpoint": "https://07tjusf2h91cunochc.us-east-1.aoss.amazonaws.com/
_dashboards"
      },
      {
         "id": "178ukvtg3i82dvopdid",
         "name": "another-collection",
         "status": "ACTIVE",
         "type": "TIMESERIES",
         "description": "",
         "arn": "arn:aws:aoss:us-east-1:123456789012:collection/178ukvtg3i82dvopdid",
         "kmsKeyArn": "arn:aws:kms:us-
east-1:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
         "createdDate": 1667446262828,
```

```
         "lastModifiedDate": 1667446300769,
         "collectionEndpoint": "https://178ukvtg3i82dvopdid.us-
east-1.aoss.amazonaws.com",
         "dashboardEndpoint": "https://178ukvtg3i82dvopdid.us-
east-1.aoss.amazonaws.com/_dashboards"
      }
   ],
   "collectionErrorDetails":[]
}
```

Supprimer des collections

La suppression d'une collection entraîne la suppression de toutes ses données et de tous ses index. Vous ne pouvez pas récupérer les collections après les avoir supprimées.

Supprimer une collection à l'aide de la console

1. Dans le panneau Collections de la console Amazon OpenSearch Service, sélectionnez la collection que vous souhaitez supprimer.
2. Choisissez Delete (Supprimer) et confirmez la suppression.

Pour supprimer une collection à l'aide du AWS CLI, envoyez une DeleteCollectiondemande :

```
aws opensearchserverless delete-collection --id 07tjusf2h91cunochc
```

Exemple de réponse

```
{
   "deleteCollectionDetail":{
      "id":"07tjusf2h91cunochc",
      "name":"my-collection",
      "status":"DELETING"
   }
}
```

Utilisation de collections de recherche vectorielle

Le type de collection de recherche vectorielle dans OpenSearch Serverless fournit une fonctionnalité de recherche de similarité évolutive et performante. Il vous permet de créer facilement des expériences modernes de recherche augmentée par apprentissage automatique (ML) et des

applications d'intelligence artificielle générative (IA) sans avoir à gérer l'infrastructure de base de données vectorielle sous-jacente.

Les exemples d'utilisation des collections de recherche vectorielle incluent les recherches d'images, les recherches de documents, la récupération de musique, les recommandations de produits, les recherches de vidéos, les recherches géolocalisées, la détection des fraudes et la détection des anomalies.

Comme le moteur vectoriel de OpenSearch Serverless est alimenté par la fonction de recherche du voisin le plus proche (k-NN) dans OpenSearch, vous bénéficiez des mêmes fonctionnalités avec la simplicité d'un environnement sans serveur. Le moteur prend en charge l'API du plugin K-nn. Grâce à ces opérations, vous pouvez tirer parti de la recherche en texte intégral, du filtrage avancé, des agrégations, des requêtes géospatiales, des requêtes imbriquées pour une extraction plus rapide des données et des résultats de recherche améliorés.

Le moteur vectoriel fournit des mesures de distance telles que la distance euclidienne, la similitude des cosinus et la similitude des produits par points, et peut prendre en charge 16 000 dimensions. Vous pouvez stocker des champs contenant différents types de données pour les métadonnées, tels que des nombres, des booléens, des dates, des mots clés et des points géographiques. Vous pouvez également stocker des champs avec du texte pour obtenir des informations descriptives afin d'ajouter du contexte aux vecteurs stockés. La colocation des types de données réduit la complexité, augmente la maintenabilité et évite la duplication des données, les problèmes de compatibilité des versions et les problèmes de licence.

> **Note**
>
> Amazon OpenSearch Serverless prend en charge la quantification scalaire 16 bits Faiss, qui peut être utilisée pour effectuer des conversions entre des vecteurs flottants 32 bits et des vecteurs 16 bits. Pour en savoir plus, consultez la section Quantification scalaire 16 bits Faiss. Vous pouvez également utiliser des vecteurs binaires pour réduire les coûts de mémoire. Pour plus d'informations, consultez la section Vecteurs binaires.

Commencer à utiliser les collections de recherche vectorielle

Dans ce didacticiel, vous allez effectuer les étapes suivantes pour stocker, rechercher et récupérer des intégrations vectorielles en temps réel :

1. Configurer des autorisations

2. Créer une collection
3. Charger et rechercher des données
4. Supprimer la collection

Étape 1 : configurer des autorisations

Pour suivre ce didacticiel (et pour utiliser OpenSearch Serverless en général), vous devez disposer des autorisations AWS Identity and Access Management (IAM) appropriées. Dans ce didacticiel, vous allez créer une collection, télécharger et rechercher des données, puis supprimer la collection.

Votre utilisateur ou votre rôle doit être associé à une politique basée sur l'identité avec les autorisations minimales suivantes :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aoss:CreateCollection",
        "aoss:ListCollections",
        "aoss:BatchGetCollection",
        "aoss:DeleteCollection",
        "aoss:CreateAccessPolicy",
        "aoss:ListAccessPolicies",
        "aoss:UpdateAccessPolicy",
        "aoss:CreateSecurityPolicy",
        "iam:ListUsers",
        "iam:ListRoles"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Pour plus d'informations sur les autorisations IAM OpenSearch sans serveur, consultez. the section called “Gestion de l’identité et des accès”

Étape 2 : créer une collection

Une collection est un groupe d' OpenSearch index qui fonctionnent ensemble pour prendre en charge une charge de travail ou un cas d'utilisation spécifique.

Pour créer une collection OpenSearch sans serveur

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Choisissez Collections dans le panneau de navigation de gauche, puis choisissez Create collection (Créer une collection).
3. Nommez le boîtier de collection.
4. Pour le type de collection, choisissez Recherche vectorielle. Pour de plus amples informations, veuillez consulter the section called “Choix d'un type de collection”.
5. Sous Type de déploiement, désélectionnez Activer la redondance (répliques actives). Cela crée une collection en mode développement ou test et réduit à deux le nombre d'unités de OpenSearch calcul (OCUs) de votre collection. Si vous souhaitez créer un environnement de production dans ce didacticiel, laissez la case cochée.
6. Sous Sécurité, sélectionnez Easy create pour rationaliser votre configuration de sécurité. Toutes les données du moteur vectoriel sont cryptées en transit et au repos par défaut. Le moteur vectoriel prend en charge les autorisations IAM détaillées afin que vous puissiez définir qui peut créer, mettre à jour et supprimer des chiffrements, des réseaux, des collections et des index.
7. Choisissez Suivant.
8. Vérifiez vos paramètres de collection et choisissez Submit (Soumettre). Attendez quelques minutes pour que le statut de la collection devienne `Active`.

Étape 3 : charger et rechercher des données

Un index est un ensemble de documents dotés d'un schéma de données commun qui vous permet de stocker, de rechercher et de récupérer vos intégrations vectorielles et d'autres champs. Vous pouvez créer et télécharger des données vers les index d'une collection OpenSearch sans serveur à l'aide de la console Dev Tools dans les OpenSearch tableaux de bord ou d'un outil HTTP tel que Postman ou awscurl. Ce didacticiel utilise les outils de développement.

Indexer et rechercher des données dans la collection movies

1. Pour créer un index unique pour votre nouvelle collection, envoyez la demande suivante dans la console Dev Tools. Par défaut, cela crée un index avec un `nmslib` moteur et une distance euclidienne.

```
PUT housing-index
{
   "settings": {
      "index.knn": true
   },
   "mappings": {
      "properties": {
         "housing-vector": {
            "type": "knn_vector",
            "dimension": 3
         },
         "title": {
            "type": "text"
         },
         "price": {
            "type": "long"
         },
         "location": {
            "type": "geo_point"
         }
      }
   }
}
```

2. Pour indexer un seul document dans housing-index, envoyez la demande suivante :

```
POST housing-index/_doc
{
  "housing-vector": [
    10,
    20,
    30
  ],
  "title": "2 bedroom in downtown Seattle",
  "price": "2800",
  "location": "47.71, 122.00"
```

```
}
```

3. Pour rechercher des propriétés similaires à celles de votre index, envoyez la requête suivante :

```
GET housing-index/_search
{
    "size": 5,
    "query": {
        "knn": {
            "housing-vector": {
                "vector": [
                    10,
                    20,
                    30
                ],
                "k": 5
            }
        }
    }
}
```

Étape 4 : supprimer la collection

La collection de logements étant destinée à des fins de test, assurez-vous de la supprimer lorsque vous aurez terminé d'expérimenter.

Pour supprimer une collection OpenSearch sans serveur

1. Revenez à la console Amazon OpenSearch Service.
2. Choisissez Collections dans le volet de navigation de gauche et sélectionnez la collection de propriétés.
3. Choisissez Supprimer et confirmez la suppression.

Recherche filtrée

Vous pouvez utiliser des filtres pour affiner les résultats de votre recherche sémantique. Pour créer un index et effectuer une recherche filtrée sur vos documents, remplacez les données de téléchargement et de recherche du didacticiel précédent par les instructions suivantes. Les autres étapes restent les mêmes. Pour plus d'informations sur les filtres, voir K-nn search with filters.

Indexer et rechercher des données dans la collection movies

1. Pour créer un index unique pour votre collection, envoyez la demande suivante dans la console Dev Tools :

```
PUT housing-index-filtered
{
  "settings": {
    "index.knn": true
  },
  "mappings": {
    "properties": {
      "housing-vector": {
        "type": "knn_vector",
        "dimension": 3,
        "method": {
          "engine": "faiss",
          "name": "hnsw"
        }
      },
      "title": {
        "type": "text"
      },
      "price": {
        "type": "long"
      },
      "location": {
        "type": "geo_point"
      }
    }
  }
}
```

2. Pour indexer un seul document dans housing-index-filtered, envoyez la demande suivante :

```
POST housing-index-filtered/_doc
{
  "housing-vector": [
    10,
    20,
    30
  ],
  "title": "2 bedroom in downtown Seattle",
```

```
  "price": "2800",
  "location": "47.71, 122.00"
}
```

3. Pour rechercher vos données pour un appartement à Seattle à un prix donné et à une distance donnée d'un point géographique, envoyez la demande suivante :

```
GET housing-index-filtered/_search
{
  "size": 5,
  "query": {
    "knn": {
      "housing-vector": {
        "vector": [
          0.1,
          0.2,
          0.3
        ],
        "k": 5,
        "filter": {
          "bool": {
            "must": [
              {
                "query_string": {
                  "query": "Find me 2 bedroom apartment in Seattle under $3000 ",
                  "fields": [
                    "title"
                  ]
                }
              },
              {
                "range": {
                  "price": {
                    "lte": 3000
                  }
                }
              },
              {
                "geo_distance": {
                  "distance": "100miles",
                  "location": {
                    "lat": 48,
                    "lon": 121
```

```
                    }
                  }
                }
              ]
            }
          }
        }
      }
    }
  }
}
```

Des milliards de charges de travail à grande échelle

Les collections de recherche vectorielle prennent en charge des charges de travail contenant des milliards de vecteurs. Il n'est pas nécessaire de réindexer à des fins de mise à l'échelle, car la mise à l'échelle automatique s'en charge pour vous. Si vous possédez des millions de vecteurs (ou plus) avec un grand nombre de dimensions et que vous en avez besoin de plus de 200 OCUs, contactez le AWS Support pour augmenter le nombre maximum d'unités de OpenSearch calcul (OCUs) pour votre compte.

Limites

Les collections de recherche vectorielle présentent les limites suivantes :

- Les collections de recherche vectorielle ne sont pas compatibles avec le moteur Apache Lucene ANN.
- Les collections de recherche vectorielle ne prennent en charge que l'algorithme HNSW avec Faiss et ne prennent pas en charge la FIV et l'IVFQ.
- Les collections de recherche vectorielle ne prennent pas en charge les opérations de l'API d'échauffement, de statistiques et d'entraînement des modèles.
- Les collections de recherche vectorielle ne prennent pas en charge les scripts en ligne ou stockés.
- Les informations sur le nombre d'index ne sont pas disponibles dans AWS Management Console les collections de recherche vectorielle.
- L'intervalle d'actualisation des index des collections de recherche vectorielle est de 60 secondes.

Étapes suivantes

Maintenant que vous savez comment créer une collection de recherche vectorielle et indexer des données, vous pouvez essayer certains des exercices suivants :

- Utilisez le client OpenSearch Python pour travailler avec des collections de recherche vectorielle. Consultez ce didacticiel sur GitHub.
- Utilisez le client OpenSearch Java pour travailler avec des collections de recherche vectorielle. Consultez ce didacticiel sur GitHub.
- Configuré LangChain pour être utilisé OpenSearch comme magasin de vecteurs. LangChain est un framework open source permettant de développer des applications basées sur des modèles de langage. Pour plus d'informations, consultez la documentation LangChain .

Utilisation des politiques de cycle de vie des données avec Amazon OpenSearch Serverless

Une politique de cycle de vie des données dans Amazon OpenSearch Serverless définit la durée pendant laquelle OpenSearch Serverless conserve les données dans une collection de séries chronologiques. Par exemple, vous pouvez définir une politique pour conserver les données du journal pendant 30 jours avant que OpenSearch Serverless ne les supprime.

Vous pouvez configurer une politique distincte pour chaque index de chaque collection de séries chronologiques de votre Compte AWS. OpenSearch Serverless conserve les documents pendant au moins la durée que vous spécifiez dans la politique. Il supprime ensuite les documents automatiquement dans la mesure du possible, généralement dans les 48 heures ou 10 % de la période de conservation, selon le délai le plus long.

Seules les collections de séries chronologiques prennent en charge les politiques de cycle de vie des données. Les collections de recherche et de recherche vectorielle ne le sont pas.

Rubriques

- Politiques relatives au cycle de vie
- Autorisations requises
- Priorité des stratégies
- Syntaxe d'une politique
- Création de politiques de cycle de vie des données

- Mise à jour des politiques de cycle de vie
- Supprimer les politiques de cycle de vie des données

Politiques relatives au cycle de vie

Dans une politique de cycle de vie des données, vous définissez une série de règles. La politique de cycle de vie des données vous permet de gérer la période de conservation des données associées aux index ou aux collections conformes à ces règles. Ces règles définissent la durée de conservation des données d'un index ou d'un groupe d'index. Chaque règle comprend un type de ressource (`index`), une période de rétention et une liste de ressources (index) auxquelles s'applique la période de rétention.

Vous définissez la période de conservation à l'aide de l'un des formats suivants :

- `"MinIndexRetention": "24h"`— OpenSearch Serverless conserve les données d'index pendant la période spécifiée en heures ou en jours. Vous pouvez définir cette période pour qu'elle soit comprise entre `24h` et`3650d`.
- `"NoMinIndexRetention": true`— OpenSearch Serverless conserve les données d'index indéfiniment.

Dans l'exemple de politique suivant, la première règle spécifie une période de conservation de 15 jours pour tous les index de la collection`marketing`. La deuxième règle précise que tous les noms d'index commençant par `log` dans la `finance` collection n'ont pas de période de conservation définie et seront conservés indéfiniment.

```
{
   "lifeCyclePolicyDetail": {
      "type": "retention",
      "name": "my-policy",
      "policyVersion": "MTY4ODI0NTM2OTk1N18x",
      "policy": {
         "Rules": [
            {
            "ResourceType":"index",
            "Resource":[
               "index/marketing/*"
            ],
            "MinIndexRetention": "15d"
         },
```

```
         {
            "ResourceType":"index",
            "Resource":[
               "index/finance/log*"
            ],
            "NoMinIndexRetention": true
         }
         ]
      },
      "createdDate": 1688245369957,
      "lastModifiedDate": 1688245369957
   }
}
```

Dans l'exemple de règle de politique suivant, OpenSearch Serverless conserve indéfiniment les données de tous les index pour toutes les collections du compte.

```
{
   "Rules": [
      {
         "ResourceType": "index",
         "Resource": [
            "index/*/*"
         ]
      }
   ],
   "NoMinIndexRetention": true
}
```

Autorisations requises

Les politiques de cycle de vie pour OpenSearch Serverless utilisent les autorisations AWS Identity and Access Management (IAM) suivantes. Vous pouvez définir des conditions IAM pour limiter les utilisateurs aux politiques de cycle de vie des données associées à des collections et à des index spécifiques.

- `aoss:CreateLifecyclePolicy`— Créez une politique de cycle de vie des données.
- `aoss:ListLifecyclePolicies`— Répertoriez toutes les politiques relatives au cycle de vie des données du compte courant.
- `aoss:BatchGetLifecyclePolicy`— Consultez une politique de cycle de vie des données associée à un compte ou à un nom de politique.

- `aoss:BatchGetEffectiveLifecyclePolicy`— Afficher une politique de cycle de vie des données pour une ressource donnée (`indexc`'est la seule ressource prise en charge).
- `aoss:UpdateLifecyclePolicy`— Modifiez une politique de cycle de vie des données donnée et modifiez son paramètre ou sa ressource de rétention.
- `aoss:DeleteLifecyclePolicy`— Supprimez une politique de cycle de vie des données.

La politique d'accès basée sur l'identité suivante permet à un utilisateur de consulter toutes les politiques relatives au cycle de vie des données et de mettre à jour les politiques en fonction du modèle de ressources : `collection/application-logs`

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aoss:UpdateLifecyclePolicy"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aoss:collection": "application-logs"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "aoss:ListLifecyclePolicies",
                "aoss:BatchGetLifecyclePolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

Priorité des stratégies

Dans certaines situations, les règles relatives au cycle de vie des données se chevauchent, au sein des politiques ou entre celles-ci. Dans ce cas, une règle avec un nom de ressource ou un modèle de ressource plus spécifique pour un index remplace une règle avec un nom de ressource ou un modèle de ressource plus général pour tous les index communs aux deux règles.

Par exemple, dans la politique suivante, deux règles s'appliquent à un index`index/sales/logstash`. Dans ce cas, la deuxième règle a priorité car elle `index/sales/log*` correspond le plus longtemps à`index/sales/logstash`. Par conséquent, OpenSearch Serverless ne définit aucune période de rétention pour l'index.

```
{
      "Rules":[
         {
            "ResourceType":"index",
            "Resource":[
               "index/sales/*",
            ],
            "MinIndexRetention": "15d"
         },
         {
            "ResourceType":"index",
            "Resource":[
               "index/sales/log*",
            ],
            "NoMinIndexRetention": true
         }
      ]
   }
```

Syntaxe d’une politique

Fournissez une ou plusieurs règles. Ces règles définissent les paramètres du cycle de vie des données pour vos index OpenSearch sans serveur.

Chaque règle contient les éléments suivants. Vous pouvez fournir `MinIndexRetention` ou `NoMinIndexRetention` dans chaque règle, mais pas les deux.

Element	Description
Type de ressource	Type de ressource auquel s'applique la règle. La seule option prise en charge pour les politiques de cycle de vie des données est`index`.
Ressource	Une liste de and/or modèles de noms de ressources. Les modèles se composent d'un préfixe et d'un caractère générique (*), qui permettent aux autorisations associées de s'appliquer à plusieurs ressources. Par exemple, `index/`*`<collection-name|pattern>`* `/`*`<index-name|pattern>`* .
MinIndexRetention	Période minimale, en jours (d) ou heures (h), pour conserver le document dans l'index. La limite inférieure est `24h` et la limite supérieure est`3650d`.
NoMinIndexRetention	Si`true`, OpenSearch Serverless conserve les documents indéfiniment.

Dans l'exemple suivant, la première règle s'applique à tous les index du `autoparts-inventory` modèle (`index/autoparts-inventory/*`) et exige que les données soient conservées pendant au moins 20 jours avant que toute action, telle que la suppression ou l'archivage, ne puisse avoir lieu.

La deuxième règle cible les index correspondant au `auto*/gear` modèle (`index/auto*/gear`), en fixant une période de conservation minimale de 24 heures.

La troisième règle s'applique spécifiquement à l'`tires`index et ne prévoit aucune durée de conservation minimale, ce qui signifie que les données de cet index peuvent être supprimées ou archivées immédiatement ou en fonction d'autres critères. Ces règles permettent de gérer la conservation des données d'index avec des durées de conservation variables ou sans restrictions de conservation.

```
{
```

```
  "Rules": [
    {
      "ResourceType": "index",
      "Resource": [
        "index/autoparts-inventory/*"
      ],
      "MinIndexRetention": "20d"
    },
    {
      "ResourceType": "index",
      "Resource": [
        "index/auto*/gear"
      ],
      "MinIndexRetention": "24h"
    },
    {
      "ResourceType": "index",
      "Resource": [
        "index/autoparts-inventory/tires"
      ],
      "NoMinIndexRetention": true
    }
  ]
}
```

Création de politiques de cycle de vie des données

Pour créer une politique de cycle de vie des données, vous définissez des règles qui gèrent la conservation et la suppression de vos données en fonction de critères spécifiques.

console

Pour créer une politique de cycle de vie des données

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, sélectionnez Data Lifecycle policies.
3. Choisissez Créer une politique de cycle de vie des données.
4. Entrez un nom descriptif pour la politique.
5. Pour le cycle de vie des données, choisissez Ajouter et sélectionnez les collections et les index pour la politique.

Commencez par choisir les collections auxquelles appartiennent les index. Choisissez ensuite l'index dans la liste ou entrez un modèle d'index. Pour sélectionner toutes les collections comme sources, entrez un astérisque (*).

6. Pour la conservation des données, vous pouvez choisir de conserver les données indéfiniment ou de désélectionner Illimité (ne jamais supprimer) et de spécifier une période après laquelle OpenSearch Serverless supprime automatiquement les données d'Amazon S3.
7. Choisissez Enregistrer, puis Créer.

AWS CLI

Pour créer une politique de cycle de vie des données à l'aide de AWS CLI, utilisez la create-lifecycle-policycommande avec les options suivantes :

- `--name`— Le nom de la politique.
- `--type`— Le type de politique. Actuellement, la seule valeur disponible est`retention`.
- `--policy`— La politique de cycle de vie des données. Ce paramètre accepte à la fois les politiques intégrées et les fichiers .json. Vous devez encoder les politiques intégrées sous forme de chaîne JSON échappée. Pour fournir la politique dans un fichier, utilisez le format`--policy file://my-policy.json`.

Example

```
aws opensearchserverless create-lifecycle-policy \
  --name my-policy \
  --type retention \
  --policy "{\"Rules\":[{\"ResourceType\":\"index\",\"Resource\":[\"index/autoparts-
inventory/*\"],\"MinIndexRetention\": \"81d\"},{\"ResourceType\":\"index\",\"Resource
\":[\"index/sales/orders*\"],\"NoMinIndexRetention\":true}]}"
```

Mise à jour des politiques de cycle de vie

Pour mettre à jour une politique de cycle de vie des données, vous pouvez modifier les règles existantes afin de tenir compte de l'évolution de vos exigences en matière de conservation ou de suppression des données. Cela vous permet d'adapter vos politiques en fonction de l'évolution de vos besoins en matière de gestion des données.

Il peut s'écouler quelques minutes entre le moment où vous mettez à jour la politique et le moment où OpenSearch Serverless commence à appliquer les nouvelles périodes de rétention.

console

Pour mettre à jour une politique de cycle de vie des données

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, sélectionnez Data Lifecycle policies.
3. Sélectionnez la politique de cycle de vie des données que vous souhaitez mettre à jour, puis choisissez Modifier.
4. Modifiez la politique à l'aide de l'éditeur visuel ou de l'éditeur JSON.
5. Choisissez Enregistrer.

AWS CLI

Pour mettre à jour une politique de cycle de vie des données à l'aide de AWS CLI, utilisez la update-lifecycle-policycommande.

Vous devez inclure le `--policy-version` paramètre dans la demande. Vous pouvez récupérer la version de stratégie à l'aide des commandes list-lifecycle-policies ou batch-get-lifecycle-policy. Nous vous recommandons d'inclure la version de politique la plus récente pour éviter de remplacer accidentellement les modifications apportées par d'autres.

La demande suivante met à jour une politique de cycle de vie des données avec un nouveau document JSON de politique.

Example

```
aws opensearchserverless update-lifecycle-policy \
  --name my-policy \
  --type retention \
  --policy-version MTY2MzY5MTY1MDA3Ml8x \
  --policy file://my-new-policy.json
```

Supprimer les politiques de cycle de vie des données

Lorsque vous supprimez une politique de cycle de vie des données, OpenSearch Serverless ne l'applique plus aux index correspondants.

console

Pour supprimer une politique de cycle de vie des données

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, sélectionnez Data Lifecycle policies.
3. Sélectionnez la politique que vous souhaitez supprimer, puis choisissez Supprimer et confirmez la suppression.

AWS CLI

Pour supprimer une politique de cycle de vie des données à l'aide de AWS CLI, utilisez la delete-lifecycle-policycommande.

Example

```
aws opensearchserverless delete-lifecycle-policy \
  --name my-policy \
  --type retention
```

Utilisation du AWS SDKs pour interagir avec Amazon OpenSearch Serverless

Cette section inclut des exemples d'utilisation AWS SDKs d'Amazon OpenSearch sans serveur. Ces exemples de code montrent comment créer des stratégies de sécurité et des collections et comment interroger des collections.

> **Note**
>
> Nous sommes en train de créer ces exemples de code. Si vous souhaitez contribuer à un exemple de code (Java, Go, etc.), veuillez ouvrir une demande d'extraction directement dans le GitHub référentiel.

Rubriques

- Python
- JavaScript

Python

L'exemple de script suivant utilise le client AWS SDK pour Python (Boto3), ainsi que le client opensearch-py pour Python, pour créer des stratégies de chiffrement, réseau et d'accès aux données, créer une collection correspondante et indexer des exemples de données.

Pour installer les dépendances requises, exécutez les commandes suivantes :

```
pip install opensearch-py
pip install boto3
pip install botocore
pip install requests-aws4auth
```

Dans le script, remplacez l'élément `Principal` par l'Amazon Resource Name (ARN) de l'utilisateur ou du rôle qui signe la requête. Vous pouvez également modifier la `region`.

```
from opensearchpy import OpenSearch, RequestsHttpConnection
from requests_aws4auth import AWS4Auth
import boto3
import botocore
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

client = boto3.client('opensearchserverless')
service = 'aoss'
region = 'us-east-1'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key,
                   region, service, session_token=credentials.token)


def createEncryptionPolicy(client):
    """Creates an encryption policy that matches all collections beginning with tv-"""
    try:
        response = client.create_security_policy(
            description='Encryption policy for TV collections',
            name='tv-policy',
            policy="""
                {
                    \"Rules\":[
```

```
                        {
                            \"ResourceType\":\"collection\",
                            \"Resource\":[
                                \"collection\/tv-*\"
                            ]
                        }
                    ],
                    \"AWSOwnedKey\":true
                }
                """,
            type='encryption'
        )
        print('\nEncryption policy created:')
        print(response)
    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ConflictException':
            print(
                '[ConflictException] The policy name or rules conflict with an existing
 policy.')
        else:
            raise error


def createNetworkPolicy(client):
    """Creates a network policy that matches all collections beginning with tv-"""
    try:
        response = client.create_security_policy(
            description='Network policy for TV collections',
            name='tv-policy',
            policy="""
                [{
                    \"Description\":\"Public access for TV collection\",
                    \"Rules\":[
                        {
                            \"ResourceType\":\"dashboard\",
                            \"Resource\":[\"collection\/tv-*\"]
                        },
                        {
                            \"ResourceType\":\"collection\",
                            \"Resource\":[\"collection\/tv-*\"]
                        }
                    ],
                    \"AllowFromPublic\":true
                }]
```

```
                """,
            type='network'
        )
        print('\nNetwork policy created:')
        print(response)
    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ConflictException':
            print(
                '[ConflictException] A network policy with this name already exists.')
        else:
            raise error


def createAccessPolicy(client):
    """Creates a data access policy that matches all collections beginning with tv-"""
    try:
        response = client.create_access_policy(
            description='Data access policy for TV collections',
            name='tv-policy',
            policy="""
                [{
                    \"Rules\":[
                        {
                            \"Resource\":[
                                \"index\/tv-*\/*\"
                            ],
                            \"Permission\":[
                                \"aoss:CreateIndex\",
                                \"aoss:DeleteIndex\",
                                \"aoss:UpdateIndex\",
                                \"aoss:DescribeIndex\",
                                \"aoss:ReadDocument\",
                                \"aoss:WriteDocument\"
                            ],
                            \"ResourceType\": \"index\"
                        },
                        {
                            \"Resource\":[
                                \"collection\/tv-*\"
                            ],
                            \"Permission\":[
                                \"aoss:CreateCollectionItems\"
                            ],
                            \"ResourceType\": \"collection\"
```

```
                        }
                    ],
                    \"Principal\":[
                        \"arn:aws:iam::123456789012:role\/Admin\"
                    ]
                }]
                """,
            type='data'
        )
        print('\nAccess policy created:')
        print(response)
    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ConflictException':
            print(
                '[ConflictException] An access policy with this name already exists.')
        else:
            raise error


def createCollection(client):
    """Creates a collection"""
    try:
        response = client.create_collection(
            name='tv-sitcoms',
            type='SEARCH'
        )
        return(response)
    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ConflictException':
            print(
                '[ConflictException] A collection with this name already exists. Try
 another name.')
        else:
            raise error


def waitForCollectionCreation(client):
    """Waits for the collection to become active"""
    response = client.batch_get_collection(
        names=['tv-sitcoms'])
    # Periodically check collection status
    while (response['collectionDetails'][0]['status']) == 'CREATING':
        print('Creating collection...')
        time.sleep(30)
```

```
        response = client.batch_get_collection(
            names=['tv-sitcoms'])
    print('\nCollection successfully created:')
    print(response["collectionDetails"])
    # Extract the collection endpoint from the response
    host = (response['collectionDetails'][0]['collectionEndpoint'])
    final_host = host.replace("https://", "")
    indexData(final_host)


def indexData(host):
    """Create an index and add some sample data"""
    # Build the OpenSearch client
    client = OpenSearch(
        hosts=[{'host': host, 'port': 443}],
        http_auth=awsauth,
        use_ssl=True,
        verify_certs=True,
        connection_class=RequestsHttpConnection,
        timeout=300
    )
    # It can take up to a minute for data access rules to be enforced
    time.sleep(45)

    # Create index
    response = client.indices.create('sitcoms-eighties')
    print('\nCreating index:')
    print(response)

    # Add a document to the index.
    response = client.index(
        index='sitcoms-eighties',
        body={
            'title': 'Seinfeld',
            'creator': 'Larry David',
            'year': 1989
        },
        id='1',
    )
    print('\nDocument added:')
    print(response)


def main():
```

```
    createEncryptionPolicy(client)
    createNetworkPolicy(client)
    createAccessPolicy(client)
    createCollection(client)
    waitForCollectionCreation(client)


if __name__ == "__main__":
    main()
```

JavaScript

L'exemple de script suivant utilise le kit SDK pour JavaScript Node.js, ainsi que le client opensearch-js pour JavaScript, pour créer des stratégies de chiffrement, réseau et d'accès aux données, créer une collection correspondante, créer un index et indexer des exemples de données.

Pour installer les dépendances requises, exécutez les commandes suivantes :

```
npm i aws-sdk
npm i aws4
npm i @opensearch-project/opensearch
```

Dans le script, remplacez l'élément `Principal` par l'Amazon Resource Name (ARN) de l'utilisateur ou du rôle qui signe la requête. Vous pouvez également modifier la `region`.

```
var AWS = require('aws-sdk');
var aws4 = require('aws4');
var {
    Client,
    Connection
} = require("@opensearch-project/opensearch");
var {
    OpenSearchServerlessClient,
    CreateSecurityPolicyCommand,
    CreateAccessPolicyCommand,
    CreateCollectionCommand,
    BatchGetCollectionCommand
} = require("@aws-sdk/client-opensearchserverless");
var client = new OpenSearchServerlessClient();

async function execute() {
    await createEncryptionPolicy(client)
```

```
    await createNetworkPolicy(client)
    await createAccessPolicy(client)
    await createCollection(client)
    await waitForCollectionCreation(client)
}

async function createEncryptionPolicy(client) {
    // Creates an encryption policy that matches all collections beginning with 'tv-'
    try {
        var command = new CreateSecurityPolicyCommand({
            description: 'Encryption policy for TV collections',
            name: 'tv-policy',
            type: 'encryption',
            policy: " \
        { \
            \"Rules\":[ \
                { \
                    \"ResourceType\":\"collection\", \
                    \"Resource\":[ \
                        \"collection\/tv-*\" \
                    ] \
                } \
            ], \
            \"AWSOwnedKey\":true \
        }"
        });
        const response = await client.send(command);
        console.log("Encryption policy created:");
        console.log(response['securityPolicyDetail']);
    } catch (error) {
        if (error.name === 'ConflictException') {
            console.log('[ConflictException] The policy name or rules conflict with an
 existing policy.');
        } else
            console.error(error);
    };
}

async function createNetworkPolicy(client) {
    // Creates a network policy that matches all collections beginning with 'tv-'
    try {
        var command = new CreateSecurityPolicyCommand({
            description: 'Network policy for TV collections',
            name: 'tv-policy',
```

```
            type: 'network',
            policy: " \
            [{ \
                \"Description\":\"Public access for television collection\", \
                \"Rules\":[ \
                    { \
                        \"ResourceType\":\"dashboard\", \
                        \"Resource\":[\"collection\/tv-*\"] \
                    }, \
                    { \
                        \"ResourceType\":\"collection\", \
                        \"Resource\":[\"collection\/tv-*\"] \
                    } \
                ], \
                \"AllowFromPublic\":true \
            }]"
        });
        const response = await client.send(command);
        console.log("Network policy created:");
        console.log(response['securityPolicyDetail']);
    } catch (error) {
        if (error.name === 'ConflictException') {
            console.log('[ConflictException] A network policy with that name already
 exists.');
        } else
            console.error(error);
    };
}

async function createAccessPolicy(client) {
    // Creates a data access policy that matches all collections beginning with 'tv-'
    try {
        var command = new CreateAccessPolicyCommand({
            description: 'Data access policy for TV collections',
            name: 'tv-policy',
            type: 'data',
            policy: " \
            [{ \
                \"Rules\":[ \
                    { \
                        \"Resource\":[ \
                            \"index\/tv-*\/*\" \
                        ], \
                        \"Permission\":[ \
```

```
                            \"aoss:CreateIndex\", \
                            \"aoss:DeleteIndex\", \
                            \"aoss:UpdateIndex\", \
                            \"aoss:DescribeIndex\", \
                            \"aoss:ReadDocument\", \
                            \"aoss:WriteDocument\" \
                        ], \
                        \"ResourceType\": \"index\" \
                    }, \
                    { \
                        \"Resource\":[ \
                            \"collection\/tv-*\" \
                        ], \
                        \"Permission\":[ \
                            \"aoss:CreateCollectionItems\" \
                        ], \
                        \"ResourceType\": \"collection\" \
                    } \
                ], \
                \"Principal\":[ \
                    \"arn:aws:iam::123456789012:role\/Admin\" \
                ] \
            }]"
        });
        const response = await client.send(command);
        console.log("Access policy created:");
        console.log(response['accessPolicyDetail']);
    } catch (error) {
        if (error.name === 'ConflictException') {
            console.log('[ConflictException] An access policy with that name already
 exists.');
        } else
            console.error(error);
    };
}

async function createCollection(client) {
    // Creates a collection to hold TV sitcoms indexes
    try {
        var command = new CreateCollectionCommand({
            name: 'tv-sitcoms',
            type: 'SEARCH'
        });
        const response = await client.send(command);
```

```
        return (response)
    } catch (error) {
        if (error.name === 'ConflictException') {
            console.log('[ConflictException] A collection with this name already
 exists. Try another name.');
        } else
            console.error(error);
    };
}

async function waitForCollectionCreation(client) {
    // Waits for the collection to become active
    try {
        var command = new BatchGetCollectionCommand({
            names: ['tv-sitcoms']
        });
        var response = await client.send(command);
        while (response.collectionDetails[0]['status'] == 'CREATING') {
            console.log('Creating collection...')
            await sleep(30000) // Wait for 30 seconds, then check the status again
            function sleep(ms) {
                return new Promise((resolve) => {
                    setTimeout(resolve, ms);
                });
            }
            var response = await client.send(command);
        }
        console.log('Collection successfully created:');
        console.log(response['collectionDetails']);
        // Extract the collection endpoint from the response
        var host = (response.collectionDetails[0]['collectionEndpoint'])
        // Pass collection endpoint to index document request
        indexDocument(host)
    } catch (error) {
        console.error(error);
    };
}

async function indexDocument(host) {

    var client = new Client({
        node: host,
        Connection: class extends Connection {
            buildRequestObject(params) {
```

```
                var request = super.buildRequestObject(params)
                request.service = 'aoss';
                request.region = 'us-east-1'; // e.g. us-east-1
                var body = request.body;
                request.body = undefined;
                delete request.headers['content-length'];
                request.headers['x-amz-content-sha256'] = 'UNSIGNED-PAYLOAD';
                request = aws4.sign(request, AWS.config.credentials);
                request.body = body;

                return request
            }
        }
    });

    // Create an index
    try {
        var index_name = "sitcoms-eighties";

        var response = await client.indices.create({
            index: index_name
        });

        console.log("Creating index:");
        console.log(response.body);

        // Add a document to the index
        var document = "{ \"title\": \"Seinfeld\", \"creator\": \"Larry David\", \"year
\": \"1989\" }\n";

        var response = await client.index({
            index: index_name,
            body: document
        });

        console.log("Adding document:");
        console.log(response.body);
    } catch (error) {
        console.error(error);
    };
}

execute()
```

Utilisation AWS CloudFormation pour créer des collections Amazon OpenSearch Serverless

Vous pouvez utiliser AWS CloudFormation pour créer des ressources Amazon OpenSearch sans serveur telles que des collections, des stratégies de sécurité et des points de terminaison d'un VPC. Pour une CloudFormation référence complète à OpenSearch Serverless, consultez Amazon OpenSearch Serverless dans le guide de l'AWS CloudFormation utilisateur.

L'exemple de CloudFormation modèle suivant crée une stratégie d'accès aux données, une stratégie réseau et une stratégie de sécurité simples, ainsi qu'une collection correspondante. Il s'agit d'une bonne méthode pour démarrer rapidement avec Amazon OpenSearch sans serveur et allouer les éléments nécessaires à la création et à l'utilisation d'une collection.

> **Important**
>
> Cet exemple utilise l'accès au réseau public, ce qui n'est pas recommandé pour les charges de travail de production. Nous vous recommandons d'utiliser l'accès VPC pour protéger vos collections. Pour plus d'informations, consultez AWS::OpenSearchServerless::VpcEndpoint et the section called “Points de terminaison d’un VPC”.

```
AWSTemplateFormatVersion: 2010-09-09
Description: 'Amazon OpenSearch Serverless template to create an IAM user, encryption
 policy, data access policy and collection'
Resources:
  IAMUSer:
    Type: 'AWS::IAM::User'
    Properties:
      UserName:  aossadmin
  DataAccessPolicy:
    Type: 'AWS::OpenSearchServerless::AccessPolicy'
    Properties:
      Name: quickstart-access-policy
      Type: data
      Description: Access policy for quickstart collection
      Policy: !Sub >-
        [{"Description":"Access for cfn user","Rules":
[{"ResourceType":"index","Resource":["index/*/*"],"Permission":["aoss:*"]},
        {"ResourceType":"collection","Resource":["collection/quickstart"],"Permission":
["aoss:*"]}],
```

```
        "Principal":["arn:aws:iam::${AWS::AccountId}:user/aossadmin"]}]
  NetworkPolicy:
    Type: 'AWS::OpenSearchServerless::SecurityPolicy'
    Properties:
      Name: quickstart-network-policy
      Type: network
      Description: Network policy for quickstart collection
      Policy: >-
        [{"Rules":[{"ResourceType":"collection","Resource":["collection/
quickstart"]}, {"ResourceType":"dashboard","Resource":["collection/
quickstart"]}],"AllowFromPublic":true}]
  EncryptionPolicy:
    Type: 'AWS::OpenSearchServerless::SecurityPolicy'
    Properties:
      Name: quickstart-security-policy
      Type: encryption
      Description: Encryption policy for quickstart collection
      Policy: >-
        {"Rules":[{"ResourceType":"collection","Resource":["collection/
quickstart"]}],"AWSOwnedKey":true}
  Collection:
    Type: 'AWS::OpenSearchServerless::Collection'
    Properties:
      Name: quickstart
      Type: TIMESERIES
      Description: Collection to holds timeseries data
    DependsOn: EncryptionPolicy
Outputs:
  IAMUser:
    Value: !Ref IAMUSer
  DashboardURL:
    Value: !GetAtt Collection.DashboardEndpoint
  CollectionARN:
    Value: !GetAtt Collection.Arn
```

Gestion des limites de capacité pour Amazon OpenSearch Serverless

Avec Amazon OpenSearch Serverless, vous n'avez pas à gérer vous-même la capacité. OpenSearch Serverless adapte automatiquement la capacité de calcul de votre compte en fonction de la charge de travail actuelle. La capacité de calcul sans serveur est mesurée en unités OpenSearch de calcul

(OCUs). Chaque OCU est une combinaison de 6 Gio de mémoire et du processeur virtuel (vCPU) correspondant et crée un transfert de données vers Amazon S3. Pour plus d'informations sur l'architecture découplée dans OpenSearch Serverless, consultez. the section called “Comment ça marche”

Lorsque vous créez votre première collection, OpenSearch Serverless instancie en OCUs fonction de vos paramètres de redondance. Par défaut, les répliques actives redondantes sont activées, ce qui signifie qu'un total de quatre OCUs sont instanciées (deux pour l'indexation et deux pour la recherche) afin de garantir une haute disponibilité avec des nœuds de secours dans une autre zone de disponibilité. À des fins de développement et de test, vous pouvez désactiver le paramètre Activer la redondance pour une collection, ce qui élimine les répliques de secours et n'en instancie que deux OCUs (une pour l'indexation et une pour la recherche). Ils existent OCUs toujours, même en l'absence d'indexation ou d'activité de recherche. Toutes les collections suivantes peuvent les partager OCUs (à l'exception des collections avec des AWS KMS clés uniques, qui instancient leur propre ensemble de OCUs clés). Si nécessaire, OpenSearch Serverless évolue automatiquement et ajoute des éléments supplémentaires à OCUs mesure que votre utilisation de l'indexation et de la recherche augmente. Lorsque le trafic sur votre point de terminaison de collecte diminue, la capacité est réduite au minimum OCUs requis pour la taille de vos données. Pour la recherche et la collecte de séries chronologiques, le nombre de données OCUs requises en cas d'inactivité est proportionnel à la taille des données et au nombre d'index. Pour les vecteurs, cela dépend à la fois de la mémoire (RAM) pour stocker les graphes vectoriels et de l'espace disque pour stocker les indices. S'il n'est pas en état d'inactivité, les exigences de l'OCU tiennent compte de ces deux facteurs.

Les collections vectorielles conservent les données d'index dans le stockage local de l'OCU. Les limites de RAM OCU sont atteintes plus rapidement que les limites de disque OCU, ce qui limite l'espace RAM des collections de vecteurs. Tout au plus, il sera réduit à 1 OCU [0,5 OCU x 2] pour l'indexation et à 1 OCU [0,5 OCU x 2] pour la recherche. La mise à l'échelle prend également en compte le nombre de partitions nécessaires à votre collection ou à votre index. Chaque OCU peut prendre en charge un certain nombre de partitions. Le nombre d'index doit être proportionnel au nombre de partitions. Le nombre total de bases OCUs requises correspond à la quantité maximale de données, de mémoire et de partitions requise. Pour plus d'informations, consultez les fonctionnalités de recherche économiques d'Amazon OpenSearch Serverless, à n'importe quelle échelle, sur le blog AWS Big Data.

Pour les collections de recherche et de recherche vectorielle, toutes les données sont stockées sur des index actifs afin de garantir des temps de réponse rapides aux requêtes. Les collections de séries chronologiques utilisent une combinaison de stockage à chaud et à chaud, ce qui permet de conserver les données les plus récentes dans un stockage à chaud afin d'optimiser les temps

de réponse aux requêtes pour les données les plus fréquemment consultées. Pour de plus amples informations, veuillez consulter the section called “Choix d'un type de collection”.

> **Note**
>
> Une collection de recherche vectorielle ne peut pas être partagée OCUs avec des collections de recherche et de séries chronologiques, même si la collection de recherche vectorielle utilise la même clé KMS que les collections de recherche ou de séries chronologiques. Un nouvel ensemble de OCUs sera créé pour votre première collection de vecteurs. Les collections OCUs de vecteurs sont partagées entre les mêmes collections de clés KMS.

Pour gérer la capacité de vos collections et contrôler les coûts, vous pouvez spécifier la capacité maximale globale d'indexation et de recherche pour le compte courant et la région, et OpenSearch Serverless adapte automatiquement vos ressources de collecte en fonction de ces spécifications.

Étant donné que les capacités d'indexation et de recherche se mettent à l'échelle séparément, vous devez définir des limites au niveau du compte pour chacune :

- Capacité d'indexation maximale — OpenSearch Serverless peut augmenter la capacité d'indexation jusqu'à ce nombre de. OCUs
- Capacité de recherche maximale — OpenSearch Serverless peut augmenter la capacité de recherche jusqu'à ce nombre de OCUs.

> **Note**
>
> À l'heure actuelle, les paramètres de capacité ne s'appliquent qu'au niveau du compte. Vous ne pouvez pas configurer de limites de capacité par collection.

Votre objectif doit être de vous assurer que la capacité maximale est suffisamment élevée pour gérer les pics de charge de travail. En fonction de vos paramètres, OpenSearch Serverless augmente automatiquement le nombre de collections OCUs pour traiter la charge de travail d'indexation et de recherche.

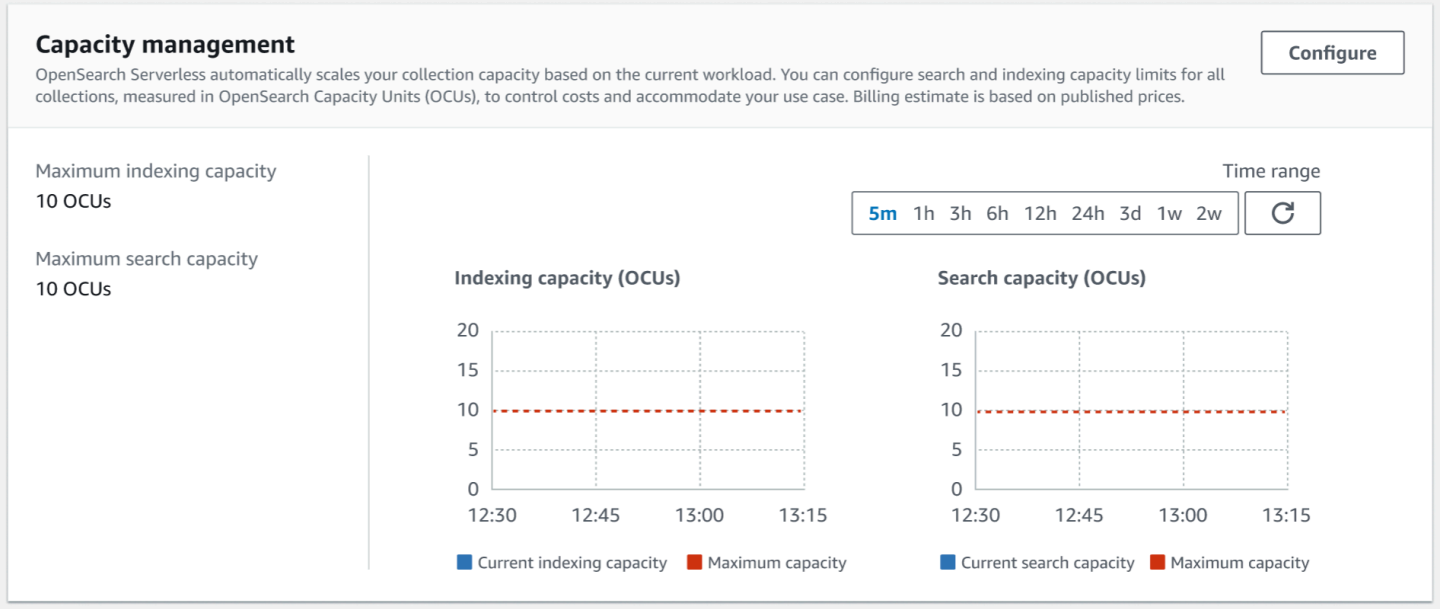
Rubriques

- Configurer les paramètres de capacité

- Limites de capacité maximale
- Surveiller l'utilisation de la capacité

Configurer les paramètres de capacité

Pour configurer les paramètres de capacité dans la console OpenSearch Serverless, développez Serverless dans le volet de navigation de gauche et sélectionnez Dashboard. Spécifiez la capacité maximale d'indexation et de recherche sous Capacity management (Gestion de la capacité) :



Pour configurer la capacité à l'aide du AWS CLI, envoyez une UpdateAccountSettingsdemande :

```
aws opensearchserverless update-account-settings \
    --capacity-limits '{ "maxIndexingCapacityInOCU": 8,"maxSearchCapacityInOCU": 9 }'
```

Limites de capacité maximale

Le nombre maximum d'index qu'une collection peut contenir est de 1 000. Pour les trois types de collections, la capacité maximale par défaut de l'OCU est de 10 OCUs pour l'indexation et de 10 OCUs pour la recherche. La capacité OCU minimale autorisée pour un compte est de 1 OCU [0,5 OCU x 2] pour l'indexation et de 1 OCU [0,5 OCU x 2] pour la recherche. Pour toutes les collections, la capacité maximale autorisée est de 1 700 OCUs pour l'indexation et de 1 700 OCUs pour la recherche. Vous pouvez configurer le nombre d'OCU pour qu'il soit compris entre 1 et la capacité maximale autorisée, par multiples de 2.

Chaque OCU inclut suffisamment de stockage éphémère à chaud pour 120 GiB de données d'index. OpenSearch Serverless prend en charge jusqu'à 1 TiB de données par index dans les collections de recherche et de recherche vectorielle, et 100 TiB de données chaudes par index dans une collection de séries chronologiques. Pour les collections de séries chronologiques, vous pouvez toujours ingérer davantage de données, qui peuvent être stockées sous forme de données chaudes dans S3.

Pour obtenir la liste de tous les quotas, consultez la section Quotas OpenSearch sans serveur.

Surveiller l'utilisation de la capacité

Vous pouvez surveiller `Search0CU` les CloudWatch indicateurs `Indexing0CU` au niveau du compte pour comprendre l'évolution de vos collections. Nous vous recommandons de définir des alertes qui vous avertissent si votre compte se rapproche d'un seuil pour les métriques liées à la capacité, afin que vous puissiez ajuster vos paramètres de capacité en conséquence.

Vous pouvez également utiliser ces métriques pour déterminer si les paramètres de capacité maximale sont appropriés ou si vous devez les ajuster. Analysez ces métriques afin de concentrer vos efforts sur l'optimisation de l'efficacité de vos collections. Pour plus d'informations sur les métriques auxquelles OpenSearch Serverless envoie CloudWatch, consultezthe section called “Surveillance OpenSearch sans serveur”.

Ingérer des données dans les collections Amazon OpenSearch sans serveur

Ces sections fournissent des informations sur les pipelines d'ingestion pris en charge pour l'ingestion de données dans les collections Amazon OpenSearch sans serveur. Elles mentionnent également certains des clients que vous pouvez utiliser pour interagir avec les opérations OpenSearch d'API. Vos clients doivent être compatibles avec OpenSearch 2.x pour s'intégrer à sans OpenSearch serveur.

Rubriques

- Autorisations minimales requises
- OpenSearch Ingestion
- Fluent Bit
- Amazon Data Firehose
- Go

- Java
- JavaScript
- Logstash
- Python
- Ruby
- Signature des demandes HTTP avec d'autres clients

Autorisations minimales requises

Pour intégrer des données dans une collection OpenSearch sans serveur, le principal qui écrit les données doit disposer des autorisations minimales suivantes, attribuées dans une stratégie d'accès aux données :

```
[
   {
      "Rules":[
         {
            "ResourceType":"index",
            "Resource":[
               "index/target-collection/logs"
            ],
            "Permission":[
               "aoss:CreateIndex",
               "aoss:WriteDocument",
               "aoss:UpdateIndex"
            ]
         }
      ],
      "Principal":[
         "arn:aws:iam::123456789012:user/my-user"
      ]
   }
]
```

Les autorisations peuvent être plus étendues si vous prévoyez d'écrire dans des index supplémentaires. Par exemple, plutôt que de spécifier un seul index cible, vous pouvez autoriser l'accès à tous les index (index/ *target-collection* /*) ou à un sous-ensemble d'index (index//). *target-collection logs**

Pour obtenir une référence de toutes les opérations d' OpenSearch API disponibles et des autorisations qui leur sont associées, veuillez consulter la rubriquethe section called “Opérations et plugins pris en charge”.

OpenSearch Ingestion

Plutôt que d'utiliser un client tiers pour envoyer des données directement à une collection OpenSearch sans serveur, vous pouvez utiliser Amazon OpenSearch Ingestion. Vous configurez vos producteurs de données pour envoyer les données OpenSearch à Ingérer, qui remet automatiquement les données à la collection que vous spécifiez. Vous pouvez également configurer OpenSearch Ingérer pour transformer vos données avant de les remettre. Pour de plus amples informations, veuillez consulter OpenSearch Ingestion d'Amazon.

Un pipeline d' OpenSearch ingestion a besoin d'une autorisation pour écrire dans une collection OpenSearch sans serveur configurée comme récepteur. Ces autorisations offrent la possibilité de décrire la collection et d'envoyer des demandes HTTP à celle-ci. Pour obtenir des instructions sur OpenSearch l'utilisation d'Ingestion pour ajouter des données à une collection, reportez-vous àthe section called “Autoriser les pipelines à accéder aux collections”.

Pour commencer à utiliser OpenSearch Ingestion, voirthe section called “Tutoriel : Ingérer des données dans une collection”.

Fluent Bit

Vous pouvez utiliser AWS for Fluent Bit image (français non disponible OpenSearch) pour ingérer des données dans des collections OpenSearch sans serveur.

> **Note**
>
> Vous devez disposer de la version 2.30.0 ou ultérieure d' AWS pour l'image Fluent Bit pour pouvoir intégrer à OpenSearch sans serveur.

Exemple de configuration :

Cet exemple de section de sortie du fichier de configuration montre comment utiliser une collection OpenSearch sans serveur en tant que destination. L'ajout important est le paramètre `AWS_Service_Name`, qui est `aoss`. `Host` est le point de terminaison de la collection.

```
[OUTPUT]
```

```
    Name  opensearch
    Match *
    Host  collection-endpoint.us-west-2.aoss.amazonaws.com
    Port  443
    Index  my_index
    Trace_Error On
    Trace_Output On
    AWS_Auth On
    AWS_Region <region>
    AWS_Service_Name aoss
    tls     On
    Suppress_Type_Name On
```

Amazon Data Firehose

Firehose prend en charge OpenSearch sans serveur comme destination de livraison. Pour connaître la procédure à suivre pour envoyer des données dans OpenSearch sans serveur, veuillez consulter les rubriques Création d'un flux de diffusion Kinesis Data Firehose et OpenSearch Choose Serverless for Your Destination (Choisir sans serveur comme destination) dans le Guide du développeur Amazon Data Firehose.

Le rôle IAM que vous fournissez à Firehose pour la livraison doit être spécifié dans une stratégie d'accès aux données avec `aoss:WriteDocument` l'autorisation minimale pour la collection cible. Vous devez avoir un index préexistant vers lequel envoyer des données. Pour de plus amples informations, veuillez consulter the section called “Autorisations minimales requises”.

Avant d'envoyer des données vers OpenSearch sans serveur, il se peut que vous devriez effectuer des transformations sur ces données. Pour en savoir plus sur l'utilisation des fonctions Lambda pour effectuer cette tâche, consultez Transformation de données Amazon Kinesis Data Firehose dans le même guide.

Go

L'exemple de code suivant utilise le client opensearch-go pour Go afin d'établir une connexion sécurisée à la collection OpenSearch sans serveur spécifiée et de créer un index unique. Vous devez fournir des valeurs pour `region` et `host`.

```
package main

import (
  "context"
```

```
  "log"
  "strings"
  "github.com/aws/aws-sdk-go-v2/aws"
  "github.com/aws/aws-sdk-go-v2/config"
  opensearch "github.com/opensearch-project/opensearch-go/v2"
  opensearchapi "github.com/opensearch-project/opensearch-go/v2/opensearchapi"
  requestsigner "github.com/opensearch-project/opensearch-go/v2/signer/awsv2"
)

const endpoint = "" // serverless collection endpoint

func main() {
 ctx := context.Background()

 awsCfg, err := config.LoadDefaultConfig(ctx,
  config.WithRegion("<AWS_REGION>"),
  config.WithCredentialsProvider(
   getCredentialProvider("<AWS_ACCESS_KEY>", "<AWS_SECRET_ACCESS_KEY>",
 "<AWS_SESSION_TOKEN>"),
  ),
 )
 if err != nil {
  log.Fatal(err) // don't log.fatal in a production-ready app
 }

 // create an AWS request Signer and load AWS configuration using default config folder
 or env vars.
 signer, err := requestsigner.NewSignerWithService(awsCfg, "aoss") // "aoss" for Amazon
 OpenSearch Serverless
 if err != nil {
  log.Fatal(err) // don't log.fatal in a production-ready app
 }

 // create an opensearch client and use the request-signer
 client, err := opensearch.NewClient(opensearch.Config{
  Addresses: []string{endpoint},
  Signer:    signer,
 })
 if err != nil {
  log.Fatal("client creation err", err)
 }

 indexName := "go-test-index"
```

```
  // define index mapping
 mapping := strings.NewReader(`{
  "settings": {
    "index": {
         "number_of_shards": 4
         }
       }
  }`)

 // create an index
 createIndex := opensearchapi.IndicesCreateRequest{
  Index: indexName,
    Body: mapping,
 }
 createIndexResponse, err := createIndex.Do(context.Background(), client)
 if err != nil {
  log.Println("Error ", err.Error())
  log.Println("failed to create index ", err)
  log.Fatal("create response body read err", err)
 }
 log.Println(createIndexResponse)

 // delete the index
 deleteIndex := opensearchapi.IndicesDeleteRequest{
  Index: []string{indexName},
 }

 deleteIndexResponse, err := deleteIndex.Do(context.Background(), client)
 if err != nil {
  log.Println("failed to delete index ", err)
  log.Fatal("delete index response body read err", err)
 }
 log.Println("deleting index", deleteIndexResponse)
}

func getCredentialProvider(accessKey, secretAccessKey, token string)
 aws.CredentialsProviderFunc {
 return func(ctx context.Context) (aws.Credentials, error) {
  c := &aws.Credentials{
   AccessKeyID:     accessKey,
   SecretAccessKey: secretAccessKey,
   SessionToken:    token,
  }
  return *c, nil
```

```
 }
}
```

Java

L'exemple de code suivant utilise le client opensearch-java pour Java afin d'établir une connexion sécurisée à la collection OpenSearch sans serveur spécifiée et de créer un index unique. Vous devez fournir des valeurs pour `region` et `host`.

La différence importante par rapport aux domaines OpenSearch Service réside dans le nom du service (`aoss`au lieu de`es`).

```
// import OpenSearchClient to establish connection to OpenSearch Serverless collection
import org.opensearch.client.opensearch.OpenSearchClient;

SdkHttpClient httpClient = ApacheHttpClient.builder().build();
// create an opensearch client and use the request-signer
OpenSearchClient client = new OpenSearchClient(
    new AwsSdk2Transport(
        httpClient,
        "...us-west-2.aoss.amazonaws.com", // serverless collection endpoint
        "aoss" // signing service name
        Region.US_WEST_2, // signing service region
        AwsSdk2TransportOptions.builder().build()
    )
);

String index = "sample-index";

// create an index
CreateIndexRequest createIndexRequest = new
 CreateIndexRequest.Builder().index(index).build();
CreateIndexResponse createIndexResponse = client.indices().create(createIndexRequest);
System.out.println("Create index reponse: " + createIndexResponse);

// delete the index
DeleteIndexRequest deleteIndexRequest = new
 DeleteIndexRequest.Builder().index(index).build();
DeleteIndexResponse deleteIndexResponse = client.indices().delete(deleteIndexRequest);
System.out.println("Delete index reponse: " + deleteIndexResponse);

httpClient.close();
```

L'exemple de code suivant établit à nouveau une connexion sécurisée, puis recherche un index.

```
import org.opensearch.client.opensearch.OpenSearchClient;
>>>>>>> aoss-slr-update

SdkHttpClient httpClient = ApacheHttpClient.builder().build();

OpenSearchClient client = new OpenSearchClient(
    new AwsSdk2Transport(
        httpClient,
        "...us-west-2.aoss.amazonaws.com", // serverless collection endpoint
        "aoss" // signing service name
        Region.US_WEST_2, // signing service region
        AwsSdk2TransportOptions.builder().build()
    )
);

Response response = client.generic()
    .execute(
        Requests.builder()
            .endpoint("/" + "users" + "/_search?typed_keys=true")
            .method("GET")
            .json("{"
                + "    \"query\": {"
                + "        \"match_all\": {}"
                + "    }"
                + "}")
            .build());

httpClient.close();
```

JavaScript

L'exemple de code suivant utilise le client opensearch-js JavaScript pour établir une connexion sécurisée à la collection OpenSearch sans serveur spécifiée, créer un index unique, ajouter un document et supprimer l'index. Vous devez fournir des valeurs pour `node` et `region`.

La différence importante par rapport aux domaines OpenSearch Service réside dans le nom du service (`aoss`au lieu dees).

Version 3

Cet exemple utilise la version 3 du kit SDK pour JavaScript in Node.js.

```
const { defaultProvider } = require('@aws-sdk/credential-provider-node');
const { Client } = require('@opensearch-project/opensearch');
const { AwsSigv4Signer } = require('@opensearch-project/opensearch/aws');

async function main() {
    // create an opensearch client and use the request-signer
    const client = new Client({
        ...AwsSigv4Signer({
            region: 'us-west-2',
            service: 'aoss',
            getCredentials: () => {
                const credentialsProvider = defaultProvider();
                return credentialsProvider();
            },
        }),
        node: '' # // serverless collection endpoint
    });

    const index = 'movies';

    // create index if it doesn't already exist
    if (!(await client.indices.exists({ index })).body) {
        console.log((await client.indices.create({ index })).body);
    }

    // add a document to the index
    const document = { foo: 'bar' };
    const response = await client.index({
        id: '1',
        index: index,
        body: document,
    });
    console.log(response.body);

    // delete the index
    console.log((await client.indices.delete({ index })).body);
}

main();
```

Version 2

Cet exemple utilise la version 2 du SDK pour JavaScript le fichier Node.js.

```
const AWS = require('aws-sdk');
const { Client } = require('@opensearch-project/opensearch');
const { AwsSigv4Signer } = require('@opensearch-project/opensearch/aws');

async function main() {
    // create an opensearch client and use the request-signer
    const client = new Client({
        ...AwsSigv4Signer({
            region: 'us-west-2',
            service: 'aoss',
            getCredentials: () =>
                new Promise((resolve, reject) => {
                    AWS.config.getCredentials((err, credentials) => {
                        if (err) {
                            reject(err);
                        } else {
                            resolve(credentials);
                        }
                    });
                }),
        }),
        node: '' # // serverless collection endpoint
    });

    const index = 'movies';

    // create index if it doesn't already exist
    if (!(await client.indices.exists({ index })).body) {
        console.log((await client.indices.create({
            index
        })).body);
    }

    // add a document to the index
    const document = {
        foo: 'bar'
    };
    const response = await client.index({
        id: '1',
        index: index,
        body: document,
    });
    console.log(response.body);
```

```
    // delete the index
    console.log((await client.indices.delete({ index })).body);
}

main();
```

Logstash

Vous pouvez utiliser le OpenSearch plugin Logstash pour publier des journaux dans des collections OpenSearch Serverless.

Utiliser Logstash afin d'envoyer des données vers sans serveur OpenSearch

1. Installez la version 2.0.0 ou ultérieure du logstash-output-opensearchplugin à l'aide de Docker ou Linux.

   Docker

   Docker héberge le logiciel Logstash OSS avec le plugin de OpenSearch sortie préinstallé : opensearchproject/ -output-plugin. logstash-oss-with-opensearch Vous pouvez extraire l'image comme n'importe quelle autre image :

   ```
   docker pull opensearchproject/logstash-oss-with-opensearch-output-plugin:latest
   ```

   Linux

   Si vous ne l'avez pas déjà fait, installez la dernière version de Logstash. Ensuite, installez la version 2.0.0 du plugin de sortie :

   ```
   cd logstash-8.5.0/
   bin/logstash-plugin install --version 2.0.0 logstash-output-opensearch
   ```

   Si le plugin est déjà installé, mettez-le à jour vers la dernière version :

   ```
   bin/logstash-plugin update logstash-output-opensearch
   ```

À partir de la version 2.0.0 du plugin, le AWS kit SDK utilise la version 3. Si vous utilisez une version de Logstash antérieure à 8.4.0, vous devez supprimer tous les AWS plugins préinstallés et installer le plugin : `logstash-integration-aws`

```
/usr/share/logstash/bin/logstash-plugin remove logstash-input-s3
/usr/share/logstash/bin/logstash-plugin remove logstash-input-sqs
/usr/share/logstash/bin/logstash-plugin remove logstash-output-s3
/usr/share/logstash/bin/logstash-plugin remove logstash-output-sns
/usr/share/logstash/bin/logstash-plugin remove logstash-output-sqs
/usr/share/logstash/bin/logstash-plugin remove logstash-output-cloudwatch

/usr/share/logstash/bin/logstash-plugin install --version 0.1.0.pre logstash-
integration-aws
```

2. Pour que le plugin OpenSearch de sortie fonctionne avec OpenSearch sans serveur, vous devez apporter les modifications suivantes à la section de `opensearch` sortie de logstash.conf :

   - Spécifiez `aoss` comme `service_name` sous `auth_type`.
   - Spécifiez le point de terminaison de votre collection pour `hosts`.
   - Ajoutez les paramètres `default_server_major_version` et `legacy_template`. Ces paramètres sont nécessaires pour que le plugin fonctionne avec OpenSearch sans serveur.

```
output {
  opensearch {
    hosts => "collection-endpoint:443"
    auth_type => {
      ...
      service_name => 'aoss'
    }
    default_server_major_version => 2
    legacy_template => false
  }
}
```

L'entrée de cet exemple de fichier de configuration repose sur les fichiers d'un compartiment S3 et les envoie vers une collection OpenSearch sans serveur :

```
input {
  s3  {
```

```
    bucket => "my-s3-bucket"
    region => "us-east-1"
  }
}

output {
  opensearch {
    ecs_compatibility => disabled
    hosts => "https://my-collection-endpoint.us-east-1.aoss.amazonaws.com:443"
    index => my-index
    auth_type => {
      type => 'aws_iam'
      aws_access_key_id => 'your-access-key'
      aws_secret_access_key => 'your-secret-key'
      region => 'us-east-1'
      service_name => 'aoss'
    }
    default_server_major_version => 2
    legacy_template => false
  }
}
```

3. Ensuite, lancez Logstash avec la nouvelle configuration pour tester le plugin :

```
bin/logstash -f config/test-plugin.conf
```

Python

L'exemple de code suivant utilise le client opensearch-py pour Python afin d'établir une connexion sécurisée à la collection OpenSearch sans serveur spécifiée, de créer un index unique et d'effectuer une recherche dans cet index. Vous devez fournir des valeurs pour `region` et `host`.

La différence importante par rapport aux domaines OpenSearch Service réside dans le nom du service (`aoss`au lieu dees).

```
from opensearchpy import OpenSearch, RequestsHttpConnection, AWSV4SignerAuth
import boto3

host = ''  # serverless collection endpoint, without https://
region = ''  # e.g. us-east-1

service = 'aoss'
```

```
credentials = boto3.Session().get_credentials()
auth = AWSV4SignerAuth(credentials, region, service)

# create an opensearch client and use the request-signer
client = OpenSearch(
    hosts=[{'host': host, 'port': 443}],
    http_auth=auth,
    use_ssl=True,
    verify_certs=True,
    connection_class=RequestsHttpConnection,
    pool_maxsize=20,
)

# create an index
index_name = 'books-index'
create_response = client.indices.create(
    index_name
)

print('\nCreating index:')
print(create_response)

# index a document
document = {
  'title': 'The Green Mile',
  'director': 'Stephen King',
  'year': '1996'
}

response = client.index(
    index = 'books-index',
    body = document,
    id = '1'
)


# delete the index
delete_response = client.indices.delete(
    index_name
)

print('\nDeleting index:')
print(delete_response)
```

Ruby

La `opensearch-aws-sigv4` gemme donne accès à OpenSearch sans serveur, ainsi qu'à OpenSearch Service, dès le départ. Elle possède toutes les fonctions du client opensearch-ruby, car elle est une dépendance de cette gemme.

Lors de l'instanciation du signataire Sigv4, spécifiez `aoss` comme nom de service :

```
require 'opensearch-aws-sigv4'
require 'aws-sigv4'

signer = Aws::Sigv4::Signer.new(service: 'aoss',
                                region: 'us-west-2',
                                access_key_id: 'key_id',
                                secret_access_key: 'secret')

# create an opensearch client and use the request-signer
client = OpenSearch::Aws::Sigv4Client.new(
  { host: 'https://your.amz-opensearch-serverless.endpoint',
    log: true },
  signer)

# create an index
index = 'prime'
client.indices.create(index: index)

# insert data
client.index(index: index, id: '1', body: { name: 'Amazon Echo',
                                            msrp: '5999',
                                            year: 2011 })

# query the index
client.search(body: { query: { match: { name: 'Echo' } } })

# delete index entry
client.delete(index: index, id: '1')

# delete the index
client.indices.delete(index: index)
```

Signature des demandes HTTP avec d'autres clients

Les exigences suivantes s'appliquent lors de la signature des demandes aux collections OpenSearch sans serveur lorsque vous formulez des demandes HTTP avec d'autres clients.

- Vous devez spécifier le nom du service sous la forme `aoss`.
- L'en-tête `x-amz-content-sha256` est obligatoire pour toutes les requêtes AWS Signature Version 4. Il fournit un hachage de la charge utile de la requête. S'il existe une charge utile de requête, définissez la valeur sur son hachage de chiffrement SHA (Secure Hash Algorithm, algorithme de hachage sécurisé). SHA256 S'il n'existe aucune charge utile de requête, définissez la valeur sur `e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855`, qui est le hachage d'une chaîne vide.

Rubriques

- Indexation avec cURL
- Indexation avec Postman

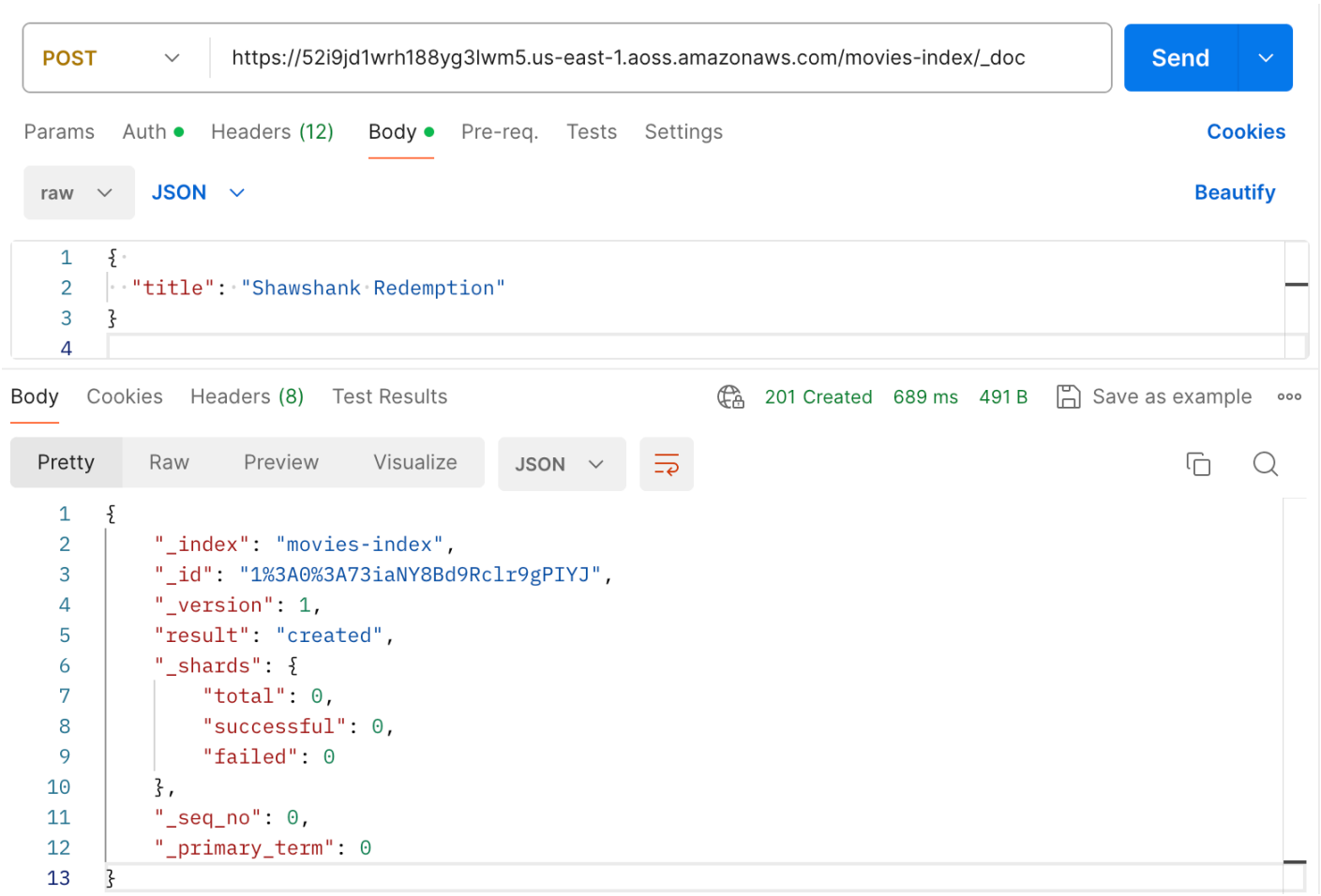
Indexation avec cURL

L'exemple de demande suivant utilise la bibliothèque de demandes d'URL du client (cURL) pour envoyer un seul document vers un index nommé `movies-index` dans une collection :

```
curl -XPOST \
    --user "$AWS_ACCESS_KEY_ID":"$AWS_SECRET_ACCESS_KEY" \
    --aws-sigv4 "aws:amz:us-east-1:aoss" \
    --header "x-amz-content-sha256: $REQUEST_PAYLOAD_SHA_HASH" \
    --header "x-amz-security-token: $AWS_SESSION_TOKEN" \
    "https://my-collection-endpoint.us-east-1.aoss.amazonaws.com/movies-index/_doc" \
    -H "Content-Type: application/json" -d '{"title": "Shawshank Redemption"}'
```

Indexation avec Postman

L'image suivante montre comment envoyer une demande à une collection à l'aide de Postman. Pour obtenir des instructions d'authentification, consultez le flux de travail d'authentification Authentifier avec AWS signature dans Postman.



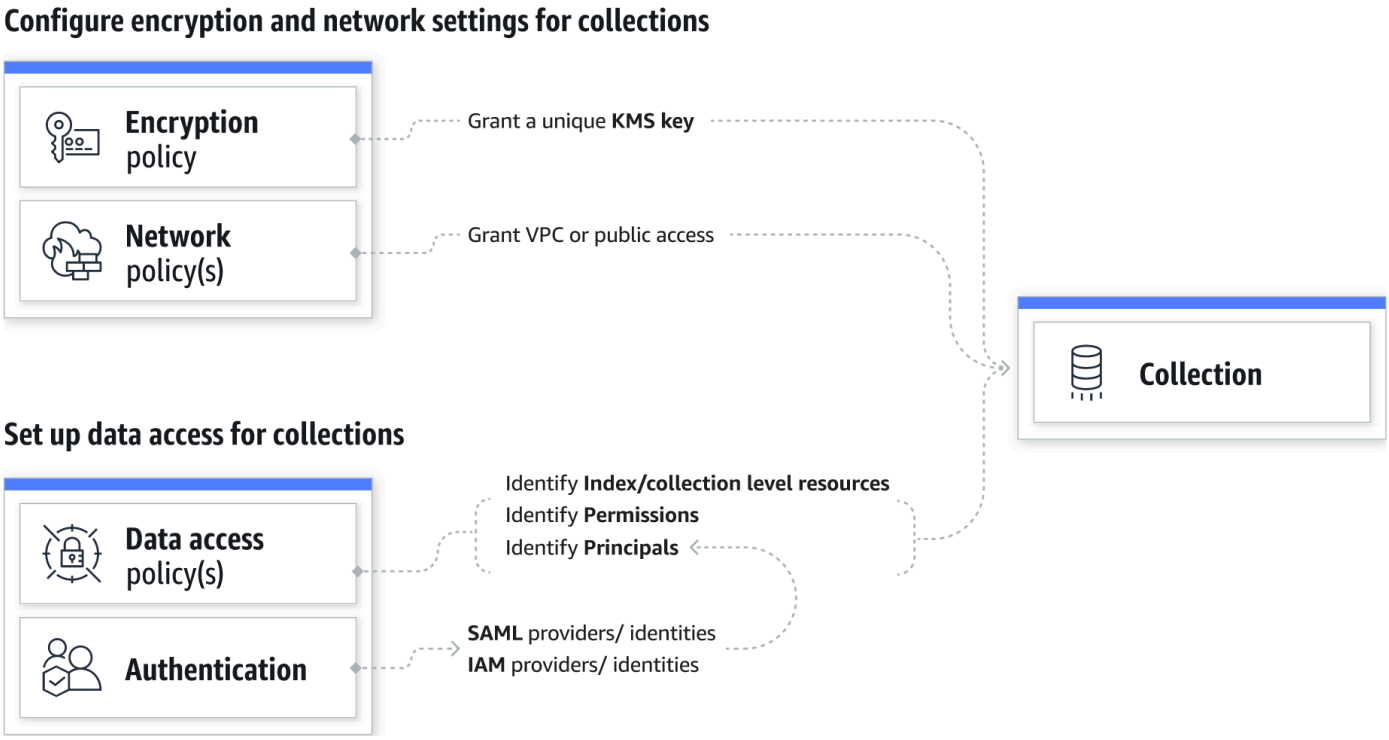
Présentation de la sécurité dans Amazon OpenSearch Serverless

La sécurité dans Amazon OpenSearch Serverless diffère fondamentalement de la sécurité dans Amazon OpenSearch Service pour les raisons suivantes :

Fonctionnalité	OpenSearch Service	OpenSearch Sans serveur
Contrôle d'accès aux données	L'accès aux données est déterminé par des politiques IAM et un contrôle d'accès précis.	L'accès aux données est déterminé par des stratégies d'accès aux données.
Chiffrement au repos	Le chiffrement au repos est facultatif pour les domaines.	Le chiffrement au repos est requis pour les collections.

Fonctionnalité	OpenSearch Service	OpenSearch Sans serveur
Configuration et administration de la sécurité	Vous devez configurer le réseau, le chiffrement et l'accès aux données individuellement pour chaque domaine.	Vous pouvez utiliser des stratégies de sécurité pour gérer les paramètres de sécurité de plusieurs collections à grande échelle.

Le schéma suivant illustre les composants de sécurité qui constituent une collection fonctionnelle. Une collection doit être associée à une clé de chiffrement, des paramètres d'accès réseau et une stratégie d'accès aux données accordant l'accès à ses ressources.



Rubriques

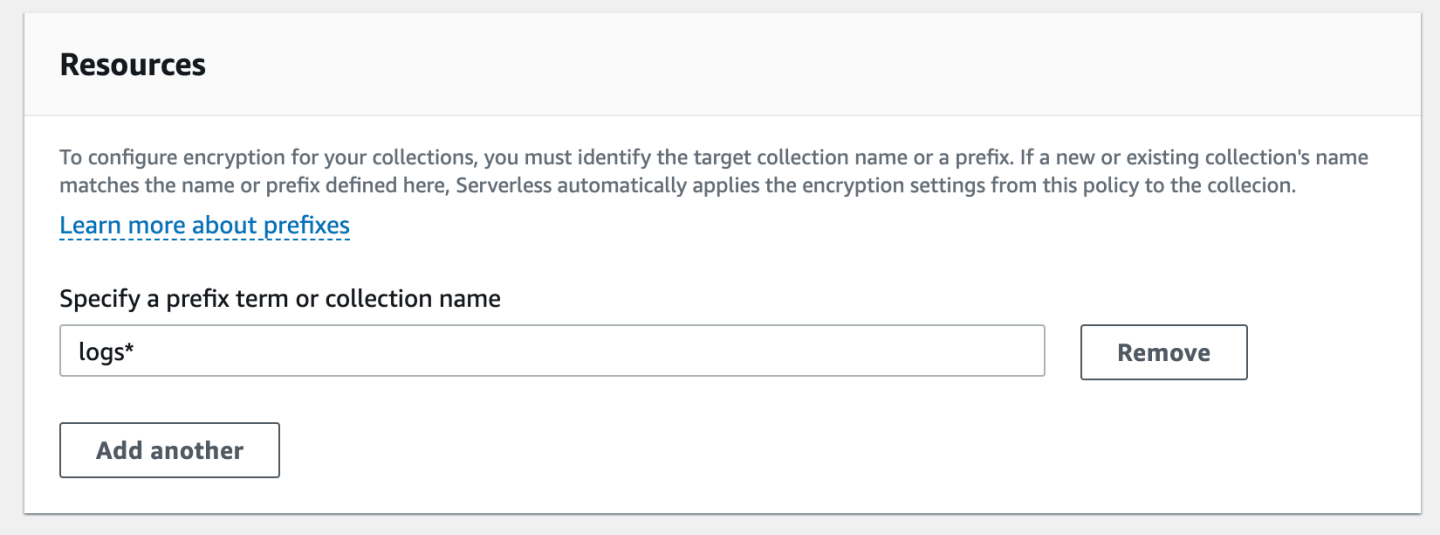
- Stratégies de chiffrement
- Stratégies réseau
- Stratégies d'accès aux données
- Authentification IAM et SAML
- Sécurité de l'infrastructure

- Commencer à utiliser la sécurité dans Amazon OpenSearch Serverless
- Identity and Access Management pour Amazon OpenSearch Serverless
- Chiffrement dans Amazon OpenSearch Serverless
- Accès réseau pour Amazon OpenSearch Serverless
- Contrôle d'accès aux données pour Amazon OpenSearch Serverless
- Accédez à Amazon OpenSearch Serverless à l'aide d'un point de terminaison d'interface ()AWS PrivateLink
- Authentification SAML pour Amazon Serverless OpenSearch
- Validation de la conformité pour Amazon OpenSearch Serverless

Stratégies de chiffrement

Les politiques de chiffrement définissent si vos collections sont chiffrées à l'aide d'une clé gérée par le client Clé détenue par AWS ou d'une clé gérée par le client. Les stratégies de chiffrement sont constituées de deux composants : un modèle de ressource et une clé de chiffrement. Le modèle de ressource définit la ou les collections auxquelles la stratégie s'applique. La clé de chiffrement détermine la manière dont les collections associées seront sécurisées.

Pour appliquer une stratégie à plusieurs collections, vous devez inclure un caractère générique (*) dans la règle de stratégie. Par exemple, la stratégie suivante s'appliquera à toutes les collections dont le nom commence par « logs ».

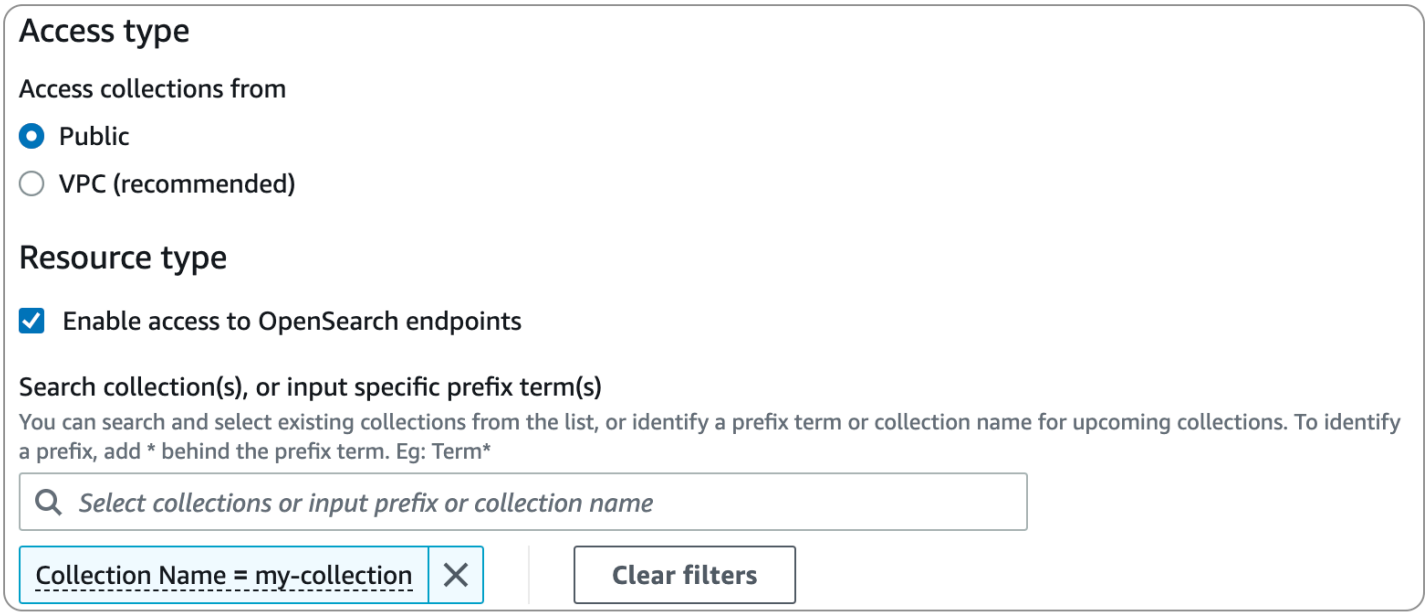


Les stratégies de chiffrement simplifient le processus de création et de gestion des collections, en particulier lorsque vous procédez par programmation. Vous pouvez créer une collection en spécifiant simplement un nom, et une clé de chiffrement lui est automatiquement attribuée lors de sa création.

Stratégies réseau

Les politiques réseau définissent si vos collections sont accessibles en privé ou via Internet à partir de réseaux publics. Les collections privées sont accessibles via des points de terminaison OpenSearch VPC gérés sans serveur, ou par des utilisateurs spécifiques Services AWS tels qu'Amazon Bedrock utilisant un accès privé.Service AWS Tout comme les stratégies de chiffrement, les stratégies réseau peuvent s'appliquer à plusieurs collections, ce qui vous permet de gérer l'accès réseau à de nombreuses collections à grande échelle.

Les stratégies réseau sont constituées de deux composants : un type d'accès et un type de ressource. Le type d'accès peut être public ou privé. Le type de ressource détermine si l'accès que vous choisissez s'applique au point de terminaison de collecte, au point de terminaison OpenSearch des tableaux de bord ou aux deux.



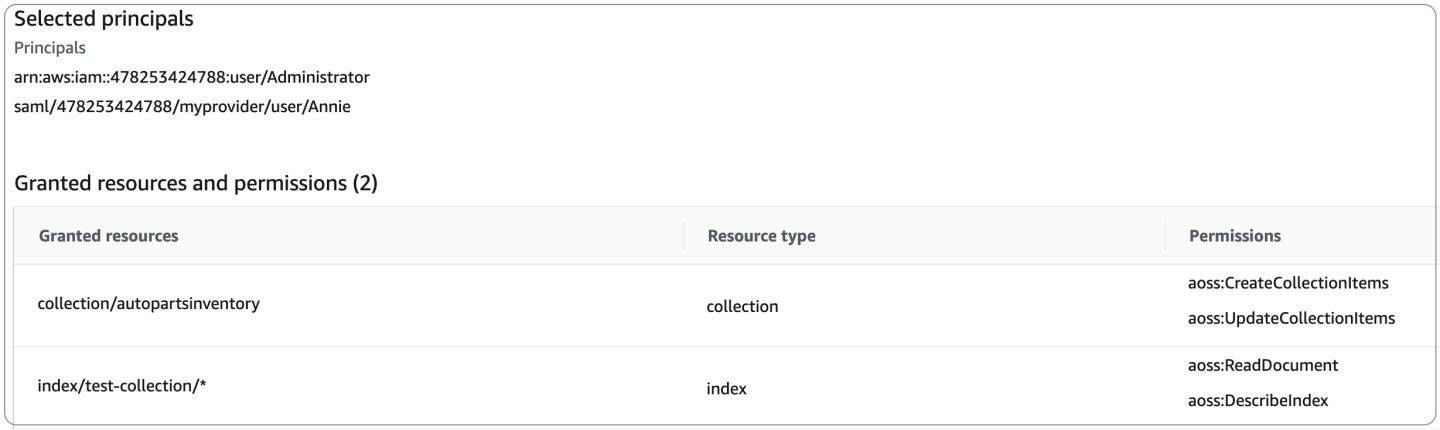
Si vous envisagez de configurer l'accès VPC dans le cadre d'une politique réseau, vous devez d'abord créer un ou plusieurs points de terminaison VPC gérés OpenSearch sans serveur. Ces points de terminaison vous permettent d'accéder à OpenSearch Serverless comme s'il se trouvait dans votre VPC, sans utiliser de passerelle Internet, de périphérique NAT, de connexion VPN ou de connexion. AWS Direct Connect

L'accès privé à ne Services AWS peut s'appliquer qu'au point de OpenSearch terminaison de la collection, et non au point de terminaison OpenSearch des tableaux de bord. Services AWS ne peut pas être autorisé à accéder aux OpenSearch tableaux de bord.

Stratégies d'accès aux données

Les stratégies d'accès aux données définissent la manière dont vos utilisateurs accèdent aux données de vos collections. Les stratégies d'accès aux données vous permettent de gérer les collections à grande échelle en attribuant automatiquement des autorisations d'accès aux collections et aux index qui correspondent à un modèle spécifique. Plusieurs stratégies peuvent s'appliquer à une seule ressource.

Les stratégies d'accès aux données se composent d'un ensemble de règles, chacune comportant trois éléments : un type de ressource, des ressources octroyées et un ensemble d'autorisations. Le type de ressource peut être une collection ou un index. Les ressources accordées peuvent être collection/index des noms ou des modèles avec un caractère générique (*). La liste des autorisations indique les opérations OpenSearch d'API auxquelles la politique accorde l'accès. En outre, la stratégie contient une liste de principaux qui spécifient les rôles IAM, les utilisateurs et les identités SAML auxquels accorder l'accès.



Pour plus d'informations sur le format d'une stratégie d'accès aux données, veuillez consulter la rubrique relative à la syntaxe de la stratégie.

Avant de créer une stratégie d'accès aux données, vous devez disposer d'un ou de plusieurs rôles ou utilisateurs IAM ou d'identités SAML, à qui accorder l'accès dans la stratégie. Pour plus de détails, veuillez consulter la section suivante.

> **Note**
>
> Le passage de l'accès public à l'accès privé pour votre collection supprimera l'onglet Indexes de la console de collecte OpenSearch sans serveur.

Authentification IAM et SAML

Les principaux IAM et les identités SAML constituent les éléments de base d'une stratégie d'accès aux données. Dans l'instruction `principal` d'une stratégie d'accès, vous pouvez inclure des rôles IAM, des utilisateurs et des identités SAML. Ces principaux se voient ensuite octroyer les autorisations que vous spécifiez dans les règles de stratégie associées.

```
[
   {
      "Rules":[
         {
            "ResourceType":"index",
            "Resource":[
               "index/marketing/orders*"
            ],
            "Permission":[
               "aoss:*"
            ]
         }
      ],
      "Principal":[
         "arn:aws:iam::123456789012:user/Dale",
         "arn:aws:iam::123456789012:role/RegulatoryCompliance",
         "saml/123456789012/myprovider/user/Annie"
      ]
   }
]
```

Vous configurez l'authentification SAML directement dans OpenSearch Serverless. Pour de plus amples informations, veuillez consulter the section called “Authentification SAML”.

Sécurité de l’infrastructure

Amazon OpenSearch Serverless est protégé par la sécurité du réseau AWS mondial. Pour plus d'informations sur les services AWS de sécurité et sur la manière dont AWS l'infrastructure est

protégée, consultez la section Sécurité du AWS cloud. Pour concevoir votre AWS environnement en utilisant les meilleures pratiques en matière de sécurité de l'infrastructure, consultez la section Protection de l'infrastructure dans le cadre AWS bien architecturé du pilier de sécurité.

Vous utilisez des appels d'API AWS publiés pour accéder à Amazon OpenSearch Serverless via le réseau. Les clients doivent prendre en charge le protocole TLS (Transport Layer Security). Nous exigeons TLS 1.2 et recommandons TLS 1.3. Pour obtenir la liste des chiffrements pris en charge pour TLS 1.3, consultez la section Protocoles et chiffrements TLS dans la documentation Elastic Load Balancing.

En outre, vous devez signer les demandes à l'aide d'un identifiant de clé d'accès et d'une clé d'accès secrète associés à un principal IAM. Vous pouvez également utiliser AWS Security Token Service (AWS STS) pour générer des informations d'identification de sécurité temporaires et signer les demandes.

Commencer à utiliser la sécurité dans Amazon OpenSearch Serverless

Les didacticiels suivants vous aideront à commencer à utiliser Amazon OpenSearch Serverless. Les deux didacticiels suivent les mêmes étapes de base, mais l'un utilise la console tandis que l'autre utilise l' AWS CLI.

Veuillez noter que les cas d'utilisation présentés dans ces didacticiels sont simplifiés. Les stratégies réseau et de sécurité sont assez légères. Pour les charges de travail de production, nous vous recommandons de configurer des fonctionnalités de sécurité plus robustes telles que l'authentification SAML, l'accès au VPC et des stratégies d'accès aux données restrictives.

Rubriques

- Tutoriel : prise en main de la sécurité dans Amazon OpenSearch Serverless (console)
- Tutoriel : prise en main de la sécurité dans Amazon OpenSearch Serverless (CLI)

Tutoriel : prise en main de la sécurité dans Amazon OpenSearch Serverless (console)

Ce didacticiel explique les étapes de base pour créer et gérer des politiques de sécurité à l'aide de la console Amazon OpenSearch Serverless.

Dans le cadre de ce didacticiel, vous suivrez les étapes suivantes :

1. Configurer des autorisations
2. Créer une stratégie de chiffrement

3. Création d'une stratégie réseau
4. Configuration d'une stratégie d'accès aux données
5. Créer une collection
6. Charger et rechercher des données

Ce didacticiel vous explique comment configurer une collection à l'aide du AWS Management Console. Pour les mêmes étapes à suivre lors de l'utilisation du AWS CLI, voirthe section called “Didacticiel : démarrer avec la sécurité (CLI)”.

Étape 1 : configurer des autorisations

> **Note**
>
> Vous pouvez ignorer cette étape si vous utilisez déjà une politique basée sur l'identité plus large, telle que `Action":"aoss:*"` ou `Action":"*"`. Toutefois, dans les environnements de production, nous vous recommandons de suivre le principe du moindre privilège et de n'attribuer que les autorisations minimales nécessaires pour effectuer une tâche.

Afin de suivre ce didacticiel, vous devez disposer des autorisations IAM appropriées. Votre utilisateur ou votre rôle doit être associé à une politique basée sur l'identité avec les autorisations minimales suivantes :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aoss:ListCollections",
        "aoss:BatchGetCollection",
        "aoss:CreateCollection",
        "aoss:CreateSecurityPolicy",
        "aoss:GetSecurityPolicy",
        "aoss:ListSecurityPolicies",
        "aoss:CreateAccessPolicy",
        "aoss:GetAccessPolicy",
        "aoss:ListAccessPolicies"
```

```
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Pour obtenir la liste complète des autorisations OpenSearch sans serveur, consultezthe section called “Gestion de l’identité et des accès”.

Étape 2 : créer une stratégie de chiffrement

Les politiques de chiffrement spécifient la AWS KMS clé que OpenSearch Serverless utilisera pour chiffrer la collection. Vous pouvez chiffrer des collections à l'aide d'une clé Clé gérée par AWS ou d'une autre clé. Par souci de simplicité, dans le cadre de ce didacticiel, nous allons chiffrer notre collection à l'aide d'une Clé gérée par AWS.

Créer une stratégie de chiffrement

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Développez Serverless (Sans serveur) dans le panneau de navigation de gauche et choisissez Encryption policies (Stratégies de chiffrement).
3. Choisissez Create encryption policy (Créer une stratégie de chiffrement).
4. Nommez la stratégie books-policy. Pour la description, saisissez Encryption policy for books collection (Stratégie de chiffrement pour la collection books).
5. Dans Resources (Ressources), saisissez books (livres), le nom que vous donnerez à votre collection. Si vous souhaitez être plus large, vous pouvez inclure un astérisque (`books*`) pour que la stratégie s'applique à toutes les collections commençant par le mot « books ».
6. Pour le chiffrement, maintenez l'option Utiliser la clé AWS détenue sélectionnée.
7. Choisissez Créer.

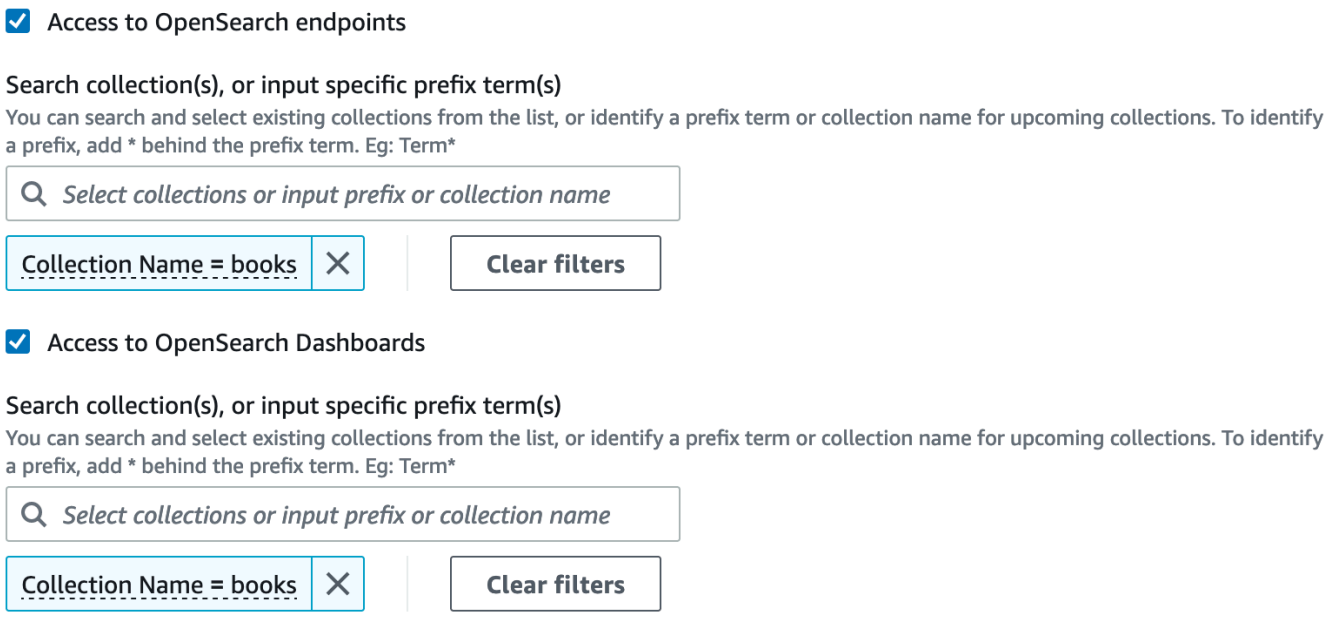
Étape 3 : créer une politique réseau

Les politiques réseau déterminent si votre collection est accessible via Internet à partir de réseaux publics ou si elle doit être accessible via des points de terminaison OpenSearch VPC gérés sans serveur. Dans le cadre de ce didacticiel, nous allons configurer l'accès public.

Créer une stratégie réseau

1. Choisissez Network policies (Stratégies réseau) dans le panneau de navigation de gauche, puis Create network policy (Créer une stratégie réseau).
2. Nommez la stratégie books-policy. Pour la description, saisissez Network policy for books collection (Stratégie réseau pour la collection books).
3. Sous Rule 1 (Règle 1), nommez la règle Public access for books collection (Accès public à la collection books).
4. Par souci de simplicité, dans le cadre de ce didacticiel, nous allons configurer l'accès public à la collection books. Pour le type d'accès, sélectionnez Public.
5. Nous allons accéder à la collection depuis les OpenSearch tableaux de bord. Pour ce faire, vous devez configurer l'accès réseau pour les tableaux de bord et le OpenSearch point de terminaison, sinon les tableaux de bord ne fonctionneront pas.

   Pour le type de ressource, activez à la fois l'accès aux OpenSearch points de terminaison et l'accès aux OpenSearch tableaux de bord.
6. Dans les deux zones de saisie, saisissez Collection Name = books (Nom de la collection = books). Ce paramètre réduit la portée de la stratégie afin qu'elle ne s'applique qu'à une seule collection (`books`). Votre règle devrait ressembler à ceci :



7. Choisissez Créer.

Étape 4 : Création d'une politique d'accès aux données

Les données de votre collection ne seront pas accessibles tant que vous n'aurez pas configuré l'accès aux données. Les stratégies d'accès aux données sont distinctes de la politique IAM basée sur l'identité que vous avez configurée à l'étape 1. Elles permettent aux utilisateurs d'accéder aux données réelles d'une collection.

Dans le cadre de ce didacticiel, nous allons fournir à un seul utilisateur les autorisations requises pour indexer des données dans la collection books.

Créer une stratégie d'accès aux données

1. Dans le panneau de navigation de gauche, choisissez Data access policies (Stratégies d'accès aux données), puis Create access policy (Créer une stratégie d'accès).
2. Nommez la stratégie books-policy. Pour la description, saisissez Data access policy for books collection (Stratégie d'accès aux données pour la collection books).
3. Sélectionnez JSON comme méthode de définition de stratégie et collez la stratégie suivante dans l'éditeur JSON.

   Remplacez l'ARN principal par l'ARN du compte que vous utiliserez pour vous connecter aux OpenSearch tableaux de bord et indexer les données.

```
[
   {
      "Rules":[
         {
            "ResourceType":"index",
            "Resource":[
               "index/books/*"
            ],
            "Permission":[
               "aoss:CreateIndex",
               "aoss:DescribeIndex",
               "aoss:ReadDocument",
               "aoss:WriteDocument",
               "aoss:UpdateIndex",
               "aoss:DeleteIndex"
            ]
         }
      ],
      "Principal":[
```

```
            "arn:aws:iam::123456789012:user/my-user"
        ]
    }
]
```

Cette stratégie fournit à un seul utilisateur les autorisations minimales requises pour créer un index dans la collection books, indexer certaines données et les rechercher.

4. Choisissez Créer.

Étape 5 : Création d'une collection

Maintenant que vous avez configuré les stratégies de chiffrement et réseau, vous pouvez créer une collection correspondante et les paramètres de sécurité lui seront automatiquement appliqués.

Pour créer une collection OpenSearch sans serveur

1. Choisissez Collections dans le panneau de navigation de gauche, puis choisissez Create collection (Créer une collection).
2. Nommez la collection books.
3. Pour le type de collection, choisissez Search (Rechercher).
4. Sous Chiffrement, OpenSearch Serverless vous informe que le nom de la collection correspond à la politique de `books-policy` chiffrement.
5. Dans les paramètres d'accès au réseau, OpenSearch Serverless vous informe que le nom de la collection correspond à la politique du `books-policy` réseau.
6. Choisissez Suivant.
7. Sous Options de politique d'accès aux données, OpenSearch Serverless vous informe que le nom de la collection correspond à la politique d'accès aux `books-policy` données.
8. Choisissez Suivant.
9. Vérifiez la configuration de la collection et choisissez Submit (Soumettre). L'initialisation des collections prend généralement moins d'une minute.

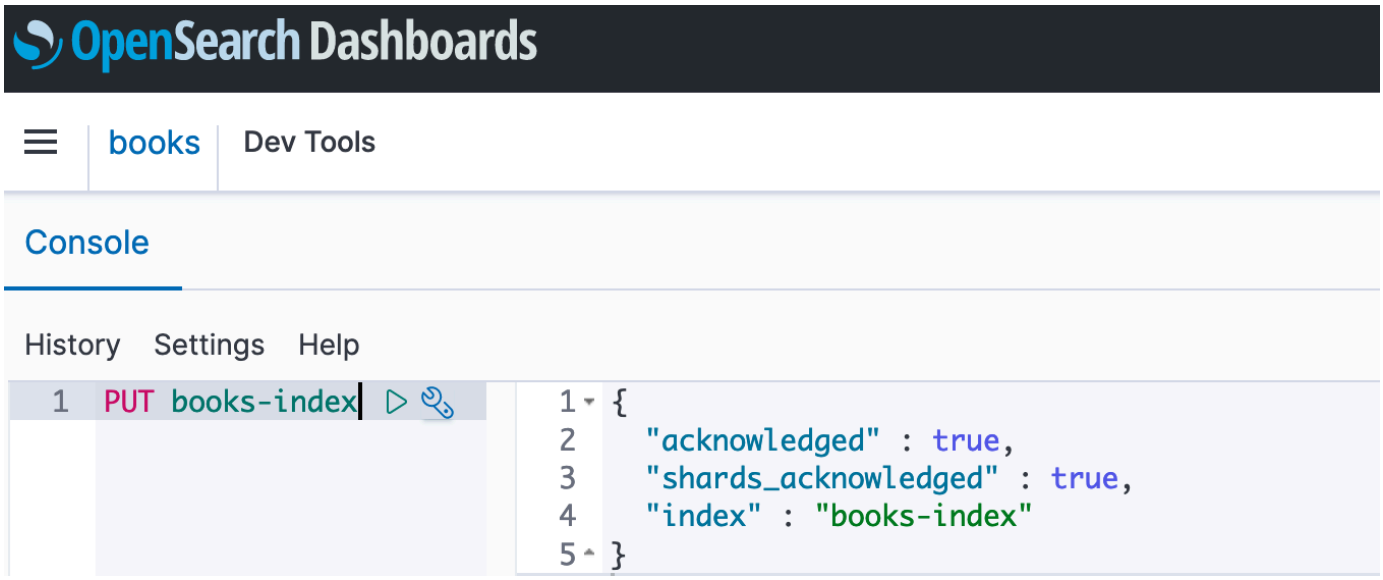
Étape 6 : charger et rechercher des données

Vous pouvez télécharger des données dans une collection OpenSearch sans serveur à l'aide de Postman ou de curl. Par souci de concision, ces exemples utilisent les outils de développement de la console OpenSearch Dashboards.

Indexer et rechercher des données dans une collection

1. Choisissez Collections dans le panneau de navigation de gauche, puis choisissez la collection books pour afficher sa page des détails.
2. Choisissez l'URL OpenSearch des tableaux de bord pour la collection. L'URL est au format `https://collection-id.us-east-1.aoss.amazonaws.com/_dashboards`.
3. Connectez-vous aux OpenSearch tableaux de bord à l'aide des clés AWS d'accès et secrètes du principal que vous avez spécifiées dans votre politique d'accès aux données.
4. Dans OpenSearch Dashboards, ouvrez le menu de navigation de gauche et choisissez Dev Tools.
5. Pour créer un index unique appelé books-index, exécutez la commande suivante :

```
PUT books-index
```



6. Pour indexer un seul document dans books-index, exécutez la commande suivante :

```
PUT books-index/_doc/1
{
  "title": "The Shining",
  "author": "Stephen King",
  "year": 1977
}
```

7. Pour rechercher des données dans OpenSearch les tableaux de bord, vous devez configurer au moins un modèle d'index. OpenSearch utilise ces modèles pour identifier les index que vous

souhaitez analyser. Ouvrez le menu principal Tableaux de bord et choisissez Stack Management (Gestion de la pile), Index Patterns (Modèles d'index), puis Create index pattern (Créer un modèle d'index). Pour ce didacticiel, saisissez books-index.

8. Choisissez Next step (Étape suivante), puis Create index pattern (Créer un modèle d'index). Une fois le modèle créé, vous pouvez consulter les différents champs du document, comme `author` et `title`.
9. Pour commencer à effectuer des recherches sur vos données, ouvrez à nouveau le menu principal et choisissez Discover (Découvrir) ou utilisez l'API de recherche.

Tutoriel : prise en main de la sécurité dans Amazon OpenSearch Serverless (CLI)

Ce didacticiel explique les étapes décrites dans le didacticiel de démarrage de la console pour la sécurité, mais utilise la console AWS CLI plutôt que la console OpenSearch de service.

Dans le cadre de ce didacticiel, vous suivrez les étapes suivantes :

1. Création d'une politique d'autorisations IAM
2. Associer la politique IAM à un rôle IAM
3. Créer une politique de chiffrement
4. Création d'une stratégie réseau
5. Créer une collection
6. Configuration d'une stratégie d'accès aux données
7. Récupérer le point de terminaison de collecte
8. Téléchargez des données sur votre connexion
9. Rechercher des données dans votre collection

L'objectif de ce didacticiel est de configurer une collection unique OpenSearch sans serveur avec des paramètres de chiffrement, de réseau et d'accès aux données assez simples. Par exemple, nous allons configurer l'accès au réseau public, un Clé gérée par AWS système de chiffrement et une politique d'accès aux données simplifiée qui accorde des autorisations minimales à un seul utilisateur.

Dans un scénario de production, envisagez de mettre en œuvre une configuration plus robuste, notamment une authentification SAML, une clé de chiffrement personnalisée et un accès VPC.

Pour commencer à utiliser les politiques de sécurité dans OpenSearch Serverless

1.

> **Note**
>
> Vous pouvez ignorer cette étape si vous utilisez déjà une politique basée sur l'identité plus large, telle que `Action":"aoss:*"` ou `Action":"*"`. Toutefois, dans les environnements de production, nous vous recommandons de suivre le principe du moindre privilège et de n'attribuer que les autorisations minimales nécessaires pour effectuer une tâche.

Pour commencer, créez une AWS Identity and Access Management politique avec les autorisations minimales requises pour effectuer les étapes de ce didacticiel. Nous nommerons la politique `TutorialPolicy` :

```
aws iam create-policy \
  --policy-name TutorialPolicy \
  --policy-document "{\"Version\": \"2012-10-17\",\"Statement\":
 [{\"Action\": [\"aoss:ListCollections\",\"aoss:BatchGetCollection\",
\"aoss:CreateCollection\",\"aoss:CreateSecurityPolicy\",\"aoss:GetSecurityPolicy\",
\"aoss:ListSecurityPolicies\",\"aoss:CreateAccessPolicy\",\"aoss:GetAccessPolicy\",
\"aoss:ListAccessPolicies\"],\"Effect\": \"Allow\",\"Resource\": \"*\"}]}"
```

Exemple de réponse

```
{
    "Policy": {
        "PolicyName": "TutorialPolicy",
        "PolicyId": "ANPAW6WRAECKG6QJWUV7U",
        "Arn": "arn:aws:iam::123456789012:policy/TutorialPolicy",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 0,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2022-10-16T20:57:18+00:00",
        "UpdateDate": "2022-10-16T20:57:18+00:00"
    }
}
```

2. Attachez `TutorialPolicy` au rôle IAM qui indexera et recherchera les données dans la collection. Nous nommerons l'utilisateur `TutorialRole` :

```
aws iam attach-role-policy \
  --role-name TutorialRole \
  --policy-arn arn:aws:iam::123456789012:policy/TutorialPolicy
```

3. Avant de créer une collection, vous devez créer une stratégie de chiffrement qui attribue une Clé détenue par AWS à la collection books que vous créerez ultérieurement.

   Envoyez la requête suivante afin de créer une stratégie de chiffrement pour la collection books :

```
aws opensearchserverless create-security-policy \
  --name books-policy \
  --type encryption --policy "{\"Rules\":[{\"ResourceType\":\"collection\",
\"Resource\":[\"collection\/books\"]}],\"AWSOwnedKey\":true}"
```

   Exemple de réponse

```
{
    "securityPolicyDetail": {
        "type": "encryption",
        "name": "books-policy",
        "policyVersion": "MTY2OTI0MDAwNTk5MF8x",
        "policy": {
            "Rules": [
                {
                    "Resource": [
                        "collection/books"
                    ],
                    "ResourceType": "collection"
                }
            ],
            "AWSOwnedKey": true
        },
        "createdDate": 1669240005990,
        "lastModifiedDate": 1669240005990
    }
}
```

4. Créez une stratégie réseau qui fournit un accès public à la collection books :

```
aws opensearchserverless create-security-policy --name books-policy --type network
 \
  --policy "[{\"Description\":\"Public access for books collection\",\"Rules
\":[{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection\/books\"]},
{\"ResourceType\":\"collection\",\"Resource\":[\"collection\/books\"]}],
\"AllowFromPublic\":true}]"
```

Exemple de réponse

```
{
    "securityPolicyDetail": {
        "type": "network",
        "name": "books-policy",
        "policyVersion": "MTY2OTI0MDI1Njk1NV8x",
        "policy": [
            {
                "Rules": [
                    {
                        "Resource": [
                            "collection/books"
                        ],
                        "ResourceType": "dashboard"
                    },
                    {
                        "Resource": [
                            "collection/books"
                        ],
                        "ResourceType": "collection"
                    }
                ],
                "AllowFromPublic": true,
                "Description": "Public access for books collection"
            }
        ],
        "createdDate": 1669240256955,
        "lastModifiedDate": 1669240256955
    }
}
```

5. Créez la collection books :

```
aws opensearchserverless create-collection --name books --type SEARCH
```

Exemple de réponse

```
{
    "createCollectionDetail": {
        "id": "8kw362bpwg4gx9b2f6e0",
        "name": "books",
        "status": "CREATING",
        "type": "SEARCH",
        "arn": "arn:aws:aoss:us-
east-1:123456789012:collection/8kw362bpwg4gx9b2f6e0",
        "kmsKeyArn": "auto",
        "createdDate": 1669240325037,
        "lastModifiedDate": 1669240325037
    }
}
```

6. Créez une stratégie d'accès aux données qui fournit les autorisations minimales nécessaires pour indexer et rechercher des données dans la collection books. Remplacez l'ARN du principal par l'ARN du `TutorialRole` de l'étape 1 :

```
aws opensearchserverless create-access-policy \
  --name books-policy \
  --type data \
  --policy "[{\"Rules\":[{\"ResourceType\":\"index\",\"Resource\":
[\"index\/books\/books-index\"],\"Permission\":[\"aoss:CreateIndex
\",\"aoss:DescribeIndex\",\"aoss:ReadDocument\",\"aoss:WriteDocument
\",\"aoss:UpdateIndex\",\"aoss:DeleteIndex\"]}],\"Principal\":
[\"arn:aws:iam::123456789012:role\/TutorialRole\"]}]"
```

Exemple de réponse

```
{
    "accessPolicyDetail": {
        "type": "data",
        "name": "books-policy",
        "policyVersion": "MTY2OTI0MDM5NDY1M18x",
        "policy": [
            {
```

```
                "Rules": [
                    {
                        "Resource": [
                            "index/books/books-index"
                        ],
                        "Permission": [
                            "aoss:CreateIndex",
                            "aoss:DescribeIndex",
                            "aoss:ReadDocument",
                            "aoss:WriteDocument",
                            "aoss:UpdateDocument",
                            "aoss:DeleteDocument"
                        ],
                        "ResourceType": "index"
                    }
                ],
                "Principal": [
                    "arn:aws:iam::123456789012:role/TutorialRole"
                ]
            }
        ],
        "createdDate": 1669240394653,
        "lastModifiedDate": 1669240394653
    }
}
```

`TutorialRole` devrait désormais pouvoir indexer et rechercher des documents dans la collection books.

7. Pour appeler l' OpenSearch API, vous avez besoin du point de terminaison de collecte. Envoyez la requête suivante pour récupérer le paramètre `collectionEndpoint` :

```
aws opensearchserverless batch-get-collection --names books
```

Exemple de réponse

```
{
    "collectionDetails": [
        {
            "id": "8kw362bpwg4gx9b2f6e0",
            "name": "books",
            "status": "ACTIVE",
```

```
            "type": "SEARCH",
            "description": "",
            "arn": "arn:aws:aoss:us-
east-1:123456789012:collection/8kw362bpwg4gx9b2f6e0",
            "createdDate": 1665765327107,
            "collectionEndpoint": "https://8kw362bpwg4gx9b2f6e0.us-
east-1.aoss.amazonaws.com",
            "dashboardEndpoint": "https://8kw362bpwg4gx9b2f6e0.us-
east-1.aoss.amazonaws.com/_dashboards"
        }
    ],
    "collectionErrorDetails": []
}
```

> **Note**
>
> Vous ne pourrez pas voir le point de terminaison de la collection tant que le statut de la collection ne sera pas passé à `ACTIVE`. Vous devrez peut-être effectuer plusieurs appels pour vérifier le statut jusqu'à ce que la collection soit correctement créée.

8. Utilisez un outil HTTP tel que Postman ou curl pour indexer les données dans la collection books. Nous allons créer un index appelé books-index et ajouter un seul document.

   Envoyez la requête suivante au point de terminaison de collection que vous avez récupéré à l'étape précédente, à l'aide des informations d'identification de `TutorialRole`.

   ```
   PUT https://8kw362bpwg4gx9b2f6e0.us-east-1.aoss.amazonaws.com/books-index/_doc/1
   {
     "title": "The Shining",
     "author": "Stephen King",
     "year": 1977
   }
   ```

   Exemple de réponse

   ```
   {
     "_index" : "books-index",
     "_id" : "1",
     "_version" : 1,
     "result" : "created",
     "_shards" : {
   ```

```
    "total" : 0,
    "successful" : 0,
    "failed" : 0
  },
  "_seq_no" : 0,
  "_primary_term" : 0
}
```

9. Pour commencer à rechercher des données dans votre collection, utilisez l'API de recherche. La requête suivante permet d'effectuer une recherche de base :

```
GET https://8kw362bpwg4gx9b2f6e0.us-east-1.aoss.amazonaws.com/books-index/_search
```

Exemple de réponse

```
{
    "took": 405,
    "timed_out": false,
    "_shards": {
        "total": 6,
        "successful": 6,
        "skipped": 0,
        "failed": 0
    },
    "hits": {
        "total": {
            "value": 2,
            "relation": "eq"
        },
        "max_score": 1.0,
        "hits": [
            {
                "_index": "books-index:0::3xJq14MBUaOS0wL26UU9:0",
                "_id": "F_bt4oMBLle5pYmm5q4T",
                "_score": 1.0,
                "_source": {
                    "title": "The Shining",
                    "author": "Stephen King",
                    "year": 1977
                }
            }
        ]
    }
```

```
}
```

Identity and Access Management pour Amazon OpenSearch Serverless

AWS Identity and Access Management (IAM) est un outil Service AWS qui permet à un administrateur de contrôler en toute sécurité l'accès aux AWS ressources. Les administrateurs IAM contrôlent qui peut être authentifié (connecté) et autorisé (autorisé) à utiliser les ressources OpenSearch sans serveur. IAM est un Service AWS outil que vous pouvez utiliser sans frais supplémentaires.

Rubriques

- Politiques basées sur l'identité pour Serverless OpenSearch
- Actions stratégiques pour OpenSearch Serverless
- Ressources relatives aux politiques pour le mode OpenSearch Serverless
- Clés de conditions de politique pour Amazon OpenSearch Serverless
- ABAC avec Serverless OpenSearch
- Utilisation d'informations d'identification temporaires avec OpenSearch Serverless
- Rôles liés à un service pour Serverless OpenSearch
- Autres types de politique
- Exemples de politiques basées sur l'identité pour Serverless OpenSearch
- Support d'IAM Identity Center pour Amazon Serverless OpenSearch

Politiques basées sur l'identité pour Serverless OpenSearch

Prend en charge les politiques basées sur l’identité : oui

Les politiques basées sur l’identité sont des documents de politique d’autorisations JSON que vous pouvez attacher à une identité telle qu’un utilisateur, un groupe d’utilisateurs ou un rôle IAM. Ces politiques contrôlent quel type d’actions des utilisateurs et des rôles peuvent exécuter, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l’identité, consultez Définition d’autorisations IAM personnalisées avec des politiques gérées par le client dans le Guide de l’utilisateur IAM.

Avec les politiques IAM basées sur l’identité, vous pouvez spécifier des actions et ressources autorisées ou refusées, ainsi que les conditions dans lesquelles les actions sont autorisées ou

refusées. Vous ne pouvez pas spécifier le principal dans une politique basée sur une identité, car celle-ci s'applique à l'utilisateur ou au rôle auquel elle est attachée. Pour découvrir tous les éléments que vous utilisez dans une politique JSON, consultez Références des éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.

Exemples de politiques basées sur l'identité pour Serverless OpenSearch

Pour consulter des exemples de politiques basées sur l'identité OpenSearch sans serveur, consultez. the section called "Exemples de politiques basées sur l'identité"

Actions stratégiques pour OpenSearch Serverless

Prend en charge les actions de politique : oui

L'élément `Action` d'une politique JSON décrit les actions que vous pouvez utiliser pour autoriser ou refuser l'accès à une politique. Les actions de stratégie portent généralement le même nom que l'opération AWS d'API associée. Il existe quelques exceptions, telles que les actions avec autorisations uniquement qui n'ont pas d'opération API correspondante. Certaines opérations nécessitent également plusieurs actions dans une politique. Ces actions supplémentaires sont nommées actions dépendantes.

Intégration d'actions dans une politique afin d'accorder l'autorisation d'exécuter les opérations associées.

Les actions de stratégie dans OpenSearch Serverless utilisent le préfixe suivant avant l'action :

```
aoss
```

Pour indiquer plusieurs actions dans une seule déclaration, séparez-les par des virgules.

```
"Action": [
      "aoss:action1",
      "aoss:action2"
         ]
```

Vous pouvez préciser plusieurs actions à l'aide de caractères génériques (*). Par exemple, pour spécifier toutes les actions qui commencent par le mot `Describe`, incluez l'action suivante :

```
"Action": "aoss:List*"
```

Pour consulter des exemples de politiques basées sur l'identité OpenSearch sans serveur, consultez. Exemples de politiques basées sur l'identité pour Serverless OpenSearch

Ressources relatives aux politiques pour le mode OpenSearch Serverless

Prend en charge les ressources de politique : oui

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément de politique JSON `Resource` indique le ou les objets auxquels l'action s'applique. Les instructions doivent inclure un élément `Resource` ou `NotResource`. Il est recommandé de définir une ressource à l'aide de son Amazon Resource Name (ARN). Vous pouvez le faire pour des actions qui prennent en charge un type de ressource spécifique, connu sous la dénomination autorisations de niveau ressource.

Pour les actions qui ne sont pas compatibles avec les autorisations de niveau ressource, telles que les opérations de liste, utilisez un caractère générique (*) afin d'indiquer que l'instruction s'applique à toutes les ressources.

```
"Resource": "*"
```

Clés de conditions de politique pour Amazon OpenSearch Serverless

Prend en charge les clés de condition de politique spécifiques au service : oui

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Condition` (ou le bloc `Condition`) vous permet de spécifier des conditions lorsqu'une instruction est appliquée. L'élément `Condition` est facultatif. Vous pouvez créer des expressions conditionnelles qui utilisent des opérateurs de condition, tels que les signes égal ou inférieur à, pour faire correspondre la condition de la politique aux valeurs de la demande.

Si vous spécifiez plusieurs éléments `Condition` dans une instruction, ou plusieurs clés dans un seul élément `Condition`, AWS les évalue à l'aide d'une opération `AND` logique. Si vous spécifiez plusieurs valeurs pour une seule clé de condition, AWS évalue la condition à l'aide d'une `OR`

opération logique. Toutes les conditions doivent être remplies avant que les autorisations associées à l'instruction ne soient accordées.

Vous pouvez aussi utiliser des variables d'espace réservé quand vous spécifiez des conditions. Par exemple, vous pouvez accorder à un utilisateur IAM l'autorisation d'accéder à une ressource uniquement si elle est balisée avec son nom d'utilisateur IAM. Pour plus d'informations, consultez Éléments d'une politique IAM : variables et identifications dans le Guide de l'utilisateur IAM.

AWS prend en charge les clés de condition globales et les clés de condition spécifiques au service. Pour voir toutes les clés de condition AWS globales, voir les clés de contexte de condition AWS globales dans le guide de l'utilisateur IAM.

Outre le contrôle d'accès basé sur les attributs (ABAC), OpenSearch Serverless prend en charge les clés de condition suivantes :

- `aoss:collection`
- `aoss:CollectionId`
- `aoss:index`

Vous pouvez même utiliser ces clés de condition afin de fournir des autorisations relatives aux stratégies d'accès et de sécurité. Par exemple :

```
[
   {
      "Effect":"Allow",
      "Action":[
         "aoss:CreateAccessPolicy",
         "aoss:CreateSecurityPolicy"
      ],
      "Resource":"*",
      "Condition":{
         "StringLike":{
            "aoss:collection":"log"
         }
      }
   }
]
```

Dans cet exemple, la condition s'applique aux stratégies qui contiennent des règles correspondant au nom ou au modèle d'une collection. Les conditions adoptent le comportement suivant :

- `StringEquals` : s'applique aux stratégies dont les règles contiennent la chaîne de ressource exacte « log » (c'est-à-dire `collection/log`).
- `StringLike` : s'applique aux stratégies dont les règles contiennent une chaîne de ressource qui contient la chaîne « log » (c'est-à-dire `collection/log`, mais également `collection/logs-application` ou `collection/applogs123`).

> **Note**
>
> Les clés de condition de collection ne s'appliquent pas au niveau de l'index. Par exemple, dans la stratégie ci-dessus, la condition ne s'appliquerait pas à une stratégie d'accès ou de sécurité contenant la chaîne de ressource `index/logs-application/*`.

Pour consulter la liste des clés de condition OpenSearch sans serveur, consultez la section Clés de condition pour Amazon OpenSearch Serverless dans la référence d'autorisation de service. Pour savoir avec quelles actions et ressources vous pouvez utiliser une clé de condition, consultez Actions définies par Amazon OpenSearch Serverless.

ABAC avec Serverless OpenSearch

Prise en charge d’ABAC (balises dans les politiques) : Oui

Le contrôle d’accès par attributs (ABAC) est une stratégie d’autorisation qui définit des autorisations en fonction des attributs. Dans AWS, ces attributs sont appelés balises. Vous pouvez associer des balises aux entités IAM (utilisateurs ou rôles) et à de nombreuses AWS ressources. L’étiquetage des entités et des ressources est la première étape d’ABAC. Vous concevez ensuite des politiques ABAC pour autoriser des opérations quand l’identification du principal correspond à celle de la ressource à laquelle il tente d’accéder.

L’ABAC est utile dans les environnements qui connaissent une croissance rapide et pour les cas où la gestion des politiques devient fastidieuse.

Pour contrôler l’accès basé sur des étiquettes, vous devez fournir les informations d’étiquette dans l’élément de condition d’une politique utilisant les clés de condition `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`* ou `aws:TagKeys`.

Si un service prend en charge les trois clés de condition pour tous les types de ressources, alors la valeur pour ce service est Oui. Si un service prend en charge les trois clés de condition pour certains types de ressources uniquement, la valeur est Partielle.

Pour plus d'informations sur ABAC, consultez Définition d'autorisations avec l'autorisation ABAC dans le Guide de l'utilisateur IAM. Pour accéder à un didacticiel décrivant les étapes de configuration de l'ABAC, consultez Utilisation du contrôle d'accès par attributs (ABAC) dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur le balisage des ressources OpenSearch sans serveur, consultez. the section called "Baliser des collections"

Utilisation d'informations d'identification temporaires avec OpenSearch Serverless

Prend en charge les informations d'identification temporaires : oui

Certains Services AWS ne fonctionnent pas lorsque vous vous connectez à l'aide d'informations d'identification temporaires. Pour plus d'informations, y compris celles qui Services AWS fonctionnent avec des informations d'identification temporaires, consultez Services AWS la section relative à l'utilisation d'IAM dans le guide de l'utilisateur d'IAM.

Vous utilisez des informations d'identification temporaires si vous vous connectez à l' AWS Management Console aide d'une méthode autre qu'un nom d'utilisateur et un mot de passe. Par exemple, lorsque vous accédez à AWS l'aide du lien d'authentification unique (SSO) de votre entreprise, ce processus crée automatiquement des informations d'identification temporaires. Vous créez également automatiquement des informations d'identification temporaires lorsque vous vous connectez à la console en tant qu'utilisateur, puis changez de rôle. Pour plus d'informations sur le changement de rôle, consultez Passage d'un rôle utilisateur à un rôle IAM (console) dans le Guide de l'utilisateur IAM.

Vous pouvez créer manuellement des informations d'identification temporaires à l'aide de l' AWS API AWS CLI or. Vous pouvez ensuite utiliser ces informations d'identification temporaires pour y accéder AWS. AWS recommande de générer dynamiquement des informations d'identification temporaires au lieu d'utiliser des clés d'accès à long terme. Pour plus d'informations, consultez Informations d'identification de sécurité temporaires dans IAM.

Rôles liés à un service pour Serverless OpenSearch

Prend en charge les rôles liés aux services : Oui

Un rôle lié à un service est un type de rôle de service lié à un. Service AWS Le service peut endosser le rôle afin d'effectuer une action en votre nom. Les rôles liés au service apparaissent dans votre Compte AWS fichier et appartiennent au service. Un administrateur IAM peut consulter, mais ne peut pas modifier, les autorisations concernant les rôles liés à un service.

Pour plus de détails sur la création et la gestion des rôles liés à un service OpenSearch sans serveur, consultez. the section called “Rôle de création d'une collection”

Autres types de politique

AWS prend en charge d'autres types de politiques moins courants. Ces types de politiques peuvent définir le nombre maximum d’autorisations qui vous sont accordées par des types de politiques plus courants.

- Politiques de contrôle des services (SCPs) : SCPs politiques JSON qui spécifient les autorisations maximales pour une organisation ou une unité organisationnelle (UO) dans AWS Organizations. AWS Organizations est un service permettant de regrouper et de gérer de manière centralisée plusieurs AWS comptes détenus par votre entreprise. Si vous activez toutes les fonctionnalités d'une organisation, vous pouvez appliquer des politiques de contrôle des services (SCPs) à l'un ou à l'ensemble de vos comptes. Le SCP limite les autorisations pour les entités des comptes membres, y compris pour chaque utilisateur root AWS du compte. Pour plus d'informations sur les Organizations et consultez SCPs les politiques de contrôle des services dans le Guide de AWS Organizations l'utilisateur.
- Politiques de contrôle des ressources (RCPs) : RCPs politiques JSON que vous pouvez utiliser pour définir le maximum d'autorisations disponibles pour les ressources de vos comptes sans mettre à jour les politiques IAM associées à chaque ressource que vous possédez. Le RCP limite les autorisations pour les ressources des comptes membres et peut avoir un impact sur les autorisations effectives pour les identités, y compris l'utilisateur root du AWS compte, qu'il appartienne ou non à votre organisation. Pour plus d'informations sur les Organizations RCPs, y compris une liste des AWS services compatibles RCPs, voir Politiques de contrôle des ressources (RCPs) dans le guide de AWS Organizations l'utilisateur.

Exemples de politiques basées sur l'identité pour Serverless OpenSearch

Par défaut, les utilisateurs et les rôles ne sont pas autorisés à créer ou à modifier des ressources OpenSearch sans serveur. Ils ne peuvent pas non plus effectuer de tâches à l'aide de l'API AWS Management Console, AWS Command Line Interface (AWS CLI) ou de AWS l'API. Pour octroyer aux utilisateurs des autorisations d’effectuer des actions sur les ressources dont ils ont besoin, un administrateur IAM peut créer des politiques IAM. L’administrateur peut ensuite ajouter les politiques IAM aux rôles et les utilisateurs peuvent assumer les rôles.

Pour apprendre à créer une politique basée sur l’identité IAM à l’aide de ces exemples de documents de politique JSON, consultez Création de politiques IAM (console) dans le Guide de l’utilisateur IAM.

Pour plus de détails sur les actions et les types de ressources définis par Amazon OpenSearch Serverless, y compris le format du ARNs pour chacun des types de ressources, consultez la section Actions, ressources et clés de condition pour Amazon OpenSearch Serverless dans la référence d'autorisation de service.

Rubriques

- Bonnes pratiques en matière de stratégies
- Utilisation de OpenSearch Serverless dans la console
- Administration des OpenSearch collections sans serveur
- Affichage de OpenSearch collections sans serveur
- Utilisation des opérations OpenSearch d'API
- ABAC pour les opérations d' OpenSearch API

Bonnes pratiques en matière de stratégies

Les politiques basées sur l'identité sont très puissantes. Ils déterminent si quelqu'un peut créer, accéder ou supprimer des ressources OpenSearch sans serveur dans votre compte. Ces actions peuvent entraîner des frais pour votre Compte AWS. Lorsque vous créez ou modifiez des politiques basées sur l'identité, suivez ces instructions et recommandations :

Les politiques basées sur l'identité déterminent si quelqu'un peut créer, accéder ou supprimer des ressources OpenSearch sans serveur dans votre compte. Ces actions peuvent entraîner des frais pour votre Compte AWS. Lorsque vous créez ou modifiez des politiques basées sur l'identité, suivez ces instructions et recommandations :

- Commencez AWS par les politiques gérées et passez aux autorisations du moindre privilège : pour commencer à accorder des autorisations à vos utilisateurs et à vos charges de travail, utilisez les politiques AWS gérées qui accordent des autorisations pour de nombreux cas d'utilisation courants. Ils sont disponibles dans votre Compte AWS. Nous vous recommandons de réduire davantage les autorisations en définissant des politiques gérées par les AWS clients spécifiques à vos cas d'utilisation. Pour plus d'informations, consultez politiques gérées par AWS ou politiques gérées par AWS pour les activités professionnelles dans le Guide de l'utilisateur IAM.
- Accordez les autorisations de moindre privilège : lorsque vous définissez des autorisations avec des politiques IAM, accordez uniquement les autorisations nécessaires à l'exécution d'une seule tâche. Pour ce faire, vous définissez les actions qui peuvent être entreprises sur des ressources spécifiques dans des conditions spécifiques, également appelées autorisations de moindre

privilège. Pour plus d'informations sur l'utilisation d'IAM pour appliquer des autorisations, consultez politiques et autorisations dans IAM dans le Guide de l'utilisateur IAM.

- Utilisez des conditions dans les politiques IAM pour restreindre davantage l'accès : vous pouvez ajouter une condition à vos politiques afin de limiter l'accès aux actions et aux ressources. Par exemple, vous pouvez écrire une condition de politique pour spécifier que toutes les demandes doivent être envoyées via SSL. Vous pouvez également utiliser des conditions pour accorder l'accès aux actions de service si elles sont utilisées par le biais d'un service spécifique Service AWS, tel que AWS CloudFormation. Pour plus d'informations, consultez Conditions pour éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.
- Utilisez l'Analyseur d'accès IAM pour valider vos politiques IAM afin de garantir des autorisations sécurisées et fonctionnelles : l'Analyseur d'accès IAM valide les politiques nouvelles et existantes de manière à ce que les politiques IAM respectent le langage de politique IAM (JSON) et les bonnes pratiques IAM. IAM Access Analyzer fournit plus de 100 vérifications de politiques et des recommandations exploitables pour vous aider à créer des politiques sécurisées et fonctionnelles. Pour plus d'informations, consultez Validation de politiques avec IAM Access Analyzer dans le Guide de l'utilisateur IAM.
- Exiger l'authentification multifactorielle (MFA) : si vous avez un scénario qui nécessite des utilisateurs IAM ou un utilisateur root, activez l'authentification MFA pour une sécurité accrue. Compte AWS Pour exiger la MFA lorsque des opérations d'API sont appelées, ajoutez des conditions MFA à vos politiques. Pour plus d'informations, consultez Sécurisation de l'accès aux API avec MFA dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur les bonnes pratiques dans IAM, consultez Bonnes pratiques de sécurité dans IAM dans le Guide de l'utilisateur IAM.

Utilisation de OpenSearch Serverless dans la console

Pour accéder à OpenSearch Serverless depuis la console de OpenSearch service, vous devez disposer d'un ensemble minimal d'autorisations. Ces autorisations doivent vous permettre de répertorier et d'afficher les détails relatifs aux ressources OpenSearch sans serveur de votre AWS compte. Si vous créez une stratégie basée sur l'identité qui est plus restrictive que les autorisations minimales requises, la console ne fonctionnera pas comme prévu pour les entités (telles que les rôles IAM) de cette stratégie.

Il n'est pas nécessaire d'accorder des autorisations de console minimales aux utilisateurs qui appellent uniquement l'API AWS CLI ou l' AWS API. Autorisez plutôt l'accès à uniquement aux actions qui correspondent à l'opération d'API que vous tentez d'effectuer.

La politique suivante permet à un utilisateur d'accéder à OpenSearch Serverless dans la console OpenSearch de service :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Resource": "*",
            "Effect": "Allow",
            "Action": [
                "aoss:ListCollections",
                "aoss:BatchGetCollection",
                "aoss:ListAccessPolicies",
                "aoss:ListSecurityConfigs",
                "aoss:ListSecurityPolicies",
                "aoss:ListTagsForResource",
                "aoss:ListVpcEndpoints",
                "aoss:GetAccessPolicy",
                "aoss:GetAccountSettings",
                "aoss:GetSecurityConfig",
                "aoss:GetSecurityPolicy"
            ]
        }
    ]
}
```

Administration des OpenSearch collections sans serveur

Cette politique est un exemple de politique « d'administration des collections » qui permet à un utilisateur de gérer et d'administrer des collections Amazon OpenSearch Serverless. L'utilisateur peut créer, consulter et supprimer des collections.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Resource": "arn:aws:aoss:region:123456789012:collection/*",
            "Action": [
                "aoss:CreateCollection",
                "aoss:DeleteCollection",
                "aoss:UpdateCollection"
            ],
            "Effect": "Allow"
        },
        {
            "Resource": "*",
            "Action": [
                "aoss:BatchGetCollection",
                "aoss:ListCollections",
                "aoss:CreateAccessPolicy",
                "aoss:CreateSecurityPolicy"
            ],
            "Effect": "Allow"
        }
    ]
}
```

Affichage de OpenSearch collections sans serveur

Cet exemple de politique permet à un utilisateur de consulter les détails de toutes les collections Amazon OpenSearch Serverless de son compte. L'utilisateur ne peut pas modifier les collections ni les stratégies de sécurité associées.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Resource": "*",
            "Action": [
                "aoss:ListAccessPolicies",
                "aoss:ListCollections",
                "aoss:ListSecurityPolicies",
                "aoss:ListTagsForResource",
                "aoss:BatchGetCollection"
            ],
```

```
            "Effect": "Allow"
        }
    ]
}
```

Utilisation des opérations OpenSearch d'API

Les opérations d'API du plan de données comprennent les fonctions que vous utilisez dans OpenSearch Serverless pour obtenir de la valeur en temps réel du service. Les opérations de l'API du plan de contrôle comprennent les fonctions que vous utilisez pour configurer l'environnement.

Pour accéder au plan de données APIs et aux OpenSearch tableaux de bord Amazon OpenSearch Serverless depuis le navigateur, vous devez ajouter deux autorisations IAM pour les ressources de collecte. Ces autorisations sont `aoss:APIAccessAll` et`aoss:DashboardsAccessAll`.

> **Note**
>
> À compter du 10 mai 2023, OpenSearch Serverless aura besoin de ces deux nouvelles autorisations IAM pour les ressources de collecte. L'`aoss:APIAccessAll`autorisation autorise l'accès au plan de données et l'`aoss:DashboardsAccessAll`autorisation autorise les OpenSearch tableaux de bord depuis le navigateur. L'échec de l'ajout des deux nouvelles autorisations IAM entraîne une erreur 403.

Cet exemple de politique permet à un utilisateur d'accéder au plan de données APIs pour une collection spécifiée dans son compte et d'accéder aux OpenSearch tableaux de bord pour toutes les collections de son compte.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
         {
            "Effect": "Allow",
            "Action": "aoss:APIAccessAll",
            "Resource": "arn:aws:aoss:region:account-id:collection/collection-id"
        },
        {
```

```
            "Effect": "Allow",
            "Action": "aoss:DashboardsAccessAll",
            "Resource": "arn:aws:aoss:region:account-id:dashboards/default"
        }
    ]
}
```

Dans `aoss:APIAccessAll` les deux cas, `aoss:DashboardsAccessAll` accordez une autorisation IAM complète aux ressources de collection, tandis que l'autorisation Dashboards fournit également un accès aux OpenSearch Dashboards. Chaque autorisation fonctionne indépendamment, de sorte qu'un refus explicite `aoss:APIAccessAll` ne bloque pas l'`aoss:DashboardsAccessAll`accès aux ressources, y compris aux outils de développement. Il en va de même pour le démenti`aoss:DashboardsAccessAll`. OpenSearch Serverless prend en charge les clés de condition globales suivantes :

- `aws:CalledVia`
- `aws:CalledViaAWSService`
- `aws:CalledViaFirst`
- `aws:CalledViaLast`
- `aws:CurrentTime`
- `aws:EpochTime`
- `aws:PrincipalAccount`
- `aws:PrincipalArn`
- `aws:PrincipallsAWSService`
- `aws:PrincipalOrgID`
- `aws:PrincipalOrgPaths`
- `aws:PrincipalType`
- `aws:PrincipalServiceName`
- `aws:PrincipalServiceNamesList`
- `aws:ResourceAccount`
- `aws:ResourceOrgID`
- `aws:ResourceOrgPaths`
- `aws:RequestedRegion`

- aws:ResourceTag
- aws:SourceIp
- aws:SourceVpce
- aws:SourceVpc
- aws:userid
- aws:username

Voici un exemple d'utilisation du bloc aws:SourceIp conditionnel de la politique IAM de votre principal pour les appels du plan de données :

```
"Condition": {
    "IpAddress": {
         "aws:SourceIp": "203.0.113.0"
    }
}
```

Voici un exemple d'utilisation du bloc aws:SourceVpc conditionnel de la politique IAM de votre principal pour les appels du plan de données :

```
"Condition": {
    "StringEquals": {
        "aws:SourceVpc": "vpc-0fdd2445d8EXAMPLE"
    }
}
```

En outre, un support est proposé pour les clés spécifiques à OpenSearch Serverless suivantes :

- aoss:CollectionId
- aoss:collection

Voici un exemple d'utilisation du bloc aoss:collection conditionnel de la politique IAM de votre principal pour les appels du plan de données :

```
"Condition": {
    "StringLike": {
         "aoss:collection": "log-*"
    }
```

```
}
```

ABAC pour les opérations d' OpenSearch API

Les politiques basées sur l'identité vous permettent d'utiliser des balises pour contrôler l'accès au plan de données Amazon OpenSearch Serverless. APIs La politique suivante est un exemple permettant aux principaux attachés d'accéder au plan de données APIs si la collection possède la `team:devops` balise :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "aoss:APIAccessAll",
            "Resource": "arn:aws:aoss:region:account-id:collection/collection-
id",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/team": "devops"
                }
            }
        }
    ]
}
```

La politique suivante est un exemple de refus aux principaux attachés d'accéder au plan de données APIs et aux tableaux de bord si la collection possède le `environment:production` tag :

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "aoss:APIAccessAll",
                "aoss:DashboardsAccessAll",
            ],
```

```
            "Resource": "arn:aws:aoss:region:account-id:collection/collection-id",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/environment": "production"
                }
            }
        }
    ]
}
```

Amazon OpenSearch Serverless ne prend pas en charge `RequestTag` les clés de condition `TagKeys` globales pour le plan APIs de données.

Support d'IAM Identity Center pour Amazon Serverless OpenSearch

Support d'IAM Identity Center pour Amazon Serverless OpenSearch

Vous pouvez utiliser les principes du centre d'identité IAM (utilisateurs et groupes) pour accéder aux données Amazon OpenSearch Serverless via Amazon Applications. OpenSearch Pour activer la prise en charge d'IAM Identity Center pour Amazon OpenSearch Serverless, vous devez activer l'utilisation d'IAM Identity Center. Pour en savoir plus sur la procédure à suivre, consultez Qu'est-ce qu'IAM Identity Center ?

Une fois l'instance IAM Identity Center créée, l'administrateur du compte client doit créer une application IAM Identity Center pour le service Amazon OpenSearch Serverless. Cela peut être fait en appelant le CreateSecurityConfig:. L'administrateur du compte client peut spécifier les attributs qui seront utilisés pour autoriser la demande. Les attributs par défaut utilisés sont `UserId` et `GroupId`.

L'intégration du centre d'identité IAM pour Amazon OpenSearch Serverless utilise les autorisations AWS IAM Identity Center (IAM) suivantes :

- `aoss:CreateSecurityConfig`— Créez un fournisseur de centre d'identité IAM
- `aoss:ListSecurityConfig`— Répertoriez tous les fournisseurs IAM Identity Center du compte courant.
- `aoss:GetSecurityConfig`— Afficher les informations du fournisseur IAM Identity Center.
- `aoss:UpdateSecurityConfig`— Modifie une configuration IAM Identity Center donnée
- `aoss:DeleteSecurityConfig`— Supprimez un fournisseur de centre d'identité IAM.

La politique d'accès basée sur l'identité suivante peut être utilisée pour gérer toutes les configurations d'IAM Identity Center :

JSON

```
{
"Version": "2012-10-17",
    "Statement": [
        {
"Action": [
                "aoss:CreateSecurityConfig",
                "aoss:DeleteSecurityConfig",
                "aoss:GetSecurityConfig",
                "aoss:UpdateSecurityConfig",
                "aoss:ListSecurityConfigs"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

> **Note**
>
> L'`Resource`élément doit être un caractère générique.

Création d'un fournisseur de centre d'identité IAM (console)

Vous pouvez créer un fournisseur de centre d'identité IAM pour activer l'authentification avec l' OpenSearchapplication. Pour activer l'authentification IAM Identity Center pour les OpenSearch tableaux de bord, effectuez les opérations suivantes :

1. Connectez-vous à la console Amazon OpenSearch Service.
2. Dans le panneau de navigation de gauche, développez Serverless et choisissez Authentication.
3. Choisissez l'authentification IAM Identity Center.
4. Sélectionnez Modifier
5. Cochez la case à côté de Authentifier auprès d'IAM Identity Center.

6. Sélectionnez la clé d'attribut de l'utilisateur et du groupe dans le menu déroulant. Les attributs utilisateur seront utilisés pour autoriser les utilisateurs en fonction de `UserName``UserId`, et`Email`. Les attributs de groupe seront utilisés pour authentifier les utilisateurs en fonction de `GroupName` et`GroupId`.
7. Sélectionnez l'instance IAM Identity Center.
8. Sélectionnez Enregistrer

Création d'un fournisseur de centre d'identité IAM ()AWS CLI

Pour créer un fournisseur de centre d'identité IAM à l'aide de AWS Command Line Interface (AWS CLI), utilisez la commande suivante :

```
aws opensearchserverless create-security-config \
--region us-east-2 \
--name "iamidentitycenter-config" \
--description "description" \
--type "iamidentitycenter" \
--iam-identity-center-options '{
    "instanceArn": "arn:aws:sso:::instance/ssoins-99199c99e99ee999",
    "userAttribute": "UserName",
    "groupAttribute": "GroupId"
}'
```

Une fois qu'un centre d'identité IAM est activé, les clients peuvent uniquement modifier les attributs des utilisateurs et des groupes.

```
aws opensearchserverless update-security-config \
--region us-east-1 \
--id <id_from_list_security_configs> \
--config-version <config_version_from_get_security_config> \
--iam-identity-center-options-updates '{
    "userAttribute": "UserId",
    "groupAttribute": "GroupId"
}'
```

Pour afficher le fournisseur du centre d'identité IAM à l'aide de AWS Command Line Interface, utilisez la commande suivante :

```
aws opensearchserverless list-security-configs --type iamidentitycenter
```

Supprimer un fournisseur de centre d'identité IAM

IAM Identity Center propose deux instances de fournisseurs, l'une pour le compte de votre organisation et l'autre pour votre compte de membre. Si vous devez modifier votre instance IAM Identity Center, vous devez supprimer votre configuration de sécurité via l'`DeleteSecurityConfig`API et créer une nouvelle configuration de sécurité à l'aide de la nouvelle instance IAM Identity Center. La commande suivante peut être utilisée pour supprimer un fournisseur IAM Identity Center :

```
aws opensearchserverless delete-security-config \
--region us-east-1 \
--id <id_from_list_security_configs>
```

Accorder à IAM Identity Center l'accès aux données de collecte

Une fois que votre fournisseur de centre d'identité IAM est activé, vous pouvez mettre à jour la politique d'accès aux données de collecte afin d'inclure les principes du centre d'identité IAM. Les principes du IAM Identity Center doivent être mis à jour dans le format suivant :

```
[
   {
"Rules":[
       ...
      ],
      "Principal":[
         "iamidentitycenter/<iamidentitycenter-instance-id>/user/<UserName>",
         "iamidentitycenter/<iamidentitycenter-instance-id>/group/<GroupId>"
      ]
   }
]
```

> **Note**
>
> Amazon OpenSearch Serverless ne prend en charge qu'une seule instance IAM Identity Center pour toutes les collections de clients et peut prendre en charge jusqu'à 100 groupes pour un seul utilisateur. Si vous essayez d'utiliser un nombre d'instances supérieur au nombre autorisé, vous rencontrerez une incohérence dans le traitement des autorisations de votre politique d'accès aux données et vous recevrez un message `403` d'erreur.

Vous pouvez octroyer l'accès aux collections, aux index ou aux deux. Si vous souhaitez que différents utilisateurs disposent d'autorisations différentes, vous devez créer plusieurs règles. Pour obtenir la liste des autorisations disponibles, consultez Identity and Access Management in Amazon OpenSearch Service. Pour plus d'informations sur le formatage d'une politique d'accès, consultez la section Accorder aux identités SAML l'accès aux données de collecte.

IAM Identity Center propose deux instances de fournisseurs, l'une pour le compte de votre organisation et l'autre pour votre compte de membre. Si vous devez modifier votre instance IAM Identity Center, vous devez supprimer votre configuration de sécurité via l'`DeleteSecurityConfig`API et créer une nouvelle configuration de sécurité à l'aide de la nouvelle instance IAM Identity Center. La commande suivante peut être utilisée pour supprimer un fournisseur IAM Identity Center :

```
aws opensearchserverless delete-security-config \
--region us-east-1 \
--id <id_from_list_security_configs>
```

Chiffrement dans Amazon OpenSearch Serverless

Chiffrement au repos

Chaque collection Amazon OpenSearch Serverless que vous créez est protégée par le chiffrement des données au repos, une fonctionnalité de sécurité qui permet d'empêcher tout accès non autorisé à vos données. Encryption at rest utilise AWS Key Management Service (AWS KMS) pour stocker et gérer vos clés de chiffrement. Il utilise l'algorithme Advanced Encryption Standard avec des clés 256 bits (AES-256) afin de procéder au chiffrement.

Rubriques

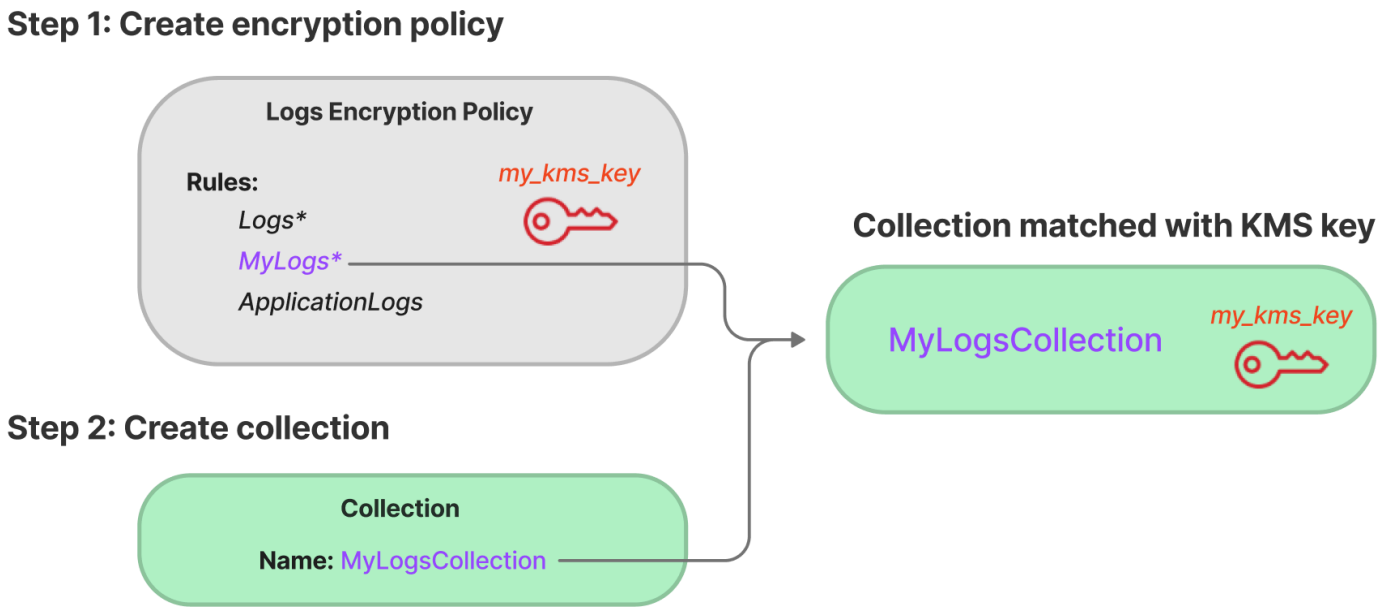
- Stratégies de chiffrement
- Considérations
- Autorisations nécessaires
- Politique de clé pour une clé gérée par le client
- Comment OpenSearch Serverless utilise les subventions dans AWS KMS
- Créer des stratégies de chiffrement (console)
- Créer des stratégies de chiffrement (AWS CLI)
- Affichage des stratégies de chiffrement

- Mise à jour des stratégies de chiffrement
- Supprimer des stratégies de chiffrement

Stratégies de chiffrement

Les stratégies de chiffrement vous permettent de gérer de nombreuses collections à grande échelle en attribuant automatiquement une clé de chiffrement aux collections nouvellement créées qui correspondent à un nom ou à un modèle spécifique.

Lorsque vous créez une stratégie de chiffrement, vous pouvez soit spécifier un préfixe, qui est une règle de correspondance basée sur des caractères génériques, par exemple `MyCollection*`, soit saisir un nom de collection unique. Ensuite, lorsque vous créez une collection correspondant à ce nom ou à ce modèle de préfixe, la stratégie et la clé KMS correspondante lui sont automatiquement attribuées.



Les stratégies de chiffrement comportent les éléments suivants :

- `Rules` : une ou plusieurs règles de correspondance des collections, chacune comportant les sous-éléments suivants :
  - `ResourceType` : à l'heure actuelle, la seule option est « collection ». Les stratégies de chiffrement s'appliquent uniquement aux ressources de collection.
  - `Resource` : un ou plusieurs noms ou modèles de collection auxquels la stratégie s'appliquera, au format `collection/<collection name|pattern>`.

- `AWSOwnedKey` : si une Clé détenue par AWS doit être utilisée.
- `KmsARN` : si vous définissez `AWSOwnedKey` sur false (faux), spécifiez l'Amazon Resource Name (ARN) de la clé KMS avec laquelle chiffrer les collections associées. Si vous incluez ce paramètre, OpenSearch Serverless l'`AWSOwnedKey`ignore.

L'exemple de stratégie suivant attribuera une clé gérée par le client à toute collection future nommée `autopartsinventory`, ainsi qu'aux collections commençant par le terme « sales » (ventes) :

```
{
   "Rules":[
      {
         "ResourceType":"collection",
         "Resource":[
            "collection/autopartsinventory",
            "collection/sales*"
         ]
      }
   ],
   "AWSOwnedKey":false,
   "KmsARN":"arn:aws:encryption:us-east-1:123456789012:key/93fd6da4-a317-4c17-
bfe9-382b5d988b36"
}
```

Même si une stratégie correspond au nom d'une collection, vous pouvez choisir de remplacer cette attribution automatique lors de la création de la collection si le modèle de ressource contient un caractère générique (*). Si vous choisissez de remplacer l'attribution automatique des clés, OpenSearch Serverless crée pour vous une politique de chiffrement nommée auto-**< *collection-name* >** et l'attache à la collection. La stratégie ne s'applique initialement qu'à une seule collection, mais vous pouvez la modifier pour inclure des collections supplémentaires.

Si vous modifiez les règles de stratégie pour qu'elles ne correspondent plus à une collection, la clé KMS associée ne sera pas dissociée de cette collection. La collection reste toujours chiffrée à l'aide de sa clé de chiffrement initiale. Si vous souhaitez modifier la clé de chiffrement d'une collection, vous devez recréer la collection.

Si les règles de plusieurs stratégies correspondent à une collection, la règle la plus spécifique est utilisée. Par exemple, si une stratégie contient une règle pour `collection/log*` et une autre pour `collection/logSpecial`, la clé de chiffrement de la seconde stratégie est utilisée, car elle est plus spécifique.

Vous ne pouvez pas utiliser de nom ou de préfixe dans une politique s'il existe déjà dans une autre stratégie. OpenSearch Serverless affiche une erreur si vous essayez de configurer des modèles de ressources identiques dans différentes politiques de chiffrement.

Considérations

Tenez compte des éléments suivants lorsque vous configurez le chiffrement de vos collections :

- Le chiffrement au repos est requis pour toutes les collections sans serveur.
- Vous avez la possibilité d'utiliser une clé gérée par le client ou une Clé détenue par AWS. Si vous choisissez une clé gérée par le client, nous vous recommandons d'activer la rotation automatique des clés.
- Vous ne pouvez pas modifier la clé de chiffrement d'une collection après la création de la collection. Choisissez soigneusement celle que vous AWS KMS souhaitez utiliser la première fois que vous configurez une collection.
- Une collection ne peut correspondre qu'à une seule stratégie de chiffrement.
- Les collections dotées de clés KMS uniques ne peuvent pas partager d'unités de OpenSearch calcul (OCUs) avec d'autres collections. Chaque collection avec une clé unique nécessite ses propres 4 clés OCUs.
- Si vous mettez à jour la clé KMS dans une stratégie de chiffrement, la modification n'affecte pas les collections correspondantes existantes auxquelles des clés KMS ont déjà été attribuées.
- OpenSearch Serverless ne vérifie pas explicitement les autorisations des utilisateurs sur les clés gérées par le client. Si un utilisateur est autorisé à accéder à une collection par le biais d'une stratégie d'accès aux données, il pourra ingérer et interroger les données chiffrées à l'aide de la clé associée.

Autorisations nécessaires

Le chiffrement au repos pour OpenSearch Serverless utilise les autorisations AWS Identity and Access Management (IAM) suivantes. Vous pouvez spécifier des conditions IAM pour restreindre les utilisateurs à des collections spécifiques.

- `aoss:CreateSecurityPolicy` : créer une stratégie de chiffrement.
- `aoss:ListSecurityPolicies` : répertorier toutes les stratégies de chiffrements et collections auxquelles elles sont associées.
- `aoss:GetSecurityPolicy` : consulter les détails d'une stratégie de chiffrement spécifique.

- `aoss:UpdateSecurityPolicy` : modifier une stratégie de chiffrement.
- `aoss:DeleteSecurityPolicy` : supprimer une stratégie de chiffrement.

L'exemple de stratégie d'accès basée sur l'identité suivant fournit les autorisations minimales nécessaires à un utilisateur pour gérer les stratégies de chiffrement à l'aide du modèle de ressource `collection/application-logs`.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "aoss:CreateSecurityPolicy",
            "aoss:UpdateSecurityPolicy",
            "aoss:DeleteSecurityPolicy",
            "aoss:GetSecurityPolicy"
         ],
         "Resource":"*",
         "Condition":{
            "StringEquals":{
               "aoss:collection":"application-logs"
            }
         }
      },
      {
         "Effect":"Allow",
         "Action":[
            "aoss:ListSecurityPolicies"
         ],
         "Resource":"*"
      }
   ]
}
```

Politique de clé pour une clé gérée par le client

Si vous sélectionnez une clé gérée par le client pour protéger une collection, OpenSearch Serverless obtient l'autorisation d'utiliser la clé KMS au nom du principal qui effectue la sélection. Ce principal, qu'il s'agisse d'un utilisateur ou d'un rôle, doit disposer des autorisations requises par OpenSearch Serverless sur la clé KMS. Vous pouvez fournir ces autorisations dans une stratégie de clé ou une politique IAM.

OpenSearch Serverless nécessite au minimum les autorisations suivantes sur une clé gérée par le client :

- km : DescribeKey
- km : CreateGrant

Par exemple :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:DescribeKey",
        "kms:CreateGrant"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "aoss.us-east-1.amazonaws.com"
        },
        "Bool": {
          "kms:GrantIsForAWSResource": "true"
        }
      }
    }
  ]
}
```

OpenSearch Serverless crée une subvention avec les autorisations kms : GenerateDataKey et KMS:Decrypt.

Pour de plus amples informations, veuillez consulter Utilisation des politiques de clé AWS KMS dans le AWS Key Management Service Guide du développeur.

Comment OpenSearch Serverless utilise les subventions dans AWS KMS

OpenSearch Serverless nécessite une autorisation pour utiliser une clé gérée par le client.

Lorsque vous créez une politique de chiffrement dans votre compte avec une nouvelle clé, OpenSearch Serverless crée une subvention en votre nom en envoyant une CreateGrantdemande à AWS KMS. AWS KMS Les autorisations sont utilisées pour donner un accès OpenSearch sans serveur à une clé KMS dans un compte client.

OpenSearch Serverless nécessite l'autorisation d'utiliser votre clé gérée par le client pour les opérations internes suivantes :

- Envoyez DescribeKeydes demandes AWS KMS à pour vérifier que l'ID de clé symétrique géré par le client fourni est valide.
- Envoyez GenerateDataKeydes demandes à KMS Key pour créer des clés de données avec lesquelles chiffrer des objets.
- Envoyez des demandes de déchiffrement AWS KMS à pour déchiffrer les clés de données chiffrées afin qu'elles puissent être utilisées pour chiffrer vos données.

Vous pouvez révoquer l'accès à l'octroi ou supprimer l'accès du service à la clé gérée par le client à tout moment. Si vous le faites, OpenSearch Serverless ne pourra accéder à aucune des données chiffrées par la clé gérée par le client, ce qui affectera toutes les opérations qui dépendent de ces données, entraînant des `AccessDeniedException` erreurs et des échecs dans les flux de travail asynchrones.

OpenSearch Serverless retire les autorisations dans un flux de travail asynchrone lorsqu'une clé gérée par le client n'est associée à aucune politique de sécurité ou à aucune collection.

Créer des stratégies de chiffrement (console)

Dans une stratégie de chiffrement, vous spécifiez une clé KMS et une série de modèles de collection auxquels la stratégie s'appliquera. Toutes les nouvelles collections correspondant à l'un des modèles définis dans la stratégie se verront attribuer les clés KMS correspondantes lors de la création de la

collection. Nous vous recommandons de créer des stratégies de chiffrement avant de commencer à créer des collections.

Pour créer une politique de chiffrement OpenSearch sans serveur

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le panneau de navigation de gauche, développez Serverless (Sans serveur) et choisissez Encryption policies (Stratégies de chiffrement).
3. Choisissez Create encryption policy (Créer une stratégie de chiffrement).
4. Saisissez un nom et une description pour la stratégie.
5. Sous Resources (Ressources), saisissez un ou plusieurs modèles de ressources pour cette stratégie de chiffrement. Toutes les collections nouvellement créées sur le Compte AWS et dans la région actuels qui correspondent à l'un des modèles sont automatiquement attribuées à cette stratégie. Par exemple, si vous saisissez `ApplicationLogs` (sans caractère générique) et que vous créez ultérieurement une collection portant ce nom, la stratégie et la clé KMS correspondante sont attribuées à cette collection.

   Vous pouvez également fournir un préfixe tel que `Logs*`, qui attribue la stratégie à toute nouvelle collection dont le nom commence par `Logs`. En utilisant des caractères génériques, vous pouvez gérer les paramètres de chiffrement de plusieurs collections à grande échelle.
6. Sous Encryption (Chiffrement), choisissez une clé KMS à utiliser.
7. Choisissez Créer.

Étape suivante : créer des collections

Après avoir configuré une ou plusieurs stratégies de chiffrement, vous pouvez commencer à créer des collections qui correspondent aux règles définies dans ces stratégies. Pour obtenir des instructions, veuillez consulter the section called “Créer des collections”.

À l'étape Chiffrements de la création de la collection, OpenSearch Serverless vous informe que le nom que vous avez saisi correspond au modèle défini dans une politique de chiffrement et attribue automatiquement la clé KMS correspondante à la collection. Si le modèle de ressource contient un caractère générique (*), vous pouvez choisir de remplacer la correspondance et de sélectionner votre propre clé.

Créer des stratégies de chiffrement (AWS CLI)

Pour créer une politique de chiffrement à l'aide des opérations de l'API OpenSearch Serverless, vous devez spécifier des modèles de ressources et une clé de chiffrement au format JSON. La CreateSecurityPolicydemande accepte à la fois les politiques intégrées et les fichiers .json.

Les stratégies de chiffrement prennent le format suivant. Cet exemple de fichier `my-policy.json` correspond à toute future collection nommée `autopartsinventory`, ainsi qu'à toutes les collections dont le nom commence par `sales`.

```
{
   "Rules":[
      {
         "ResourceType":"collection",
         "Resource":[
            "collection/autopartsinventory",
            "collection/sales*"
         ]
      }
   ],
   "AWSOwnedKey":false,
   "KmsARN":"arn:aws:encryption:us-east-1:123456789012:key/93fd6da4-a317-4c17-
bfe9-382b5d988b36"
}
```

Pour utiliser une clé appartenant au service, définissez `AWSOwnedKey` sur `true` :

```
{
   "Rules":[
      {
         "ResourceType":"collection",
         "Resource":[
            "collection/autopartsinventory",
            "collection/sales*"
         ]
      }
   ],
   "AWSOwnedKey":true
}
```

La requête suivante crée la stratégie de chiffrement :

```
aws opensearchserverless create-security-policy \
    --name sales-inventory \
    --type encryption \
    --policy file://my-policy.json
```

Utilisez ensuite l'opération CreateCollectionAPI pour créer une ou plusieurs collections correspondant à l'un des modèles de ressources.

Affichage des stratégies de chiffrement

Avant de créer une collection, vous souhaiterez peut-être prévisualiser les stratégies de chiffrement existantes dans votre compte pour voir laquelle possède un modèle de ressource correspondant au nom de votre collection. La ListSecurityPoliciesdemande suivante répertorie toutes les politiques de chiffrement de votre compte :

```
aws opensearchserverless list-security-policies --type encryption
```

La requête renvoie des informations sur toutes les stratégies de chiffrement configurées. Utilisez le contenu de l'élément `policy` pour consulter les règles de modèle définies dans la stratégie :

```
{
   "securityPolicyDetails": [
      {
         "createdDate": 1663693217826,
         "description": "Sample encryption policy",
         "lastModifiedDate": 1663693217826,
         "name": "my-policy",
         "policy": "{\"Rules\":[{\"ResourceType\":\"collection\",\"Resource\":
[\"collection/autopartsinventory\",\"collection/sales*\"]}],\"AWSOwnedKey\":true}",
         "policyVersion": "MTY2MzY5MzIxNzgyNl8x",
         "type": "encryption"
      }
   ]
}
```

Pour afficher des informations détaillées sur une politique spécifique, y compris la clé KMS, utilisez la GetSecurityPolicycommande.

Mise à jour des stratégies de chiffrement

Si vous mettez à jour la clé KMS dans une stratégie de chiffrement, la modification s'applique uniquement aux collections nouvellement créées qui correspondent au nom ou au modèle configuré. Cela n'affecte pas les collections existantes auxquelles des clés KMS ont déjà été attribuées.

Il en va de même des règles de correspondance de stratégie. Si vous ajoutez, modifiez ou supprimez une règle, la modification ne s'applique qu'aux collections nouvellement créées. Les collections existantes ne perdent pas la clé KMS qui leur est attribuée si vous modifiez les règles d'une stratégie afin qu'elle ne corresponde plus au nom d'une collection.

Pour mettre à jour une politique de chiffrement dans la console OpenSearch Serverless, choisissez Politiques de chiffrement, sélectionnez la politique à modifier, puis choisissez Modifier. Effectuez les modifications souhaitées, puis choisissez Save (Enregistrer).

Pour mettre à jour une politique de chiffrement à l'aide de l'API OpenSearch Serverless, utilisez l'UpdateSecurityPolicyopération. La requête suivante met à jour une stratégie de chiffrement avec un nouveau document JSON de stratégie :

```
aws opensearchserverless update-security-policy \
    --name sales-inventory \
    --type encryption \
    --policy-version 2 \
    --policy file://my-new-policy.json
```

Supprimer des stratégies de chiffrement

Lorsque vous supprimez une stratégie de chiffrement, toutes les collections qui utilisent actuellement la clé KMS définie dans la stratégie ne sont pas affectées. Pour supprimer une politique dans la console OpenSearch Serverless, sélectionnez-la, puis choisissez Supprimer.

Vous pouvez également utiliser l'DeleteSecurityPolicyopération :

```
aws opensearchserverless delete-security-policy --name my-policy --type encryption
```

Chiffrement en transit

Dans OpenSearch Serverless, tous les chemins d'une collection sont chiffrés en transit à l'aide du protocole TLS (Transport Layer Security) avec un algorithme de chiffrement AES-256 conforme aux normes du secteur. L'accès à tous APIs et aux tableaux de bord pour Opensearch se fait également

via TLS 1.2. Le protocole TLS est un ensemble de protocoles cryptographiques conformes aux normes de l'industrie utilisés pour chiffrer les informations échangées sur le réseau.

Accès réseau pour Amazon OpenSearch Serverless

Les paramètres réseau d'une collection Amazon OpenSearch Serverless déterminent si la collection est accessible via Internet à partir de réseaux publics ou si elle doit être accessible de manière privée.

L'accès privé peut s'appliquer à l'un des éléments suivants ou aux deux :

- OpenSearch Points de terminaison VPC gérés sans serveur
- Supportés Services AWS , tels qu'Amazon Bedrock

Vous pouvez configurer l'accès au réseau séparément pour le point de OpenSearchterminaison d'une collection et pour le point de terminaison OpenSearch Dashboards correspondant.

L'accès réseau est le mécanisme d'isolation permettant l'accès à partir de différents réseaux source. Par exemple, si le point de terminaison OpenSearch des tableaux de bord d'une collection est accessible au public mais que le point de terminaison de l' OpenSearch API ne l'est pas, un utilisateur peut accéder aux données de la collection uniquement via les tableaux de bord lorsqu'il se connecte depuis un réseau public. S'ils essaient de les appeler OpenSearch APIs directement depuis un réseau public, ils seront bloqués. Les paramètres réseau peuvent être utilisés pour de telles permutations de la source au type de ressource. Amazon OpenSearch Serverless prend en charge les deux, IPv4 ainsi que la IPv6 connectivité.

Rubriques

- Stratégies réseau
- Considérations
- Autorisations requises pour configurer les politiques réseau
- Priorité des stratégies
- Créer des stratégies réseau (console)
- Création de stratégies réseau (AWS CLI)
- Affichage des stratégies réseau
- Mettre à jour des stratégies réseau
- Supprimer des stratégies réseau

Stratégies réseau

Les stratégies réseau vous permettent de gérer de nombreuses collections à grande échelle en attribuant automatiquement des paramètres d'accès réseau aux collections qui correspondent aux règles définies dans la stratégie.

Dans une stratégie réseau, vous spécifiez une série de règles. Ces règles définissent les autorisations d'accès aux points de terminaison de collecte et aux points de terminaison des OpenSearch tableaux de bord. Chaque règle comprend un type d'accès (public ou privé) et un type de ressource (collection et/ou point de terminaison OpenSearch Dashboards). Pour chaque type de ressource (`collection` et `dashboard`), vous spécifiez une série de règles qui définissent à quelles collections la stratégie s'appliquera.

Dans cet exemple de politique, la première règle spécifie l'accès du point de terminaison VPC à la fois au point de terminaison de collecte et au point de terminaison du tableau de bord pour toutes les collections commençant par le terme. `marketing*` Il spécifie également l'accès à Amazon Bedrock.

> **Note**
>
> L'accès privé à Services AWS Amazon Bedrock ne s'applique qu'au point de terminaison de la collection, et non au OpenSearch point de terminaison OpenSearch des tableaux de bord. Même si `ResourceType` c'est le cas`dashboard`, l'accès aux OpenSearch tableaux de bord Services AWS ne peut pas être accordé.

La deuxième règle spécifie l'accès public à la collection `finance`, mais uniquement pour le point de terminaison de la collection (aucun accès aux tableaux de bord).

```
[
   {
      "Description":"Marketing access",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/marketing*"
            ]
         },
         {
            "ResourceType":"dashboard",
```

```
            "Resource":[
               "collection/marketing*"
            ]
         }
      ],
      "AllowFromPublic":false,
      "SourceVPCEs":[
         "vpce-050f79086ee71ac05"
      ],
      "SourceServices":[
         "bedrock.amazonaws.com"
      ],
   },
   {
      "Description":"Sales access",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/finance"
            ]
         }
      ],
      "AllowFromPublic":true
   }
]
```

Cette politique fournit un accès public uniquement aux OpenSearch tableaux de bord pour les collections commençant par « finance ». Toute tentative d'accès direct à l' OpenSearch API échouera.

```
[
  {
    "Description": "Dashboards access",
    "Rules": [
      {
        "ResourceType": "dashboard",
        "Resource": [
          "collection/finance*"
        ]
      }
    ],
    "AllowFromPublic": true
```

```
  }
]
```

Les stratégies réseau peuvent s'appliquer aux collections existantes ainsi qu'aux collections futures. Par exemple, vous pouvez créer une collection, puis créer une stratégie réseau avec une règle correspondant au nom de la collection. Vous n'avez pas besoin de créer des stratégies réseau avant de créer des collections.

Considérations

Tenez compte des éléments suivants lorsque vous configurez l'accès réseau de vos collections :

- Si vous envisagez de configurer l'accès au point de terminaison VPC pour une collection, vous devez d'abord créer au OpenSearch moins un point de terminaison VPC géré sans serveur.
- L'accès privé à Services AWS ne s'applique qu'au point de OpenSearch terminaison de la collection, et non au point de terminaison OpenSearch des tableaux de bord. Même si `ResourceType` c'est le cas`dashboard`, l'accès aux OpenSearch tableaux de bord Services AWS ne peut pas être accordé.
- Si une collection est accessible depuis les réseaux publics, elle est également accessible depuis tous les points de terminaison VPC OpenSearch gérés sans serveur. Services AWS
- Plusieurs stratégies réseau peuvent s'appliquer à une seule collection. Pour de plus amples informations, veuillez consulter the section called “Priorité des stratégies”.

Autorisations requises pour configurer les politiques réseau

L'accès réseau pour OpenSearch Serverless utilise les autorisations AWS Identity and Access Management (IAM) suivantes. Vous pouvez spécifier des conditions IAM pour restreindre les utilisateurs à des stratégies réseau associées à des collections spécifiques.

- `aoss:CreateSecurityPolicy` : créer une stratégie d'accès au réseau.
- `aoss:ListSecurityPolicies` : répertorier toutes les stratégies réseau du compte actuel.
- `aoss:GetSecurityPolicy` : afficher une spécification de stratégie d'accès au réseau.
- `aoss:UpdateSecurityPolicy` : modifier une stratégie d'accès réseau donnée et modifier l'ID du VPC ou la désignation d'accès public.
- `aoss:DeleteSecurityPolicy` : supprimer une stratégie d'accès au réseau (après l'avoir détachée de toutes les collections).

La stratégie d'accès basée sur l'identité suivante permet à un utilisateur de consulter toutes les stratégies réseau et de mettre à jour les stratégies qui contiennent le modèle de ressource `collection/application-logs` :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aoss:UpdateSecurityPolicy"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aoss:collection": "application-logs"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "aoss:ListSecurityPolicies",
                "aoss:GetSecurityPolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

> **Note**
>
> En outre, OpenSearch Serverless nécessite les `aoss:DashboardsAccessAll` autorisations `aoss:APIAccessAll` et pour les ressources de collecte. Pour de plus amples informations, veuillez consulter the section called “Utilisation des opérations OpenSearch d'API”.

Priorité des stratégies

Dans certains cas, les règles de stratégie réseau se chevauchent, au sein des stratégies ou entre elles. Dans ce cas, une règle qui spécifie l'accès public remplace une règle qui spécifie l'accès privé pour toutes les collections communes aux deux règles.

Par exemple, dans la stratégie suivante, les deux règles attribuent un accès réseau à la collection `finance`, mais une règle spécifie l'accès VPC tandis que l'autre spécifie l'accès public. Dans ce cas, l'accès public outrepasse l'accès VPC uniquement pour la collection finance (car il existe dans les deux règles), de sorte que la collection finance sera accessible depuis les réseaux publics. La collection sales (ventes) bénéficiera d'un accès VPC à partir du point de terminaison spécifié.

```
[
   {
      "Description":"Rule 1",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/sales",
               "collection/finance"
            ]
         }
      ],
      "AllowFromPublic":false,
      "SourceVPCEs":[
         "vpce-050f79086ee71ac05"
      ]
   },
   {
      "Description":"Rule 2",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/finance"
            ]
         }
      ],
      "AllowFromPublic":true
   }
]
```

Si plusieurs points de terminaison d'un VPC issus de règles différentes s'appliquent à une collection, les règles s'additionnent et la collection sera accessible depuis tous les points de terminaison spécifiés. Si vous définissez `AllowFromPublic` `true` mais fournissez également un ou plusieurs `SourceVPCEs` ou`SourceServices`, OpenSearch Serverless ignore les points de terminaison et les identifiants de service VPC, et les collections associées seront accessibles au public.

Créer des stratégies réseau (console)

Les stratégies réseau peuvent s'appliquer aux stratégies existantes ainsi qu'aux stratégies futures. Nous vous recommandons de créer des stratégies réseau avant de commencer à créer des collections.

Pour créer une politique réseau OpenSearch sans serveur

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le panneau de navigation de gauche, développez Serverless (Sans serveur) et choisissez Network policies (Stratégies réseau).
3. Choisissez Create network policy (Créer une stratégie réseau).
4. Saisissez un nom et une description pour la stratégie.
5. Fournissez une ou plusieurs règles. Ces règles définissent les autorisations d'accès pour vos collections OpenSearch sans serveur et leurs points de terminaison de OpenSearch tableaux de bord.

   Chaque règle contient les éléments suivants :

Element	Description
Nom de la règle	Nom décrivant le contenu de la règle. Par exemple, « Accès au VPC pour l'équipe marketing ».
Type d'accès	Choisissez un accès public ou privé. Sélectionnez ensuite l'une des options suivantes ou les deux : • Points de terminaison VPC pour l'accès — Spécifiez un ou plusieurs points de

Element	Description
	terminaison VPC gérés sans serveur — points de terminaison VPC gérés. OpenSearch • Service AWS accès privé — Sélectionnez un ou plusieurs accès pris en charge Services AWS.
Type de ressource	Indiquez si vous souhaitez donner accès aux OpenSearch points de terminaison (qui permettent d'appeler l' OpenSearch API), aux OpenSearch tableaux de bord (qui permettent d'accéder aux visualisations et à l'interface utilisateur pour les OpenSearch plug-ins), ou les deux. **Note** Service AWS l'accès privé s'applique uniquement au point de OpenSearch terminaison de la collection, et non au point de terminaison OpenSearch des tableaux de bord. Même si vous sélectionnez OpenSearch Tableaux de bord, Services AWS seul l'accès aux terminaux est autorisé.

Pour chaque type de ressource que vous sélectionnez, vous pouvez choisir des collections existantes auxquelles appliquer les paramètres de politique, and/or créer un ou plusieurs modèles de ressources. Les modèles de ressources se composent d'un préfixe et d'un caractère générique (*) et définissent les collections auxquelles les paramètres de stratégie s'appliqueront.

Par exemple, si vous incluez un modèle appelé `Marketing*`, les paramètres réseau de cette stratégie seront automatiquement appliqués à toutes les collections nouvelles ou existantes dont

le nom commence par « Marketing ». Un seul caractère générique (*) applique la stratégie à toutes les collections actuelles et futures.

En outre, vous pouvez spécifier le nom d'une future collection sans caractère générique, par exemple`Finance`. OpenSearch Serverless appliquera les paramètres de politique à toute collection nouvellement créée portant le même nom exact.

6. Lorsque la configuration de votre stratégie vous satisfait, choisissez Create (Créer).

Création de stratégies réseau (AWS CLI)

Pour créer une politique réseau à l'aide des opérations de l'API OpenSearch Serverless, vous devez spécifier des règles au format JSON. La CreateSecurityPolicydemande accepte à la fois les politiques intégrées et les fichiers .json. Toutes les collections et tous les modèles doivent être sous la forme `collection/<collection name|pattern>`.

> **Note**
>
> Le type de ressource autorise `dashboards` uniquement l'accès aux OpenSearch tableaux de bord, mais pour que les OpenSearch tableaux de bord fonctionnent, vous devez également autoriser l'accès aux collections à partir des mêmes sources. La deuxième stratégie ci-dessous sert d'exemple.

Pour spécifier un accès privé, incluez l'un des éléments suivants ou les deux :

- `SourceVPCEs`— Spécifiez un ou plusieurs points de OpenSearch terminaison VPC gérés sans serveur.
- `SourceServices`— Spécifiez l'identifiant d'un ou de plusieurs appareils pris en charge Services AWS. Les identifiants de service suivants sont actuellement pris en charge :
  - `bedrock.amazonaws.com`— Amazon Bedrock

L'exemple de politique réseau suivant fournit un accès privé, à un point de terminaison VPC et à Amazon Bedrock, aux points de terminaison de collecte uniquement pour les collections commençant par le préfixe. `log*` Les utilisateurs authentifiés ne peuvent pas se connecter aux OpenSearch tableaux de bord ; ils ne peuvent accéder au point de terminaison de collecte que par programmation.

```
[
```

```
   {
      "Description":"Private access for log collections",
      "Rules":[
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/log*"
            ]
         }
      ],
      "AllowFromPublic":false,
      "SourceVPCEs":[
         "vpce-050f79086ee71ac05"
      ],
      "SourceServices":[
         "bedrock.amazonaws.com"
      ],
   }
]
```

La politique suivante fournit un accès public au OpenSearch point de terminaison et aux OpenSearch tableaux de bord pour une seule collection nomméefinance. Si la collection n'existe pas, les paramètres réseau seront appliqués à la collection si et quand elle sera créée.

```
[
   {
      "Description":"Public access for finance collection",
      "Rules":[
         {
            "ResourceType":"dashboard",
            "Resource":[
               "collection/finance"
            ]
         },
         {
            "ResourceType":"collection",
            "Resource":[
               "collection/finance"
            ]
         }
      ],
      "AllowFromPublic":true
   }
```

```
]
```

La requête suivante crée la stratégie réseau ci-dessus :

```
aws opensearchserverless create-security-policy \
    --name sales-inventory \
    --type network \
    --policy "[{\"Description\":\"Public access for finance collection\",\"Rules
\":[{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection\/finance\"]},
{\"ResourceType\":\"collection\",\"Resource\":[\"collection\/finance\"]}],
\"AllowFromPublic\":true}]"
```

Pour fournir la stratégie dans un fichier JSON, utilisez le format `--policy file://my-policy.json`.

Affichage des stratégies réseau

Avant de créer une collection, vous souhaiterez peut-être prévisualiser les stratégies réseau existantes dans votre compte pour voir laquelle possède un modèle de ressource correspondant au nom de votre collection. La ListSecurityPoliciesdemande suivante répertorie toutes les politiques réseau de votre compte :

```
aws opensearchserverless list-security-policies --type network
```

La requête renvoie des informations sur toutes les stratégies réseau configurées. Pour consulter les règles de modèle définies dans une politique spécifique, recherchez les informations de stratégie dans le contenu de l'`securityPolicySummaries`élément de la réponse. Notez la `name` fin `type` de cette politique et utilisez ces propriétés dans une GetSecurityPolicydemande pour recevoir une réponse contenant les détails de politique suivants :

```
{
    "securityPolicyDetail": [
        {
            "type": "network",
            "name": "my-policy",
            "policyVersion": "MTY2MzY5MTY1MDA3Ml8x",
            "policy": "[{\"Description\":\"My network policy rule\",\"Rules\":
[{\"ResourceType\":\"dashboard\",\"Resource\":[\"collection/*\"]}],\"AllowFromPublic
\":true}]",
            "createdDate": 1663691650072,
```

```
            "lastModifiedDate": 1663691650072
        }
    ]
}
```

Pour afficher des informations détaillées sur une politique spécifique, utilisez la GetSecurityPolicycommande.

Mettre à jour des stratégies réseau

Lorsque vous modifiez les points de terminaison d'un VPC ou la désignation d'accès public pour un réseau, toutes les collections associées sont affectées. Pour mettre à jour une politique réseau dans la console OpenSearch Serverless, développez les politiques réseau, sélectionnez la politique à modifier, puis choisissez Modifier. Effectuez les modifications souhaitées, puis choisissez Save (Enregistrer).

Pour mettre à jour une politique réseau à l'aide de l'API OpenSearch Serverless, utilisez la UpdateSecurityPolicycommande. Vous devez inclure une version de stratégie dans la requête. Vous pouvez récupérer la version de stratégie à l'aide des commandes `ListSecurityPolicies` ou `GetSecurityPolicy`. En incluant la version la plus récente de la stratégie, vous vous assurez de ne pas annuler par inadvertance une modification apportée par quelqu'un d'autre.

La requête suivante met à jour une stratégie réseau avec un nouveau document JSON de stratégie :

```
aws opensearchserverless update-security-policy \
    --name sales-inventory \
    --type network \
    --policy-version MTY2MzY5MTY1MDA3Ml8x \
    --policy file://my-new-policy.json
```

Supprimer des stratégies réseau

Avant de supprimer une stratégie réseau, vous devez la détacher de toutes les collections. Pour supprimer une politique dans la console OpenSearch Serverless, sélectionnez-la, puis choisissez Supprimer.

Vous pouvez également utiliser la DeleteSecurityPolicycommande :

```
aws opensearchserverless delete-security-policy --name my-policy --type network
```

Contrôle d'accès aux données pour Amazon OpenSearch Serverless

Grâce au contrôle d'accès aux données dans Amazon OpenSearch Serverless, vous pouvez autoriser les utilisateurs à accéder aux collections et aux index, quel que soit leur mécanisme d'accès ou leur source réseau. Vous pouvez accorder l'accès aux rôles IAM et aux identités SAML.

Vous gérez les autorisations d'accès par le biais de stratégies d'accès aux données, qui s'appliquent aux collections et aux ressources d'index. Les stratégies d'accès aux données vous permettent de gérer les collections à grande échelle en attribuant automatiquement des autorisations d'accès aux collections et aux index qui correspondent à un modèle spécifique. Plusieurs stratégies d'accès aux données peuvent s'appliquer à une seule ressource. Notez que vous devez disposer d'une politique d'accès aux données pour votre collection afin d'accéder à l'URL de vos OpenSearch tableaux de bord.

Rubriques

- Stratégies d'accès aux données ou politiques IAM
- Autorisations IAM requises pour configurer les politiques d'accès aux données
- Syntaxe d’une politique
- Autorisations de stratégies prises en charge
- Exemples de jeux de données sur les tableaux de bord OpenSearch
- Création de stratégies d'accès aux données (console)
- Créer des stratégies d'accès aux données (AWS CLI)
- Afficher les stratégies d'accès aux données
- Mettre à jour les stratégies d'accès aux données
- Supprimer des stratégies d'accès aux données
- Accès aux données entre comptes

Stratégies d'accès aux données ou politiques IAM

Les politiques d'accès aux données sont logiquement distinctes des politiques AWS Identity and Access Management (IAM). Les autorisations IAM contrôlent l'accès aux opérations d'API sans serveur, telles que `CreateCollection` et `ListAccessPolicies`. Les politiques d'accès aux données contrôlent l'accès aux OpenSearch opérations prises en charge par OpenSearch Serverless, telles que `PUT <index>` ou`GET _cat/indices`.

Les autorisations IAM qui contrôlent l'accès aux opérations d'API de stratégie d'accès aux données, telles que `aoss:CreateAccessPolicy` et `aoss:GetAccessPolicy` (décrites dans la section suivante), n'affectent pas l'autorisation spécifiée dans une stratégie d'accès aux données.

Supposons, par exemple, qu'une politique IAM empêche un utilisateur de créer des stratégies d'accès aux données pour `collection-a`, mais lui permette de créer des stratégies d'accès aux données pour toutes les collections (`*`) :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "aoss:CreateAccessPolicy"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "aoss:collection": "collection-a"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "aoss:CreateAccessPolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

Si l'utilisateur crée une stratégie d'accès aux données qui autorise certaines autorisations pour toutes les collections (`collection/*` ou `index/*/*`), la stratégie s'appliquera à toutes les collections, y compris la collection A.

> **Important**
>
> L'octroi d'autorisations dans le cadre d'une politique d'accès aux données n'est pas suffisant pour accéder aux données de votre collection OpenSearch Serverless. Un principal associé doit également avoir accès aux autorisations IAM `aoss:APIAccessAll` et`aoss:DashboardsAccessAll`. Les deux autorisations accordent un accès complet aux ressources de collection, tandis que l'autorisation Dashboards donne également accès aux OpenSearch Dashboards. Si un principal ne dispose pas de ces deux autorisations IAM, il recevra 403 erreurs lorsqu'il tentera d'envoyer des demandes à la collection. Pour de plus amples informations, veuillez consulter the section called “Utilisation des opérations OpenSearch d'API”.

Autorisations IAM requises pour configurer les politiques d'accès aux données

Le contrôle d'accès aux données pour OpenSearch Serverless utilise les autorisations IAM suivantes. Vous pouvez spécifier des conditions IAM pour restreindre les utilisateurs à des noms de stratégie d'accès spécifiques.

- `aoss:CreateAccessPolicy` : créer une stratégie d'accès.
- `aoss:ListAccessPolicies` : répertorier toutes les stratégies d'accès.
- `aoss:GetAccessPolicy` : afficher les informations relatives à une stratégie d'accès spécifique.
- `aoss:UpdateAccessPolicy` : modifier une stratégie d'accès.
- `aoss:DeleteAccessPolicy` : supprimer une stratégie d'accès.

La stratégie d'accès basée sur l'identité suivante permet à un utilisateur de consulter toutes les stratégies d'accès et de mettre à jour les stratégies qui contiennent le modèle de ressource `collection/logs`.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
```

```
                "aoss:ListAccessPolicies",
                "aoss:GetAccessPolicy"
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": [
                "aoss:UpdateAccessPolicy"
            ],
            "Effect": "Allow",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aoss:collection": [
                        "logs"
                    ]
                }
            }
        }
    ]
}
```

> **Note**
>
> En outre, OpenSearch Serverless nécessite les `aoss:DashboardsAccessAll` autorisations `aoss:APIAccessAll` et pour les ressources de collecte. Pour de plus amples informations, veuillez consulter the section called “Utilisation des opérations OpenSearch d'API”.

Syntaxe d’une politique

Une stratégie d'accès aux données inclut un jeu de règles, chacune avec les éléments suivants :

Element	Description
`ResourceType`	Type de ressource (collection ou index) auquel les autorisations s'appliquent. Les autorisations relatives aux alias et aux modèles se situent au niveau de la collection, tandis que les autorisations de création, de modification et de

Element	Description
	recherche de données se situent au niveau de l'index. Pour plus d'informations, veuillez consulter la rubrique Autorisations de stratégie prises en charge.
`Resource`	Liste de and/or modèles de noms de ressources. Les modèles sont des préfixes suivis d'un caractère générique (*), qui permettent aux autorisations associées de s'appliquer à plusieurs ressources. • Les collections prennent le format `collection/` *`<name|pattern>`* . • Les index prennent le format `index/`*`<collection-name|pattern>`* `/`*`<index-name|pattern/>`* .
`Permission`	Liste d'autorisations à accorder pour les ressources spécifiées. Pour obtenir une liste complète des autorisations et des opérations d'API qu'elles autorisent, veuillez consulter la rubrique the section called “Opérations et autorisations d' OpenSearch API prises en charge”.
`Principal`	Liste d'un ou de plusieurs principaux auxquels accorder l'accès. Les principaux peuvent être un rôle IAM ARNs ou des identités SAML. Ces principaux doivent se trouver au sein du Compte AWS actuel. Les politiques d'accès aux données ne prennent pas directement en charge l'accès entre comptes, mais vous pouvez inclure dans votre politique un rôle qu'un utilisateur d'un autre compte Compte AWS peut assumer dans le compte propriétaire de la collection. Pour de plus amples informations, veuillez consulter the section called “Accès aux données entre comptes”.

L'exemple de stratégie suivant accorde des autorisations d'alias et de modèle à la collection nommée `autopartsinventory`, ainsi qu'à toutes les collections commençant par le préfixe `sales*`.
Il accorde également des autorisations de lecture et d'écriture à tous les index de la collection `autopartsinventory`, ainsi qu'à tous les index de la collection `salesorders` commençant par le préfixe `orders*`.

```
[
   {
      "Description": "Rule 1",
      "Rules":[
```

```
        {
           "ResourceType":"collection",
           "Resource":[
              "collection/autopartsinventory",
              "collection/sales*"
           ],
           "Permission":[
              "aoss:CreateCollectionItems",
              "aoss:UpdateCollectionItems",
              "aoss:DescribeCollectionItems"
           ]
        },
        {
           "ResourceType":"index",
           "Resource":[
              "index/autopartsinventory/*",
              "index/salesorders/orders*"
           ],
           "Permission":[
              "aoss:*"
           ]
        }
     ],
     "Principal":[
        "arn:aws:iam::123456789012:user/Dale",
        "arn:aws:iam::123456789012:role/RegulatoryCompliance",
        "saml/123456789012/myprovider/user/Annie",
        "saml/123456789012/anotherprovider/group/Accounting"
     ]
  }
]
```

Vous ne pouvez pas refuser explicitement l'accès dans le cadre d'une stratégie. Par conséquent, toutes les autorisations de stratégies sont cumulatives. Par exemple, si une stratégie accorde à un utilisateur `aoss:ReadDocument` et qu'une autre stratégie accorde `aoss:WriteDocument`, l'utilisateur disposera des deux autorisations. Si une troisième stratégie accorde au même utilisateur `aoss:*`, celui-ci peut effectuer toutes les actions sur l'index associé. Les autorisations plus restrictives ne remplacent pas les autorisations moins restrictives.

Autorisations de stratégies prises en charge

Les autorisations suivantes sont prises en charge dans les stratégies d'accès aux données. Pour les opérations OpenSearch d'API autorisées par chaque autorisation, consultezthe section called “Opérations et autorisations d' OpenSearch API prises en charge”.

Autorisations de collection

- `aoss:CreateCollectionItems`
- `aoss:DeleteCollectionItems`
- `aoss:UpdateCollectionItems`
- `aoss:DescribeCollectionItems`
- `aoss:*`

Autorisations d'index

- `aoss:ReadDocument`
- `aoss:WriteDocument`
- `aoss:CreateIndex`
- `aoss:DeleteIndex`
- `aoss:UpdateIndex`
- `aoss:DescribeIndex`
- `aoss:*`

Exemples de jeux de données sur les tableaux de bord OpenSearch

OpenSearch Les tableaux de bord fournissent des exemples de jeux de données accompagnés de visualisations, de tableaux de bord et d'autres outils pour vous aider à explorer les tableaux de bord avant d'ajouter vos propres données. Pour créer des index à partir de ces exemples de données, vous avez besoin d'une politique d'accès aux données qui fournit des autorisations pour l'ensemble de données avec lequel vous souhaitez travailler. La politique suivante utilise un caractère générique (*) pour fournir des autorisations aux trois exemples de jeux de données.

```
[
  {
    "Rules": [
```

```
      {
        "Resource": [
          "index/<collection-name>/opensearch_dashboards_sample_data_*"
        ],
        "Permission": [
          "aoss:CreateIndex",
          "aoss:DescribeIndex",
          "aoss:ReadDocument"
        ],
        "ResourceType": "index"
      }
    ],
    "Principal": [
      "arn:aws:iam::<account-id>:user/<user>"
    ]
  }
]
```

Création de stratégies d'accès aux données (console)

Vous pouvez créer une stratégie d'accès aux données à l'aide de l'éditeur visuel ou au format JSON. Toutes les nouvelles collections correspondant à l'un des modèles définis dans la stratégie se verront attribuer les autorisations correspondantes lors de la création de la collection.

Pour créer une politique d'accès aux données OpenSearch sans serveur

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, développez Serverless et sous Sécurité, choisissez Politiques d'accès aux données.
3. Choisissez Create access policy (Créer une stratégie d'accès).
4. Saisissez un nom et une description pour la stratégie.
5. Saisissez un nom pour la première règle de votre stratégie. Par exemple, « Accès à la collection de journaux ».
6. Choisissez Add principals (Ajouter des principaux) et sélectionnez un ou plusieurs rôles IAM ou utilisateurs et groupes SAML pour accorder l'accès aux données.

> **Note**
>
> Pour sélectionner des principaux dans les menus déroulants, vous devez disposer des autorisations `iam:ListUsers` et `iam:ListRoles` (pour les principaux IAM) et de l'autorisation `aoss:ListSecurityConfigs` (pour les identités SAML).

7. Choisissez Grant (Accorder) et sélectionnez les autorisations d'alias, de modèle et d'index à accorder aux principaux associés. Pour obtenir la liste complète des autorisations et des accès qu'elles octroient, veuillez consulter la rubrique the section called “Opérations et autorisations d' OpenSearch API prises en charge”.
8. (Facultatif) Configurez des règles supplémentaires pour la stratégie.
9. Choisissez Créer. Il peut s'écouler environ une minute entre le moment où vous créez la stratégie et le moment où les autorisations sont appliquées. Si cela prend plus de 5 minutes, contactez Support.

> **Important**
>
> Si votre stratégie ne comprend que des autorisations d'index (et aucune autorisation de collection), vous pouvez toujours recevoir un message pour les collections correspondantes indiquant `Collection cannot be accessed yet. Configure data access policies so that users can access the data within this collection.` Vous pouvez ignorer cet avertissement. Les principaux autorisés peuvent toujours effectuer les opérations liées à l'index qui leur sont attribuées sur la collection.

Créer des stratégies d'accès aux données (AWS CLI)

Pour créer une politique d'accès aux données à l'aide de l'API OpenSearch Serverless, utilisez la `CreateAccessPolicy` commande. La commande accepte à la fois les stratégies en ligne et les fichiers .json. Les stratégies en ligne doivent être codées sous la forme d'une chaîne d'échappement JSON.

La requête suivante crée une stratégie d'accès aux données :

```
aws opensearchserverless create-access-policy \
    --name marketing \
```

```
    --type data \
    --policy "[{\"Rules\":[{\"ResourceType\":\"collection\",\"Resource\":
[\"collection/autopartsinventory\",\"collection/sales*\"],\"Permission\":
[\"aoss:UpdateCollectionItems\"]},{\"ResourceType\":\"index\",\"Resource\":
[\"index/autopartsinventory/*\",\"index/salesorders/orders*\"],\"Permission
\":[\"aoss:ReadDocument\",\"aoss:DescribeIndex\"]}],\"Principal\":
[\"arn:aws:iam::123456789012:user/Shaheen\"]}]"
```

Pour fournir la stratégie dans un fichier .json, utilisez le format `--policy file://my-policy.json`.

Les principaux inclus dans la politique peuvent désormais utiliser les OpenSearch opérations auxquelles ils ont été autorisés à accéder.

Afficher les stratégies d'accès aux données

Avant de créer une collection, vous souhaiterez peut-être prévisualiser les stratégies d'accès aux données existantes dans votre compte pour voir laquelle possède un modèle de ressource correspondant au nom de votre collection. La ListAccessPoliciesdemande suivante répertorie toutes les politiques d'accès aux données de votre compte :

```
aws opensearchserverless list-access-policies --type data
```

La requête renvoie des informations sur toutes les stratégies d'accès aux données configurées. Pour consulter les règles de modèle définies dans une politique spécifique, recherchez les informations de stratégie dans le contenu de l'`accessPolicySummaries`élément de la réponse. Notez la `name` fin `type` de cette politique et utilisez ces propriétés dans une GetAccessPolicydemande pour recevoir une réponse contenant les détails de politique suivants :

```
{
    "accessPolicyDetails": [
        {
            "type": "data",
            "name": "my-policy",
            "policyVersion": "MTY2NDA1NDE4MDg1OF8x",
            "description": "My policy",
            "policy": "[{\"Rules\":[{\"ResourceType\":\"collection\",
\"Resource\":[\"collection/autopartsinventory\",\"collection/sales*\"],
\"Permission\":[\"aoss:UpdateCollectionItems\"]},{\"ResourceType\":\"index\",
\"Resource\":[\"index/autopartsinventory/*\",\"index/salesorders/orders*\"],
```

```
\"Permission\":[\"aoss:ReadDocument\",\"aoss:DescribeIndex\"]}],\"Principal\":
[\"arn:aws:iam::123456789012:user/Shaheen\"]}]",
            "createdDate": 1664054180858,
            "lastModifiedDate": 1664054180858
        }
    ]
}
```

Vous pouvez inclure des filtres de ressources pour limiter les résultats aux stratégies contenant des collections ou des index spécifiques :

```
aws opensearchserverless list-access-policies --type data --resource
 "index/autopartsinventory/*"
```

Pour afficher les détails d'une politique spécifique, utilisez la GetAccessPolicycommande.

Mettre à jour les stratégies d'accès aux données

Lorsque vous mettez à jour une stratégie d'accès aux données, toutes les collections associées sont affectées. Pour mettre à jour une politique d'accès aux données dans la console OpenSearch Serverless, choisissez Contrôle d'accès aux données, sélectionnez la politique à modifier, puis choisissez Modifier. Effectuez les modifications souhaitées, puis choisissez Save (Enregistrer).

Pour mettre à jour une politique d'accès aux données à l'aide de l'API OpenSearch Serverless, envoyez une `UpdateAccessPolicy` demande. Vous devez inclure une version de la stratégie, que vous pouvez récupérer à l'aide des commandes `ListAccessPolicies` ou `GetAccessPolicy`. En incluant la version la plus récente de la stratégie, vous vous assurez de ne pas annuler par inadvertance une modification apportée par quelqu'un d'autre.

La UpdateAccessPolicydemande suivante met à jour une politique d'accès aux données avec un nouveau document JSON de politique :

```
aws opensearchserverless update-access-policy \
    --name sales-inventory \
    --type data \
    --policy-version MTY2NDA1NDE4MDg1OF8x \
    --policy file://my-new-policy.json
```

Il peut s'écouler quelques minutes entre le moment où vous mettez à jour la stratégie et le moment où les nouvelles autorisations sont appliquées.

Supprimer des stratégies d'accès aux données

Lorsque vous supprimez une stratégie d'accès aux données, toutes les collections associées perdent l'accès défini dans la stratégie. Assurez-vous que vos utilisateurs IAM et SAML disposent de l'accès approprié à la collection avant de supprimer une stratégie. Pour supprimer une politique dans la console OpenSearch Serverless, sélectionnez-la, puis choisissez Supprimer.

Vous pouvez également utiliser la DeleteAccessPolicycommande :

```
aws opensearchserverless delete-access-policy --name my-policy --type data
```

Accès aux données entre comptes

Bien que vous ne puissiez pas créer de politique d'accès aux données avec une identité entre comptes ou des collections entre comptes, vous pouvez toujours configurer un accès entre comptes avec l'option assumer un rôle. Par exemple, s'il *account-a* possède une collection à laquelle il *account-b* faut accéder, l'utilisateur de *account-b* peut jouer un rôle dans*account-a*. Le rôle doit disposer des autorisations IAM `aoss:APIAccessAll` et `aoss:DashboardsAccessAll` être inclus dans la politique d'accès aux données sur*account-a*.

Accédez à Amazon OpenSearch Serverless à l'aide d'un point de terminaison d'interface ()AWS PrivateLink

Vous pouvez l'utiliser AWS PrivateLink pour créer une connexion privée entre votre VPC et Amazon OpenSearch Serverless. Vous pouvez accéder à OpenSearch Serverless comme s'il se trouvait dans votre VPC, sans utiliser de passerelle Internet, de périphérique NAT, de connexion VPN ou de connexion. AWS Direct Connect Les instances de votre VPC n'ont pas besoin d'adresses IP publiques pour accéder OpenSearch à Serverless. Pour plus d'informations sur l'accès au réseau VPC, consultez Modèles de connectivité réseau pour Amazon OpenSearch Serverless.

Vous établissez cette connexion privée en créant un point de terminaison d'interface à technologie AWS PrivateLink. Nous créons une interface réseau du point de terminaison dans chaque sous-réseau que vous spécifiez pour le point de terminaison d'interface. Il s'agit d'interfaces réseau gérées par les demandeurs qui servent de point d'entrée pour le trafic destiné OpenSearch à Serverless.

Pour plus d’informations, consultez Accès aux Services AWS via AWS PrivateLink dans le Guide AWS PrivateLink .

Rubriques

- Résolution DNS des points de terminaison de collecte
- VPCs et politiques d'accès au réseau
- VPCs et politiques relatives aux terminaux
- Considérations
- Autorisations nécessaires
- Création d'un point de terminaison d'interface pour OpenSearch Serverless
- Configuration d'un VPC partagé pour Amazon Serverless OpenSearch

Résolution DNS des points de terminaison de collecte

Lorsque vous créez un point de terminaison VPC, le service crée une nouvelle zone hébergée Amazon Route 53 privée et l'attache au VPC. Cette zone hébergée privée consiste en un enregistrement permettant de résoudre l'enregistrement DNS générique pour les collections OpenSearch sans serveur (`*.aoss.us-east-1.amazonaws.com`) aux adresses d'interface utilisées pour le point de terminaison. Vous n'avez besoin que d'un point de terminaison VPC OpenSearch sans serveur dans un VPC pour accéder à toutes les collections et à tous les tableaux de bord de chaque VPC. Région AWS Chaque VPC doté d'un point de terminaison pour OpenSearch Serverless possède sa propre zone hébergée privée attachée.

OpenSearch Serverless crée également un enregistrement DNS générique Route 53 public pour toutes les collections de la région. Le nom DNS correspond aux adresses IP publiques OpenSearch sans serveur. Les clients VPCs qui ne disposent pas d'un point de terminaison VPC OpenSearch sans serveur ou les clients des réseaux publics peuvent utiliser le résolveur Route 53 public et accéder aux collections et aux tableaux de bord avec ces adresses IP. Le type d'adresse IP (IPv4 IPv6, ou Dualstack) du point de terminaison VPC est déterminé en fonction des sous-réseaux fournis lorsque vous créez un point de terminaison d'interface pour Serverless. OpenSearch

> **Note**
>
> OpenSearch Serverless crée une zone hébergée privée Amazon Route 53 supplémentaire `` `<region>.opensearch.amazonaws.com` (`) pour la résolution OpenSearch d'un domaine de service. Vous pouvez mettre à jour votre point de terminaison IPv4 VPC existant vers Dualstack en utilisant la commande du update-vpc-endpoint. AWS CLI

L'adresse du résolveur DNS pour un VPC donné est la deuxième adresse IP du CIDR du VPC. Tout client du VPC doit utiliser ce résolveur pour obtenir l'adresse du point de terminaison du VPC pour toute collection. Le résolveur utilise une zone hébergée privée créée par OpenSearch Serverless. Il suffit d'utiliser ce résolveur pour toutes les collections, quel que soit le compte. Il est également possible d'utiliser le résolveur VPC pour certains points de terminaison de collection et le résolveur public pour d'autres, bien que cela ne soit généralement pas nécessaire.

VPCs et politiques d'accès au réseau

Pour accorder des autorisations réseau OpenSearch APIs et des tableaux de bord pour vos collections, vous pouvez utiliser des politiques d'accès réseau OpenSearch sans serveur. Vous pouvez contrôler cet accès réseau à partir de vos points de terminaison VPC ou de l'Internet public. Étant donné que votre politique réseau ne contrôle que les autorisations de trafic, vous devez également définir une politique d'accès aux données qui spécifie l'autorisation d'opérer sur les données d'une collection et de ses index. Imaginez un point de terminaison VPC OpenSearch sans serveur comme un point d'accès au service, une politique d'accès réseau comme le point d'accès au niveau du réseau aux collections et aux tableaux de bord, et une politique d'accès aux données comme le point d'accès permettant un contrôle d'accès précis pour toute opération sur les données de la collection.

Comme vous pouvez spécifier plusieurs points de terminaison VPC IDs dans une politique réseau, nous vous recommandons de créer un point de terminaison VPC pour chaque VPC devant accéder à une collection. Ils VPCs peuvent appartenir à des AWS comptes différents de ceux du compte propriétaire de la collection OpenSearch Serverless et de la politique réseau. Nous vous déconseillons de créer une solution de VPC-to-VPC peering ou autre solution de proxy entre deux comptes afin que le VPC d'un compte puisse utiliser le point de terminaison VPC d'un autre compte. Cela est moins sûr et moins rentable que le fait que chaque VPC possède son propre point de terminaison. Le premier VPC ne sera pas facilement visible pour l'administrateur de l'autre VPC, qui a configuré l'accès au point de terminaison de ce VPC dans la politique réseau.

VPCs et politiques relatives aux terminaux

Amazon OpenSearch Serverless prend en charge les politiques relatives aux terminaux pour VPCs. Une politique de point de terminaison est une politique basée sur les ressources IAM que vous attachez à un point de terminaison VPC pour contrôler quels AWS principaux peuvent utiliser le point de terminaison pour accéder à votre service. AWS Pour plus d'informations, consultez Contrôler l'accès aux points de terminaison VPC à l'aide de politiques de point de terminaison.

Pour utiliser une politique de point de terminaison, vous devez d'abord créer un point de terminaison d'interface. Vous pouvez créer un point de terminaison d'interface à l'aide de la console OpenSearch Serverless ou de l'API OpenSearch Serverless. Après avoir créé le point de terminaison de votre interface, vous devez ajouter la politique du point de terminaison au point de terminaison. Pour plus d'informations, consultez Accéder à Amazon OpenSearch Serverless à l'aide d'un point de terminaison d'interface (AWS PrivateLink).

> **Note**
>
> Vous ne pouvez pas définir une politique de point de terminaison directement dans la console OpenSearch de service.

Une politique de point de terminaison ne remplace ni ne remplace les autres politiques basées sur l'identité, les politiques basées sur les ressources, les politiques réseau ou les politiques d'accès aux données que vous avez éventuellement configurées. Pour plus d'informations sur la mise à jour des politiques de point de terminaison, consultez Contrôler l'accès aux points de terminaison VPC à l'aide de politiques de point de terminaison.

Par défaut, une politique de point de terminaison accorde un accès complet à votre point de terminaison VPC.

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

Bien que la politique de point de terminaison VPC par défaut accorde un accès complet au point de terminaison, vous pouvez configurer une politique de point de terminaison VPC pour autoriser l'accès à des rôles et à des utilisateurs spécifiques. Pour ce faire, consultez l'exemple suivant :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "123456789012",
                    "987654321098"
                ]
            },
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

Vous pouvez spécifier une collection OpenSearch sans serveur à inclure en tant qu'élément conditionnel dans votre politique de point de terminaison VPC. Pour ce faire, consultez l'exemple suivant :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aoss:collection": [
                        "coll-abc"
                    ]
                }
            }
```

```
        }
    ]
}
```

Support pour `aoss:CollectionId` est pris en charge.

```
Condition": {
         "StringEquals": {
              "aoss:CollectionId": "collection-id"
         }
}
```

Vous pouvez utiliser les identités SAML dans votre politique de point de terminaison VPC pour déterminer l'accès aux points de terminaison VPC. Vous devez utiliser un caractère générique (*) dans la section principale de votre politique de point de terminaison VPC. Pour ce faire, consultez l'exemple suivant :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aws:SamlGroups": [
                        "saml/123456789012/idp123/group/football",
                        "saml/123456789012/idp123/group/soccer",
                        "saml/123456789012/idp123/group/cricket"
                    ]
                }
            }
        }
    ]
}
```

En outre, vous pouvez configurer votre politique de point de terminaison pour inclure une politique principale SAML spécifique. Pour ce faire, consultez les informations suivantes :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aoss:SamlPrincipal": [
                        "saml/123456789012/idp123/user/user1234"]
                    }
                }
            }
        ]
    }
```

Pour plus d'informations sur l'utilisation de l'authentification SAML avec Amazon OpenSearch Serverless, consultez Authentification SAML pour Amazon Serverless. OpenSearch

Vous pouvez également inclure les utilisateurs IAM et SAML dans la même politique de point de terminaison VPC. Pour ce faire, consultez l'exemple suivant :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "*",
            "Resource": "*",
```

```
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aoss:SamlGroups": [
                        "saml/123456789012/idp123/group/football",
                        "saml/123456789012/idp123/group/soccer",
                        "saml/123456789012/idp123/group/cricket"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "123456789012"
                ]
            },
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

Vous pouvez également accéder à une collection Amazon OpenSearch Serverless depuis Amazon EC2 via des points de terminaison VPC d'interface. Pour plus d'informations, consultez Accéder à une collection OpenSearch sans serveur depuis Amazon EC2 (via des points de terminaison VPC d'interface).

Considérations

Avant de configurer un point de terminaison d'interface pour OpenSearch Serverless, tenez compte des points suivants :

- OpenSearch Serverless permet d'appeler toutes les opérations d'OpenSearch API prises en charge (et non les opérations d'API de configuration) via le point de terminaison de l'interface.
- Après avoir créé un point de terminaison d'interface pour OpenSearch Serverless, vous devez toujours l'inclure dans les politiques d'accès au réseau afin qu'il puisse accéder aux collections sans serveur.
- Par défaut, l'accès complet à OpenSearch Serverless est autorisé via le point de terminaison de l'interface. Vous pouvez associer un groupe de sécurité aux interfaces réseau du point de

terminaison pour contrôler le trafic vers OpenSearch Serverless via le point de terminaison de l'interface.

- Un seul Compte AWS peut avoir un maximum de 50 points de terminaison VPC OpenSearch sans serveur.
- Si vous activez l'accès Internet public à l'API ou aux tableaux de bord de votre collection dans le cadre d'une politique réseau, votre collection est accessible par n'importe quel VPC et par Internet public.
- Si vous êtes sur site et en dehors du VPC, vous ne pouvez pas utiliser directement un résolveur DNS pour la résolution des points de terminaison du VPC OpenSearch sans serveur. Si vous avez besoin d'un accès VPN, le VPC a besoin d'un résolveur de proxy DNS que les clients externes peuvent utiliser. Route 53 fournit une option de point de terminaison entrant que vous pouvez utiliser pour résoudre les requêtes DNS adressées à votre VPC à partir de votre réseau local ou d'un autre VPC.
- La zone hébergée privée que OpenSearch Serverless crée et attache au VPC est gérée par le service, mais elle apparaît dans Amazon Route 53 vos ressources et est facturée à votre compte.
- Pour d'autres considérations, veuillez consulter les Considérations dans le Guide AWS PrivateLink .

Autorisations nécessaires

L'accès VPC pour OpenSearch Serverless utilise les autorisations AWS Identity and Access Management (IAM) suivantes. Vous pouvez spécifier des conditions IAM pour restreindre les utilisateurs à des collections spécifiques.

- `aoss:CreateVpcEndpoint` : créer un point de terminaison d'un VPC.
- `aoss:ListVpcEndpoints` : répertorier tous les points de terminaison d'un VPC.
- `aoss:BatchGetVpcEndpoint` : consulter les détails d'un sous-ensemble de points de terminaison d'un VPC.
- `aoss:UpdateVpcEndpoint` : modifier un point de terminaison d'un VPC.
- `aoss:DeleteVpcEndpoint` : supprimer un point de terminaison d'un VPC.

En outre, vous avez besoin des autorisations Amazon EC2 et Route 53 suivantes pour créer un point de terminaison VPC.

- `ec2:CreateTags`

- `ec2:CreateVpcEndpoint`
- `ec2:DeleteVpcEndPoints`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`
- `ec2:DescribeVpcEndpoints`
- `ec2:DescribeVpcs`
- `ec2:ModifyVpcEndPoint`
- `route53:AssociateVPCWithHostedZone`
- `route53:ChangeResourceRecordSets`
- `route53:CreateHostedZone`
- `route53:DeleteHostedZone`
- `route53:GetChange`
- `route53:GetHostedZone`
- `route53:ListHostedZonesByName`
- `route53:ListHostedZonesByVPC`
- `route53:ListResourceRecordSets`

Création d'un point de terminaison d'interface pour OpenSearch Serverless

Vous pouvez créer un point de terminaison d'interface pour OpenSearch Serverless à l'aide de la console ou de l'API OpenSearch Serverless.

Pour créer un point de terminaison d'interface pour une OpenSearch collection sans serveur

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le panneau de navigation de gauche, développez Serverless (Sans serveur) et choisissez VPC endpoints (Points de terminaison d'un VPC).
3. Choisissez Create VPC endpoint (Créer un point de terminaison d'un VPC).
4. Saisissez un nom pour le point de terminaison.
5. Pour le VPC, sélectionnez le VPC à partir duquel vous allez accéder sans serveur. OpenSearch
6. Pour les sous-réseaux, sélectionnez un sous-réseau à partir duquel vous accéderez OpenSearch sans serveur.

- L'adresse IP et le type DNS du point de terminaison sont basés sur le type de sous-réseau
  - Dualstack : si tous les sous-réseaux ont à la fois IPv4 des plages d'adresses IPv6
  - IPv6: si tous les sous-réseaux IPv6 ne sont que des sous-réseaux
  - IPv4: si tous les sous-réseaux ont des plages d' IPv4 adresses

7. Pour Security groups (Groupes de sécurité), sélectionnez les groupes de sécurité à associer aux interfaces réseau du point de terminaison. Il s'agit d'une étape essentielle dans le cadre de laquelle vous devez limiter les ports, les protocoles et les sources de trafic entrant que vous autorisez dans votre point de terminaison. Assurez-vous que les règles du groupe de sécurité autorisent les ressources qui utiliseront le point de terminaison VPC pour communiquer avec OpenSearch Serverless à communiquer avec l'interface réseau du point de terminaison.
8. Choisissez Créer un point de terminaison.

Pour créer un point de terminaison VPC à l'aide de l'API OpenSearch Serverless, utilisez la commande. `CreateVpcEndpoint`

> **Note**
>
> Après avoir créé un point de terminaison, prenez note de son ID (par exemple, `vpce-abc123def4EXAMPLE`). Afin de fournir au point de terminaison un accès à vos collections, vous devez inclure cet ID dans une ou plusieurs stratégies d'accès réseau.

Après avoir créé un point de terminaison d'interface, vous devez lui donner accès aux collections par le biais de stratégies d'accès réseau. Pour de plus amples informations, veuillez consulter the section called “Accès réseau”.

Configuration d'un VPC partagé pour Amazon Serverless OpenSearch

Vous pouvez utiliser Amazon Virtual Private Cloud (VPC) pour partager des sous-réseaux VPC avec d'autres Comptes AWS membres de votre organisation, ainsi que pour partager une infrastructure réseau telle qu'un VPN entre plusieurs ressources. Comptes AWS

Actuellement, Amazon OpenSearch Serverless ne prend pas en charge la création d'une AWS PrivateLink connexion à un VPC partagé, sauf si vous êtes propriétaire de ce VPC. AWS PrivateLink ne prend pas non plus en charge le partage de connexions entre Comptes AWS.

Cependant, sur la base de l'architecture flexible et modulaire de OpenSearch Serverless, vous pouvez toujours configurer un VPC partagé. Cela est dû au fait que l'infrastructure réseau OpenSearch sans serveur est distincte de celle de l'infrastructure de collecte individuelle (OpenSearch service). Vous pouvez donc créer un AWS PrivateLink VPCe point de terminaison pour un compte sur lequel se trouve un VPC, puis utiliser un VPCe ID dans la politique réseau des autres comptes afin de limiter le trafic provenant uniquement de ce VPC partagé.

Les procédures suivantes font référence à un compte propriétaire et à un compte client.

Un compte propriétaire agit comme un compte réseau commun dans lequel vous configurez un VPC et le partagez avec d'autres comptes. Les comptes clients sont les comptes qui créent et gèrent leurs collections OpenSearch sans serveur dans le VPC partagé avec eux par le compte propriétaire.

Prérequis

Assurez-vous que les conditions suivantes sont remplies avant de configurer le VPC partagé :

- Le compte du propriétaire prévu doit déjà avoir configuré un VPC, des sous-réseaux, une table de routage et les autres ressources requises dans Amazon Virtual Private Cloud. Pour de plus amples informations, consultez le Guide de l'utilisateur Amazon VPC.
- Le compte du propriétaire prévu et les comptes du consommateur doivent appartenir à la même organisation dans AWS Organizations. Pour plus d’informations, consultez le Guide de l’utilisateur AWS Organizations.

Pour configurer un VPC partagé dans un compte account/common réseau propriétaire.

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Suivez les étapes de the section called “Création d’un point de terminaison d’interface”. Au fur et à mesure, effectuez les sélections suivantes :
   - Sélectionnez un VPC et des sous-réseaux partagés avec les comptes clients de votre organisation.
3. Après avoir créé le point de terminaison, notez l' VPCe identifiant généré et fournissez-le aux administrateurs chargés d'effectuer la tâche de configuration dans les comptes clients.

   VPCe IDs sont au format`vpce-abc123def4EXAMPLE`.

Pour configurer un VPC partagé dans un compte client

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Utilisez les informations the section called “Gestion des collections” pour créer une collection, si vous n'en avez pas déjà une.
3. Utilisez les informations contenues the section called “Créer des stratégies réseau (console)” pour créer une politique réseau. Au fur et à mesure, effectuez les sélections suivantes.

   > **Note**
   >
   > Vous pouvez également mettre à jour une politique réseau existante à cette fin.

   a. Pour le type d'accès, sélectionnez VPC (recommandé).
   b. Pour les points de terminaison VPC auxquels vous pouvez accéder, choisissez l' VPCe ID qui vous a été fourni par le compte propriétaire, au format. `vpce-abc123def4EXAMPLE`
   c. Dans la zone Type de ressource, procédez comme suit :

      - Cochez la case Activer l'accès au OpenSearch point de terminaison, puis sélectionnez le nom ou le modèle de collection à utiliser pour activer l'accès depuis ce VPC partagé.
      - Sélectionnez la case Activer l'accès au OpenSearch tableau de bord, puis sélectionnez le nom ou le modèle de collection à utiliser pour autoriser l'accès depuis ce VPC partagé.
4. Pour une nouvelle politique, choisissez Create. Pour une politique existante, choisissez Mettre à jour.

Authentification SAML pour Amazon Serverless OpenSearch

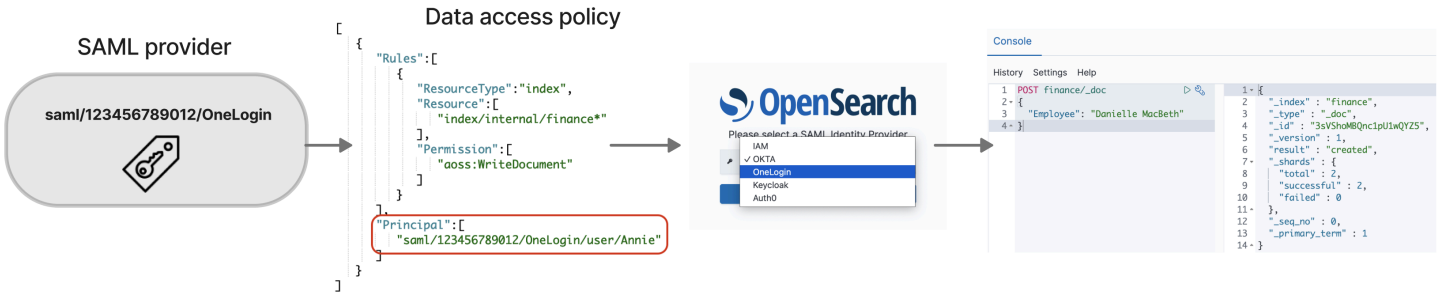
Avec l'authentification SAML pour Amazon OpenSearch Serverless, vous pouvez utiliser votre fournisseur d'identité existant pour proposer l'authentification unique (SSO) pour les points de terminaison des OpenSearch tableaux de bord des collections sans serveur.

L'authentification SAML vous permet d'utiliser des fournisseurs d'identité tiers pour vous connecter aux OpenSearch tableaux de bord afin d'indexer et de rechercher des données. OpenSearch Serverless prend en charge les fournisseurs qui utilisent la norme SAML 2.0, tels que IAM Identity Center, Okta, Keycloak, Active Directory Federation Services (AD FS) et Auth0. Vous pouvez configurer IAM Identity Center pour synchroniser les utilisateurs et les groupes provenant d'autres

sources d'identité telles qu'Okta et Microsoft Entra ID. OneLogin Pour obtenir la liste des sources d'identité prises en charge par IAM Identity Center et les étapes à suivre pour les configurer, consultez les didacticiels de mise en route du guide de l'utilisateur d'IAM Identity Center.

> **Note**
>
> L'authentification SAML permet uniquement d'accéder aux OpenSearch tableaux de bord via un navigateur Web. Les utilisateurs authentifiés peuvent uniquement envoyer des demandes aux opérations de l' OpenSearch API via les outils de développement dans les OpenSearch tableaux de bord. Vos informations d'identification SAML ne vous permettent pas d'envoyer des requêtes HTTP directes aux opérations de l' OpenSearch API.

Pour configurer l'authentification SAML, vous devez d'abord configurer un fournisseur d'identité (IdP) SAML. Vous incluez ensuite un ou plusieurs utilisateurs de cet IdP dans une stratégie d'accès aux données. Cette politique lui accorde certaines autorisations sur les and/or index de collections. Un utilisateur peut ensuite se connecter aux OpenSearch tableaux de bord et effectuer les actions autorisées dans la politique d'accès aux données.



Rubriques

- Considérations
- Autorisations nécessaires
- Créer des fournisseurs SAML (console)
- Accès aux OpenSearch tableaux de bord
- Octroyer aux identités SAML l'accès aux données de collection
- Créer des fournisseurs SAML (AWS CLI)
- Consulter des fournisseurs SAML
- Mettre à jour des fournisseurs SAML

- Supprimer des fournisseurs SAML

Considérations

Tenez compte des éléments suivants lors de la configuration de l'authentification SAML :

- Les requêtes signées et chiffrées ne sont pas prises en charge.
- Les assertions chiffrées ne sont pas prises en charge.
- L'authentification et la déconnexion initiées par l'IdP ne sont pas prises en charge.
- Les politiques de contrôle des services (SCP) ne seront pas applicables ni évaluées dans le cas d'identités non IAM (comme le SAML dans OpenSearch Amazon Serverless et SAML et l'autorisation utilisateur interne de base pour Amazon Service). OpenSearch

Autorisations nécessaires

L'authentification SAML pour OpenSearch Serverless utilise les autorisations AWS Identity and Access Management (IAM) suivantes :

- `aoss:CreateSecurityConfig` : créer un fournisseur SAML.
- `aoss:ListSecurityConfig` : répertorier tous les fournisseurs SAML du compte actuel.
- `aoss:GetSecurityConfig` : afficher les informations du fournisseur SAML.
- `aoss:UpdateSecurityConfig` : modifier la configuration d'un fournisseur SAML donné, y compris les métadonnées XML.
- `aoss:DeleteSecurityConfig` : supprimer un fournisseur SAML.

La stratégie d'accès basée sur l'identité suivante permet à un utilisateur de gérer toutes les configurations IdP :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "aoss:CreateSecurityConfig",
```

```
                "aoss:DeleteSecurityConfig",
                "aoss:GetSecurityConfig",
                "aoss:UpdateSecurityConfig",
                "aoss:ListSecurityConfigs"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Veuillez noter que l'élément `Resource` doit être un caractère générique.

Créer des fournisseurs SAML (console)

Ces étapes expliquent comment créer des fournisseurs SAML. Cela permet l'authentification SAML avec l'authentification initiée par le fournisseur de services (SP) pour les OpenSearch tableaux de bord. L'authentification initiée par l'IdP n'est pas prise en charge.

Pour activer l'authentification SAML pour les tableaux de bord OpenSearch

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le panneau de navigation de gauche, développez Serverless (Sans serveur) et sélectionnez SAML authentication (Authentification SAML).
3. Choisissez Add SAML provider (Ajouter un fournisseur SAML).
4. Saisissez un nom et une description pour le fournisseur.

   > **Note**
   >
   > Le nom que vous spécifiez est accessible au public et apparaît dans un menu déroulant lorsque les utilisateurs se connectent à OpenSearch Dashboards. Assurez-vous que le nom est facilement reconnaissable et qu'il ne révèle pas d'informations sensibles sur votre fournisseur d'identité.

5. Sous Configure your IdP (Configurer votre IdP), copiez l'URL Assertion Consumer Service (ACS).

6. Utilisez l'URL ACS que vous venez de copier pour configurer votre fournisseur d'identité. La terminologie et les étapes varient selon le fournisseur. Consultez la documentation de votre fournisseur.

   Dans Okta, par exemple, vous créez une « application web SAML 2.0 » et vous spécifiez l'URL ACS comme URL d'authentification unique, URL du destinataire et URL de destination. Pour Auth0, vous le spécifiez dans Allowed URLs Callback.

7. Indiquez la restriction d'audience si votre IdP dispose d'un champ à cet effet. La restriction d'audience est une valeur de l'assertion SAML qui indique à qui l'assertion est destinée. Pour OpenSearch Serverless, spécifiez`aws:opensearch:<aws account id>`. Par exemple, `aws:opensearch:`*`123456789012`*.

   Le nom du champ de restriction d'audience varie selon le fournisseur. Pour Okta, il s'agit d'URI d'audience (ID d'entité du fournisseur de services). Pour IAM Identity Center, il s'agit de l'audience SAML des applications.

8. Si vous utilisez IAM Identity Center, vous devez également spécifier le mappage d'attributs suivant : `Subject=${user:name}`, au format `unspecified`.

9. Une fois votre fournisseur d'identité configuré, il génère un fichier de métadonnées de fournisseur d'identité. Ce fichier XML contient des informations sur le fournisseur, telles qu'un certificat TLS, des points de terminaison d'authentification unique et l'ID d'entité du fournisseur d'identité.

   Copiez le texte dans le fichier de métadonnées IdP et collez-le sous le champ Provide metadata from your IdP (Fournir les métadonnées de votre IdP). Vous pouvez également choisir Importer depuis un fichier XML, puis charger le fichier. Le fichier de métadonnées doit se présenter comme suit :

```
<?xml version="1.0" encoding="UTF-8"?>
<md:EntityDescriptor entityID="entity-id"
 xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata">
  <md:IDPSSODescriptor WantAuthnRequestsSigned="false"
 protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
    <md:KeyDescriptor use="signing">
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:X509Data>
          <ds:X509Certificate>tls-certificate</ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>s
    </md:KeyDescriptor>
```

```
    <md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified</
md:NameIDFormat>
    <md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress</
md:NameIDFormat>
    <md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
POST" Location="idp-sso-url"/>
    <md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
Redirect" Location="idp-sso-url"/>
  </md:IDPSSODescriptor>
</md:EntityDescriptor>
```

10. Laissez le champ d'attribut Custom user ID vide pour utiliser l'`NameID`élément de l'assertion SAML comme nom d'utilisateur. Si votre assertion n'utilise pas cet élément standard et inclut plutôt le nom d'utilisateur comme attribut personnalisé, spécifiez cet attribut ici. Les attributs sont sensibles à la casse. Seul un attribut d'utilisateur unique est pris en charge.

    L'exemple suivant montre un attribut de remplacement pour `NameID` dans l'assertion SAML :

    ```
    <saml2:Attribute Name="UserId" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
    format:basic">
      <saml2:AttributeValue xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="xs:string">annie</saml2:AttributeValue>
    </saml2:Attribute>
    ```

11. (Facultatif) Spécifiez un attribut personnalisé dans le champ Group attribute (Attribut de groupe), tel que `role` ou `group`. Seul un attribut de groupe unique est pris en charge. Il n'existe aucun attribut de groupe par défaut. Si vous n'en spécifiez pas, vos stratégies d'accès aux données ne peuvent contenir que des principaux d'utilisateur.

    L'exemple suivant montre un attribut de groupe dans l'assertion SAML :

    ```
    <saml2:Attribute Name="department"
     NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
        <saml2:AttributeValue xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="xs:string">finance</saml2:AttributeValue>
    </saml2:Attribute>
    ```

12. Par défaut, OpenSearch Dashboards déconnecte les utilisateurs au bout de 24 heures. Vous pouvez configurer cette valeur sur un nombre compris entre 1 et 12 heures (15 et 720 minutes)

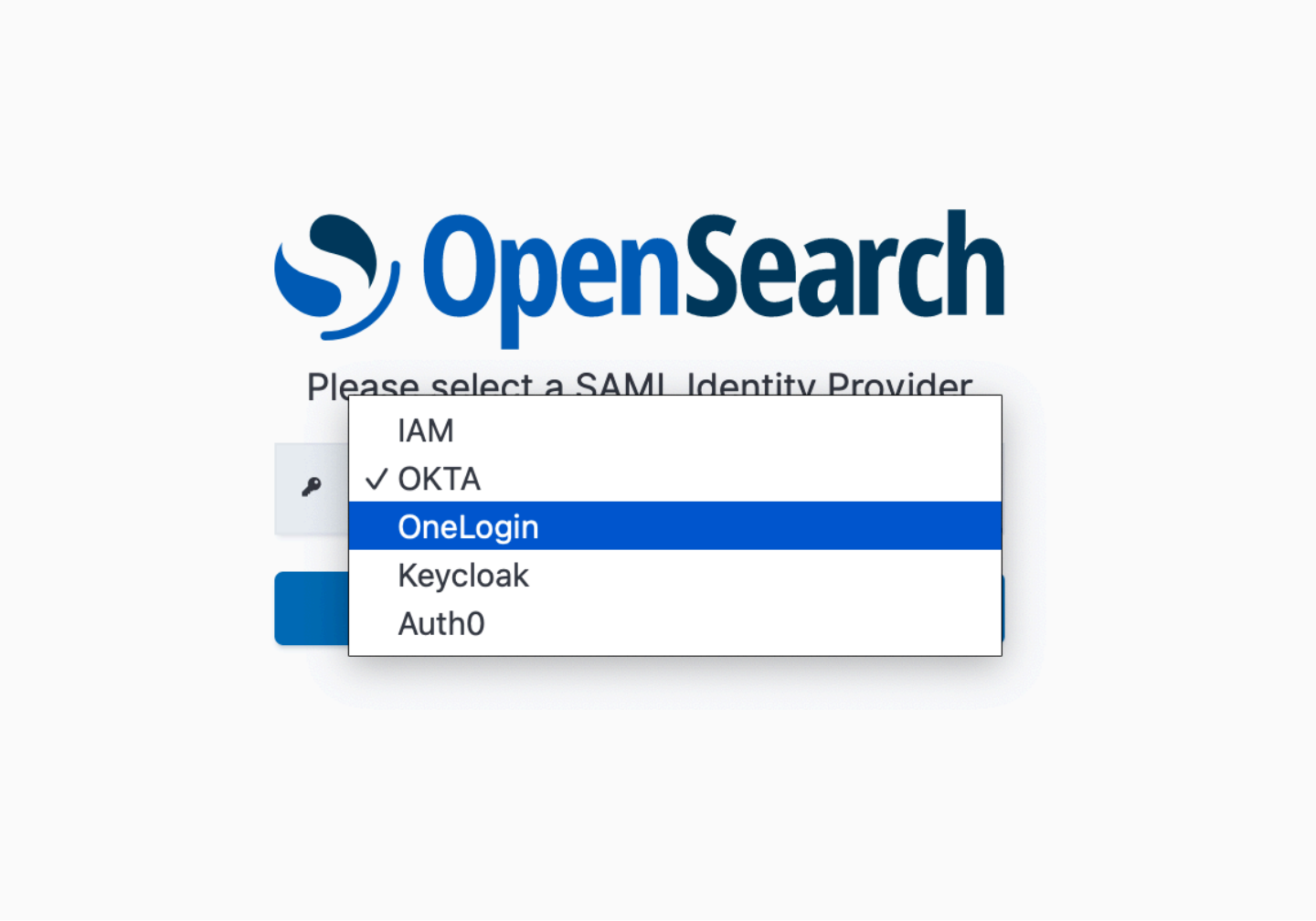
en spécifiant le délai d'expiration OpenSearch des tableaux de bord. Si vous essayez de régler le délai d'attente égal ou inférieur à 15 minutes, votre session sera réinitialisée à une heure.

13. Choisissez Create SAML provider (Créer un fournisseur SAML).

Accès aux OpenSearch tableaux de bord

Après avoir configuré un fournisseur SAML, tous les utilisateurs et groupes associés à ce fournisseur peuvent accéder au point de terminaison OpenSearch Dashboards. Le format de l'URL des tableaux de bord correspond à celui *collection-endpoint*`/_dashboards/` de toutes les collections.

Si le protocole SAML est activé, vous êtes AWS Management Console redirigé vers la page de sélection de l'IdP, où vous pouvez vous connecter à l'aide de vos informations d'identification SAML. Tout d'abord, utilisez la liste déroulante pour sélectionner un fournisseur d'identité :



Connectez-vous ensuite à l'aide de vos informations d'identification d'IdP.

Si le protocole SAML n'est pas activé, vous pouvez cliquer sur le lien dans le pour AWS Management Console vous connecter en tant qu'utilisateur ou en tant que rôle IAM, sans aucune option pour le protocole SAML.

Octroyer aux identités SAML l'accès aux données de collection

Après avoir créé un fournisseur SAML, vous devez toujours octroyer aux utilisateurs et aux groupes sous-jacents l'accès aux données de vos collections. Vous octroyez l'accès par le biais de stratégies d'accès aux données. Tant que vous n'aurez pas octroyé l'accès aux utilisateurs, ils ne pourront pas lire, écrire ou supprimer les données de vos collections.

Pour accorder l'accès, créez une politique d'accès aux données et spécifiez votre and/or groupe d'utilisateurs SAML IDs dans la `Principal` déclaration suivante :

```
[
   {
      "Rules":[
       ...
      ],
      "Principal":[
         "saml/987654321098/myprovider/user/Shaheen",
         "saml/987654321098/myprovider/group/finance"
      ]
   }
]
```

Vous pouvez octroyer l'accès aux collections, aux index ou aux deux. Si vous souhaitez que différents utilisateurs aient des autorisations différentes, créez plusieurs règles. Pour obtenir la liste des autorisations disponibles, veuillez consulter la rubrique Autorisations de stratégie prises en charge. Pour plus d'informations sur le formatage d'une stratégie d'accès, veuillez consulter la rubrique Policy syntax (Syntaxe de stratégie).

Créer des fournisseurs SAML (AWS CLI)

Pour créer un fournisseur SAML à l'aide de l'API OpenSearch Serverless, envoyez une CreateSecurityConfigdemande :

```
aws opensearchserverless create-security-config \
    --name myprovider \
    --type saml \
    --saml-options file://saml-auth0.json
```

Spécifiez `saml-options`, y compris le fichier XML des métadonnées, sous la forme d'un mappage clé-valeur dans un fichier .json. Le fichier XML des métadonnées doit être codé sous la forme d'une chaîne d'échappement JSON.

```
{
   "sessionTimeout": 70,
   "groupAttribute": "department",
   "userAttribute": "userid",
   "metadata": "<EntityDescriptor xmlns=\"urn:oasis:names:tc:SAML:2.0:metadata
\" ... ... ... IDPSSODescriptor>\r\n<\/EntityDescriptor>"
}
```

Consulter des fournisseurs SAML

La ListSecurityConfigsdemande suivante répertorie tous les fournisseurs SAML de votre compte :

```
aws opensearchserverless list-security-configs --type saml
```

La requête renvoie des informations sur tous les fournisseurs SAML existants, y compris les métadonnées IdP complètes générées par votre fournisseur d'identité :

```
{
   "securityConfigDetails": [
      {
         "configVersion": "MTY2NDA1MjY4NDQ5M18x",
         "createdDate": 1664054180858,
         "description": "Example SAML provider",
         "id": "saml/123456789012/myprovider",
         "lastModifiedDate": 1664054180858,
         "samlOptions": {
            "groupAttribute": "department",
            "metadata": "<EntityDescriptor xmlns=\"urn:oasis:names:tc:SAML:2.0:metadata
\" ... ... ... IDPSSODescriptor>\r\n<\/EntityDescriptor>",
            "sessionTimeout": 120,
            "userAttribute": "userid"
         }
      }
   ]
}
```

Pour consulter les détails d'un fournisseur spécifique, y compris la `configVersion` des mises à jour futures, envoyez une requête `GetSecurityConfig`.

Mettre à jour des fournisseurs SAML

Pour mettre à jour un fournisseur SAML à l'aide de la console OpenSearch Serverless, choisissez l'authentification SAML, sélectionnez votre fournisseur d'identité, puis choisissez Modifier. Vous pouvez modifier tous les champs, y compris les métadonnées et les attributs personnalisés.

Pour mettre à jour un fournisseur via l'API OpenSearch Serverless, envoyez une UpdateSecurityConfigdemande et incluez l'identifiant de la politique à mettre à jour. Vous devez également inclure une version de configuration, que vous pouvez récupérer à l'aide des commandes `ListSecurityConfigs` ou `GetSecurityConfig`. En incluant la version la plus récente, vous vous assurez de ne pas annuler par inadvertance une modification apportée par quelqu'un d'autre.

La requête suivante met à jour les options SAML d'un fournisseur :

```
aws opensearchserverless update-security-config \
    --id saml/123456789012/myprovider \
    --type saml \
    --saml-options file://saml-auth0.json \
    --config-version MTY2NDA1MjY4NDQ5M18x
```

Spécifiez vos options de configuration SAML sous la forme d'un mappage clé-valeur dans un fichier .json.

> **Important**
>
> Les mises à jour des options SAML ne sont pas progressives. Si vous ne spécifiez aucune valeur pour un paramètre de l'objet `SAMLOptions` lorsque vous effectuez une mise à jour, les valeurs existantes seront remplacées par des valeurs vides. Par exemple, si la configuration actuelle contient une valeur pour `userAttribute`, puis que vous effectuez une mise à jour sans inclure cette valeur, la valeur est supprimée de la configuration. Assurez-vous de connaître les valeurs existantes avant de procéder à une mise à jour en appelant l'opération `GetSecurityConfig`.

Supprimer des fournisseurs SAML

Lorsque vous supprimez un fournisseur SAML, les références aux utilisateurs et aux groupes associés dans vos stratégies d'accès aux données ne sont plus fonctionnelles. Pour éviter toute confusion, nous vous suggérons de supprimer toutes les références au point de terminaison dans vos stratégies d'accès avant de supprimer le point de terminaison.

Pour supprimer un fournisseur SAML à l'aide de la console OpenSearch sans serveur, choisissez Authentification, sélectionnez le fournisseur, puis choisissez Supprimer.

Pour supprimer un fournisseur via l'API OpenSearch Serverless, envoyez une DeleteSecurityConfigdemande :

```
aws opensearchserverless delete-security-config --id saml/123456789012/myprovider
```

Validation de la conformité pour Amazon OpenSearch Serverless

Des auditeurs tiers évaluent la sécurité et la conformité d'Amazon OpenSearch Serverless dans le cadre de plusieurs programmes de AWS conformité. Il s'agit notamment des programmes SOC, PCI et HIPAA.

Pour savoir si un Service AWS fait partie ou non du champ d'application de programmes de conformité spécifiques, veuillez consulter Services AWS dans le champ d'application par programme de de conformité et sélectionnez le programme de conformité qui vous intéresse. Pour des informations générales, voir Programmes de AWS conformité Programmes AWS de .

Vous pouvez télécharger les rapports de l'audit externe avec AWS Artifact. Pour plus d'informations, consultez Téléchargement de rapports dans AWS Artifact .

Votre responsabilité de conformité lors de l'utilisation de vos Services AWS est déterminée par la sensibilité de vos données, les objectifs de conformité de votre entreprise, ainsi que par la législation et la réglementation applicables. AWS fournit les ressources suivantes pour faciliter la conformité :

- Conformité et gouvernance de la sécurité : ces guides de mise en œuvre de solutions traitent des considérations architecturales et fournissent les étapes à suivre afin de déployer des fonctionnalités de sécurité et de conformité.
- Référence des services éligibles HIPAA : liste les services éligibles HIPAA. Tous les ne Services AWS sont pas éligibles à HIPAA.

- AWS Ressources de https://aws.amazon.com/compliance/resources/ de conformité : cet ensemble de manuels et de guides peut s'appliquer à votre secteur et à votre emplacement.
- AWS Guides de conformité destinés aux clients — Comprenez le modèle de responsabilité partagée sous l'angle de la conformité. Les guides résument les meilleures pratiques pour sécuriser les Services AWS et décrivent les directives relatives aux contrôles de sécurité dans de nombreux cadres (y compris l'Institut national de normalisation et de technologie (NIST), le Conseil de normes de sécurité PCI (Payment Card Industry) et l'Organisation internationale de normalisation (ISO).
- Évaluation des ressources à l'aide de règles dans le Guide du AWS Config développeur : le AWS Config service évalue dans quelle mesure vos configurations de ressources sont conformes aux pratiques internes, aux directives sectorielles et aux réglementations.
- AWS Security Hub: ce Service AWS service fournit une vue complète de votre état de sécurité interne AWS. Security Hub utilise des contrôles de sécurité pour évaluer vos ressources AWS et vérifier votre conformité par rapport aux normes et aux bonnes pratiques du secteur de la sécurité. Pour obtenir la liste des services et des contrôles pris en charge, consultez Référence des contrôles Security Hub.
- Amazon GuardDuty : ce Service AWS détecte les menaces potentielles qui pèsent sur vos Comptes AWS, vos charges de travail, vos conteneurs et vos données en surveillant votre environnement à la recherche d'activités suspectes et malveillantes. GuardDuty peut vous aider à satisfaire diverses exigences de conformité, comme la conformité à la norme PCI DSS, en répondant aux exigences de détection d'intrusion imposées par certains frameworks de conformité.
- AWS Audit Manager: ce service vous Service AWS aide à auditer en continu votre AWS utilisation d'pour simplifier la gestion des risques et la conformité aux réglementations et aux normes du secteur.

Baliser des collections Amazon OpenSearch sans serveur

Les balises vous permettent d'affecter des informations arbitraires à une collection Amazon OpenSearch sans serveur afin d'effectuer un classement et un filtrage sur ces informations. Une balise est une étiquette de métadonnées que vous attribuez ou que AWS attribue à une AWS ressource.

Chaque balise se compose d’une clé et d’une valeur. Pour les balises que vous affectez, vous définissez la clé et la valeur. Par exemple, vous pouvez définir la clé sur `stage` et la valeur pour une ressource sur `test`.

Les balises vous permettent d'identifier et d'organiser vos AWS ressources. De nombreux AWS services prennent en charge le balisage. Vous pouvez donc attribuer la même balise à des ressources à partir de différents services pour indiquer que les ressources sont liées. Par exemple, vous pouvez affecter à une collection OpenSearch sans serveur la même balise que celle que vous affectez à un domaine Amazon OpenSearch Service.

Dans OpenSearch sans serveur, la ressource principale est une collection. Vous pouvez utiliser la console de OpenSearch service AWS CLI, les opérations de l'API OpenSearch Serverless ou AWS SDKs pour ajouter, gérer et supprimer des balises dans une collection.

Autorisations nécessaires

OpenSearch sans serveur utilise les autorisations AWS Identity and Access Management Access Analyzer (IAM) suivantes pour baliser des collections :

- `aoss:TagResource`
- `aoss:ListTagsForResource`
- `aoss:UntagResource`

Baliser des collections (console)

La console constitue le moyen le plus simple de baliser une collection.

Pour créer une identification (console)

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Développez Serverless (Sans serveur) dans le panneau de navigation de gauche et choisissez Collections.
3. Sélectionnez la collection à laquelle vous souhaitez ajouter des balises et accédez à l'onglet Tags (Balises).
4. Choisissez Manage (Gérer) et Add new tag (Ajouter une nouvelle balise).
5. Saisissez une clé de balise et une valeur de balise facultative.
6. Choisissez Save (Enregistrer).

Pour supprimer une balise, suivez les mêmes étapes et choisissez Remove (Supprimer) sur la page Manage tags (Gérer les balises).

Pour plus d'informations sur l'utilisation de la console avec des identifications, veuillez consulter Éditeur d'identification dans le Guide de démarrage de la console de gestion AWS .

Balisage de collections ()AWS CLI

Pour étiqueter une collection à l'aide du AWS CLI, envoyez une TagResourcedemande :

```
aws opensearchserverless tag-resource
  --resource-arn arn:aws:aoss:us-east-1:123456789012:collection/my-collection
  --tags Key=service,Value=aoss Key=source,Value=logs
```

Affichez les balises existantes pour une collection à l'aide de la ListTagsForResourcecommande :

```
aws opensearchserverless list-tags-for-resource
  --resource-arn arn:aws:aoss:us-east-1:123456789012:collection/my-collection
```

Supprimez les tags d'une collection à l'aide de la UntagResourcecommande :

```
aws opensearchserverless untag-resource
  --resource-arn arn:aws:aoss:us-east-1:123456789012:collection/my-collection
  --tag-keys service
```

Opérations et plugins pris en charge dans Amazon OpenSearch sans serveur

Amazon OpenSearch sans serveur prend en charge divers OpenSearch plugins, ainsi qu'un sous-ensemble des opérations d'API d'indexation, de recherche et de métadonnées disponibles dans. OpenSearch Vous pouvez inclure les autorisations dans la colonne de gauche du tableau dans les stratégies d'accès aux données afin de limiter l'accès à certaines opérations.

Rubriques

- Opérations et autorisations d' OpenSearch API prises en charge
- OpenSearch Plugins pris en charge

Opérations et autorisations d' OpenSearch API prises en charge

Le tableau suivant répertorie les opérations d'API prises en charge par OpenSearch sans serveur, ainsi que les autorisations d'accès aux données correspondantes :

Autorisation de stratégie d'accès aux données	OpenSearch Opérations d'API	Description et mises en garde
`aoss:CreateIndex`	PUT <index>	Créer des index. Pour plus d'informations, veuillez consulter la rubrique Créer un index (langue française non garantie). **Note** Cette autorisation s'applique également à la création d'index avec les exemples de données sur les OpenSearch tableaux de bord.
`aoss:DescribeIndex`	• GET <index> • GET <index>/_mapping • GET <index>/_mappings • GET <index>/_setting • GET <index>/_setting/<setting> • GET <index>/_settings • GET <index>/_settings/<setting> • GET _cat/indices • GET _mapping • GET _mappings • GET _resolve/index/<index> • TÊTE <index>	Décrire des index. Pour plus d'informations, consultez les ressources suivantes : • Obtenir un index • Obtenir un mappage • Obtenir les paramètres • L'index existe • Indices CAT (la réponse n'inclut `health` pas de `status` champs.)

Autorisation de stratégie d'accès aux données	OpenSearch Opérations d'API	Description et mises en garde
`aoss:WriteDocument`	• SUPPRIMER <index>/_doc/ <id> • POST <index>/_bulk • POST <index>/_create/<id> (uniquement pour les types de collection de recherche) • POST <index>/_doc • POST <index>/_update/<id> (uniquement pour les types de collection de recherche) • POST _bulk • PUT <index>/_create/<id> (uniquement pour les types de collection de recherche) • PUT <index>/_doc/<id> (uniquement pour les types de collection de recherche)	Rédiger et mettre à jour des documents. Pour plus d'informations, consultez les ressources suivantes : • En bloc • Données d'index **Note** Certaines opérations ne sont autorisées que pour les collections de type `SEARCH`. Pour de plus amples informations, veuillez consulter the section called "Choix d'un type de collection".

Autorisation de stratégie d'accès aux données	OpenSearch Opérations d'API	Description et mises en garde
`aoss:ReadDocument`	• OBTENEZ <index>/_analyz • GET <index>/_doc/<id> • GET <index>/_explain/<id> • GET <index>/_mget • GET <index>/_source/<id> • GET <index>/_count • GET <index>/_field_caps • GET <index>/_msearch • GET <index>/_rank_eval • GET <index>/_search • GET <index>/_validate/<query> • GET _analyze • GET _field_caps • GET _mget • GET _search • OBTENEZ /_search/point_in_time/_all • HEAD <index>/_doc/<id> • HEAD <index>/_source/<id> • POST /_plugins/_sql • PUBLIEZ /_plugins/_ppl • POST /_plugins/_sql/_explain • POST /_plugins/_ppl/_explain • PUBLIEZ /_plugins/_ppl/_close • POST <index>/_analyze • POST /_search/point_in_time • POST <index>/_explain/<id> • POST <index>/_count	Lire des documents. Pour plus d'informations, consultez les ressources suivantes : • Réaliser une analyse de texte • Obtenir un document • Count • Requête DSL • Évaluation du classement • Analyser l'API • Expliquer • Point dans le temps • SQL et PPL

Autorisation de stratégie d'accès aux données	OpenSearch Opérations d'API	Description et mises en garde
	• POST <index>/_field_caps • POST <index>/_rank_eval • POST <index>/_search • POST _analyze • POST _field_caps • POST _search • SUPPRIMER /_search/point_in_time/_all • SUPPRIMER /_search/point_in_time	
`aoss:DeleteIndex`	DELETE <target>	Supprimer les index. Pour plus d'informations, veuillez consulter la rubrique Supprimer un index (langue française non garantie).

Autorisation de stratégie d'accès aux données	OpenSearch Opérations d'API	Description et mises en garde
`aoss:UpdateIndex`	• POST _mapping • POST <index>/_mapping/ • POST <index>/_mappings/ • POST <index>/_setting • POST <index>/_settings • POST _setting • POST _settings • PUT _mapping • PUT <index>/_mapping • PUT <index>/_mappings/ • PUT <index>/_setting • PUT <index>/_settings • PUT _setting • PUT _settings	Mettre à jour les paramètres d'index. Pour plus d'informations, consultez les ressources suivantes : • Mappage • Mettre à jour les paramètres
`aoss:CreateCollectionItems`	POST _aliases	Créer des alias d'index. Pour plus d'informations, veuillez consulter la rubrique Créer des alias (langue française non garantie).

Autorisation de stratégie d'accès aux données	OpenSearch Opérations d'API	Description et mises en garde
`aoss:DescribeCollectionItems`	• GET <index>/_alias/<alias> • GET _alias • GET _alias/<alias> • GET _cat/aliases • GET _cat/templates • GET _cat/templates/<template_name> • GET _component_template • GET _component_template/<component-template> • GET _index_template • GET _index_template/<index-template> • HEAD _alias/<alias> • HEAD _component_template/<component-template> • HEAD _index_template/<name> • HEAD <index>/_alias/<alias>	Décrire les alias et les modèles d'index. Pour plus d’informations, consultez les ressources suivantes : • Gérer les alias • Modèles d'index

Autorisation de stratégie d'accès aux données	OpenSearch Opérations d'API	Description et mises en garde
`aoss:UpdateCollectionItems`	• POST <index>/_alias/<alias> • POST <index>/_aliases/<alias> • POST _component_template/<component-template> • POST _index_template/<index-template> • PUT <index>/_alias/<alias> • PUT <index>/_aliases/<alias> • PUT _component_template/<component-template> • PUT _index_template/<index-template>	Mettre à jour les alias et les modèles d'index. Pour plus d'informations, consultez les ressources suivantes : • Alias d'index • Modèles d'index
`aoss:DeleteCollectionItems`	• DELETE <index>/_alias/<alias> • DELETE _component_template/<component-template> • DELETE _index_template/<index-template> • DELETE <index>/_aliases/<alias>	Supprimer les alias et les modèles d'index. Pour plus d'informations, consultez les ressources suivantes : • Supprimer des alias • Supprimer un modèle

OpenSearch Plugins pris en charge

OpenSearch Les collections sans serveur sont fournies avec les plugins suivants, provenant de la OpenSearch communauté. La technologie sans serveur déploie et gère automatiquement les plugins pour vous.

Plugins d'analyse

- ICU Analysis
- Japanese (kuromoji) Analysis
- Korean (Nori) Analysis

- Phonetic Analysis
- Smart Chinese Analysis
- Stempel Polish Analysis
- Ukrainian Analysis

Plugins de mappage

- Mapper Size
- Mapper Murmur3
- Mapper Annotated Text

Plugins de script

- Painless
- Expression
- Mustache

En outre, OpenSearch sans serveur inclut tous les plugins fournis sous forme de modules.

Surveillance d'Amazon OpenSearch sans serveur

La surveillance constitue une partie importante du maintien de la fiabilité, de la disponibilité et des performances d'Amazon OpenSearch sans serveur et de vos autres AWS solutions. AWS fournit les outils de surveillance suivants pour contrôler OpenSearch sans serveur, signaler les problèmes et déclencher des actions automatiques, si nécessaire :

- Amazon CloudWatch contrôle vos AWS ressources et les applications que vous exécutez sur AWS en temps réel. Vous pouvez collecter et suivre les métriques, créer des tableaux de bord personnalisés, et définir des alarmes qui vous informent ou prennent des mesures lorsqu'une métrique spécifique atteint un seuil que vous spécifiez.

  Par exemple, vous pouvez connaître l'utilisation d'UC pour une CloudWatch piste ou d'autres métriques de vos EC2 instances Amazon et démarrer automatiquement de nouvelles instances lorsque cela est nécessaire. Pour de plus amples informations, veuillez consulter le Guide de CloudWatch l'utilisateur Amazon.

- AWS CloudTrail capture les appels d'API et les événements associés effectués par votre Compte AWS ou au nom de ce dernier. Il remet les fichiers journaux au compartiment Amazon S3 que vous spécifiez. Vous pouvez identifier les utilisateurs et les comptes qui ont appelé AWS, l'adresse IP source à partir de laquelle les appels ont été émis, et le moment où les appels ont eu lieu. Pour plus d'informations, consultez le Guide de l'utilisateur AWS CloudTrail.
- Amazon EventBridge fournit un flux d'événements système en temps quasi réel qui décrivent les modifications apportées à vos domaines OpenSearch de service. Vous pouvez créer des règles qui recherchent certains événements et déclenchent des actions automatisées dans d'autres services Services AWS lorsque ces événements se produisent. Pour de plus amples informations, veuillez consulter le Guide de EventBridge l'utilisateur Amazon.

Surveillance OpenSearch sans serveur avec Amazon CloudWatch

Vous pouvez contrôler Amazon OpenSearch sans serveur CloudWatch, qui collecte et traite les données brutes pour les transformer en métriques lisibles en quasi temps réel. Ces statistiques sont enregistrées pour une durée de 15 mois ; par conséquent, vous pouvez accéder aux informations historiques et acquérir un meilleur point de vue de la façon dont votre service ou application web s'exécute.

Vous pouvez également définir des alarmes qui surveillent certains seuils et envoient des notifications ou prennent des mesures lorsque ces seuils sont atteints. Pour de plus amples informations, veuillez consulter le Guide de CloudWatch l'utilisateur Amazon.

OpenSearch La technologie sans serveur signale les métriques suivantes dans l'espace de `AWS/AOSS` noms.

Métrique	Description
`ActiveCollection`	Indique si une collection est active. La valeur 1 signifie que la collection est à l'état `ACTIVE`. Cette valeur est émise lors de la création réussie d'une collection et reste égale à 1 jusqu'à ce que vous supprimiez la collection. La métrique ne peut pas être égale à 0. Statistiques pertinentes : maximum Dimensions : `ClientId`, `CollectionId` , `CollectionName`

Métrique	Description
	Fréquence : 60 secondes
`DeletedDocuments`	Nombre total de documents supprimés. Statistiques pertinentes : moyenne, somme Dimensions : `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Fréquence : 60 secondes
`IndexingOCU`	Nombre d'unités de OpenSearch calcul (OCUs) utilisées pour ingérer les données de collection. Cette métrique s'applique au niveau du compte. Statistiques pertinentes : somme Dimensions : `ClientId` Fréquence : 60 secondes
`IngestionDataRate`	Taux d'indexation en Gio par seconde vers une collection ou un index. Cette métrique s'applique uniquement aux requêtes d'indexation en bloc. Statistiques pertinentes : somme Dimensions : `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Fréquence : 60 secondes

Métrique	Description
`IngestionDocumentErrors`	Nombre total d'erreurs de document lors de l'ingesti on pour une collection ou un index. Après une requête d'indexation en bloc réussie, les enregistreurs traitent la requête et émettent des erreurs pour tous les documents ayant échoué inclus dans la requête. Statistiques pertinentes : somme Dimensions : `ClientId`, `CollectionId` , `Collectio nName` , `IndexId`, `IndexName` Fréquence : 60 secondes
`IngestionDocumentRate`	Taux par seconde auquel les documents sont ingérés dans une collection ou un index. Cette métrique s'applique uniquement aux requêtes d'indexation en bloc. Statistiques pertinentes : somme Dimensions : `ClientId`, `CollectionId` , `Collectio nName` , `IndexId`, `IndexName` Fréquence : 60 secondes
`IngestionRequestErrors`	Nombre total d'erreurs de requête d'indexation en bloc dans une collection. OpenSearch La technologie sans serveur émet cette métrique lorsqu'une requête d'indexat ion en bloc échoue pour une raison quelconque, telle qu'un problème d'authentification ou de disponibilité. Statistiques pertinentes : somme Dimensions : `ClientId`, `CollectionId` , `Collectio nName` Fréquence : 60 secondes

Métrique	Description
`IngestionRequestLatency`	Latence, en secondes, pour les opérations d'écriture en bloc dans une collection. Statistiques pertinentes : minimum, maximum, moyenne Dimensions : `ClientId`, `CollectionId` , `CollectionName` Fréquence : 60 secondes
`IngestionRequestRate`	Nombre total d'opérations d'écriture en bloc reçues par une collection. Statistiques pertinentes : minimum, maximum, moyenne Dimensions : `ClientId`, `CollectionId` , `CollectionName` Fréquence : 60 secondes
`IngestionRequestSuccess`	Nombre total d'opérations d'indexation réussies dans une collection. Statistiques pertinentes : somme Dimensions : `ClientId`, `CollectionId` , `CollectionName` Fréquence : 60 secondes
`SearchableDocuments`	Nombre total de documents consultables dans une collection ou un index. Statistiques pertinentes : somme Dimensions : `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Fréquence : 60 secondes

Métrique	Description
`SearchRequestErrors`	Nombre total d'erreurs de requête par minute pour une collection. Statistiques pertinentes : somme Dimensions : `ClientId`, `CollectionId` , `CollectionName` Fréquence : 60 secondes
`SearchRequestLatency`	Temps moyen nécessaire, en millisecondes, pour terminer une opération de recherche dans une collection. Statistiques pertinentes : minimum, maximum, moyenne Dimensions : `ClientId`, `CollectionId` , `CollectionName` Fréquence : 60 secondes
`SearchOCU`	Nombre d'unités de OpenSearch calcul (OCUs) utilisées pour rechercher des données de collection. Cette métrique s'applique au niveau du compte. Statistiques pertinentes : somme Dimensions : `ClientId` Fréquence : 60 secondes
`SearchRequestRate`	Nombre total de requêtes de recherche par minute pour une collection. Statistiques pertinentes : moyenne, maximum, somme Dimensions : `ClientId`, `CollectionId` , `CollectionName` Fréquence : 60 secondes

Métrique	Description
`StorageUsedInS3`	La quantité, en octets, de stockage Amazon S3 utilisée. OpenSearch sans serveur stocke les données indexées dans Amazon S3. Vous devez sélectionner la période à une minute pour obtenir une valeur précise. Statistiques pertinentes : somme Dimensions : `ClientId`, `CollectionId` , `CollectionName` , `IndexId`, `IndexName` Fréquence : 60 secondes
`2xx, 3xx, 4xx, 5xx`	Nombre de requêtes adressées à une collection ayant entraîné le code de réponse HTTP donné (2xx, 3xx, 4xx, 5xx). Statistiques pertinentes : somme Dimensions : `ClientId`, `CollectionId` , `CollectionName` Fréquence : 60 secondes

Journaliser les appels d'API OpenSearch sans serveur à l'aide de AWS CloudTrail

Amazon OpenSearch sans serveur est intégré à AWS CloudTrail, service qui enregistre les actions effectuées par un utilisateur, un rôle ou un AWS service dans la technologie sans serveur.

CloudTrail capture tous les appels d'API pour la technologie OpenSearch sans serveur en tant qu'événements. Les appels enregistrés contiennent les appels de la section sans serveur de la console de OpenSearch service et les appels de code vers les opérations d'API OpenSearch sans serveur.

Si vous créez un journal d'activité, vous pouvez activer la livraison continue d' CloudTrail événements à un compartiment Amazon S3, y compris des événements pour OpenSearch sans serveur. Si vous

ne configurez pas de journal de suivi, vous pouvez toujours afficher les événements les plus récents dans la CloudTrail console dans Event history (Historique des événements).

À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été envoyée à la demande, l'adresse IP à partir de laquelle la demande a été effectuée, l' OpenSearch auteur de la demande, la date de la demande, la demande, la demande, l'auteur de la demande, la date de la demande, les informations collectées par, l'auteur de la demande, la date de la demande, les informations collectées par

Pour en savoir plus CloudTrail, consultez le guide de AWS CloudTrail l'utilisateur.

OpenSearch Informations sans serveur dans CloudTrail

CloudTrail est activé sur votre Compte AWS lorsque vous créez le compte. Lorsqu'une activité a lieu dans la OpenSearch technologie sans serveur, cette activité est enregistrée dans un CloudTrail événement avec d'autres événements de AWS service dans Event history (Historique des événements). Vous pouvez afficher, rechercher et télécharger les événements récents dans votre Compte AWS. Pour de plus amples informations, veuillez consulter Affichage des événements avec l'historique des CloudTrail événements.

Pour un enregistrement continu des événements dans votre Compte AWS, y compris les événements pour la technologie OpenSearch sans serveur, créez un journal d'activité. Un journal de suivi CloudTrail permet de livrer des fichiers journaux dans un compartiment Amazon S3. Par défaut, lorsque vous créez un journal d’activité dans la console, il s’applique à toutes les régions Régions AWS.

Le journal de suivi consigne les événements de toutes les Régions dans la AWS partition et livre les fichiers journaux dans le compartiment Amazon S3 de votre choix. En outre, vous pouvez configurer d'autres AWS services pour analyser plus en profondeur les données d'événement collectées dans les CloudTrail journaux et agir sur celles-ci. Pour plus d’informations, consultez les ressources suivantes :

- Présentation de la création d’un journal de suivi
- CloudTrail services et intégrations pris en charge
- Configuration des notifications Amazon SNS pour CloudTrail
- Réception de fichiers CloudTrail journaux de plusieurs régions et Réception de fichiers CloudTrail journaux de plusieurs comptes

Toutes les actions OpenSearch Serverless sont enregistrées CloudTrail et documentées dans la référence de l'API OpenSearch Serverless. Par exemple, les appels aux `CreateCollection` `ListCollections`, et `DeleteCollection` les actions génèrent des entrées dans les fichiers CloudTrail journaux.

Chaque événement ou entrée de journal contient des informations sur la personne ayant initié la demande. Les informations relatives à l'identité vous permettent de déterminer :

- Si la demande a été effectuée avec les informations d'identification utilisateur racine ou AWS Identity and Access Management (IAM).
- Si la demande a été effectuée avec les informations d'identification de sécurité temporaires d'un rôle ou d'un utilisateur fédéré.
- Si la demande a été effectuée par un autre AWS service.

Pour de plus amples informations, veuillez consulter l'élément userIdentity CloudTrail .

Présentation des OpenSearch entrées des fichiers journaux sans serveur

Un journal d'activité est une configuration qui permet d'envoyer des événements sous forme de fichiers journaux à un compartiment Simple Storage Service (Amazon S3) que vous spécifiez. CloudTrail les fichiers journaux peuvent contenir une ou plusieurs entrées de journal.

Un événement représente une demande individuelle d'une source quelconque. Il comprend les informations sur l'action demandée, la date et l'heure de l'action, les paramètres de la demande, etc. CloudTrail les fichiers journaux ne constituent pas une pile ordonnée retraçant les appels d'API publics. donc ils ne suivent aucun ordre précis.

L'exemple suivant montre une entrée de CloudTrail journal qui illustre l'`CreateCollection`action.

```
{
   "eventVersion":"1.08",
   "userIdentity":{
      "type":"AssumedRole",
      "principalId":"AIDACKCEVSQ6C2EXAMPLE",
      "arn":"arn:aws:iam::123456789012:user/test-user",
      "accountId":"123456789012",
      "accessKeyId":"access-key",
      "sessionContext":{
         "sessionIssuer":{
            "type":"Role",
```

```
            "principalId":"AIDACKCEVSQ6C2EXAMPLE",
            "arn":"arn:aws:iam::123456789012:role/Admin",
            "accountId":"123456789012",
            "userName":"Admin"
         },
         "webIdFederationData":{

         },
         "attributes":{
            "creationDate":"2022-04-08T14:11:34Z",
            "mfaAuthenticated":"false"
         }
      }
   },
   "eventTime":"2022-04-08T14:11:49Z",
   "eventSource":"aoss.amazonaws.com",
   "eventName":"CreateCollection",
   "awsRegion":"us-east-1",
   "sourceIPAddress":"AWS Internal",
   "userAgent":"aws-cli/2.1.30 Python/3.8.8 Linux/5.4.176-103.347.amzn2int.x86_64 exe/
x86_64.amzn.2 prompt/off command/aoss.create-collection",
   "errorCode":"HttpFailureException",
   "errorMessage":"An unknown error occurred",
   "requestParameters":{
      "accountId":"123456789012",
      "name":"test-collection",
      "description":"A sample collection",
      "clientToken":"d3a227d2-a2a7-49a6-8fb2-e5c8303c0718"
   },
   "responseElements": null,
   "requestID":"12345678-1234-1234-1234-987654321098",
   "eventID":"12345678-1234-1234-1234-987654321098",
   "readOnly":false,
   "eventType":"AwsApiCall",
   "managementEvent":true,
   "recipientAccountId":"123456789012",
   "eventCategory":"Management",
   "tlsDetails":{
      "clientProvidedHostHeader":"user.aoss-sample.us-east-1.amazonaws.com"
   }
}
```

Surveillance des événements OpenSearch sans serveur à l'aide d'Amazon EventBridge

Amazon OpenSearch Service s'intègre EventBridge à Amazon pour vous avertir de certains événements qui affectent vos domaines. Les événements des AWS services sont fournis à presque EventBridge en temps réel. Les mêmes événements sont également envoyés à Amazon CloudWatch Events, le prédécesseur d'Amazon EventBridge. Vous pouvez écrire des règles pour indiquer quels événements vous intéressent et les actions automatisées à effectuer quand un événement correspond à une règle. Voici quelques exemples d'actions que vous pouvez activer automatiquement :

- Appel d'une fonction AWS Lambda
- Appel d'une Run Command Amazon Run Command Amazon EC2 Run
- Relais de l'événement à Amazon Kinesis Data Streams
- Activation d'une machine d'état AWS Step Functions
- Notification d'une rubrique Amazon SNS ou d'une file d'attente Amazon SQS

Pour plus d'informations, consultez Commencer avec Amazon EventBridge dans le guide de EventBridge l'utilisateur Amazon.

Configuration des notifications

Vous pouvez utiliser les notifications AWS utilisateur pour recevoir des notifications lorsqu'un événement OpenSearch sans serveur se produit. Un événement est un indicateur d'un changement dans l'environnement OpenSearch sans serveur, par exemple lorsque vous atteignez la limite maximale d'utilisation de votre OCU. Amazon EventBridge reçoit l'événement et envoie une notification au centre de AWS Management Console notifications et aux canaux de diffusion que vous avez choisis. Vous recevez une notification lorsqu'un événement correspond à une règle que vous avez spécifiée.

OpenSearch Événements relatifs aux unités de calcul (OCU)

OpenSearch La technologie sans serveur envoie des événements EventBridge lorsque l'un des événements suivants liés à l'OCU se produit.

L'utilisation de l'OCU approche de la limite maximale

OpenSearch Serverless envoie cet événement lorsque l'utilisation de votre OCU de recherche ou d'indexation atteint 75 % de votre limite de capacité. L'utilisation de votre OCU est calculée en fonction de votre limite de capacité configurée et de votre consommation actuelle d'OCU.

Exemple

Voici un exemple d'événement de ce type (recherche OCU) :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "OCU Utilization Approaching Max Limit",
  "source": "aws.aoss",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "eventTime" : 1678943345789,
    "description": "Your search OCU usage is at 75% and is approaching the configured
 maximum limit."
  }
}
```

Voici un exemple d'événement de ce type (index OCU) :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "OCU Utilization Approaching Max Limit",
  "source": "aws.aoss",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "eventTime" : 1678943345789,
    "description": "Your indexing OCU usage is at 75% and is approaching the configured
 maximum limit."
  }
```

L'utilisation de l'OCU a atteint la limite maximale

OpenSearch Serverless envoie cet événement lorsque l'utilisation de votre OCU de recherche ou d'indexation atteint 100 % de votre limite de capacité. L'utilisation de votre OCU est calculée en fonction de votre limite de capacité configurée et de votre consommation actuelle d'OCU.

Exemple

Voici un exemple d'événement de ce type (recherche OCU) :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "OCU Utilization Reached Max Limit",
  "source": "aws.aoss",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "eventTime" : 1678943345789,
    "description": "Your search OCU usage has reached the configured maximum limit."
  }
}
```

Voici un exemple d'événement de ce type (index OCU) :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "OCU Utilization Reached Max Limit",
  "source": "aws.aoss",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "eventTime" : 1678943345789,
    "description": "Your indexing OCU usage has reached the configured maximum limit."
  }
}
```

Création et gestion de domaines Amazon OpenSearch Service

Ce chapitre explique comment créer et gérer des domaines Amazon OpenSearch Service. Un domaine est l'équivalent AWS-provisionné d'un cluster open source OpenSearch . Lorsque vous créez un domaine, vous spécifiez ses paramètres, les types d'instances, le nombre d'instances et l'allocation de stockage. Pour plus d'informations sur les clusters open source, consultez la section Création d'un cluster dans la OpenSearch documentation.

Contrairement aux instructions rapidement exposées dans le didacticiel Mise en route, ce chapitre décrit toutes les options et fournit des informations de référence pertinentes. Vous pouvez effectuer chaque procédure en utilisant les instructions de la console de OpenSearch service, du AWS Command Line Interface (AWS CLI) ou du AWS SDKs.

Création de domaines OpenSearch de service

Cette section décrit comment créer des domaines de OpenSearch service à l'aide de la console de OpenSearch service ou à l'aide de la `create-domain` commande AWS CLI with the.

Création OpenSearch de domaines de service (console)

Utilisez la procédure suivante pour créer un domaine OpenSearch de service à l'aide de la console.

Pour créer un domaine OpenSearch de service (console)

1. Accédez à la console https://aws.amazon.comet choisissez Se connecter à la console.
2. Sous Analytics, sélectionnez Amazon OpenSearch Service.
3. Choisissez Create domain (Créer un domaine).
4. Pour Nom de domaine, entrez un nom de domaine. Le nom doit répondre aux critères suivants :
   - Unique à votre compte et Région AWS
   - Commence par une lettre minuscule
   - Contient entre 3 et 28 caractères
   - Contient uniquement les lettres minuscules a-z, les chiffres 0-9 et le trait d'union (-)
5. Pour la méthode de création de domaine, choisissez Standard create.

6. Pour les modèles, choisissez l'option qui correspond le mieux à l'objectif de votre domaine :

   - Domaines de production pour les charges de travail nécessitant une disponibilité et des performances élevées. Ces domaines utilisent des nœuds multi-AZ (avec ou sans veille) et des nœuds maîtres dédiés pour une meilleure disponibilité.
   - Développement/test pour le développement ou les tests. Ces domaines peuvent utiliser le mode multi-AZ (avec ou sans veille) ou une seule zone de disponibilité.

   > **Important**
   >
   > Les différents types de déploiement offrent différentes options sur les pages suivantes. Ces étapes incluent toutes les options.

7. Pour les options de déploiement, choisissez Domaine avec veille pour configurer un domaine 3-AZ, les nœuds de l'une des zones étant réservés en mode veille. Cette option applique un certain nombre de bonnes pratiques, telles que le nombre de nœuds de données spécifié, le nombre de nœuds maîtres, le type d'instance, le nombre de répliques et les paramètres de mise à jour logicielle.
8. Dans Version, choisissez la version OpenSearch ou l'ancienne version d'Elasticsearch OSS à utiliser. Nous vous recommandons de choisir la dernière version de OpenSearch. Pour de plus amples informations, veuillez consulter the section called “Versions prises en charge d'Elasticsearch et OpenSearch”.

   (Facultatif) Si vous avez choisi une OpenSearch version pour votre domaine, sélectionnez Activer le mode de compatibilité pour OpenSearch signaler sa version 7.10, ce qui permet à certains clients et plugins Elasticsearch OSS qui vérifient la version avant de se connecter de continuer à utiliser le service.
9. Pour Instance type (Type d'instance), choisissez un type d'instance pour vos nœuds de données. Pour en savoir plus, consultez the section called “Types d’instance pris en charge”.

   > **Note**
   >
   > Certaines zones de disponibilité ne prennent pas en charge certains types d'instance. Si vous choisissez le mode Multi-AZ avec ou sans veille, nous vous recommandons de choisir les types d'instances de génération actuelle, tels que R5 ou I3.

10. Pour Nombre d'instances, tapez le nombre de nœuds de données.

Pour les valeurs maximales, consultez la section Quotas de domaine et d'instance du OpenSearch service. Les clusters à nœud unique conviennent aux phases de développement et de test, mais pas pour des charges de travail en production. Pour de plus amples informations, veuillez consulter the section called “Dimensionnement des domaines” et the section called “Configuration d'un domaine Multi-AZ”.

> **Note**
>
> (Facultatif) Les nœuds de coordination dédiés prennent en charge toutes les OpenSearch versions et ElasticSearch les versions 6.8 à 7.10. Des nœuds de coordination dédiés peuvent être utilisés avec des domaines pour lesquels un gestionnaire de cluster dédié est activé. Pour activer les nœuds de coordination dédiés, vous devez sélectionner le type et le nombre d'instances. Il est recommandé de conserver la même famille d'instances pour votre nœud coordinateur dédié que pour vos nœuds de données (instances basées sur Intel ou instances basées sur Graviton).

11. Pour le type de stockage, sélectionnez Amazon EBS. Les types de volumes disponibles dans la liste dépendent du type d’instance que vous avez sélectionné. Pour obtenir des conseils sur la création de domaines particulièrement volumineux, consultez the section called “Mise à l'échelle d'une capacité de plusieurs péta-octets”.
12. Pour le stockage EBS, configurez les paramètres supplémentaires suivants. Certains paramètres peuvent ne pas apparaître en fonction du type de volume choisi.

Paramètre	Description
Type de volume EBS	Choisissez entre Usage général (SSD) – gp3 et Usage général (SSD) – gp2, ou la génération précédente IOPS provisionnés (SSD), et Magnétique (standard).
Taille de stockage EBS par nœud	Saisissez la taille du volume EBS que vous souhaitez attacher à chaque nœud de données. La taille du volume EBS est indiquée par nœud. Vous pouvez calculer la taille totale du cluster pour le domaine de OpenSearch service en multipliant le nombre de nœuds de données par la taille du volume EBS. Les tailles minimale et maximale d'un volume EBS dépendent à la fois du type de volume EBS

Paramètre	Description
	spécifié et du type d'instance auquel il est attaché. Pour plus d'informations, consultez Limites relatives à la taille du volume EBS.
IOPS provisionnés	Si vous avez sélectionné un type de volume SSD IOPS provisionné, entrez le nombre d' I/O opérations par seconde (IOPS) que le volume peut prendre en charge.

13. (Facultatif) Si vous avez sélectionné un type de gp3 volume, étendez les paramètres avancés et spécifiez des IOPS supplémentaires (jusqu'à 16 000 pour chaque volume de 3 TiB fourni par nœud de données) et un débit (jusqu'à 1 000 pour MiB/s chaque volume de 3 TiB fourni par nœud de données) au-delà de ce qui est inclus dans le prix du stockage, moyennant un coût supplémentaire. Pour plus d'informations, consultez les tarifs d'Amazon OpenSearch Service.
14. (Facultatif) Pour activer le UltraWarm stockage, choisissez Activer UltraWarm les nœuds de données. Chaque type d'instance dispose d'une quantité maximale de stockage qu'il peut traiter. Multipliez cette quantité par le nombre de nœuds de données à chaud pour le stockage à chaud adressable total.
15. (Facultatif) Pour activer le stockage à froid, choisissez Enable cold storage (Activer le stockage à froid). Vous devez activer le stockage UltraWarm à froid pour activer le stockage à froid.
16. Si vous utilisez le mode Multi-AZ en mode veille, trois nœuds maîtres dédiés sont déjà activés. Choisissez le type de nœuds maîtres que vous souhaitez. Si vous avez choisi un domaine Multi-AZ sans veille, sélectionnez Activer les nœuds maîtres dédiés et choisissez le type et le nombre de nœuds principaux souhaités. Les nœuds principaux dédiés augmentent la stabilité du cluster et sont obligatoires pour les domaines dont le nombre d'instances est supérieur à 10. Pour les domaines de production, nous vous recommandons trois nœuds principaux dédiés.

> **Note**
>
> Vous pouvez choisir différents types d'instance pour les nœuds principaux dédiés et les nœuds de données. Par exemple, vous pouvez sélectionner des instances à visée générale ou optimisées pour le stockage pour vos nœuds de données, mais des instances optimisées pour le calcul pour vos nœuds principaux dédiés.

17. (Facultatif) Pour les domaines exécutant OpenSearch Elasticsearch 5.3 et versions ultérieures, la configuration Snapshot n'est pas pertinente. Pour plus d'informations sur les instantanés automatiques, consultez the section called “Création d'instantanés d'index”.
18. Si vous souhaitez utiliser un point de terminaison personnalisé plutôt que le point de terminaison standard de `https://search-`*`mydomain`*`-`*`1a2a3a4a5a6a7a8a9a0a9a8a7a`*`.`*`us-east-1`*`.es.amazonaws.com`, choisissez Enable custom endpoint (Activer un point de terminaison personnalisé), puis fournissez un nom et un certificat. Pour de plus amples informations, veuillez consulter the section called “Création d'un point de terminaison personnalisé”.
19. Sous Network (Réseau), choisissez soit VPC Access (Accès VPC), soit Public access (Accès public). Si vous choisissez Public access (Accès public), passez à l'étape suivante. Si vous choisissez VPC Access (Accès VPC), assurez-vous d'avoir respecté les conditions préalables, puis configurez les paramètres suivants :

Paramètre	Description
VPC	Choisissez le cloud privé virtuel (VPC) que vous souhaitez utiliser. Le VPC et le domaine doivent être identiques Région AWS, et vous devez sélectionner un VPC dont la location est définie sur Par défaut. OpenSearch Le service ne prend pas encore en charge VPCs l'utilisation de la location dédiée.
Sous-réseau	Choisissez un sous-réseau. Si vous avez activé le mode multi-AZ, vous devez choisir deux ou trois sous-réseaux. OpenSearch Le service placera un point de terminaison VPC et des interfaces réseau élastiques dans les sous-réseaux. Vous devez réserver suffisamment d'adresses IP pour les interfaces réseau dans le/les sous-réseau(x). Pour plus d'informations, consultez Réservation d'adresses IP dans un sous-réseau VPC.
Groupes de sécurité	Choisissez un ou plusieurs groupes de sécurité VPC qui permettent à l'application requise d'atteindre le domaine de OpenSearch service sur les ports (80 ou 443) et les protocoles (HTTP ou HTTPS) exposés par le domaine. Pour de plus amples informations, veuillez consulter the section called “Prise en charge de VPC”.

Paramètre	Description
IAM Role	Conservez le rôle par défaut. OpenSearch Le service utilise ce rôle prédéfini (également appelé rôle lié au service) pour accéder à votre VPC et pour placer un point de terminaison VPC et des interfaces réseau dans le sous-réseau du VPC. Pour plus d'informations, consultez Rôle lié à un service pour l'accès VPC.
Type d'adresse IP	Choisissez le type d'adresse IP à double pile ou IPv4 le type d'adresse IP. La double pile vous permet de partager les ressources du domaine entre IPv4 différents types d' IPv6 adresses. C'est l'option recommandée. Si vous définissez le type d'adresse IP sur Dual Stack, vous ne pourrez pas le modifier ultérieurement.

20. Activer ou désactiver le contrôle précis des accès :

    - Si vous souhaitez utiliser IAM pour la gestion des utilisateurs, choisissez Set IAM ARN as master user (Définir l'ARN IAM en tant qu'utilisateur principal), puis indiquez l'ARN d'un rôle IAM.
    - Si vous souhaitez utiliser la base de données utilisateur interne, choisissez Create master user et spécifiez un nom d'utilisateur et un mot de passe.

    Quelle que soit l'option choisie, l'utilisateur principal peut accéder à tous les index du cluster et à tout OpenSearch APIs. Pour de plus amples informations sur l'option à choisir, veuillez consulter the section called “Concepts clés”.

    Si vous désactivez le contrôle d'accès affiné, vous pouvez toujours contrôler l'accès à votre domaine en le plaçant dans un VPC, en appliquant une stratégie d'accès restrictive, ou les deux. Vous devez activer node-to-node le chiffrement et le chiffrement au repos pour bénéficier d'un contrôle d'accès précis.

    > **Note**
    >
    > Nous vous recommandons vivement d'activer le contrôle précis des accès pour protéger les données de votre domaine. Le contrôle précis des accès assure la sécurité au niveau du cluster, de l'index, du document et du champ.

21. (Facultatif) Si vous souhaitez utiliser l'authentification SAML pour les OpenSearch tableaux de bord, choisissez Activer l'authentification SAML et configurez les options SAML pour le domaine. Pour obtenir des instructions, veuillez consulter the section called “Authentification SAML pour les tableaux de bord OpenSearch ”.
22. (Facultatif) Si vous souhaitez utiliser l'authentification Amazon Cognito pour les OpenSearch tableaux de bord, choisissez Activer l'authentification Amazon Cognito. Choisissez ensuite le groupe d'utilisateurs et le groupe d'identités Amazon Cognito que vous souhaitez utiliser pour l'authentification des OpenSearch tableaux de bord. Pour obtenir de l'aide pour créer ces ressources, consultez the section called “Authentification Amazon Cognito pour les tableaux de bord OpenSearch ”.
23. Pour la politique d'accès, choisissez une politique d'accès ou configurez l'une des vôtres. Si vous choisissez de créer une politique personnalisée, vous pouvez la configurer vous-même ou en importer une à partir d'un autre domaine. Pour en savoir plus, consultez the section called “Gestion de l’identité et des accès”.

    > **Note**
    >
    > Si vous avez activé l'accès VPC, vous ne pouvez pas utiliser des stratégies d'accès basées sur l'IP. Vous devez plutôt utiliser des groupes de sécurité pour contrôler les adresses IP qui peuvent accéder au domaine. Pour en savoir plus, consultez the section called “À propos des stratégies d'accès pour les domaines de VPC”.

24. (Facultatif) Pour exiger que toutes les demandes envoyées au domaine arrivent via HTTPS, cochez la case Exiger HTTPS pour tout le trafic vers le domaine). Pour activer node-to-node le chiffrement, sélectionnez le ode-to-node chiffrement N. Pour de plus amples informations, veuillez consulter the section called “Node-to-node chiffrement”. Pour activer le chiffrement des données au repos, sélectionnez Activer le chiffrement des données au repos. Ces options sont présélectionnées si vous avez choisi l'option de déploiement Multi-AZ avec mode veille.
25. (Facultatif) Sélectionnez Utiliser une clé AWS détenue pour que le OpenSearch Service crée une clé de AWS KMS chiffrement en votre nom (ou utilise celle qu'il a déjà créée). Sinon, choisissez votre propre clé KMS. Pour de plus amples informations, veuillez consulter the section called “Chiffrement au repos”.
26. Pour les périodes creuses, sélectionnez une heure de début pour planifier les mises à jour du logiciel de service et les optimisations Auto-Tune nécessitant un déploiement. blue/green Les mises à jour hors pointe permettent de minimiser la pression sur les nœuds principaux dédiés d'un cluster pendant les périodes de trafic élevé.

27. Pour Auto-Tune, choisissez d'autoriser le OpenSearch service à suggérer des modifications de configuration liées à la mémoire pour votre domaine afin d'améliorer la vitesse et la stabilité. Pour de plus amples informations, veuillez consulter the section called “Auto-Tune”.

    (Facultatif) Sélectionnez Fenêtre creuse pour planifier une fenêtre récurrente au cours de laquelle Auto-Tune met à jour le domaine.
28. (Facultatif) Sélectionnez Mise à jour logicielle automatique pour activer les mises à jour logicielles automatiques.
29. (Facultatif) Ajoutez des étiquettes pour décrire votre domaine afin de pouvoir classer et filtrer ces informations. Pour plus d'informations, consultez the section called “Balisage des domaines”.
30. (Facultatif) Développez et configurez Paramètres avancés de cluster. Pour un résumé de ces options, consultez la section the section called “Paramètres avancés du cluster”.
31. Choisissez Créer.

Création OpenSearch de domaines de service (AWS CLI)

Au lieu de créer un domaine OpenSearch de service à l'aide de la console, vous pouvez utiliser le AWS CLI. Pour la syntaxe, consultez Amazon OpenSearch Service dans la référence de commande AWS CLI a.

Exemples de commandes

Ce premier exemple illustre la configuration du domaine OpenSearch de service suivante :

- Crée un domaine OpenSearch de service nommé mylogs avec OpenSearch la version 1.2
- Remplit le domaine avec deux instances du type `r6g.large.search`
- Utilise un volume de stockage EBS `gp3` à usage général (SSD) de 100 Gio pour chaque nœud de données
- Autorise l'accès anonyme, mais uniquement à partir d'une seule adresse IP : 192.0.2.0/32

```
aws opensearch create-domain \
    --domain-name mylogs \
    --engine-version OpenSearch_1.2 \
    --cluster-config  InstanceType=r6g.large.search,InstanceCount=2 \
    --ebs-options
 EBSEnabled=true,VolumeType=gp3,VolumeSize=100,Iops=3500,Throughput=125 \
```

```
    --access-policies '{"Version": "2012-10-17", "Statement": [{"Action": "es:*",
 "Principal":"*","Effect": "Allow", "Condition": {"IpAddress":{"aws:SourceIp":
["192.0.2.0/32"]}}}]}'
```

L'exemple suivant illustre la configuration du domaine OpenSearch de service suivante :

- Crée un domaine OpenSearch de service nommé mylogs avec Elasticsearch version 7.10
- Remplit le domaine avec six instances du type `r6g.large.search`
- Utilise un volume de stockage EBS gp2 à usage général (SSD) de 100 Gio pour chaque nœud de données
- Restreint l'accès au service à un seul utilisateur, identifié par son Compte AWS identifiant : 555555555555
- Répartit des instances dans trois zones de disponibilité

```
aws opensearch create-domain \
    --domain-name mylogs \
    --engine-version Elasticsearch_7.10 \
    --cluster-config
 InstanceType=r6g.large.search,InstanceCount=6,ZoneAwarenessEnabled=true,ZoneAwarenessConfig={A
 \
    --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=100 \
    --access-policies '{"Version": "2012-10-17", "Statement": [ { "Effect": "Allow",
 "Principal": {"AWS": "arn:aws:iam::555555555555:root" }, "Action":"es:*", "Resource":
 "arn:aws:es:us-east-1:555555555555:domain/mylogs/*" } ] }'
```

L'exemple suivant illustre la configuration du domaine OpenSearch de service suivante :

- Crée un domaine OpenSearch de service nommé mylogs avec OpenSearch la version 1.0
- Remplit le domaine avec dix instances du type `r6g.xlarge.search`
- Remplit le domaine avec trois instances du type `r6g.large.search` comme nœuds principaux dédiés
- Utilise un volume de stockage EBS d'IOPS provisionnés de 100 Gio, configuré avec des performances de base de 1 000 IOPS pour chaque nœud de données
- Limite l'accès à un seul utilisateur et à une seule sous-ressource, l'API `_search`

```
aws opensearch create-domain \
```

```
    --domain-name mylogs \
    --engine-version OpenSearch_1.0 \
    --cluster-config
InstanceType=r6g.xlarge.search,InstanceCount=10,DedicatedMasterEnabled=true,DedicatedMasterTyp
\
    --ebs-options EBSEnabled=true,VolumeType=io1,VolumeSize=100,Iops=1000 \
    --access-policies '{"Version": "2012-10-17", "Statement": [ { "Effect": "Allow",
"Principal": { "AWS": "arn:aws:iam::555555555555:root" }, "Action": "es:*",
"Resource": "arn:aws:es:us-east-1:555555555555:domain/mylogs/_search" } ] }'
```

> **Note**
>
> Si vous tentez de créer un domaine de OpenSearch service alors qu'un domaine portant le même nom existe déjà, la CLI ne signale aucune erreur. Au lieu de cela, elle renvoie les détails pour le domaine existant.

Création OpenSearch de domaines de service (AWS SDKs)

AWS SDKs (sauf Android et iOS SDKs) prennent en charge toutes les actions définies dans le Amazon OpenSearch Service API Reference, y compris`CreateDomain`. Pour un exemple de code, consultez the section called “À l'aide du AWS SDKs”. Pour plus d'informations sur l'installation et l'utilisation du AWS SDKs, voir Kits de développement AWS logiciel.

Création OpenSearch de domaines de service (AWS CloudFormation)

OpenSearch Le service est intégré à AWS CloudFormation un service qui vous aide à modéliser et à configurer vos AWS ressources afin que vous puissiez passer moins de temps à créer et à gérer vos ressources et votre infrastructure. Vous créez un modèle qui décrit le OpenSearch domaine que vous souhaitez créer, et qui CloudFormation approvisionne et configure le domaine pour vous. Pour plus d'informations, notamment des exemples de modèles JSON et YAML pour les OpenSearch domaines, consultez la référence au type de ressource Amazon OpenSearch Service dans le guide de l'AWS CloudFormation utilisateur.

Configuration des politiques d'accès

Amazon OpenSearch Service propose plusieurs méthodes pour configurer l'accès à vos domaines OpenSearch de service. Pour plus d’informations, consultez the section called “Gestion de l’identité et des accès” et the section called “Contrôle précis des accès”.

La console fournit des stratégies d'accès préconfigurées que vous pouvez personnaliser en fonction des besoins spécifiques de votre domaine. Vous pouvez également importer des politiques d'accès depuis d'autres domaines OpenSearch de service. Pour plus d'informations sur la manière dont ces stratégies d'accès interagissent avec l'accès VPC, consultez the section called “À propos des stratégies d'accès pour les domaines de VPC”.

Pour configurer les stratégies d'accès (console)

1. Connectez-vous à https://aws.amazon.com puis choisissez Sign In to the Console (Connectez-vous à la console).
2. Sous Analytics, sélectionnez Amazon OpenSearch Service.
3. Dans le panneau de navigation, sous Domains (Domaines), choisissez le domaine que vous souhaitez mettre à jour.
4. Choisissez Actions et Edit security configuration (Modifier la configuration de sécurité).
5. Modifiez la stratégie d'accès JSON ou importez une option préconfigurée.
6. Sélectionnez Enregistrer les modifications.

Paramètres avancés du cluster

Utilisez les options avancées pour configurer les éléments suivants :

Index dans les corps de requête

Spécifie si des références explicites aux index sont autorisées dans le corps des requêtes HTTP. Si vous affectez la valeur `false` à cette propriété, les utilisateurs ne peuvent pas outrepasser le contrôle d'accès aux sous-ressources. Par défaut, la valeur est `true`. Pour en savoir plus, consultez the section called “Options avancées et considérations relatives aux API”.

Allocation de cache de données de champ

Spécifie le pourcentage d'espace du tas Java alloué aux données de champ. Par défaut, ce paramètre correspond à 20 % de la pile de la JVM.

> **Note**
>
> De nombreux clients interrogent les index quotidiens en rotation.
> Nous vous recommandons de débuter des tests comparatifs avec `indices.fielddata.cache.size` configuré à 40 % de la pile de la JVM pour la

plupart de tels cas d'utilisation. Pour les index très volumineux, vous aurez peut-être besoin d'un grand cache de données de champ.

Nombre max. de clauses

Spécifie le nombre maximal de clauses autorisées dans une requête booléenne Lucene. La valeur par défaut est 1 024. Les requêtes comportant davantage de clauses que le nombre autorisé génèrent une erreur `TooManyClauses`. Pour plus d'informations, consultez la documentation Lucene.

Modifier la configuration dans Amazon OpenSearch Service

Amazon OpenSearch Service utilise un processus de déploiement bleu/vert lors de la mise à jour des domaines. Un blue/green déploiement crée un environnement inactif pour les mises à jour de domaine qui copie l'environnement de production et dirige les utilisateurs vers le nouvel environnement une fois ces mises à jour terminées. Dans un blue/green déploiement, l'environnement bleu est l'environnement de production actuel. L'environnement vert est un environnement inactif.

Les données sont migrées de l'environnement bleu vers l'environnement vert. Lorsque le nouvel environnement est prêt, le OpenSearch service change d'environnement pour promouvoir l'environnement vert en tant que nouvel environnement de production. Le basculement s'effectue sans perte de données. Cette pratique minimise les temps d'arrêt et préserve l'environnement d'origine en cas d'échec du déploiement dans le nouvel environnement.

Rubriques

- Changements qui entraînent généralement des blue/green déploiements
- Changements qui n'entraînent généralement pas de blue/green déploiements
- Déterminer si une modification entraînera un déploiement bleu/vert
- Suivi d'une modification de configuration
- Étapes d'un changement de configuration
- Impact sur les performances des déploiements bleu/vert
- Frais associés aux modifications de la configuration
- Résolution des erreurs de validation

Changements qui entraînent généralement des blue/green déploiements

Les opérations suivantes entraînent des blue/green déploiements :

- Modification du type d'instance
- Activation du contrôle précis des accès
- Mises à jour du logiciel de service
- Activation ou désactivation des nœuds principaux dédiés
- Activation ou désactivation du mode multi-AZ sans mode veille
- Modification du type de stockage, du type de volume ou de la taille du volume
- Sélection de sous-réseaux VPC différents
- Ajout ou suppression de groupes de sécurité VPC
- Ajouter ou supprimer des nœuds de coordination dédiés
- Activation ou désactivation de l'authentification Amazon Cognito pour les tableaux de bord OpenSearch
- Sélection d'un groupe d'utilisateurs ou d'un groupe d'identités Amazon Cognito différent
- Modification des paramètres avancés
- Mise à niveau vers une nouvelle OpenSearch version (OpenSearch les tableaux de bord peuvent ne pas être disponibles pendant une partie ou la totalité de la mise à niveau)
- Activation du chiffrement des données au repos ou du node-to-node chiffrement
- Activation ou désactivation du UltraWarm stockage à froid
- Désactivation d'Auto-Tune et annulation des modifications
- Associer un plugin optionnel à un domaine et dissocier un plugin optionnel d'un domaine
- Augmenter le nombre de nœuds maîtres dédiés pour les domaines multi-AZ avec deux nœuds maîtres dédiés
- Diminution de la taille du volume EBS
- Modification de la taille du volume, des IOPS ou du débit EBS, si la dernière modification que vous avez apportée est en cours ou s'est produite il y a moins de 6 heures
- Permettre la publication des journaux d'audit pour CloudWatch.

Pour les domaines Multi-AZ avec Standby, vous ne pouvez effectuer qu'une seule demande de modification à la fois. Si une modification est déjà en cours, la nouvelle

demande est rejetée. Vous pouvez vérifier l'état de la modification en cours à l'aide de l'`DescribeDomainChangeProgress`API.

Changements qui n'entraînent généralement pas de blue/green déploiements

Dans la plupart des cas, les opérations suivantes n'entraînent pas de blue/green déploiements :

- Modifier la politique d'accès
- Modification du point de terminaison personnalisé
- Modification de la politique TLS (Transport Layer Security)
- Modification de l'heure de début des instantanés automatiques
- Activation ou désactivation de l'option Exiger HTTPS
- Activation d'Auto-Tune ou désactivation sans annulation des modifications
- Si votre domaine possède des nœuds maîtres dédiés, modification du nœud de données ou du nombre de UltraWarm nœuds
- Si votre domaine possède des nœuds principaux dédiés, modification du type ou du nombre d'instances principales dédiées (sauf pour les domaines multi-AZ avec deux nœuds principaux dédiés)
- Activation ou désactivation de la publication de journaux d'erreurs ou de journaux lents sur CloudWatch
- Désactivation de la publication des journaux d'audit sur CloudWatch
- Augmentation de la taille du volume jusqu'à 3 TiB par nœud de données, modification du type de volume, des IOPS ou du débit
- Ajout ou suppression de balises

> **Note**
>
> Il existe quelques exceptions en fonction de la version de votre logiciel de service. Si vous voulez être sûr qu'une modification n'entraînera pas de blue/green déploiement, effectuez un essai avant de mettre à jour votre domaine, si cette option est disponible. Certaines modifications ne proposent pas d'option de fonctionnement à sec. Nous vous recommandons généralement d'apporter des modifications à votre cluster en dehors des heures de pointe.

Déterminer si une modification entraînera un déploiement bleu/vert

Vous pouvez tester certains types de modifications de configuration planifiées pour déterminer si elles entraîneront un blue/green déploiement, sans avoir à les valider. Avant de lancer une modification de la configuration, utilisez la console ou une API pour exécuter une vérification de validation afin de vous assurer que votre domaine est éligible à une mise à jour.

Console

Pour valider une modification de configuration

1. Accédez à la console Amazon OpenSearch Service à l'adressehttps://console.aws.amazon.com/aos/.
2. Dans le volet de navigation de gauche, choisissez Domains (Domaines).
3. Sélectionnez le domaine pour lequel que vous voluez effectuer une modification de configuration. Cette action ouvre la page des détails du domaine. Sélectionnez le menu déroulant Actions, puis choisissez Edit cluster configuration (Modifier la configuration du cluster).
4. Apportez des modifications au domaine, par exemple en modifiant le type d'instance ou le nombre de nœuds.
5. Sous Analyse du cycle à sec, sélectionnez Exécuter. L'essai à sec valide votre modification de configuration pour détecter les erreurs et détermine si elle nécessite un blue/green déploiement.
6. Lorsque le cycle de séchage est terminé, les résultats apparaissent au bas de la page, accompagnés d'un identifiant de séchage. L'analyse indique si le changement de configuration nécessite un blue/green déploiement.

   Chaque essai à sec remplace le précédent. Pour conserver les détails de chaque cycle, enregistrez son identifiant de cycle à sec. Les essais à sec sont disponibles pendant 90 jours ou jusqu'à ce que vous effectuiez une mise à jour de configuration.
7. Pour procéder à la mise à jour de votre configuration, choisissez Save changes (Enregistrer les modifications). Sinon, sélectionnez Annuler. Chaque option a pour effet de vous ramener à l'onglet Cluster configuration (Configuration du cluster). Dans cet onglet, vous pouvez sélectionner Dry run details (Détails du test à blanc) pour afficher les détails de votre dernier test à blanc. Cette page inclut également une side-by-side comparaison entre la configuration avant le cycle à sec et la configuration du cycle à sec.

API

Vous pouvez également effectuer une validation du test à blanc via l'API de configuration. Pour tester vos modifications avec l'API, définissez `DryRun` sur `true` et `DryRunMode` sur`Verbose`. Le mode Verbose exécute une vérification de validation en plus de déterminer si la modification déclenchera un déploiement bleu/vert. Par exemple, cette UpdateDomainConfigdemande teste le type de déploiement qui résulte de l'activation de UltraWarm :

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/
config
{
   "ClusterConfig": {
    "WarmCount": 3,
    "WarmEnabled": true,
    "WarmType": "ultrawarm1.large.search"
   },
   "DryRun": true,
   "DryRunMode": "Verbose"
}
```

La demande effectue une vérification de validation et renvoie le type de déploiement que la modification entraînera, mais n'effectue pas réellement la mise à jour :

```
{
   "ClusterConfig": {
     ...
    },
   "DryRunResults": {
      "DeploymentType": "Blue/Green",
      "Message": "This change will require a blue/green deployment."
    }
}
```

Les types de déploiement possibles sont les suivants :

- `Blue/Green` : la modification entraînera un déploiement bleu/vert.
- `DynamicUpdate` : la modification n'entraînera pas de déploiement bleu/vert.
- `Undetermined` : le domaine est toujours en traitement, de sorte que le type de déploiement ne peut pas être déterminé.
- `None` : aucune modification de configuration.

Si la validation échoue, elle renvoie une liste des échecs de validation.

```
{
   "ClusterConfig":{
      "..."
   },
   "DryRunProgressStatus":{
      "CreationDate":"2023-01-12T01:14:33.847Z",
      "DryRunId":"db00ca39-48b2-4774-bbd3-252cf094d205",
      "DryRunStatus":"failed",
      "UpdateDate":"2023-01-12T01:14:33.847Z",
      "ValidationFailures":[
         {
            "Code":"Cluster.Index.WriteBlock",
            "Message":"Cluster has index write blocks."
         }
      ]
   }
}
```

Si le statut est toujours le mêmepending, vous pouvez utiliser l'identifiant de course à sec dans votre UpdateDomainConfig réponse lors des DescribeDryRunProgressappels suivants pour vérifier l'état de la validation.

```
GET https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/
dryRun?dryRunId=my-dry-run-id
{
    "DryRunConfig": null,
    "DryRunProgressStatus": {
        "CreationDate": "2023-01-12T01:14:42.998Z",
        "DryRunId": "db00ca39-48b2-4774-bbd3-252cf094d205",
        "DryRunStatus": "succeeded",
        "UpdateDate": "2023-01-12T01:14:49.334Z",
        "ValidationFailures": null
    },
    "DryRunResults": {
        "DeploymentType": "Blue/Green",
        "Message": "This change will require a blue/green deployment."
    }
}
```

Pour exécuter une analyse de test à blanc sans vérification de validation, définissez `DryRunMode` sur`Basic` lorsque vous utilisez l'API de configuration.

Python

Le code Python suivant utilise l'UpdateDomainConfigAPI pour effectuer une vérification de validation à sec et, si la vérification aboutit, appelle la même API sans exécution à sec pour démarrer la mise à jour. Si la vérification échoue, le script affiche l'erreur et s'arrête.

```
import time
import boto3

client = boto3.client('opensearch')

response = client.UpdateDomainConfig(
    ClusterConfig={
        'WarmCount': 3,
        'WarmEnabled': True,
        'WarmCount': 123,
    },
    DomainName='test-domain',
    DryRun=True,
    DryRunMode='Verbose'
)

dry_run_id = response.DryRunProgressStatus.DryRunId

retry_count = 0

while True:

    if retry_count == 5:
        print('An error occured')
        break

    dry_run_progress_response = client.DescribeDryRunProgress('test-domain',
 dry_run_id)
    dry_run_status = dry_run_progress_response.DryRunProgressStatus.DryRunStatus

    if dry_run_status == 'succeeded':
        client.UpdateDomainConfig(
            ClusterConfig={
            'WarmCount': 3,
            'WarmEnabled': True,
```

```
            'WarmCount': 123,
        })
        break

    elif dry_run_status == 'failed':
        validation_failures_list =
dry_run_progress_response.DryRunProgressStatus.ValidationFailures
        for item in validation_failures_list:
            print(f"Code: {item['Code']}, Message: {item['Message']}")
        break

    retry_count += 1
    time.sleep(30)
```

Suivi d'une modification de configuration

Vous pouvez demander une modification de configuration à la fois ou regrouper plusieurs modifications dans une seule demande. Utilisez les champs État du traitement du domaine et État des modifications de configuration de la console pour suivre les modifications de configuration. Attendez que le statut du domaine change `Active` avant de demander des modifications supplémentaires.

Un domaine peut avoir les statuts de traitement suivants :

- `Active`— Aucune modification de configuration n'est en cours. Vous pouvez soumettre une nouvelle demande de modification de configuration.
- `Creating`— Le domaine est en cours de création.
- `Modifying`— Des modifications de configuration, telles que l'ajout de nouveaux nœuds de données, EBS, gp3, le provisionnement IOPS ou la configuration de clés KMS, sont en cours.
- `Upgrading engine version`— Une mise à niveau de la version du moteur est en cours.
- `Updating service software`— Une mise à jour du logiciel de service est en cours.
- `Deleting`— Le domaine est en cours de suppression.
- `Isolated`— Le domaine est suspendu.

Un domaine peut avoir les statuts de modification de configuration suivants :

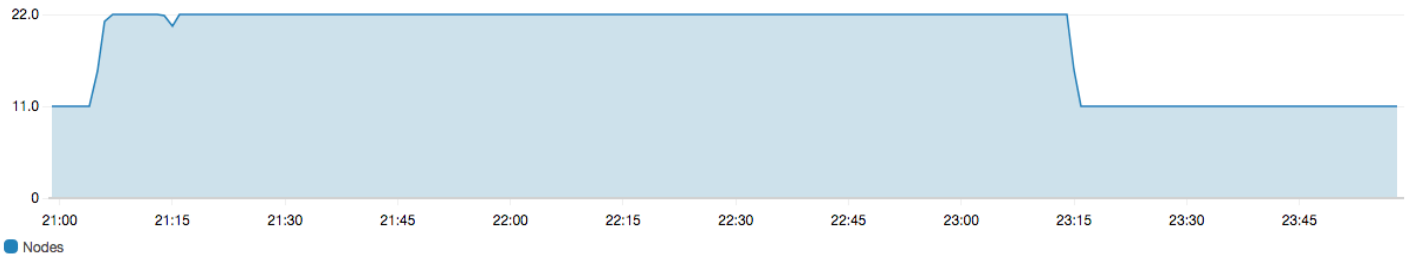
- `Pending`— Une demande de modification de configuration a été soumise.

- `Initializing`— Le service initialise une modification de configuration.
- `Validating`— Le service valide les modifications demandées et les ressources requises.
- `Awaiting user inputs`— Le service s'attend à ce que les modifications de configuration, telles qu'une modification du type d'instance, se poursuivent. Vous pouvez modifier les modifications de configuration.
- `Applying changes`— Le service applique les modifications de configuration demandées.
- `Cancelled`— La modification de configuration est annulée. Choisissez Annuler pour annuler toutes les modifications.
- `Completed`— Les modifications de configuration demandées ont été effectuées avec succès.
- `Validation failed`— Les modifications de configuration demandées n'ont pas pu être effectuées. Aucune modification de configuration n'a été appliquée.

> **Note**
>
> Les échecs de validation peuvent être le résultat d'index rouges présents dans votre domaine, de l'indisponibilité d'un type d'instance choisi ou d'un espace disque insuffisant. Pour obtenir la liste des erreurs de validation, consultezthe section called “Résolution des erreurs de validation”. Lors d'un échec de validation, vous pouvez annuler, réessayer ou modifier les modifications de configuration.

Lorsque les modifications de configuration sont terminées, le statut du domaine redevient`Active`.

Vous pouvez consulter l'état du cluster et CloudWatch les métriques Amazon et constater que le nombre de nœuds du cluster augmente temporairement, voire double, pendant la mise à jour du domaine. Dans l'illustration suivante, vous pouvez voir le nombre de nœuds passer de 11 à 22 pendant la modification de la configuration et revenir à 11 une fois la mise à jour terminée.



Cette augmentation temporaire peut faire peser davantage de contraintes sur les nœuds principaux dédiés du cluster, ce dernier ayant soudainement beaucoup plus de nœuds à gérer. Cela peut

également augmenter les latences de recherche et d'indexation lorsque le OpenSearch service copie les données de l'ancien cluster vers le nouveau. Il est important de maintenir une capacité suffisante sur le cluster pour gérer la surcharge associée à ces blue/green déploiements.

> **Important**
>
> Vous ne paierez aucuns frais supplémentaires pendant les changements de configuration et de maintenance. Des frais vous sont facturés uniquement pour le nombre de nœuds demandé pour votre cluster. Pour plus de détails, reportez-vous à la section the section called “Frais associés aux modifications de la configuration”.

Pour éviter de surcharger les nœuds principaux dédiés, vous pouvez surveiller l'utilisation à l'aide des CloudWatch métriques Amazon. Pour connaître les valeurs maximales recommandées, consultez the section called “ CloudWatch Alarmes recommandées”.

Étapes d'un changement de configuration

Une fois que vous avez initié une modification de configuration, le OpenSearch service effectue une série d'étapes pour mettre à jour votre domaine. Vous pouvez consulter la progression du changement de configuration sous État du changement de configuration dans la console. Les étapes exactes par lesquelles passe une mise à jour dépendent du type de changement que vous effectuez. Vous pouvez également surveiller un changement de configuration à l'aide de l'opération d'API DescribeDomainChangeProgress.

Voici les étapes possibles par lesquelles une mise à jour peut passer lors d'un changement de configuration :

Nom de l’environnement	Description
Validation	Validation de l'éligibilité du domaine à une mise à jour et mise en évidence des problèmes de

Nom de l’environnement	Description
	validation si nécessaire.
Création d'un nouvel environnement	Compléter les prérequis nécessaires et créer les ressources nécessaires pour démarrer le blue/green déploiement.
Approvisionnement de nouveaux nœuds	Création d'un nouvel ensemble d'instances dans le nouvel environnement.
Routage du trafic sur les nouveaux nœuds	Redirection du trafic vers les nœuds de données nouvellement créés.
Routage du trafic sur les anciens nœuds	Désactivation du trafic sur les anciens nœuds de données.

Nom de l’environnement	Description
Préparation des nœuds à la suppression	Préparation de la suppression des nœuds. Cette étape ne se produit que lorsque vous réduisez la taille de votre domaine (par exemple, de 8 nœuds à 6 nœuds).
Copie des partitions vers les nouveaux nœuds	Déplacement des partitions des anciens nœuds vers les nouveaux nœuds.
Résiliation des nœuds	Résiliation et suppression des anciens nœuds après la suppression des partitions.
Suppression des anciennes ressources	Suppression des ressources associées à l'ancien environnement (par exemple, l'équilibreur de charge).

Nom de l’environnement	Description
Mise à jour dynamique	Affiché lorsque la mise à jour ne nécessite pas de blue/green déploiement et peut être appliquée dynamiquement.
Appliquer les modifications liées au master dédié	Affiché lorsque le type ou le nombre d'instances principales dédiées est modifié.
Appliquer les modifications liées au volume	Affiché lorsque la taille, le type, les IOPS et le débit du volume sont modifiés.

Impact sur les performances des déploiements bleu/vert

Pendant blue/green le déploiement, votre cluster Amazon OpenSearch Service est disponible pour les demandes de recherche et d'indexation entrantes. Toutefois, vous pouvez rencontrer les problèmes de performances suivants :

- Augmentation temporaire de l'utilisation sur les nœuds principaux, car les clusters ont davantage de nœuds à gérer.

- Latence de recherche et d'indexation accrue, car le OpenSearch service copie les données des anciens nœuds vers les nouveaux nœuds.
- Augmentation du nombre de rejets de demandes entrantes à mesure que la charge du cluster augmente pendant les blue/green déploiements.
- Pour éviter les problèmes de latence et les rejets de demandes, vous devez exécuter blue/green les déploiements lorsque le cluster est sain et que le trafic réseau est faible.

Frais associés aux modifications de la configuration

Si vous modifiez la configuration d'un domaine, le OpenSearch service crée un nouveau cluster comme décrit dansthe section called “Configuration changes”. Lors de la migration de l'ancien cluster vers le nouveau, les frais suivants vous incombent :

- Si vous modifiez le type d'instance, les deux clusters vous sont facturés pendant la première heure. À l'issue de la première heure, seul le nouveau cluster vous est facturé. Les volumes EBS ne sont pas facturés deux fois car ils font partie de votre cluster. Leur facturation suit donc celle des instances.

  Exemple : Vous modifiez la configuration pour la faire passer de trois instances `m3.xlarge` à quatre instances `m4.large`. Pour la première heure, vous êtes facturé pour les deux clusters (3 * `m3.xlarge` + 4 * `m4.large`). Après la première heure, seul le nouveau cluster vous est facturé (4 * `m4.large`).
- Si vous ne modifiez pas le type d'instance, seul le plus grand cluster vous est facturé pour la première heure. À l'issue de la première heure, seul le nouveau cluster vous est facturé.

  Exemple : Vous modifiez la configuration pour la faire passer de six instances `m3.xlarge` à trois instances `m3.xlarge`. Pendant la première heure, le plus grand cluster vous est facturé (6 * `m3.xlarge`). Après la première heure, seul le nouveau cluster vous est facturé (3 * `m3.xlarge`).

Résolution des erreurs de validation

Lorsque vous initiez un changement de configuration ou effectuez une mise à niveau de version OpenSearch ou d'Elasticsearch, le OpenSearch Service effectue d'abord une série de contrôles de validation pour s'assurer que votre domaine est éligible à une mise à jour. Si l'un de ces contrôles échoue, vous recevez une notification dans la console contenant les problèmes spécifiques que vous devez résoudre avant de mettre à jour votre domaine. Le tableau suivant répertorie les problèmes de

domaine que le OpenSearch Service peut éventuellement rencontrer, ainsi que les étapes à suivre pour les résoudre.

Problème	Code d'erreur	Étapes de résolution des problèmes
Groupe de sécurité introuvable	`SecurityGroupNotFound`	Le groupe de sécurité associé à votre domaine OpenSearch de service n'existe pas. Pour résoudre ce problème, créez un groupe de sécurité avec le nom spécifié.
Sous-réseau introuvable	`SubnetNotFound`	Le sous-réseau associé à votre domaine OpenSearch de service n'existe pas. Pour résoudre ce problème, créer un sous-réseau dans votre VPC.
Rôle lié à un service non configuré	`SLRNotConfigured`	Le rôle lié au service pour OpenSearch Service n'est pas configuré. Le rôle lié au service est prédéfini par le OpenSearch service et inclut toutes les autorisations dont le service a besoin pour appeler d'autres AWS services en votre nom. Si le rôle n'existe pas, vous devrez peut-être le créer manuellement.
Pas assez d'adresses IP	`InsufficientFreeIPsForSubnets`	Un ou plusieurs de vos sous-réseaux VPC ne disposent pas de suffisamment d'adresses IP pour mettre à jour votre domaine. Pour calculer le nombre d'adresses IP dont vous avez besoin, consultez the section called “Réservation d'adresses IP dans un sous-réseau VPC”.
Le groupe d'utilisateurs Cognito n'existe pas	`CognitoUserPoolNotFound`	OpenSearch Le service ne trouve pas le groupe d'utilisateurs Amazon Cognito. Vérifiez que vous en avez créé un et qu'il a l'ID correct. Pour rechercher l'ID, vous pouvez utiliser la console Amazon Cognito ou la commande AWS CLI suivante : `aws cognito-idp list-user-pools --max-results 60 --region us-east-1`
Le groupe d'identités Cognito	`CognitoIdentityPoolNotFound`	OpenSearch Le service ne trouve pas le pool d'identités Cognito. Vérifiez que vous en avez créé un et qu'il a l'ID correct. Pour rechercher l'ID, vous pouvez utiliser la console Amazon Cognito ou la commande AWS CLI suivante :

Problème	Code d'erreur	Étapes de résolution des problèmes
n'existe pas		`aws cognito-identity list-identity-pools --max-results 60 --region us-east-1`
Domaine Cognito introuvable pour le groupe d'utilisateurs	`CognitoDomainNotFound`	Le groupe d'utilisateurs n'a pas de nom de domaine. Vous pouvez en configurer un à l'aide de la console Amazon Cognito ou de la commande suivante : AWS CLI `aws cognito-idp create-user-pool-domain --domain my-domain --user-pool-id id`
Le rôle Cognito n'est pas configuré	`CognitoRoleNotConfigured`	Le rôle IAM qui accorde au OpenSearch service l'autorisation de configurer les groupes d'utilisateurs et d'identités Amazon Cognito, et de les utiliser pour l'authentification, n'est pas configuré. Configurez le rôle avec un jeu d'autorisations et une relation d'approbation appropriés. Vous pouvez utiliser la console, qui crée le CognitoAccessForAmazonOpenSearchrôle par défaut pour vous, ou vous pouvez configurer manuellement un rôle à l'aide du AWS CLI ou du AWS SDK.
Impossible de décrire le groupe d'utilisateurs	`UserPoolNotDescribable`	Le rôle Amazon Cognito spécifié n'a pas l'autorisation de décrire le groupe d'utilisateurs associé à votre domaine. Assurez-vous que la politique d'autorisation du rôle autorise l'action `cognito-identity:DescribeUserPool`. Consultez the section called "À propos du rôle CognitoAccessForAmazonOpenSearch" pour connaître la politique d'autorisation complète.
Impossible de décrire le groupe d'identités	`IdentityPoolNotDescribable`	Le rôle Amazon Cognito spécifié n'a pas l'autorisation de décrire le groupe d'identités associé à votre domaine. Assurez-vous que la politique d'autorisation du rôle autorise l'action `cognito-identity:DescribeIdentityPool`. Consultez the section called "À propos du rôle CognitoAccessForAmazonOpenSearch" pour connaître la politique d'autorisation complète.

Problème	Code d'erreur	Étapes de résolution des problèmes
Impossible de décrire le groupe d'utilisateurs et d'identités	`CognitoPoolsNotDescribable`	Le rôle Amazon Cognito spécifié n'a pas l'autorisation de décrire les groupes d'utilisateurs et d'identités associés à votre domaine. Assurez-vous que la politique d'autorisation du rôle autorise les actions `cognito-identity:DescribeIdentityPool` et `cognito-identity:DescribeUserPool`. Consultez the section called "À propos du rôle CognitoAccessForAmazonOpenSearch" pour connaître la politique d'autorisation complète.
La clé KMS n'est pas activée	`KMSKeyNotEnabled`	La clé AWS Key Management Service (AWS KMS) utilisée pour chiffrer votre domaine est désactivée. Réactivez la clé immédiatement.
Le certificat personnalisé n'est pas dans l'état ISSUED (ÉMIS)	`InvalidCertificate`	Si votre domaine utilise un point de terminaison personnalisé, vous le sécurisez soit en générant un certificat SSL dans AWS Certificate Manager (ACM), soit en important le vôtre. L'état du certificat doit être Issued (Émis). Si vous recevez cette erreur, vérifiez le statut de votre certificat dans la console ACM. Si le statut est Expired (Expiré), Failed (Échec), Inactive (Inactif) ou Pending validation (En attente de validation), consultez la documentation de résolution des problèmes ACM pour résoudre le problème.
Pas assez de capacité pour lancer le type d'instance choisi	`InsufficientInstanceCapacity`	La capacité du type d'instance demandé n'est pas disponible. Par exemple, vous avez peut-être demandé cinq `i3.16xlarge.search` nœuds, mais le OpenSearch service ne dispose pas de suffisamment d'`i3.16xlarge.search` hôtes disponibles. La demande ne peut donc pas être traitée. Vérifiez les types d'instances pris en charge dans OpenSearch Service et choisissez un autre type d'instance.
Index rouges dans le cluster	`RedCluster`	Un ou plusieurs index de votre cluster ont un statut rouge, ce qui entraîne un statut rouge global du cluster. Pour résoudre ce problème et y remédier, consultez the section called "Statut de cluster rouge".

Problème	Code d'erreur	Étapes de résolution des problèmes
Disjoncteur de mémoire, trop de demandes	`TooManyRequests`	Votre domaine reçoit trop de requêtes de recherche et d'écriture. Le OpenSearch Service ne peut donc pas mettre à jour sa configuration. Vous pouvez réduire le nombre de demandes, mettre à l'échelle les instances verticalement jusqu'à 64 GiB de RAM, ou mettre à l'échelle horizontalement en ajoutant des instances.
La nouvelle configuration ne peut pas contenir de données (faible espace disque)	`InsufficientStorageCapacity`	La taille de stockage configurée ne peut pas contenir toutes les données de votre domaine. Pour résoudre ce problème, choisissez un volume plus grand, supprimez les index inutilisés ou augmentez le nombre de nœuds dans le cluster pour libérer immédiatement de l'espace disque.

Partitions épinglées à des nœuds spécifiques	`ShardMovementBlocked`	Un ou plusieurs index de votre domaine sont attachés à des nœuds spécifiques et ne peuvent pas être réaffectés. Cela s'est très probablement produit parce que vous avez configuré le filtrage d'allocation des partitions, qui vous permet de spécifier les nœuds autorisés à héberger les partitions d'un index particulier.
		
		Pour résoudre ce problème, supprimez les filtres d'allocation de partitions de tous les index concernés :
		
		`PUT my-index/_settings`
		`{`
		`  "settings": {`
		`  "index.routing.allocation.require._name": null`
		`  }`
		`}`

Problème	Code d'erreur	Étapes de résolution des problèmes
La nouvelle configuration ne peut pas contenir toutes les partitions (nombre de partitions)	`TooManyShards`	Le nombre de partitions de votre domaine est trop élevé, ce qui empêche le OpenSearch Service de les déplacer vers la nouvelle configuration. Pour résoudre ce problème, mettez à l'échelle horizontalement votre domaine en ajoutant des nœuds du même type de configuration que vos nœuds de cluster actuels. Notez que la taille maximale des volumes EBS dépend du type d'instance du nœud. Pour éviter ce problème à l'avenir, consultez the section called “Choix du nombre de partitions” et définissez une stratégie de partitionnement adaptée à votre cas d'utilisation.
Le sous-réseau associé à votre domaine ne prend pas en charge IPv4 les adresses	`ResultCodeIPv4BlockNotExists`	Pour résoudre ce problème, créez un sous-réseau ou mettez à jour le sous-réseau existant dans votre VPC en fonction du type d'adresse IP configuré pour le domaine. Si votre domaine utilise un type d'adresse IPv4uniquement, utilisez un IPv4 sous-réseau uniquement. Si votre domaine utilise le mode double pile, utilisez un sous-réseau à double pile.
Le sous-réseau associé à votre domaine ne prend pas en charge IPv6 les adresses	`ResultCodeIPv6BlockNotExists`	Pour résoudre ce problème, créez un sous-réseau ou mettez à jour le sous-réseau existant dans votre VPC en fonction du type d'adresse IP configuré pour le domaine. Si votre domaine utilise un type d'adresse IPv4uniquement, utilisez un IPv4 sous-réseau uniquement. Si votre domaine utilise le mode double pile, utilisez un sous-réseau à double pile.

Mises à jour du logiciel de service dans Amazon OpenSearch Service

> **Note**
>
> Pour obtenir des explications sur les modifications et les ajouts apportés dans chaque mise à jour du logiciel de service majeure (sans correctifs), veuillez consulter les notes de mise à jour.

Amazon OpenSearch Service publie régulièrement des mises à jour du logiciel de service qui ajoutent des fonctions ou améliorent vos domaines. Le panneau Notifications de la console constitue le moyen le plus simple de savoir si une mise à jour est disponible ou de vérifier le statut d'une mise à jour. Chaque notification inclut des détails sur la mise à jour du logiciel de service. Toutes les mises à jour du logiciel de service utilisent des déploiements bleu/vert pour limiter les temps d'arrêt.

Les mises à jour du logiciel de service diffèrent des mises à niveau de OpenSearch version. Pour plus d'informations sur la mise à niveau vers une version ultérieure de OpenSearch, consultezthe section called “Mise à niveau de domaines”.

Mises à jour facultatives ou obligatoires

OpenSearch Service propose deux grandes catégories de mises à jour du logiciel de service :

mises à jour facultatives

Les mises à jour facultatives du logiciel de service incluent généralement des améliorations et la prise en charge de nouvelles fonctionnalités. Les mises à jour facultatives ne sont pas appliquées à vos domaines, et il n'y a pas de date limite stricte pour les installer. La disponibilité de la mise à jour est communiquée par e-mail et par notification sur la console. Vous pouvez choisir d'appliquer la mise à jour immédiatement ou de la reprogrammer à une date et à une heure plus appropriées. Vous pouvez également le planifier pendant les heures creuses du domaine. La plupart des mises à jour logicielles sont facultatives.

Que vous planifiez ou non une mise à jour, si vous apportez une modification au domaine qui entraîne un déploiement bleu/vert, le OpenSearch Service met automatiquement à jour votre logiciel de service pour vous.

Vous pouvez configurer votre domaine pour appliquer automatiquement les mises à jour facultatives en dehors des heures de pointe. Lorsque cette option est activée, le OpenSearch service attend au moins 13 jours à compter de la date à laquelle une mise à jour facultative est disponible, puis planifie la mise à jour après 72 heures (trois jours). Vous recevez une notification de console lorsque la mise à jour est planifiée et vous pouvez choisir de la reprogrammer à une date ultérieure.

Pour activer les mises à jour logicielles automatiques, sélectionnez Activer les mises à jour logicielles automatiques lorsque vous créez ou mettez à jour votre domaine. Pour configurer le même paramètre à l'aide du AWS CLI, définissez `--software-update-options` sur `true` lorsque vous créez ou mettez à jour votre domaine.

Mises à jour requises

Les mises à jour du logiciel de service requises incluent généralement des correctifs de sécurité critiques ou d'autres mises à jour obligatoires pour garantir l'intégrité et les fonctionnalités continues de votre domaine. Parmi les mises à jour requises figurent Log4j Common Vulnerabilities and Exposures (CVEs) et l'application de la version 2 du service de métadonnées d'instance (). IMDSv2 Le nombre de mises à jour obligatoires par an est généralement inférieur à trois.

OpenSearch Le service planifie automatiquement ces mises à jour et vous avertit 72 heures (trois jours) avant la mise à jour planifiée par e-mail et par une notification sur console. Vous pouvez choisir d'appliquer la mise à jour immédiatement ou de la replanifier à une date et à une heure plus appropriées dans le délai imparti. Vous pouvez également le planifier lors de la prochaine période creuse du domaine. Si vous ne prenez aucune mesure concernant une mise à jour requise et que vous n'apportez aucune modification au domaine entraînant un déploiement bleu/vert, le OpenSearch Service peut lancer la mise à jour à tout moment au-delà de la date limite spécifiée (généralement 14 jours à compter de la date de disponibilité), pendant la période creuse du domaine.

Quelle que soit la date prévue de la mise à jour, si vous apportez une modification au domaine qui entraîne un déploiement bleu/vert, le OpenSearch Service met automatiquement à jour votre domaine pour vous.

mises à jour des correctifs

Les versions du logiciel de service qui se terminent par « -P » et un numéro, comme R20211203-*P4*, sont des versions de correctifs. Les correctifs sont susceptibles d'inclure des améliorations de performance, des corrections de bogues mineurs et des corrections de sécurité ou des améliorations de posture. Les versions de correctifs n'incluent pas de nouvelles fonctions ni de modifications

importantes, et elles n'ont généralement pas d'impact direct ou notable sur les utilisateurs. La notification du logiciel de service vous indique si la publication d'un correctif est facultative ou obligatoire.

Considérations

Si vous envisagez de mettre à jour votre domaine, prenez en compte les éléments suivants :

- La mise à jour manuelle de votre domaine vous permet de tirer parti des nouvelles fonctionnalités plus rapidement. Lorsque vous choisissez Update (Mettre à jour), OpenSearch Service place la demande dans une file d'attente et lance la mise à jour lorsqu'il en a le temps.
- Lorsque vous lancez une mise à jour du logiciel OpenSearch de service, Service envoie une notification au début et à la fin de celle-ci.
- Les mises à jour du logiciel utilisent des déploiements bleu/vert pour limiter les temps d'arrêt. Les mises à jour mettent temporairement à rude épreuve les nœuds maîtres dédiés d'un cluster. Aussi, veillez à conserver une capacité suffisante pour gérer la surcharge qui en découle.
- Les mises à jour se terminent généralement en quelques minutes, voire plusieurs heures ou si votre système est très sollicité. Envisagez de mettre à jour votre domaine pendant la période creuse configurée pour éviter de longues périodes de mise à jour.

Lancement d'une mise à jour du logiciel de service

Vous pouvez demander une mise à jour du logiciel de OpenSearch service via la console de service AWS CLI, le ou l'un des SDKs.

console

Pour demander une mise à jour du logiciel de service

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Sélectionnez le nom du domaine pour ouvrir sa configuration.
3. Choisissez Actions, Mettre à jour, puis sélectionnez l'une des options suivantes :

    - Appliquer la mise à jour maintenant : planifie immédiatement l'action pour qu'elle se produise à l'heure en cours si la capacité est disponible. Si la capacité n'est pas disponible, nous proposons d'autres créneaux horaires disponibles parmi lesquels choisir.

- Planifiez-le en période creuse : disponible uniquement si la fenêtre hors pointe est activée pour le domaine. Planifie la mise à jour pour qu'elle ait lieu pendant la période creuse configurée pour le domaine. Il n'y a aucune garantie que la mise à jour aura lieu au cours de la prochaine fenêtre immédiate. En fonction de la capacité, cela peut se produire dans les jours suivants. Pour de plus amples informations, veuillez consulter the section called “Fenêtres creuses”.
- Planifier pour une date et une heure spécifiques — Planifie la mise à jour pour qu'elle ait lieu à une date et à une heure spécifiques. Si l'heure que vous spécifiez n'est pas disponible pour des raisons de capacité, vous pouvez sélectionner un autre créneau horaire.

Si vous planifiez la mise à jour pour une date ultérieure (pendant ou en dehors de la période creuse du domaine), vous pouvez la reprogrammer à tout moment. Pour obtenir des instructions, veuillez consulter the section called “Planifier à nouveau les actions”.

4. Choisissez Confirmer.

AWS CLI

Envoyez une start-service-software-update AWS CLI demande pour lancer une mise à jour logicielle du service. Cet exemple ajoute immédiatement la mise à jour à la file d'attente :

```
aws opensearch start-service-software-update \
  --domain-name my-domain \
  --schedule-at "NOW"
```

Réponse :

```
{
    "ServiceSoftwareOptions": {
        "CurrentVersion": "R20220928-P1",
        "NewVersion": "R20220928-P2",
        "UpdateAvailable": true,
        "Cancellable": true,
        "UpdateStatus": "PENDING_UPDATE",
        "Description": "",
        "AutomatedUpdateDate": "1969-12-31T16:00:00-08:00",
        "OptionalDeployment": true
    }
}
```

> **Tip**
>
> Après avoir demandé une mise à jour, vous disposez d'un court délai pour l'annuler. La durée de cet `PENDING_UPDATE` état peut considérablement varier et dépend de votre Région AWS et du nombre de mises à jour simultanées que le OpenSearch Service effectue. Pour annuler une mise à jour, utilisez la console ou la `cancel-service-software-update` AWS CLI commande.

Si la demande échoue avec un`BaseException`, cela signifie que l'heure que vous avez spécifiée n'est pas disponible pour des raisons de capacité, et vous devez spécifier une autre heure. OpenSearch Le service fournit d'autres suggestions de créneaux disponibles dans la réponse.

AWS SDKs

Cet exemple de script Python utilise les méthodes describe_domain et start_service_software_update de AWS SDK pour Python (Boto3) pour vérifier si un domaine est éligible à une mise à jour du logiciel de service et, si c'est le cas, il démarre la mise à jour. Vous devez fournir une valeur pour `domain_name`.

```
import boto3
from botocore.config import Config
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

my_config = Config(
    # Optionally lets you specify a Region other than your default.
    region_name='us-east-1'
)

domain_name = ''  # The name of the domain to check and update

client = boto3.client('opensearch', config=my_config)


def getUpdateStatus(client):
    """Determines whether the domain is eligible for an update"""
    response = client.describe_domain(
```

```
        DomainName=domain_name
    )
    sso = response['DomainStatus']['ServiceSoftwareOptions']
    if sso['UpdateStatus'] == 'ELIGIBLE':
        print('Domain [' + domain_name + '] is eligible for a service software update
 from version ' +
              sso['CurrentVersion'] + ' to version ' + sso['NewVersion'])
        updateDomain(client)
    else:
        print('Domain is not eligible for an update at this time.')


def updateDomain(client):
    """Starts a service software update for the eligible domain"""
    response = client.start_service_software_update(
        DomainName=domain_name
    )
    print('Updating domain [' + domain_name + '] to version ' +
          response['ServiceSoftwareOptions']['NewVersion'] + '...')
    waitForUpdate(client)


def waitForUpdate(client):
    """Waits for the domain to finish updating"""
    response = client.describe_domain(
        DomainName=domain_name
    )
    status = response['DomainStatus']['ServiceSoftwareOptions']['UpdateStatus']
    if status == 'PENDING_UPDATE' or status == 'IN_PROGRESS':
        time.sleep(30)
        waitForUpdate(client)
    elif status == 'COMPLETED':
        print('Domain [' + domain_name +
              '] successfully updated to the latest software version')
    else:
        print('Domain is not currently being updated.')

def main():
    getUpdateStatus(client)
```

Planifier des mises à jour du logiciel pendant les périodes creuses

Chaque domaine OpenSearch de service créé après le 16 février 2023 dispose d'une fenêtre quotidienne de 10 heures entre 22 h 00 et 8 h 00, heure locale, que nous considérons comme une fenêtre creuse. OpenSearch Le service utilise cette fenêtre pour planifier les mises à jour du logiciel de service pour le domaine. Les mises à jour hors pointe permettent de minimiser la pression sur les nœuds principaux dédiés d'un cluster pendant les périodes de trafic élevé. OpenSearch Le service ne peut pas lancer de mises à jour en dehors de cette fenêtre de 10 heures sans votre consentement.

- Pour les mises à jour facultatives, le OpenSearch Service vous informe de la disponibilité de la mise à jour et vous invite à planifier la mise à jour lors d'une prochaine période creuse.
- Pour les mises à jour requises, le OpenSearch Service planifie automatiquement la mise à jour lors d'une prochaine période creuse et vous en informe trois jours à l'avance. Vous pouvez reprogrammer la mise à jour (pendant ou en dehors de la période creuse), mais uniquement dans le délai requis pour que la mise à jour soit terminée.

Pour chaque domaine, vous pouvez choisir de remplacer l'heure de début par défaut de 22 h 00 par une heure personnalisée. Pour obtenir des instructions, veuillez consulter the section called “Configuration d'une fenêtre personnalisée en dehors des heures de pointe”.

console

Pour planifier une mise à jour lors d'une prochaine période creuse

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Sélectionnez le nom du domaine pour ouvrir sa configuration.
3. Choisissez Actions, Mettre à jour.
4. Sélectionnez Planifier en dehors des heures de pointe.
5. Choisissez Confirmer.

Vous pouvez consulter l'action planifiée dans l'onglet Fenêtre hors pointe et la reprogrammer à tout moment. Consultez the section called “Affichage des actions planifiées”.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour planifier une mise à jour lors d'une prochaine période creuse à l'aide du AWS CLI, envoyez une StartServiceSoftwareUpdatedemande et spécifiez `OFF_PEAK_WINDOW` le `--schedule-at` paramètre :

```
aws opensearch start-service-software-update \
  --domain-name my-domain \
  --schedule-at "OFF_PEAK_WINDOW"
```

Mises à jour du logiciel de service

OpenSearch Le service envoie une notification lorsqu'une mise à jour du logiciel de service est disponible, requise, démarrée, terminée ou échouée. Vous pouvez afficher ces notifications sur le panneau de Notifications de la console de OpenSearch service. La sévérité de la notification est `Informational` si la mise à jour est facultative et `High` si elle est nécessaire.

OpenSearch Le service envoie également des événements liés au logiciel de service à Amazon EventBridge. Vous pouvez l'utiliser EventBridge pour configurer des règles qui envoient un e-mail ou effectuent une action spécifique lorsqu'un événement est reçu. Pour afficher un exemple de procédure, consultez the section called “Tutoriel : Envoi d'alertes SNS pour les mises à jour disponibles”.

Pour afficher le format de chaque événement du logiciel de service envoyé à Amazon EventBridge, veuillez consulter la rubriquethe section called “Événements de mise à jour du logiciel de service”.

Lorsque les domaines ne sont pas éligibles à une mise à jour

Votre domaine n'est pas éligible à une mise à jour du logiciel de service si son état correspond à l'un des états suivants :

État	Description
Domaine en cours de traitement	Un changement de configuration est actuellement en cours pour le domaine. Une fois l'opération terminée, vérifiez l'éligibilité pour la mise à jour.

État	Description
Statut de cluster rouge	Un ou plusieurs index dans le cluster sont rouges. Pour obtenir les étapes de dépannage, consultez the section called “Statut de cluster rouge”.
Taux d'erreur élevé	Le OpenSearch cluster renvoie un grand nombre d'erreurs 5 xx lors de sa tentative de traitement des demandes. Ce problème est généralem ent dû à un trop grand nombre de demandes simultanées de lecture ou d'écriture. Envisagez de réduire le trafic vers le cluster ou de redimensi onner votre domaine.
Split-Brain	Split-Brain signifie que votre OpenSearch cluster possède plusieurs nœuds principaux et s'est scindé en deux clusters qui ne refusionneront jamais d'eux-mêmes. Vous pouvez éviter le problème Split-Brain en utilisant le nombre recommandé de nœuds principaux dédiés. Pour obtenir de l'aide afin de résoudre le problème Split-Brain, contactez Support.
Problème d'intégra tion Amazon Cognito	Votre domaine utilise l'authentification pour les OpenSearch tableaux de bord et le OpenSearch service ne parvient pas à trouver une ou plusieurs ressources Amazon Cognito. Ce problème se produit généralement lorsque le groupe d'utilisateurs Amazon Cognito est manquant. Pour résoudre le problème, recréez la ressource manquante et configurez le domaine OpenSearch de service pour l'utiliser.
Autre problème de service	Les problèmes liés OpenSearch au service risquent d'afficher votre domaine comme étant inéligible à une mise à jour. Si aucune des conditions précédentes ne s'applique à votre domaine et que le problème persiste pendant plus d'une journée, contactez Support.

Définition des périodes creuses pour Amazon Service OpenSearch

Lorsque vous créez un domaine Amazon OpenSearch Service, vous définissez une fenêtre quotidienne de 10 heures considérée comme des heures creuses. OpenSearch Le service utilise cette fenêtre pour planifier les mises à jour du logiciel de service et les optimisations Auto-Tune qui nécessitent un déploiement bleu/vert pendant des périodes de trafic relativement faibles,

dans la mesure du possible. Le bleu/vert fait référence au processus consistant à créer un nouvel environnement pour les mises à jour de domaine et à acheminer les utilisateurs vers ce nouvel environnement après l'application des mises à jour.

Bien que les déploiements bleu/vert ne soient pas perturbateurs, afin de minimiser tout impact potentiel sur les performances lorsque les ressources sont consommées pour un déploiement bleu/vert, nous vous recommandons de planifier ces déploiements pendant la période creuse configurée pour le domaine. Les mises à jour, telles que le remplacement de nœuds ou celles qui doivent être déployées immédiatement sur le domaine, n'utilisent pas les périodes creuses.

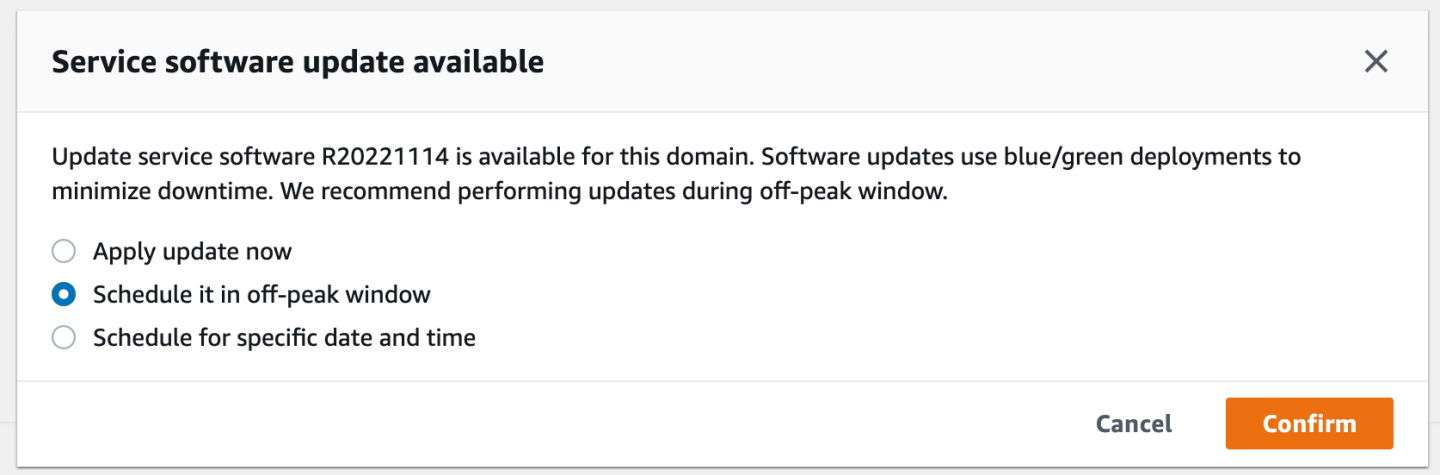
Vous pouvez modifier l'heure de début des heures creuses, mais vous ne pouvez pas modifier la durée de la fenêtre.

> **Note**
>
> Les fenêtres hors pointe ont été introduites le 16 février 2023. La fenêtre creuse est désactivée par défaut pour tous les domaines créés avant cette date. Vous devez activer et configurer manuellement la période creuse pour ces domaines. La fenêtre hors pointe sera activée par défaut pour tous les domaines créés après cette date. Vous ne pouvez pas désactiver la période creuse pour un domaine une fois qu'il est activé.

Mises à jour du logiciel de service

OpenSearch Le service comprend deux grandes catégories de mises à jour logicielles : facultatives et obligatoires. Les deux types nécessitent des déploiements bleu/vert. Les mises à jour facultatives ne sont pas appliquées à vos domaines, tandis que les mises à jour requises sont automatiquement installées si vous ne prenez aucune mesure avant la date limite spécifiée (généralement deux semaines après la date de disponibilité). Pour de plus amples informations, veuillez consulter the section called “Mises à jour facultatives ou obligatoires”.

Lorsque vous lancez une mise à jour facultative, vous avez le choix de l'appliquer immédiatement, de la planifier pour une prochaine période creuse ou de spécifier une date et une heure personnalisées pour l'appliquer.



Pour les mises à jour requises, le OpenSearch service planifie automatiquement une date et une heure pendant les heures creuses pour effectuer la mise à jour. Vous recevez une notification trois jours avant la mise à jour planifiée, et vous pouvez choisir de la reprogrammer à une date et une heure ultérieures pendant la période de déploiement requise. Pour obtenir des instructions, veuillez consulter the section called “Planifier à nouveau les actions”.

Optimisations Auto-Tune en dehors des heures de pointe

Auparavant, Auto-Tune utilisait des fenêtres de maintenance pour planifier les modifications nécessitant un déploiement bleu/vert. Les domaines pour lesquels Auto-Tune et les fenêtres de maintenance étaient déjà activées avant l'introduction des fenêtres creuses continueront à utiliser les fenêtres de maintenance pour ces mises à jour, sauf si vous les migrez pour utiliser les périodes creuses.

Nous vous recommandons de migrer vos domaines pour utiliser la période creuse, car elle est utilisée pour planifier d'autres activités sur le domaine, telles que les mises à jour du logiciel de service. Pour obtenir des instructions, veuillez consulter the section called “Migration depuis les fenêtres de maintenance Auto-Tune”. Vous ne pouvez pas recommencer à utiliser les fenêtres de maintenance après avoir migré votre domaine vers les périodes creuses.

Tous les domaines créés après le 16 février 2023 utiliseront la période creuse, plutôt que les fenêtres de maintenance traditionnelles, pour planifier les blue/green deployments. You can't disable the off-peak window for a domain. For a list of Auto-Tune optimizations that require blue/green déploiements, voir. the section called “Types de modifications”

Activation de la fenêtre hors pointe

La fonctionnalité est désactivée par défaut pour tous les domaines créés avant le 16 février 2023 (date d'introduction des périodes creuses). Vous devez l'activer manuellement pour ces domaines. Vous ne pouvez pas désactiver la fenêtre hors pointe une fois qu'elle est activée.

console

Pour activer la fenêtre creuse pour un domaine

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Sélectionnez le nom du domaine pour ouvrir sa configuration.
3. Accédez à l'onglet Fenêtre hors pointe et choisissez Modifier.
4. Spécifiez une heure de début personnalisée en heure UTC (temps universel coordonné). Par exemple, pour configurer une heure de début à 23 h 30 dans la région USA Ouest (Oregon), spécifiez 7 h 30.
5. Sélectionnez Enregistrer les modifications.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour modifier la fenêtre hors pointe à l'aide du AWS CLI, envoyez une UpdateDomainConfigdemande :

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --off-peak-window-options 'Enabled=true,
 OffPeakWindow={WindowStartTime={Hours=02,Minutes=00}}'
```

Si vous ne spécifiez pas d'heure de début de fenêtre personnalisée, la valeur par défaut est 00:00 UTC.

Configuration d'une fenêtre personnalisée en dehors des heures de pointe

Vous spécifiez une fenêtre de temps creux personnalisée pour votre domaine en heure universelle coordonnée (UTC). Par exemple, si vous souhaitez que la période creuse commence à 23 h 00 pour un domaine de la région USA Est (Virginie du Nord), vous devez spécifier 4 h 00 UTC.

console

Pour modifier la fenêtre creuse d'un domaine

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Sélectionnez le nom du domaine pour ouvrir sa configuration.
3. Accédez à l'onglet Fenêtre hors pointe. Vous pouvez consulter la fenêtre creuse configurée et la liste des actions planifiées à venir pour le domaine.
4. Choisissez Modifier et spécifiez une nouvelle heure de début en UTC. Par exemple, pour configurer l'heure de début à 21 h 00 dans la région USA Est (N. Virginia), spécifiez 02h00 UCT.
5. Sélectionnez Enregistrer les modifications.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour configurer une période creuse personnalisée à l'aide du AWS CLI, envoyez une UpdateDomainConfigdemande et spécifiez l'heure et les minutes au format 24 heures.

Par exemple, la requête suivante modifie l'heure de début de la fenêtre à 2 h 00 UTC :

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --off-peak-window-options 'OffPeakWindow={WindowStartTime={Hours=02,Minutes=00}}'
```

Si vous ne spécifiez pas d'heure de début de fenêtre, la valeur par défaut est 22 h 00, heure locale, pour Région AWS celle dans laquelle le domaine a été créé.

Affichage des actions planifiées

Vous pouvez consulter toutes les actions actuellement planifiées, en cours ou en attente pour chacun de vos domaines. Les actions peuvent avoir une gravité de `HIGH`MEDIUM, et`LOW`.

Les actions peuvent avoir les statuts suivants :

- `Pending update`— L'action se trouve dans la file d'attente à traiter.
- `In progress`— L'action est actuellement en cours.
- `Failed`— L'action a échoué.

- `Completed`— L'action s'est terminée avec succès.
- `Not eligible`— Uniquement pour les mises à jour du logiciel de service. La mise à jour ne peut pas être effectuée car le cluster est dans un état défectueux.
- `Eligible`— Uniquement pour les mises à jour du logiciel de service. Le domaine est éligible à une mise à jour.

console

La console OpenSearch de service affiche toutes les actions planifiées dans la configuration du domaine, ainsi que la gravité et l'état actuel de chaque action.

Pour afficher les actions planifiées pour un domaine

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Sélectionnez le nom du domaine pour ouvrir sa configuration.
3. Accédez à l'onglet Fenêtre hors pointe.
4. Sous Actions planifiées, consultez toutes les actions actuellement planifiées, en cours ou en attente pour le domaine.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour consulter les actions planifiées à l'aide du AWS CLI, envoyez une ListScheduledActionsdemande :

```
aws opensearch list-scheduled-actions \
  --domain-name my-domain
```

Réponse :

```
{
    "ScheduledActions": [
        {
            "Cancellable": true,
            "Description": "The Deployment type is : BLUE_GREEN.",
            "ID": "R20220721-P13",
            "Mandatory": false,
```

```
            "Severity": "HIGH",
            "ScheduledBy": "CUSTOMER",
            "ScheduledTime": 1.673871601E9,
            "Status": "PENDING_UPDATE",
            "Type": "SERVICE_SOFTWARE_UPDATE",
        },
        {
            "Cancellable": true,
            "Description": "Amazon Opensearch will adjust the young generation JVM
 arguments on your domain to improve performance",
            "ID": "Auto-Tune",
            "Mandatory": true,
            "Severity": "MEDIUM",
            "ScheduledBy": "SYSTEM",
            "ScheduledTime": 1.673871601E9,
            "Status": "PENDING_UPDATE",
            "Type": "JVM_HEAP_SIZE_TUNING",
        }
    ]
}
```


Planifier à nouveau les actions

OpenSearch Le service vous informe des mises à jour programmées du logiciel de service et des optimisations d'Auto-Tune. Vous pouvez choisir d'appliquer la modification immédiatement ou de la replanifier pour une date et une heure ultérieures.

> **Note**
>
> OpenSearch Le service peut planifier l'action dans l'heure qui suit l'heure que vous avez sélectionnée. Par exemple, si vous choisissez d'appliquer une mise à jour à 17 h, elle peut être appliquée entre 17 h et 18 h.

console

Pour replanifier une action

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Sélectionnez le nom du domaine pour ouvrir sa configuration.

3. Accédez à l'onglet Fenêtre hors pointe.
4. Sous Actions planifiées, sélectionnez l'action et choisissez Replanifier.
5. Choisissez l'une des options suivantes :

   - Appliquer la mise à jour maintenant : planifie immédiatement l'action pour qu'elle se produise à l'heure en cours si la capacité est disponible. Si la capacité n'est pas disponible, nous proposons d'autres plages horaires disponibles parmi lesquelles choisir.
   - Programmez-la en période creuse : marque l'action à reprendre lors d'une prochaine fenêtre hors pointe. Il n'y a aucune garantie que le changement sera mis en œuvre au cours de la fenêtre suivante. En fonction de la capacité, cela peut se produire dans les jours suivants.
   - Replanifier cette mise à jour : permet de spécifier une date et une heure personnalisées pour appliquer la modification. Si l'heure que vous spécifiez n'est pas disponible pour des raisons de capacité, vous pouvez sélectionner un autre créneau horaire.
   - Annuler la mise à jour planifiée : annule la mise à jour. Cette option est disponible uniquement pour les mises à jour du logiciel de service optionnelles. Il n'est pas disponible pour les actions Auto-Tune ou les mises à jour logicielles obligatoires.
6. Sélectionnez Enregistrer les modifications.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour replanifier une action à l'aide du AWS CLI, envoyez une UpdateScheduledActiondemande. Pour récupérer l'identifiant de l'action, envoyez une `ListScheduledActions` demande.

La demande suivante replanifie une mise à jour du logiciel de service pour une date et une heure spécifiques :

```
aws opensearch update-scheduled-action \
  --domain-name my-domain \
  --action-id R20220721-P13 \
  --action-type "SERVICE_SOFTWARE_UPDATE" \
  --desired-start-time 1677348395000 \
  --schedule-at TIMESTAMP
```

Réponse :

```
{
   "ScheduledAction": {
```

```
      "Cancellable": true,
      "Description": "Cluster status is updated.",
      "Id": "R20220721-P13",
      "Mandatory": false,
      "ScheduledBy": "CUSTOMER",
      "ScheduledTime": 1677348395000,
      "Severity": "HIGH",
      "Status": "PENDING_UPDATE",
      "Type": "SERVICE_SOFTWARE_UPDATE"
   }
}
```

Si la demande échoue avec un`SlotNotAvailableException`, cela signifie que l'heure que vous avez spécifiée n'est pas disponible pour des raisons de capacité, et vous devez spécifier une autre heure. OpenSearch Le service fournit d'autres suggestions de créneaux disponibles dans la réponse.

Migration depuis les fenêtres de maintenance Auto-Tune

Si un domaine a été créé avant le 16 février 2023, il peut utiliser les fenêtres de maintenance pour planifier les optimisations Auto-Tune nécessitant un déploiement bleu/vert. Vous pouvez migrer vos domaines Auto-Tune existants pour utiliser plutôt les périodes creuses.

> **Note**
>
> Vous ne pouvez pas revenir à l'utilisation des fenêtres de maintenance après avoir migré votre domaine pour utiliser les périodes creuses.

console

Pour migrer un domaine afin d'utiliser les heures creuses

1. Dans la console Amazon OpenSearch Service, sélectionnez le nom du domaine pour ouvrir sa configuration.
2. Accédez à l'onglet Auto-Tune et choisissez Modifier.
3. Sélectionnez Migrer vers les heures creuses.
4. Pour l'heure de début (UTC), indiquez une heure de début quotidienne pour la période creuse en temps universel coordonné (UTC).
5. Sélectionnez Enregistrer les modifications.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour passer d'une fenêtre de maintenance Auto-Tune à une période creuse à l'aide du AWS CLI, envoyez une UpdateDomainConfigdemande :

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --auto-tune-options
 DesiredState=ENABLED,UseOffPeakWindow=true,MaintenanceSchedules=[]
```

La période creuse doit être activée pour que vous puissiez migrer un domaine de la fenêtre de maintenance d'Auto-Tune vers la période creuse. Vous pouvez activer la période creuse dans une demande séparée ou dans la même demande. Pour obtenir des instructions, veuillez consulter the section called “Activation de la fenêtre hors pointe”.

Notifications dans Amazon OpenSearch Service

Les notifications d'Amazon OpenSearch Service contiennent des informations importantes sur les performances et l'état de santé de vos domaines. OpenSearch Le service vous informe des mises à jour du logiciel de service, des améliorations apportées à Auto-Tune, des événements liés à l'état du cluster et des erreurs de domaine. Les notifications sont disponibles pour toutes les versions d' OpenSearch Elasticsearch OSS.

Vous pouvez consulter les notifications dans le panneau Notifications de la console de OpenSearch service. Toutes les notifications relatives OpenSearch au service sont également affichées sur Amazon EventBridge. Pour afficher la liste complète des exemples de notifications et d'événements, veuillez consulter la rubrique the section called “Surveillance des événements”.

Prise en main des notifications

Les notifications sont automatiquement activées lorsque vous créez un domaine. Accédez au panneau Notifications de la console de OpenSearch service pour surveiller les notifications et en accuser réception. Chaque notification comprend des informations telles que l'heure à laquelle elle a été publiée, le domaine auquel elle se rapporte, un niveau de gravité et d'état, ainsi qu'une brève explication. Vous pouvez consulter l'historique des notifications pendant 90 jours dans la console.

Après avoir accédé au volet Notifications ou avoir accusé réception d'une notification, un message d'erreur peut s'afficher pour indiquer que vous n'avez pas les autorisations nécessaires pour effectuer

l'opération `es:ListNotificationsV2` ou `es:UpdateNotificationStatus`. Pour résoudre ce problème, octroyez à votre utilisateur ou à votre rôle les autorisations suivantes dans IAM :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "es:UpdateNotificationStatus",
      "es:ListNotifications"
    ],
    "Resource": "arn:aws:es:*:123456789012:domain/*"
  }]
}
```

La console IAM génère une erreur (« IAM ne reconnaît pas une ou plusieurs actions ») que vous pouvez ignorer en toute sécurité. Vous pouvez également restreindre l'action `es:UpdateNotificationStatus` à certains domaines. Pour en savoir plus, veuillez consulter la section the section called “Références des éléments de stratégie”.

Niveaux de gravité des notifications

Les notifications dans le cadre du OpenSearch Service peuvent être informatives, relatives à toute action que vous avez déjà entreprise ou aux opérations de votre domaine, ou actionnables, qui nécessitent que vous preniez des mesures spécifiques telles que l'application d'un correctif de sécurité obligatoire. Un niveau de gravité est associé à chaque notification qui peut être `Informational`, `Low`, `Medium`, `High` ou `Critical`. Le tableau suivant récapitule chaque niveau de sévérité :

Sévérité	Description	Exemples
`Informational`	Informations relatives au fonctionnement de votre domaine.	• Mise à jour du logiciel de service disponible • Auto-Tune démarré

Sévérité	Description	Exemples
`Low`	Action recommandée, mais aucun impact négatif n'est à déplorer sur la disponibilité ou les performances du domaine si aucune action n'est entreprise.	• Auto-Tune annulé • Avertissement sur le nombre élevé de partitions
`Medium`	Impact possible si l'action recommandée n'est pas entreprise, mais le délai est prolongé pour encourager sa mise en œuvre.	• Échec de la mise à jour du logiciel de service • Limite du nombre de partitions dépassée
`High`	Une action urgente est nécessaire pour éviter tout impact négatif.	• Mise à jour du logiciel de service requise • Clé KMS inaccessible
`Critical`	Une action immédiate est nécessaire pour éviter tout impact négatif, ou pour s'en relever.	Aucune disponible actuellement.

Exemple d' EventBridge événement

L'exemple suivant montre un événement OpenSearch de notification de service envoyé à Amazon EventBridge. Le niveau de gravité de la notification correspondante est de `Informational`, car la mise à jour est facultative :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
```

```
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Available",
    "severity": "Informational",
    "description": "Service software update [R20200330-p1] available."
  }
}
```

Configuration d'un domaine multi-AZ dans Amazon Service OpenSearch

Pour éviter les pertes de données et minimiser les interruptions de OpenSearch service du cluster Amazon Service en cas d'interruption de service, vous pouvez répartir les nœuds sur deux ou trois zones de disponibilité de la même région, selon une configuration connue sous le nom de Multi-AZ. Les zones de disponibilité sont des emplacements isolés au sein de chaque AWS région.

Pour les domaines qui exécutent des charges de travail de production, nous recommandons l'option de déploiement Multi-AZ avec veille, qui crée la configuration suivante :

- Le domaine est déployé sur trois zones.
- Types d'instances de génération actuelle pour les nœuds maîtres et les nœuds de données dédiés.
- Trois nœuds maîtres dédiés et trois nœuds de données (ou un multiple de trois).
- Au moins deux répliques pour chaque index de votre domaine, ou un multiple de trois copies de données (y compris les nœuds principaux et les répliques).

Le reste de cette section fournit des explications et le contexte de ces configurations.

Multi-AZ avec mode veille

Multi-AZ with Standby est une option de déploiement pour les domaines Amazon OpenSearch Service qui offre une disponibilité de 99,99 %, des performances constantes pour les charges de travail de production, ainsi qu'une configuration et une gestion de domaines simplifiées. Lorsque vous utilisez le mode Multi-AZ en mode veille, les domaines résistent aux défaillances de l'infrastructure, sans aucun impact sur les performances ou la disponibilité. Cette option de déploiement répond à cette norme en imposant un certain nombre de bonnes pratiques, telles que le nombre de nœuds

de données spécifié, le nombre de nœuds principaux, le type d'instance, le nombre de répliques, les paramètres de mise à jour logicielle et l'activation du réglage automatique.

> **Note**
>
> Multi-AZ with Standby est disponible pour les OpenSearch versions 1.3 et supérieures et nécessite des régions comportant au moins trois zones de disponibilité.

Lorsque vous utilisez Multi-AZ avec Standby, le OpenSearch service crée un domaine dans trois zones de disponibilité, chaque zone contenant une copie complète des données et les données étant réparties de manière égale dans chacune des zones. Votre domaine réserve des nœuds dans l'une de ces zones en attente, ce qui signifie qu'ils ne répondent pas aux demandes de recherche. Lorsque le OpenSearch Service détecte une défaillance dans l'infrastructure sous-jacente, il active automatiquement les nœuds de secours en moins d'une minute. Le domaine continue de traiter les demandes d'indexation et de recherche, et tout impact est limité au temps nécessaire pour effectuer le basculement. Il n'y a aucune redistribution des données ou des ressources, ce qui n'affecte pas les performances du cluster et aucun risque de dégradation de la disponibilité. Le mode Multi-AZ avec mode veille est disponible sans frais supplémentaires.

Deux options s'offrent à vous pour créer un domaine en mode veille activé sur AWS Management Console. Tout d'abord, vous pouvez créer un domaine à l'aide de la méthode de création Easy create, et le OpenSearch service utilisera automatiquement une configuration prédéterminée, qui inclut les éléments suivants :

- Trois zones de disponibilité, dont l'une fait office de veille
- Trois nœuds principaux et nœuds de données dédiés
- Auto-Tune activé sur le domaine
- GP3 stockage pour les nœuds de données

Vous pouvez également choisir la méthode de création standard et sélectionner le domaine avec veille comme option de déploiement. Cela vous permet de personnaliser votre domaine tout en imposant les principales fonctionnalités de veille, telles que trois zones et trois nœuds principaux. Nous vous recommandons de choisir un nombre de nœuds de données multiple de trois (le nombre de zones de disponibilité).

Une fois que vous avez créé votre domaine, vous pouvez accéder aux pages de détails du domaine et, dans l'onglet Configuration du cluster, vérifier que 3-AZ avec veille apparaît sous Zone (s) de disponibilité.

Si vous rencontrez des problèmes lors de la migration d'un domaine existant vers le mode Multi-AZ avec veille, consultez la section Erreur lors de la migration vers le mode Multi-AZ avec mode veille dans le guide de dépannage.

Limites

Lorsque vous configurez un domaine avec Multi-AZ avec mode veille, tenez compte des limites suivantes :

- Le nombre total de partitions sur un nœud ne peut pas dépasser 1 000, le nombre total de partitions sur un cluster ne peut pas dépasser 75 000 et la taille d'une seule partition ne peut pas dépasser 65 Go.
- Multi-AZ with Standby ne fonctionne qu'avec les types d'`i3`instance `m5` `c5` `r5` `r6gr7g`,`c6g`,`m6g`,,, `r6gd` et. Pour plus d'informations sur les instances prises en charge, consultez la section Types d'instances pris en charge.
- Vous ne pouvez utiliser qu'un IOPs SSD provisionné, un SSD à usage général (GP3) ou un stockage sauvegardé par instance en mode veille.
- Si vous l'activez UltraWarmsur un domaine Multi-AZ avec veille, le nombre de nœuds chauds doit être un multiple du nombre de zones de disponibilité utilisées.

Multi-AZ sans mode veille

OpenSearch Le service prend toujours en charge le mode multi-AZ sans mode veille, ce qui offre une disponibilité de 99,9 %. Les nœuds sont répartis entre les zones de disponibilité, et la disponibilité dépend du nombre de zones de disponibilité et de copies des données. Alors qu'avec le mode veille, vous devez configurer votre domaine selon les meilleures pratiques, sans mode veille, vous pouvez choisir votre propre nombre de zones de disponibilité, de nœuds et de répliques. Nous ne recommandons pas cette option, sauf si vous avez des flux de travail existants qui seraient perturbés par la création de domaines en veille.

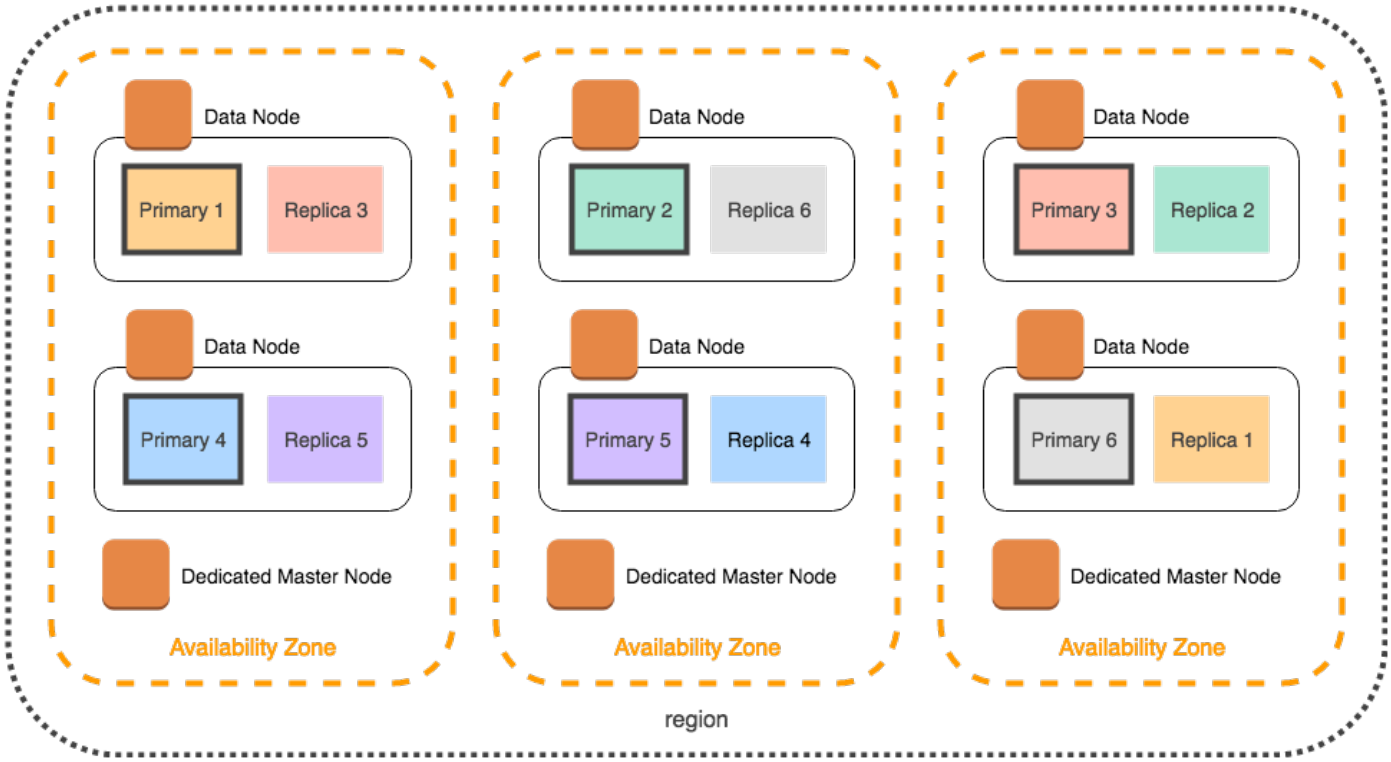
Si vous choisissez cette option, nous vous recommandons tout de même de sélectionner trois zones de disponibilité afin de rester résilient aux défaillances des nœuds, des disques et des défaillances mono-AZ. En cas de panne, le cluster redistribue les données entre les ressources restantes afin de maintenir la disponibilité et la redondance. Ce mouvement de données augmente l'utilisation

des ressources sur le cluster et peut avoir un impact sur les performances. Si le cluster n'est pas correctement dimensionné, sa disponibilité peut se dégrader, ce qui va largement à l'encontre de l'objectif du Multi-AZ.

La seule façon de configurer un domaine sans veille AWS Management Console est de choisir la méthode de création standard et de sélectionner Domaine sans veille comme option de déploiement.

Répartition des partitions

Si vous activez le mode multi-AZ sans mode veille, vous devez créer au moins une réplique pour chaque index de votre cluster. Sans répliques, le OpenSearch Service ne peut pas distribuer de copies de vos données vers d'autres zones de disponibilité. Heureusement, la configuration par défaut pour n'importe quel index est un nombre de réplica de 1. Comme le montre le schéma suivant, OpenSearch Service fait de son mieux pour distribuer les partitions principales et leurs répliques correspondantes dans différentes zones.

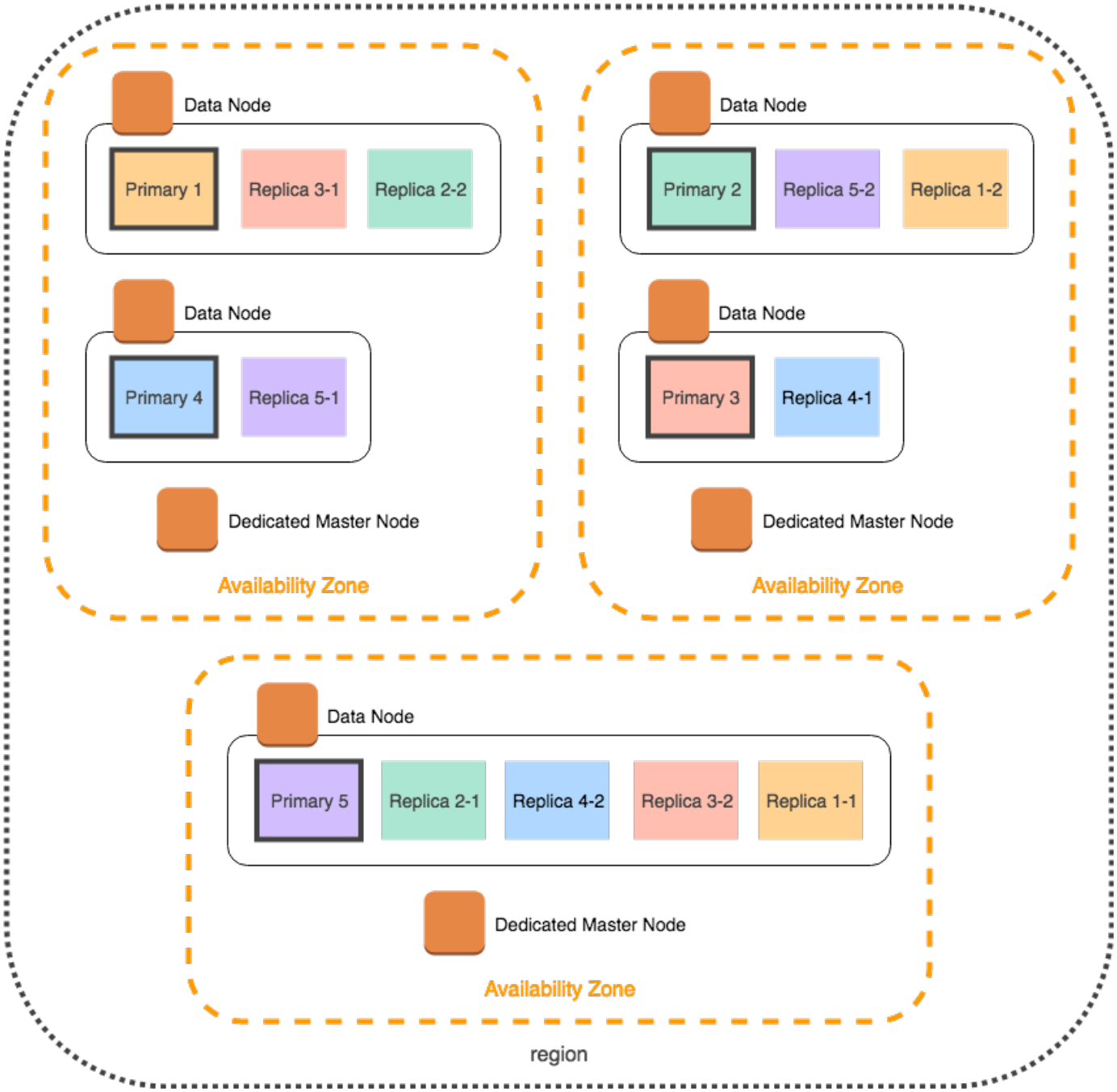


Outre la distribution des partitions par zone de disponibilité, OpenSearch Service les distribue par nœud. Toutefois, certaines configurations de domaine peuvent créer des nombres de partitions déséquilibrés. Prenons l'exemple de domaine suivant :

- 5 nœuds de données

- 5 partitions principales
- 2 réplicas
- 3 zones de disponibilité

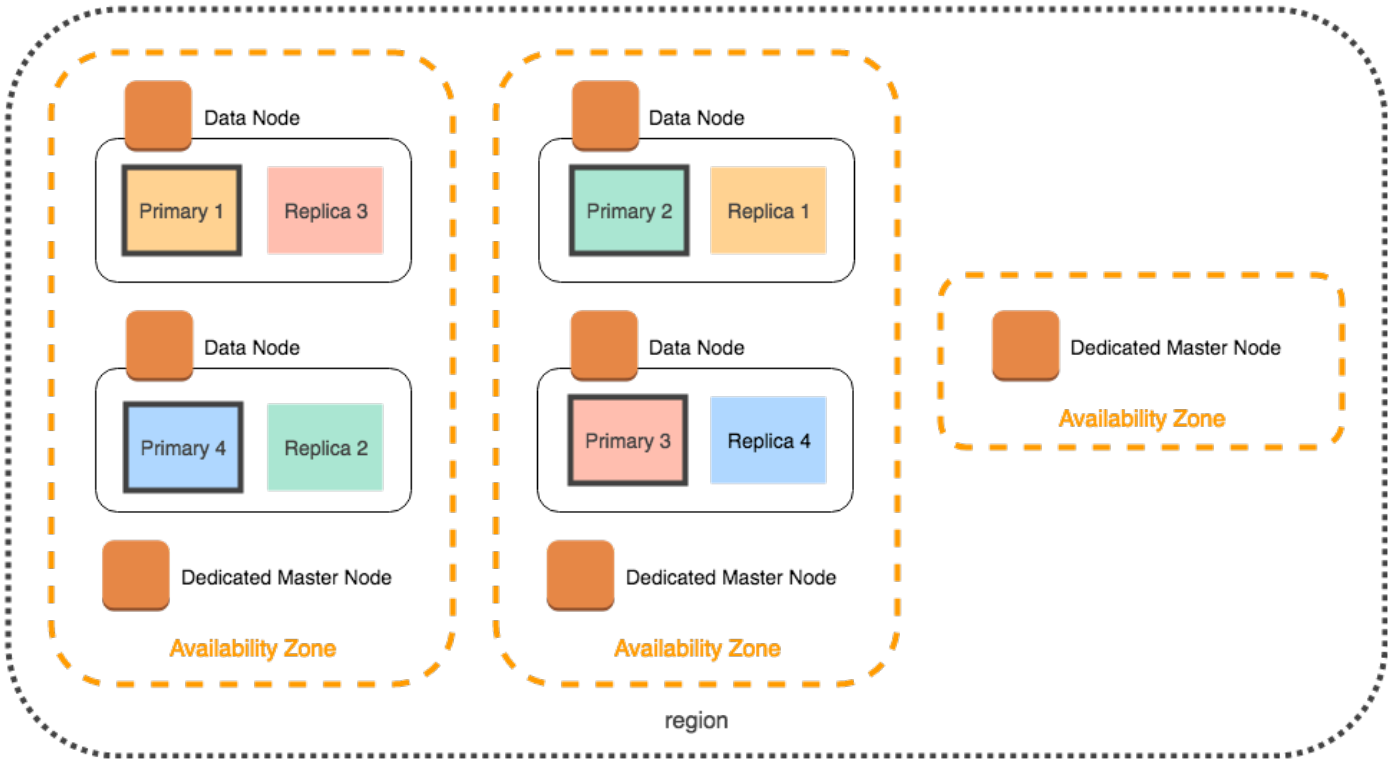
Dans ce cas, le OpenSearch service doit surcharger un nœud afin de distribuer les partitions principales et répliques entre les zones, comme indiqué dans le schéma suivant.



Pour éviter ce type de situation, qui peut mettre à rude épreuve des nœuds individuels et nuire aux performances, nous vous recommandons de choisir le mode Multi-AZ avec mode veille, ou de choisir un nombre d'instances multiple de trois lorsque vous prévoyez d'avoir deux répliques ou plus par index.

Répartition des nœuds principaux dédiés

Même si vous sélectionnez deux zones de disponibilité lors de la configuration de votre domaine, le OpenSearch service distribue automatiquement des nœuds maîtres dédiés sur trois zones de disponibilité. Cette distribution permet d'éviter les temps d'arrêt du cluster si une zone subit une interruption de service. Si vous utilisez les trois nœuds principaux dédiés recommandés et qu'une zone de disponibilité tombe en panne, votre cluster dispose encore d'un quorum (2) des nœuds principaux dédiés et peut choisir un nouveau maître. Le schéma suivant illustre cette configuration.



Si vous choisissez un type d'instance d'une génération plus ancienne qui n'est pas disponible dans trois zones de disponibilité, les scénarios suivants s'appliquent :

- Si vous avez choisi trois zones de disponibilité pour le domaine, le OpenSearch service génère une erreur. Choisissez un type d'instance différent, puis réessayez.

- Si vous avez choisi deux zones de disponibilité pour le domaine, le OpenSearch service distribue les nœuds maîtres dédiés sur deux zones.

Interruptions des zones de disponibilité

Ces interruptions sont rares, mais peuvent arriver. Le tableau suivant répertorie les différentes configurations et comportements Multi-AZ lors d'une interruption. La dernière ligne du tableau s'applique au mode multi-AZ avec mode veille, tandis que toutes les autres lignes ont des configurations qui ne s'appliquent qu'au mode multi-AZ sans mode veille.

Nombre de zones de disponibilité dans une région	Nombre de zones de disponibilité que vous avez choisies	Nombre de nœuds principaux dédiés	Comportement si une zone de disponibilité subit une interruption
2 ou plus	2	0	Temps d'arrêt. Votre cluster perd la moitié de ses nœuds de données et doit en remplacer au moins un dans les autres zones de disponibilité avant de pouvoir choisir un maître.
2	2	3	50 % de chances d'indisponibilité. OpenSearch Le service distribue deux nœuds maîtres dédiés dans une zone de disponibilité et l'un dans l'autre : • Si la zone de disponibilité à un nœud principal dédié subit une interruption, les deux nœuds principaux dédiés dans la zone restante peuvent choisir un maître. • Si la zone de disponibilité à deux nœuds principaux dédiés subit une interruption, le cluster est indisponible jusqu'à ce que la zone restante soit rétablie.
3 ou plus	2	3	Aucun temps d'arrêt. OpenSearch Le service distribue automatiquement les nœuds maîtres

Nombre de zones de disponibilité dans une région	Nombre de zones de disponibilité que vous avez choisies	Nombre de nœuds principaux dédiés	Comportement si une zone de disponibilité subit une interruption
			dédiés sur trois zones de disponibilité, de sorte que les deux nœuds principaux dédiés restants peuvent élire un maître.
3 ou plus	3	0	Pas de temps d'arrêt. Environ deux tiers de vos nœuds de données sont toujours disponibles pour choisir un maître.
3 ou plus	3	3	Pas de temps d'arrêt. Les deux autres nœuds principaux dédiés peuvent choisir un maître.

Dans toutes les configurations, quelle qu'en soit la cause, les défaillances de nœuds peuvent entraîner une période de charge accrue sur les nœuds de données restants du cluster, tandis que le OpenSearch service configure automatiquement de nouveaux nœuds pour remplacer les nœuds manquants.

Par exemple, en cas de défaillance d'une zone de disponibilité dans une configuration à trois zones, les deux tiers des nœuds de données doivent traiter le même nombre de requêtes dans le cluster. À mesure qu'ils traitent ces requêtes, les nœuds restants répliquent également des partitions sur de nouveaux nœuds au fur et à mesure qu'ils apparaissent en ligne, ce qui peut affecter d'autant plus les performances. Si la disponibilité est essentielle pour votre charge de travail, nous vous conseillons d'ajouter des ressources à votre cluster pour atténuer ce soucis.

> **Note**
>
> OpenSearch Le service gère les domaines multi-AZ de manière transparente, de sorte que vous ne pouvez pas simuler manuellement les perturbations des zones de disponibilité.

Lancement de vos domaines Amazon OpenSearch Service au sein d'un VPC

Vous pouvez lancer AWS des ressources, telles que des domaines Amazon OpenSearch Service, dans un cloud privé virtuel (VPC). Un VPC est un réseau virtuel qui vous est dédié. Compte AWS Il est logiquement isolé des autres réseaux virtuels du AWS cloud. Le placement d'un domaine de OpenSearch service au sein d'un VPC permet une communication sécurisée entre le OpenSearch service et les autres services du VPC sans avoir besoin d'une passerelle Internet, d'un périphérique NAT ou d'une connexion VPN. Tout le trafic reste sécurisé dans le AWS cloud.

> **Note**
>
> Si vous placez votre domaine de OpenSearch service dans un VPC, votre ordinateur doit pouvoir se connecter au VPC. Cette connexion s'effectue souvent via un réseau VPN, une passerelle de transit, un réseau géré ou un serveur proxy. Vous ne pouvez pas accéder directement à vos domaines depuis l'extérieur du VPC.

VPC contre domaines publics

Voici quelques différences entre les domaines VPC et les domaines publics. Chaque différence est décrite de façon plus détaillée par la suite.

- En raison de leur isolement logique, les domaines résidant au sein d'un VPC possèdent une couche de sécurité supplémentaire par rapport aux domaines qui utilisent des points de terminaison publics.
- Bien que les domaines publics soient accessibles depuis n'importe quel appareil connecté à Internet, les domaines VPC nécessitent une forme quelconque de VPN ou de proxy.
- Par rapport aux domaines publics, les domaines de VPC affichent moins d'informations dans la console . En particulier, l'onglet État du cluster n'inclut pas d'informations sur les partitions et l'onglet Index n'est pas présent.
- Les points de terminaison du domaine prennent différentes formes (`https://search-domain-name` contre `https://vpc-domain-name`).
- Vous ne pouvez pas appliquer des stratégies d'accès basées sur l'adresse IP aux domaines résidant au sein d'un VPC étant donné que les groupes de sécurité appliquent déjà les stratégies d'accès basées sur l'adresse IP.

Limites

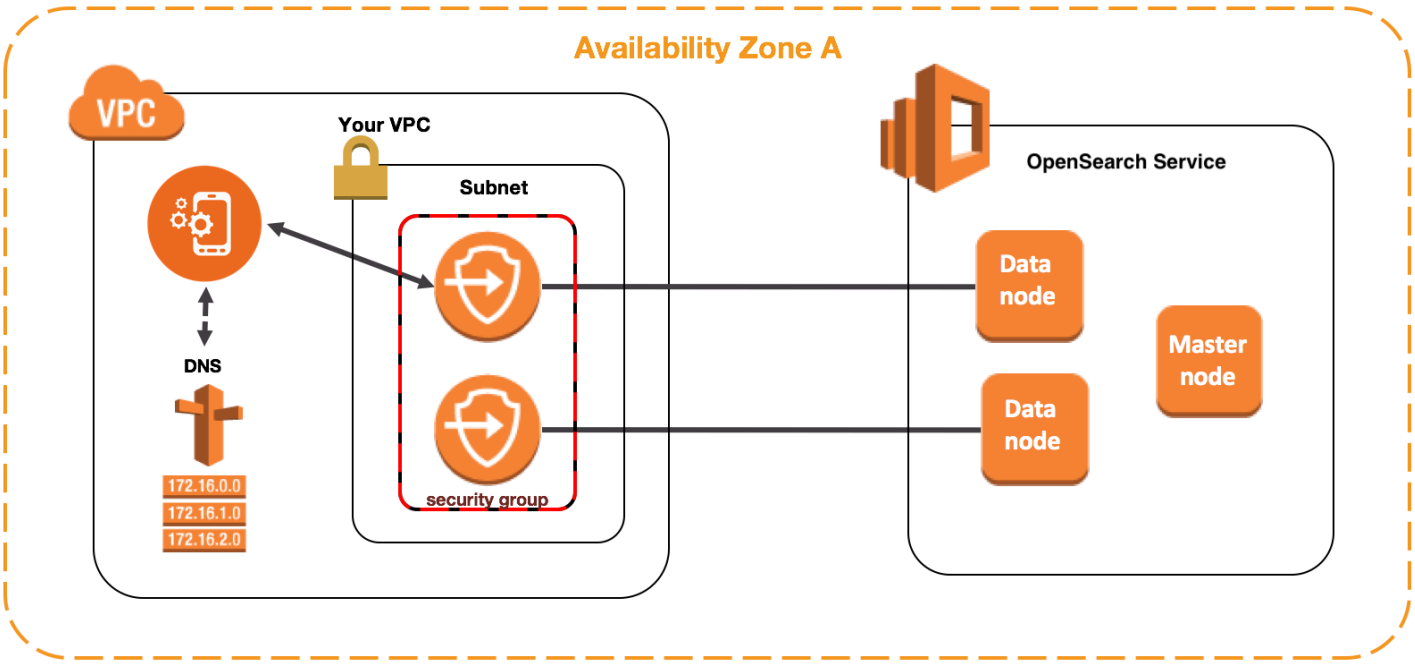
L'exploitation d'un domaine de OpenSearch service au sein d'un VPC présente les limites suivantes :

- Si vous lancez un nouveau domaine au sein d'un VPC, vous ne pouvez pas ultérieurement lui faire utiliser un point de terminaison public. L'inverse est également vrai : si vous créez un domaine avec un point de terminaison public, vous ne pouvez pas ultérieurement le placer au sein d'un VPC. Au lieu de cela, vous devez créer un nouveau domaine et migrer vos données.
- Vous pouvez lancer votre domaine au sein d'un VPC ou utiliser un point de terminaison public, mais vous ne pouvez pas faire les deux. Vous devez choisir l'un ou l'autre lorsque vous créez votre domaine.
- Vous ne pouvez pas lancer votre domaine au sein d'un VPC qui utilise une location dédiée. Vous devez utiliser un VPC avec une location définie sur Par défaut.
- Après avoir placé un domaine au sein d'un VPC, vous ne pouvez plus le déplacer vers un autre VPC, mais vous pouvez modifier les paramètres des sous-réseaux et des groupes de sécurité.
- Pour accéder à l'installation par défaut des OpenSearch tableaux de bord pour un domaine résidant dans un VPC, les utilisateurs doivent avoir accès au VPC. Ce processus varie selon la configuration du réseau, mais implique généralement la connexion à un VPN ou à un réseau géré ou l'utilisation d'un serveur proxy ou d'une passerelle de transit. Pour en savoir plus, consultez the section called “À propos des stratégies d'accès pour les domaines de VPC”, le Guide de l'utilisateur Amazon VPC et the section called “Contrôle de l'accès aux tableaux de bord ”.

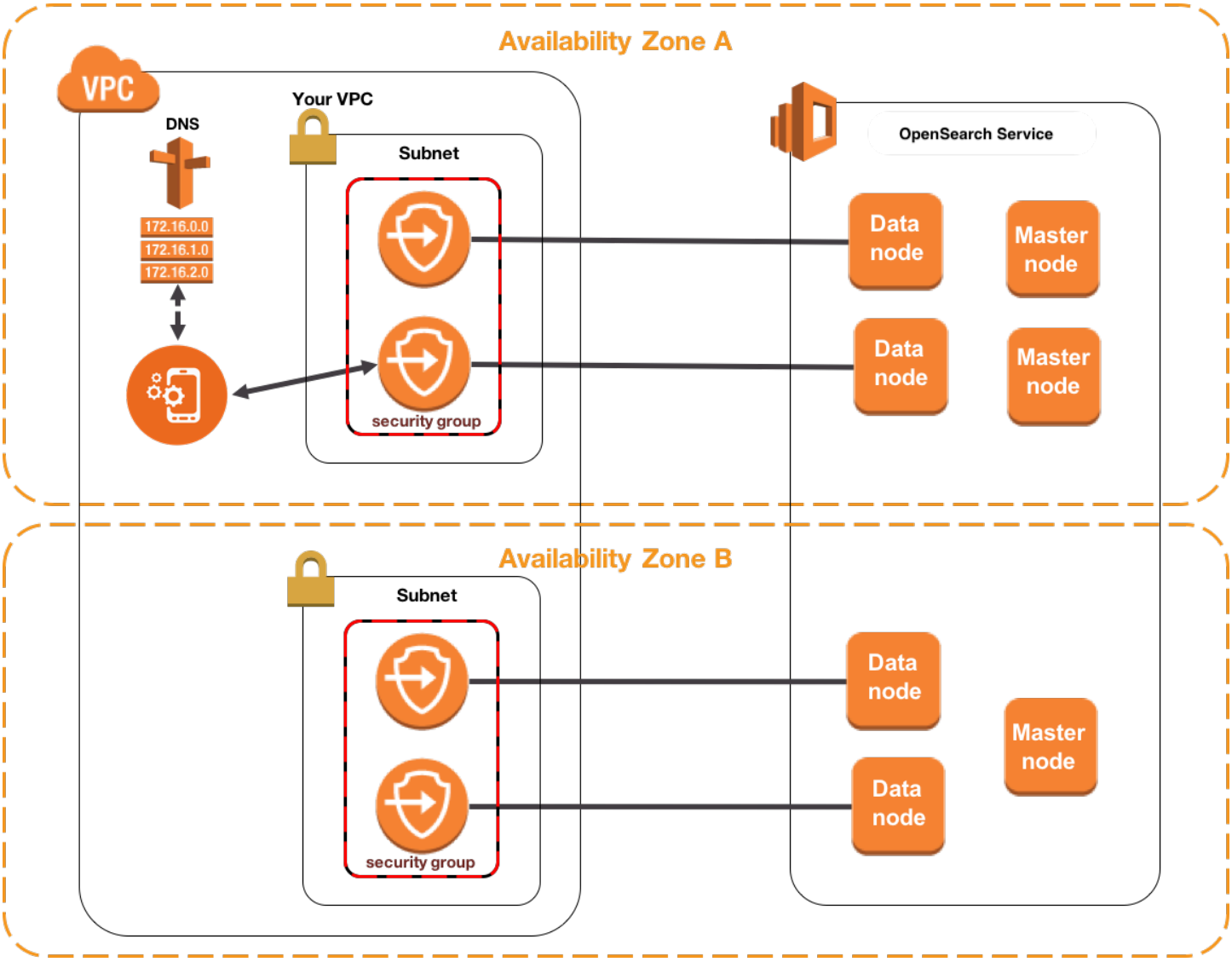
Architecture

À des fins d'assistance VPCs, OpenSearch Service place un point de terminaison dans un, deux ou trois sous-réseaux de votre VPC. Si vous activez plusieurs zones de disponibilité pour votre domaine, chaque sous-réseau doit se trouver dans une zone de disponibilité différente de la même région. Si vous n'utilisez qu'une seule zone de disponibilité, le OpenSearch service place un point de terminaison dans un seul sous-réseau.

L'illustration suivante montre l'architecture VPC avec une zone de disponibilité :



L'illustration suivante montre l'architecture VPC avec deux zones de disponibilité :



OpenSearch Le service place également une Elastic Network Interface (ENI) dans le VPC pour chacun de vos nœuds de données. OpenSearch Le service attribue à chaque ENI une adresse IP privée issue de la plage d' IPv4adresses de votre sous-réseau. Le service attribue également un nom d'hôte DNS public (qui est le point de terminaison du domaine) aux adresses IP. Vous devez utiliser un service DNS public pour résoudre le point de terminaison (qui est un nom d'hôte DNS) par les adresses IP appropriées des nœuds de données :

- Si votre VPC utilise le serveur DNS fourni par Amazon en définissant l'`enableDnsSupport`option sur `true` (valeur par défaut), la résolution du point de terminaison du OpenSearch service aboutira.

- Si votre VPC utilise un serveur DNS privé et que le serveur peut accéder aux serveurs DNS publics faisant autorité pour résoudre les noms d'hôte DNS, la résolution du point de terminaison du OpenSearch service aboutira également.

Dans la mesure où les adresses IP peuvent changer, vous devez résoudre le point de terminaison du domaine régulièrement afin de pouvoir toujours accéder aux nœuds de données corrects. Nous vous recommandons de définir un intervalle de résolution DNS d'une minute. Si vous utilisez un client, vous devez également vous assurer que le cache DNS du client est nettoyé.

Migration d'un accès public vers un accès VPC

Lorsque vous créez un domaine, vous devez spécifier s'il doit avoir un point de terminaison public ou résider au sein d'un VPC. Une fois le domaine créé, vous ne pouvez pas passer de l'un à l'autre. Au lieu de cela, vous devez créer un nouveau domaine et réindexer ou migrer vos données manuellement. Les instantanés constituent un moyen simple de migrer vos données. Pour plus d'informations sur la prise d'instantanés et leur restauration, consultez the section called “Création d'instantanés d'index”.

À propos des stratégies d'accès pour les domaines de VPC

Le fait de placer votre domaine de OpenSearch service au sein d'un VPC fournit une couche de sécurité intrinsèque et solide. Lorsque vous créez un domaine avec un accès public, le point de terminaison prend la forme suivante :

```
https://search-domain-name-identifier.region.es.amazonaws.com
```

Comme l'étiquette « public » l'indique, ce point de terminaison est accessible à partir de n'importe quel appareil connecté à Internet, même si vous pouvez (et devriez) en contrôler l'accès. Si vous accédez au point de terminaison via un navigateur Web, il est possible que vous receviez un message `Not Authorized`, mais la demande atteindra le domaine.

Lorsque vous créez un domaine avec un accès VPC, le point de terminaison semble identique à un point de terminaison public :

```
https://vpc-domain-name-identifier.region.es.amazonaws.com
```

toutefois, si vous essayez d'accéder au point de terminaison via un navigateur Web, vous verrez peut-être la demande expirer. Pour envoyer des demandes `GET` mêmes basiques, votre ordinateur doit pouvoir se connecter au VPC. Cette connexion s'effectue souvent via un réseau VPN, une

passerelle de transit, un réseau géré ou un serveur proxy. Pour en savoir plus sur les différentes formes de connexion, consultez Exemples pour VPC dans le Guide de l'utilisateur Amazon VPC. Pour bénéficier d'un exemple centré sur le développement, consultez the section called “Test des domaines de VPC”.

Outre cette exigence de connectivité, vous pouvez VPCs gérer l'accès au domaine par le biais de groupes de sécurité. Dans de nombreux cas d'utilisation, cette combinaison de fonctions de sécurité est suffisante, et vous pouvez sans problème appliquer une stratégie d'accès ouverte au domaine.

Le fait de fonctionner avec une politique d'accès ouvert ne signifie pas que n'importe qui sur Internet peut accéder au domaine du OpenSearch Service. Cela signifie plutôt que si une demande atteint le domaine de OpenSearch service et que les groupes de sécurité associés l'autorisent, le domaine accepte la demande. La seule exception est si vous utilisez un contrôle d'accès précis ou une stratégie d'accès qui spécifie les rôles IAM. Dans ces différentes situations, pour que le domaine accepte une demande, les groupes de sécurité doivent l'autoriser et la demande doit être signée avec des informations d'identification valides.

> **Note**
>
> Étant donné que les groupes de sécurité appliquent déjà des politiques d'accès basées sur l'IP, vous ne pouvez pas appliquer de politiques d'accès basées sur l'IP aux domaines de OpenSearch service qui résident au sein d'un VPC. Si vous utilisez un accès public, les stratégies basées sur l'adresse IP sont toujours disponibles.

Avant de commencer : Prérequis pour l'accès à un VPC

Avant de pouvoir activer une connexion entre un VPC et votre nouveau domaine de OpenSearch service, vous devez effectuer les opérations suivantes :

- Créer un VPC

  Pour créer votre VPC, vous pouvez utiliser la console Amazon VPC, la AWS CLI ou l'une des. AWS SDKs Pour plus d'informations, consultez la section Travailler avec VPCs dans le guide de l'utilisateur Amazon VPC. Si vous avez déjà un VPC, vous pouvez ignorer cette étape.
- Réserver des adresses IP

  OpenSearch Le service permet de connecter un VPC à un domaine en plaçant des interfaces réseau dans un sous-réseau du VPC. Chaque interface réseau est associée à une adresse IP.

Vous devez réserver un nombre suffisant d'adresses IP dans le sous-réseau pour les interfaces réseau. Pour plus d'informations, consultez Réservation d'adresses IP dans un sous-réseau VPC.

Test des domaines de VPC

La sécurité renforcée d'un VPC peut transformer en défi la connexion à votre domaine et l'exécution des tests de base. Si vous possédez déjà un domaine OpenSearch Service VPC et que vous préférez ne pas créer de serveur VPN, essayez la procédure suivante :

1. Pour la stratégie d'accès de votre domaine, choisissez Only use fine-grained access control (Utiliser uniquement le contrôle précis des accès). Vous pouvez toujours mettre à jour ce paramètre après avoir fini les tests.
2. Créez une EC2 instance Amazon Amazon Linux dans le même VPC, le même sous-réseau et le même groupe de sécurité que votre OpenSearch domaine de service.

   Comme cette instance est conçue à des fins de test et n'a besoin d'effectuer que très peu de tâches, choisissez un type d'instance peu coûteux comme `t2.micro`. Attribuez à l'instance une adresse IP publique et créez une nouvelle paire de clés ou choisissez-en une déjà existante. Si vous créez une nouvelle clé, téléchargez-la dans votre répertoire `~/.ssh`.

   Pour en savoir plus sur la création d'instances, consultez Getting started with Amazon EC2 Linux instances.
3. Ajoutez une passerelle Internet à votre VPC.
4. Dans la table de routage pour votre VPC, ajoutez une nouvelle route. Pour Destination, spécifiez un bloc d'adresse CIDR qui contient l'adresse IP publique de votre ordinateur. Pour Cible, spécifiez la passerelle Internet que vous venez de créer.

   Par exemple, vous pouvez spécifier `123.123.123.123/32` pour votre ordinateur seulement ou `123.123.123.0/24` pour une gamme d'ordinateurs.
5. Pour le groupe de sécurité, spécifiez deux règles entrantes :

Type	Protocole	Plage de ports	Source
SSH (22)	TCP (6)	22	*your-cidr-block*
HTTPS (443)	TCP (6)	443	*your-security-group-id*

La première règle vous permet d'accéder à votre EC2 instance par SSH. Le second permet à l' EC2 instance de communiquer avec le domaine OpenSearch de service via HTTPS.

6. Depuis le terminal, exécutez la commande suivante :

```
ssh -i ~/.ssh/your-key.pem ec2-user@your-ec2-instance-public-ip -N -L
 9200:vpc-domain-name-identifier.region.es.amazonaws.com:443
```

Cette commande crée un tunnel SSH qui transmet les demandes à https://localhost:9200 à votre domaine OpenSearch de service via l' EC2 instance. La spécification du port 9200 dans la commande simule une OpenSearch installation locale, mais utilisez le port de votre choix. OpenSearch Le service accepte uniquement les connexions via le port 80 (HTTP) ou 443 (HTTPS).

La commande ne fournit pas de commentaires et s'exécute indéfiniment. Pour l'arrêter, appuyez sur `Ctrl + C`.

7. Accédez à https://localhost:9200/_dashboards/ dans votre navigateur Web. Vous devrez peut-être accepter une exception de sécurité.

Vous pouvez également envoyer des demandes à https://localhost:9200 en utilisant curl, Postman ou votre langage de programmation favori.

> **Tip**
>
> Si vous rencontrez des erreurs curl en raison d'une incompatibilité de certificat, essayez l'indicateur `--insecure`.

Réservation d'adresses IP dans un sous-réseau VPC

OpenSearch Le service connecte un domaine à un VPC en plaçant des interfaces réseau dans un sous-réseau du VPC (ou dans plusieurs sous-réseaux du VPC si vous activez plusieurs zones de disponibilité). Chaque interface réseau est associée à une adresse IP. Avant de créer votre domaine de OpenSearch service, vous devez disposer d'un nombre suffisant d'adresses IP disponibles dans chaque sous-réseau pour accueillir les interfaces réseau.

Voici la formule de base : le nombre d'adresses IP que le OpenSearch Service réserve dans chaque sous-réseau est trois fois supérieur au nombre de nœuds de données, divisé par le nombre de zones de disponibilité.

Exemples

- Si un domaine possède 9 nœuds de données dans trois zones de disponibilité, le nombre d'adresses IP par sous-réseau est de 9 * 3 / 3 = 9.
- Si un domaine possède 8 nœuds de données dans deux zones de disponibilité, le nombre d'adresses IP par sous-réseau est de 8 * 3 / 2 = 12.
- Si un domaine possède 6 nœuds de données dans une zone de disponibilité, le nombre d'adresses IP par sous-réseau est de 6 * 3 / 1 = 18.

Lorsque vous créez le domaine, le OpenSearch Service réserve les adresses IP, en utilise certaines pour le domaine et réserve le reste aux déploiements bleu/vert. Vous pouvez voir les interfaces réseau et leurs adresses IP associées dans la section Interfaces réseau de la EC2 console Amazon. La colonne Description indique le domaine OpenSearch de service auquel l'interface réseau est associée.

> **Tip**
>
> Nous vous recommandons de créer des sous-réseaux dédiés pour les adresses IP réservées au OpenSearch Service. En utilisant des sous-réseaux dédiés, vous évitez les chevauchements avec les autres applications et services et vous êtes sûr de pouvoir réserver des adresses IP supplémentaires si vous avez besoin de faire évoluer votre cluster ultérieurement. Pour plus d'informations, consultez Création d'un sous-réseau dans votre VPC.

Vous pouvez également envisager de configurer des nœuds de coordination dédiés afin de réduire le nombre de réservations d'adresses IP privées requises pour votre domaine VPC. OpenSearchattache une Elastic Network Interface (ENI) à vos nœuds de coordination dédiés au lieu de vos nœuds de données. Les nœuds coordinateurs dédiés représentent généralement environ 10 % du total des nœuds de données. Par conséquent, un plus petit nombre d'adresses IP privées sera réservé aux domaines VPC.

Rôle lié à un service pour l'accès VPC

Un rôle lié à un service est un type unique de rôle IAM qui délègue des autorisations à un service afin qu'il puisse créer et gérer des ressources en votre nom. OpenSearch Le service nécessite un rôle lié au service pour accéder à votre VPC, créer le point de terminaison du domaine et placer les interfaces réseau dans un sous-réseau de votre VPC.

OpenSearch Le service crée automatiquement le rôle lorsque vous utilisez la console de OpenSearch service pour créer un domaine au sein d'un VPC. Pour que cette création automatique aboutisse, vous devez avoir les autorisations permettant d'effectuer l'action `iam:CreateServiceLinkedRole`. Pour en savoir plus, consultez Autorisations de rôles liés à un service dans le Guide de l'utilisateur IAM.

Une fois que OpenSearch Service a créé le rôle, vous pouvez le visualiser (`AWSServiceRoleForAmazonOpenSearchService`) à l'aide de la console IAM.

Pour plus d'informations sur ce rôle et la manière de le supprimer des autorisations, reportez-vous à la section the section called “Utilisation des rôles liés à un service”.

Création d'instantanés d'index dans Amazon Service OpenSearch

Les instantanés d'Amazon OpenSearch Service sont des sauvegardes des index et de l'état d'un cluster. L'état inclut les paramètres de cluster, les informations de nœud, les paramètres d'index et l'allocation des partitions.

OpenSearch Les instantanés de service se présentent sous les formes suivantes :

- Les instantanés automatiques sont uniquement destinés à la récupération de cluster. Vous pouvez les utiliser pour restaurer votre domaine en cas de statut de cluster rouge ou de perte de données. Pour plus d'informations, consultez la section Restauration des instantanés ci-dessous. OpenSearch Le service stocke les instantanés automatisés dans un compartiment Amazon S3 préconfiguré sans frais supplémentaires.
- Les instantanés manuels sont destinés à la récupération de cluster ou au déplacement de données d'un cluster à un autre. Vous devez initier les instantanés manuels. Ces instantanés sont stockés dans votre propre compartiment Amazon S3 et des frais S3 standard s'appliquent. Si vous disposez d'un instantané provenant d'un OpenSearch cluster autogéré, vous pouvez l'utiliser pour migrer vers un domaine de OpenSearch service. Pour plus d'informations, consultez la section Migration vers Amazon OpenSearch Service.

Tous les domaines de OpenSearch service prennent des instantanés automatisés, mais la fréquence varie comme suit :

- Pour les domaines exécutant Elasticsearch 5.3 OpenSearch ou version ultérieure, OpenSearch Service prend des instantanés automatisés toutes les heures et en conserve jusqu'à 336 pendant 14 jours. Les instantanés horaires sont moins perturbateurs en raison de leur nature progressive. Ils fournissent également un point de récupération plus récent en cas de problèmes de domaine.
- Pour les domaines exécutant Elasticsearch 5.1 et versions antérieures, le OpenSearch Service prend des instantanés automatisés quotidiens pendant l'heure que vous spécifiez, en conserve jusqu'à 14 et ne conserve aucune donnée d'instantané pendant plus de 30 jours.

Si votre cluster passe au statut rouge, tous les instantanés automatiques échouent tant que l'état du cluster persiste. Si vous ne corrigez pas le problème dans un délai de deux semaines, vous risquez de perdre définitivement les données de votre cluster. Pour obtenir les étapes de dépannage, consultez the section called “Statut de cluster rouge”.

Prérequis

Pour créer manuellement des instantanés, vous devez utiliser IAM et Amazon S3. Vérifiez que vous répondez aux conditions préalables suivantes avant d'essayer de prendre un instantané.

Prérequis	Description
Compartiment S3	Créez un compartiment S3 pour stocker les instantanés manuels de votre domaine OpenSearch de service. Pour obtenir des instructions, consultez la section Création d'un bucket à usage général dans le guide de l'utilisateur d'Amazon Simple Storage Service. Mémorisez le nom du compartiment pour l'utiliser aux emplacements suivants : • Déclaration `Resource` de la politique IAM attachée à votre rôle IAM • Client Python utilisé pour enregistrer un référentiel d'instantanés (si vous utilisez cette méthode)

Prérequis	Description
	Important N'appliquez pas de règle de cycle de vie S3 Glacier à ce compartiment. Les instantanés manuels ne prennent pas en charge la classe de stockage S3 Glacier.

Prérequis	Description
Rôle IAM	Créez un rôle IAM pour déléguer des autorisations au OpenSearch service. Pour obtenir des instructions, consultez la section Création d'un rôle IAM (console) du Guide de l'utilisateur IAM. Le reste de ce chapitre fait référence à ce rôle en tant que `TheSnapshotRole` . Attachement d'une politique IAM Attachez la stratégie suivante à `TheSnapshotRole` pour autoriser l'accès au compartiment S3 : JSON `{ "Version": "2012-10-17", "Statement": [{ "Action": [ "s3:ListBucket" ], "Effect": "Allow", "Resource": [ "arn:aws:s3::: amzn-s3-demo-bucket " ] }, { "Action": [ "s3:GetObject", "s3:PutObject", "s3:DeleteObject" ], "Effect": "Allow", "Resource": [ "arn:aws:s3::: amzn-s3-demo-bucket /*" ] } ] }`

Prérequis	Description
	Pour savoir comment associer une politique à un rôle, consultez la section Ajout d'autorisations d'identité IAM (console) dans le guide de l'utilisateur IAM. Modification de la relation d'approbation Modifiez la relation de confiance de `TheSnapshotRole` pour spécifier le OpenSearch service dans l'`Principal` instruction, comme indiqué dans l'exemple suivant : JSON `{ "Version": "2012-10-17", "Statement": [{ "Sid": "", "Effect": "Allow", "Principal": { "Service": "es.amazonaws.com" }, "Action": "sts:AssumeRole" }] }` Pour obtenir des instructions sur la modification de la relation de confiance, voir Mettre à jour une politique d'approbation de rôle dans le Guide de l'utilisateur IAM.

Prérequis	Description

Autorisations	Pour enregistrer le référentiel de snapshots, vous devez être en mesure de passer `TheSnapshotRole` au OpenSearch service. Vous avez également besoin de l'accès à l'action `es:ESHttpPut` . Pour accorder ces deux autorisations, attachez la politique suivante au rôle IAM dont les informations d'identification sont utilisées pour signer la demande :
	
	JSON
	
	`{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": "iam:PassRole", "Resource": "arn:aws:iam:: 123456789012 :role/TheSnapshotRole " }, { "Effect": "Allow", "Action": "es:ESHttpPut", "Resource": "arn:aws:es: region:123456789012 :domain/domain-name /*" } ] }`
	
	Si votre utilisateur ou votre rôle ne dispose pas d'`iam:PassRole` autorisations à transmettre`TheSnapshotRole` , vous risquez de rencontrer l'erreur courante suivante lorsque vous tenterez d'enregistrer un référentiel à l'étape suivante :
	
	`$ python register-repo.py {"Message":"User: arn:aws:iam:: 123456789012 :user/MyUserAccount is not authorized to perform: iam:PassRole on resource: arn:aws:iam:: 123456789012 :role/TheSnapshotRole "}`

Inscription d'un référentiel d'instantanés manuels

Vous devez enregistrer un référentiel de clichés auprès du OpenSearch Service avant de pouvoir prendre des instantanés d'index manuels. Cette opération unique nécessite que vous signiez votre AWS demande avec des informations d'identification autoriséesTheSnapshotRole, comme décrit dansthe section called “Prérequis”.

Étape 1 : Cartographier le rôle du snapshot dans les OpenSearch tableaux de bord (si vous utilisez un contrôle d'accès précis)

Le contrôle précis des accès introduit une étape supplémentaire lors de l'inscription d'un référentiel. Même si vous utilisez l'authentification de base HTTP à toutes les autres fins, vous devez mapper le rôle `manage_snapshots` à votre rôle IAM qui a les autorisations `iam:PassRole` pour transmettre `TheSnapshotRole`.

1. Accédez au plugin OpenSearch Dashboards correspondant à votre domaine OpenSearch de service. Vous pouvez trouver le point de terminaison Dashboards sur le tableau de bord de votre domaine sur la console OpenSearch de service.
2. Dans le menu principal, choisissez Security (Sécurité), Roles (Rôles), puis sélectionnez le rôle manage_snapshots.
3. Choisissez Mapped users (Utilisateurs mappés), Manage mapping (Gérer le mappage).
4. Ajoutez l'ARN du rôle ayant les autorisations de transmettre `TheSnapshotRole`. Placez le rôle ARNs sous Rôles du backend.

   ```
   arn:aws:iam::123456789123:role/role-name
   ```

5. Sélectionnez Map (Mapper) et vérifiez que l'utilisateur ou le rôle s'affiche sous Mapped users (Utilisateurs mappés).

Étape 2 : Inscrire un référentiel

L'onglet Snapshots suivant montre comment enregistrer un répertoire de snapshots. Pour les options spécifiques au chiffrement d'un instantané manuel et à l'enregistrement d'un instantané après la migration vers un nouveau domaine, consultez les onglets correspondants.

Snapshots

Pour enregistrer un référentiel de snapshots, envoyez une demande PUT au point de terminaison du domaine de OpenSearch service. Vous pouvez utiliser curl, le client Python d'exemple, Postman ou une autre méthode pour envoyer une demande signée afin d'enregistrer le référentiel de snapshots. Notez que vous ne pouvez pas utiliser de requête PUT dans la console OpenSearch Dashboards pour enregistrer le référentiel.

La demande se présente au format suivant :

```
PUT domain-endpoint/_snapshot/my-snapshot-repo-name
{
  "type": "s3",
  "settings": {
    "bucket": "amzn-s3-demo-bucket",
    "base_path": "my/snapshot/directory",
    "region": "region",
    "role_arn": "arn:aws:iam::123456789012:role/TheSnapshotRole"
  }
}
```

> **Note**
>
> Les noms de référentiels ne peuvent pas commencer par « cs- ». De plus, vous ne devriez pas écrire dans le même référentiel à partir de plusieurs domaines. Un seul domaine doit avoir un accès en écriture au référentiel.

Si votre domaine réside au sein d'un cloud privé virtuel (VPC), votre ordinateur doit être connecté au VPC pour que la demande puisse enregistrer le référentiel d'instantanés. L'accès à un VPC varie selon la configuration du réseau, mais implique généralement la connexion à un VPN ou un réseau d'entreprise. Pour vérifier que vous pouvez accéder au domaine de OpenSearch service, accédez `https://your-vpc-domain.region.es.amazonaws.com` à un navigateur Web et vérifiez que vous recevez la réponse JSON par défaut.

Lorsque votre compartiment Amazon S3 se trouve dans un autre domaine Région AWS que votre OpenSearch domaine, ajoutez le paramètre `"endpoint": "s3.amazonaws.com"` à la demande.

Encrypted snapshots

Vous ne pouvez actuellement pas utiliser de clés AWS Key Management Service (KMS) pour chiffrer les instantanés manuels, mais vous pouvez les protéger à l'aide du chiffrement côté serveur (SSE).

Pour activer SSE avec des clés gérées par S3 pour le bucket que vous utilisez comme référentiel de snapshots, ajoutez-le `"server_side_encryption": true` au `"settings"` bloc de la demande PUT. Pour plus d'informations, consultez Utilisation du chiffrement côté serveur avec des clés gérées par Amazon S3 (SSE-S3) dans le Guide de l'utilisateur Amazon Simple Storage Service.

Vous pouvez également utiliser des AWS KMS clés pour le chiffrement côté serveur sur le compartiment S3 que vous utilisez comme référentiel de snapshots. Si vous utilisez cette approche, veillez à `TheSnapshotRole` autoriser la AWS KMS clé utilisée pour chiffrer le compartiment S3. Pour plus d'informations, consultez Stratégies de clé dans le AWS KMS.

Domain migration

L'enregistrement d'un référentiel d'instantanés est une opération ponctuelle. Cela étant, pour migrer d'un domaine à un autre, vous devez enregistrer le référentiel d'instantanés sur l'ancien et le nouveau domaine. Le nom du référentiel est arbitraire.

Prenez en compte les instructions suivantes lors de la migration vers un nouveau domaine ou de l'enregistrement du même référentiel auprès de plusieurs domaines :

- Lors de l'enregistrement du référentiel sur le nouveau domaine, ajoutez `"readonly": true` au bloc `"settings"` de la demande PUT. Ce paramètre vous empêche d'écraser malencontreusement des données de l'ancien domaine. Un seul domaine doit avoir un accès en écriture au référentiel.
- Si vous migrez des données vers un domaine différent (par exemple Région AWS, d'un ancien domaine et d'un compartiment situés dans us-east-2 vers un nouveau domaine dans us-west-2), remplacez-les par dans l'instruction PUT et réessayez la demande. `"region": "region" "endpoint": "s3.amazonaws.com"`

Utilisation de l'exemple de client Python

Le client Python est plus facile à automatiser qu'une simple requête HTTP et peut être réutilisé. Si vous choisissez d'utiliser cette méthode pour enregistrer un référentiel d'instantanés, enregistrez l'exemple de code Python suivant en tant que fichier Python, comme `register-repo.py`. Le client

a besoin des packages AWS SDK pour Python (Boto3), requests et requests-aws4auth. Le client contient des exemples mis en commentaire pour d'autres opérations d'instantanés.

Mettez à jour les variables suivantes dans l'exemple de code : `host`, `region`, `path` et `payload`.

```
import boto3
import requests
from requests_aws4auth import AWS4Auth

host = '' # domain endpoint
region = '' # e.g. us-west-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

# Register repository

path = '/_snapshot/my-snapshot-repo-name' # the OpenSearch API endpoint
url = host + path

payload = {
  "type": "s3",
  "settings": {
    "bucket": "amzn-s3-demo-bucket",
    "base_path": "my/snapshot/directory",
    "region": "us-west-1",
    "role_arn": "arn:aws:iam::123456789012:role/snapshot-role"
  }
}

headers = {"Content-Type": "application/json"}

r = requests.put(url, auth=awsauth, json=payload, headers=headers)

print(r.status_code)
print(r.text)

# # Take snapshot
#
# path = '/_snapshot/my-snapshot-repo-name/my-snapshot'
# url = host + path
#
# r = requests.put(url, auth=awsauth)
```

```
#
# print(r.text)
#
# # Delete index
#
# path = 'my-index'
# url = host + path
#
# r = requests.delete(url, auth=awsauth)
#
# print(r.text)
#
# # Restore snapshot (all indexes except Dashboards and fine-grained access control)
#
# path = '/_snapshot/my-snapshot-repo-name/my-snapshot/_restore'
# url = host + path
#
# payload = {
#   "indices": "-.kibana*,-.opendistro_security,-.opendistro-*",
#   "include_global_state": False
# }
#
# headers = {"Content-Type": "application/json"}
#
# r = requests.post(url, auth=awsauth, json=payload, headers=headers)
#
# print(r.text)
#
# # Restore snapshot (one index)
#
# path = '/_snapshot/my-snapshot-repo-name/my-snapshot/_restore'
# url = host + path
#
# payload = {"indices": "my-index"}
#
# headers = {"Content-Type": "application/json"}
#
# r = requests.post(url, auth=awsauth, json=payload, headers=headers)
#
# print(r.text)
```

Prise d'instantanés manuels

Les instantanés ne sont pas créés instantanément. Ils prennent du temps et ne représentent pas point-in-time une vue parfaite du cluster. Lors de la création d'un instantané, vous pouvez toujours indexer des documents et adresser d'autres demandes au cluster. Toutefois, les nouveaux documents et les mises à jour des documents existants ne sont généralement pas inclus dans l'instantané. Le cliché inclut les partitions principales telles qu'elles existaient au moment OpenSearch de son lancement. En fonction de la taille de votre groupe de threads d'instantanés, différentes partitions peuvent être incluses dans l'instantané à différents moments. Pour connaître les meilleures pratiques en matière de capture d'écran, voirthe section called “Améliorez les performances des instantanés”.

Stockage et performances des instantanés

OpenSearch les instantanés sont incrémentiels, ce qui signifie qu'ils ne stockent que les données modifiées depuis le dernier instantané réussi. Cette nature incrémentielle signifie que la différence d'utilisation de disque entre des instantanés fréquents et rares est souvent minime. En d'autres termes, la réalisation d'instantanés horaires pendant une semaine (avec un total de 168 instantanés) peut ne pas utiliser beaucoup plus d'espace disque que celle d'un seul instantané à la fin de la semaine. De plus, plus vous prenez des instantanés fréquemment, plus vous les réalisez rapidement. Par exemple, les instantanés quotidiens peuvent prendre 20 à 30 minutes, alors que les instantanés horaires peuvent se terminer en quelques minutes. Certains OpenSearch utilisateurs prennent des instantanés toutes les demi-heures.

Prendre un instantané

Lorsque vous créez un paramètre, vous spécifiez les informations suivantes :

- Nom de votre référentiel d'instantanés
- Nom de l'instantané

Les exemples de ce chapitre utilisent curl, un client HTTP courant, pour des raisons de commodité et de concision. Pour transmettre un nom d'utilisateur et un mot de passe à votre demande curl, consultez le didacticiel de démarrage.

Si vos politiques d'accès spécifient des utilisateurs ou des rôles, vous devez signer vos demandes de capture instantanée. Pour curl, vous pouvez utiliser l'`--aws-sigv4`option avec la version 7.75.0 ou ultérieure. Vous pouvez également utiliser les exemples commentés de l'exemple de client Python

pour envoyer des requêtes HTTP signées aux mêmes points de terminaison que ceux utilisés par les commandes curl.

Pour prendre un instantané manuel, procédez comme suit :

1. Vous ne pouvez pas prendre un instantané si un instantané est en cours. Pour vérifier, exécutez la commande suivante :

   ```
   curl -XGET 'domain-endpoint/_snapshot/_status'
   ```

2. Exécutez la commande suivante pour prendre un instantané manuel :

   ```
   curl -XPUT 'domain-endpoint/_snapshot/repository-name/snapshot-name'
   ```

   Pour inclure ou exclure certains index et spécifier d'autres paramètres, ajoutez un corps de requête. Pour la structure de la demande, consultez la section Prendre des instantanés dans la OpenSearch documentation.

> **Note**
>
> Le temps nécessaire pour prendre un instantané augmente en fonction de la taille du domaine de OpenSearch service. Les opérations d'instantanés de longue durée rencontrent parfois l'erreur suivante : `504 GATEWAY_TIMEOUT`. Vous pouvez généralement ignorer ces erreurs et attendre que l'opération se termine avec succès. Exécutez la commande suivante pour vérifier l'état de tous les instantanés de votre domaine :
>
> ```
> curl -XGET 'domain-endpoint/_snapshot/repository-name/_all?pretty'
> ```

Restauration des instantanés

Avant de restaurer un instantané, assurez-vous que le domaine de destination n'utilise pas le mode Multi-AZ avec mode veille. L'activation du mode veille entraîne l'échec de l'opération de restauration.

> **Warning**
>
> Si vous utilisez des alias d'index, vous devez soit cesser d'écrire des demandes à un alias, soit passer de l'alias à un autre index avant de supprimer son index. L'arrêt des demandes d'écriture contribue à éviter le scénario suivant :
>
> 1. Vous supprimez un index, ce qui supprime également son alias.
> 2. Une demande d'écriture errante à l'alias maintenant supprimé crée un nouvel index avec le même nom que celui de l'alias.
> 3. Vous ne pouvez plus utiliser l'alias en raison d'un conflit de noms avec le nouvel index. Si vous avez basculé l'alias vers un autre index, spécifiez `"include_aliases": false` lorsque vous restaurez à partir d'un instantané.

Pour restaurer un instantané

1. Identifiez l'instantané que vous voulez restaurer. Assurez-vous que tous les paramètres de cet index, tels que les packages d'analyseurs personnalisés ou les paramètres d'exigences d'allocation, sont compatibles avec le domaine. Pour afficher tous les référentiels d'instantanés, exécutez la commande suivante :

   ```
   curl -XGET 'domain-endpoint/_snapshot?pretty'
   ```

   Une fois le référentiel identifié, exécutez la commande suivante pour afficher tous les instantanés :

   ```
   curl -XGET 'domain-endpoint/_snapshot/repository-name/_all?pretty'
   ```

   > **Note**
   >
   > La plupart des instantanés automatiques sont stockés dans le référentiel `cs-automated`. Si votre domaine chiffre les données au repos, elles sont stockées dans le référentiel `cs-automated-enc`. Si vous ne voyez pas le référentiel d'instantanés manuels que vous recherchez, vérifiez que vous l'avez enregistré dans le domaine.

2. (Facultatif) Supprimez ou renommez un ou plusieurs index du domaine de OpenSearch service en cas de conflit de dénomination entre les index du cluster et ceux du cliché. Vous ne pouvez

pas restaurer un instantané de vos index sur un OpenSearch cluster qui contient déjà des index portant le même nom.

En cas de conflits de noms d'index, les options suivantes s'offrent à vous :

- Supprimez les index du domaine de OpenSearch service existant, puis restaurez le snapshot.
- Renommez les index à mesure que vous les restaurez à partir de l'instantané, puis réindexez-les ultérieurement. Pour savoir comment renommer des index, consultez cet exemple de demande dans la OpenSearch documentation.
- Restaurez le snapshot dans un autre domaine OpenSearch de service (uniquement possible avec les snapshots manuels).

La commande suivante supprime tous les index existants d'un domaine :

```
curl -XDELETE 'domain-endpoint/_all'
```

Cependant, si vous ne prévoyez pas de restaurer tous les index, vous pouvez simplement en supprimer un :

```
curl -XDELETE 'domain-endpoint/index-name'
```

3. Pour restaurer un instantané, exécutez la commande suivante :

```
curl -XPOST 'domain-endpoint/_snapshot/repository-name/snapshot-name/_restore'
```

En raison d'autorisations spéciales sur les OpenSearch tableaux de bord et d'index de contrôle d'accès précis, les tentatives de restauration de tous les index peuvent échouer, en particulier si vous essayez de restaurer à partir d'un instantané automatique. L'exemple suivant restaure un seul index, `my-index`, depuis `2020-snapshot` vers le référentiel d'instantanés `cs-automated` :

```
curl -XPOST 'domain-endpoint/_snapshot/cs-automated/2020-snapshot/_restore' \
-d '{"indices": "my-index"}' \
-H 'Content-Type: application/json'
```

Vous pouvez également restaurer tous les index à l'exception des index Dashboards et des index de contrôle précis des accès :

```
curl -XPOST 'domain-endpoint/_snapshot/cs-automated/2020-snapshot/_restore' \
-d '{"indices": "-.kibana*,-.opendistro*"}' \
-H 'Content-Type: application/json'
```

Vous pouvez restaurer un instantané sans supprimer ses données à l'aide des `rename_replacement` paramètres `rename_pattern` et. Pour plus d'informations sur ces paramètres, consultez les champs de demande de l'API Restore Snapshot et l'exemple de demande dans la OpenSearch documentation.

> **Note**
>
> Si seule une partie des partitions primaires était disponible pour les index impliqués, un instantané peut avoir l'`state PARTIAL`. Cette valeur indique que les données d'au moins une partition n'ont pas été stockées avec succès. Vous pouvez toujours effectuer une restauration à partir d'un instantané partiel, mais vous aurez probablement besoin d'anciens instantanés pour restaurer les index manquants.

Suppression d’instantanés manuels

Pour supprimer un instantané manuel, exécutez la commande suivante :

```
DELETE _snapshot/repository-name/snapshot-name
```

Automatisation des instantanés grâce à la gestion des instantanés

Vous pouvez configurer une politique de gestion des instantanés (SM) dans les OpenSearch tableaux de bord afin d'automatiser la création et la suppression périodiques d'instantanés. SM peut prendre un instantané d'un groupe d'indices, tandis que Index State Management ne peut prendre qu'un seul instantané par index. Pour utiliser SM in OpenSearch Service, vous devez enregistrer votre propre référentiel Amazon S3. Pour obtenir des instructions relatives à l'enregistrement de votre référentiel, consultez la section Enregistrement d'un référentiel de snapshots manuel.

Avant SM, OpenSearch Service offrait une fonction de capture instantanée gratuite et automatisée qui est toujours activée par défaut. Cette fonctionnalité envoie des instantanés dans le référentiel géré par le service. `cs-*` Pour désactiver cette fonctionnalité, contactez. Support

Pour plus d'informations sur la fonctionnalité SM, consultez la section Gestion des snapshots dans la OpenSearch documentation.

SM ne prend actuellement pas en charge la création de snapshots sur plusieurs types d'index. Par exemple, si vous essayez de créer un instantané sur plusieurs index avec le niveau chaud * et que certains index se trouvent dans le niveau chaud, la création de l'instantané échouera. Si vous avez besoin que votre instantané contienne plusieurs types d'index, utilisez l'action Instantané ISM jusqu'à ce que SM accepte cette option.

Configurer des autorisations

Si vous effectuez une mise à niveau vers la version 2.5 à partir d'une version précédente du domaine de OpenSearch service, il est possible que les autorisations de sécurité de gestion des snapshots ne soient pas définies sur le domaine. Les utilisateurs non administrateurs doivent être mappés à ce rôle afin d'utiliser la gestion des instantanés sur les domaines à l'aide d'un contrôle d'accès précis. Pour créer manuellement le rôle de gestion des snapshots, effectuez les opérations suivantes :

1. Dans les OpenSearch tableaux de bord, accédez à Sécurité, puis sélectionnez Autorisations.
2. Choisissez Create action group (Créer un groupe d'actions) et configurez les groupes suivants :

Nom du groupe	Autorisations
`snapshot_management_full_access`	• `cluster:admin/opensearch/snapshot_management/*` • `cluster:admin/opensearch/notifications/feature/publish` • `cluster:admin/repository/*` • `cluster:admin/snapshot/*`
`snapshot_management_read_access`	• `cluster:admin/opensearch/snapshot_management/policy/get` • `cluster:admin/opensearch/snapshot_management/policy/search` • `cluster:admin/opensearch/snapshot_management/policy/explain` • `cluster:admin/repository/get` • `cluster:admin/snapshot/get`

3. Choisissez Roles (Rôles), puis Create role (Créer un rôle).
4. Nommez le rôle snapshot_management_role.
5. Pour les autorisations du cluster, sélectionnez `snapshot_management_full_access` ou`snapshot_management_read_access`.
6. Choisissez Créer.
7. Après avoir créé le rôle, associez-le à n'importe quel rôle d'utilisateur ou de backend qui gérera les instantanés.

Considérations

Lorsque vous configurez la gestion des snapshots, tenez compte des points suivants :

- Une politique est autorisée par référentiel.
- Jusqu'à 400 instantanés sont autorisés pour une politique.
- Cette fonctionnalité ne s'exécute pas si le statut de votre domaine est rouge, s'il est soumis à une pression JVM élevée (85 % ou plus) ou si la fonction de capture instantanée est bloquée. Lorsque les performances globales d'indexation et de recherche de votre cluster sont affectées, SM peut également être affecté.
- Une opération de capture instantanée ne démarre qu'une fois l'opération précédente terminée, de sorte qu'aucune opération de capture instantanée simultanée n'est activée par une seule politique.
- Plusieurs politiques associées au même calendrier peuvent entraîner un pic de ressources. Si les indices instantanés des politiques se chevauchent, les opérations de capture instantanée au niveau des partitions ne peuvent être exécutées que de manière séquentielle, ce qui peut entraîner un problème de performance en cascade. Si les politiques partagent un référentiel, il y aura un pic d'opérations d'écriture dans ce référentiel.
- Nous vous recommandons de planifier l'automatisation de vos opérations de capture instantanée au maximum une fois par heure, sauf si vous avez un cas d'utilisation particulier.

Automatisation des instantanés avec Index State Management

Vous pouvez utiliser l'opération de capture instantanée ISM (Index State Management) pour déclencher automatiquement des instantanés d'index en fonction de l'évolution de leur âge, de leur taille ou du nombre de documents. ISM est préférable lorsque vous avez besoin d'un instantané par index. Si vous avez besoin d'un instantané d'un groupe d'indices, reportez-vous àAutomatisation des instantanés grâce à la gestion des instantanés.

Pour utiliser SM in OpenSearch Service, vous devez enregistrer votre propre référentiel Amazon S3. Pour un exemple de politique ISM utilisant l'opération `snapshot`, consultez Exemples de politiques.

Utilisation de Curator pour les instantanés

Si ISM ne fonctionne pas pour la gestion des index et des instantanés, vous pouvez utiliser Curator à la place. Curator offre des fonctionnalités de filtrage avancé qui peuvent simplifier les tâches de gestion sur les clusters complexes. Utilisez pip pour installer Curator.

```
pip install elasticsearch-curator
```

Vous pouvez utiliser Curator comme une interface de ligne de commande (CLI) ou API Python. Si vous utilisez l'API Python, vous devez utiliser la version 7.13.4 ou une version antérieure du client elasticsearch-py existant. Le client opensearch-py n'est pas pris en charge.

Si vous utilisez l'interface CLI, exportez vos informations d'identification au niveau de la ligne de commande et configurez `curator.yml` comme suit :

```
client:
  hosts: search-my-domain.us-west-1.es.amazonaws.com
  port: 443
  use_ssl: True
  aws_region: us-west-1
  aws_sign_request: True
  ssl_no_validate: False
  timeout: 60

logging:
  loglevel: INFO
```

Mise à niveau des domaines Amazon OpenSearch Service

> **Note**
>
> OpenSearch et les mises à niveau de la version d'Elasticsearch diffèrent des mises à jour du logiciel de service. Pour plus d'informations sur la mise à jour du logiciel de OpenSearch service pour votre domaine de service, consultezthe section called "Mises à jour du logiciel de service".

Amazon OpenSearch Service propose des mises à niveau pour les domaines qui exécutent la OpenSearch version 1.0 ou ultérieure, ou Elasticsearch 5.1 ou ultérieur. Si vous utilisez des services tels qu'Amazon Data Firehose ou Amazon CloudWatch Logs pour diffuser des données vers OpenSearch Service, vérifiez que ces services prennent en charge la dernière version de OpenSearch avant de procéder à la migration.

Chemins de mise à niveau pris en charge

OpenSearch Service prend actuellement en charge les chemins de mise à niveau suivants :

De la version	Vers la version
OpenSearch 1.3 ou 2. x	OpenSearch 2. x OpenSearch 2.17 activera la recherche de segments simultanés par défaut en mode auto si le domaine répond aux conditions suivantes : • Aucun paramètre de recherche simultanée antérieur n'est défini de manière explicite. • Toutes les instances de données (chaudes et chaudes) sont de type 2.xl ou supérieur. • L'utilisation moyenne du processeur p90 sur les instances de données (chaudes et chaudes) pendant plus d'une semaine est inférieure à 45 %. Pour plus de détails sur les paramètres de recherche de segments simultanés ici, voir Recherche de segments simultanés. La version 2.3 comporte les modifications majeures suivantes : • Le `type` paramètre a été supprimé de tous les points de terminaison d' OpenSearch API dans la version 2.0. Pour plus d'informations, veuillez consulter la rubrique Modifications majeures (langue française non garantie). • Si votre domaine contient des index (à chaud ou à froid) initialement créés dans Elasticsearch 6.8, ceux-ci ne sont pas compatibles avec la version 2.3. UltraWarm OpenSearch Avant de passer à la version 2.3, vous devez réindexer les index incompati bles. Pour les index incompatibles UltraWarm ou à froid, migrez-les vers un

De la version	Vers la version
	stockage à chaud, réindexez les données, puis faites-les migrer à nouveau vers un stockage tiède ou à froid. Vous pouvez également supprimer les index si vous n'en avez plus besoin. Si vous mettez accidentellement à niveau votre domaine vers la version 2.3 sans effectuer ces étapes au préalable, vous ne pourrez pas migrer les index incompatibles hors de leur niveau de stockage actuel. Votre seule option est de les supprimer.
OpenSearch 1. x	OpenSearch 1. x
Elasticsearch 7.x	Elasticsearch 7. x ou OpenSearch 1. x **Important** OpenSearch 1. x introduit de nombreuses modifications importantes. Pour en savoir plus, consultez OpenSearch Renommage d'Amazon Service.

De la version	Vers la version
Elasticsearch 6.8	Elasticsearch 7. x ou OpenSearch 1. x **Important** Elasticsearch 7.0 et OpenSearch 1.0 incluent de nombreuses modifications importantes. Avant de lancer une mise à niveau sur place, nous vous recommandons de prendre un instantané manuel du 6. domaine x, en le restaurant lors d'un test 7. x ou OpenSearch 1. domaine x, et utilisation de ce domaine de test pour identifier les problèmes de mise à niveau potentiels. Pour annuler les modifications apportées à la OpenSearch version 1.0, voir OpenSearch Renommage d'Amazon Service. Comme Elasticsearch 6.x, les index ne peuvent contenir qu'un seul type de mappage, mais celui-ci doit désormais être nommé `_doc`. Par conséquent, certaines APIs n'ont plus besoin d'un type de mappage dans le corps de la demande (comme l'`_bulk`API). Pour les nouveaux index, auto-hébergez Elasticsearch 7. x et OpenSearch 1. x ont un nombre de partitions par défaut de un. OpenSearch Domaines de service sur Elasticsearch 7. x et conserver ultérieurement la valeur par défaut précédente, à savoir cinq.
Elasticsearch 6.x	Elasticsearch 6.x

De la version	Vers la version
Elasticsearch 5.6	Elasticsearch 6.x **Important** Les index créés dans la version 6.x ne prennent plus en charge plusieurs types de mappages. Les index créés dans la version 5.x prennent toujours en charge plusieurs types de mappages lors de leur restauration dans un cluster 6.x. Vérifiez que votre code client ne crée qu'un seul type de mappage par index. Pour minimiser les temps d'arrêt lors de la mise à niveau d'Elasticsearch 5.6 vers la version 6. x, OpenSearch Service réindexe l'`.kibana`index, le supprime `.kibana-6` `.kibana`, crée un alias nommé `.kibana` et mappe le nouvel index avec le nouvel alias.
Elasticsearch 5.x	Elasticsearch 5.x

Le processus de mise à niveau comporte trois étapes :

1. Vérifications avant la mise à niveau : le OpenSearch service effectue des vérifications afin de détecter tout problème susceptible de bloquer une mise à niveau, et ne passe à l'étape suivante que lorsque ces vérifications ont abouti.
2. Instantané : le OpenSearch service prend un instantané du cluster OpenSearch ou Elasticsearch et ne passe à l'étape suivante qu'une fois l'instantané réalisé. Si la mise à niveau échoue, OpenSearch Service utilise cet instantané pour restaurer le cluster à son état d'origine. Pour de plus amples informations, veuillez consulter the section called “Impossible de revenir à une version plus ancienne après la mise à niveau”.
3. Mise à niveau : le OpenSearch service démarre la mise à niveau, ce qui peut prendre de 15 minutes à plusieurs heures. OpenSearch Les tableaux de bord peuvent ne pas être disponibles pendant la totalité ou une partie de la mise à niveau.

Mise à niveau d'un domaine (console)

Le processus de mise à niveau est irréversible et ne peut pas être suspendu ou annulé. Au cours d'une mise à niveau, vous ne pouvez pas apporter de modifications à la configuration du domaine. Avant de commencer une mise à niveau, assurez-vous de vouloir poursuivre. Vous pouvez utiliser ces mêmes étapes pour effectuer la vérification avant la mise à niveau, sans réellement démarrer de mise à niveau.

Si le cluster inclut des nœuds principaux dédiés, les OpenSearch met à niveau sans temps d'arrêt. Sinon, le cluster peut ne pas répondre pendant plusieurs secondes après la mise à niveau pendant qu'il choisit un nœud maître.

Pour mettre à niveau un domaine vers une version ultérieure d' OpenSearch Elasticsearch

1. Prenez un instantané manuel de votre domaine. Cet instantané sert de sauvegarde que vous pouvez restaurer sur un nouveau domaine si vous voulez utiliser à nouveau la OpenSearch version précédente.
2. Accédez à la console https://aws.amazon.comet choisissez Se connecter à la console.
3. Sous Analytics, sélectionnez Amazon OpenSearch Service.
4. Dans le volet de navigation, sous Domains (Domaines), choisissez le domaine que vous voulez mettre à jour.
5. Choisissez Actions et Upgrade (Mise à niveau).
6. Choisissez la version cible de la mise à niveau. Si vous effectuez une mise à niveau vers une OpenSearch version, l'option Enable compatibility mode (Activer le mode de compatibilité) apparaît. Si vous activez ce paramètre, OpenSearch signale sa version 7.10 pour permettre aux clients et plug-ins Elasticsearch OSS tels que Logstash de continuer à travailler avec Amazon Service. OpenSearch Vous pouvez désactiver cette configuration plus tard
7. Choisissez Upgrade (Mise à niveau).
8. Vérifiez le Statut sur le tableau de bord du domaine pour surveiller le statut de la mise à niveau.

Mise à niveau d'un domaine (CLI)

Vous pouvez utiliser les opérations suivantes pour identifier la version d' OpenSearch Elasticsearch appropriée pour votre domaine, démarrer une mise à niveau sur place, effectuer la vérification avant mise à niveau et voir l'état d'avancement :

- `get-compatible-versions` (`GetCompatibleVersions`)
- `upgrade-domain` (`UpgradeDomain`)
- `get-upgrade-status` (`GetUpgradeStatus`)
- `get-upgrade-history` (`GetUpgradeHistory`)

Pour plus d'informations, consultez la référence des commandes AWS CLI et la référence OpenSearch de l'API Amazon Service.

Mise à niveau d'un domaine (SDK)

Cet exemple utilise le client Python de OpenSearchServicebas niveau de l' AWS SDK for Python (Boto) pour vérifier si un domaine est éligible pour une mise à niveau vers une version spécifique, le met à niveau et vérifie continuellement le statut de la mise à niveau.

```
import boto3
from botocore.config import Config
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default Region.

DOMAIN_NAME = ''  # The name of the domain to upgrade
TARGET_VERSION = ''  # The version you want to upgrade the domain to. For example,
 OpenSearch_1.1

my_config = Config(
    # Optionally lets you specify a Region other than your default.
    region_name='us-east-1'
)
client = boto3.client('opensearch', config=my_config)


def check_versions():
    """Determine whether domain is eligible for upgrade"""
    response = client.get_compatible_versions(
        DomainName=DOMAIN_NAME
    )
    compatible_versions = response['CompatibleVersions']
    for i in range(len(compatible_versions)):
        if TARGET_VERSION in compatible_versions[i]["TargetVersions"]:
```

```
            print('Domain is eligible for upgrade to ' + TARGET_VERSION)
            upgrade_domain()
            print(response)
        else:
            print('Domain not eligible for upgrade to ' + TARGET_VERSION)


def upgrade_domain():
    """Upgrades the domain"""
    response = client.upgrade_domain(
        DomainName=DOMAIN_NAME,
        TargetVersion=TARGET_VERSION
    )
    print('Upgrading domain to ' + TARGET_VERSION + '...' + response)
    time.sleep(5)
    wait_for_upgrade()


def wait_for_upgrade():
    """Get the status of the upgrade"""
    response = client.get_upgrade_status(
        DomainName=DOMAIN_NAME
    )
    if (response['UpgradeStep']) == 'UPGRADE' and (response['StepStatus']) ==
 'SUCCEEDED':
        print('Domain successfully upgraded to ' + TARGET_VERSION)
    elif (response['StepStatus']) == 'FAILED':
        print('Upgrade failed. Please try again.')
    elif (response['StepStatus']) == 'SUCCEEDED_WITH_ISSUES':
        print('Upgrade succeeded with issues')
    elif (response['StepStatus']) == 'IN_PROGRESS':
        time.sleep(30)
        wait_for_upgrade()


def main():
    check_versions()


if __name__ == "__main__":
    main()
```

Résolution des problèmes liés aux échecs de validation

Lorsque vous lancez une mise à niveau de la version OpenSearch ou Elasticsearch, le OpenSearch Service effectue d'abord une série de contrôles de validation pour s'assurer que votre domaine est éligible à une mise à niveau. Si l'un de ces contrôles échoue, vous recevez une notification contenant les problèmes spécifiques que vous devez résoudre avant de mettre à niveau votre domaine. Pour obtenir la liste des problèmes potentiels et les étapes à suivre pour les résoudre, consultez the section called “Résolution des erreurs de validation”.

Dépannage d'une mise à niveau

Les mises à niveau sur place requièrent des domaines sains. Votre domaine peut être inéligible pour une mise à niveau ou peut échouer à effectuer une mise à niveau pour une multitude de raisons. Le tableau suivant indique les problèmes les plus courants.

Problème	Description
Plug-in optionnel non pris en charge	Lorsque vous mettez à niveau un domaine avec des plugins optionnel s, le OpenSearch Service met également automatiquement à jour les plugins. Par conséquent, la version cible de votre domaine doit également prendre en charge ces plugins facultatifs. Si le domaine possède un plug-in optionnel qui n'est pas disponible pour la version cible, la demande de mise à niveau échoue.
Trop de partitions par nœud	OpenSearch, ainsi que 7. x versions d'Elasticsearch, présentent un paramètre par défaut qui ne dépasse pas 1 000 partitions par nœud. Si un nœud de votre cluster actuel dépasse la valeur de ce paramètre, le OpenSearch Service ne vous permet pas d'effectuer la mise à niveau. Consultez the section called “Limite maximale de partitions dépassée” pour les options de dépannage.
Domaine en cours de traitement	Un changement de configuration est actuellement en cours pour le domaine. Une fois l'opération terminée, vérifiez l'éligibilité pour la mise à niveau.
Statut de cluster rouge	Un ou plusieurs index dans le cluster sont rouges. Pour obtenir les étapes de dépannage, consultez the section called “Statut de cluster rouge”.

Problème	Description
Taux d'erreur élevé	Le cluster renvoie un grand nombre d'erreurs 5xx lors de sa tentative de traitement des demandes. Ce problème est généralement dû à un trop grand nombre de demandes simultanées de lecture ou d'écriture. Envisagez de réduire le trafic vers le cluster ou de redimensionner votre domaine.
Split-Brain	Split-Brain signifie que votre cluster possède plusieurs nœuds principaux et s'est scindé en deux clusters qui ne refusionneront jamais d'eux-mêmes. Vous pouvez éviter le problème Split-Brain en utilisant le nombre recommandé de nœuds principaux dédiés. Pour obtenir de l'aide afin de résoudre le problème Split-Brain, contactez Support.
Nœud principal introuvable	OpenSearch Le service ne peut pas trouver le nœud principal du cluster. Si votre domaine utilise multi-AZ, il se peut qu'un échec de zones de disponibilité ait entraîné la perte du quorum par le cluster, empêchant ainsi ce dernier de choisir un nouveau nœud principal. Si le problème ne se résout pas automatiquement, contactez Support.
Trop de tâches en attente	Le nœud principal est soumis à une charge importante et a de nombreuses tâches en attente. Envisagez de réduire le trafic vers le cluster ou de redimensionner votre domaine.
Volume de stockage défaillant	Le volume de disque d'un ou plusieurs nœuds ne fonctionne pas correctement. Ce problème se produit souvent parallèlement à d'autres problèmes, comme un taux d'erreur élevé ou un trop grand nombre de tâches en attente. Si ce problème est isolé et ne se résout pas automatiquement, contactez Support.
Problème lié à la clé KMS	La clé KMS utilisée pour chiffrer le domaine est inaccessible ou manquante. Pour plus d'informations, consultez the section called “Surveillance des domaines qui chiffrent les données au repos”.

Problème	Description
Instantané en cours	Le domaine prend actuellement un instantané. Une fois l'instantané pris, vérifiez l'éligibilité pour la mise à niveau. Vérifiez également que vous pouvez lister les référentiels d'instantanés manuels, lister les instantan és dans ces référentiels et prendre des instantanés manuels. Si le OpenSearch Service n'est pas en mesure de vérifier si un instantané est en cours, les mises à niveau peuvent échouer.
Délai d'expiration ou échec de l'instantané	La création de l'instantané avant la mise à niveau a été trop longue ou a échoué. Vérifiez l'état du cluster et réessayez. Si le problème persiste, contactez Support.
Index incompatibles	Une ou plusieurs index sont incompatibles avec la version cible. Ce problème peut se produire si vous avez migré les index à partir d'une ancienne version d' OpenSearch Elasticsearch. Réindexez les index et réessayez.
Utilisation du disque élevée	L'utilisation du disque pour le cluster est supérieure à 90 %. Supprimez des données ou redimensionnez le domaine, puis réessayez.
Utilisation JVM élevée	La sollicitation de mémoire JVM est supérieure à 75 %. Réduisez le trafic vers le cluster ou redimensionnez le domaine, puis réessayez.
OpenSearch Problème d'alias de tableaux de bord	`.dashboards` est déjà configuré en tant qu'alias et mappé à un index incompatible, qui provient probablement d'une version antérieure de OpenSearch Dashboards. Réindexez et réessayez.
Statut de Dashboards rouge	OpenSearch Le statut du tableau de bord est rouge. Essayez d'utilise r Dashboards une fois la mise à niveau terminée. Si le statut rouge persiste, résolvez-le manuellement, puis réessayez.

Problème	Description
Compatibilité inter-clusters	Vous pouvez uniquement effectuer la mise à niveau si, après cette dernière, la compatibilité inter-clusters est maintenue entre les domaines source et destination. Au cours du processus de mise à niveau, toutes les connexions incompatibles sont identifiées. Ensuite, mettez à niveau le domaine distant ou supprimez les connexions incompatibles. Notez que si la réplication est active sur le domaine, vous ne pouvez pas la reprendre une fois la connexion supprimée.
Autre problème de OpenSearch service	Les problèmes liés OpenSearch au Service risquent d'afficher votre domaine comme étant inéligible à une mise à niveau. Si aucune des conditions précédentes ne s'applique à votre domaine et que le problème persiste pendant plus d'une journée, contactez Support.

Utilisation d'un instantané pour migrer des données

Les mises à niveau sur place constituent le moyen le plus facile, le plus rapide et le plus fiable pour mettre à niveau un domaine vers une version OpenSearch ultérieure d'Elasticsearch. Les instantanés sont une bonne option si vous devez effectuer une migration à partir d'une version d'Elasticsearch antérieure à 5.1 ou que vous voulez migrer vers un tout nouveau cluster.

Le tableau suivant montre comment utiliser des instantanés pour migrer des données vers un domaine qui utilise une autre version OpenSearch d'Elasticsearch. Pour plus d'informations sur la prise d'instantanés et leur restauration, consultez the section called “Création d'instantanés d'index”.

De la version	Vers la version	Processus de migration
OpenSearch 1.3 ou 2. x	OpenSearch 2. x	1. Examinez les modifications importantes de la version OpenSearch 2.3 pour savoir si vous devez faire des modifications aux index ou aux applications. 2. Créez un instantané manuel de la version 1.3 ou 2. domaine x. 3. Créez un 2. domaine x dont la version est supérieure à celle de votre version 1.3 ou 2 d'origine. domaine x.

De la version	Vers la version	Processus de migration

		4. Restaurez l'instantané du domaine d'origine vers le domaine 2. domaine x. Au cours de l'opération, vous devrez probablement restaurer votre `.opensearch` index sous un nouveau nom :
		
		`POST _snapshot/ <repository-name> /<snapshot-name>/_restore`
		`{`
		`  "indices": "*",`
		`  "ignore_unavailable": true,`
		`  "rename_pattern": ".opensearch",`
		`  "rename_replacement": ".backup-opensearch"`
		`}`
		
		Ensuite, vous pouvez réindexer `.backup-opensearch` sur le nouveau domaine et lui donner l'alias `.opensearch` . Notez que l'appel `_restore` REST n'inclut pas `include_global_state` car la valeur par défaut `_restore` est false. Par conséquent, le domaine de test n'inclura aucun modèle d'index et ne disposera pas de l'état complet de la sauvegarde.
		5. Si vous n'avez plus besoin de votre domaine d'origine , supprimez-le. Sinon, vous continuez à payer des frais pour le domaine.

De la version	Vers la version	Processus de migration

OpenSearch 1. x	OpenSearch 1. x	1. Créez un instantané manuel du 1. domaine x.
		2. Créez un 1. x domaine dont la version est supérieure à votre 1 d'origine. domaine x.
		3. Restaurez l'instantané du domaine d'origine vers le nouveau 1. domaine x. Au cours de l'opération, vous devrez probablement restaurer votre `.opensearch` index sous un nouveau nom :
		`POST _snapshot/ <repository-name> /<snapshot-name>/_restore`
		`{`
		`  "indices": "*",`
		`  "ignore_unavailable": true,`
		`  "rename_pattern": ".opensearch",`
		`  "rename_replacement": ".backup-opensearch"`
		`}`
		Ensuite, vous pouvez réindexer `.backup-opensearch` sur le nouveau domaine et lui donner l'alias `.opensearch` . Notez que l'appel `_restore` REST n'inclut pas `include_global_state` car la valeur par défaut `_restore` est false. Par conséquent, le domaine de test n'inclura aucun modèle d'index et ne disposera pas de l'état complet de la sauvegarde.
		4. Si vous n'avez plus besoin de votre domaine d'origine , supprimez-le. Sinon, vous continuez à payer des frais pour le domaine.

De la version	Vers la version	Processus de migration

Elasticsearch 6.x ou 7.x	OpenSearch 1. x	1. Examinez les modifications importantes de la OpenSearch version 1.0 pour savoir si vous devez faire des modifications aux index ou aux applications.
		2. Créez un instantané manuel de votre domaine Elasticsearch 7.x ou 6.x
		3. Créez un OpenSearch 1. domaine x.
		4. Restaurez l'instantané du domaine Elasticsearch vers le OpenSearch domaine. Au cours de l'opération, vous devrez probablement restaurer votre `.elasticsearch` index sous un nouveau nom :
		`POST _snapshot/ <repository-name> /<snapshot-name>/_restore`
		`{`
		`  "indices": "*",`
		`  "ignore_unavailable": true,`
		`  "rename_pattern": ".elasticsearch",`
		`  "rename_replacement": ".backup-opensearch"`
		`}`
		Ensuite, vous pouvez réindexer `.backup-opensearch` sur le nouveau domaine et lui donner l'alias `.elasticsearch` . Notez que l'appel `_restore` REST n'inclut pas `include_global_state` car la valeur par défaut `_restore` est false. Par conséquent, le domaine de test n'inclura aucun modèle d'index et ne disposera pas de l'état complet de la sauvegarde.
		5. Si vous n'avez plus besoin de votre domaine d'origine , supprimez-le. Sinon, vous continuez à payer des frais pour le domaine.

De la version	Vers la version	Processus de migration

Elasticsearch 6.x	Elasticsearch 7.x	1. Examinez les modifications importantes de la version 7.0 pour savoir si vous devez faire des modifications aux index ou aux applications.
		2. Créez un instantané manual du domaine 6.x.
		3. Créez un domaine 7.x.
		4. Restaurez l'instantané du domaine d'origine vers le domaine 7.x. Au cours de l'opération, vous devrez probablement restaurer l'index `.opensearch` sous un nouveau nom :
		`POST _snapshot/ <repository-name> /<snapshot-name>/_restore`
		`{`
		`  "indices": "*",`
		`  "ignore_unavailable": true,`
		`  "rename_pattern": ".elasticsearch",`
		`  "rename_replacement": ".backup-elasticsearch"`
		`}`
		Ensuite, vous pouvez réindexer `.backup-elasticsearch` sur le nouveau domaine et lui donner l'alias `.elasticsearch` . Notez que l'appel `_restore` REST n'inclut pas `include_global_state` car la valeur par défaut `_restore` est false. Par conséquent, le domaine de test n'inclura aucun modèle d'index et ne disposera pas de l'état complet de la sauvegarde.
		5. Si vous n'avez plus besoin de votre domaine d'origine , supprimez-le. Sinon, vous continuez à payer des frais pour le domaine.

De la version	Vers la version	Processus de migration
Elasticsearch 6.x	Elasticsearch 6.8	1. Créez un instantané manuel du domaine 6.x. 2. Créez un domaine 6.8. 3. Restaurez l'instantané du domaine d'origine vers le domaine 6.8. 4. Si vous n'avez plus besoin de votre domaine d'origine , supprimez-le. Sinon, vous continuez à payer des frais pour le domaine.
Elasticsearch 5.x	Elasticsearch 6.x	1. Passez en revue les modifications importantes pour la version 6.0 afin de voir si vous devez apporter des modifications à vos index ou à vos applications. 2. Créez un instantané manuel du domaine 5.x. 3. Créez un domaine 6.x. 4. Restaurez l'instantané du domaine d'origine vers le domaine 6.x. 5. Si vous n'avez plus besoin de votre domaine 5x, supprimez-le. Sinon, vous continuez à payer des frais pour le domaine.
Elasticsearch 5.x	Elasticsearch 5.6	1. Créez un instantané manuel du domaine 5.x. 2. Créez un domaine 5.6. 3. Restaurez l'instantané du domaine d'origine vers le domaine 5.6. 4. Si vous n'avez plus besoin de votre domaine d'origine , supprimez-le. Sinon, vous continuez à payer des frais pour le domaine.

De la version	Vers la version	Processus de migration
Elasticsearch 2.3	Elasticsearch 6.x	Les instantanés Elasticsearch 2.3 ne sont pas compatibles avec la version 6.x. Pour migrer les données directement de la version 2.3 vers la version 6.x, vous devez recréer manuellement les index dans le nouveau domaine. Vous pouvez également suivre les étapes pour migrer de la version 2.3 vers la version 5.x dans ce tableau, effectuer des opérations `_reindex` dans le nouveau domaine 5.x pour convertir les index 2.3 en index 5.x, puis suivre les étapes pour migrer de la version 5.x vers la version 6.x.
Elasticsearch 2.3	Elasticsearch 5.x	1. Examinez les modifications importantes de la version 5.0 pour savoir si vous devez faire des modifications aux index ou aux applications. 2. Créez un instantané manuel du domaine 2.3. 3. Créez un domaine 5.x. 4. Restaurez l'instantané du domaine 2.3 vers le domaine 5x. 5. Si vous n'avez plus besoin de votre domaine 2.3, supprimez-le. Sinon, vous continuez à payer des frais pour le domaine.

De la version	Vers la version	Processus de migration
Elasticsearch 1.5	Elasticsearch 5.x	Les instantanés Elasticsearch 1.5 ne sont pas compatibles avec la version 5.x. Pour migrer vos données de la version 1.5 vers la version 5.x, vous devez recréer manuellement les index dans le nouveau domaine. **Important** Les instantanés 1.5 sont compatibles avec la version 2.3, mais les domaines OpenSearch Service 2.3 ne prennent pas en charge l'`_reindex`opération. Comme vous ne pouvez pas les réindexer, les index issus d'un domaine 1.5 ne peuvent toujours pas être restaurés depuis des instantanés 2.3 vers des domaines 5.x.

De la version	Vers la version	Processus de migration
Elasticsearch 1.5	Elasticsearch 2.3	1. Utilisez le plugin de migration pour déterminer si vous pouvez directement passer à la version 2.3. Vous aurez peut-être besoin d'apporter des modifications à vos données avant de procéder à la migration. a. Dans un navigateur web, ouvrez `http://`*`domain-endpoint`*` /_plugin/migration/` . b. Choisissez Run checks now (Exécuter les vérifications maintenant). c. Vérifiez les résultats et, si nécessaire, suivez les instructions pour apporter des modifications à vos données. 2. Créez un instantané manuel du domaine 1.5. 3. Créez un domaine 2.3. 4. Restaurez l'instantané du domaine 1.5 vers le domaine 2.3. 5. Si vous n'avez plus besoin de votre domaine 1.5, supprimez-le. Sinon, vous continuez à payer des frais pour le domaine.

Création d'un point de terminaison personnalisé pour Amazon OpenSearch Service

La création d'un point de terminaison personnalisé pour votre domaine Amazon OpenSearch Service vous permettra de vous référer plus facilement à vos tableaux de OpenSearch bord OpenSearch et à vos tableaux URLs de bord. Vous pouvez inclure l'image de marque de votre entreprise ou simplement utiliser un point de easier-to-remember terminaison plus court que le point de terminaison standard.

Si vous devez passer à un nouveau domaine, il vous suffit de mettre à jour votre DNS pour qu'il pointe vers la nouvelle URL et de continuer à utiliser le même point de terminaison qu'auparavant.

Pour sécuriser les points de terminaison personnalisés, vous devez générer un certificat dans AWS Certificate Manager (ACM) ou importer un de vos propres certificats.

Points de terminaison personnalisés pour les nouveaux domaines

Pour activer un point de terminaison personnalisé pour un nouveau domaine de OpenSearch services, vous pouvez utiliser la console OpenSearch de services, la AWS CLI, ou une API de configuration.

Pour personnaliser votre point de terminaison (console)

1. Dans la console OpenSearch Service, choisissez Create domain (Créer un domaine) et donnez un nom au domaine.
2. Sous Custom endpoint (Point de terminaison personnalisé), sélectionnez Enable custom endpoint (Activer un point de terminaison personnalisé).
3. Dans le champ Nom d'hôte personnalisé, saisissez le nom d'hôte personnalisé de votre choix. Le nom d'hôte doit être un nom de domaine complet (FQDN), tel que www.votredomaine.com ou exemple.votredomaine.com.

   > **Note**
   >
   > En l'absence de certificat générique, vous devez vous procurer un nouveau certificat pour les sous-domaines de votre point de terminaison personnalisé.

4. Dans le champ Certificat AWS , choisissez le certificat SSL à utiliser pour votre domaine. Si aucun certificat n'est disponible, vous pouvez en importer un dans ACM ou utiliser ACM pour en approvisionner un. Pour plus d'informations, consultez Émission et gestion des certificats dans le Guide de l'utilisateur AWS Certificate Manager.

   > **Note**
   >
   > Le certificat doit porter le nom du point de terminaison personnalisé et se trouver sur le même compte que votre domaine OpenSearch de services. Le statut du certificat devrait être ISSUED (ÉMIS).

   - Suivez les autres étapes pour créer votre domaine et choisissez Create (Créer).
   - Une fois le traitement terminé, sélectionnez le domaine pour afficher votre point de terminaison personnalisé.

Pour utiliser l'interface CLI ou l'API de configuration, utilisez les opérations `CreateDomain` et `UpdateDomainConfig`. Pour plus d'informations, consultez le AWS CLI Command Reference et le Amazon OpenSearch Service API Reference.

Points de terminaison personnalisés pour les domaines existants

Pour ajouter un point de terminaison personnalisé à un domaine de OpenSearch services existant, choisissez Edit (Modifier) et suivez les étapes 2 à 4 ci-dessus.

Cartographie CNAME

Après avoir activé un point de terminaison personnalisé pour votre domaine de OpenSearch services, vous pouvez créer un mappage CNAME dans Amazon Route 53 (ou le fournisseur de services DNS de votre choix). La création d'un mappage CNAME vous permettra d'acheminer le trafic vers votre point de terminaison personnalisé et ses sous-domaines. Sans ce mappage, vous ne pourrez pas acheminer le trafic vers votre point de terminaison personnalisé. Pour connaître la procédure à suivre pour créer ce mappage dans Route 53, consultez Configuration du routage DNS pour un nouveau domaine et Création d'une nouvelle zone hébergée pour un sous-domaine. Pour les autres fournisseurs, consultez leur documentation.

Créez un enregistrement CNAME pointant le point de terminaison personnalisé vers le point de terminaison de domaine généré automatiquement. Si votre domaine est à double pile, vous pouvez faire pointer votre enregistrement CNAME vers l'un des deux points de terminaison générés par le service. La capacité de double pile de votre point de terminaison personnalisé dépend du point de terminaison généré par le service vers lequel vous pointez l'enregistrement CNAME. Le nom d'hôte du point de terminaison personnalisé est le nom de l'enregistrement CNAME et le nom d'hôte du point de terminaison du domaine est la valeur de l'enregistrement CNAME.

Si vous utilisez l'authentification SAML pour les OpenSearch tableaux de bord, vous devrez mettre à jour votre fournisseur d'identité avec la nouvelle URL SSO.

Pour utiliser Amazon Route 53, vous pouvez créer un type d'enregistrement alias pour pointer le point de terminaison personnalisé de votre domaine vers un point de terminaison personnalisé de votre domaine vers un point de terminaison à double pile. Pour créer un type d'enregistrement d'alias, vous devez configurer votre domaine pour utiliser le type d'adresse IP à double pile. Pour ce faire, vous pouvez utiliser l'API Route 53.

Pour créer un type d'enregistrement d'alias à l'aide de l'API Route 53, spécifiez l'alias cible de votre domaine. Vous pouvez trouver l'alias cible de votre domaine dans le champ Hosted Zone (dual stack) de la section point de terminaison personnalisé de la console de OpenSearch service ou en utilisant l'`DescribeDomain`API et en copiant la valeur du`DomainEndpointV2HostedZoneId`.

Auto-Tune pour Amazon Service OpenSearch

La fonctionnalité Auto-Tune d'Amazon OpenSearch Service utilise les métriques de performances et d'utilisation de votre OpenSearch cluster pour suggérer des modifications de configuration en lien avec la mémoire, comme la taille des files d'attente ou du cache et les paramètres de machine virtuelle Java (JVM) sur vos nœuds. Ces modifications facultatives améliorent la vitesse et la stabilité du cluster.

Certaines modifications sont déployées immédiatement, tandis que d'autres sont planifiées pendant la période creuse de votre domaine. Vous pouvez revenir aux paramètres par défaut du OpenSearch Service à tout moment. Pendant qu'Auto-Tune collecte et analyse les métriques de performances de votre domaine, vous pouvez consulter ses recommandations sur la page Notifications de la console OpenSearch Service.

Auto-Tune est disponible dans la région commerciale Régions AWS sur les domaines qui exécutent n'importe quelle OpenSearch version, ou Elasticsearch 6.7 ou version ultérieure avec un type d'instance pris en charge.

Types de modifications

Auto-Tune propose deux grandes catégories de modifications :

- Modifications sans perturbation appliquées pendant l'exécution du cluster.
- Modifications nécessitant un déploiement bleu/vert, qui s'applique pendant la période creuse du domaine.

En fonction des métriques de performances de votre domaine, Auto-Tune peut suggérer d'ajuster les paramètres suivants :

Type de modification	Catégorie	Description
Taille de la pile de la JVM	Bleu/vert	Par défaut, OpenSearch Service utilise 50 % de la RAM d'une instance pour la pile de la JVM, pour une taille maximale de 32 Gio. L'augmentation de ce pourcentage confère OpenSearch plus de mémoire, au détriment du système d'exploitation et d'autres processus. Des valeurs plus élevées peuvent réduire le nombre de pauses effectuées pour le nettoyage de la mémoire, mais augmenter la durée de ces pauses.
Paramètres de la « jeune génération » de JVM	Bleu/vert	Les paramètres de la « jeune génération » de JVM déterminent la fréquence des nettoyages mineurs de la mémoire. Des nettoyages mineurs plus fréquents peuvent réduire le nombre de nettoyages majeurs et de pauses.
Taille des files d'attente	Sans perturbation	Par défaut, la taille de la file d'attente de recherche est définie sur `1000` et celle de la file d'attente d'écriture sur `10000`. Auto-Tune met automatiquement à l'échelle les files d'attente de recherche et d'écriture si une pile supplémentaire est disponible pour traiter les demandes.
Taille du cache	Sans perturbation	Le cache des champs surveille les structures des données sur la pile, d'où l'importance de surveiller l'utilisation du cache. Auto-Tune met à l'échelle la taille du cache de données des champs pour éviter les problèmes de mémoire insuffisante et de disjoncteur de circuit. Le cache des demandes de partition est géré au niveau du nœud, et sa taille maximale par défaut correspond à 1 % de la pile. Auto-Tune met à l'échelle la taille du cache des demandes de partition pour accepter plus de demandes de recherche et d'index que ce que le cluster configuré peut gérer.
Taille des demandes	Sans perturbation	Par défaut, lorsque la taille cumulée des demandes en vol dépasse 10 % de la JVM totale (2 % pour les types d'`t2`instance et 1 %

Type de modification	Catégorie	Description
		pour`t3.small`), limite toutes OpenSearch les nouvelles `_bulk` demandes jusqu'à ce que `_search` les demandes existantes soient terminées. Auto-Tune règle automatiquement ce seuil, généralement entre 5 et 15 %, en fonction de la quantité de JVM actuellement occupée sur le système. Par exemple, si la sollicitation de la mémoire JVM est élevée, Auto-Tune peut réduire le seuil à 5 %, auquel cas vous risquez de voir davantage de rejets jusqu'à ce que le cluster se stabilise et que le seuil augmente.

Activation ou désactivation d'Auto-Tune

OpenSearch Le service active Auto-Tune par défaut sur de nouveaux domaines. Pour activer ou désactiver Auto-Tune sur des domaines existants, nous vous recommandons d'utiliser la console, qui simplifie le processus. L'activation d'Auto-Tune ne provoque pas de déploiement bleu/vert.

Actuellement, vous ne pouvez pas activer ou désactiver Auto-Tune avec AWS CloudFormation.

console

Pour activer Auto-Tune sur un domaine existant

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le panneau de navigation, sous Domains (Domaines), choisissez le nom de domaine pour ouvrir la configuration du cluster.
3. Choisissez Activer si Auto-Tune n'est pas déjà activé.
4. Vous pouvez éventuellement sélectionner Fenêtre creuse pour planifier les optimisations qui nécessitent un déploiement bleu/vert pendant la période creuse configurée pour le domaine. Pour de plus amples informations, veuillez consulter the section called “Optimisation”.
5. Sélectionnez Save Changes (Enregistrer les modifications).

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour activer Auto-Tune à l'aide du AWS CLI, envoyez une UpdateDomainConfigdemande :

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --auto-tune-options DesiredState=ENABLED
```

Optimisation

Avant le 16 février 2023, Auto-Tune utilisait des fenêtres de maintenance pour planifier les modifications nécessitant un déploiement bleu/vert. Les fenêtres de maintenance sont désormais déconseillées au profit de la période creuse, qui correspond à une période quotidienne de 10 heures pendant laquelle votre domaine connaît généralement un faible trafic. Vous pouvez modifier l'heure de début par défaut pour la fenêtre hors pointe, mais vous ne pouvez pas en modifier la durée.

Tous les domaines pour lesquels les fenêtres de maintenance Auto-Tune étaient activées avant l'introduction des périodes creuses le 16 février 2023 peuvent continuer à utiliser les fenêtres de maintenance existantes sans interruption. Cependant, nous vous recommandons de migrer vos domaines existants afin d'utiliser plutôt la période creuse pour la maintenance des domaines. Pour obtenir des instructions, veuillez consulter the section called “Migration depuis les fenêtres de maintenance Auto-Tune”.

console

Pour planifier des actions Auto-Tune en dehors des heures de pointe

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le panneau de navigation, sous Domains (Domaines), choisissez le nom de domaine pour ouvrir la configuration du cluster.
3. Accédez à l'onglet Auto-Tune et choisissez Modifier.
4. Choisissez Activer si Auto-Tune n'est pas déjà activé.
5. Sous Planifier les optimisations pendant les périodes creuses, sélectionnez Fenêtre creuse.
6. Sélectionnez Enregistrer les modifications.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour configurer votre domaine afin de planifier des actions Auto-Tune pendant la période creuse configurée, incluez `UseOffPeakWindow` dans la UpdateDomainConfigdemande :

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --auto-tune-options
 DesiredState=ENABLED,UseOffPeakWindow=true,MaintenanceSchedules=null
```

Auto-Tune

Vous pouvez surveiller les statistiques d'Auto-Tune dans. Amazon CloudWatch Pour accéder à une liste complète des métriques , veuillez consulter the section called “Auto-Tune”.

OpenSearch Le service envoie des événements Auto-Tune à Amazon EventBridge. Vous pouvez l'utiliser EventBridge pour configurer des règles qui envoient un e-mail ou effectuent une action spécifique lorsqu'un événement est reçu. Pour afficher le format de chaque événement Auto-Tune envoyé à EventBridge, veuillez consulter the section called “Événements Auto-Tune” la rubrique.

Balisage des domaines Amazon OpenSearch Service

Les balises vous permettent d'affecter des informations arbitraires à un domaine Amazon OpenSearch Service afin d'effectuer un classement et un filtrage sur ces informations. Une balise est une paire nom-valeur que vous définissez et associez à un domaine OpenSearch Service. Vous pouvez utiliser ces balises pour suivre les coûts en regroupant les dépenses pour des ressources balisées de la même façon. AWS n'applique aucune signification sémantique à vos balises. Les balises sont interprétées de façon stricte, en tant que chaîne de caractères. Toutes les balises comprennent les éléments suivants :

Élément de balise	Description	Obligatoire
Clé de balise	il s'agit du nom de la balise. La clé doit être propre au domaine de OpenSearch service auquel elle est attachée. Pour obtenir une liste des restrictions de base sur les valeurs et les clés de balise, consultez User-Defined Tag Restrictions.	Oui

Élément de balise	Description	Obligatoire
Valeur de balise	La valeur de balise correspond à la valeur de chaîne de la balise. Les valeurs de balise peuvent être `null` et ne doivent pas nécessair ement être uniques dans un ensemble de balises. Par exemple, vous pouvez avoir une paire clé-valeur dans un ensemble de balises. project/Trinity and cost-center/Trinity Pour obtenir une liste des restrictions de base sur les valeurs et les clés de balise, consultez User-Defined Tag Restrictions.	Non

Chaque domaine OpenSearch de service possède un ensemble de balises qui contient toutes les balises affectées à ce domaine OpenSearch de service. AWS n'attribue pas automatiquement de balises aux domaines OpenSearch de service. Un ensemble de balises peut contenir entre 0 et 50 balises. Si vous ajoutez une balise à un domaine doté de la même clé qu'une balise existante, la nouvelle valeur remplace l'ancienne.

Exemples de balisage

Vous pouvez utiliser une clé pour définir une catégorie, et la valeur de balise peut être un élément de cette catégorie. Par exemple, vous pouvez définir une clé de balise `project` et une valeur de balise`Salix`, en indiquant que la OpenSearch ressource Service est affectée au projet Salix. Vous pouvez également utiliser des balises pour désigner des domaines de OpenSearch service destinés aux tests ou à la production en utilisant une clé telle que `environment=test` ou`environment=production`. Essayez d'utiliser un ensemble cohérent de clés de balise pour faciliter le suivi des métadonnées associées aux domaines OpenSearch Service.

Vous pouvez également utiliser des balises pour organiser votre AWS facture afin de refléter votre propre structure de coût. Pour ce faire, inscrivez-vous pour obtenir votre Compte AWS facture avec les valeurs de clé de balise incluses. Ensuite, organisez vos informations de facturation en fonction des ressources possédant les mêmes valeurs de clé de balise pour voir le coût de vos ressources combinées. Par exemple, vous pouvez marquer plusieurs domaines de OpenSearch service avec des paires clé-valeur, puis organisez vos informations de facturation pour voir le coût total pour chaque domaine sur plusieurs services. Pour plus d'informations, consultez Utilisation des balises d'allocation des coûts dans la documentation Gestion de la facturation et des coûts AWS .

> **Note**
>
> Les balises sont mises en cache à des fins d'autorisation. Pour cette raison, les ajouts et les mises à jour des balises sur des domaines OpenSearch Service peuvent prendre plusieurs minutes avant d'être disponibles.

Balisage des domaines (console)

La console constitue le moyen le plus simple de baliser un domaine.

Pour créer une identification (console)

1. Connectez-vous à https://aws.amazon.com puis choisissez Sign In to the Console (Connectez-vous à la console).
2. Sous Analytics, sélectionnez Amazon OpenSearch Service.
3. Sélectionnez le domaine auquel vous voulez ajouter des balises et accédez à l'onglet Tags (Balises).
4. Choisissez Manage (Gérer) et Add new tag (Ajouter une nouvelle balise).
5. Saisissez une clé de balise et une valeur de balise facultative.
6. Choisissez Save (Enregistrer).

Pour supprimer une balise, suivez les mêmes étapes et choisissez Remove (Supprimer) sur la page Manage tags (Gérer les balises).

Pour plus d'informations sur l'utilisation de la console avec des identifications, veuillez consulter Éditeur d'identification dans le Guide de démarrage de la console de gestion AWS .

Marquage de domaines ()AWS CLI

Vous pouvez créer des balises de ressources à l'aide AWS CLI de la --add-tags commande with.

Syntaxe

```
add-tags --arn=<domain_arn> --tag-list Key=<key>,Value=<value>
```

Paramètre	Description
`--arn`	Amazon Resource Name pour le domaine OpenSearch Service auquel la balise est attachée.
`--tag-list`	Ensemble de paires clé-valeur séparées par des espaces dans le format suivant : `Key=<key>,Value=<value>`

Exemple

L'exemple suivant crée deux balises pour le domaine journaux :

```
aws opensearch add-tags --arn arn:aws:es:us-east-1:379931976431:domain/logs --tag-list
 Key=service,Value=OpenSearch Key=instances,Value=m3.2xlarge
```

Vous pouvez supprimer des balises d'un domaine OpenSearch Service à l'aide de la --remove-tags commande.

Syntaxe

```
remove-tags --arn=<domain_arn> --tag-keys Key=<key>,Value=<value>
```

Paramètre	Description
`--arn`	Amazon Resource Name (ARN) pour le domaine de OpenSearch service auquel la balise est attachée.
`--tag-keys`	Ensemble de paires clé-valeur séparées par des espaces que vous souhaitez supprimer du domaine Service. OpenSearch

Exemple

L'exemple suivant supprime les deux balises du domaine journaux qui ont été créées dans l'exemple précédent :

```
aws opensearch remove-tags --arn arn:aws:es:us-east-1:379931976431:domain/logs --tag-
keys service instances
```

Vous pouvez afficher les balises existantes pour un domaine OpenSearch Service avec la --list-tags commande :

Syntaxe

```
list-tags --arn=<domain_arn>
```

Paramètre	Description
`--arn`	Amazon Resource Name (ARN) pour le domaine de OpenSearch service auquel les balises sont attachées.

Exemple

L'exemple suivant répertorie toutes les balises de ressources pour le domaine journaux :

```
aws opensearch list-tags --arn arn:aws:es:us-east-1:379931976431:domain/logs
```

Marquage de domaines ()AWS SDKs

AWS SDKs (sauf Android et iOS SDKs) prennent en charge toutes les actions définies dans le Amazon OpenSearch Service API Reference, y compris les `RemoveTags` opérations `AddTagsListTags`, et. Pour plus d'informations sur l'installation et l'utilisation de AWS SDKs, consultez Kits de développement AWS logiciel.

Python

Cet exemple utilise le client Python de OpenSearchServicebas niveau de l'AWS SDK pour Python (Boto) pour ajouter une identification à un domaine, lister l'identification attachée au domaine et supprimer une identification du domaine. Vous devez fournir des valeurs pour `DOMAIN_ARN`, `TAG_KEY` et `TAG_VALUE`.

```
import boto3
from botocore.config import Config  # import configuration

DOMAIN_ARN = ''  # ARN for the domain. i.e "arn:aws:es:us-east-1:123456789012:domain/
my-domain
TAG_KEY = ''  # The name of the tag key. i.e 'Smileyface'
TAG_VALUE = ''  # The value assigned to the tag. i.e 'Practicetag'
```

```
# defines the configurations parameters such as region

my_config = Config(region_name='us-east-1')
client = boto3.client('opensearch', config=my_config)


# defines the client variable

def addTags():
    """Adds tags to the domain"""

    response = client.add_tags(ARN=DOMAIN_ARN,
                               TagList=[{'Key': TAG_KEY,
                                         'Value': TAG_VALUE}])

    print(response)


def listTags():
    """List tags that have been added to the domain"""

    response = client.list_tags(ARN=DOMAIN_ARN)
    print(response)


def removeTags():
    """Remove tags that have been added to the domain"""

    response = client.remove_tags(ARN=DOMAIN_ARN, TagKeys=[TAG_KEY])

    print('Tag removed')
    return response
```

Exécution d'actions administratives sur les clés OpenSearch Amazon EFS

Amazon OpenSearch Service propose plusieurs options administratives qui fournissent un contrôle granulaire si vous devez résoudre des problèmes liés à votre domaine. Ces options incluent la possibilité de redémarrer le OpenSearch processus sur un nœud de données et la possibilité de redémarrer un nœud de données.

OpenSearch Le service surveille les paramètres de santé des nœuds et, en cas d'anomalies, prend des mesures correctives pour maintenir la stabilité des domaines. Grâce aux options administratives permettant de redémarrer le OpenSearch processus sur un nœud et de redémarrer un nœud lui-même, vous pouvez contrôler certaines de ces mesures d'atténuation.

Vous pouvez utiliser le AWS Management Console AWS CLI, ou le AWS SDK pour effectuer ces actions. Les sections suivantes expliquent comment effectuer ces actions avec la console.

Redémarrer le OpenSearch processus sur un nœud dans Amazon Service OpenSearch

Pour redémarrer le OpenSearch processus sur un nœud

1. Accédez à la console de OpenSearch service à l'adressehttps://console.aws.amazon.com/aos/.
2. Dans le volet de navigation de gauche, choisissez Domains (Domaines). Choisissez le nom du domaine que vous souhaitez transférer vers un autre.
3. Une fois la page des détails du domaine ouverte, accédez à l'onglet État de l'instance.
4. Cliquez sur le bouton situé en regard du nœud géré dont vous souhaitez modifier l'enregistrement.
5. Sélectionnez le menu déroulant Actions et choisissez Redémarrer le processus OpenSearch / Elasticsearch.
6. Choisissez Confirmer sur le modal.
7. Pour connaître l'état de l'action que vous avez initiée, sélectionnez le nom du nœud. Une fois la page de détails du nœud ouverte, choisissez l'onglet Événements sous le nom du nœud pour voir la liste des événements associés à ce nœud.

Redémarrage d'un nœud de données dans Amazon Service OpenSearch

Pour redémarrer un nœud de données

1. Accédez à la console de OpenSearch service à l'adressehttps://console.aws.amazon.com/aos/.
2. Dans le volet de navigation de gauche, choisissez Domains (Domaines). Choisissez le nom du domaine que vous souhaitez transférer vers un autre.
3. Une fois la page des détails du domaine ouverte, accédez à l'onglet État de l'instance.
4. Cliquez sur le bouton situé en regard du nœud géré dont vous souhaitez modifier l'enregistrement.

5. Sélectionnez le menu déroulant Actions, puis choisissez Redémarrer le nœud.
6. Choisissez Confirmer sur le modal.
7. Pour connaître l'état de l'action que vous avez initiée, sélectionnez le nom du nœud. Une fois la page de détails du nœud ouverte, choisissez l'onglet Événements sous le nom du nœud pour voir la liste des événements associés à ce nœud.

Redémarrer le processus OpenSearch des tableaux de bord sur un nœud dans Amazon Service OpenSearch

Vous pouvez redémarrer le processus OpenSearch des tableaux de bord (anciennement Kibana) pour résoudre des problèmes tels qu'une interface figée, des échecs de chargement ou des visualisations qui ne répondent pas. L'option permettant de redémarrer OpenSearch les tableaux de bord n'est disponible que pour le nœud qui exécute activement le processus des tableaux de bord. Dans la plupart des domaines OpenSearch de service, ce processus s'exécute sur des nœuds de coordination dédiés, et non sur des nœuds de données. Par conséquent, lorsque vous ouvrez le menu déroulant Actions dans la console, l'option apparaît généralement uniquement pour les nœuds coordinateurs. Pour de plus amples informations, veuillez consulter the section called “Nœuds de coordination dédiés”.

Ce comportement dépend de la configuration de votre domaine.

- Domaines dotés de nœuds de coordination dédiés : le processus des tableaux de bord s'exécute exclusivement sur ces nœuds, et seuls ceux-ci affichent l'option de redémarrage.
- Domaines sans nœuds de coordination dédiés : dans les configurations plus simples ou plus anciennes, les tableaux de bord peuvent plutôt s'exécuter sur un nœud de données, et l'option de redémarrage y apparaît.
- Nœuds principaux : ces nœuds sont uniquement destinés à gérer l'état du cluster et les élections. Ils n'exécutent pas de tableaux de bord et n'affichent jamais l'option de redémarrage.

Pour déterminer quel nœud exécute le processus des tableaux de bord, accédez à la section Configuration du cluster de votre domaine et passez en revue les rôles des nœuds. L'option de redémarrage n'est disponible que pour le nœud hébergeant le processus Dashboards.

Pour redémarrer le processus des tableaux de bord sur un nœud

1. Accédez à la console de OpenSearch service à l'adressehttps://console.aws.amazon.com/aos/.

2. Dans le volet de navigation de gauche, choisissez Domains (Domaines). Choisissez le nom du domaine que vous souhaitez transférer vers un autre.
3. Une fois la page des détails du domaine ouverte, accédez à l'onglet État de l'instance.
4. Dans la section relative aux nœuds qui exécutent le processus Dashboards, sélectionnez le bouton situé à côté du nœud sur lequel vous souhaitez redémarrer le processus.
5. Sélectionnez le menu déroulant Actions, puis choisissez Redémarrer le tableau de bord/le processus Kibana.
6. Choisissez Confirmer sur le modal.
7. Pour connaître l'état de l'action que vous avez initiée, sélectionnez le nom du nœud. Une fois la page de détails du nœud ouverte, choisissez l'onglet Événements sous le nom du nœud pour voir la liste des événements associés à ce nœud.

Limites

Les limitations suivantes sont les suivantes :

- Les options d'administration sont prises en charge sur les versions 7.x et supérieures d'Elasticsearch.
- Les options d'administration ne prennent pas en charge les domaines dans lesquels le mode multi-AZ est activé avec le mode veille activé.
- Le redémarrage du processus Elasticsearch OpenSearch et le redémarrage du nœud de données sont pris en charge sur les domaines comportant au moins trois nœuds de données.
- La prise en charge des tableaux de bord et des processus Kibana est prise en charge sur les domaines comportant au moins deux nœuds de données.
- Pour redémarrer le OpenSearch processus sur un nœud ou redémarrer un nœud, le domaine ne doit pas être en rouge et des répliques doivent être configurées pour tous les index.

Utilisation des requêtes directes Amazon OpenSearch Service

Utilisez la requête directe Amazon OpenSearch Service pour analyser les données dans Amazon CloudWatch Logs, Amazon S3 et Amazon Security Lake sans créer de pipelines d'ingestion. Cette intégration zéro ETL vous permet d'interroger les données sur place à l'aide de OpenSearch SQL ou PPL, et de les explorer dans Discover.

Pour commencer, configurez votre source de données dans la console OpenSearch de service. Pour Amazon S3, utilisez des connexions de domaine et créez des tables avec SQL dans Query Workbench. CloudWatch Logs et Security Lake utilisent des sources et des AWS Glue Data Catalog tables préconfigurées.

Tarification des requêtes directes

Lorsque vous utilisez des requêtes directes via le OpenSearch service, vous devez payer des frais distincts pour le OpenSearch service et les ressources utilisées pour traiter et stocker vos données sur Amazon S3, Amazon CloudWatch Logs et Amazon Security Lake. Lorsque vous exécutez des requêtes directes, vous verrez les frais d'unités de OpenSearch calcul (OCUs) par heure, indiqués comme type d'utilisation de l' DirectQuery OCU sur votre facture.

Les requêtes directes sont de deux types : les requêtes de vue interactives et les requêtes de vue indexées.

- Les requêtes interactives sont utilisées pour renseigner le sélecteur de données et effectuer des analyses sur vos données dans S3, CloudWatch Logs ou Security Lake.

  Pour les requêtes directes Amazon S3, lorsque vous exécutez une nouvelle requête depuis Discover, le OpenSearch service démarre une nouvelle session d'une durée minimale de trois minutes. OpenSearch Le service maintient cette session active pour garantir que les requêtes suivantes s'exécutent rapidement.

  Pour les requêtes CloudWatch Logs et Security Lake, le OpenSearch service gère chaque requête avec une tâche préchauffée distincte, sans maintenir de session prolongée.
- Les requêtes de vues indexées utilisent le calcul pour maintenir les vues indexées dans OpenSearch Service. Ces requêtes prennent généralement plus de temps car elles ingèrent une quantité variable de données dans un index nommé. En indexant les données, vous pouvez

accélérer l'exécution des futures requêtes interactives ou débloquer des fonctionnalités d'analyse avancées telles que des tableaux de bord ou des alertes, qui nécessitent un index pour les référencer.

Pour les sources de données Amazon S3, les données indexées sont stockées dans un domaine en fonction du type d'instance acheté. Pour les sources de données connectées à CloudWatch Logs et Security Lake, les données indexées sont stockées dans une collection OpenSearch sans serveur où vous êtes facturé pour les données indexées (IndexingOCU), les données recherchées (SearchOCU) et les données stockées en Go.

Pour plus d'informations, consultez les sections Direct Query et Serverless d'Amazon OpenSearch Service Pricing.

Limites relatives aux requêtes directes

Limitations générales

Les restrictions suivantes s'appliquent aux requêtes directes du OpenSearch Service.

- Certains types de données ne sont pas pris en charge. Les types de données pris en charge sont limités à Parquet, CSV et JSON.
- Si la structure de vos données change au fil du temps, vous devrez mettre à jour vos vues indexées ou vos out-of-the-box intégrations pour tenir compte des modifications de structure des données.
- AWS CloudFormation les modèles ne sont pas encore pris en charge.
- OpenSearch Les instructions SQL et OpenSearch PPL présentent des limites différentes lors de l'utilisation d' OpenSearch index par rapport à l'utilisation de requêtes directes. La requête directe prend en charge les commandes avancées telles que JOINs les sous-requêtes et les recherches, tandis que la prise en charge de ces commandes sur les OpenSearch index est limitée, voire inexistante. Pour de plus amples informations, veuillez consulter the section called “Commandes SQL et PPL prises en charge”.

Limitations pour Amazon S3

Si vous interrogez directement des données dans Amazon S3, les restrictions supplémentaires suivantes s'appliquent :

- La requête directe pour S3 n'est disponible que sur les domaines de OpenSearch service exécutant OpenSearch la version 2.13 ou ultérieure et nécessite un accès à AWS Glue Data Catalog. AWS Glue Data Catalog Les tables existantes doivent être recréées à l'aide de SQL dans OpenSearch Query Workbench.
- La requête directe pour S3 nécessite que vous spécifiiez un compartiment de point de contrôle sur Amazon S3. Ce compartiment conserve l'état de vos vues indexées, y compris l'heure de la dernière actualisation et les dernières données ingérées.
- Votre OpenSearch domaine AWS Glue Data Catalog doit appartenir au même Compte AWS. Votre compartiment S3 peut se trouver dans un autre compte (une condition doit être ajoutée à votre politique IAM), mais il doit se trouver dans le même compte Région AWS que votre domaine.
- OpenSearch Les requêtes directes avec S3 ne prennent en charge que les tables Spark générées à partir de Query Workbench. Les tables générées dans AWS Glue Data Catalog ou Athena ne sont pas prises en charge par le streaming Spark, qui est nécessaire pour conserver les vues indexées.
- OpenSearch les types d'instance ont des limites de charge utile en réseau de 10 MiB ou 100 MiB, selon le type d'instance spécifique que vous choisissez.

Limitations pour Amazon CloudWatch Logs

Si vous interrogez directement des données dans CloudWatch Logs, les restrictions supplémentaires suivantes s'appliquent :

- L'intégration directe des requêtes avec CloudWatch Logs n'est disponible que sur les collections de OpenSearch services et sur OpenSearch l'interface utilisateur.
- OpenSearch Les collections sans serveur sont soumises à des limites de charge utile en réseau de 100 MiB.
- CloudWatch Logs prend en charge les intégrations de VPC Flow et AWS WAF de tableau de bord installées depuis la console. CloudTrail

Limitations d'Amazon Security Lake

Si vous interrogez directement des données dans Security Lake, les restrictions supplémentaires suivantes s'appliquent :

- L'intégration directe des requêtes à Security Lake n'est disponible que sur les collections OpenSearch de services et sur OpenSearch l'interface utilisateur.

- OpenSearch Les collections sans serveur sont soumises à des limites de charge utile en réseau de 100 MiB.
- La gestion des tables pour Security Lake est effectuée dans Lake Formation.
- Security Lake prend uniquement en charge les vues matérialisées sous forme de vues indexées. Les index de couverture ne sont pas pris en charge.

Recommandations pour l'utilisation des requêtes directes dans Amazon OpenSearch Service

Cette page fournit des recommandations sur l'utilisation des requêtes directes Amazon OpenSearch Service pour analyser les données provenant de CloudWatch Logs, d'Amazon S3 et d'Amazon Security Lake. Ces meilleures pratiques vous aident à optimiser les performances et à garantir l'efficacité des requêtes sans devoir ingérer ou dupliquer les données.

Rubriques

- Recommandations générales
- Recommandations d'Amazon S3
- CloudWatch Recommandations relatives aux journaux
- Recommandations de Security Lake

Recommandations générales

Nous vous recommandons de procéder comme suit lorsque vous utilisez la requête directe :

- Utilisez cette `COALESCE SQL` fonction pour gérer les colonnes manquantes et vous assurer que les résultats sont renvoyés.

- Limitez vos requêtes pour vous assurer de ne pas récupérer trop de données.
- Si vous prévoyez d'analyser plusieurs fois le même jeu de données, créez une vue indexée pour intégrer et indexer complètement les données OpenSearch et les déposer une fois l'analyse terminée.
- Supprimez les tâches d'accélération et les index lorsqu'ils ne sont plus nécessaires.
- Les requêtes contenant des noms de champs identiques mais ne différant que par des majuscules (tels que `field1` et`FIELD1`) ne sont pas prises en charge.

Par exemple, les requêtes suivantes ne sont pas prises en charge :

```
Select AWSAccountId, AwsAccountId from LogGroup
Select a.@LogStream, b.@logStream from Table A INNER Join Table B ona.id = b.id
```

Cependant, la requête suivante est prise en charge car le nom du champ (@logStream) est identique dans les deux groupes de journaux :

```
Select a.@logStream, b.@logStream from Table A INNER Join Table B on a.id = b.id
```

- Les fonctions et expressions doivent agir sur les noms de champs et faire partie d'une `SELECT` instruction avec un groupe de journaux spécifié dans la `FROM` clause.

  Par exemple, cette requête n'est pas prise en charge :

  ```
  SELECT cos(10) FROM LogGroup
  ```

  Cette requête est prise en charge :

  ```
  SELECT cos(field1) FROM LogGroup
  ```

Recommandations d'Amazon S3

Si vous utilisez Amazon OpenSearch Service pour diriger les données de requête dans Amazon S3, nous vous recommandons également ce qui suit :

- Ingérez des données dans Amazon S3 en utilisant des formats de partition tels que l'année, le mois, le jour et l'heure pour accélérer les requêtes.
- Lorsque vous créez des index de saut, utilisez des filtres Bloom pour les champs présentant une cardinalité élevée et des min/max index pour les champs contenant de grandes plages de valeurs. Pour les champs à cardinalité élevée, envisagez d'utiliser une approche basée sur les valeurs afin d'améliorer l'efficacité des requêtes.
- Utilisez la gestion de l'état des index pour conserver le stockage des vues matérialisées et des index de couverture.

CloudWatch Recommandations relatives aux journaux

Si vous utilisez Amazon OpenSearch Service pour envoyer des données de requête dans CloudWatch Logs, nous vous recommandons également ce qui suit :

- Lorsque vous recherchez plusieurs groupes de journaux dans une seule requête, utilisez la syntaxe appropriée. Pour de plus amples informations, veuillez consulter the section called “Fonctions de groupe multilog”.
- Lorsque vous utilisez des commandes SQL ou PPL, entourez certains champs de backticks pour les interroger correctement. Les champs contenant des caractères spéciaux (non alphabétiques et non numériques) doivent être cochés. Par exemple, joignez `Operation.Export,` et `@message` insérez des `Test::Field` backticks. Il n'est pas nécessaire de placer les colonnes avec des noms purement alphabétiques en backticks.

  Exemple de requête avec des champs simples :

  ```
  SELECT SessionToken, Operation, StartTime  FROM `LogGroup-A`
  LIMIT 1000;
  ```

  Requête similaire avec backticks ajoutés :

  ```
  SELECT `@SessionToken`, `@Operation`, `@StartTime` FROM `LogGroup-A`
  LIMIT 1000;
  ```

Recommandations de Security Lake

Si vous utilisez Amazon OpenSearch Service pour envoyer des données de requête dans Security Lake, nous vous recommandons également ce qui suit :

- Vérifiez l'état de votre Security Lake et assurez-vous qu'il fonctionne correctement sans aucun problème. Pour connaître les étapes de résolution des problèmes détaillées, consultez la section Résolution des problèmes liés à l'état du lac de données dans le guide de l'utilisateur d'Amazon Security Lake.
- Vérifiez l'accès à votre requête :
  - Si vous interrogez Security Lake à partir d'un compte différent du compte d'administrateur délégué de Security Lake, configurez un abonné avec accès aux requêtes dans Security Lake.

  - Si vous interrogez Security Lake depuis le même compte, vérifiez s'il y a des messages dans Security Lake concernant l'enregistrement de vos compartiments S3 gérés auprès de ce compte. LakeFormation
- Explorez les modèles de requêtes et les tableaux de bord prédéfinis pour démarrer votre analyse.
- Familiarisez-vous avec Open Cybersecurity Schema Framework (OCSF) et Security Lake :
  - Consultez les exemples de mappage de schéma pour les AWS sources du référentiel OCSF GitHub
  - Apprenez à interroger Security Lake de manière efficace en consultant les requêtes Security Lake pour la version AWS source 2 (OCSF 1.1.0)
  - Améliorez les performances des requêtes en utilisant des partitions : `accountidregion`, et `time_dt`
- Familiarisez-vous avec la syntaxe SQL, prise en charge par Security Lake pour les requêtes. Pour de plus amples informations, veuillez consulter the section called “Commandes SQL prises en charge”.

Quotas de requêtes directes

Votre compte possède les quotas suivants relatifs aux requêtes directes du OpenSearch Service.

Quotas pour Amazon S3

Chaque fois que vous lancez une requête vers une source de données Amazon S3, OpenSearch Service ouvre une session et la maintient active pendant au moins trois minutes. Cela réduit la latence des requêtes en supprimant le temps de démarrage des sessions lors des requêtes suivantes.

Description	Maximum	Peut annuler
Connexions par domaine	10	Oui
Sources de données par domaine	20	Oui
Index par domaine	5	Oui

Description	Maximum	Peut annuler
Sessions simultanées par source de données	10	Oui
OCU maximum par requête	60	Oui
Durée maximale d'exécution des requêtes (minutes)	30	Oui
Maximum OCUs par accélérat ion	20	Oui
Stockage éphémère maximal	20	Oui

Quotas pour les CloudWatch journaux

> **Note**
>
> Si vous souhaitez effectuer des requêtes directes à l'aide de CloudWatch Logs Insights, assurez-vous de vous référer àthe section called “CloudWatch Informations sur les journaux”.

Description	Valeur	Limite souple ?	Remarques
Limite TPS au niveau du compte pour les requêtes directes APIs	3 TPS	Oui	
Nombre maximum de sources de données	20	Oui	La limite est par Compte AWS.
Nombre maximum d'index ou de vues matérialisées automatiq uement actualisés	30	Oui	La limite est fixée par source de données.

Description	Valeur	Limite souple ?	Remarques
Nombre maximal de requêtes simultanées	15	Oui	La limite s'applique aux requêtes en attente ou en cours d'exécution. Inclut des requêtes interactives (par exemple, des commandes de récupérat ion de données comme`SELECT`) et des requêtes d'index (par exemple, des opérations comme`CREATE/ALTER/DROP`).
Nombre maximal d'OCU simultanés par requête	512	Oui	OpenSearch Unités de calcul (OCU). Limite basée sur 15 exécuteurs et 1 pilote, chacun doté de 16 vCPU et de 32 Go de mémoire. Représente la puissance de traitement simultanée.
Durée maximale d'exécution des requêtes en minutes	60	Non	La limite s'applique aux requêtes OpenSearch PPL/SQL dans CloudWatch Logs Insights.
Période de purge des requêtes périmées IDs	90 jours	Oui	Il s'agit de la période après laquelle le OpenSearc h Service purge les métadonnées des requêtes pour les anciennes entrées. Par exemple, appel GetDirectQuery ou GetDirectQueryResult échec pour des requêtes datant de plus de 90 jours.

Quotas pour Security Lake

Description	Valeur	Limite souple ?	Remarques
Limite TPS au niveau du compte pour les requêtes directes APIs	3 TPS	Oui	

Description	Valeur	Limite souple ?	Remarques
Nombre maximum de sources de données	20	Oui	La limite est par Compte AWS.
Nombre maximum d'index ou de vues matérialisées automatiquement actualisés	30	Oui	La limite s'applique par source de données. Inclut uniquement les indices et les vues matérialisées (MVs) dont l'actualisation automatique est définie sur true.
Nombre maximal de requêtes simultanées	30	Oui	La limite s'applique aux requêtes en attente ou en cours d'exécution. Inclut des requêtes interactives (par exemple, des commandes de récupération de données comme`SELECT`) et des requêtes d'index (par exemple, des opérations comme`CREATE/ALTER/DROP`).
Nombre maximal d'OCU simultanés par requête	512	Oui	OpenSearch Unités de calcul (OCU). Limite basée sur 15 exécuteurs et 1 pilote, chacun doté de 16 vCPU et de 32 Go de mémoire. Représente la puissance de traitement simultanée.
Durée maximale d'exécution des requêtes en minutes	30	Non	S'applique uniquement aux requêtes interactives (par exemple, les commandes de récupération de données telles que`SELECT`). Pour les `REFRESH` requêtes, la limite est de 6 heures.
Période de purge des requêtes périmées IDs	90 jours	Oui	Il s'agit de la période après laquelle le OpenSearch Service purge les métadonnées des requêtes pour les anciennes entrées. Par exemple, appel GetDirectQuery ou GetDirectQueryResult échec pour des requêtes datant de plus de 90 jours.

Soutenu Régions AWS

Régions AWS Les éléments suivants sont pris en charge pour les requêtes directes de OpenSearch service dans Amazon S3, CloudWatch Logs et Security Lake :

Disponible Régions AWS pour Amazon S3

- Asie-Pacifique (Hong Kong)
- Asie-Pacifique (Mumbai)
- Asie-Pacifique (Séoul)
- Asie-Pacifique (Singapour)
- Asie-Pacifique (Sydney)
- Asie-Pacifique (Tokyo)
- Canada (Centre)
- Europe (Francfort)
- Europe (Irlande)
- Europe (Stockholm)
- USA Est (Virginie du Nord)
- USA Est (Ohio)
- USA Ouest (Oregon)

Disponible Régions AWS pour les CloudWatch journaux

- Asie-Pacifique (Mumbai)
- Asie-Pacifique (Hong Kong)
- Asie-Pacifique (Singapour)
- Asie-Pacifique (Sydney)
- Asie-Pacifique (Tokyo)
- Canada (Centre)
- Europe (Francfort)
- Europe (Irlande)

- Europe (Stockholm)
- USA Est (Virginie du Nord)
- USA Est (Ohio)
- USA Ouest (Oregon)
- Europe (Paris)
- Europe (Londres)
- Amérique du Sud (Sao Paulo)

Disponible Régions AWS pour Security Lake

- Asie-Pacifique (Mumbai)
- Asie-Pacifique (Singapour)
- Asie-Pacifique (Sydney)
- Asie-Pacifique (Tokyo)
- Canada (Centre)
- Europe (Francfort)
- Europe (Irlande)
- Europe (Stockholm)
- USA Est (Virginie du Nord)
- USA Est (Ohio)
- USA Ouest (Oregon)
- Europe (Paris)
- Europe (Londres)
- Amérique du Sud (Sao Paulo)

Interrogation directe des données Amazon S3 dans Service OpenSearch

Cette section explique le processus de création et de configuration d'une intégration de source de données dans Amazon OpenSearch Service, vous permettant ainsi d'interroger et d'analyser efficacement vos données stockées dans Amazon S3.

Dans les pages suivantes, vous apprendrez à configurer une source de données à requête directe Amazon S3, à définir les prérequis nécessaires et à suivre les step-by-step procédures utilisant à la fois l'API et l' OpenSearch API de AWS Management Console service. Il couvre également les prochaines étapes importantes, notamment le mappage AWS Glue Data Catalog des rôles et la configuration des contrôles d'accès dans les OpenSearch tableaux de bord.

Rubriques

- Création d'une intégration de source de données Amazon S3 dans OpenSearch Service
- Configuration et interrogation d'une source de données S3 dans OpenSearch les tableaux de bord

Création d'une intégration de source de données Amazon S3 dans OpenSearch Service

Vous pouvez créer une nouvelle source de données à requête directe Amazon S3 pour OpenSearch Service par le biais de l'API AWS Management Console ou de l'API. Chaque nouvelle source de données utilise le AWS Glue Data Catalog pour gérer les tables qui représentent les compartiments Amazon S3.

Rubriques

- Prérequis
- Procédure
- Étapes suivantes
- Cartographier le AWS Glue Data Catalog rôle
- Ressources supplémentaires

Prérequis

Avant de commencer, assurez-vous d'avoir pris connaissance de la documentation suivante :

- the section called “Limitations pour Amazon S3”
- the section called “Recommandations d'Amazon S3”
- the section called “Quotas pour Amazon S3”

Avant de créer une source de données, vous devez disposer des ressources suivantes dans votre Compte AWS :

- Un OpenSearch domaine doté de la version 2.13 ou ultérieure. C'est la base de la configuration de l'intégration directe des requêtes. Pour obtenir des instructions sur cette configuration, reportez-vous à la sectionthe section called “ Création de domaines OpenSearch de service”.
- Un ou plusieurs compartiments S3. Vous devez spécifier les compartiments contenant les données que vous souhaitez interroger, ainsi qu'un compartiment dans lequel stocker les points de contrôle de votre requête. Pour obtenir des instructions sur la création d'un compartiment S3, consultez la section Création d'un compartiment dans le guide de l'utilisateur Amazon S3.
- (Facultatif) Une ou plusieurs AWS Glue tables.Pour interroger des données sur Amazon S3, vous devez configurer des tables AWS Glue Data Catalog pour pointer vers les données S3. Vous devez créer les tables à l'aide de OpenSearch Query Workbench. Les tables Hive existantes ne sont pas compatibles.

  Si c'est la première fois que vous configurez une source de données Amazon S3, vous devez créer une source de données d'administration pour configurer toutes vos AWS Glue Data Catalog tables. Vous pouvez le faire en installant OpenSearch out-of-the-box des intégrations ou en utilisant OpenSearch Query Workbench pour créer des tables SQL personnalisées pour des cas d'utilisation avancés. Pour des exemples sur la création de tables pour VPC et de journaux AWS WAF, consultez la documentation sur le GitHub VPC, et. CloudTrail CloudTrailAWS WAF Après avoir créé vos tables, vous pouvez créer de nouvelles sources de données Amazon S3 et restreindre l'accès à un nombre limité de tables.
- (Facultatif) Rôle IAM créé manuellement.Vous pouvez utiliser ce rôle pour gérer l'accès à votre source de données. Vous pouvez également demander à OpenSearch Service de créer automatiquement un rôle pour vous avec les autorisations requises. Si vous choisissez d'utiliser un rôle IAM créé manuellement, suivez les instructions figurant dansthe section called “Autorisations requises pour les rôles IAM créés manuellement”.

Procédure

Vous pouvez configurer une source de données à requête directe sur un domaine à l'aide de l'API AWS Management Console ou du OpenSearch service.

Pour configurer une source de données à l'aide du AWS Management Console

1. Accédez à la console Amazon OpenSearch Service à l'adressehttps://console.aws.amazon.com/aos/.
2. Dans le volet de navigation de gauche, choisissez Domains (Domaines).

3. Sélectionnez le domaine pour lequel vous souhaitez configurer une nouvelle source de données. Cette action ouvre la page des détails du domaine.
4. Cliquez sur l'onglet Connexions sous les détails généraux du domaine et recherchez la section Requête directe.
5. Choisissez Configurer la source de données.
6. Entrez un nom et une description facultative pour votre nouvelle source de données.
7. Choisissez Amazon S3 avec AWS Glue Data Catalog.
8. Dans les paramètres d'accès aux autorisations IAM, choisissez le mode de gestion de l'accès.

   a. Si vous souhaitez créer automatiquement un rôle pour cette source de données, procédez comme suit :

      i. Sélectionnez Créer un nouveau rôle.

      ii. Saisissez un nom pour le rôle IAM.

      iii. Sélectionnez un ou plusieurs compartiments S3 contenant les données que vous souhaitez interroger.

      iv. Sélectionnez un compartiment S3 de points de contrôle dans lequel stocker les points de contrôle des requêtes.

      v. Sélectionnez une ou plusieurs AWS Glue bases de données ou tables pour définir les données qui peuvent être consultées. Si les tables n'ont pas encore été créées, donnez accès à la base de données par défaut.

   b. Si vous souhaitez utiliser un rôle existant que vous gérez vous-même, procédez comme suit :

      i. Sélectionnez Utiliser un rôle existant.

      ii. Sélectionnez un rôle existant dans le menu déroulant.

> **Note**
>
> Lorsque vous utilisez votre propre rôle, vous devez vous assurer qu'il dispose de toutes les autorisations nécessaires en joignant les politiques requises depuis la console IAM. Pour plus d'informations, reportez-vous à l'exemple de politique présenté dansthe section called “Autorisations requises pour les rôles IAM créés manuellement”.

9. Choisissez Configurer. Cela ouvre l'écran des détails de la source de données avec une URL de tableau de OpenSearch bord. Vous pouvez accéder à cette URL pour effectuer les étapes suivantes.

OpenSearch API de service

Utilisez l'opération AddDataSourceAPI pour créer une nouvelle source de données dans votre domaine.

```
POST https://es.region.amazonaws.com/2021-01-01/opensearch/domain/domain-name/
dataSource

{
   "DataSourceType": {
        "S3GlueDataCatalog": {
            "RoleArn": "arn:aws:iam::account-id:role/role-name"
        }
    }
   "Description": "data-source-description",
   "Name": "my-data-source"
}
```

Étapes suivantes

Visitez les OpenSearch tableaux de bord

Une fois que vous avez créé une source de données, OpenSearch Service vous fournit un lien vers OpenSearch les tableaux de bord. Vous pouvez l'utiliser pour configurer le contrôle d'accès, définir des tables, installer out-of-the-box des intégrations et interroger vos données.

Pour de plus amples informations, veuillez consulter the section called “Configuration d'une source de données S3”.

Cartographier le AWS Glue Data Catalog rôle

Si vous avez activé le contrôle d'accès détaillé après avoir créé une source de données, vous devez associer les utilisateurs non administrateurs à un rôle IAM disposant d'un AWS Glue Data Catalog accès afin d'exécuter des requêtes directes. Pour créer manuellement un `glue_access` rôle principal que vous pouvez mapper au rôle IAM, effectuez les opérations suivantes :

> **Note**
>
> Les index sont utilisés pour toutes les requêtes portant sur la source de données. Un utilisateur disposant d'un accès en lecture à l'index des requêtes pour une source de données donnée peut lire toutes les requêtes relatives à cette source de données. Un utilisateur disposant d'un accès en lecture à l'index des résultats peut lire les résultats de toutes les requêtes portant sur cette source de données.

1. Dans le menu principal OpenSearch des tableaux de bord, sélectionnez Sécurité, Rôles et Créer des rôles.
2. Nommez le rôle glue_access.
3. Pour les autorisations du cluster`indices:data/write/bulk*`, sélectionnez`indices:data/read/scroll`,`indices:data/read/scroll/clear`.
4. Dans Index, entrez les index suivants auxquels vous souhaitez accorder l'accès à l'utilisateur ayant le rôle :
   - `.query_execution_request_`*`<name of data source>`*
   - `query_execution_result_`*`<name of data source>`*
   - `.async-query-scheduler`
   - `flint_*`
5. Pour les autorisations d'indexation, sélectionnez`indices_all`.
6. Choisissez Créer.
7. Choisissez Mapped users (Utilisateurs mappés), Manage mapping (Gérer le mappage).
8. Sous Rôles principaux, ajoutez l'ARN du AWS Glue rôle qui a besoin d'une autorisation pour appeler votre domaine.

   ```
   arn:aws:iam::account-id:role/role-name
   ```

9. Sélectionnez Carte et confirmez que le rôle apparaît sous Utilisateurs mappés.

Pour plus d'informations sur le mappage des rôles, consultezthe section called “Mappage des rôles aux utilisateurs”.

Ressources supplémentaires

Autorisations requises pour les rôles IAM créés manuellement

Lorsque vous créez une source de données pour votre domaine, vous choisissez un rôle IAM pour gérer l'accès à vos données. Vous avez deux options :

1. Création automatique d'un nouveau rôle IAM
2. Utiliser un rôle IAM existant que vous avez créé manuellement

Si vous utilisez un rôle créé manuellement, vous devez associer les autorisations appropriées au rôle. Les autorisations doivent autoriser l'accès à la source de données spécifique et permettre au OpenSearch Service d'assumer le rôle. Cela est nécessaire pour que le OpenSearch Service puisse accéder à vos données et interagir avec celles-ci en toute sécurité.

L'exemple de politique suivant illustre les autorisations de moindre privilège requises pour créer et gérer une source de données. Si vous disposez d'autorisations plus larges, telles que `s3:*` la `AdminstratorAccess` politique, ces autorisations incluent les autorisations de moindre privilège indiquées dans l'exemple de politique.

Dans l'exemple de politique suivant, remplacez les *`placeholder text`* par vos propres informations.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"HttpActionsForOpenSearchDomain",
         "Effect":"Allow",
         "Action":"es:ESHttp*",
         "Resource":"arn:aws:es:region:account:domain/<domain_name>/*"
      },
      {
         "Sid":"AmazonOpenSearchS3GlueDirectQueryReadAllS3Buckets",
         "Effect":"Allow",
         "Action":[
            "s3:GetObject",
            "s3:GetObjectVersion",
```

```
            "s3:ListBucket"
         ],
         "Condition":{
            "StringEquals":{
               "aws:ResourceAccount":"account"
            }
         },
         "Resource":"*"
      },
      {
         "Sid":"AmazonOpenSearchDirectQueryGlueCreateAccess",
         "Effect":"Allow",
         "Action":[
            "glue:CreateDatabase",
            "glue:CreatePartition",
            "glue:CreateTable",
            "glue:BatchCreatePartition"
         ],
         "Resource":"*"
      },
      {
         "Sid":"AmazonOpenSearchS3GlueDirectQueryModifyAllGlueResources",
         "Effect":"Allow",
         "Action":[
            "glue:DeleteDatabase",
            "glue:DeletePartition",
            "glue:DeleteTable",
            "glue:GetDatabase",
            "glue:GetDatabases",
            "glue:GetPartition",
            "glue:GetPartitions",
            "glue:GetTable",
            "glue:GetTableVersions",
            "glue:GetTables",
            "glue:UpdateDatabase",
            "glue:UpdatePartition",
            "glue:UpdateTable",
            "glue:BatchGetPartition",
            "glue:BatchDeletePartition",
            "glue:BatchDeleteTable"
         ],
         "Resource":[
            "arn:aws:glue:us-east-1:account:table/*",
            "arn:aws:glue:us-east-1:account:database/*",
```

```
            "arn:aws:glue:us-east-1:account:catalog",
            "arn:aws:es:region:account:domain/domain_name"
         ],
         "Condition":{
            "StringEquals":{
               "aws:ResourceAccount":"account"
            }
         }
      },
      {
         "Sid":"ReadAndWriteActionsForS3CheckpointBucket",
         "Effect":"Allow",
         "Action":[
            "s3:ListMultipartUploadParts",
            "s3:DeleteObject",
            "s3:GetObject",
            "s3:PutObject",
            "s3:GetBucketLocation",
            "s3:ListBucket"
         ],
         "Condition":{
            "StringEquals":{
               "aws:ResourceAccount":"account"
            }
         },
         "Resource":[
            "arn:aws:s3:::amzn-s3-demo-bucket",
            "arn:aws:s3:::amzn-s3-demo-bucket/*"
         ]
      }
   ]
}
```

Pour prendre en charge les compartiments Amazon S3 dans différents comptes, vous devez inclure une condition dans la politique Amazon S3 et ajouter le compte approprié.

Dans l'état d'exemple suivant, remplacez-le *placeholder text* par vos propres informations.

```
"Condition": {
               "StringEquals": {
                   "aws:ResourceAccount": "{{accountId}}"
               }
```

Le rôle doit également avoir la politique de confiance suivante, qui spécifie l'ID cible.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":[
       {
          "Effect":"Allow",
          "Principal":{
             "Service": "directquery.opensearchservice.amazonaws.com"
          },
          "Action":"sts:AssumeRole"
       }
     ]
}
```

Pour obtenir des instructions quant à la création du rôle, consultez Création d'un rôle à l'aide de politiques d'approbation personnalisées.

Si le contrôle d'accès détaillé est activé dans OpenSearch Service, un nouveau rôle de contrôle OpenSearch d'accès détaillé sera automatiquement créé pour votre source de données. Le nom du nouveau rôle de contrôle d'accès détaillé sera. `AWS OpenSearchDirectQuery` *`<name of data source>`*

Par défaut, le rôle a uniquement accès aux index de sources de données de requête directe. Bien que vous puissiez configurer le rôle pour limiter ou autoriser l'accès à votre source de données, il est recommandé de ne pas ajuster l'accès de ce rôle. Si vous supprimez la source de données, ce rôle sera supprimé. Cela supprimera l'accès de tous les autres utilisateurs s'ils sont mappés au rôle.

Configuration et interrogation d'une source de données S3 dans OpenSearch les tableaux de bord

Maintenant que vous avez créé votre source de données, vous pouvez configurer les paramètres de sécurité, définir vos tables Amazon S3 ou configurer l'indexation accélérée des données. Cette section vous présente les différents cas d'utilisation de votre source de données dans les OpenSearch tableaux de bord avant que vous n'interrogiez vos données.

Pour configurer les sections suivantes, vous devez d'abord accéder à votre source de données dans les OpenSearch tableaux de bord. Dans le menu de navigation de gauche, sous Gestion, sélectionnez Sources de données. Sous Gérer les sources de données, sélectionnez le nom de la source de données que vous avez créée dans la console.

Création de tables Spark à l'aide de Query Workbench

Les requêtes directes de OpenSearch Service à Amazon S3 utilisent les tables Spark dans le AWS Glue Data Catalog. Vous pouvez créer des tables depuis le Query Workbench sans avoir à quitter les OpenSearch tableaux de bord.

Pour gérer les bases de données et les tables existantes dans votre source de données, ou pour créer de nouvelles tables sur lesquelles vous souhaitez utiliser des requêtes directes, choisissez Query Workbench dans le menu de navigation de gauche et sélectionnez la source de données Amazon S3 dans le menu déroulant des sources de données.

Pour configurer une table pour les journaux de flux VPC stockés au format S3 au format Parquet, exécutez la requête suivante :

```
CREATE TABLE
datasourcename.gluedatabasename.vpclogstable (version INT, account_id STRING,
 interface_id STRING,
srcaddr STRING, dstaddr STRING, srcport INT, dstport INT, protocol INT, packets
 BIGINT,
bytes BIGINT, start BIGINT, end BIGINT, action STRING, log_status STRING,
`aws-account-id` STRING, `aws-service` STRING, `aws-region` STRING, year STRING,
month STRING, day STRING, hour STRING)

USING parquet PARTITIONED BY (aws-account-id, aws-service, aws-region, year, month,
day, hour)

LOCATION "s3://accountnum-vpcflow/AWSLogs"
```

Après avoir créé la table, exécutez la requête suivante pour vous assurer qu'elle est compatible avec les requêtes directes :

```
MSCK REPAIR TABLE  datasourcename.databasename.vpclogstable
```

Configurer des intégrations pour les types de AWS journaux les plus courants

Vous pouvez intégrer les types de AWS journaux stockés dans Amazon S3 à OpenSearch Service. Utilisez OpenSearch les tableaux de bord pour installer des intégrations qui créent des AWS Glue Data Catalog tables, des requêtes enregistrées et des tableaux de bord. Ces intégrations utilisent des vues indexées pour maintenir les tableaux de bord à jour.

Pour obtenir des instructions sur l'installation d'une intégration, consultez la section Installation d'une ressource d'intégration dans la OpenSearch documentation.

Lorsque vous sélectionnez une intégration, assurez-vous qu'elle possède le `S3 Glue` tag.

Lorsque vous configurez l'intégration, spécifiez S3 Connection pour le type de connexion. Sélectionnez ensuite la source de données pour l'intégration, l'emplacement des données sur Amazon S3, le point de contrôle pour gérer l'indexation accélérée et les ressources requises pour votre cas d'utilisation.

> **Note**
>
> Assurez-vous que le compartiment S3 de votre point de contrôle dispose d'autorisations d'écriture pour l'emplacement du point de contrôle. Sans ces autorisations, les accélérations de l'intégration échoueront.

Configurer le contrôle d'accès

Sur la page de détails de votre source de données, recherchez la section Contrôles d'accès et choisissez Modifier. Si le contrôle d'accès détaillé est activé dans le domaine, choisissez Restreint et sélectionnez les rôles que vous souhaitez fournir pour accéder à la nouvelle source de données. Vous pouvez également choisir Admin uniquement si vous souhaitez que l'administrateur ait uniquement accès à la source de données.

> **Important**
>
> Les index sont utilisés pour toutes les requêtes portant sur la source de données. Un utilisateur disposant d'un accès en lecture à l'index des requêtes pour une source de données donnée peut lire toutes les requêtes relatives à cette source de données. Un utilisateur disposant d'un accès en lecture à l'index des résultats peut lire les résultats de toutes les requêtes portant sur cette source de données.

Interrogation de données S3 dans Discover OpenSearch

Après avoir configuré vos tables et configuré l'accélération de requête optionnelle que vous souhaitez, vous pouvez commencer à analyser vos données. Pour interroger vos données, sélectionnez votre source de données dans le menu déroulant. Si vous utilisez Amazon S3 et OpenSearch Dashboards, accédez à Discover et sélectionnez le nom de la source de données.

Si vous utilisez un index à ignorer ou si vous n'en avez pas créé, vous pouvez utiliser SQL ou PPL pour interroger vos données. Si vous avez configuré une vue matérialisée ou un index de couverture, vous disposez déjà d'un index et vous pouvez utiliser le langage de requête DQL (Dashboards Query Language) dans tous les tableaux de bord. Vous pouvez également utiliser PPL avec le plug-in Observability et SQL avec le plug-in Query Workbench. Actuellement, seuls les plugins Observability et Query Workbench prennent en charge les protocoles PPL et SQL. Pour interroger des données à l'aide de l'API OpenSearch de service, reportez-vous à la documentation de l'API asynchrone.

> **Note**
>
> Toutes les instructions, commandes et fonctions SQL et PPL ne sont pas prises en charge. Pour obtenir la liste des commandes prises en charge, consultezthe section called “Commandes SQL et PPL prises en charge”.
> Si vous avez créé une vue matérialisée ou un index de couverture, vous pouvez utiliser DQL pour interroger vos données étant donné que vous les y avez indexées.

Résolution des problèmes

Il peut arriver que les résultats ne s'affichent pas comme prévu. Si vous rencontrez des problèmes, assurez-vous de suivre lethe section called “Recommandations”.

Interrogation directe des données Amazon CloudWatch Logs dans Service OpenSearch

Cette section explique le processus de création et de configuration d'une intégration de source de données dans Amazon OpenSearch Service, vous permettant ainsi d'interroger et d'analyser efficacement vos données stockées dans CloudWatch Logs.

Dans les pages suivantes, vous allez apprendre à configurer une source de données CloudWatch Logs à requête directe, à parcourir les prérequis nécessaires et à suivre les step-by-step procédures à l'aide du. AWS Management Console

Rubriques

- Création d'une intégration de source de données Amazon CloudWatch Logs dans OpenSearch Service
- Configuration et interrogation d'une source de données CloudWatch Logs dans OpenSearch les tableaux de bord

Création d'une intégration de source de données Amazon CloudWatch Logs dans OpenSearch Service

Si vous utilisez Amazon OpenSearch Serverless pour vos besoins d'observabilité, vous pouvez désormais analyser vos Amazon CloudWatch Logs sans copier ni ingérer les données dans Service. OpenSearch Cette fonctionnalité utilise les requêtes directes pour interroger les données, de la même manière que l'analyse des données dans Amazon S3 à partir de OpenSearch Service. Vous pouvez commencer par créer une nouvelle source de données connectée à partir de la console de AWS gestion.

Vous pouvez créer une nouvelle source de données pour analyser les données CloudWatch des journaux sans avoir à créer Amazon OpenSearch Serverless pour interroger directement les journaux opérationnels dans les CloudWatch journaux. Cela vous permet d'analyser les données opérationnelles auxquelles vous avez accédé et qui se trouvent en dehors du OpenSearch Service. En interrogeant le OpenSearch service et CloudWatch les journaux, vous pouvez commencer à analyser les CloudWatch journaux dans Logs, puis revenir à la surveillance des sources de données OpenSearch sans avoir à changer d'outil.

Pour utiliser cette fonctionnalité, vous devez créer une source de données de requête directe CloudWatch Logs pour le OpenSearch service via la console AWS de gestion.

Rubriques

- Prérequis
- Procédure
- Étapes suivantes
- Ressources supplémentaires

Prérequis

Avant de commencer, assurez-vous d'avoir pris connaissance de la documentation suivante :

- the section called “Limitations pour Amazon CloudWatch Logs”
- the section called “CloudWatch Recommandations relatives aux journaux”
- the section called “Quotas pour les CloudWatch journaux”

Avant de créer une source de données, vous devez disposer des ressources suivantes dans votre Compte AWS :

- Activez CloudWatch les journaux. Configurez les CloudWatch journaux pour collecter les journaux au même Compte AWS titre que votre OpenSearch ressource. Pour obtenir des instructions, consultez Getting started with CloudWatch Logs dans le guide de l'utilisateur Amazon CloudWatch Logs.
- Un ou plusieurs groupes de CloudWatch journaux.Vous pouvez spécifier les groupes de journaux contenant les données que vous souhaitez interroger. Pour obtenir des instructions sur la création d'un groupe de journaux, consultez la section Créer un groupe de CloudWatch journaux dans le guide de l'utilisateur d'Amazon CloudWatch Logs.
- (Facultatif) Rôle IAM créé manuellement.Vous pouvez utiliser ce rôle pour gérer l'accès à votre source de données. Vous pouvez également demander à OpenSearch Service de créer automatiquement un rôle pour vous avec les autorisations requises. Si vous choisissez d'utiliser un rôle IAM créé manuellement, suivez les instructions figurant dansthe section called “Autorisations requises pour les rôles IAM créés manuellement”.

Procédure

Vous pouvez configurer une source de données de requête au niveau de la collection à l'aide du. AWS Management Console

Pour configurer une source de données au niveau de la collection à l'aide du AWS Management Console

1. Accédez à la console Amazon OpenSearch Service à l'adressehttps://console.aws.amazon.com/aos/.
2. Dans le volet de navigation de gauche, accédez à Gestion centrale et choisissez Sources de données connectées.

3. Choisissez Se connecter.
4. Choisissez CloudWatchcomme type de source de données.
5. Choisissez Suivant.
6. Sous Détails de la connexion aux données, entrez un nom et une description facultative.
7. Sous Rôles IAM, choisissez le mode de gestion de l'accès aux groupes de journaux.

    a. Si vous souhaitez créer automatiquement un rôle pour cette source de données, procédez comme suit :

        i. Sélectionnez Créer un nouveau rôle.

        ii. Saisissez un nom pour le rôle IAM.

        iii. Sélectionnez un ou plusieurs groupes de journaux pour définir les données qui peuvent être consultées.

    b. Si vous souhaitez utiliser un rôle existant que vous gérez vous-même, procédez comme suit :

        i. Sélectionnez Utiliser un rôle existant.

        ii. Sélectionnez un rôle existant dans le menu déroulant.

> **Note**
>
> Lorsque vous utilisez votre propre rôle, vous devez vous assurer qu'il dispose de toutes les autorisations nécessaires en joignant les politiques requises depuis la console IAM. Pour de plus amples informations, veuillez consulter the section called “Autorisations requises pour les rôles IAM créés manuellement”.

8. (Facultatif) Sous Balises, ajoutez des balises à votre source de données.
9. Choisissez Suivant.
10. Sous Configuration OpenSearch, choisissez le mode de configuration OpenSearch.

    a. Utilisez les paramètres par défaut :

        - Vérifiez les noms des ressources par défaut et les paramètres de conservation des données. Nous vous suggérons d'utiliser des noms personnalisés.

Lorsque vous utilisez les paramètres par défaut, une nouvelle OpenSearch application et un nouvel espace de travail Essentials sont créés pour vous sans frais supplémentaires. OpenSearch vous permet d'analyser plusieurs sources de données. Il inclut des espaces de travail, qui offrent des expériences personnalisées pour les cas d'utilisation courants. Les espaces de travail prennent en charge le contrôle d'accès, ce qui vous permet de créer des espaces privés pour vos cas d'utilisation et de les partager uniquement avec vos collaborateurs.

b. Utilisez des paramètres personnalisés :

   i. Choisissez Personnaliser.

   ii. Modifiez le nom de la collection et les paramètres de conservation des données selon vos besoins.

   iii. Sélectionnez l' OpenSearch application et l'espace de travail que vous souhaitez utiliser.

11. Choisissez Suivant.
12. Passez en revue vos choix et choisissez Modifier si vous devez apporter des modifications.
13. Choisissez Connect pour configurer la source de données. Restez sur cette page pendant la création de votre source de données. Lorsqu'elle sera prête, vous serez redirigé vers la page de détails de la source de données.

Étapes suivantes

Visitez les OpenSearch tableaux de bord

Une fois que vous avez créé une source de données, OpenSearch Service vous fournit une URL de tableau de OpenSearch bord. Vous l'utilisez pour configurer le contrôle d'accès, définir des tables, configurer des tableaux de bord basés sur le type de journal pour les types de journaux les plus courants et interroger vos données à l'aide de SQL ou PPL.

Pour de plus amples informations, veuillez consulter the section called “Configuration d'une source de données CloudWatch Logs”.

Ressources supplémentaires

Autorisations requises pour les rôles IAM créés manuellement

Lorsque vous créez une source de données, vous choisissez un rôle IAM pour gérer l'accès à vos données. Vous avez deux options :

1. Création automatique d'un nouveau rôle IAM
2. Utiliser un rôle IAM existant que vous avez créé manuellement

Si vous utilisez un rôle créé manuellement, vous devez associer les autorisations appropriées au rôle. Les autorisations doivent autoriser l'accès à la source de données spécifique et permettre au OpenSearch Service d'assumer le rôle. Cela est nécessaire pour que le OpenSearch Service puisse accéder à vos données et interagir avec celles-ci en toute sécurité.

L'exemple de politique suivant illustre les autorisations de moindre privilège requises pour créer et gérer une source de données. Si vous disposez d'autorisations plus larges, telles que `logs:*` la `AdminstratorAccess` politique, ces autorisations incluent les autorisations de moindre privilège indiquées dans l'exemple de politique.

Dans l'exemple de politique suivant, remplacez les *`placeholder text`* par vos propres informations.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonOpenSearchDirectQueryAllLogsAccess",
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogGroups",
                "logs:StartQuery",
                "logs:GetLogGroupFields"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceAccount": "accountId"
                }
            },
            "Resource": [
                "arn:aws:logs:region:accountId:log-group:*"
```

```
            ]
        }
    ]
}

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonOpenSearchDirectQueryServerlessAccess",
            "Effect": "Allow",
            "Action": [
                "aoss:APIAccessAll",
                "aoss:DashboardsAccessAll"
            ],
            "Resource": [
                "arn:aws:aoss:region:accountId:collection/ARN/*",
                "arn:aws:aoss:region:accountId:collection/ARN"
            ]
        }
    ]
}
```

Le rôle doit également avoir la politique de confiance suivante, qui spécifie l'ID cible.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "TrustPolicyForAmazonOpenSearchDirectQueryService",
            "Effect": "Allow",
            "Principal": {
                "Service": "directquery.opensearchservice.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "ArnLike": {
                    "aws:SourceArn":
 "arn:aws:opensearch:region:accountId:datasource/rolename"
                }
            }
```

```
        }
    ]
}
```

Pour obtenir des instructions quant à la création du rôle, consultez Création d'un rôle à l'aide de politiques d'approbation personnalisées.

Par défaut, le rôle a uniquement accès aux index de sources de données de requête directe. Bien que vous puissiez configurer le rôle pour limiter ou autoriser l'accès à votre source de données, il est recommandé de ne pas ajuster l'accès de ce rôle. Si vous supprimez la source de données, ce rôle sera supprimé. Cela supprimera l'accès de tous les autres utilisateurs s'ils sont mappés au rôle.

Configuration et interrogation d'une source de données CloudWatch Logs dans OpenSearch les tableaux de bord

Maintenant que vous avez créé votre source de données, vous pouvez commencer à utiliser ses OpenSearch tableaux de bord. Cette section vous présente les différents cas d'utilisation de votre source de données dans les OpenSearch tableaux de bord.

Groupes de journaux de requêtes depuis la page Découvrir

Sur la page OpenSearch Découvrir, vous pouvez utiliser la nouvelle source de données de requête directe que vous avez configurée pour interroger vos groupes de CloudWatch journaux Logs. Pour ce faire, choisissez Explore les journaux, puis utilisez la barre de recherche pour créer votre requête à l'aide de SQL ou PPL. Vous pouvez filtrer, trier et visualiser les données renvoyées par vos groupes de journaux. Pour savoir quelles instructions, commandes et limitations sont prises en charge pour l'intégration CloudWatch des journaux, consultezthe section called “Commandes SQL et PPL prises en charge”.

Créez une vue de tableau de bord pour votre source de données

Lorsque vous utilisez OpenSearch Service, vous pouvez analyser rapidement les types de AWS journaux les plus courants à l'aide de modèles de tableau de bord prédéfinis. Pour les CloudWatch journaux, il existe des modèles pour les journaux VPC et WAF. CloudTrail Ces modèles vous permettent de créer rapidement un tableau de bord adapté à vos données spécifiques. Ils incluent des tableaux de bord adaptés à ce type de journal spécifique. Cela vous permet de vous lancer rapidement dans l'analyse de ces sources de AWS journaux populaires, sans avoir à tout créer à partir de zéro.

> **Note**
>
> Les tableaux de bord utilisent des vues indexées, qui ingèrent les données des CloudWatch journaux à l'aide d'unités de OpenSearch calcul (OCUs) à requête directe, ainsi que de l'indexationOCUs, de la recherche et du stockage des collections sans serveur. OCUs

Suivez ces étapes pour créer un tableau de bord à l'aide de l'un de ces modèles prédéfinis, afin de pouvoir commencer à explorer et à analyser vos données immédiatement.

Pour créer une vue de tableau de bord

1. Accédez à la console Amazon OpenSearch Service à l'adressehttps://console.aws.amazon.com/aos/.
2. Dans le volet de navigation de gauche, choisissez Gestion centrale, puis Sources de données connectées.
3. Sélectionnez la source de données pour ouvrir la page de détails.
4. Choisissez Create dashboard (Créer un tableau de bord).
5. Choisissez le type de tableau de bord que vous souhaitez créer.
6. Entrez un nom pour votre tableau de bord.
7. Entrez une description facultative pour votre tableau de bord.
8. Sélectionnez un ou plusieurs groupes de journaux à afficher sur votre tableau de bord.
9. Choisissez la fréquence à laquelle vous souhaitez actualiser les données de votre tableau de bord.
10. Choisissez l' OpenSearch espace de travail que vous souhaitez utiliser.
    a. Pour créer un nouvel espace de travail, sélectionnez Créer un nouvel espace de travail et entrez un nom.
    b. Pour utiliser un espace de travail existant, sélectionnez Sélectionner un espace de travail existant.
11. Choisissez Create dashboard (Créer un tableau de bord).

Interrogation des données CloudWatch des journaux dans Discover OpenSearch

Pour interroger vos données, sélectionnez votre source de données dans le menu déroulant. Si vous utilisez CloudWatch Logs, accédez à Discover depuis votre espace de travail Essentials et commencez à interroger les données à l'aide du langage OpenSearch SQL ou PPL (Piped Processing Language). Pour obtenir la liste des commandes prises en charge, consultezthe section called “Commandes SQL et PPL prises en charge”.

> **Note**
>
> Si vous avez créé une vue matérialisée, vous pouvez utiliser DQL pour interroger vos données étant donné que vous les y avez indexées.

Résolution des problèmes

Il peut arriver que les résultats ne s'affichent pas comme prévu. Si vous rencontrez des problèmes, assurez-vous de suivre lethe section called “Recommandations”.

Interrogation directe des données Amazon Security Lake dans Service OpenSearch

Cette section explique le processus de création et de configuration d'une intégration de source de données dans Amazon OpenSearch Service, vous permettant ainsi d'interroger et d'analyser efficacement vos données stockées dans Security Lake.

Dans les pages suivantes, vous allez apprendre à configurer une source de données à requête directe Security Lake, à définir les prérequis nécessaires et à suivre les step-by-step procédures à l'aide du. AWS Management Console

Rubriques

- Création d'une intégration de source de données Amazon Security Lake dans OpenSearch Service
- Configuration et interrogation d'une source de données Security Lake dans OpenSearch les tableaux de bord

Création d'une intégration de source de données Amazon Security Lake dans OpenSearch Service

Vous pouvez utiliser Amazon OpenSearch Serverless pour interroger directement les données de sécurité dans Amazon Security Lake. Pour ce faire, vous créez une source de données qui vous permet d'utiliser les fonctionnalités OpenSearch Zero-ETL sur les données de Security Lake. Lorsque vous créez une source de données, vous pouvez effectuer des recherches, obtenir des informations et analyser directement les données stockées dans Security Lake. Vous pouvez accélérer les performances de vos requêtes et utiliser des OpenSearch analyses avancées sur certains ensembles de données Security Lake grâce à l'indexation à la demande.

Rubriques

- Prérequis
- Procédure
- Étapes suivantes
- Ressources supplémentaires

Prérequis

Avant de commencer, assurez-vous d'avoir pris connaissance de la documentation suivante :

- the section called “Limitations d'Amazon Security Lake”
- the section called “Recommandations de Security Lake”
- the section called “Quotas pour Security Lake”

Avant de créer une source de données, effectuez les actions suivantes dans Security Lake :

- Activez Security Lake. Configurez Security Lake pour collecter des journaux en même temps Région AWS que votre OpenSearch ressource. Pour obtenir des instructions, consultez Getting started with Amazon Security Lake dans le guide de l'utilisateur d'Amazon Security Lake.
- Configurez les autorisations de Security Lake. Assurez-vous que vous avez accepté les autorisations de rôle liées au service pour la gestion des ressources et que la console n'affiche aucun problème sur la page Problèmes. Pour plus d'informations, consultez la section Rôle lié à un service pour Security Lake dans le guide de l'utilisateur d'Amazon Security Lake.

- Partagez les sources de données de Security Lake. Lorsque vous accédez OpenSearch avec le même compte que Security Lake, assurez-vous qu'aucun message ne vous invite à enregistrer vos seaux Security Lake auprès de Lake Formation dans la console Security Lake. Pour un OpenSearch accès entre comptes, configurez un abonné à la requête Lake Formation dans la console Security Lake. Utilisez le compte associé à votre OpenSearch ressource en tant qu'abonné. Pour plus d'informations, consultez la section Gestion des abonnés dans Security Lake dans le guide de l'utilisateur d'Amazon Security Lake.

En outre, vous devez également avoir les ressources suivantes dans votre Compte AWS :

- (Facultatif) Rôle IAM créé manuellement. Vous pouvez utiliser ce rôle pour gérer l'accès à votre source de données. Vous pouvez également demander à OpenSearch Service de créer automatiquement un rôle pour vous avec les autorisations requises. Si vous choisissez d'utiliser un rôle IAM créé manuellement, suivez les instructions figurant dansthe section called “Autorisations requises pour les rôles IAM créés manuellement”.

Procédure

Vous pouvez configurer une source de données pour vous connecter à une base de données Security Lake depuis le AWS Management Console.

Pour configurer une source de données à l'aide du AWS Management Console

1. Accédez à la console Amazon OpenSearch Service à l'adressehttps://console.aws.amazon.com/aos/.
2. Dans le volet de navigation de gauche, accédez à Gestion centrale et choisissez Sources de données connectées.
3. Choisissez Se connecter.
4. Choisissez Security Lake comme type de source de données.
5. Choisissez Suivant.
6. Sous Détails de la connexion aux données, entrez un nom et une description facultative.
7. Dans les paramètres d'accès aux autorisations IAM, choisissez comment gérer l'accès à votre source de données.

    a. Si vous souhaitez créer automatiquement un rôle pour cette source de données, procédez comme suit :

i. Sélectionnez Créer un nouveau rôle.

ii. Saisissez un nom pour le rôle IAM.

iii. Sélectionnez une ou plusieurs AWS Glue tables pour définir les données qui peuvent être consultées.

b. Si vous souhaitez utiliser un rôle existant que vous gérez vous-même, procédez comme suit :

i. Sélectionnez Utiliser un rôle existant.

ii. Sélectionnez un rôle existant dans le menu déroulant.

> **Note**
>
> Lorsque vous utilisez votre propre rôle, vous devez vous assurer qu'il dispose de toutes les autorisations nécessaires en joignant les politiques requises depuis la console IAM. Pour de plus amples informations, veuillez consulter the section called “Autorisations requises pour les rôles IAM créés manuellement”.

8. (Facultatif) Sous Balises, ajoutez des balises à votre source de données.
9. Choisissez Suivant.
10. Sous Configuration OpenSearch, choisissez le mode de configuration OpenSearch.

    - Vérifiez les noms des ressources par défaut et les paramètres de conservation des données.

      Lorsque vous utilisez les paramètres par défaut, une nouvelle OpenSearch application et un nouvel espace de travail Essentials sont créés pour vous sans frais supplémentaires. OpenSearch vous permet d'analyser plusieurs sources de données. Il inclut des espaces de travail, qui offrent des expériences personnalisées pour les cas d'utilisation courants. Les espaces de travail prennent en charge le contrôle d'accès, ce qui vous permet de créer des espaces privés pour vos cas d'utilisation et de les partager uniquement avec vos collaborateurs.

11. Utilisez des paramètres personnalisés :

    a. Choisissez Personnaliser.

b. Modifiez le nom de la collection et les paramètres de conservation des données selon vos besoins.

c. Sélectionnez l' OpenSearch application et l'espace de travail que vous souhaitez utiliser.

12. Choisissez Suivant.
13. Passez en revue vos choix et choisissez Modifier si vous devez apporter des modifications.
14. Choisissez Connect pour configurer la source de données. Restez sur cette page pendant la création de votre source de données. Lorsqu'elle sera prête, vous serez redirigé vers la page de détails de la source de données.

Étapes suivantes

Consultez les OpenSearch tableaux de bord et créez un tableau de bord

Une fois que vous avez créé une source de données, OpenSearch Service vous fournit une URL de tableau de OpenSearch bord. Vous l'utilisez pour interroger vos données à l'aide de SQL ou PPL. L'intégration de Security Lake est fournie avec des modèles de requêtes prédéfinis pour SQL et PPL afin de vous permettre de commencer à analyser vos journaux.

Pour de plus amples informations, veuillez consulter the section called “Configuration d'une source de données Security Lake”.

Ressources supplémentaires

Autorisations requises pour les rôles IAM créés manuellement

Lorsque vous créez une source de données, vous choisissez un rôle IAM pour gérer l'accès à vos données. Vous avez deux options :

1. Création automatique d'un nouveau rôle IAM
2. Utiliser un rôle IAM existant que vous avez créé manuellement

Si vous utilisez un rôle créé manuellement, vous devez associer les autorisations appropriées au rôle. Les autorisations doivent autoriser l'accès à la source de données spécifique et permettre au OpenSearch Service d'assumer le rôle. Cela est nécessaire pour que le OpenSearch Service puisse accéder à vos données et interagir avec celles-ci en toute sécurité.

L'exemple de politique suivant illustre les autorisations de moindre privilège requises pour créer et gérer une source de données. Si vous disposez d'autorisations plus étendues, telles que la

`AdminstratorAccess` politique, ces autorisations incluent les autorisations de moindre privilège indiquées dans l'exemple de politique.

Dans l'exemple de politique suivant, remplacez les *`placeholder text`* par vos propres informations.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonOpenSearchDirectQueryServerlessAccess",
            "Effect": "Allow",
            "Action": [
                "aoss:APIAccessAll",
                "aoss:DashboardsAccessAll"
            ],
            "Resource": "arn:aws:aoss:region:account:collection/collectionname/*"
        },
        {
            "Sid": "AmazonOpenSearchDirectQueryGlueAccess",
            "Effect": "Allow",
            "Action": [
                "glue:GetDatabase",
                "glue:GetDatabases",
                "glue:GetPartition",
                "glue:GetPartitions",
                "glue:GetTable",
                "glue:GetTableVersions",
                "glue:GetTables",
                "glue:SearchTables",
                "glue:BatchGetPartition"
            ],
            "Resource": [
                "arn:aws:glue:region:account:table/databasename/*",
                "arn:aws:glue:region:account:database/databasename",
                "arn:aws:glue:region:account:catalog",
                "arn:aws:glue:region:account:database/default"
            ]
        },
        {
            "Sid": "AmazonOpenSearchDirectQueryLakeFormationAccess",
```

```
            "Effect": "Allow",
            "Action": [
                "lakeformation:GetDataAccess"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

Le rôle doit également avoir la politique de confiance suivante, qui spécifie l'ID cible.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "directquery.opensearchservice.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Pour obtenir des instructions quant à la création du rôle, consultez Création d'un rôle à l'aide de politiques d'approbation personnalisées.

Par défaut, le rôle a uniquement accès aux index de sources de données de requête directe. Bien que vous puissiez configurer le rôle pour limiter ou autoriser l'accès à votre source de données, il est recommandé de ne pas ajuster l'accès de ce rôle. Si vous supprimez la source de données, ce rôle sera supprimé. Cela supprimera l'accès de tous les autres utilisateurs s'ils sont mappés au rôle.

Interrogation des données de Security Lake chiffrées à l'aide d'une clé gérée par le client

Si le bucket Security Lake associé à la connexion de données est chiffré à l'aide d'un chiffrement côté serveur géré par le client AWS KMS key, vous devez ajouter le rôle de LakeFormation service à la politique clé. Cela permet au service d'accéder aux données pour vos requêtes et de les lire.

Dans l'exemple de politique suivant, remplacez les *placeholder text* par vos propres informations.

```
{
    "Sid": "Allow LakeFormation to access the key",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::account:role/aws-service-role/lakeformation.amazonaws.com/
AWSServiceRoleForLakeFormationDataAccess"
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*"
}
```

Configuration et interrogation d'une source de données Security Lake dans OpenSearch les tableaux de bord

Maintenant que vous avez créé votre source de données, vous pouvez la configurer dans les OpenSearch tableaux de bord.

Cette section vous présente les différents cas d'utilisation de votre source de données dans les OpenSearch tableaux de bord avant que vous n'interrogiez vos données. Pour commencer, vous devez accéder à votre source de données dans les OpenSearch tableaux de bord. Dans le menu de gauche, sous Gestion, sélectionnez Sources de données. Sélectionnez ensuite le nom de la source de données que vous avez créée précédemment dans la console OpenSearch de service.

Interrogez les tables Security Lake depuis Discover

Si vous avez créé des tables à partir de vos journaux Security Lake, vous pouvez désormais interroger ces tables directement depuis OpenSearch Discover. Cela vous permet d'accéder et d'analyser facilement les données stockées dans Security Lake, directement à partir de l'interface Discover familière. En interrogeant Security Lake directement depuis Discover, vous pouvez éviter d'avoir à extraire, transformer et charger manuellement les données dans un index de recherche distinct. Pour commencer rapidement à analyser vos journaux, Discover inclut un ensemble de requêtes enregistrées en PPL et SQL.

Commencez par sélectionner la source de données que vous avez configurée. Sélectionnez la base de données et la table associées que vous souhaitez interroger, puis utilisez la barre de recherche pour écrire des requêtes sur vos tables. Pour savoir quelles instructions, commandes et limitations sont prises en charge pour l'intégration de Security Lake, consultezthe section called “Commandes SQL et PPL prises en charge”.

Pour tirer parti des requêtes prédéfinies disponibles pour Security Lake, rendez-vous sur... en haut à droite de Discover, choisissez Open Query, puis Templates. De nombreuses requêtes prédéfinies sont disponibles pour les sources de journaux prises en charge dans Security Lake. Recherchez les modèles qui correspondent à votre cas d'utilisation, copiez la requête à utiliser dans la barre de recherche et remplacez les champs du modèle (tels que Région et action) par vos propres informations.

Accélérez les données issues de Discover

Pour améliorer les performances et accélérer les requêtes et analyses ultérieures OpenSearch, vous pouvez intégrer les résultats de votre requête depuis Discover dans une vue OpenSearch indexée.

Pour créer une vue indexée

1. Dans Discover, choisissez Create indexed View.
2. Dans l'éditeur de requêtes, saisissez la requête de votre choix. Vous pouvez créer une nouvelle requête ici ou utiliser une requête existante issue de vos recherches précédentes.
3. Spécifiez le nom de votre nouvelle vue indexée. Choisissez un nom descriptif qui vous aidera à identifier la vue ultérieurement.
4. Configurez les paramètres de conservation des données pour votre vue indexée. Vous pouvez spécifier la durée pendant laquelle les données doivent être conservées dans l'index, ce qui vous permet d'équilibrer les performances avec les coûts de stockage.

5. Créez la vue indexée. Une fois créée, votre vue indexée sera disponible pour accélérer les requêtes et les analyses.

Si vous avez déjà créé des vues indexées, vous pouvez y accéder depuis Discover.

Pour utiliser une vue d'index existante

1. Dans Discover, choisissez Select Indexed View pour voir la liste de vos vues indexées existantes pour Security Lake.
2. Choisissez la vue indexée que vous souhaitez utiliser. Cela appliquera la vue à votre requête actuelle, accélérant potentiellement de manière significative la récupération et l'analyse des données.

Créez une vue de tableau de bord pour votre source de données

Lorsque vous utilisez OpenSearch Service, vous pouvez analyser les types de AWS journaux les plus courants à l'aide de modèles de tableau de bord prédéfinis. Pour Security Lake, il existe des modèles pour les journaux VPC et WAF. CloudTrail Ces modèles vous permettent de créer un tableau de bord adapté à vos données spécifiques. Ils incluent des requêtes prédéfinies et des tableaux de bord adaptés à ce type de journal spécifique. Cela vous permet de vous lancer rapidement dans l'analyse de ces sources de AWS journaux populaires, sans avoir à tout créer à partir de zéro.

> **Note**
>
> Les tableaux de bord utilisent des vues indexées, qui ingèrent les données de Security Lake et contribuent au calcul direct des requêtes et des collectes.

Suivez ces étapes pour créer un tableau de bord à l'aide de l'un de ces modèles prédéfinis, afin de pouvoir commencer à explorer et à analyser vos données immédiatement.

Pour créer une vue de tableau de bord

1. Accédez à la console Amazon OpenSearch Service à l'adressehttps://console.aws.amazon.com/aos/.
2. Dans le volet de navigation de gauche, choisissez Gestion centrale, puis Sources de données connectées.

3. Sélectionnez la source de données pour ouvrir la page de détails.
4. Choisissez Create dashboard (Créer un tableau de bord).
5. Choisissez le type de tableau de bord que vous souhaitez créer.
6. Entrez un nom pour votre tableau de bord.
7. Entrez une description facultative pour votre tableau de bord.
8. Sélectionnez une ou plusieurs tables AWS Glue à afficher sur votre tableau de bord.
9. Choisissez la fréquence à laquelle vous souhaitez actualiser les données de votre tableau de bord.
10. Choisissez l' OpenSearch espace de travail que vous souhaitez utiliser.
    a. Pour créer un nouvel espace de travail, sélectionnez Créer un nouvel espace de travail.
    b. Pour utiliser un espace de travail existant, sélectionnez Sélectionner un espace de travail existant.
11. Entrez un nom pour votre espace de travail.
12. Choisissez Create dashboard (Créer un tableau de bord).

Résolution des problèmes

Il peut arriver que les résultats ne s'affichent pas comme prévu. Si vous rencontrez des problèmes, assurez-vous de suivre lethe section called “Recommandations”.

Gestion d'une source de données dans Amazon OpenSearch Service

La gestion de votre source de données est essentielle au maintien de la fiabilité, de la disponibilité et des performances des sources de données à requêtes directes et de vos autres AWS solutions. AWS fournit les outils suivants pour surveiller, signaler en cas de problème et prendre des mesures automatiques le cas échéant.

Rubriques

- Surveillance à l'aide CloudWatch de sources de données métriques
- Activation et désactivation des sources de données
- Surveillance avec le AWS budget

- Suppression d'une source de données

Surveillance à l'aide CloudWatch de sources de données métriques

Vous pouvez surveiller les requêtes directes à l'aide de CloudWatch. CloudWatch collecte des données brutes et les transforme en indicateurs lisibles en temps quasi réel. Ces statistiques sont enregistrées pour une durée de 15 mois ; par conséquent, vous pouvez accéder aux informations historiques et acquérir un meilleur point de vue de la façon dont votre service ou application web s'exécute.

Vous pouvez également configurer des alarmes pour surveiller certains seuils, envoyer des notifications ou prendre des mesures lorsque ces seuils sont atteints. Pour plus d'informations, consultez What is Amazon CloudWatch.

Amazon S3 indique les métriques suivantes :

Métrique	Description
AsyncQueryCreateAPI	Nombre total de demandes adressées à l'API pour créer des requêtes asynchrones. Statistiques pertinentes : moyenne, maximum, somme Dimensions :`ClientId`, `DomainName` Fréquence : 60 secondes
AsyncQueryGetApiRequestCount	Nombre total de demandes adressées à l'API pour récupérer les résultats de requêtes asynchrones. Statistiques pertinentes : moyenne, maximum, somme Dimensions :`ClientId`, `DomainName` Fréquence : 60 secondes
AsyncQueryCancelApiRequestCount	Nombre total de demandes adressées à l'API pour annuler des requêtes asynchrones. Statistiques pertinentes : moyenne, maximum, somme

Métrique	Description
	Dimensions :`ClientId`, `DomainName` Fréquence : 60 secondes
AsyncQueryGetApiFailedRequestCusErrCount	Le nombre de demandes ayant échoué lors de la récupération des résultats de requêtes asynchrones en raison d'erreurs liées au client (par exemple, un ID de requête non valide). Statistiques pertinentes : moyenne, maximum, somme Dimensions :`ClientId`, `DomainName` Fréquence : 60 secondes
AsyncQueryCancelApiFailedRequestCusErrCount	Le nombre de demandes ayant échoué lors de la récupération des résultats de requêtes asynchrones en raison d'erreurs liées au client (par exemple, un ID de requête non valide). Statistiques pertinentes : moyenne, maximum, somme Dimensions :`ClientId`, `DomainName` Fréquence : 60 secondes
AsyncQueryCancelApiFailedRequestSysErrCount	Le nombre de demandes ayant échoué lors de la création de requêtes asynchrones en raison d'erreurs liées au client. Statistiques pertinentes : moyenne, maximum, somme Dimensions :`ClientId`, `DomainName` Fréquence : 60 secondes

Métrique	Description
AsyncQueryGetApiFailedRequestSysErrCount	Nombre de demandes ayant échoué lors de la récupération des résultats de requêtes asynchrones en raison d'erreurs liées au système. Statistiques pertinentes : moyenne, maximum, somme Dimensions :`ClientId`, `DomainName` Fréquence : 60 secondes

CloudWatch Logs et Security Lake signalent les indicateurs suivants :

Métrique	Description
DirectQueryRate	Le taux de demandes effectuées par rapport aux sources de données. Statistiques pertinentes : somme, maximum, minimum, moyenne Dimensions : `DataSourceName` Fréquence : 60 secondes
DirectQueryLatency	Latence observée lors de l'exécution de requêtes sur les sources de données. Statistiques pertinentes : moyenne, P90, P99, somme, minimum, maximum Dimensions : `DataSourceName` Fréquence : 60 secondes
FailedDirectQueries	Nombre total d'échecs de requête observés sur les requêtes de source de données.

Métrique	Description
	Statistiques pertinentes : somme, maximum, minimum, moyenne Dimensions : `DataSourceName` Fréquence : 60 secondes
DirectQueryConsumedOCU	Le nombre de OCUs ces quantités consommées pour exécuter les requêtes sur les sources de données. Statistiques pertinentes : moyenne, P90, P99, somme, minimum, maximum Dimensions : `DataSourceName` Fréquence : 60 secondes

Activation et désactivation des sources de données

> **Note**
>
> Les informations suivantes s'appliquent uniquement aux sources de données Amazon S3.

Si vous souhaitez interrompre l'utilisation des requêtes directes pour une source de données, vous pouvez choisir de désactiver la source de données. La désactivation d'une source de données terminera l'exécution des requêtes existantes et empêchera l'exécution de toutes les nouvelles requêtes.

Les accélérations configurées pour améliorer les performances des requêtes, telles que le fait de sauter des index, de visualiser des vues matérialisées ou de couvrir des index, seront définies manuellement une fois qu'une source de données est désactivée. Une fois qu'une source de données est définie comme active après avoir été désactivée, les requêtes des utilisateurs s'exécutent comme prévu. Les accélérations qui étaient auparavant configurées et réglées sur manuel devront être configurées manuellement pour s'exécuter à nouveau selon un calendrier.

Surveillance avec le AWS budget

Amazon OpenSearch Service saisit les données d'utilisation de l'OCU au niveau du compte dans Cost Explorer de Billing and Cost Management. Vous pouvez prendre en compte l'utilisation de l'OCU au niveau du compte et définir des seuils et des alertes lorsque les seuils sont dépassés.

Le format du type d'utilisation à filtrer dans Cost Explorer est le suivant RegionCode : DirectQuery OCU (OCU-Hours). Si vous souhaitez être averti lorsque l'utilisation des heures DirectQuery OCU (OCU-hours) atteint votre seuil, vous pouvez créer un compte AWS Budgets et configurer une alerte en fonction du seuil que vous avez défini. En option, pour Amazon S3, vous pouvez configurer une rubrique Amazon SNS, qui désactivera une source de données si un critère de seuil est atteint.

> **Note**
>
> Les données d'utilisation dans AWS les budgets ne sont pas en temps réel et peuvent être retardées jusqu'à 8 heures.

Suppression d'une source de données

Lorsque vous supprimez une source de données, Amazon OpenSearch Service la supprime de votre domaine ou de votre collection. OpenSearch Le service supprime également les index associés à la source de données. Vos données transactionnelles ne sont pas supprimées de l'autre Service AWS, mais l'autre Service AWS n'envoie pas de nouvelles données au OpenSearch Service.

Vous pouvez supprimer une intégration de source de données à l'aide de l'API AWS Management Console ou du OpenSearch service.

AWS Management Console

Pour supprimer une source de données Amazon S3

1. Accédez à la console Amazon OpenSearch Service à l'adressehttps://console.aws.amazon.com/aos/.
2. Dans le volet de navigation de gauche, choisissez Domains (Domaines).
3. Sélectionnez le domaine pour lequel vous souhaitez supprimer une source de données. Cette action ouvre la page des détails du domaine. Cliquez sur l'onglet Connexions sous les informations générales et recherchez la section Requête directe.

4. Sélectionnez la source de données que vous souhaitez supprimer, choisissez Supprimer et confirmez la suppression.

Pour supprimer une source de données CloudWatch Logs ou Security Lake

1. Accédez à la console Amazon OpenSearch Service à l'adressehttps://console.aws.amazon.com/aos/.
2. Dans le volet de navigation de gauche, choisissez Gestion centrale, puis Sources de données connectées.
3. Sélectionnez la source de données que vous souhaitez supprimer, choisissez Supprimer et confirmez la suppression.

OpenSearch API de service

Pour supprimer une source de données Amazon S3, utilisez l'opération DeleteDataSourceAPI.

```
POST https://es.region.amazonaws.com/2021-01-01/opensearch/domain/domain-name/
dataSource/data-source-name
```

Pour supprimer une source de données CloudWatch Logs ou Security Lake, utilisez l'opération DeleteDirectQueryDataSourceAPI.

Optimisation des performances des requêtes pour les sources OpenSearch de données Amazon Service

Les performances des requêtes dans Amazon OpenSearch Service peuvent être ralenties lorsque vous accédez à des sources de données externes. Cela peut être dû à des facteurs tels que la latence du réseau, la transformation des données ou des volumes de données importants. Pour améliorer les performances, pensez à indexer certaines quantités de données en fonction du cas d'utilisation :

- Accélération des requêtes directes sur Amazon S3 (index ignoré)
- Création de visualisations de tableau de bord sur Security Lake et sur Security Lake (vues matérialisées)
- Ingestion des résultats des requêtes à l'aide de vues indexées pour un examen hors ligne ou pour une amélioration des performances sur Security Lake (vues matérialisées)

Pour une documentation complète sur les requêtes accélérées, y compris des exemples de requêtes, voir Optimiser les performances des requêtes à l'aide de l' OpenSearch indexation dans la documentation open source.

Rubriques

- Ignorer les index
- Vues matérialisées
- Index de couverture

Ignorer les index

Un index ignoré ingère uniquement les métadonnées des données stockées dans Amazon S3. Lorsque vous interrogez une table dont l'index est ignoré, le planificateur de requêtes utilise l'index pour réécrire la requête, identifiant ainsi efficacement l'emplacement des données sans scanner toutes les partitions et tous les fichiers. Cette approche permet de préciser l'emplacement exact des données stockées.

Il existe deux méthodes pour créer un index à sauter. La première méthode consiste à générer automatiquement l'index des sauts à partir des détails de la source de données. La seconde consiste à utiliser Query Workbench pour créer manuellement l'index de saut à l'aide d'une instruction SQL.

Pour générer automatiquement un index à ignorer à partir de votre source de données, accédez à Gestion du tableau de bord et Accélération des données, puis sélectionnez votre base de données et votre table (vous devrez peut-être les actualiser pour obtenir les dernières bases de données et tables). Vous pouvez ensuite choisir Generate pour générer automatiquement un index de saut, ou sélectionner manuellement chaque champ que vous souhaitez indexer et spécifier l'accélération (type d'index de saut). Enfin, choisissez Créer une accélération pour créer une tâche récurrente qui renseigne le nouvel indice de saut.

Les index ignorés ne sont pris en charge que pour les sources de données Amazon S3.

Pour plus d'informations sur la configuration des index ignorés à l'aide de Query Workbench, consultez la section Ignorer les index dans la documentation. OpenSearch

Vues matérialisées

Les vues matérialisées utilisent des requêtes complexes, telles que des agrégations, pour prendre en charge les visualisations des OpenSearch tableaux de bord. Ils ingèrent un sous-ensemble de vos

données en fonction de la requête et les stockent dans un OpenSearch index. Vous pouvez ensuite utiliser cet index pour créer des visualisations.

Les vues matérialisées sont prises en charge pour les sources de données Amazon S3 et Security Lake.

Pour plus d'informations sur la configuration de vues matérialisées à l'aide de Query Workbench, consultez la section Vues matérialisées dans la documentation. OpenSearch

Index de couverture

Un index de couverture ingère les données d'une colonne spécifiée d'une table et OpenSearch crée un nouvel index basé sur ces données. Vous pouvez utiliser ce nouvel index pour des visualisations et d'autres OpenSearch fonctionnalités, telles que la détection d'anomalies ou l'analyse géospatiale.

Les index de couverture ne sont pris en charge que pour les sources de données Amazon S3.

Pour plus d'informations sur la configuration des index de couverture, consultez la section Index de couverture dans la OpenSearch documentation.

Commandes SQL et PPL prises en charge

OpenSearch SQL et OpenSearch Pipeline Processing Language (PPL) sont des langages permettant d'interroger, d'analyser et de traiter des données dans OpenSearch CloudWatch Logs Insights et Security Lake. Vous pouvez utiliser OpenSearch SQL et OpenSearch PPL dans OpenSearch Discover pour interroger des données dans CloudWatch Logs, Amazon S3 ou Security Lake. CloudWatch Logs Insights prend également en charge les langages de requête OpenSearch PPL et OpenSearch SQL, en plus de Logs Insights QL, un langage de requête spécialement conçu pour analyser les journaux. CloudWatch

- OpenSearch SQL : OpenSearch SQL fournit une option familière si vous avez l'habitude de travailler avec des bases de données relationnelles. OpenSearch SQL offre un sous-ensemble de fonctionnalités SQL, ce qui en fait un bon choix pour effectuer des requêtes ad hoc et des tâches d'analyse de données. Avec OpenSearch SQL, vous pouvez utiliser des commandes telles que SELECT, FROM, WHERE, GROUP BY, HAVING, ainsi que diverses autres commandes et fonctions SQL disponibles dans SQL. Vous pouvez exécuter des tâches JOINs entre des tables (ou des groupes de journaux), corréler des données entre des tables (ou des groupes de journaux) à l'aide de sous-requêtes et utiliser le riche ensemble de fonctions JSON, mathématiques, de chaîne,

conditionnelles et autres fonctions SQL pour effectuer une analyse approfondie des données de journal et de sécurité.

- OpenSearch PPL (Piped Processing Language) : avec OpenSearch PPL, vous pouvez récupérer, interroger et analyser des données à l'aide de commandes groupées, ce qui facilite la compréhension et la composition de requêtes complexes. Sa syntaxe est basée sur les canaux Unix et permet d'enchaîner les commandes pour transformer et traiter les données. Avec PPL, vous pouvez filtrer et agréger des données, et utiliser des commandes telles que des sous-requêtes JOINs, LOOKUP, ainsi qu'un ensemble complet de fonctions mathématiques, de chaîne, de date, conditionnelles et autres à des fins d'analyse.

Bien que la plupart des commandes des langages de requête OpenSearch PPL et OpenSearch SQL soient communes à CloudWatch Logs OpenSearch, il existe des différences dans les ensembles de commandes et de fonctions pris en charge dans chacun de ces services. Pour plus de détails, consultez les tableaux des pages suivantes.

- the section called “Commandes SQL prises en charge”
  - the section called “CloudWatch Informations sur les journaux”
  - the section called “Restrictions SQL générales”
- the section called “Commandes PPL prises en charge”
  - the section called “Informations supplémentaires pour les utilisateurs de CloudWatch Logs Insights utilisant OpenSearch PPL”

Commandes et fonctions OpenSearch SQL prises en charge

Les tableaux de référence suivants indiquent quelles commandes SQL sont prises en charge dans OpenSearch Discover pour interroger des données dans Amazon S3, Security Lake ou CloudWatch Logs, et quelles commandes SQL sont prises en charge dans CloudWatch Logs Insights. La syntaxe SQL prise en charge dans CloudWatch Logs Insights et celle prise en charge dans OpenSearch Discover pour interroger les CloudWatch journaux sont identiques et référencées sous le nom de CloudWatch journaux dans les tableaux suivants.

> **Note**
>
> OpenSearch dispose également d'un support SQL pour interroger les données ingérées OpenSearch et stockées dans des index. Ce dialecte SQL est différent du SQL utilisé dans les requêtes directes et est appelé OpenSearch SQL sur les index.

Rubriques

- Commandes
- Fonctions
- Restrictions SQL générales
- Informations supplémentaires pour les utilisateurs de CloudWatch Logs Insights utilisant OpenSearch SQL

Commandes

> **Note**
>
> Dans la colonne des exemples de commandes, remplacez le cas *<tableName/logGroup>* échéant en fonction de la source de données que vous interrogez.
>
> - Exemple de commande : `SELECT Body , Operation FROM` **`<tableName/logGroup>`**
> - Si vous interrogez Amazon S3 ou Security Lake, utilisez : `SELECT Body , Operation FROM` **`table_name`**
> - Si vous interrogez CloudWatch Logs, utilisez : `SELECT Body , Operation FROM` **`` `LogGroupA` ``**

Command	Description	CloudWatch Journaux	Amazon S	Security Lake	Exemple de commande
the section	Affiche les	S	S	S	`SELECT method,`

Command	Description	CloudWatch Journaux	Amazon	Security Lake	Exemple de commande
called “Clause SELECT”	valeurs projetées.				status FROM <tableName/logGroup>
the section called “Clause WHERE”	Les filtres enregistrent les événements en fonction des critères de champ fournis.	S	S	S	SELECT * FROM <tableName/logGroup> WHERE status = 100
the section called “Clause GROUP BY”	Les groupes enregistrent les événements par catégorie et trouvent la moyenne en fonction des statistiques.	S	S	S	SELECT method, status, COUNT(*) AS request_count, SUM(bytes) AS total_bytes FROM <tableName/logGroup> GROUP BY method, status

Command	Description	CloudWatch Journaux	Amazon	Security Lake	Exemple de commande
the section called “Clause HAVING”	Filtre les résultats en fonction des conditions de regroupement.	S	S	S	`SELECT method, status, COUNT(*) AS request_count, SUM(bytes) AS total_bytes FROM <tableName/logGroup> GROUP BY method, status HAVING COUNT(*) > 5`
the section called “Clause ORDER BY”	Trie les résultats en fonction des champs de la clause de comman Vous pouvez trier par ordre décroiss nt ou croissan .	S	S	S	`SELECT * FROM <tableName/logGroup> ORDER BY status DESC`

Command	Description	CloudWatch Journaux	Amazon S	Security Lake	Exemple de commande

the section called “Clause JOIN”
 	
(INNER | CROSS | LEFT OUTER)
	Joint les résultats de deux tables sur la base de champs commun	S (doit utiliser Inner des Left Outer mots clés pour la jointure ; une seule opération JOIN est prise en charge dans une instruction SELECT	S (vous devez utiliser les mots clés Inner, Left Outer et Cross pour la jointure)	S (vous devez utiliser les mots clés Inner, Left Outer et Cross pour la jointure)	`SELECT`
					`  A.Body,`
					`  B.Timestamp`
					`FROM`
					`  <tableNameA/logGroupA>  AS A`
					`INNER JOIN`
					`  <tableNameB/logGroupB>  AS B`
					`  ON A.`requestId` = B.`requestId``

the section called “Clause LIMIT”	Limite les résultats aux N première lignes.	S	S	S	`SELECT`
					`  *`
					`FROM`
					`  <tableName/logGroup>`
					`LIMIT`
					`  10`

Command	Description	CloudWatch Journaux	Amazon S	Security Lake	Exemple de commande
the section called “Clause CASE”	Évalue les conditions et renvoie une valeur lorsque la première condition est remplie.	S	S	S	`SELECT method, status, CASE WHEN status BETWEEN 100 AND 199 THEN 'Informational' WHEN status BETWEEN 200 AND 299 THEN 'Success' WHEN status BETWEEN 300 AND 399 THEN 'Redirection' WHEN status BETWEEN 400 AND 499 THEN 'Client Error' WHEN status BETWEEN 500 AND 599 THEN 'Server Error' ELSE 'Unknown Status' END AS status_category, CASE method WHEN 'GET' THEN 'Read Operation' WHEN 'POST' THEN 'Create Operation' WHEN 'PUT' THEN 'Update Operation' WHEN 'PATCH' THEN 'Partial Update Operation' WHEN 'DELETE' THEN 'Delete Operation' ELSE 'Other Operation' END AS operation_type, bytes, datetime FROM <tableName/logGroup>`

Command	Description	CloudWatch Journaux	Amazon S	Security Lake	Exemple de commande

the section called “Expression de table commune”	Crée un jeu de résultats temporaire nommé dans une instruction SELECT, INSERT, UPDATE, DELETE ou MERGE.	N pris en charge	S	S	`WITH RequestStats AS (`
					`  SELECT`
					`  method,`
					`  status,`
					`  bytes,`
					`  COUNT(*) AS request_count`
					`  FROM`
					`  tableName`
					`  GROUP BY`
					`  method,`
					`  status,`
					`  bytes`
					`)`
					`SELECT`
					`  method,`
					`  status,`
					`  bytes,`
					`  request_count`
					`FROM`
					`  RequestStats`
					`WHERE`
					`  bytes > 1000`

Command	Description	CloudWatch Journaux	Amazon S	Security Lake	Exemple de commande
the section called “EXPLAIN”	Affiche le plan d'exécution d'une instruction SQL sans l'exécuter réellement.	Non pris en charge	S	S	`EXPLAIN SELECT k, SUM(v) FROM VALUES (1, 2), (1, 3) AS t(k, v) GROUP BY k`

Command	Description	CloudWatch Journaux	Amazon S	Security Lake S	Exemple de commande
the section called “Clause LATERAL SUBQUEF	Permet à une sous-requête de la clause FROM de référencer les colonnes des éléments précédents de la même clause FROM.	N pris en charge			`SELECT * FROM tableName LATERAL ( SELECT * FROM t2 WHERE t1.c1 = t2.c1 )`

Command	Description	CloudWatch Journaux	Amazon S	Security Lake	Exemple de commande
the section called “Clause de vue latérale”	Génère une table virtuelle en appliquant une fonction de génération de table à chaque ligne d'une table de base.	Non pris en charge	S	S	`SELECT * FROM tableName LATERAL VIEW EXPLODE(ARRAY(30, 60)) tableName AS c_age LATERAL VIEW EXPLODE(ARRAY(40, 80)) AS d_age`
the section called “Prédicat LIKE”	Fait correspondre une chaîne à un modèle à l'aide de caractères génériques.	S	S	S	`SELECT method, status, request, host FROM <tableName/logGroup> WHERE method LIKE 'D%'`

Command	Description	CloudWatch Journaux	Amazon S	Security Lake S	Exemple de commande
the section called “OFFSET”	Spécifie le nombre de lignes à ignorer avant de commencer à renvoyer des lignes depuis la requête.	Pris en charge lorsqu'il est utilisé conjointement avec une `LIMIT` clause dans une requête. Par exemple • Pris en charge `SELECT * FROM Table LIMIT 100 OFFSET 10` • Non pris en			`SELECT method, status, bytes, datetime FROM <tableName/logGroup> ORDER BY datetime OFFSET 10`

Command	Description	CloudWatch Journaux	Amazon S	Security Lake	Exemple de commande
		charge SELEC * FROM Table OFFSE 10			
[the section called "Clause PIVOT"]	Transforme les lignes en colonnes faisant passer les données d'un format basé sur des lignes à un format basé sur des colonnes	N pris en charge	S	S	`SELECT * FROM ( SELECT method, status, bytes FROM <tableName/logGroup> ) AS SourceTable PIVOT ( SUM(bytes) FOR method IN ('GET', 'POST', 'PATCH', 'PUT', 'DELETE') ) AS PivotTable`

Command	Description	CloudWatch Journaux	Amazon	Security Lake	Exemple de commande

the section called “Définir les opérateurs”	Combine les résultats de deux ou plusieurs instructions SELECT (par exemple UNION, INTERSE, EXCEPT	S	S	S	`SELECT method, status, bytes FROM <tableName/logGroup> WHERE status = '416'`
					
					`UNION`
					
					`SELECT method, status, bytes FROM <tableName/logGroup> WHERE bytes > 20000`

the section called “Clause TRIER PAR”	Spécifie l'ordre dans lequel les résultats de la requête doivent être renvoyés	S	S	S	`SELECT method, status, bytes FROM <tableName/logGroup> SORT BY bytes DESC`

Command	Description	CloudWatch Journaux	Amazon S	Security Lake	Exemple de commande
the section called “UNPIVOT	Transforme les colonnes en lignes, en faisant passer les données d'un format basé sur des colonnes à un format basé sur des lignes.	N pris en charge	S	S	`SELECT status, REPLACE(method, '_bytes', '') AS request_method, bytes, datetime FROM PivotedData UNPIVOT ( bytes FOR method IN ( GET_bytes, POST_bytes, PATCH_bytes, PUT_bytes, DELETE_bytes ) ) AS UnpivotedData`

Fonctions

> **Note**
>
> Dans la colonne des exemples de commandes, remplacez le cas *<tableName/logGroup>* échéant en fonction de la source de données que vous interrogez.
>
> - Exemple de commande : `SELECT Body , Operation FROM` **<tableName/logGroup>**

- Si vous interrogez Amazon S3 ou Security Lake, utilisez : `SELECT Body , Operation FROM` **`table_name`**
- Si vous interrogez CloudWatch Logs, utilisez : `SELECT Body , Operation FROM` **`` `LogGroupA` ``**

Gramm SQL disponi e	Description	CloudV h Journa	Amazo	Securit Lake	Exemple de commande
the section called "Fonctis de chaîne	Fonctions intégrées permettant de manipuler et de transformer des chaînes et des données de texte dans des requêtes SQL. Par exemple, convertir des majuscules, combiner des chaînes, extraire des parties et nettoyer du texte.	S	S	S	`SELECT UPPER(method) AS upper_method, LOWER(host) AS lower_host FROM <tableName/logGroup>`
the section called	Fonctions intégrées pour gérer et	S	S	S	`SELECT TO_TIMESTAMP(datetime) AS timestamp,`

Gramm SQL disponi e	Description	CloudV h Journa	Amazo	Securit Lake	Exemple de commande
“Fonctis de date et d’heure	transformer les données de date et d'horodatage dans les requêtes. Par exemple, date_add, date_format, datediff et current_date.				TIMESTAMP_SECONDS(UNIX_TIMESTAMP(datetime)) AS from_seconds, UNIX_TIMESTAMP(datetime) AS to_unix, FROM_UTC_TIMESTAMP (datetime, 'PST') AS to_pst, TO_UTC_TIMESTAMP(datetime, 'EST') AS from_est FROM *<tableName/logGroup>*
the section called “Fonctis d’agrégation”	Fonctions intégrées qui effectuent des calculs sur plusieurs lignes pour produire une seule valeur résumée. Par exemple, sum, count, avg, max et min.	S	S	S	SELECT COUNT(*) AS total_records, COUNT(DISTINCT method) AS unique_methods, SUM(bytes) AS total_bytes, AVG(bytes) AS avg_bytes, MIN(bytes) AS min_bytes, MAX(bytes) AS max_bytes FROM *<tableName/logGroup>*

Gramm SQL disponi e	Description	CloudV h Journa	Amazo	Securit Lake	Exemple de commande
the section called “Fonctis conditionelles”	Fonctions intégrées qui exécutent des actions en fonction de conditions spécifiées ou qui évaluent les expressions de manière conditionnelle. Par exemple, CASE et IF.	S	S	S	`SELECT CASE WHEN method = 'GET' AND bytes < 1000 THEN 'Small Read' WHEN method = 'POST' AND bytes > 10000 THEN 'Large Write' WHEN status >= 400 OR bytes = 0 THEN 'Problem' ELSE 'Normal' END AS request_type FROM <tableName/logGroup>`

Gramm SQL disponi e	Description	CloudV h Journa	Amazo	Securit Lake	Exemple de commande
the section called “Fonctis JSON”	Fonctions intégrées pour analyser, extraire, modifier et interroger des données au format JSON dans des requêtes SQL (par exemple, from_json , to_json, get_json_ object, json_tuple) permettant de manipuler les structure s JSON dans les ensembles de données.	S	S	S	`SELECT FROM_JSON( @message, 'STRUCT< host: STRING, user-identifier: STRING, datetime: STRING, method: STRING, status: INT, bytes: INT >' ) AS parsed_json FROM <tableName/logGroup>`

Gramm SQL disponi e	Description	CloudV h Journa	Amazo	Securit Lake	Exemple de commande
the section called “Fonctis de tableau	Fonctions intégrées permettant de travaille r avec des colonnes de type tableau dans les requêtes SQL, permettan t des opérations telles que l'accès, la modificat ion et l'analyse de données de tableau (par exemple, size, explode, array_con tains).	S	S	S	`SELECT scores, size(scores) AS length, array_contains(scores, 90) AS has_90 FROM <tableName/logGroup>`

Gramm SQL disponi e	Description	CloudV h Journa	Amazo	Securit Lake	Exemple de commande
the section called “Fonctis de fenêtra ” -	Fonctions intégrées qui effectuent des calculs sur un ensemble spécifique de lignes liées à la ligne actuelle (fenêtre), permettan t des opérations telles que le classemen t, les totaux cumulés et les moyennes mobiles (par exemple, ROW_NUMBE R, RANK, LAG, LEAD)	S	S	S	`SELECT field1, field2, RANK() OVER (ORDER BY field2 DESC) AS field2Rank FROM <tableName/logGroup>`

Gramm SQL disponi e	Description	CloudV h Journa	Amazo	Securit Lake	Exemple de commande
the section called “Fonctis de conver n”	Fonctions intégrées pour convertir des données d'un type à un autre dans les requêtes SQL, permettant des transformations de type de données et des conversions de format (par exemple, CAST, TO_DATE, TO_TIMESTAMP, BINARY)	S	S	S	`SELECT CAST('123' AS INT) AS converted_number, CAST(123 AS STRING) AS converted_string FROM <tableName/logGroup>`

Gramm SQL disponi e	Description	CloudV h Journa	Amazo	Securit Lake	Exemple de commande
the section called “Fonctis de prédica	Fonctions intégrées qui évaluent les conditions et renvoient des valeurs booléennes (vrai/faux) en fonction de critères ou de modèles spécifiés (par exemple, IN, LIKE, BETWEEN, IS NULL, EXISTS)	S	S	S	`SELECT * FROM <tableName/logGroup> WHERE id BETWEEN 50000 AND 75000`

Gramm SQL disponi e	Description	CloudV h Journa	Amazo	Securit Lake	Exemple de commande
the section called “Fonctis cartogr hiques”	Applique une fonction spécifiée à chaque élément d'une collection, transformant les données en un nouvel ensemble de valeurs.	N pris en charge	S	S	`SELECT MAP_FILTER( MAP( 'method', method, 'status', CAST(status AS STRING), 'bytes', CAST(bytes AS STRING) ), (k, v) -> k IN ('method', 'status') AND v != 'null' ) AS filtered_map FROM <tableName/logGroup> WHERE status = 100`
the section called “Fonctis mathér ques”	Effectue des opérations mathémati ques sur des données numérique s, telles que le calcul de moyennes, de sommes ou de valeurs trigonomé triques.	S	S	S	`SELECT bytes, bytes + 1000 AS added, bytes - 1000 AS subtracted, bytes * 2 AS doubled, bytes / 1024 AS kilobytes, bytes % 1000 AS remainder FROM <tableName/logGroup>`

Gramm SQL disponi e	Description	CloudV h Journa	Amazo	Securit Lake	Exemple de commande
the section called “Fonctis de groupe multilo	Permet aux utilisateurs de spécifier plusieurs groupes de journaux dans une instructi on SQL SELECT	S	Ne s’applic e pas	Ne s’applic e pas	`SELECT lg1.Column1, lg1.Column2 FROM `logGroups(logGrou pIdentifier: ['LogGroup1', 'LogGroup2'])` AS lg1 WHERE lg1.Column3 = "Success"`
the section called “Fonctis du généra r”	Crée un objet itérateur qui produit une séquence de valeurs, permettan t une utilisation efficace de la mémoire dans les grands ensembles de données.	N pris en charge	S	S	`SELECT explode(array(10, 20))`

Restrictions SQL générales

Les restrictions suivantes s'appliquent lors de l'utilisation de OpenSearch SQL with CloudWatch Logs, Amazon S3 et Security Lake.

1. Vous ne pouvez utiliser qu'une seule opération JOIN dans une instruction SELECT.
2. Un seul niveau de sous-requêtes imbriquées est pris en charge.
3. Les requêtes d'instructions multiples séparées par des points-virgules ne sont pas prises en charge.
4. Les requêtes contenant des noms de champs identiques mais ne différant que dans certains cas (tels que field1 et FIELD1) ne sont pas prises en charge.

   Par exemple, les requêtes suivantes ne sont pas prises en charge :

   ```
   Select AWSAccountId, awsaccountid from LogGroup
   ```

   Cependant, la requête suivante est due au fait que le nom du champ (@logStream) est identique dans les deux groupes de journaux :

   ```
   Select a.`@logStream`, b.`@logStream` from Table A INNER Join Table B on a.id = b.id
   ```
5. Les fonctions et expressions doivent agir sur les noms de champs et faire partie d'une instruction SELECT avec un groupe de journaux spécifié dans la clause FROM.

   Par exemple, cette requête n'est pas prise en charge :

   ```
   SELECT cos(10) FROM LogGroup
   ```

   Cette requête est prise en charge :

   ```
   SELECT cos(field1) FROM LogGroup
   ```

Informations supplémentaires pour les utilisateurs de CloudWatch Logs Insights utilisant OpenSearch SQL

CloudWatch Logs prend en charge les requêtes OpenSearch SQL dans la console, l'API et la CLI de Logs Insights. Il prend en charge la plupart des commandes, notamment SELECT, FROM, WHERE, GROUP BY, HAVING, JOINS et les requêtes imbriquées, ainsi que les fonctions JSON, mathématiques, de chaîne et conditionnelles. Cependant, CloudWatch Logs ne prend en charge que les opérations de lecture et n'autorise donc pas les instructions DDL ou DML. Consultez les tableaux des sections précédentes pour obtenir la liste complète des commandes et fonctions prises en charge.

Fonctions de groupe multilog

CloudWatch Logs Insights permet d'interroger plusieurs groupes de journaux. Pour résoudre ce cas d'utilisation dans SQL, vous pouvez utiliser la `logGroups` commande. Cette commande est spécifique à l'interrogation de données dans CloudWatch Logs Insights impliquant un ou plusieurs groupes de journaux. Utilisez cette syntaxe pour interroger plusieurs groupes de journaux en les spécifiant dans la commande, au lieu d'écrire une requête pour chacun des groupes de journaux et de les combiner avec une `UNION` commande.

Syntaxe :

```
`logGroups(
    logGroupIdentifier: ['LogGroup1','LogGroup2', ...'LogGroupn']
)
```

Dans cette syntaxe, vous pouvez spécifier jusqu'à 50 groupes de journaux dans le `logGroupIndentifier` paramètre. Pour référencer des groupes de journaux dans un compte de surveillance, utilisez ARNs plutôt que des `LogGroup` noms.

Exemple de requête :

```
SELECT LG1.Column1, LG1.Column2 from `logGroups(
    logGroupIdentifier: ['LogGroup1', 'LogGroup2']
)` as LG1
WHERE LG1.Column1 = 'ABC'
```

La syntaxe suivante impliquant plusieurs groupes de journaux après l'`FROM`instruction n'est pas prise en charge lors de l'interrogation CloudWatch des journaux :

```
SELECT Column1, Column2 FROM 'LogGroup1', 'LogGroup2', ...'LogGroupn'
WHERE Column1 = 'ABC'
```

Restrictions

Lorsque vous utilisez des commandes SQL ou PPL, entourez certains champs de backticks pour les interroger. Les champs contenant des caractères spéciaux (non alphabétiques et non numériques) doivent être cochés. Par exemple, joignez `0peration.Export,` et `@message` insérez des `Test::Field` backticks. Il n'est pas nécessaire de placer des colonnes avec des noms purement alphabétiques en backticks.

Exemple de requête avec des champs simples :

```
SELECT SessionToken, Operation, StartTime  FROM `LogGroup-A`
LIMIT 1000;
```

Même requête avec des backticks ajoutés :

```
SELECT `SessionToken`, `Operation`, `StartTime`  FROM `LogGroup-A`
LIMIT 1000;
```

Pour des restrictions générales supplémentaires qui ne sont pas spécifiques aux CloudWatch journaux, consultezthe section called “Restrictions SQL générales”.

Exemples de requêtes et de quotas

> **Note**
>
> Ce qui suit s'applique à la fois aux utilisateurs de CloudWatch Logs Insights et OpenSearch aux utilisateurs interrogeant CloudWatch des données.

Pour des exemples de requêtes SQL que vous pouvez utiliser dans CloudWatch Logs, consultez la section Requêtes enregistrées et exemples de requêtes dans la console Amazon CloudWatch Logs Insights pour des exemples.

Pour plus d'informations sur les limites applicables lors de l'interrogation de CloudWatch Logs from OpenSearch Service, consultez la section Quotas de CloudWatch journaux dans le guide de l'utilisateur Amazon CloudWatch Logs. Les limites concernent le nombre de groupes de CloudWatch journaux que vous pouvez interroger, le nombre maximal de requêtes simultanées que vous pouvez exécuter, le temps d'exécution maximal des requêtes et le nombre maximal de lignes renvoyées dans les résultats. Les limites sont les mêmes quel que soit le langage que vous utilisez pour interroger les CloudWatch journaux (à savoir, OpenSearch PPL, SQL et Logs Insights).

Commandes SQL

Rubriques

- Fonctions de chaîne
- Fonctions de date et d’heure

- Fonctions d’agrégation
- Fonctions conditionnelles
- Fonctions JSON
- Fonctions de tableau
- Fonctions de fenêtrage
- Fonctions de conversion
- Fonctions de prédicat
- Fonctions cartographiques
- Fonctions mathématiques
- Fonctions du générateur
- Clause SELECT
- Clause WHERE
- Clause GROUP BY
- Clause HAVING
- Clause ORDER BY
- Clause JOIN
- Clause LIMIT
- Clause CASE
- Expression de table commune
- EXPLAIN
- Clause LATERAL SUBQUERY
- Clause de vue latérale
- Prédicat LIKE
- OFFSET
- Clause PIVOT
- Définir les opérateurs
- Clause TRIER PAR
- UNPIVOT

Fonctions de chaîne

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Fonction	Description
ascii (étoile)	Renvoie la valeur numérique du premier caractère `destr`.
base64 (bac)	Convertit l'argument d'un binaire `bin` en une chaîne de base 64.
bit_length (expr)	Renvoie la longueur en bits des données de chaîne ou le nombre de bits de données binaires.
btrim (étoile)	Supprime les espaces de début et de fin `destr`.
btrim (str, TrimStr)	Supprimez les `trimStr` caractères de début et de fin `destr`.
chaise (expr)	Renvoie le caractère ASCII dont l'équivalent binaire est. `expr` Si n est supérieur à 256, le résultat est équivalent à chr (n % 256)
char_length (expr)	Renvoie la longueur en caractères d'une chaîne de données ou le nombre d'octets de données binaires. La longueur des données de chaîne inclut les espaces de fin. La longueur des données binaires inclut des zéros binaires.
longueur_caractère (expr)	Renvoie la longueur en caractères d'une chaîne de données ou le nombre d'octets de données binaires. La longueur des données de

Fonction	Description
	chaîne inclut les espaces de fin. La longueur des données binaires inclut des zéros binaires.
chr (expr)	Renvoie le caractère ASCII dont l'équivalent binaire est. `expr` Si n est supérieur à 256, le résultat est équivalent à chr (n % 256)
concat_ws (sep [, str | array (str)] +)	Renvoie la concaténation des chaînes séparées par`sep`, en ignorant les valeurs nulles.
contient (gauche, droite)	Renvoie une valeur booléenne. La valeur est True si la droite se trouve à l'intérieur de la gauche. Renvoie NULL si l'une des expressio ns d'entrée est NULL. Dans le cas contraire, renvoie False. La gauche ou la droite doivent être de type STRING ou BINARY.
décoder (bin, charset)	Décode le premier argument à l'aide du jeu de caractères du second argument.
décoder (expr, recherche, résultat [, recherche, résultat]... [, par défaut])	Compare expr à chaque valeur de recherche dans l'ordre. Si expr est égal à une valeur de recherche, decode renvoie le résultat correspondant. Si aucune correspondance n'est trouvée, elle renvoie la valeur par défaut. Si la valeur par défaut est omise, elle renvoie null.
elt (n, entrée1, entrée2,...)	Renvoie la n -ème entrée, par exemple, renvoie `input2` quand n est égal à 2.
encoder (str, jeu de caractères)	Encode le premier argument en utilisant le jeu de caractères du second argument.

Fonction	Description
se termine par (gauche, droite)	Renvoie une valeur booléenne. La valeur est True si la gauche se termine par la droite. Renvoie NULL si l'une des expressions d'entrée est NULL. Dans le cas contraire, renvoie False. La gauche ou la droite doivent être de type STRING ou BINARY.
find_in_set (str, str_array)	Renvoie l'index (basé sur 1) de la chaîne donnée (`str`) dans la liste séparée par des virgules (). `str_array` Renvoie 0, si la chaîne n'a pas été trouvée ou si la chaîne donnée (`str`) contient une virgule.
numéro_format (expr1, expr2)	Formate le nombre `expr1` comme « #, ###, ###.## », arrondi à la décimale près. `expr2` S'il `expr2` est égal à 0, le résultat ne comporte ni virgule décimale ni partie fractionnaire. `expr2`accepte également un format spécifié par l'utilisateur. Ceci est censé fonctionner comme le FORMAT de MySQL.
format_string (strfmt, obj,...)	Renvoie une chaîne formatée à partir de chaînes de format de style printf.
initcap (star)	Renvoie `str` la première lettre de chaque mot en majuscule. Toutes les autres lettres sont en minuscules. Les mots sont délimités par des espaces blancs.
instr (étoile, substance)	Renvoie l'indice (basé sur 1) de la première occurrence de `substr` in`str`.
lcase (étoile)	Retourne `str` avec tous les caractères changés en minuscules.

Fonction	Description
gauche (str, len)	Renvoie les caractères les plus à gauche `len` (`len`peuvent être de type chaîne) de la chaîne. S'il `len` est inférieur ou égal à `0`str, le résultat est une chaîne vide.
objectif (expr)	Renvoie la longueur en caractères d'une chaîne de données ou le nombre d'octets de données binaires. La longueur des données de chaîne inclut les espaces de fin. La longueur des données binaires inclut des zéros binaires.
longueur (expr)	Renvoie la longueur en caractères d'une chaîne de données ou le nombre d'octets de données binaires. La longueur des données de chaîne inclut les espaces de fin. La longueur des données binaires inclut des zéros binaires.
levenshtein (str1, str2 [, seuil])	Renvoie la distance de Levenshtein entre les deux chaînes données. Si le seuil est défini et que la distance est supérieure à celui-ci, renvoie -1.
localiser (substr, str [, pos])	Renvoie la position de la première occurrence de `substr` in `str` after positionpos. La valeur donnée `pos` et la valeur renvoyée sont basées sur 1.
inférieur (étoile)	Retourne `str` avec tous les caractères changés en minuscules.

Fonction	Description
lpad (str, len [, pad])	Renvoie`str`, rempli `pad` à gauche avec une longueur de. `len` Si `str` cette valeur est supérieure à`len`, la valeur renvoyée est réduite en `len` caractères ou en octets. Si `pad` ce n'est pas spécifié, il `str` sera complété vers la gauche par des espaces s'il s'agit d'une chaîne de caractères, et par des zéros s'il s'agit d'une séquence d'octets.
ltrim (étoile)	Supprime les espaces principaux de`str`.
lun_check (star)	Vérifie qu'une chaîne de chiffres est valide selon l'algorithme de Luhn. Cette fonction de somme de contrôle est largement appliquée aux numéros de carte de crédit et aux numéros d'identification gouvernementaux pour distinguer les numéros valides des numéros mal orthographiés et incorrects.
masque (input [, UpperChar, LowerChar, DigitChar, OtherChar])	masque la valeur de chaîne donnée. La fonction remplace les caractères par « X » ou « x », et les nombres par « n ». Cela peut être utile pour créer des copies de tables contenant des informations sensibles supprimées.
longueur d'octet (expr)	Renvoie la longueur en octets des données de chaîne ou le nombre d'octets de données binaires.
superposition (entrée, remplacement, pos [, len])	Remplacez `input` par un nom `replace` qui commence à `pos` et qui est long`len`.
position (substr, str [, pos])	Renvoie la position de la première occurrence de `substr` in `str` after position`pos`. La valeur donnée `pos` et la valeur renvoyée sont basées sur 1.

Fonction	Description
printf (strfmt, obj,...)	Renvoie une chaîne formatée à partir de chaînes de format de style printf.
regexp_count (str, regexp)	Renvoie le nombre de fois que le modèle d'expression régulière `regexp` est mis en correspondance dans la chaîne`str`.
regexp_extract (str, regexp [, idx])	Extrayez la première chaîne `str` qui correspon d à l'`regexp`expression et qui correspond à l'index du groupe regex.
regexp_extract_all (str, regexp [, idx])	Extrayez toutes les chaînes du `str` qui correspondent à l'`regexp`expression et qui correspondent à l'index du groupe regex.
regexp_instr (str, regexp)	Recherche une expression régulière dans une chaîne et renvoie un entier indiquant la position de début de la sous-chaîne correspondante. Les positions sont basées sur 1 et non sur 0. Si aucune correspondance n'est trouvée, renvoie 0.
regexp_replace (str, regexp, rep [, position])	Remplace toutes les sous-chaînes `str` correspondant `regexp` par`rep`.
regexp_substr (str, regexp)	Renvoie la sous-chaîne qui correspond à l'expression régulière contenue `regexp` dans la chaîne`str`. Si l'expression régulière n'est pas trouvée, le résultat est nul.
répéter (str, n)	Renvoie la chaîne qui répète n fois la valeur de chaîne donnée.
remplacer (str, search [, remplacer])	Remplace toutes les occurrences de `search` with`replace`.

Fonction	Description
droite (star, len)	Renvoie les caractères les plus à droite `len` (`len`peut être de type chaîne) de la chaîne. S'il `len` est inférieur ou égal à `0``str`, le résultat est une chaîne vide.
rpad (str, len [, pad])	Renvoie`str`, rembourré `pad` à droite avec une longueur de. `len` Si `str` cette valeur est supérieure à`len`, la valeur renvoyée est réduite en `len` caractères. Si `pad` ce n'est pas spécifié, il `str` sera complété vers la droite par des espaces s'il s'agit d'une chaîne de caractères, et par des zéros s'il s'agit d'une chaîne binaire.
rtrim (étoile)	Supprime les espaces de fin de`str`.
phrases (str [, lang, country])	Se `str` divise en un tableau de mots.
soundex (star)	Renvoie le code Soundex de la chaîne.
espace (n)	Renvoie une chaîne composée d'nespaces.
split (str, regex, limite)	Divise `str` les occurrences qui correspondent `regex` et renvoie un tableau d'une longueur maximale de `limit`
split_part (str, délimiteur, PartNum)	Divise `str` par délimiteur et renvoie la partie demandée de la division (base 1). Si une entrée est nulle, renvoie null. si elle `partNum` est hors de portée des parties divisées, renvoie une chaîne vide. Si la `partNum` valeur est 0, renvoie une erreur. S'il `partNum` est négatif, les parties sont comptées à rebours à partir de la fin de la chaîne. S'il s'`delimiter` agit d'une chaîne vide, elle n'`str`est pas divisée.

Fonction	Description
commence par (gauche, droite)	Renvoie une valeur booléenne. La valeur est True si la gauche commence par la droite. Renvoie NULL si l'une des expressions d'entrée est NULL. Dans le cas contraire, renvoie False. La gauche ou la droite doivent être de type STRING ou BINARY.
substr (str, pos [, len])	Renvoie la sous-chaîne `str` dont la longueur commence à `pos` et est longue`len`, ou la tranche de tableau d'octets qui commence à `pos` et est de longueur`len`.
substr (str FROM pos [POUR len]])	Renvoie la sous-chaîne `str` dont la longueur commence à `pos` et est longue`len`, ou la tranche de tableau d'octets qui commence à `pos` et est de longueur`len`.
sous-chaîne (str, pos [, len])	Renvoie la sous-chaîne `str` dont la longueur commence à `pos` et est longue`len`, ou la tranche de tableau d'octets qui commence à `pos` et est de longueur`len`.
sous-chaîne (str FROM pos [POUR len]])	Renvoie la sous-chaîne `str` dont la longueur commence à `pos` et est longue`len`, ou la tranche de tableau d'octets qui commence à `pos` et est de longueur`len`.

Fonction	Description
substring_index (str, delim, count)	Renvoie la sous-chaîne située `str` avant les `count` occurrences du `delim` délimiteur. S'il `count` est positif, tout ce qui se trouve à gauche du délimiteur final (en partant de la gauche) est renvoyé. S'il `count` est négatif, tout ce qui se trouve à droite du délimiteu r final (en partant de la droite) est renvoyé. La fonction substring_index effectue une correspondance entre majuscules et minuscule s lors de la recherche. `delim`
to_binary (str [, fmt])	Convertit l'entrée `str` en une valeur binaire basée sur la valeur fournie`fmt`. `fmt`peut être une chaîne littérale insensible aux majuscule s et minuscules de « hex », « utf-8 », « utf8 » ou « base64 ». Par défaut, le format binaire de conversion est « hexadécimal » s'il `fmt` est omis. La fonction renvoie NULL si au moins un des paramètres d'entrée est NULL.

Fonction	Description
to_char (NumberExpr, FormatExpr)	Convertir `numberExpr` en une chaîne basée sur`formatExpr` . Lance une exception en cas d'échec de la conversion. Le format peut être composé des caractères suivants, sans distinction majuscules/minuscules : « 0 » ou « 9 » : Spécifie un chiffre attendu compris entre 0 et 9. Une séquence de 0 ou 9 dans la chaîne de format correspond à une séquence de chiffres dans la valeur d'entrée, générant une chaîne de résultat de la même longueur que la séquence correspondante dans la chaîne de format. La chaîne de résultat est remplie de zéros à gauche si la séquence 0/9 comprend plus de chiffres que la partie correspondante de la valeur décimale, commence par 0 et se situe avant le point décimal. Sinon, il est rempli d'espaces. '.' ou 'D' : Spécifie la position du point décimal (facultatif, autorisé une seule fois). ',' ou 'G' : Spécifie la position du séparateur de regroupement (milliers) (,). Il doit y avoir un 0 ou un 9 à gauche et à droite de chaque séparateur de regroupement. '

Fonction	Description
to_number (expr, fmt)	Convertissez la chaîne « expr » en un nombre basé sur le format de chaîne « fmt ». Lance une exception en cas d'échec de la conversio n. Le format peut être composé des caractères suivants, sans distinction majuscules/minuscu les : « 0 » ou « 9 » : Spécifie un chiffre attendu compris entre 0 et 9. Une séquence de 0 ou 9 dans la chaîne de format correspond à une séquence de chiffres dans la chaîne d'entrée. Si la séquence 0/9 commence par 0 et se situe avant le point décimal, elle ne peut correspondre qu'à une séquence de chiffres de même taille. Sinon, si la séquence commence par 9 ou après la virgule décimale, elle peut correspondre à une séquence de chiffres de taille identique ou inférieure. '.' ou 'D' : Spécifie la position du point décimal (facultatif, autorisé une seule fois). ',' ou 'G' : Spécifie la position du séparateur de regroupement (milliers) (,). Il doit y avoir un 0 ou un 9 à gauche et à droite de chaque séparateur de regroupem ent. « expr » doit correspondre au séparateur de regroupement correspondant à la taille du numéro. '

Fonction	Description
to_varchar (NumberExpr, FormatExpr)	Convertir `numberExpr` en une chaîne basée sur`formatExpr` . Lance une exception en cas d'échec de la conversion. Le format peut être composé des caractères suivants, sans distinction majuscules/minuscules : « 0 » ou « 9 » : Spécifie un chiffre attendu compris entre 0 et 9. Une séquence de 0 ou 9 dans la chaîne de format correspond à une séquence de chiffres dans la valeur d'entrée, générant une chaîne de résultat de la même longueur que la séquence correspondante dans la chaîne de format. La chaîne de résultat est remplie de zéros à gauche si la séquence 0/9 comprend plus de chiffres que la partie correspondante de la valeur décimale, commence par 0 et se situe avant le point décimal. Sinon, il est rempli d'espaces. '.' ou 'D' : Spécifie la position du point décimal (facultatif, autorisé une seule fois). ',' ou 'G' : Spécifie la position du séparateur de regroupement (milliers) (,). Il doit y avoir un 0 ou un 9 à gauche et à droite de chaque séparateur de regroupement. '
traduire (entrée, de, vers)	Traduit la `input` chaîne en remplaçant les caractères présents dans la `from` chaîne par les caractères correspondants de la `to` chaîne.
trim (étoile)	Supprime les espaces de début et de fin de`str`.
trim (LES DEUX À PARTIR DE str)	Supprime les espaces de début et de fin de`str`.
trim (EN PARTANT DE L'ÉTOILE)	Supprime les espaces principaux de`str`.
trim (À LA TRAÎNE DE la rue)	Supprime les espaces de fin de`str`.

Fonction	Description
trim (TrimStr FROM str)	Supprimez les `trimStr` caractères de début et de fin de`str`.
trim (À LA FOIS TRIMSTR ET str)	Supprimez les `trimStr` caractères de début et de fin de`str`.
trim (PREMIER TRIMSTR À PARTIR DE str)	Supprimez les premiers `trimStr` caractères de`str`.
trim (TIRANT TRIMSTR DE str)	Supprimez les derniers `trimStr` caractères de. `str`
try_to_binary (str [, fmt])	Il s'agit d'une version spéciale `to_binary` qui effectue la même opération, mais renvoie une valeur NULL au lieu de générer une erreur si la conversion ne peut pas être effectuée.
try_to_number (expr, fmt)	Convertissez la chaîne « expr » en nombre en fonction du format `fmt` de chaîne. Renvoie NULL si la chaîne « expr » ne correspond pas au format attendu. Le format suit la même sémantique que la fonction to_number.
Ucase (étoile)	Retourne `str` avec tous les caractères changés en majuscules.
unbase64 (str)	Convertit l'argument d'une chaîne de base 64 `str` en binaire.
supérieur (étoile)	Retourne `str` avec tous les caractères changés en majuscules.

Exemples

```
-- ascii
SELECT ascii('222');
```

```
+----------+
|ascii(222)|
+----------+
|        50|
+----------+
SELECT ascii(2);
+--------+
|ascii(2)|
+--------+
|      50|
+--------+
-- base64
SELECT base64('Feathers');
+-----------------+
|base64(Feathers)|
+-----------------+
|     RmVhdGhlcnM=|
+-----------------+
SELECT base64(x'537061726b2053514c');
+-----------------------------+
|base64(X'537061726B2053514C')|
+-----------------------------+
|                 U3BhcmsgU1FM|
+-----------------------------+
-- bit_length
SELECT bit_length('Feathers');
+---------------------+
|bit_length(Feathers)|
+---------------------+
|                   64|
+---------------------+
SELECT bit_length(x'537061726b2053514c');
+---------------------------------+
|bit_length(X'537061726B2053514C')|
+---------------------------------+
|                               72|
+---------------------------------+
-- btrim
SELECT btrim('    Feathers   ');
+----------------------+
|btrim(    Feathers   )|
+----------------------+
|              Feathers|
+----------------------+
```

```
SELECT btrim(encode('    Feathers   ', 'utf-8'));
+-----------------------------------+
|btrim(encode(    Feathers   , utf-8))|
+-----------------------------------+
|                           Feathers|
+-----------------------------------+
SELECT btrim('Feathers', 'Fe');
+---------------------+
|btrim(Alphabet, Al)|
+---------------------+
|               athers|
+---------------------+
SELECT btrim(encode('Feathers', 'utf-8'), encode('Al', 'utf-8'));
+---------------------------------------------------+
|btrim(encode(Feathers, utf-8), encode(Al, utf-8))|
+---------------------------------------------------+
|                                             athers|
+---------------------------------------------------+
-- char
SELECT char(65);
+--------+
|char(65)|
+--------+
|       A|
+--------+
-- char_length
SELECT char_length('Feathers ');
+-----------------------+
|char_length(Feathers )|
+-----------------------+
|                      9 |
+-----------------------+
SELECT char_length(x'537061726b2053514c');
+---------------------------------+
|char_length(X'537061726B2053514C')|
+---------------------------------+
|                                9|
+---------------------------------+
SELECT CHAR_LENGTH('Feathers ');
+-----------------------+
|char_length(Feathers )|
+-----------------------+
|                      9|
+-----------------------+
```

```
SELECT CHARACTER_LENGTH('Feathers ');
+---------------------------+
|character_length(Feathers )|
+---------------------------+
|                          9|
+---------------------------+
-- character_length
SELECT character_length('Feathers ');
+---------------------------+
|character_length(Feathers )|
+---------------------------+
|                          9|
+---------------------------+
SELECT character_length(x'537061726b2053514c');
+--------------------------------------+
|character_length(X'537061726B2053514C')|
+--------------------------------------+
|                                     9|
+--------------------------------------+
SELECT CHAR_LENGTH('Feathers ');
+-----------------------+
|char_length(Feathers )|
+-----------------------+
|                      9|
+-----------------------+
SELECT CHARACTER_LENGTH('Feathers ');
+---------------------------+
|character_length(Feathers )|
+---------------------------+
|                          9|
+---------------------------+
-- chr
SELECT chr(65);
+-------+
|chr(65)|
+-------+
|      A|
+-------+
-- concat_ws
SELECT concat_ws(' ', 'Fea', 'thers');
+-----------------------+
|concat_ws( , Fea, thers)|
+-----------------------+
|               Feathers|
```

```
+------------------------+
SELECT concat_ws('s');
+------------+
|concat_ws(s)|
+------------+
|            |
+------------+
SELECT concat_ws('/', 'foo', null, 'bar');
+----------------------------+
|concat_ws(/, foo, NULL, bar)|
+----------------------------+
|                     foo/bar|
+----------------------------+
SELECT concat_ws(null, 'Fea', 'thers');
+---------------------------+
|concat_ws(NULL, Fea, thers)|
+---------------------------+
|                       NULL|
+---------------------------+
-- contains
SELECT contains('Feathers', 'Fea');
+--------------------------+
|contains(Feathers, Fea)|
+--------------------------+
|                      true|
+--------------------------+
SELECT contains('Feathers', 'SQL');
+--------------------------+
|contains(Feathers, SQL)|
+--------------------------+
|                     false|
+--------------------------+
SELECT contains('Feathers', null);
+-------------------------+
|contains(Feathers, NULL)|
+-------------------------+
|                     NULL|
+-------------------------+
SELECT contains(x'537061726b2053514c', x'537061726b');
+-----------------------------------------------+
|contains(X'537061726B2053514C', X'537061726B')|
+-----------------------------------------------+
|                                           true|
+-----------------------------------------------+
```

```
-- decode
SELECT decode(encode('abc', 'utf-8'), 'utf-8');
+---------------------------------+
|decode(encode(abc, utf-8), utf-8)|
+---------------------------------+
|                              abc|
+---------------------------------+
SELECT decode(2, 1, 'Southlake', 2, 'San Francisco', 3, 'New Jersey', 4, 'Seattle',
 'Non domestic');
+-------------------------------------------------------------------------------+
|decode(2, 1, Southlake, 2, San Francisco, 3, New Jersey, 4, Seattle, Non domestic)|
+-------------------------------------------------------------------------------+
|                                                                  San Francisco|
+-------------------------------------------------------------------------------+
SELECT decode(6, 1, 'Southlake', 2, 'San Francisco', 3, 'New Jersey', 4, 'Seattle',
 'Non domestic');
+-------------------------------------------------------------------------------+
|decode(6, 1, Southlake, 2, San Francisco, 3, New Jersey, 4, Seattle, Non domestic)|
+-------------------------------------------------------------------------------+
|                                                                   Non domestic|
+-------------------------------------------------------------------------------+
SELECT decode(6, 1, 'Southlake', 2, 'San Francisco', 3, 'New Jersey', 4, 'Seattle');
+-----------------------------------------------------------------+
|decode(6, 1, Southlake, 2, San Francisco, 3, New Jersey, 4, Seattle)|
+-----------------------------------------------------------------+
|                                                             NULL|
+-----------------------------------------------------------------+
SELECT decode(null, 6, 'Fea', NULL, 'thers', 4, 'rock');
+----------------------------------------+
|decode(NULL, 6, Fea, NULL, thers, 4, rock)|
+----------------------------------------+
|                                   thers|
+----------------------------------------+
-- elt
SELECT elt(1, 'scala', 'java');
+-------------------+
|elt(1, scala, java)|
+-------------------+
|              scala|
+-------------------+
SELECT elt(2, 'a', 1);
+------------+
|elt(2, a, 1)|
+------------+
```

```
|           1|
+------------+
-- encode
SELECT encode('abc', 'utf-8');
+------------------+
|encode(abc, utf-8)|
+------------------+
|        [61 62 63]|
+------------------+
-- endswith
SELECT endswith('Feathers', 'ers');
+------------------------+
|endswith(Feathers, ers)|
+------------------------+
|                    true|
+------------------------+
SELECT endswith('Feathers', 'SQL');
+-------------------------+
|endswith(Feathers, SQL)|
+-------------------------+
|                     false|
+-------------------------+
SELECT endswith('Feathers', null);
+-------------------------+
|endswith(Feathers, NULL)|
+-------------------------+
|                     NULL|
+-------------------------+
SELECT endswith(x'537061726b2053514c', x'537061726b');
+-----------------------------------------------+
|endswith(X'537061726B2053514C', X'537061726B')|
+-----------------------------------------------+
|                                          false|
+-----------------------------------------------+
SELECT endswith(x'537061726b2053514c', x'53514c');
+---------------------------------------+
|endswith(X'537061726B2053514C', X'53514C')|
+---------------------------------------+
|                                   true|
+---------------------------------------+
-- find_in_set
SELECT find_in_set('ab','abc,b,ab,c,def');
+-------------------------------+
|find_in_set(ab, abc,b,ab,c,def)|
```

```
+------------------------------+
|                             3|
+------------------------------+
-- format_number
SELECT format_number(12332.123456, 4);
+------------------------------+
|format_number(12332.123456, 4)|
+------------------------------+
|                  12,332.1235|
+------------------------------+
SELECT format_number(12332.123456, '##################.###');
+---------------------------------------------------+
|format_number(12332.123456, ##################.###)|
+---------------------------------------------------+
|                                           12332.123|
+---------------------------------------------------+
-- format_string
SELECT format_string("Hello World %d %s", 100, "days");
+-------------------------------------------+
|format_string(Hello World %d %s, 100, days)|
+-------------------------------------------+
|                       Hello World 100 days|
+-------------------------------------------+
-- initcap
SELECT initcap('Feathers');
+------------------+
|initcap(Feathers)|
+------------------+
|         Feathers|
+------------------+
-- instr
SELECT instr('Feathers', 'ers');
+--------------------+
|instr(Feathers, ers)|
+--------------------+
|                   6|
+--------------------+
-- lcase
SELECT lcase('Feathers');
+---------------+
|lcase(Feathers)|
+---------------+
|       feathers|
+---------------+
```

```
-- left
SELECT left('Feathers', 3);
+------------------+
|left(Feathers, 3)|
+------------------+
|               Fea|
+------------------+
SELECT left(encode('Feathers', 'utf-8'), 3);
+---------------------------------+
|left(encode(Feathers, utf-8), 3)|
+---------------------------------+
|                       [RmVh]|
+---------------------------------+
-- len
SELECT len('Feathers ');
+---------------+
|len(Feathers )|
+---------------+
|             9|
+---------------+
SELECT len(x'537061726b2053514c');
+--------------------------+
|len(X'537061726B2053514C')|
+--------------------------+
|                         9|
+--------------------------+
SELECT CHAR_LENGTH('Feathers ');
+-----------------------+
|char_length(Feathers )|
+-----------------------+
|                      9|
+-----------------------+
SELECT CHARACTER_LENGTH('Feathers ');
+----------------------------+
|character_length(Feathers )|
+----------------------------+
|                           9|
+----------------------------+
-- length
SELECT length('Feathers ');
+------------------+
|length(Feathers )|
+------------------+
|                9|
```

```
+------------------+
SELECT length(x'537061726b2053514c');
+----------------------------+
|length(X'537061726B2053514C')|
+----------------------------+
|                           9|
+----------------------------+
SELECT CHAR_LENGTH('Feathers ');
+----------------------+
|char_length(Feathers )|
+----------------------+
|                     9|
+----------------------+
SELECT CHARACTER_LENGTH('Feathers ');
+---------------------------+
|character_length(Feathers )|
+---------------------------+
|                          9|
+---------------------------+
-- levenshtein
SELECT levenshtein('kitten', 'sitting');
+---------------------------+
|levenshtein(kitten, sitting)|
+---------------------------+
|                           3|
+---------------------------+
SELECT levenshtein('kitten', 'sitting', 2);
+-------------------------------+
|levenshtein(kitten, sitting, 2)|
+-------------------------------+
|                             -1|
+-------------------------------+
-- locate
SELECT locate('bar', 'foobarbar');
+-------------------------+
|locate(bar, foobarbar, 1)|
+-------------------------+
|                        4|
+-------------------------+
SELECT locate('bar', 'foobarbar', 5);
+-------------------------+
|locate(bar, foobarbar, 5)|
+-------------------------+
|                        7|
```

```
+-------------------------+
SELECT POSITION('bar' IN 'foobarbar');
+-------------------------+
|locate(bar, foobarbar, 1)|
+-------------------------+
|                        4|
+-------------------------+
-- lower
SELECT lower('Feathers');
+---------------+
|lower(Feathers)|
+---------------+
|       feathers|
+---------------+
-- lpad
SELECT lpad('hi', 5, '??');
+---------------+
|lpad(hi, 5, ??)|
+---------------+
|          ???hi|
+---------------+
SELECT lpad('hi', 1, '??');
+---------------+
|lpad(hi, 1, ??)|
+---------------+
|              h|
+---------------+
SELECT lpad('hi', 5);
+--------------+
|lpad(hi, 5,  )|
+--------------+
|            hi|
+--------------+
SELECT hex(lpad(unhex('aabb'), 5));
+--------------------------------+
|hex(lpad(unhex(aabb), 5, X'00'))|
+--------------------------------+
|                      000000AABB|
+--------------------------------+
SELECT hex(lpad(unhex('aabb'), 5, unhex('1122')));
+-------------------------------------+
|hex(lpad(unhex(aabb), 5, unhex(1122)))|
+-------------------------------------+
|                           112211AABB|
```

```
+--------------------------------------+
-- ltrim
SELECT ltrim('    Feathers   ');
+----------------------+
|ltrim(    Feathers   )|
+----------------------+
|           Feathers   |
+----------------------+
-- luhn_check
SELECT luhn_check('8112189876');
+----------------------+
|luhn_check(8112189876)|
+----------------------+
|                  true|
+----------------------+
SELECT luhn_check('79927398713');
+-----------------------+
|luhn_check(79927398713)|
+-----------------------+
|                   true|
+-----------------------+
SELECT luhn_check('79927398714');
+-----------------------+
|luhn_check(79927398714)|
+-----------------------+
|                  false|
+-----------------------+
-- mask
SELECT mask('abcd-EFGH-8765-4321');
+---------------------------------------+
|mask(abcd-EFGH-8765-4321, X, x, n, NULL)|
+---------------------------------------+
|                    xxxx-XXXX-nnnn-nnnn|
+---------------------------------------+
SELECT mask('abcd-EFGH-8765-4321', 'Q');
+---------------------------------------+
|mask(abcd-EFGH-8765-4321, Q, x, n, NULL)|
+---------------------------------------+
|                    xxxx-QQQQ-nnnn-nnnn|
+---------------------------------------+
SELECT mask('AbCD123-@$#', 'Q', 'q');
+--------------------------------+
|mask(AbCD123-@$#, Q, q, n, NULL)|
+--------------------------------+
```

```
|                     QqQQnnn-@$#|
+--------------------------------+
SELECT mask('AbCD123-@$#');
+--------------------------------+
|mask(AbCD123-@$#, X, x, n, NULL)|
+--------------------------------+
|                     XxXXnnn-@$#|
+--------------------------------+
SELECT mask('AbCD123-@$#', 'Q');
+--------------------------------+
|mask(AbCD123-@$#, Q, x, n, NULL)|
+--------------------------------+
|                     QxQQnnn-@$#|
+--------------------------------+
SELECT mask('AbCD123-@$#', 'Q', 'q');
+--------------------------------+
|mask(AbCD123-@$#, Q, q, n, NULL)|
+--------------------------------+
|                     QqQQnnn-@$#|
+--------------------------------+
SELECT mask('AbCD123-@$#', 'Q', 'q', 'd');
+--------------------------------+
|mask(AbCD123-@$#, Q, q, d, NULL)|
+--------------------------------+
|                     QqQQddd-@$#|
+--------------------------------+
SELECT mask('AbCD123-@$#', 'Q', 'q', 'd', 'o');
+-----------------------------+
|mask(AbCD123-@$#, Q, q, d, o)|
+-----------------------------+
|                  QqQQdddoooo|
+-----------------------------+
SELECT mask('AbCD123-@$#', NULL, 'q', 'd', 'o');
+--------------------------------+
|mask(AbCD123-@$#, NULL, q, d, o)|
+--------------------------------+
|                     AqCDdddoooo|
+--------------------------------+
SELECT mask('AbCD123-@$#', NULL, NULL, 'd', 'o');
+-----------------------------------+
|mask(AbCD123-@$#, NULL, NULL, d, o)|
+-----------------------------------+
|                        AbCDdddoooo|
+-----------------------------------+
```

```
SELECT mask('AbCD123-@$#', NULL, NULL, NULL, 'o');
+-------------------------------------+
|mask(AbCD123-@$#, NULL, NULL, NULL, o)|
+-------------------------------------+
|                           AbCD123oooo|
+-------------------------------------+
SELECT mask(NULL, NULL, NULL, NULL, 'o');
+-------------------------------+
|mask(NULL, NULL, NULL, NULL, o)|
+-------------------------------+
|                           NULL|
+-------------------------------+
SELECT mask(NULL);
+-------------------------+
|mask(NULL, X, x, n, NULL)|
+-------------------------+
|                     NULL|
+-------------------------+
SELECT mask('AbCD123-@$#', NULL, NULL, NULL, NULL);
+----------------------------------------+
|mask(AbCD123-@$#, NULL, NULL, NULL, NULL)|
+----------------------------------------+
|                             AbCD123-@$#|
+----------------------------------------+
-- octet_length
SELECT octet_length('Feathers');
+----------------------+
|octet_length(Feathers)|
+----------------------+
|                     8|
+----------------------+
SELECT octet_length(x'537061726b2053514c');
+----------------------------------+
|octet_length(X'537061726B2053514C')|
+----------------------------------+
|                                 9|
+----------------------------------+
-- overlay
SELECT overlay('Feathers' PLACING '_' FROM 6);
+---------------------------+
|overlay(Feathers, _, 6, -1)|
+---------------------------+
|                  Feathe_ers|
+---------------------------+
```

```
SELECT overlay('Feathers' PLACING 'ures' FROM 5);
+------------------------------+
|overlay(Feathers, ures, 5, -1)|
+------------------------------+
|                     Features  |
+------------------------------+
-- position
SELECT position('bar', 'foobarbar');
+---------------------------+
|position(bar, foobarbar, 1)|
+---------------------------+
|                          4|
+---------------------------+
SELECT position('bar', 'foobarbar', 5);
+---------------------------+
|position(bar, foobarbar, 5)|
+---------------------------+
|                          7|
+---------------------------+
SELECT POSITION('bar' IN 'foobarbar');
+-------------------------+
|locate(bar, foobarbar, 1)|
+-------------------------+
|                        4|
+-------------------------+
-- printf
SELECT printf("Hello World %d %s", 100, "days");
+-----------------------------------+
|printf(Hello World %d %s, 100, days)|
+-----------------------------------+
|                Hello World 100 days|
+-----------------------------------+
-- regexp_count
SELECT regexp_count('Steven Jones and Stephen Smith are the best players', 'Ste(v|
ph)en');
+-----------------------------------------------------------------------------+
|regexp_count(Steven Jones and Stephen Smith are the best players, Ste(v|ph)en)|
+-----------------------------------------------------------------------------+
|                                                                            2|
+-----------------------------------------------------------------------------+
SELECT regexp_count('abcdefghijklmnopqrstuvwxyz', '[a-z]{3}');
+-------------------------------------------------+
|regexp_count(abcdefghijklmnopqrstuvwxyz, [a-z]{3})|
+-------------------------------------------------+
```

```
|                                                  8|
+---------------------------------------------------+
-- regexp_extract
SELECT regexp_extract('100-200', '(\\d+)-(\\d+)', 1);
+--------------------------------------+
|regexp_extract(100-200, (\d+)-(\d+), 1)|
+--------------------------------------+
|                                   100|
+--------------------------------------+
-- regexp_extract_all
SELECT regexp_extract_all('100-200, 300-400', '(\\d+)-(\\d+)', 1);
+-----------------------------------------------------+
|regexp_extract_all(100-200, 300-400, (\d+)-(\d+), 1)|
+-----------------------------------------------------+
|                                          [100, 300]|
+-----------------------------------------------------+
-- regexp_instr
SELECT regexp_instr('user@opensearch.org', '@[^.]*');
+-----------------------------------------------+
|regexp_instr(user@opensearch.org, @[^.]*, 0)|
+-----------------------------------------------+
|                                              5|
+-----------------------------------------------+
-- regexp_replace
SELECT regexp_replace('100-200', '(\\d+)', 'num');
+--------------------------------------+
|regexp_replace(100-200, (\d+), num, 1)|
+--------------------------------------+
|                               num-num|
+--------------------------------------+
-- regexp_substr
SELECT regexp_substr('Steven Jones and Stephen Smith are the best players', 'Ste(v|
ph)en');
+-----------------------------------------------------------------------------------+
|regexp_substr(Steven Jones and Stephen Smith are the best players, Ste(v|ph)en)|
+-----------------------------------------------------------------------------------+
|                                                                             Steven|
+-----------------------------------------------------------------------------------+
SELECT regexp_substr('Steven Jones and Stephen Smith are the best players', 'Jeck');
+-----------------------------------------------------------------------+
|regexp_substr(Steven Jones and Stephen Smith are the best players, Jeck)|
+-----------------------------------------------------------------------+
|                                                                   NULL|
+-----------------------------------------------------------------------+
```

```
-- repeat
SELECT repeat('123', 2);
+--------------+
|repeat(123, 2)|
+--------------+
|        123123|
+--------------+
-- replace
SELECT replace('ABCabc', 'abc', 'DEF');
+-------------------------+
|replace(ABCabc, abc, DEF)|
+-------------------------+
|                   ABCDEF|
+-------------------------+
-- right
SELECT right('Feathers', 3);
+-------------------+
|right(Feathers, 3)|
+-------------------+
|                ers|
+-------------------+
-- rpad
SELECT rpad('hi', 5, '??');
+---------------+
|rpad(hi, 5, ??)|
+---------------+
|          hi???|
+---------------+
SELECT rpad('hi', 1, '??');
+---------------+
|rpad(hi, 1, ??)|
+---------------+
|              h|
+---------------+
SELECT rpad('hi', 5);
+--------------+
|rpad(hi, 5,  )|
+--------------+
|         hi   |
+--------------+
SELECT hex(rpad(unhex('aabb'), 5));
+--------------------------------+
|hex(rpad(unhex(aabb), 5, X'00'))|
+--------------------------------+
```

```
|                      AABB000000|
+--------------------------------+
SELECT hex(rpad(unhex('aabb'), 5, unhex('1122')));
+-------------------------------------+
|hex(rpad(unhex(aabb), 5, unhex(1122)))|
+-------------------------------------+
|                           AABB112211|
+-------------------------------------+
-- rtrim
SELECT rtrim('    Feathers   ');
+----------------------+
|rtrim(    Feathers   )|
+----------------------+
|              Feathers|
+----------------------+
-- sentences
SELECT sentences('Hi there! Good morning.');
+-------------------------------------+
|sentences(Hi there! Good morning., , )|
+-------------------------------------+
|                  [[Hi, there], [Go...|
+-------------------------------------+
-- soundex
SELECT soundex('Miller');
+---------------+
|soundex(Miller)|
+---------------+
|           M460|
+---------------+
-- space
SELECT concat(space(2), '1');
+-------------------+
|concat(space(2), 1)|
+-------------------+
|                  1|
+-------------------+
-- split
SELECT split('oneAtwoBthreeC', '[ABC]');
+--------------------------------+
|split(oneAtwoBthreeC, [ABC], -1)|
+--------------------------------+
|             [one, two, three, ]|
+--------------------------------+
SELECT split('oneAtwoBthreeC', '[ABC]', -1);
```

```
+-------------------------------+
|split(oneAtwoBthreeC, [ABC], -1)|
+-------------------------------+
|             [one, two, three, ]|
+-------------------------------+
SELECT split('oneAtwoBthreeC', '[ABC]', 2);
+-------------------------------+
|split(oneAtwoBthreeC, [ABC], 2)|
+-------------------------------+
|              [one, twoBthreeC]|
+-------------------------------+
-- split_part
SELECT split_part('11.12.13', '.', 3);
+--------------------------+
|split_part(11.12.13, ., 3)|
+--------------------------+
|                        13|
+--------------------------+
-- startswith
SELECT startswith('Feathers', 'Fea');
+---------------------------+
|startswith(Feathers, Fea)|
+---------------------------+
|                       true|
+---------------------------+
SELECT startswith('Feathers', 'SQL');
+-------------------------+
|startswith(Feathers, SQL)|
+-------------------------+
|                     false|
+-------------------------+
SELECT startswith('Feathers', null);
+-------------------------+
|startswith(Feathers, NULL)|
+-------------------------+
|                      NULL|
+-------------------------+
SELECT startswith(x'537061726b2053514c', x'537061726b');
+------------------------------------------------+
|startswith(X'537061726B2053514C', X'537061726B')|
+------------------------------------------------+
|                                            true|
+------------------------------------------------+
SELECT startswith(x'537061726b2053514c', x'53514c');
```

```
+-----------------------------------------+
|startswith(X'537061726B2053514C', X'53514C')|
+-----------------------------------------+
|                                    false|
+-----------------------------------------+
-- substr
SELECT substr('Feathers', 5);
+--------------------------------+
|substr(Feathers, 5, 2147483647)|
+--------------------------------+
|                           hers |
+--------------------------------+
SELECT substr('Feathers', -3);
+---------------------------------+
|substr(Feathers, -3, 2147483647)|
+---------------------------------+
|                              ers|
+---------------------------------+
SELECT substr('Feathers', 5, 1);
+-----------------------+
|substr(Feathers, 5, 1)|
+-----------------------+
|                      h|
+-----------------------+
SELECT substr('Feathers' FROM 5);
+----------------------------------+
|substring(Feathers, 5, 2147483647)|
+----------------------------------+
|                              hers |
+----------------------------------+
SELECT substr('Feathers' FROM -3);
+-----------------------------------+
|substring(Feathers, -3, 2147483647)|
+-----------------------------------+
|                                ers|
+-----------------------------------+
SELECT substr('Feathers' FROM 5 FOR 1);
+--------------------------+
|substring(Feathers, 5, 1)|
+--------------------------+
|                         h|
+--------------------------+
-- substring
SELECT substring('Feathers', 5);
```

```
+----------------------------------+
|substring(Feathers, 5, 2147483647)|
+----------------------------------+
|                              hers |
+----------------------------------+
SELECT substring('Feathers', -3);
+-----------------------------------+
|substring(Feathers, -3, 2147483647)|
+-----------------------------------+
|                                ers|
+-----------------------------------+
SELECT substring('Feathers', 5, 1);
+--------------------------+
|substring(Feathers, 5, 1)|
+--------------------------+
|                         h|
+--------------------------+
SELECT substring('Feathers' FROM 5);
+----------------------------------+
|substring(Feathers, 5, 2147483647)|
+----------------------------------+
|                              hers |
+----------------------------------+
SELECT substring('Feathers' FROM -3);
+-----------------------------------+
|substring(Feathers, -3, 2147483647)|
+-----------------------------------+
|                                ers|
+-----------------------------------+
SELECT substring('Feathers' FROM 5 FOR 1);
+--------------------------+
|substring(Feathers, 5, 1)|
+--------------------------+
|                         h|
+--------------------------+
-- substring_index
SELECT substring_index('www.apache.org', '.', 2);
+-------------------------------------+
|substring_index(www.apache.org, ., 2)|
+-------------------------------------+
|                           www.apache|
+-------------------------------------+
-- to_binary
SELECT to_binary('abc', 'utf-8');
```

```
+---------------------+
|to_binary(abc, utf-8)|
+---------------------+
|           [61 62 63]|
+---------------------+
-- to_char
SELECT to_char(454, '999');
+-----------------+
|to_char(454, 999)|
+-----------------+
|              454|
+-----------------+
SELECT to_char(454.00, '000D00');
+-----------------------+
|to_char(454.00, 000D00)|
+-----------------------+
|                 454.00|
+-----------------------+
SELECT to_char(12454, '99G999');
+----------------------+
|to_char(12454, 99G999)|
+----------------------+
|                12,454|
+----------------------+
SELECT to_char(78.12, '$99.99');
+----------------------+
|to_char(78.12, $99.99)|
+----------------------+
|                $78.12|
+----------------------+
SELECT to_char(-12454.8, '99G999D9S');
+----------------------------+
|to_char(-12454.8, 99G999D9S)|
+----------------------------+
|                   12,454.8-|
+----------------------------+
-- to_number
SELECT to_number('454', '999');
+-------------------+
|to_number(454, 999)|
+-------------------+
|                454|
+-------------------+
SELECT to_number('454.00', '000.00');
```

```
+-------------------------+
|to_number(454.00, 000.00)|
+-------------------------+
|                   454.00|
+-------------------------+
SELECT to_number('12,454', '99,999');
+-------------------------+
|to_number(12,454, 99,999)|
+-------------------------+
|                    12454|
+-------------------------+
SELECT to_number('$78.12', '$99.99');
+-------------------------+
|to_number($78.12, $99.99)|
+-------------------------+
|                    78.12|
+-------------------------+
SELECT to_number('12,454.8-', '99,999.9S');
+-------------------------------+
|to_number(12,454.8-, 99,999.9S)|
+-------------------------------+
|                       -12454.8|
+-------------------------------+
-- to_varchar
SELECT to_varchar(454, '999');
+-----------------+
|to_char(454, 999)|
+-----------------+
|              454|
+-----------------+
SELECT to_varchar(454.00, '000D00');
+-----------------------+
|to_char(454.00, 000D00)|
+-----------------------+
|                 454.00|
+-----------------------+
SELECT to_varchar(12454, '99G999');
+----------------------+
|to_char(12454, 99G999)|
+----------------------+
|                12,454|
+----------------------+
SELECT to_varchar(78.12, '$99.99');
+----------------------+
```

```
|to_char(78.12, $99.99)|
+----------------------+
|               $78.12|
+----------------------+
SELECT to_varchar(-12454.8, '99G999D9S');
+---------------------------+
|to_char(-12454.8, 99G999D9S)|
+---------------------------+
|                  12,454.8-|
+---------------------------+
-- translate
SELECT translate('AaBbCc', 'abc', '123');
+--------------------------+
|translate(AaBbCc, abc, 123)|
+--------------------------+
|                    A1B2C3|
+--------------------------+
-- try_to_binary
SELECT try_to_binary('abc', 'utf-8');
+-------------------------+
|try_to_binary(abc, utf-8)|
+-------------------------+
|              [61 62 63]|
+-------------------------+
select try_to_binary('a!', 'base64');
+-------------------------+
|try_to_binary(a!, base64)|
+-------------------------+
|                     NULL|
+-------------------------+
select try_to_binary('abc', 'invalidFormat');
+---------------------------------+
|try_to_binary(abc, invalidFormat)|
+---------------------------------+
|                             NULL|
+---------------------------------+
-- try_to_number
SELECT try_to_number('454', '999');
+----------------------+
|try_to_number(454, 999)|
+----------------------+
|                   454|
+----------------------+
SELECT try_to_number('454.00', '000.00');
```

```
+-----------------------------+
|try_to_number(454.00, 000.00)|
+-----------------------------+
|                       454.00|
+-----------------------------+
SELECT try_to_number('12,454', '99,999');
+-----------------------------+
|try_to_number(12,454, 99,999)|
+-----------------------------+
|                        12454|
+-----------------------------+
SELECT try_to_number('$78.12', '$99.99');
+-----------------------------+
|try_to_number($78.12, $99.99)|
+-----------------------------+
|                        78.12|
+-----------------------------+
SELECT try_to_number('12,454.8-', '99,999.9S');
+-----------------------------------+
|try_to_number(12,454.8-, 99,999.9S)|
+-----------------------------------+
|                           -12454.8|
+-----------------------------------+
-- ucase
SELECT ucase('Feathers');
+---------------+
|ucase(Feathers)|
+---------------+
|       FEATHERS|
+---------------+
-- unbase64
SELECT unbase64('U3BhcmsgU1FM');
+----------------------+
|unbase64(U3BhcmsgU1FM)|
+----------------------+
|  [53 70 61 72 6B 2...|
+----------------------+
-- upper
SELECT upper('Feathers');
+---------------+
|upper(Feathers)|
+---------------+
|       FEATHERS|
```

```
+---------------+
```

Fonctions de date et d'heure

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called "Commandes SQL prises en charge".

Fonction	Description
ajouter_mois (date_début, num_mois)	Renvoie la date `num_months` postérieu `restart_date` .
convert_timezone ([SourceTZ,] Targettz, SourceTS)	Convertit l'horodatage sans fuseau horaire `sourceTs` du fuseau horaire `sourceTz` en. `targetTz`
curateur ()	Renvoie la date actuelle au début de l'évaluat ion de la requête. Tous les appels de curdate au sein d'une même requête renvoient la même valeur.
date_actuelle ()	Renvoie la date actuelle au début de l'évaluat ion de la requête. Tous les appels de current_d ate dans la même requête renvoient la même valeur.
date_actuelle	Renvoie la date actuelle au début de l'évaluat ion de la requête.
horodatage actuel ()	Renvoie l'horodatage actuel au début de l'évaluation de la requête. Tous les appels de current_timestamp dans la même requête renvoient la même valeur.

Fonction	Description
horodatage actuel	Renvoie l'horodatage actuel au début de l'évaluation de la requête.
fuseau horaire actuel ()	Renvoie le fuseau horaire local de la session en cours.
date_add (date_début, num_jours)	Renvoie la date `num_days` postérieu `restart_date` .
date_diff (date de fin, date de début)	Renvoie le nombre de jours entre `startDate` etendDate.
date_format (horodatage, fmt)	Convertit `timestamp` en une valeur de chaîne au format spécifié par le format de date`fmt`.
date_from_unix_date (jours)	Créez une date à partir du nombre de jours écoulés depuis le 01/01/1970.
date_part (champ, source)	Extrait une partie de la source date/timestamp ou de l'intervalle.
date_sub (date_début, num_jours)	Renvoie la date `num_days` antérieur `estart_date` .
date_trunc (fmt, ts)	Renvoie l'horodatage `ts` tronqué selon l'unité spécifiée par le modèle de format. `fmt`
dateadd (date_début, num_jours)	Renvoie la date `num_days` postérieu `restart_date` .
datediff (date de fin, date de début)	Renvoie le nombre de jours entre `startDate` etendDate.
datepart (champ, source)	Extrait une partie de la source date/timestamp ou de l'intervalle.

Fonction	Description
jour (date)	Renvoie le jour du mois correspondant à la date/à l'horodatage.
jour du mois (date)	Renvoie le jour du mois correspondant à la date/à l'horodatage.
jour de la semaine (date)	Renvoie le jour de la semaine pour date/time stamp (1 = dimanche, 2 = lundi,..., 7 = samedi).
jour de l'année (date)	Renvoie le jour de l'année correspondant à la date/à l'horodatage.
extrait (champ DEPUIS la source)	Extrait une partie de la source date/timestamp ou de l'intervalle.
from_unixtime (unix_time [, fmt])	Renvoie `unix_time` la valeur spécifiée`fmt`.
from_utc_timestamp (horodatage, fuseau horaire)	Avec un horodatage tel que « 2017-07-1 4 02:40:00.0 », il l'interprète comme une heure en UTC et affiche cette heure sous forme d'horodatage dans le fuseau horaire donné. Par exemple, « GMT+1 » donnerait « 2017-07-14 03:40:00.0 ».
heure (horodatage)	Renvoie le composant horaire de la chaîne/de l'horodatage.
dernier_jour (date)	Renvoie le dernier jour du mois auquel appartient la date.
horodatage local ()	Renvoie l'horodatage actuel sans le fuseau horaire au début de l'évaluation de la requête. Tous les appels de localtimestamp au sein d'une même requête renvoient la même valeur.

Fonction	Description
horodatage local	Renvoie la date et l'heure locales actuelles au fuseau horaire de la session au début de l'évaluation de la requête.
make_date (année, mois, jour)	Créez une date à partir des champs de l'année, du mois et du jour.
make_dt_interval ([jours [, heures [, minutes [, secondes]]]])	Définissez DayTimeIntervalType la durée en jours, heures, minutes et secondes.
make_interval ([années [, mois [, semaines [, jours [, heures [, minutes [, secondes]]]]]]]))	Établissez un intervalle à partir des années, des mois, des semaines, des jours, des heures, des minutes et des secondes.
make_timestamp (année, mois, jour, heure, min, sec [, fuseau horaire])	Créez un horodatage à partir des champs année, mois, jour, heure, minute, seconde et fuseau horaire.
make_timestamp_ltz (année, mois, jour, heure, min, sec [, fuseau horaire])	Créez l'horodatage actuel avec le fuseau horaire local à partir des champs année, mois, jour, heure, min, sec et fuseau horaire.
make_timestamp_ntz (année, mois, jour, heure, minute, seconde)	Créez une date et une heure locales à partir des champs année, mois, jour, heure, minute et seconde.
make_ym_interval ([années [, mois]])	Faites un intervalle année-mois à partir des années, des mois.
minute (horodatage)	Renvoie la composante minute de la chaîne/de l'horodatage.
mois (date)	Renvoie le composant mensuel de la date/de l'horodatage.

Fonction	Description
months_between (horodatage1, horodatage2 [, arrondOff])	Si `timestamp1` c'est plus tard que`timestamp2` , le résultat est positif. Si `timestamp1` et si `timestamp2` c'est le même jour du mois, ou si les deux sont le dernier jour du mois, l'heure ne sera pas prise en compte. Sinon, la différence est calculée sur la base de 31 jours par mois et arrondie à 8 chiffres, sauf RoundOff=False.
jour_suivant (date de début, jour_de_semaine)	Renvoie la première date postérieure à la date indiquée `start_date` et nommée comme indiqué. La fonction renvoie NULL si au moins un des paramètres d'entrée est NULL.
maintenant ()	Renvoie l'horodatage actuel au début de l'évaluation de la requête.
trimestre (date)	Renvoie le trimestre de l'année pour la date, dans la plage de 1 à 4.
seconde (horodatage)	Renvoie le deuxième composant de la chaîne/de l'horodatage.
fenêtre_session (colonne horaire, durée de l'intervalle)	Génère une fenêtre de session avec un horodatage spécifiant la durée de la colonne et de l'intervalle. Voir « Types de fenêtres temporelles » dans le document du guide du streaming structuré pour des explications détaillées et des exemples.
timestamp_micros (microsecondes)	Crée un horodatage à partir du nombre de microsecondes écoulées depuis l'époque UTC.
timestamp_millis (millisecondes)	Crée un horodatage à partir du nombre de millisecondes écoulées depuis l'époque UTC.

Fonction	Description
timestamp_seconds (secondes)	Crée un horodatage à partir du nombre de secondes (peut être fractionnaire) depuis l'époque UTC.
to_date (date_str [, fmt])	Analyse l'`date_str`expression avec l'`fmt`expression jusqu'à une date. Renvoie une valeur nulle si la saisie n'est pas valide. Par défaut, il suit les règles de casting jusqu'à une date si le `fmt` est omis.
to_timestamp (timestamp_str [, fmt])	Analyse l'`timestamp_str` expression avec l'`fmt`expression selon un horodatage. Renvoie une valeur nulle si la saisie n'est pas valide. Par défaut, il applique les règles de conversion à un horodatage si le `fmt` est omis.
to_timestamp_ltz (timestamp_str [, fmt])	Analyse l'expression associée à l'`timestamp_str fmt`expression selon un horodatage avec le fuseau horaire local. Renvoie une valeur nulle si la saisie n'est pas valide. Par défaut, il applique les règles de conversion à un horodatage si le `fmt` est omis.
to_timestamp_ntz (timestamp_str [, fmt])	Analyse l'expression associée à l'`timestamp_str fmt`expression selon un horodatage sans fuseau horaire. Renvoie une valeur nulle si la saisie n'est pas valide. Par défaut, il applique les règles de conversion à un horodatage si le `fmt` est omis.
to_unix_timestamp (TimeExp [, fmt])	Renvoie l'horodatage UNIX de l'heure donnée.

Fonction	Description
to_utc_timestamp (horodatage, fuseau horaire)	Avec un horodatage tel que « 2017-07-14 02:40:00.0 », il l'interprète comme une heure dans le fuseau horaire donné et affiche cette heure sous forme d'horodatage en UTC. Par exemple, « GMT+1 » donnerait « 2017-07-14 01:40:00.0 ».
tronc (date, fmt)	Renvoie `date` la partie horaire du jour tronquée à l'unité spécifiée par le modèle `fmt` de format.
try_to_timestamp (timestamp_str [, fmt])	Analyse l'`timestamp_str` expression avec l'`fmt`expression selon un horodatage.
unix_date (date)	Renvoie le nombre de jours écoulés depuis le 01/01/1970.
unix_micros (horodatage)	Renvoie le nombre de microsecondes écoulées depuis le 01/01/1970 00:00:00 UTC.
unix_millis (horodatage)	Renvoie le nombre de millisecondes écoulées depuis le 01/01/1970 00:00:00 UTC. Tronque les niveaux de précision les plus élevés.
unix_seconds (horodatage)	Renvoie le nombre de secondes écoulées depuis le 01/01/1970 à 00:00:00 UTC. Tronque les niveaux de précision les plus élevés.
unix_timestamp ([TimeExp [, fmt]])	Renvoie l'horodatage UNIX de l'heure actuelle ou spécifiée.
jour de la semaine (date)	Renvoie le jour de la semaine pour date/time stamp (0 = lundi, 1 = mardi,..., 6 = dimanche).

Fonction	Description
semaine de l'année (date)	Renvoie la semaine de l'année à la date donnée. Une semaine est considérée comme commençant un lundi et la semaine 1 est la première semaine de plus de 3 jours.
fenêtre (time_column, window_duration [, slide_duration [, start_time]])	Classez les lignes en une ou plusieurs fenêtres temporelles en fonction d'une colonne spécifiant l'horodatage. Les démarrages de fenêtres sont inclusifs mais les fins de fenêtre sont exclusives, par exemple 12:05 sera dans la fenêtre [12:05,12:10) mais pas dans [12:00,12:05). Windows peut prendre en charge une précision de l'ordre de la microseconde. Les fenêtres de l'ordre des mois ne sont pas prises en charge. Reportez-vous à la section « Opérations de fenêtre sur l'heure des événements » dans le document du guide de streaming structuré pour des explications détaillées et des exemples.
window_time (window_column)	Extrayez la valeur temporelle de time/session la colonne de fenêtre qui peut être utilisée pour la valeur temporelle de l'événement de la fenêtre. L'heure extraite est (window.end - 1), ce qui reflète le fait que les fenêtres d'agrégation ont une limite supérieure exclusive - [start, end). Voir « Opérations de fenêtre sur l'heure de l'événement » dans le document du guide de streaming structuré pour des explications détaillées et des exemples.
année (date)	Renvoie la composante annuelle de la date/de l'horodatage.

Exemples

```
-- add_months
SELECT add_months('2016-08-31', 1);
+-------------------------+
|add_months(2016-08-31, 1)|
+-------------------------+
|               2016-09-30|
+-------------------------+
-- convert_timezone
SELECT convert_timezone('Europe/Brussels', 'America/Los_Angeles',
 timestamp_ntz'2021-12-06 00:00:00');
+-----------------------------------------------------------------------------------------------
+
|convert_timezone(Europe/Brussels, America/Los_Angeles, TIMESTAMP_NTZ '2021-12-06
 00:00:00')|
+-----------------------------------------------------------------------------------------------
+
|                                                                                2021-12-05
 15:00:00|
+-----------------------------------------------------------------------------------------------
+
SELECT convert_timezone('Europe/Brussels', timestamp_ntz'2021-12-05 15:00:00');
+-----------------------------------------------------------------------------------------------
+
|convert_timezone(current_timezone(), Europe/Brussels, TIMESTAMP_NTZ '2021-12-05
 15:00:00')|
+-----------------------------------------------------------------------------------------------
+
|                                                                              2021-12-05
 07:00:00|
+-----------------------------------------------------------------------------------------------
+
-- curdate
SELECT curdate();
+--------------+
|current_date()|
+--------------+
|    2024-02-24|
+--------------+
-- current_date
SELECT current_date();
+--------------+
|current_date()|
+--------------+
```

```
|    2024-02-24|
+--------------+
SELECT current_date;
+--------------+
|current_date()|
+--------------+
|    2024-02-24|
+--------------+
-- current_timestamp
SELECT current_timestamp();
+--------------------+
| current_timestamp()|
+--------------------+
|2024-02-24 16:36:...|
+--------------------+
SELECT current_timestamp;
+--------------------+
| current_timestamp()|
+--------------------+
|2024-02-24 16:36:...|
+--------------------+
-- current_timezone
SELECT current_timezone();
+------------------+
|current_timezone()|
+------------------+
|        Asia/Seoul|
+------------------+
-- date_add
SELECT date_add('2016-07-30', 1);
+-----------------------+
|date_add(2016-07-30, 1)|
+-----------------------+
|             2016-07-31|
+-----------------------+
-- date_diff
SELECT date_diff('2009-07-31', '2009-07-30');
+---------------------------------+
|date_diff(2009-07-31, 2009-07-30)|
+---------------------------------+
|                                1|
+---------------------------------+
SELECT date_diff('2009-07-30', '2009-07-31');
+---------------------------------+
```

```
|date_diff(2009-07-30, 2009-07-31)|
+---------------------------------+
|                               -1|
+---------------------------------+
-- date_format
SELECT date_format('2016-04-08', 'y');
+--------------------------+
|date_format(2016-04-08, y)|
+--------------------------+
|                      2016|
+--------------------------+
-- date_from_unix_date
SELECT date_from_unix_date(1);
+----------------------+
|date_from_unix_date(1)|
+----------------------+
|            1970-01-02|
+----------------------+
-- date_part
SELECT date_part('YEAR', TIMESTAMP '2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|date_part(YEAR, TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                       2019|
+-----------------------------------------------------------+
SELECT date_part('week', timestamp'2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|date_part(week, TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                         33|
+-----------------------------------------------------------+
SELECT date_part('doy', DATE'2019-08-12');
+---------------------------------+
|date_part(doy, DATE '2019-08-12')|
+---------------------------------+
|                              224|
+---------------------------------+
SELECT date_part('SECONDS', timestamp'2019-10-01 00:00:01.000001');
+-------------------------------------------------------------+
|date_part(SECONDS, TIMESTAMP '2019-10-01 00:00:01.000001')|
+-------------------------------------------------------------+
|                                                     1.000001|
+-------------------------------------------------------------+
SELECT date_part('days', interval 5 days 3 hours 7 minutes);
```

```
+-----------------------------------------------+
|date_part(days, INTERVAL '5 03:07' DAY TO MINUTE)|
+-----------------------------------------------+
|                                              5|
+-----------------------------------------------+
SELECT date_part('seconds', interval 5 hours 30 seconds 1 milliseconds 1 microseconds);
+-------------------------------------------------------------+
|date_part(seconds, INTERVAL '05:00:30.001001' HOUR TO SECOND)|
+-------------------------------------------------------------+
|                                                    30.001001|
+-------------------------------------------------------------+
SELECT date_part('MONTH', INTERVAL '2021-11' YEAR TO MONTH);
+-----------------------------------------------+
|date_part(MONTH, INTERVAL '2021-11' YEAR TO MONTH)|
+-----------------------------------------------+
|                                             11|
+-----------------------------------------------+
SELECT date_part('MINUTE', INTERVAL '123 23:55:59.002001' DAY TO SECOND);
+-------------------------------------------------------------+
|date_part(MINUTE, INTERVAL '123 23:55:59.002001' DAY TO SECOND)|
+-------------------------------------------------------------+
|                                                           55|
+-------------------------------------------------------------+
-- date_sub
SELECT date_sub('2016-07-30', 1);
+----------------------+
|date_sub(2016-07-30, 1)|
+----------------------+
|             2016-07-29|
+----------------------+
-- date_trunc
SELECT date_trunc('YEAR', '2015-03-05T09:32:05.359');
+---------------------------------------+
|date_trunc(YEAR, 2015-03-05T09:32:05.359)|
+---------------------------------------+
|                    2015-01-01 00:00:00|
+---------------------------------------+
SELECT date_trunc('MM', '2015-03-05T09:32:05.359');
+-------------------------------------+
|date_trunc(MM, 2015-03-05T09:32:05.359)|
+-------------------------------------+
|                   2015-03-01 00:00:00|
+-------------------------------------+
SELECT date_trunc('DD', '2015-03-05T09:32:05.359');
```

```
+--------------------------------------+
|date_trunc(DD, 2015-03-05T09:32:05.359)|
+--------------------------------------+
|                   2015-03-05 00:00:00|
+--------------------------------------+
SELECT date_trunc('HOUR', '2015-03-05T09:32:05.359');
+---------------------------------------+
|date_trunc(HOUR, 2015-03-05T09:32:05.359)|
+---------------------------------------+
|                     2015-03-05 09:00:00|
+---------------------------------------+
SELECT date_trunc('MILLISECOND', '2015-03-05T09:32:05.123456');
+-----------------------------------------------------+
|date_trunc(MILLISECOND, 2015-03-05T09:32:05.123456)|
+-----------------------------------------------------+
|                              2015-03-05 09:32:...|
+-----------------------------------------------------+
-- dateadd
SELECT dateadd('2016-07-30', 1);
+----------------------+
|date_add(2016-07-30, 1)|
+----------------------+
|             2016-07-31|
+----------------------+
-- datediff
SELECT datediff('2009-07-31', '2009-07-30');
+--------------------------------+
|datediff(2009-07-31, 2009-07-30)|
+--------------------------------+
|                               1|
+--------------------------------+
SELECT datediff('2009-07-30', '2009-07-31');
+--------------------------------+
|datediff(2009-07-30, 2009-07-31)|
+--------------------------------+
|                              -1|
+--------------------------------+
-- datepart
SELECT datepart('YEAR', TIMESTAMP '2019-08-12 01:00:00.123456');
+-------------------------------------------------------------+
|datepart(YEAR FROM TIMESTAMP '2019-08-12 01:00:00.123456')|
+-------------------------------------------------------------+
|                                                         2019|
+-------------------------------------------------------------+
```

```
SELECT datepart('week', timestamp'2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|datepart(week FROM TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                         33|
+-----------------------------------------------------------+
SELECT datepart('doy', DATE'2019-08-12');
+-----------------------------------+
|datepart(doy FROM DATE '2019-08-12')|
+-----------------------------------+
|                                224|
+-----------------------------------+
SELECT datepart('SECONDS', timestamp'2019-10-01 00:00:01.000001');
+-------------------------------------------------------------+
|datepart(SECONDS FROM TIMESTAMP '2019-10-01 00:00:01.000001')|
+-------------------------------------------------------------+
|                                                     1.000001|
+-------------------------------------------------------------+
SELECT datepart('days', interval 5 days 3 hours 7 minutes);
+---------------------------------------------------+
|datepart(days FROM INTERVAL '5 03:07' DAY TO MINUTE)|
+---------------------------------------------------+
|                                                  5|
+---------------------------------------------------+
SELECT datepart('seconds', interval 5 hours 30 seconds 1 milliseconds 1 microseconds);
+-----------------------------------------------------------------+
|datepart(seconds FROM INTERVAL '05:00:30.001001' HOUR TO SECOND)|
+-----------------------------------------------------------------+
|                                                        30.001001|
+-----------------------------------------------------------------+
SELECT datepart('MONTH', INTERVAL '2021-11' YEAR TO MONTH);
+---------------------------------------------------+
|datepart(MONTH FROM INTERVAL '2021-11' YEAR TO MONTH)|
+---------------------------------------------------+
|                                                 11|
+---------------------------------------------------+
SELECT datepart('MINUTE', INTERVAL '123 23:55:59.002001' DAY TO SECOND);
+-------------------------------------------------------------------+
|datepart(MINUTE FROM INTERVAL '123 23:55:59.002001' DAY TO SECOND)|
+-------------------------------------------------------------------+
|                                                                 55|
+-------------------------------------------------------------------+
-- day
SELECT day('2009-07-30');
```

```
+---------------+
|day(2009-07-30)|
+---------------+
|             30|
+---------------+
-- dayofmonth
SELECT dayofmonth('2009-07-30');
+----------------------+
|dayofmonth(2009-07-30)|
+----------------------+
|                    30|
+----------------------+
-- dayofweek
SELECT dayofweek('2009-07-30');
+---------------------+
|dayofweek(2009-07-30)|
+---------------------+
|                    5|
+---------------------+
-- dayofyear
SELECT dayofyear('2016-04-09');
+---------------------+
|dayofyear(2016-04-09)|
+---------------------+
|                  100|
+---------------------+
-- extract
SELECT extract(YEAR FROM TIMESTAMP '2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|extract(YEAR FROM TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                       2019|
+-----------------------------------------------------------+
SELECT extract(week FROM timestamp'2019-08-12 01:00:00.123456');
+-----------------------------------------------------------+
|extract(week FROM TIMESTAMP '2019-08-12 01:00:00.123456')|
+-----------------------------------------------------------+
|                                                         33|
+-----------------------------------------------------------+
SELECT extract(doy FROM DATE'2019-08-12');
+----------------------------------+
|extract(doy FROM DATE '2019-08-12')|
+----------------------------------+
|                               224|
```

```
+----------------------------------+
SELECT extract(SECONDS FROM timestamp'2019-10-01 00:00:01.000001');
+------------------------------------------------------------+
|extract(SECONDS FROM TIMESTAMP '2019-10-01 00:00:01.000001')|
+------------------------------------------------------------+
|                                                    1.000001|
+------------------------------------------------------------+
SELECT extract(days FROM interval 5 days 3 hours 7 minutes);
+---------------------------------------------------+
|extract(days FROM INTERVAL '5 03:07' DAY TO MINUTE)|
+---------------------------------------------------+
|                                                  5|
+---------------------------------------------------+
SELECT extract(seconds FROM interval 5 hours 30 seconds 1 milliseconds 1 microseconds);
+-------------------------------------------------------------------+
|extract(seconds FROM INTERVAL '05:00:30.001001' HOUR TO SECOND)|
+-------------------------------------------------------------------+
|                                                          30.001001|
+-------------------------------------------------------------------+
SELECT extract(MONTH FROM INTERVAL '2021-11' YEAR TO MONTH);
+---------------------------------------------------+
|extract(MONTH FROM INTERVAL '2021-11' YEAR TO MONTH)|
+---------------------------------------------------+
|                                                 11|
+---------------------------------------------------+
SELECT extract(MINUTE FROM INTERVAL '123 23:55:59.002001' DAY TO SECOND);
+-------------------------------------------------------------------+
|extract(MINUTE FROM INTERVAL '123 23:55:59.002001' DAY TO SECOND)|
+-------------------------------------------------------------------+
|                                                                 55|
+-------------------------------------------------------------------+
-- from_unixtime
SELECT from_unixtime(0, 'yyyy-MM-dd HH:mm:ss');
+-----------------------------------+
|from_unixtime(0, yyyy-MM-dd HH:mm:ss)|
+-----------------------------------+
|                1970-01-01 09:00:00|
+-----------------------------------+
SELECT from_unixtime(0);
+-----------------------------------+
|from_unixtime(0, yyyy-MM-dd HH:mm:ss)|
+-----------------------------------+
|                1970-01-01 09:00:00|
+-----------------------------------+
```

```
-- from_utc_timestamp
SELECT from_utc_timestamp('2016-08-31', 'Asia/Seoul');
+-----------------------------------------+
|from_utc_timestamp(2016-08-31, Asia/Seoul)|
+-----------------------------------------+
|                      2016-08-31 09:00:00|
+-----------------------------------------+
-- hour
SELECT hour('2009-07-30 12:58:59');
+-------------------------+
|hour(2009-07-30 12:58:59)|
+-------------------------+
|                       12|
+-------------------------+
-- last_day
SELECT last_day('2009-01-12');
+--------------------+
|last_day(2009-01-12)|
+--------------------+
|          2009-01-31|
+--------------------+
-- localtimestamp
SELECT localtimestamp();
+--------------------+
|    localtimestamp()|
+--------------------+
|2024-02-24 16:36:...|
+--------------------+
-- make_date
SELECT make_date(2013, 7, 15);
+----------------------+
|make_date(2013, 7, 15)|
+----------------------+
|            2013-07-15|
+----------------------+
SELECT make_date(2019, 7, NULL);
+------------------------+
|make_date(2019, 7, NULL)|
+------------------------+
|                    NULL|
+------------------------+
-- make_dt_interval
SELECT make_dt_interval(1, 12, 30, 01.001001);
+-------------------------------------+
```

```
|make_dt_interval(1, 12, 30, 1.001001)|
+-------------------------------------+
|                 INTERVAL '1 12:30...|
+-------------------------------------+
SELECT make_dt_interval(2);
+----------------------------------+
|make_dt_interval(2, 0, 0, 0.000000)|
+----------------------------------+
|               INTERVAL '2 00:00...|
+----------------------------------+
SELECT make_dt_interval(100, null, 3);
+---------------------------------------+
|make_dt_interval(100, NULL, 3, 0.000000)|
+---------------------------------------+
|                                   NULL|
+---------------------------------------+
-- make_interval
SELECT make_interval(100, 11, 1, 1, 12, 30, 01.001001);
+-----------------------------------------------+
|make_interval(100, 11, 1, 1, 12, 30, 1.001001)|
+-----------------------------------------------+
|                          100 years 11 mont...|
+-----------------------------------------------+
SELECT make_interval(100, null, 3);
+-----------------------------------------------+
|make_interval(100, NULL, 3, 0, 0, 0, 0.000000)|
+-----------------------------------------------+
|                                           NULL|
+-----------------------------------------------+
SELECT make_interval(0, 1, 0, 1, 0, 0, 100.000001);
+-------------------------------------------+
|make_interval(0, 1, 0, 1, 0, 0, 100.000001)|
+-------------------------------------------+
|                       1 months 1 days 1...|
+-------------------------------------------+
-- make_timestamp
SELECT make_timestamp(2014, 12, 28, 6, 30, 45.887);
+-----------------------------------------+
|make_timestamp(2014, 12, 28, 6, 30, 45.887)|
+-----------------------------------------+
|                      2014-12-28 06:30:...|
+-----------------------------------------+
SELECT make_timestamp(2014, 12, 28, 6, 30, 45.887, 'CET');
+-------------------------------------------------+
```

```
|make_timestamp(2014, 12, 28, 6, 30, 45.887, CET)|
+------------------------------------------------+
|                            2014-12-28 14:30:...|
+------------------------------------------------+
SELECT make_timestamp(2019, 6, 30, 23, 59, 60);
+--------------------------------------+
|make_timestamp(2019, 6, 30, 23, 59, 60)|
+--------------------------------------+
|                     2019-07-01 00:00:00|
+--------------------------------------+
SELECT make_timestamp(2019, 6, 30, 23, 59, 1);
+-------------------------------------+
|make_timestamp(2019, 6, 30, 23, 59, 1)|
+-------------------------------------+
|                    2019-06-30 23:59:01|
+-------------------------------------+
SELECT make_timestamp(null, 7, 22, 15, 30, 0);
+-------------------------------------+
|make_timestamp(NULL, 7, 22, 15, 30, 0)|
+-------------------------------------+
|                                  NULL|
+-------------------------------------+
-- make_timestamp_ltz
SELECT make_timestamp_ltz(2014, 12, 28, 6, 30, 45.887);
+-----------------------------------------------+
|make_timestamp_ltz(2014, 12, 28, 6, 30, 45.887)|
+-----------------------------------------------+
|                           2014-12-28 06:30:...|
+-----------------------------------------------+
SELECT make_timestamp_ltz(2014, 12, 28, 6, 30, 45.887, 'CET');
+--------------------------------------------------+
|make_timestamp_ltz(2014, 12, 28, 6, 30, 45.887, CET)|
+--------------------------------------------------+
|                               2014-12-28 14:30:...|
+--------------------------------------------------+
SELECT make_timestamp_ltz(2019, 6, 30, 23, 59, 60);
+-----------------------------------------+
|make_timestamp_ltz(2019, 6, 30, 23, 59, 60)|
+-----------------------------------------+
|                       2019-07-01 00:00:00|
+-----------------------------------------+
SELECT make_timestamp_ltz(null, 7, 22, 15, 30, 0);
+-----------------------------------------+
|make_timestamp_ltz(NULL, 7, 22, 15, 30, 0)|
```

```
+----------------------------------------+
|                                    NULL|
+----------------------------------------+
-- make_timestamp_ntz
SELECT make_timestamp_ntz(2014, 12, 28, 6, 30, 45.887);
+-----------------------------------------------+
|make_timestamp_ntz(2014, 12, 28, 6, 30, 45.887)|
+-----------------------------------------------+
|                        2014-12-28 06:30:...|
+-----------------------------------------------+
SELECT make_timestamp_ntz(2019, 6, 30, 23, 59, 60);
+-----------------------------------------+
|make_timestamp_ntz(2019, 6, 30, 23, 59, 60)|
+-----------------------------------------+
|                      2019-07-01 00:00:00|
+-----------------------------------------+
SELECT make_timestamp_ntz(null, 7, 22, 15, 30, 0);
+-----------------------------------------+
|make_timestamp_ntz(NULL, 7, 22, 15, 30, 0)|
+-----------------------------------------+
|                                    NULL|
+-----------------------------------------+
-- make_ym_interval
SELECT make_ym_interval(1, 2);
+----------------------+
|make_ym_interval(1, 2)|
+----------------------+
|  INTERVAL '1-2' YE...|
+----------------------+
SELECT make_ym_interval(1, 0);
+----------------------+
|make_ym_interval(1, 0)|
+----------------------+
|  INTERVAL '1-0' YE...|
+----------------------+
SELECT make_ym_interval(-1, 1);
+----------------------+
|make_ym_interval(-1, 1)|
+----------------------+
|   INTERVAL '-0-11' ...|
+----------------------+
SELECT make_ym_interval(2);
+----------------------+
|make_ym_interval(2, 0)|
```

```
+----------------------+
|  INTERVAL '2-0' YE...|
+----------------------+
-- minute
SELECT minute('2009-07-30 12:58:59');
+--------------------------+
|minute(2009-07-30 12:58:59)|
+--------------------------+
|                        58|
+--------------------------+
-- month
SELECT month('2016-07-30');
+-----------------+
|month(2016-07-30)|
+-----------------+
|                7|
+-----------------+
-- months_between
SELECT months_between('1997-02-28 10:30:00', '1996-10-30');
+-------------------------------------------------------+
|months_between(1997-02-28 10:30:00, 1996-10-30, true)|
+-------------------------------------------------------+
|                                             3.94959677|
+-------------------------------------------------------+
SELECT months_between('1997-02-28 10:30:00', '1996-10-30', false);
+--------------------------------------------------------+
|months_between(1997-02-28 10:30:00, 1996-10-30, false)|
+--------------------------------------------------------+
|                                    3.9495967741935485|
+--------------------------------------------------------+
-- next_day
SELECT next_day('2015-01-14', 'TU');
+------------------------+
|next_day(2015-01-14, TU)|
+------------------------+
|              2015-01-20|
+------------------------+
-- now
SELECT now();
+--------------------+
|               now()|
+--------------------+
|2024-02-24 16:36:...|
+--------------------+
```

```
-- quarter
SELECT quarter('2016-08-31');
+-------------------+
|quarter(2016-08-31)|
+-------------------+
|                  3|
+-------------------+
-- second
SELECT second('2009-07-30 12:58:59');
+--------------------------+
|second(2009-07-30 12:58:59)|
+--------------------------+
|                        59|
+--------------------------+
-- session_window
SELECT a, session_window.start, session_window.end, count(*) as cnt FROM VALUES ('A1',
 '2021-01-01 00:00:00'), ('A1', '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:10:00'),
 ('A2', '2021-01-01 00:01:00') AS tab(a, b) GROUP by a, session_window(b, '5 minutes')
 ORDER BY a, start;
+---+-------------------+-------------------+---+
|  a|              start|                end|cnt|
+---+-------------------+-------------------+---+
| A1|2021-01-01 00:00:00|2021-01-01 00:09:30|  2|
| A1|2021-01-01 00:10:00|2021-01-01 00:15:00|  1|
| A2|2021-01-01 00:01:00|2021-01-01 00:06:00|  1|
+---+-------------------+-------------------+---+
SELECT a, session_window.start, session_window.end, count(*) as cnt FROM VALUES ('A1',
 '2021-01-01 00:00:00'), ('A1', '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:10:00'),
 ('A2', '2021-01-01 00:01:00'), ('A2', '2021-01-01 00:04:30') AS tab(a, b) GROUP by a,
 session_window(b, CASE WHEN a = 'A1' THEN '5 minutes' WHEN a = 'A2' THEN '1 minute'
 ELSE '10 minutes' END) ORDER BY a, start;
+---+-------------------+-------------------+---+
|  a|              start|                end|cnt|
+---+-------------------+-------------------+---+
| A1|2021-01-01 00:00:00|2021-01-01 00:09:30|  2|
| A1|2021-01-01 00:10:00|2021-01-01 00:15:00|  1|
| A2|2021-01-01 00:01:00|2021-01-01 00:02:00|  1|
| A2|2021-01-01 00:04:30|2021-01-01 00:05:30|  1|
+---+-------------------+-------------------+---+
-- timestamp_micros
SELECT timestamp_micros(1230219000123123);
+---------------------------------+
|timestamp_micros(1230219000123123)|
+---------------------------------+
```

```
|              2008-12-26 00:30:...|
+----------------------------------+
-- timestamp_millis
SELECT timestamp_millis(1230219000123);
+------------------------------+
|timestamp_millis(1230219000123)|
+------------------------------+
|           2008-12-26 00:30:...|
+------------------------------+
-- timestamp_seconds
SELECT timestamp_seconds(1230219000);
+----------------------------+
|timestamp_seconds(1230219000)|
+----------------------------+
|          2008-12-26 00:30:00|
+----------------------------+
SELECT timestamp_seconds(1230219000.123);
+---------------------------------+
|timestamp_seconds(1230219000.123)|
+---------------------------------+
|             2008-12-26 00:30:...|
+---------------------------------+
-- to_date
SELECT to_date('2009-07-30 04:17:52');
+---------------------------+
|to_date(2009-07-30 04:17:52)|
+---------------------------+
|                 2009-07-30|
+---------------------------+
SELECT to_date('2016-12-31', 'yyyy-MM-dd');
+-------------------------------+
|to_date(2016-12-31, yyyy-MM-dd)|
+-------------------------------+
|                     2016-12-31|
+-------------------------------+
-- to_timestamp
SELECT to_timestamp('2016-12-31 00:12:00');
+---------------------------------+
|to_timestamp(2016-12-31 00:12:00)|
+---------------------------------+
|              2016-12-31 00:12:00|
+---------------------------------+
SELECT to_timestamp('2016-12-31', 'yyyy-MM-dd');
+---------------------------------+
```

```
|to_timestamp(2016-12-31, yyyy-MM-dd)|
+------------------------------------+
|                 2016-12-31 00:00:00|
+------------------------------------+
-- to_timestamp_ltz
SELECT to_timestamp_ltz('2016-12-31 00:12:00');
+-------------------------------------+
|to_timestamp_ltz(2016-12-31 00:12:00)|
+-------------------------------------+
|                  2016-12-31 00:12:00|
+-------------------------------------+
SELECT to_timestamp_ltz('2016-12-31', 'yyyy-MM-dd');
+----------------------------------------+
|to_timestamp_ltz(2016-12-31, yyyy-MM-dd)|
+----------------------------------------+
|                     2016-12-31 00:00:00|
+----------------------------------------+
-- to_timestamp_ntz
SELECT to_timestamp_ntz('2016-12-31 00:12:00');
+-------------------------------------+
|to_timestamp_ntz(2016-12-31 00:12:00)|
+-------------------------------------+
|                  2016-12-31 00:12:00|
+-------------------------------------+
SELECT to_timestamp_ntz('2016-12-31', 'yyyy-MM-dd');
+----------------------------------------+
|to_timestamp_ntz(2016-12-31, yyyy-MM-dd)|
+----------------------------------------+
|                     2016-12-31 00:00:00|
+----------------------------------------+
-- to_unix_timestamp
SELECT to_unix_timestamp('2016-04-08', 'yyyy-MM-dd');
+----------------------------------------+
|to_unix_timestamp(2016-04-08, yyyy-MM-dd)|
+----------------------------------------+
|                                1460041200|
+----------------------------------------+
-- to_utc_timestamp
SELECT to_utc_timestamp('2016-08-31', 'Asia/Seoul');
+----------------------------------------+
|to_utc_timestamp(2016-08-31, Asia/Seoul)|
+----------------------------------------+
|                     2016-08-30 15:00:00|
+----------------------------------------+
```

```
-- trunc
SELECT trunc('2019-08-04', 'week');
+-----------------------+
|trunc(2019-08-04, week)|
+-----------------------+
|             2019-07-29|
+-----------------------+
SELECT trunc('2019-08-04', 'quarter');
+--------------------------+
|trunc(2019-08-04, quarter)|
+--------------------------+
|                2019-07-01|
+--------------------------+
SELECT trunc('2009-02-12', 'MM');
+---------------------+
|trunc(2009-02-12, MM)|
+---------------------+
|           2009-02-01|
+---------------------+
SELECT trunc('2015-10-27', 'YEAR');
+-----------------------+
|trunc(2015-10-27, YEAR)|
+-----------------------+
|             2015-01-01|
+-----------------------+
-- try_to_timestamp
SELECT try_to_timestamp('2016-12-31 00:12:00');
+-------------------------------------+
|try_to_timestamp(2016-12-31 00:12:00)|
+-------------------------------------+
|                  2016-12-31 00:12:00|
+-------------------------------------+
SELECT try_to_timestamp('2016-12-31', 'yyyy-MM-dd');
+---------------------------------------+
|try_to_timestamp(2016-12-31, yyyy-MM-dd)|
+---------------------------------------+
|                     2016-12-31 00:00:00|
+---------------------------------------+
SELECT try_to_timestamp('foo', 'yyyy-MM-dd');
+---------------------------------+
|try_to_timestamp(foo, yyyy-MM-dd)|
+---------------------------------+
|                             NULL|
+---------------------------------+
```

```
-- unix_date
SELECT unix_date(DATE("1970-01-02"));
+--------------------+
|unix_date(1970-01-02)|
+--------------------+
|                   1|
+--------------------+
-- unix_micros
SELECT unix_micros(TIMESTAMP('1970-01-01 00:00:01Z'));
+--------------------------------+
|unix_micros(1970-01-01 00:00:01Z)|
+--------------------------------+
|                         1000000|
+--------------------------------+
-- unix_millis
SELECT unix_millis(TIMESTAMP('1970-01-01 00:00:01Z'));
+--------------------------------+
|unix_millis(1970-01-01 00:00:01Z)|
+--------------------------------+
|                            1000|
+--------------------------------+
-- unix_seconds
SELECT unix_seconds(TIMESTAMP('1970-01-01 00:00:01Z'));
+--------------------------------+
|unix_seconds(1970-01-01 00:00:01Z)|
+--------------------------------+
|                                1|
+--------------------------------+
-- unix_timestamp
SELECT unix_timestamp();
+-----------------------------------------------------------+
|unix_timestamp(current_timestamp(), yyyy-MM-dd HH:mm:ss)|
+-----------------------------------------------------------+
|                                                 1708760216|
+-----------------------------------------------------------+
SELECT unix_timestamp('2016-04-08', 'yyyy-MM-dd');
+-------------------------------------+
|unix_timestamp(2016-04-08, yyyy-MM-dd)|
+-------------------------------------+
|                           1460041200|
+-------------------------------------+
-- weekday
SELECT weekday('2009-07-30');
+------------------+
```

```
|weekday(2009-07-30)|
+-------------------+
|                  3|
+-------------------+
-- weekofyear
SELECT weekofyear('2008-02-20');
+----------------------+
|weekofyear(2008-02-20)|
+----------------------+
|                     8|
+----------------------+
-- window
SELECT a, window.start, window.end, count(*) as cnt FROM VALUES ('A1', '2021-01-01
 00:00:00'), ('A1', '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:06:00'), ('A2',
 '2021-01-01 00:01:00') AS tab(a, b) GROUP by a, window(b, '5 minutes') ORDER BY a,
 start;
+---+-------------------+-------------------+---+
|  a|              start|                end|cnt|
+---+-------------------+-------------------+---+
| A1|2021-01-01 00:00:00|2021-01-01 00:05:00|  2|
| A1|2021-01-01 00:05:00|2021-01-01 00:10:00|  1|
| A2|2021-01-01 00:00:00|2021-01-01 00:05:00|  1|
+---+-------------------+-------------------+---+
SELECT a, window.start, window.end, count(*) as cnt FROM VALUES ('A1', '2021-01-01
 00:00:00'), ('A1', '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:06:00'), ('A2',
 '2021-01-01 00:01:00') AS tab(a, b) GROUP by a, window(b, '10 minutes', '5 minutes')
 ORDER BY a, start;
+---+-------------------+-------------------+---+
|  a|              start|                end|cnt|
+---+-------------------+-------------------+---+
| A1|2020-12-31 23:55:00|2021-01-01 00:05:00|  2|
| A1|2021-01-01 00:00:00|2021-01-01 00:10:00|  3|
| A1|2021-01-01 00:05:00|2021-01-01 00:15:00|  1|
| A2|2020-12-31 23:55:00|2021-01-01 00:05:00|  1|
| A2|2021-01-01 00:00:00|2021-01-01 00:10:00|  1|
+---+-------------------+-------------------+---+
-- window_time
SELECT a, window.start as start, window.end as end, window_time(window), cnt FROM
 (SELECT a, window, count(*) as cnt FROM VALUES ('A1', '2021-01-01 00:00:00'), ('A1',
 '2021-01-01 00:04:30'), ('A1', '2021-01-01 00:06:00'), ('A2', '2021-01-01 00:01:00')
 AS tab(a, b) GROUP by a, window(b, '5 minutes') ORDER BY a, window.start);
+---+-------------------+-------------------+--------------------+---+
|  a|              start|                end| window_time(window)|cnt|
+---+-------------------+-------------------+--------------------+---+
```

```
| A1|2021-01-01 00:00:00|2021-01-01 00:05:00|2021-01-01 00:04:...|  2|
| A1|2021-01-01 00:05:00|2021-01-01 00:10:00|2021-01-01 00:09:...|  1|
| A2|2021-01-01 00:00:00|2021-01-01 00:05:00|2021-01-01 00:04:...|  1|
+---+-------------------+-------------------+-------------------+---+
-- year
SELECT year('2016-07-30');
+----------------+
|year(2016-07-30)|
+----------------+
|            2016|
+----------------+
```

Fonctions d’agrégation

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Les fonctions d'agrégation agissent sur les valeurs des lignes pour effectuer des calculs mathématiques tels que la somme, la moyenne, le comptage, minimum/maximum les valeurs, l'écart type et l'estimation, ainsi que certaines opérations non mathématiques.

Syntaxe

```
aggregate_function(input1 [, input2, ...]) FILTER (WHERE boolean_expression)
```

Paramètres

- `boolean_expression`- Spécifie toute expression dont le résultat est un booléen de type booléen. Deux expressions ou plus peuvent être combinées à l'aide des opérateurs logiques (AND, OR).

Fonctions d'agrégation ordonnées

Ces fonctions d'agrégation utilisent une syntaxe différente de celle des autres fonctions d'agrégation afin de spécifier une expression (généralement un nom de colonne) permettant de classer les valeurs.

Syntaxe

```
{ PERCENTILE_CONT | PERCENTILE_DISC }(percentile) WITHIN GROUP (ORDER BY
 { order_by_expression [ ASC | DESC ] [ NULLS { FIRST | LAST } ] [ , ... ] }) FILTER
 (WHERE boolean_expression)
```

Paramètres

- `percentile`- Le percentile de la valeur que vous souhaitez rechercher. Le percentile doit être une constante comprise entre 0,0 et 1,0.
- `order_by_expression`- L'expression (généralement un nom de colonne) permettant d'ordonner les valeurs avant de les agréger.
- `boolean_expression`- Spécifie toute expression dont le résultat est un booléen de type booléen. Deux expressions ou plus peuvent être combinées à l'aide des opérateurs logiques (AND, OR).

Exemples

```
CREATE OR REPLACE TEMPORARY VIEW basic_pays AS SELECT * FROM VALUES
('Jane Doe','Accounting',8435),
('Akua Mansa','Accounting',9998),
('John Doe','Accounting',8992),
('Juan Li','Accounting',8870),
('Carlos Salazar','Accounting',11472),
('Arnav Desai','Accounting',6627),
('Saanvi Sarkar','IT',8113),
('Shirley Rodriguez','IT',5186),
('Nikki Wolf','Sales',9181),
('Alejandro Rosalez','Sales',9441),
('Nikhil Jayashankar','Sales',6660),
('Richard Roe','Sales',10563),
('Pat Candella','SCM',10449),
('Gerard Hernandez','SCM',6949),
('Pamela Castillo','SCM',11303),
('Paulo Santos','SCM',11798),
('Jorge Souza','SCM',10586)
AS basic_pays(employee_name, department, salary);
SELECT * FROM basic_pays;
+------------------+----------+------+
|    employee_name  |department|salary|
+------------------+----------+------+
| Arnav Desai       |Accounting|  6627|
| Jorge Souza       |       SCM| 10586|
| Jane Doe          |Accounting|  8435|
```

```
| Nikhil Jayashankar|     Sales|  6660|
| Diego Vanauf      |     Sales| 10563|
| Carlos Salazar    |Accounting| 11472|
| Gerard Hernandez  |       SCM|  6949|
| John Doe          |Accounting|  8992|
| Nikki Wolf        |     Sales|  9181|
| Paulo Santos      |       SCM| 11798|
| Saanvi Sarkar     |        IT|  8113|
| Shirley Rodriguez |        IT|  5186|
| Pat Candella      |       SCM| 10449|
| Akua Mansa        |Accounting|  9998|
| Pamela Castillo   |       SCM| 11303|
| Alejandro Rosalez |     Sales|  9441|
| Juan Li           |Accounting|  8870|
+-------------------+----------+------+
SELECT
department,
percentile_cont(0.25) WITHIN GROUP (ORDER BY salary) AS pc1,
percentile_cont(0.25) WITHIN GROUP (ORDER BY salary) FILTER (WHERE employee_name LIKE
 '%Bo%') AS pc2,
percentile_cont(0.25) WITHIN GROUP (ORDER BY salary DESC) AS pc3,
percentile_cont(0.25) WITHIN GROUP (ORDER BY salary DESC) FILTER (WHERE employee_name
 LIKE '%Bo%') AS pc4,
percentile_disc(0.25) WITHIN GROUP (ORDER BY salary) AS pd1,
percentile_disc(0.25) WITHIN GROUP (ORDER BY salary) FILTER (WHERE employee_name LIKE
 '%Bo%') AS pd2,
percentile_disc(0.25) WITHIN GROUP (ORDER BY salary DESC) AS pd3,
percentile_disc(0.25) WITHIN GROUP (ORDER BY salary DESC) FILTER (WHERE employee_name
 LIKE '%Bo%') AS pd4
FROM basic_pays
GROUP BY department
ORDER BY department;
+----------+-------+--------+-------+--------+-----+-----+-----+-----+
|department|    pc1|     pc2|    pc3|     pc4|  pd1|  pd2|  pd3|  pd4|
+----------+-------+--------+-------+--------+-----+-----+-----+-----+
|Accounting|8543.75| 7838.25| 9746.5|10260.75| 8435| 6627| 9998|11472|
|        IT|5917.75|    NULL|7381.25|    NULL| 5186| NULL| 8113| NULL|
|     Sales|8550.75|    NULL| 9721.5|    NULL| 6660| NULL|10563| NULL|
|       SCM|10449.0|10786.25|11303.0|11460.75|10449|10449|11303|11798|
+----------+-------+--------+-------+--------+-----+-----+-----+-----+
```

Fonctions conditionnelles

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Fonction	Description
fusionner (expr1, expr 2,...)	Renvoie le premier argument non nul s'il existe. Null dans le cas contraire.
si (expr1, expr 2, expr 3)	Si la `expr1` valeur est vraie, elle est renvoyée `expr2` ; dans le cas contraire, elle renvoie`expr3`.
ifnull (expr1, expr 2)	Renvoie `expr2` si la valeur `expr1` est nulle ou `expr1` non.
nanvl (expr1, expr 2)	Renvoie `expr1` si ce n'est pas NaN ou `expr2` non.
nul (expr 1, expr 2)	Renvoie la valeur nulle si elle `expr1` est égale à`expr2`, ou `expr1` non.
nvl (expr1, expr 2)	Renvoie `expr2` si la valeur `expr1` est nulle ou `expr1` non.
nvl2 (expr1, expr 2, expr 3)	Renvoie `expr2` s'`expr1`il n'est pas nul ou `expr3` non.
CAS OÙ EXPR1 PUIS EXPR2 [QUAND EXPR3 PUIS EXPR4] * [ELSE expr5] FIN	When `expr1` = true, renvoie `expr2` ; sinon when `expr3` = true, renvoie `expr4` ; sinon renvoie`expr5`.

Exemples

```
-- coalesce
SELECT coalesce(NULL, 1, NULL);
+----------------------+
|coalesce(NULL, 1, NULL)|
+----------------------+
|                     1|
+----------------------+
-- if
SELECT if(1 < 2, 'a', 'b');
+------------------+
|(IF((1 < 2), a, b))|
+------------------+
|                 a|
+------------------+
-- ifnull
SELECT ifnull(NULL, array('2'));
+----------------------+
|ifnull(NULL, array(2))|
+----------------------+
|                  [2]|
+----------------------+
-- nanvl
SELECT nanvl(cast('NaN' as double), 123);
+------------------------------+
|nanvl(CAST(NaN AS DOUBLE), 123)|
+------------------------------+
|                         123.0|
+------------------------------+
-- nullif
SELECT nullif(2, 2);
+------------+
|nullif(2, 2)|
+------------+
|        NULL|
+------------+
-- nvl
SELECT nvl(NULL, array('2'));
+------------------+
|nvl(NULL, array(2))|
+------------------+
|               [2]|
+------------------+
-- nvl2
```

```
SELECT nvl2(NULL, 2, 1);
+----------------+
|nvl2(NULL, 2, 1)|
+----------------+
|               1|
+----------------+
-- when
SELECT CASE WHEN 1 > 0 THEN 1 WHEN 2 > 0 THEN 2.0 ELSE 1.2 END;
+-------------------------------------------------------------+
|CASE WHEN (1 > 0) THEN 1 WHEN (2 > 0) THEN 2.0 ELSE 1.2 END|
+-------------------------------------------------------------+
|                                                          1.0|
+-------------------------------------------------------------+
SELECT CASE WHEN 1 < 0 THEN 1 WHEN 2 > 0 THEN 2.0 ELSE 1.2 END;
+-------------------------------------------------------------+
|CASE WHEN (1 < 0) THEN 1 WHEN (2 > 0) THEN 2.0 ELSE 1.2 END|
+-------------------------------------------------------------+
|                                                          2.0|
+-------------------------------------------------------------+
SELECT CASE WHEN 1 < 0 THEN 1 WHEN 2 < 0 THEN 2.0 END;
+-------------------------------------------------+
|CASE WHEN (1 < 0) THEN 1 WHEN (2 < 0) THEN 2.0 END|
+-------------------------------------------------+
|                                             NULL|
+-------------------------------------------------+
```

Fonctions JSON

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Fonction	Description
from_json (JSONStr, schéma [, options])	Renvoie une valeur de structure avec le `JSONStr` et le `schema` donnés.

Fonction	Description
get_json_object (json_txt, chemin)	Extrait un objet JSON de `path`.
json_array_length (JSONArray)	Renvoie le nombre d'éléments du tableau JSON le plus externe.
json_object_keys (json_object)	Renvoie toutes les clés de l'objet JSON le plus externe sous forme de tableau.
json_tuple (JSONStr, p1, p2,..., pn)	Renvoie un tuple comme la fonction get_json_object, mais il prend plusieurs noms. Tous les paramètres d'entrée et les types de colonnes de sortie sont des chaînes de caractères.
schema_of_json (json [, options])	Renvoie le schéma au format DDL d'une chaîne JSON.
to_json (expr [, options])	Renvoie une chaîne JSON avec une valeur de structure donnée

Exemples

```
-- from_json
SELECT from_json('{"a":1, "b":0.8}', 'a INT, b DOUBLE');
+---------------------------+
| from_json({"a":1, "b":0.8}) |
+---------------------------+
| {1, 0.8}                  |
+---------------------------+

SELECT from_json('{"time":"26/08/2015"}', 'time Timestamp', map('timestampFormat', 'dd/
MM/yyyy'));
+--------------------------------+
```

```
| from_json({"time":"26/08/2015"}) |
+----------------------------------+
| {2015-08-26 00:00...             |
+----------------------------------+

SELECT from_json('{"teacher": "Alice", "student": [{"name": "Bob", "rank": 1}, {"name":
 "Charlie", "rank": 2}]}', 'STRUCT<teacher: STRING, student: ARRAY<STRUCT<name: STRING,
 rank: INT>>>');
+-------------------------------------------------------------------------------------------------------
+
| from_json({"teacher": "Alice", "student": [{"name": "Bob", "rank": 1}, {"name":
 "Charlie", "rank": 2}]}) |
+-------------------------------------------------------------------------------------------------------
+
| {Alice, [{Bob, 1}...
                   |
+-------------------------------------------------------------------------------------------------------
+

-- get_json_object
SELECT get_json_object('{"a":"b"}', '$.a');
+--------------------------------+
| get_json_object({"a":"b"}, $.a) |
+--------------------------------+
| b                              |
+--------------------------------+

-- json_array_length
SELECT json_array_length('[1,2,3,4]');
+----------------------------+
| json_array_length([1,2,3,4]) |
+----------------------------+
| 4                          |
+----------------------------+

SELECT json_array_length('[1,2,3,{"f1":1,"f2":[5,6]},4]');
+-------------------------------------------------+
| json_array_length([1,2,3,{"f1":1,"f2":[5,6]},4]) |
+-------------------------------------------------+
| 5                                               |
+-------------------------------------------------+

SELECT json_array_length('[1,2');
+----------------------+
```

```
| json_array_length([1,2) |
+-------------------------+
| NULL                    |
+-------------------------+

-- json_object_keys
SELECT json_object_keys('{}');
+--------------------+
| json_object_keys({}) |
+--------------------+
| []                 |
+--------------------+

SELECT json_object_keys('{"key": "value"}');
+-----------------------------------+
| json_object_keys({"key": "value"}) |
+-----------------------------------+
| [key]                             |
+-----------------------------------+

SELECT json_object_keys('{"f1":"abc","f2":{"f3":"a", "f4":"b"}}');
+-----------------------------------------------------------+
| json_object_keys({"f1":"abc","f2":{"f3":"a", "f4":"b"}}) |
+-----------------------------------------------------------+
| [f1, f2]                                                  |
+-----------------------------------------------------------+

-- json_tuple
SELECT json_tuple('{"a":1, "b":2}', 'a', 'b');
+---+---+
| c0| c1|
+---+---+
|  1|  2|
+---+---+

-- schema_of_json
SELECT schema_of_json('[{"col":0}]');
+----------------------------+
| schema_of_json([{"col":0}]) |
+----------------------------+
| ARRAY<STRUCT<col:...      |
+----------------------------+

SELECT schema_of_json('[{"col":01}]', map('allowNumericLeadingZeros', 'true'));
```

```
+----------------------------+
| schema_of_json([{"col":01}]) |
+----------------------------+
| ARRAY<STRUCT<col:...       |
+----------------------------+

-- to_json
SELECT to_json(named_struct('a', 1, 'b', 2));
+---------------------------------+
| to_json(named_struct(a, 1, b, 2)) |
+---------------------------------+
| {"a":1,"b":2}                   |
+---------------------------------+

SELECT to_json(named_struct('time', to_timestamp('2015-08-26', 'yyyy-MM-dd')),
 map('timestampFormat', 'dd/MM/yyyy'));
+------------------------------------------------------------------------+
| to_json(named_struct(time, to_timestamp(2015-08-26, yyyy-MM-dd))) |
+------------------------------------------------------------------------+
| {"time":"26/08/20...                                                   |
+------------------------------------------------------------------------+

SELECT to_json(array(named_struct('a', 1, 'b', 2)));
+-----------------------------------------+
| to_json(array(named_struct(a, 1, b, 2))) |
+-----------------------------------------+
| [{"a":1,"b":2}]                         |
+-----------------------------------------+

SELECT to_json(map('a', named_struct('b', 1)));
+-----------------------------------+
| to_json(map(a, named_struct(b, 1))) |
+-----------------------------------+
| {"a":{"b":1}}                     |
+-----------------------------------+

SELECT to_json(map(named_struct('a', 1),named_struct('b', 2)));
+-------------------------------------------------------+
| to_json(map(named_struct(a, 1), named_struct(b, 2))) |
+-------------------------------------------------------+
| {"[1]":{"b":2}}                                       |
+-------------------------------------------------------+

SELECT to_json(map('a', 1));
```

```
+------------------+
| to_json(map(a, 1)) |
+------------------+
| {"a":1}          |
+------------------+

SELECT to_json(array(map('a', 1)));
+-------------------------+
| to_json(array(map(a, 1))) |
+-------------------------+
| [{"a":1}]               |
+-------------------------+
```

Fonctions de tableau

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Fonction	Description
tableau (expr,...)	Renvoie un tableau avec les éléments donnés.
array_append (tableau, élément)	Ajoutez l'élément à la fin du tableau passé en premier argument. Le type d'élément doit être similaire au type des éléments du tableau. L'élément nul est également ajouté au tableau. Mais si le tableau est passé, si la valeur est NULL, la sortie est NULL
array_compact (tableau)	Supprime les valeurs nulles du tableau.
array_contains (tableau, valeur)	Renvoie vrai si le tableau contient la valeur.
array_distinct (tableau)	Supprime les valeurs dupliquées du tableau.
array_except (tableau1, tableau2)	Renvoie un tableau des éléments du tableau 1 mais pas du tableau 2, sans doublons.

Fonction	Description
array_insert (x, pos, val)	Place val dans l'index pos du tableau x. Les indices du tableau commencent à 1. L'indice négatif maximal est -1 pour lequel la fonction insère un nouvel élément après le dernier élément actuel. L'index au-dessus de la taille du tableau ajoute le tableau, ou le préfixe si l'indice est négatif, avec des éléments « nuls ».
array_intersect (matrice1, matrice2)	Renvoie un tableau des éléments situés à l'intersection de array1 et array2, sans doublons.
array_join (tableau, délimiteur [, NullRepla cement])	Concatène les éléments du tableau donné à l'aide du délimiteur et d'une chaîne facultative pour remplacer les valeurs nulles. Si aucune valeur n'est définie pour NullReplacement, toute valeur nulle est filtrée.
array_max (tableau)	Renvoie la valeur maximale du tableau. NaN est supérieur à tous les éléments non NaN pour le double/float type. Les éléments NULL sont ignorés.
array_min (tableau)	Renvoie la valeur minimale du tableau. NaN est supérieur à tous les éléments non NaN pour le double/float type. Les éléments NULL sont ignorés.
array_position (tableau, élément)	Renvoie l'index (basé sur 1) du premier élément correspondant du tableau sous forme longue, ou 0 si aucune correspondance n'est trouvée.

Fonction	Description
array_prepend (tableau, élément)	Ajoutez l'élément au début du tableau passé en premier argument. Le type d'élément doit être le même que le type des éléments du tableau. L'élément nul est également ajouté au tableau. Mais si le tableau transmis est NULL, la sortie est NULL
array_remove (tableau, élément)	Supprime tous les éléments égaux à un élément du tableau.
array_repeat (élément, nombre)	Renvoie le tableau contenant le nombre de fois où les éléments sont dénombrés.
array_union (matrice1, matrice2)	Renvoie un tableau des éléments de l'union de array1 et array2, sans doublons.
arrays_overlap (a1, a2)	Renvoie vrai si a1 contient au moins un élément non nul présent également dans a2. Si les tableaux n'ont aucun élément commun, qu'ils ne sont pas vides et que l'un d'eux contient un élément nul, la valeur null est renvoyée, false dans le cas contraire.
tableaux_zip (a1, a2,...)	Renvoie un tableau fusionné de structures dans lequel la N-ième structure contient toutes les N-ièmes valeurs des tableaux d'entrée.
aplatir () arrayOfArrays	Transforme un tableau de tableaux en un seul tableau.
obtenir (tableau, index)	Renvoie l'élément du tableau à un index donné (basé sur 0). Si l'index pointe en dehors des limites du tableau, cette fonction renvoie la valeur NULL.

Fonction	Description
séquence (démarrage, arrêt, étape)	Génère un tableau d'éléments du début à la fin (inclus), en les incrémentant pas à pas. Le type des éléments renvoyés est le même que celui des expressions d'argument. Les types pris en charge sont les suivants : octet, court, entier, long, date, horodatage. Les expressions de début et d'arrêt doivent être résolues dans le même type. Si les expressions de début et de fin sont du type « date » ou « horodatage », l'expression d'étape doit être du type « intervall e », « intervalle année-mois » ou « intervall e jour-heure », sinon du même type que les expressions de début et de fin.
shuffle (tableau)	Renvoie une permutation aléatoire du tableau donné.
tranche (x, début, longueur)	Sous-ensembles le tableau x à partir du début de l'index (les indices du tableau commencent à 1, ou à partir de la fin si le début est négatif) avec la longueur spécifiée.
sort_array (tableau [, ordre croissant])	Trie le tableau d'entrée par ordre croissant ou décroissant selon l'ordre naturel des éléments du tableau. NaN est supérieur à tous les éléments non NaN pour le double/float type. Les éléments nuls seront placés au début du tableau renvoyé par ordre croissant ou à la fin du tableau renvoyé par ordre décroissant.

Exemples

```
-- array
SELECT array(1, 2, 3);
+--------------+
```

```
|array(1, 2, 3)|
+--------------+
|     [1, 2, 3]|
+--------------+
-- array_append
SELECT array_append(array('b', 'd', 'c', 'a'), 'd');
+----------------------------------+
|array_append(array(b, d, c, a), d)|
+----------------------------------+
|                   [b, d, c, a, d]|
+----------------------------------+
SELECT array_append(array(1, 2, 3, null), null);
+---------------------------------------+
|array_append(array(1, 2, 3, NULL), NULL)|
+---------------------------------------+
|                    [1, 2, 3, NULL, N...|
+---------------------------------------+
SELECT array_append(CAST(null as Array<Int>), 2);
+---------------------+
|array_append(NULL, 2)|
+---------------------+
|                 NULL|
+---------------------+
-- array_compact
SELECT array_compact(array(1, 2, 3, null));
+----------------------------------+
|array_compact(array(1, 2, 3, NULL))|
+----------------------------------+
|                          [1, 2, 3]|
+----------------------------------+
SELECT array_compact(array("a", "b", "c"));
+----------------------------+
|array_compact(array(a, b, c))|
+----------------------------+
|                    [a, b, c]|
+----------------------------+
-- array_contains
SELECT array_contains(array(1, 2, 3), 2);
+---------------------------------+
|array_contains(array(1, 2, 3), 2)|
+---------------------------------+
|                             true|
+---------------------------------+
-- array_distinct
```

```
SELECT array_distinct(array(1, 2, 3, null, 3));
+--------------------------------------+
|array_distinct(array(1, 2, 3, NULL, 3))|
+--------------------------------------+
|                        [1, 2, 3, NULL]|
+--------------------------------------+
-- array_except
SELECT array_except(array(1, 2, 3), array(1, 3, 5));
+-------------------------------------------+
|array_except(array(1, 2, 3), array(1, 3, 5))|
+-------------------------------------------+
|                                        [2]|
+-------------------------------------------+
-- array_insert
SELECT array_insert(array(1, 2, 3, 4), 5, 5);
+-------------------------------------+
|array_insert(array(1, 2, 3, 4), 5, 5)|
+-------------------------------------+
|                     [1, 2, 3, 4, 5]|
+-------------------------------------+
SELECT array_insert(array(5, 4, 3, 2), -1, 1);
+-------------------------------------+
|array_insert(array(5, 4, 3, 2), -1, 1)|
+-------------------------------------+
|                      [5, 4, 3, 2, 1]|
+-------------------------------------+
SELECT array_insert(array(5, 3, 2, 1), -4, 4);
+-------------------------------------+
|array_insert(array(5, 3, 2, 1), -4, 4)|
+-------------------------------------+
|                      [5, 4, 3, 2, 1]|
+-------------------------------------+
-- array_intersect
SELECT array_intersect(array(1, 2, 3), array(1, 3, 5));
+-----------------------------------------------+
|array_intersect(array(1, 2, 3), array(1, 3, 5))|
+-----------------------------------------------+
|                                         [1, 3]|
+-----------------------------------------------+
-- array_join
SELECT array_join(array('hello', 'world'), ' ');
+----------------------------------+
|array_join(array(hello, world),  )|
+----------------------------------+
```

```
|                       hello world|
+----------------------------------+
SELECT array_join(array('hello', null ,'world'), ' ');
+---------------------------------------+
|array_join(array(hello, NULL, world),  )|
+---------------------------------------+
|                            hello world|
+---------------------------------------+
SELECT array_join(array('hello', null ,'world'), ' ', ',');
+-----------------------------------------+
|array_join(array(hello, NULL, world),  , ,)|
+-----------------------------------------+
|                             hello , world|
+-----------------------------------------+
-- array_max
SELECT array_max(array(1, 20, null, 3));
+--------------------------------+
|array_max(array(1, 20, NULL, 3))|
+--------------------------------+
|                              20|
+--------------------------------+
-- array_min
SELECT array_min(array(1, 20, null, 3));
+--------------------------------+
|array_min(array(1, 20, NULL, 3))|
+--------------------------------+
|                               1|
+--------------------------------+
-- array_position
SELECT array_position(array(312, 773, 708, 708), 708);
+-----------------------------------------------+
|array_position(array(312, 773, 708, 708), 708)|
+-----------------------------------------------+
|                                              3|
+-----------------------------------------------+
SELECT array_position(array(312, 773, 708, 708), 414);
+-----------------------------------------------+
|array_position(array(312, 773, 708, 708), 414)|
+-----------------------------------------------+
|                                              0|
+-----------------------------------------------+
-- array_prepend
SELECT array_prepend(array('b', 'd', 'c', 'a'), 'd');
+----------------------------------+
```

```
|array_prepend(array(b, d, c, a), d)|
+-----------------------------------+
|                    [d, b, d, c, a]|
+-----------------------------------+
SELECT array_prepend(array(1, 2, 3, null), null);
+----------------------------------------+
|array_prepend(array(1, 2, 3, NULL), NULL)|
+----------------------------------------+
|                     [NULL, 1, 2, 3, N...|
+----------------------------------------+
SELECT array_prepend(CAST(null as Array<Int>), 2);
+----------------------+
|array_prepend(NULL, 2)|
+----------------------+
|                  NULL|
+----------------------+
-- array_remove
SELECT array_remove(array(1, 2, 3, null, 3), 3);
+---------------------------------------+
|array_remove(array(1, 2, 3, NULL, 3), 3)|
+---------------------------------------+
|                           [1, 2, NULL]|
+---------------------------------------+
-- array_repeat
SELECT array_repeat('123', 2);
+--------------------+
|array_repeat(123, 2)|
+--------------------+
|          [123, 123]|
+--------------------+
-- array_union
SELECT array_union(array(1, 2, 3), array(1, 3, 5));
+-------------------------------------------+
|array_union(array(1, 2, 3), array(1, 3, 5))|
+-------------------------------------------+
|                               [1, 2, 3, 5]|
+-------------------------------------------+
-- arrays_overlap
SELECT arrays_overlap(array(1, 2, 3), array(3, 4, 5));
+-----------------------------------------------+
|arrays_overlap(array(1, 2, 3), array(3, 4, 5))|
+-----------------------------------------------+
|                                           true|
+-----------------------------------------------+
```

```
-- arrays_zip
SELECT arrays_zip(array(1, 2, 3), array(2, 3, 4));
+-----------------------------------------+
|arrays_zip(array(1, 2, 3), array(2, 3, 4))|
+-----------------------------------------+
|                      [{1, 2}, {2, 3}, ...|
+-----------------------------------------+
SELECT arrays_zip(array(1, 2), array(2, 3), array(3, 4));
+-------------------------------------------------+
|arrays_zip(array(1, 2), array(2, 3), array(3, 4))|
+-------------------------------------------------+
|                             [{1, 2, 3}, {2, 3...|
+-------------------------------------------------+
-- flatten
SELECT flatten(array(array(1, 2), array(3, 4)));
+---------------------------------------+
|flatten(array(array(1, 2), array(3, 4)))|
+---------------------------------------+
|                           [1, 2, 3, 4]|
+---------------------------------------+
-- get
SELECT get(array(1, 2, 3), 0);
+----------------------+
|get(array(1, 2, 3), 0)|
+----------------------+
|                     1|
+----------------------+
SELECT get(array(1, 2, 3), 3);
+----------------------+
|get(array(1, 2, 3), 3)|
+----------------------+
|                  NULL|
+----------------------+
SELECT get(array(1, 2, 3), -1);
+-----------------------+
|get(array(1, 2, 3), -1)|
+-----------------------+
|                   NULL|
+-----------------------+
-- sequence
SELECT sequence(1, 5);
+---------------+
| sequence(1, 5)|
+---------------+
```

```
|[1, 2, 3, 4, 5]|
+---------------+
SELECT sequence(5, 1);
+---------------+
| sequence(5, 1)|
+---------------+
|[5, 4, 3, 2, 1]|
+---------------+
SELECT sequence(to_date('2018-01-01'), to_date('2018-03-01'), interval 1 month);
+--------------------------------------------------------------------------+
|sequence(to_date(2018-01-01), to_date(2018-03-01), INTERVAL '1' MONTH)|
+--------------------------------------------------------------------------+
|                                                  [2018-01-01, 2018...|
+--------------------------------------------------------------------------+
SELECT sequence(to_date('2018-01-01'), to_date('2018-03-01'), interval '0-1' year to
 month);
+-----------------------------------------------------------------------------------+
|sequence(to_date(2018-01-01), to_date(2018-03-01), INTERVAL '0-1' YEAR TO MONTH)|
+-----------------------------------------------------------------------------------+
|                                                               [2018-01-01, 2018...|
+-----------------------------------------------------------------------------------+
-- shuffle
SELECT shuffle(array(1, 20, 3, 5));
+---------------------------+
|shuffle(array(1, 20, 3, 5))|
+---------------------------+
|              [5, 1, 20, 3]|
+---------------------------+
SELECT shuffle(array(1, 20, null, 3));
+-----------------------------+
|shuffle(array(1, 20, NULL, 3))|
+-----------------------------+
|              [1, NULL, 20, 3]|
+-----------------------------+
-- slice
SELECT slice(array(1, 2, 3, 4), 2, 2);
+-----------------------------+
|slice(array(1, 2, 3, 4), 2, 2)|
+-----------------------------+
|                        [2, 3]|
+-----------------------------+
SELECT slice(array(1, 2, 3, 4), -2, 2);
+-----------------------------+
|slice(array(1, 2, 3, 4), -2, 2)|
```

```
+------------------------------+
|                        [3, 4]|
+------------------------------+
-- sort_array
SELECT sort_array(array('b', 'd', null, 'c', 'a'), true);
+---------------------------------------+
|sort_array(array(b, d, NULL, c, a), true)|
+---------------------------------------+
|                      [NULL, a, b, c, d]|
+---------------------------------------+
```

Fonctions de fenêtrage

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Les fonctions de fenêtre opèrent sur un groupe de lignes, appelé fenêtre, et calculent une valeur de retour pour chaque ligne en fonction du groupe de lignes. Les fonctions de fenêtre sont utiles pour traiter des tâches telles que le calcul d'une moyenne mobile, le calcul d'une statistique cumulée ou l'accès à la valeur des lignes en fonction de la position relative de la ligne en cours.

Syntaxe

```
window_function [ nulls_option ] OVER ( [ { PARTITION | DISTRIBUTE } BY
 partition_col_name = partition_col_val ( [ , ... ] ) ] { ORDER | SORT } BY expression
 [ ASC | DESC ] [ NULLS { FIRST | LAST } ] [ , ... ] [ window_frame ] )
```

Paramètres

- Fonctions de classement

  Syntaxe : `RANK | DENSE_RANK | PERCENT_RANK | NTILE | ROW_NUMBER`

  Fonctions analytiques

  Syntaxe : `CUME_DIST | LAG | LEAD | NTH_VALUE | FIRST_VALUE | LAST_VALUE`

  Fonctions d’agrégation

Syntaxe : `MAX | MIN | COUNT | SUM | AVG | ...`

- `nulls_option`- Spécifie s'il faut ou non ignorer les valeurs nulles lors de l'évaluation de la fonction de fenêtre. RESPECTER LES VALEURS NULLS signifie ne pas ignorer les valeurs nulles, tandis que IGNORER NULLS signifie les ignorer. Si ce n'est pas spécifié, la valeur par défaut est RESPECT NULLS.

  Syntaxe : `{ IGNORE | RESPECT } NULLS`

  Remarque : `Only LAG` | `LEAD` | `NTH_VALUE` | `FIRST_VALUE` | `LAST_VALUE` peut être utilisé avec`IGNORE NULLS`.
- `window_frame`- Spécifie sur quelle ligne commencer la fenêtre et où la terminer.

  Syntaxe : `{ RANGE | ROWS } { frame_start | BETWEEN frame_start AND frame_end }`

  frame_start et frame_end ont la syntaxe suivante :

  Syntaxe : `UNBOUNDED PRECEDING | offset PRECEDING | CURRENT ROW | offset FOLLOWING | UNBOUNDED FOLLOWING`

  offset : indique le décalage par rapport à la position de la ligne en cours.

  Remarque Si frame_end est omis, la valeur par défaut est CURRENT ROW.

Exemples

```
CREATE TABLE employees (name STRING, dept STRING, salary INT, age INT);
INSERT INTO employees VALUES ("Lisa", "Sales", 10000, 35);
INSERT INTO employees VALUES ("Evan", "Sales", 32000, 38);
INSERT INTO employees VALUES ("Fred", "Engineering", 21000, 28);
INSERT INTO employees VALUES ("Alex", "Sales", 30000, 33);
INSERT INTO employees VALUES ("Tom", "Engineering", 23000, 33);
INSERT INTO employees VALUES ("Jane", "Marketing", 29000, 28);
INSERT INTO employees VALUES ("Jeff", "Marketing", 35000, 38);
INSERT INTO employees VALUES ("Paul", "Engineering", 29000, 23);
INSERT INTO employees VALUES ("Chloe", "Engineering", 23000, 25);
SELECT * FROM employees;
+-----+-----------+------+-----+
| name|       dept|salary|  age|
+-----+-----------+------+-----+
```

```
|Chloe|Engineering| 23000|   25|
| Fred|Engineering| 21000|   28|
| Paul|Engineering| 29000|   23|
|Helen|  Marketing| 29000|   40|
|  Tom|Engineering| 23000|   33|
| Jane|  Marketing| 29000|   28|
| Jeff|  Marketing| 35000|   38|
| Evan|      Sales| 32000|   38|
| Lisa|      Sales| 10000|   35|
| Alex|      Sales| 30000|   33|
+-----+-----------+------+-----+
SELECT name, dept, salary, RANK() OVER (PARTITION BY dept ORDER BY salary) AS rank FROM
 employees;
+-----+-----------+------+----+
| name|       dept|salary|rank|
+-----+-----------+------+----+
| Lisa|      Sales| 10000|   1|
| Alex|      Sales| 30000|   2|
| Evan|      Sales| 32000|   3|
| Fred|Engineering| 21000|   1|
|  Tom|Engineering| 23000|   2|
|Chloe|Engineering| 23000|   2|
| Paul|Engineering| 29000|   4|
|Helen|  Marketing| 29000|   1|
| Jane|  Marketing| 29000|   1|
| Jeff|  Marketing| 35000|   3|
+-----+-----------+------+----+
SELECT name, dept, salary, DENSE_RANK() OVER (PARTITION BY dept ORDER BY salary ROWS
 BETWEEN
UNBOUNDED PRECEDING AND CURRENT ROW) AS dense_rank FROM employees;
+-----+-----------+------+----------+
| name|       dept|salary|dense_rank|
+-----+-----------+------+----------+
| Lisa|      Sales| 10000|         1|
| Alex|      Sales| 30000|         2|
| Evan|      Sales| 32000|         3|
| Fred|Engineering| 21000|         1|
|  Tom|Engineering| 23000|         2|
|Chloe|Engineering| 23000|         2|
| Paul|Engineering| 29000|         3|
|Helen|  Marketing| 29000|         1|
| Jane|  Marketing| 29000|         1|
| Jeff|  Marketing| 35000|         2|
+-----+-----------+------+----------+
```

```
SELECT name, dept, age, CUME_DIST() OVER (PARTITION BY dept ORDER BY age
RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS cume_dist FROM employees;
+-----+-----------+------+------------------+
| name|       dept|age   |         cume_dist|
+-----+-----------+------+------------------+
| Alex|      Sales|    33|0.3333333333333333|
| Lisa|      Sales|    35|0.6666666666666666|
| Evan|      Sales|    38|               1.0|
| Paul|Engineering|    23|              0.25|
|Chloe|Engineering|    25|              0.75|
| Fred|Engineering|    28|              0.25|
|  Tom|Engineering|    33|               1.0|
| Jane|  Marketing|    28|0.3333333333333333|
| Jeff|  Marketing|    38|0.6666666666666666|
|Helen|  Marketing|    40|               1.0|
+-----+-----------+------+------------------+
SELECT name, dept, salary, MIN(salary) OVER (PARTITION BY dept ORDER BY salary) AS min
FROM employees;
+-----+-----------+------+-----+
| name|       dept|salary|  min|
+-----+-----------+------+-----+
| Lisa|      Sales| 10000|10000|
| Alex|      Sales| 30000|10000|
| Evan|      Sales| 32000|10000|
|Helen|  Marketing| 29000|29000|
| Jane|  Marketing| 29000|29000|
| Jeff|  Marketing| 35000|29000|
| Fred|Engineering| 21000|21000|
|  Tom|Engineering| 23000|21000|
|Chloe|Engineering| 23000|21000|
| Paul|Engineering| 29000|21000|
+-----+-----------+------+-----+
SELECT name, salary,
LAG(salary) OVER (PARTITION BY dept ORDER BY salary) AS lag,
LEAD(salary, 1, 0) OVER (PARTITION BY dept ORDER BY salary) AS lead
FROM employees;
+-----+-----------+------+-----+-----+
| name|       dept|salary|  lag| lead|
+-----+-----------+------+-----+-----+
| Lisa|      Sales| 10000|NULL |30000|
| Alex|      Sales| 30000|10000|32000|
| Evan|      Sales| 32000|30000|    0|
| Fred|Engineering| 21000| NULL|23000|
|Chloe|Engineering| 23000|21000|23000|
```

```
|  Tom|Engineering| 23000|23000|29000|
| Paul|Engineering| 29000|23000|    0|
|Helen|  Marketing| 29000| NULL|29000|
| Jane|  Marketing| 29000|29000|35000|
| Jeff|  Marketing| 35000|29000|    0|
+-----+-----------+------+-----+-----+
SELECT id, v,
LEAD(v, 0) IGNORE NULLS OVER w lead,
LAG(v, 0) IGNORE NULLS OVER w lag,
NTH_VALUE(v, 2) IGNORE NULLS OVER w nth_value,
FIRST_VALUE(v) IGNORE NULLS OVER w first_value,
LAST_VALUE(v) IGNORE NULLS OVER w last_value
FROM test_ignore_null
WINDOW w AS (ORDER BY id)
ORDER BY id;
+--+----+----+----+---------+-----------+----------+
|id|   v|lead| lag|nth_value|first_value|last_value|
+--+----+----+----+---------+-----------+----------+
| 0|NULL|NULL|NULL|     NULL|       NULL|      NULL|
| 1|   x|   x|   x|     NULL|          x|         x|
| 2|NULL|NULL|NULL|     NULL|          x|         x|
| 3|NULL|NULL|NULL|     NULL|          x|         x|
| 4|   y|   y|   y|        y|          x|         y|
| 5|NULL|NULL|NULL|        y|          x|         y|
| 6|   z|   z|   z|        y|          x|         z|
| 7|   v|   v|   v|        y|          x|         v|
| 8|NULL|NULL|NULL|        y|          x|         v|
+--+----+----+----+---------+-----------+----------+
```

Fonctions de conversion

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Fonction	Description
bigint (expr)	Convertit la valeur « expr » en type de données cible « bigint ».

Fonction	Description
binaire (expr)	Convertit la valeur « expr » en type de données cible « binaire ».
booléen (expr)	Convertit la valeur « expr » en type de données cible « booléen ».
fonte (type expr AS)	Convertit la valeur « expr » en type de données cible « type ».
date d'expiration	Convertit la valeur « expr » en type de données cible « date ».
décimal (expr)	Convertit la valeur « expr » en type de données cible « decimal ».
double (expr)	Convertit la valeur « expr » en type de données cible « double ».
flotteur (expr)	Convertit la valeur « expr` en type de données cible « float`.
entier (expr)	Convertit la valeur « expr » en type de données cible « int ».
smallint (expr)	Convertit la valeur « expr » en type de données cible « smallint ».
chaîne (expr)	Convertit la valeur « expr » en type de données cible « string ».
horodatage (expr)	Convertit la valeur « expr » en type de données cible « timestamp`.
tinyint (expr)	Convertit la valeur « expr » en type de données cible « tinyint ».

Exemples

```
-- cast
SELECT cast(field as int);
+---------------+
|CAST(field AS INT)|
+---------------+
|             10|
+---------------+
```

Fonctions de prédicat

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Fonction	Description
! expr	C'est logique, non.
expr 1 < expr 2	Renvoie vrai si `expr1` est inférieur à `expr2`.
expr 1 <= expr2	Renvoie vrai si `expr1` est inférieur ou égal à `expr2`.
expr 1 <=> expr2	Renvoie le même résultat que l'opérateur EQUAL (=) pour les opérandes non nuls, mais renvoie true si les deux sont nuls, false si l'un des deux est nul.
expr 1 = expr 2	Renvoie vrai si « expr1 » est égal à « expr2 », ou faux dans le cas contraire.
expr 1 = expr 2	Renvoie vrai si « expr1 » est égal à « expr2 », ou faux dans le cas contraire.
expr1 > expr2	Renvoie vrai si `expr1` est supérieur à `expr2`.
expr1 >= expr2	Renvoie vrai si `expr1` est supérieur ou égal à `expr2`.
expr1 et expr2	Logique ET.
motif semblable à une étoile [ESCAPE escape]	Renvoie true si str fait correspondre « pattern » à « escape » sans distinction majuscules et minuscules, null si l'un des arguments est nul, false dans le cas contraire.
expr1 dans (expr2, expr3,...)	Renvoie vrai si `expr` est égal à un ValN quelconque.
Nisnan (expr)	Renvoie vrai si « expr » est NaN, ou faux dans le cas contraire.

Fonction	Description
n'est pas nul (expr)	Renvoie vrai si « expr » n'est pas nul, ou faux dans le cas contraire.
est nul (expr)	Renvoie vrai si « expr » est nul, ou faux dans le cas contraire.
motif semblable à une étoile [ESCAPE escape]	Renvoie true si str correspond à `pattern` avec `escape`, null si l'un des arguments est nul, false dans le cas contraire.
pas expiré	C'est logique, non.
expr1 ou expr2	OU logique.
regexp (str, regexp)	Renvoie vrai si `str` correspond à `regexp`, ou faux dans le cas contraire.
regexp_like (str, regexp)	Renvoie vrai si `str` correspond à `regexp`, ou faux dans le cas contraire.
rlike (étoile, expression régulière)	Renvoie vrai si `str` correspond à `regexp`, ou faux dans le cas contraire.

Exemples

```
-- !
SELECT ! true;
+----------+
|(NOT true)|
+----------+
|     false|
+----------+
SELECT ! false;
+-----------+
|(NOT false)|
+-----------+
|       true|
+-----------+
SELECT ! NULL;
+----------+
|(NOT NULL)|
+----------+
```

```
|      NULL|
+----------+
-- <
SELECT to_date('2009-07-30 04:17:52') < to_date('2009-07-30 04:17:52');
+------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) < to_date(2009-07-30 04:17:52))|
+------------------------------------------------------------+
|                                                       false|
+------------------------------------------------------------+
SELECT to_date('2009-07-30 04:17:52') < to_date('2009-08-01 04:17:52');
+------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) < to_date(2009-08-01 04:17:52))|
+------------------------------------------------------------+
|                                                        true|
+------------------------------------------------------------+
SELECT 1 < NULL;
+----------+
|(1 < NULL)|
+----------+
|      NULL|
+----------+
-- <=
SELECT 2 <= 2;
+--------+
|(2 <= 2)|
+--------+
|    true|
+--------+
SELECT 1.0 <= '1';
+----------+
|(1.0 <= 1)|
+----------+
|      true|
+----------+
SELECT to_date('2009-07-30 04:17:52') <= to_date('2009-07-30 04:17:52');
+------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) <= to_date(2009-07-30 04:17:52))|
+------------------------------------------------------------+
|                                                        true|
+------------------------------------------------------------+
SELECT to_date('2009-07-30 04:17:52') <= to_date('2009-08-01 04:17:52');
+------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) <= to_date(2009-08-01 04:17:52))|
+------------------------------------------------------------+
```

```
|                                                                  true|
+------------------------------------------------------------------+
SELECT 1 <= NULL;
+-----------+
|(1 <= NULL)|
+-----------+
|       NULL|
+-----------+
-- <=>
SELECT 2 <=> 2;
+---------+
|(2 <=> 2)|
+---------+
|     true|
+---------+
SELECT 1 <=> '1';
+---------+
|(1 <=> 1)|
+---------+
|     true|
+---------+
SELECT true <=> NULL;
+---------------+
|(true <=> NULL)|
+---------------+
|          false|
+---------------+
SELECT NULL <=> NULL;
+---------------+
|(NULL <=> NULL)|
+---------------+
|           true|
+---------------+
-- =
SELECT 2 = 2;
+-------+
|(2 = 2)|
+-------+
|   true|
+-------+
SELECT 1 = '1';
+-------+
|(1 = 1)|
+-------+
```

```
|   true|
+-------+
SELECT true = NULL;
+-------------+
|(true = NULL)|
+-------------+
|         NULL|
+-------------+
SELECT NULL = NULL;
+-------------+
|(NULL = NULL)|
+-------------+
|         NULL|
+-------------+
-- ==
SELECT 2 == 2;
+-------+
|(2 = 2)|
+-------+
|   true|
+-------+
SELECT 1 == '1';
+-------+
|(1 = 1)|
+-------+
|   true|
+-------+
SELECT true == NULL;
+-------------+
|(true = NULL)|
+-------------+
|         NULL|
+-------------+
SELECT NULL == NULL;
+-------------+
|(NULL = NULL)|
+-------------+
|         NULL|
+-------------+
-- >
SELECT 2 > 1;
+-------+
|(2 > 1)|
+-------+
```

```
|   true|
+-------+
SELECT 2 > 1.1;
+-------+
|(2 > 1)|
+-------+
|   true|
+-------+
SELECT to_date('2009-07-30 04:17:52') > to_date('2009-07-30 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) > to_date(2009-07-30 04:17:52))|
+-------------------------------------------------------------+
|                                                        false|
+-------------------------------------------------------------+
SELECT to_date('2009-07-30 04:17:52') > to_date('2009-08-01 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) > to_date(2009-08-01 04:17:52))|
+-------------------------------------------------------------+
|                                                        false|
+-------------------------------------------------------------+
SELECT 1 > NULL;
+----------+
|(1 > NULL)|
+----------+
|      NULL|
+----------+
-- >=
SELECT 2 >= 1;
+--------+
|(2 >= 1)|
+--------+
|    true|
+--------+
SELECT 2.0 >= '2.1';
+------------+
|(2.0 >= 2.1)|
+------------+
|       false|
+------------+
SELECT to_date('2009-07-30 04:17:52') >= to_date('2009-07-30 04:17:52');
+-------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) >= to_date(2009-07-30 04:17:52))|
+-------------------------------------------------------------+
|                                                        true|
```

```
+------------------------------------------------------------------+
SELECT to_date('2009-07-30 04:17:52') >= to_date('2009-08-01 04:17:52');
+------------------------------------------------------------------+
|(to_date(2009-07-30 04:17:52) >= to_date(2009-08-01 04:17:52))|
+------------------------------------------------------------------+
|                                                             false|
+------------------------------------------------------------------+
SELECT 1 >= NULL;
+-----------+
|(1 >= NULL)|
+-----------+
|       NULL|
+-----------+
-- and
SELECT true and true;
+---------------+
|(true AND true)|
+---------------+
|           true|
+---------------+
SELECT true and false;
+----------------+
|(true AND false)|
+----------------+
|           false|
+----------------+
SELECT true and NULL;
+---------------+
|(true AND NULL)|
+---------------+
|           NULL|
+---------------+
SELECT false and NULL;
+----------------+
|(false AND NULL)|
+----------------+
|           false|
+----------------+
-- ilike
SELECT ilike('Wagon', '_Agon');
+-------------------+
|ilike(Wagon, _Agon)|
+-------------------+
|               true|
```

```
+-------------------+
SELECT '%SystemDrive%\Users\John' ilike '\%SystemDrive\%\\users%';
+-----------------------------------------------------------+
|ilike(%SystemDrive%\Users\John, \%SystemDrive\%\\users%)|
+-----------------------------------------------------------+
|                                                       true|
+-----------------------------------------------------------+
SELECT '%SystemDrive%\\USERS\\John' ilike '\%SystemDrive\%\\\\Users%';
+-----------------------------------------------------------+
|ilike(%SystemDrive%\USERS\John, \%SystemDrive\%\\Users%)|
+-----------------------------------------------------------+
|                                                       true|
+-----------------------------------------------------------+
SELECT '%SystemDrive%/Users/John' ilike '/%SYSTEMDrive/%//Users%' ESCAPE '/';
+-----------------------------------------------------------+
|ilike(%SystemDrive%/Users/John, /%SYSTEMDrive/%//Users%)|
+-----------------------------------------------------------+
|                                                       true|
+-----------------------------------------------------------+
-- in
SELECT 1 in(1, 2, 3);
+----------------+
|(1 IN (1, 2, 3))|
+----------------+
|            true|
+----------------+
SELECT 1 in(2, 3, 4);
+----------------+
|(1 IN (2, 3, 4))|
+----------------+
|           false|
+----------------+
SELECT named_struct('a', 1, 'b', 2) in(named_struct('a', 1, 'b', 1), named_struct('a',
 1, 'b', 3));
+-------------------------------------------------------------------------------------+
|(named_struct(a, 1, b, 2) IN (named_struct(a, 1, b, 1), named_struct(a, 1, b, 3)))|
+-------------------------------------------------------------------------------------+
|                                                                                false|
+-------------------------------------------------------------------------------------+
SELECT named_struct('a', 1, 'b', 2) in(named_struct('a', 1, 'b', 2), named_struct('a',
 1, 'b', 3));
+-------------------------------------------------------------------------------------+
|(named_struct(a, 1, b, 2) IN (named_struct(a, 1, b, 2), named_struct(a, 1, b, 3)))|
+-------------------------------------------------------------------------------------+
```

```
|                                                                                                  true|
+------------------------------------------------------------------------------------------------------+
-- isnan
SELECT isnan(cast('NaN' as double));
+--------------------------+
|isnan(CAST(NaN AS DOUBLE))|
+--------------------------+
|                      true|
+--------------------------+
-- isnotnull
SELECT isnotnull(1);
+---------------+
|(1 IS NOT NULL)|
+---------------+
|           true|
+---------------+
-- isnull
SELECT isnull(1);
+-----------+
|(1 IS NULL)|
+-----------+
|      false|
+-----------+
-- like
SELECT like('Wagon', '_Agon');
+----------------+
|Wagon LIKE _Agon|
+----------------+
|            true|
+----------------+
-- not
SELECT not true;
+----------+
|(NOT true)|
+----------+
|     false|
+----------+
SELECT not false;
+-----------+
|(NOT false)|
+-----------+
|       true|
+-----------+
SELECT not NULL;
```

```
+----------+
|(NOT NULL)|
+----------+
|      NULL|
+----------+
-- or
SELECT true or false;
+---------------+
|(true OR false)|
+---------------+
|           true|
+---------------+
SELECT false or false;
+----------------+
|(false OR false)|
+----------------+
|           false|
+----------------+
SELECT true or NULL;
+--------------+
|(true OR NULL)|
+--------------+
|          true|
+--------------+
SELECT false or NULL;
+---------------+
|(false OR NULL)|
+---------------+
|           NULL|
+---------------+
```

Fonctions cartographiques

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Fonction	Description
element_at (tableau, index)	Renvoie l'élément du tableau à un index donné (basé sur 1).
element_at (carte, clé)	Renvoie la valeur d'une clé donnée. La fonction renvoie NULL si la clé n'est pas contenue dans la carte.
carte (clé0, valeur0, clé1, valeur1,...)	Crée une carte avec les key/value paires indiquées.
map_concat (carte,...)	Renvoie l'union de toutes les cartes données
map_contains_key (carte, clé)	Renvoie vrai si la carte contient la clé.
map_entries (carte)	Renvoie un tableau non ordonné de toutes les entrées de la carte donnée.
map_from_arrays (clés, valeurs)	Crée une carte avec une paire des key/value tableaux donnés. Tous les éléments des clés ne doivent pas être nuls
map_from_entries () arrayOfEntries	Renvoie une carte créée à partir du tableau d'entrées donné.
map_keys (carte)	Renvoie un tableau non ordonné contenant les clés de la carte.
map_values (carte)	Renvoie un tableau non ordonné contenant les valeurs de la carte.
str_to_map (texte [, PairDelim [,]]) keyValueDelim	Crée une carte après avoir divisé le texte en paires clé/valeur à l'aide de délimiteurs. Les délimiteurs par défaut sont ',' pour `PairDelim` et ':' pour ``. keyValueDelim `PairDelim` et `keyValueDelim` sont traités comme des expressions régulières.

Fonction	Description
try_element_at (tableau, index)	Renvoie l'élément du tableau à un index donné (basé sur 1). Si l'index est égal à 0, le système génère une erreur. Si l'indice est inférieur à 0, accède aux éléments du dernier au premier. La fonction renvoie toujours la valeur NULL si l'index dépasse la longueur du tableau.
try_element_at (carte, clé)	Renvoie la valeur d'une clé donnée. La fonction renvoie toujours NULL si la clé n'est pas contenue dans la carte.

Exemples

```
-- element_at
SELECT element_at(array(1, 2, 3), 2);
+----------------------------+
|element_at(array(1, 2, 3), 2)|
+----------------------------+
|                           2|
+----------------------------+
SELECT element_at(map(1, 'a', 2, 'b'), 2);
+----------------------------+
|element_at(map(1, a, 2, b), 2)|
+----------------------------+
|                           b|
+----------------------------+
-- map
SELECT map(1.0, '2', 3.0, '4');
+--------------------+
| map(1.0, 2, 3.0, 4)|
+--------------------+
|{1.0 -> 2, 3.0 -> 4}|
+--------------------+
-- map_concat
SELECT map_concat(map(1, 'a', 2, 'b'), map(3, 'c'));
+------------------------------------+
|map_concat(map(1, a, 2, b), map(3, c))|
+------------------------------------+
```

```
|                  {1 -> a, 2 -> b, ...|
+--------------------------------------+
-- map_contains_key
SELECT map_contains_key(map(1, 'a', 2, 'b'), 1);
+-----------------------------------+
|map_contains_key(map(1, a, 2, b), 1)|
+-----------------------------------+
|                               true|
+-----------------------------------+
SELECT map_contains_key(map(1, 'a', 2, 'b'), 3);
+-----------------------------------+
|map_contains_key(map(1, a, 2, b), 3)|
+-----------------------------------+
|                              false|
+-----------------------------------+
-- map_entries
SELECT map_entries(map(1, 'a', 2, 'b'));
+---------------------------+
|map_entries(map(1, a, 2, b))|
+---------------------------+
|            [{1, a}, {2, b}]|
+---------------------------+
-- map_from_arrays
SELECT map_from_arrays(array(1.0, 3.0), array('2', '4'));
+-------------------------------------------+
|map_from_arrays(array(1.0, 3.0), array(2, 4))|
+-------------------------------------------+
|                        {1.0 -> 2, 3.0 -> 4}|
+-------------------------------------------+
-- map_from_entries
SELECT map_from_entries(array(struct(1, 'a'), struct(2, 'b')));
+---------------------------------------------------+
|map_from_entries(array(struct(1, a), struct(2, b)))|
+---------------------------------------------------+
|                                {1 -> a, 2 -> b}|
+---------------------------------------------------+
-- map_keys
SELECT map_keys(map(1, 'a', 2, 'b'));
+-------------------------+
|map_keys(map(1, a, 2, b))|
+-------------------------+
|                   [1, 2]|
+-------------------------+
-- map_values
```

```
SELECT map_values(map(1, 'a', 2, 'b'));
+--------------------------+
|map_values(map(1, a, 2, b))|
+--------------------------+
|                    [a, b]|
+--------------------------+
-- str_to_map
SELECT str_to_map('a:1,b:2,c:3', ',', ':');
+----------------------------+
|str_to_map(a:1,b:2,c:3, ,, :)|
+----------------------------+
|         {a -> 1, b -> 2, ...|
+----------------------------+
SELECT str_to_map('a');
+-------------------+
|str_to_map(a, ,, :)|
+-------------------+
|        {a -> NULL}|
+-------------------+
-- try_element_at
SELECT try_element_at(array(1, 2, 3), 2);
+---------------------------------+
|try_element_at(array(1, 2, 3), 2)|
+---------------------------------+
|                                2|
+---------------------------------+
SELECT try_element_at(map(1, 'a', 2, 'b'), 2);
+---------------------------------+
|try_element_at(map(1, a, 2, b), 2)|
+---------------------------------+
|                                 b|
+---------------------------------+
```

Fonctions mathématiques

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Fonction	Description
expr 1 % expr2	Renvoie le reste après `expr1`/`expr2`.
expr 1 * expr2	Renvoie `expr1`*`expr2`.
expr 1 + expr2	Renvoie `expr1`+`expr2`.
expr1 - expr2	Renvoie `expr1`-`expr2`.
expr1/expr2	Renvoie `expr1`/`expr2`. Il effectue toujours une division en virgule flottante.
Tabs (expr)	Renvoie la valeur absolue de la valeur numérique ou de la valeur d'intervalle.
Macos (expert)	Renvoie le cosinus inverse (alias arc cosinus) de `expr`, comme s'il était calculé par `java.lang.Math.acos`.
Lacosh (expr)	Renvoie le cosinus hyperbolique inverse de « expr ».
ASIN (expr)	Renvoie le sinus inverse (alias arc sinus), l'arc sin de `expr`, comme s'il était calculé par `java.lang.Math.asin`.
asinh (expr)	Renvoie le sinus hyperbolique inverse de « expr ».
Satan (expert)	Renvoie la tangente inverse (alias arc tangente) de `expr`, comme si elle était calculée par `java.lang.Math.Atan`
atan2 (ExPry, ExPrx)	Renvoie l'angle en radians entre l'axe X positif d'un plan et le point donné par les coordonnées (`ExprX`, `ExprY`), comme s'il était calculé par `java.lang.Math.atan2`.

Fonction	Description
Katanh (expr)	Renvoie la tangente hyperbolique inverse de « expr ».
poubelle (expr)	Renvoie la représentation sous forme de chaîne de la valeur longue « expr » représent ée en binaire.
sol (expr, d)	Renvoie `expr` arrondi à `d` décimales en utilisant le mode d'arrondissement HALF_EVEN.
cbrt (expr)	Renvoie la racine cubique de `expr`.
plafond (expr [, échelle])	Renvoie le plus petit nombre après arrondiss ement qui n'est pas inférieur à « expr ». Un paramètre optionnel « scale » peut être spécifié pour contrôler le comportement d'arrondissement.
plafond (expr [, échelle])	Renvoie le plus petit nombre après arrondiss ement qui n'est pas inférieur à « expr ». Un paramètre optionnel « scale » peut être spécifié pour contrôler le comportement d'arrondissement.
conv (num, from_base, to_base)	Convertissez `num` de `from_base` en `to_base`.
coût (expr)	Renvoie le cosinus de `expr`, comme s'il était calculé par `java.lang.Math.cos`.
cosy (expr)	Renvoie le cosinus hyperbolique de `expr`, comme s'il était calculé par `java.lang.Math.Co sh`.
lit (expr)	Renvoie la cotangente de `expr`, comme si elle était calculée par `1/java.lang.Math.tan`.

Fonction	Description
csc (expr)	Renvoie la cosécante de `expr`, comme si elle était calculée par `1/java.lang.Math.sin`.
diplômes (expr)	Convertit les radians en degrés.
expr1 div expr2	Divisez « expr1 » par « expr2 ». Elle renvoie NULL si un opérande est NULL ou si « expr2 » vaut 0. Le résultat est trop long.
e ()	Renvoie le numéro d'Euler, e.
exp (expr)	Rétablit e à la puissance de « expr ».
expm1 (expr) - Renvoie exp (`expr`)	1
factoriel (expr)	Renvoie la factorielle de « expr ». `expr` est [0.. 20]. Null dans le cas contraire.
étage (expr [, échelle])	Renvoie le plus grand nombre après arrondiss ement inférieur qui n'est pas supérieur à « expr ». Un paramètre optionnel « scale » peut être spécifié pour contrôler le comportem ent d'arrondissement.
meilleur (expr,...)	Renvoie la plus grande valeur de tous les paramètres, en omettant les valeurs nulles.
hexadécimal (expr)	Convertit `expr` en hexadécimal.
hypot (expr 1, expr 2)	Renvoie sqrt (`expr1`**2 + `expr2`**2).
moins (expr,...)	Renvoie la plus petite valeur de tous les paramètres, en omettant les valeurs nulles.
ln (expr)	Renvoie le logarithme naturel (base e) de « expr ».
journal (base, expr)	Renvoie le logarithme de `expr` avec `base`.

Fonction	Description
log10 (expr)	Renvoie le logarithme de « expr » en base 10.
log1p (expr)	Renvoie log (1 + `expr`).
log2 (expr)	Renvoie le logarithme de « expr » en base 2.
expr1 mod expr2	Renvoie le reste après `expr1`/`expr2`.
négatif (expr)	Renvoie la valeur négative de « expr ».
épi ()	Renvoie pi.
pmod (expr1, expr 2)	Renvoie la valeur positive de `expr1` mod `expr2`.
positif (expr)	Renvoie la valeur de « expr ».
pow (expr1, expr 2)	Augmente `expr1` à la puissance de `expr2`.
puissance (expr1, expr2)	Augmente `expr1` à la puissance de `expr2`.
radians (expr)	Convertit les degrés en radians.
rand ([graine])	Renvoie une valeur aléatoire avec des valeurs indépendantes et distribuées de manière identique (i.i.d.) uniformément dans [0, 1).
randn ([graine])	Renvoie une valeur aléatoire avec des valeurs indépendantes et distribuées de manière identique (i.i.d.) tirées de la distribution normale standard.
aléatoire ([graine])	Renvoie une valeur aléatoire avec des valeurs indépendantes et distribuées de manière identique (i.i.d.) uniformément dans [0, 1).

Fonction	Description
Imprimer (expr)	Renvoie la valeur double dont la valeur est la plus proche de l'argument et qui est égale à un entier mathématique.
rond (expr, d)	Renvoie `expr` arrondi à `d` décimales en utilisant le mode d'arrondissement HALF_UP.
seconde (expr)	Renvoie le sécant de `expr`, comme s'il était calculé par `1/java.lang.Math.cos`.
shiftleft (base, expr)	Décalage bit par bit vers la gauche.
signe (expr)	Renvoie -1.0, 0.0 ou 1.0 car « expr » est négatif, 0 ou positif.
signature (expr)	Renvoie -1.0, 0.0 ou 1.0 car « expr » est négatif, 0 ou positif.
péché (expr)	Renvoie le sinus de `expr`, comme s'il était calculé par `java.lang.Math.sin`.
sinh (expr)	Renvoie le sinus hyperbolique de `expr`, comme s'il était calculé par `java.lang.Math.SinH`.
carré (expr)	Renvoie la racine carrée de « expr ».
bronzage (expr)	Renvoie la tangente de `expr`, comme si elle était calculée par `java.lang.Math.tan`.
tanh (expr)	Renvoie la tangente hyperbolique de `expr`, comme si elle était calculée par `java.lang.Math.TANH`.

Fonction	Description
try_add (expr 1, expr 2)	Renvoie la somme de « expr 1 » et « expr2 » et le résultat est nul en cas de débordeme nt. Les types d'entrée acceptables sont les mêmes avec l'opérateur « + ».
try_divide (dividende, diviseur)	Renvoie « dividende » ou « diviseur ». Il effectue toujours une division en virgule flottante. Son résultat est toujours nul si « expr2 » vaut 0. le « dividende » doit être un chiffre ou un intervalle. le « diviseur » doit être un chiffre.
try_multiply (expr 1, expr 2)	Renvoie `expr1`*`expr2` et le résultat est nul en cas de débordement. Les types d'entrée acceptables sont les mêmes avec l'opérateur `*`.
try_subtract (expr1, expr 2)	Renvoie `expr1`-`expr2` et le résultat est nul en cas de débordement. Les types d'entrée acceptables sont les mêmes avec l'opérateur « - ».
unhex (expr)	Convertit l'hexadécimal `expr` en binaire.
width_bucket (valeur, valeur minimale, valeur maximale, num_bucket)	Renvoie le numéro de compartiment auquel « value » serait attribuée dans un histogram me d'équilargeur avec des compartiments « num_bucket », compris entre « min_value » et « max_value ». »

Exemples

```
-- %
SELECT 2 % 1.8;
+---------+
|(2 % 1.8)|
```

```
+---------+
|      0.2|
+---------+
SELECT MOD(2, 1.8);
+-----------+
|mod(2, 1.8)|
+-----------+
|        0.2|
+-----------+
-- *
SELECT 2 * 3;
+-------+
|(2 * 3)|
+-------+
|      6|
+-------+
-- +
SELECT 1 + 2;
+-------+
|(1 + 2)|
+-------+
|      3|
+-------+
-- -
SELECT 2 - 1;
+-------+
|(2 - 1)|
+-------+
|      1|
+-------+
-- /
SELECT 3 / 2;
+-------+
|(3 / 2)|
+-------+
|    1.5|
+-------+
SELECT 2L / 2L;
+-------+
|(2 / 2)|
+-------+
|    1.0|
+-------+
-- abs
```

```
SELECT abs(-1);
+-------+
|abs(-1)|
+-------+
|      1|
+-------+
SELECT abs(INTERVAL -'1-1' YEAR TO MONTH);
+----------------------------------+
|abs(INTERVAL '-1-1' YEAR TO MONTH)|
+----------------------------------+
|              INTERVAL '1-1' YE...|
+----------------------------------+
-- acos
SELECT acos(1);
+-------+
|ACOS(1)|
+-------+
|    0.0|
+-------+
SELECT acos(2);
+-------+
|ACOS(2)|
+-------+
|    NaN|
+-------+
-- acosh
SELECT acosh(1);
+--------+
|ACOSH(1)|
+--------+
|     0.0|
+--------+
SELECT acosh(0);
+--------+
|ACOSH(0)|
+--------+
|     NaN|
+--------+
-- asin
SELECT asin(0);
+-------+
|ASIN(0)|
+-------+
|    0.0|
```

```
+-------+
SELECT asin(2);
+-------+
|ASIN(2)|
+-------+
|    NaN|
+-------+
-- asinh
SELECT asinh(0);
+--------+
|ASINH(0)|
+--------+
|     0.0|
+--------+
-- atan
SELECT atan(0);
+-------+
|ATAN(0)|
+-------+
|    0.0|
+-------+
-- atan2
SELECT atan2(0, 0);
+-----------+
|ATAN2(0, 0)|
+-----------+
|        0.0|
+-----------+
-- atanh
SELECT atanh(0);
+--------+
|ATANH(0)|
+--------+
|     0.0|
+--------+
SELECT atanh(2);
+--------+
|ATANH(2)|
+--------+
|     NaN|
+--------+
-- bin
SELECT bin(13);
+-------+
```

```
|bin(13)|
+-------+
|   1101|
+-------+
SELECT bin(-13);
+--------------------+
|            bin(-13)|
+--------------------+
|1111111111111111...|
+--------------------+
SELECT bin(13.3);
+---------+
|bin(13.3)|
+---------+
|     1101|
+---------+
-- bround
SELECT bround(2.5, 0);
+--------------+
|bround(2.5, 0)|
+--------------+
|             2|
+--------------+
SELECT bround(25, -1);
+--------------+
|bround(25, -1)|
+--------------+
|            20|
+--------------+
-- cbrt
SELECT cbrt(27.0);
+----------+
|CBRT(27.0)|
+----------+
|       3.0|
+----------+
-- ceil
SELECT ceil(-0.1);
+----------+
|CEIL(-0.1)|
+----------+
|         0|
+----------+
SELECT ceil(5);
```

```
+-------+
|CEIL(5)|
+-------+
|      5|
+-------+
SELECT ceil(3.1411, 3);
+---------------+
|ceil(3.1411, 3)|
+---------------+
|          3.142|
+---------------+
SELECT ceil(3.1411, -3);
+----------------+
|ceil(3.1411, -3)|
+----------------+
|            1000|
+----------------+
-- ceiling
SELECT ceiling(-0.1);
+-------------+
|ceiling(-0.1)|
+-------------+
|            0|
+-------------+
SELECT ceiling(5);
+----------+
|ceiling(5)|
+----------+
|         5|
+----------+
SELECT ceiling(3.1411, 3);
+------------------+
|ceiling(3.1411, 3)|
+------------------+
|             3.142|
+------------------+
SELECT ceiling(3.1411, -3);
+-------------------+
|ceiling(3.1411, -3)|
+-------------------+
|               1000|
+-------------------+
-- conv
SELECT conv('100', 2, 10);
```

```
+----------------+
|conv(100, 2, 10)|
+----------------+
|               4|
+----------------+
SELECT conv(-10, 16, -10);
+------------------+
|conv(-10, 16, -10)|
+------------------+
|               -16|
+------------------+
-- cos
SELECT cos(0);
+------+
|COS(0)|
+------+
|   1.0|
+------+
-- cosh
SELECT cosh(0);
+-------+
|COSH(0)|
+-------+
|    1.0|
+-------+
-- cot
SELECT cot(1);
+------------------+
|            COT(1)|
+------------------+
|0.6420926159343306|
+------------------+
-- csc
SELECT csc(1);
+------------------+
|            CSC(1)|
+------------------+
|1.1883951057781212|
+------------------+
-- degrees
SELECT degrees(3.141592653589793);
+--------------------------+
|DEGREES(3.141592653589793)|
+--------------------------+
```

```
|                     180.0|
+--------------------------+
-- div
SELECT 3 div 2;
+---------+
|(3 div 2)|
+---------+
|        1|
+---------+
SELECT INTERVAL '1-1' YEAR TO MONTH div INTERVAL '-1' MONTH;
+-----------------------------------------------------------+
|(INTERVAL '1-1' YEAR TO MONTH div INTERVAL '-1' MONTH)|
+-----------------------------------------------------------+
|                                                        -13|
+-----------------------------------------------------------+
-- e
SELECT e();
+-----------------+
|              E()|
+-----------------+
|2.718281828459045|
+-----------------+
-- exp
SELECT exp(0);
+------+
|EXP(0)|
+------+
|   1.0|
+------+
-- expm1
SELECT expm1(0);
+--------+
|EXPM1(0)|
+--------+
|     0.0|
+--------+
-- factorial
SELECT factorial(5);
+------------+
|factorial(5)|
+------------+
|         120|
+------------+
-- floor
```

```
SELECT floor(-0.1);
+-----------+
|FLOOR(-0.1)|
+-----------+
|         -1|
+-----------+
SELECT floor(5);
+--------+
|FLOOR(5)|
+--------+
|       5|
+--------+
SELECT floor(3.1411, 3);
+----------------+
|floor(3.1411, 3)|
+----------------+
|           3.141|
+----------------+
SELECT floor(3.1411, -3);
+-----------------+
|floor(3.1411, -3)|
+-----------------+
|                0|
+-----------------+
-- greatest
SELECT greatest(10, 9, 2, 4, 3);
+------------------------+
|greatest(10, 9, 2, 4, 3)|
+------------------------+
|                      10|
+------------------------+
-- hex
SELECT hex(17);
+-------+
|hex(17)|
+-------+
|     11|
+-------+
SELECT hex('SQL');
+------------------+
|    hex(SQL)|
+------------------+
|53514C|
+------------------+
```

```
-- hypot
SELECT hypot(3, 4);
+-----------+
|HYPOT(3, 4)|
+-----------+
|        5.0|
+-----------+
-- least
SELECT least(10, 9, 2, 4, 3);
+---------------------+
|least(10, 9, 2, 4, 3)|
+---------------------+
|                    2|
+---------------------+
-- ln
SELECT ln(1);
+-----+
|ln(1)|
+-----+
|  0.0|
+-----+
-- log
SELECT log(10, 100);
+------------+
|LOG(10, 100)|
+------------+
|         2.0|
+------------+
-- log10
SELECT log10(10);
+---------+
|LOG10(10)|
+---------+
|      1.0|
+---------+
-- log1p
SELECT log1p(0);
+--------+
|LOG1P(0)|
+--------+
|     0.0|
+--------+
-- log2
SELECT log2(2);
```

```
+-------+
|LOG2(2)|
+-------+
|    1.0|
+-------+
-- mod
SELECT 2 % 1.8;
+---------+
|(2 % 1.8)|
+---------+
|      0.2|
+---------+
SELECT MOD(2, 1.8);
+-----------+
|mod(2, 1.8)|
+-----------+
|        0.2|
+-----------+
-- negative
SELECT negative(1);
+-----------+
|negative(1)|
+-----------+
|         -1|
+-----------+
-- pi
SELECT pi();
+-----------------+
|             PI()|
+-----------------+
|3.141592653589793|
+-----------------+
-- pmod
SELECT pmod(10, 3);
+-----------+
|pmod(10, 3)|
+-----------+
|          1|
+-----------+
SELECT pmod(-10, 3);
+------------+
|pmod(-10, 3)|
+------------+
|           2|
```

```
+------------+
-- positive
SELECT positive(1);
+-----+
|(+ 1)|
+-----+
|    1|
+-----+
-- pow
SELECT pow(2, 3);
+---------+
|pow(2, 3)|
+---------+
|      8.0|
+---------+
-- power
SELECT power(2, 3);
+-----------+
|POWER(2, 3)|
+-----------+
|        8.0|
+-----------+
-- radians
SELECT radians(180);
+-----------------+
|     RADIANS(180)|
+-----------------+
|3.141592653589793|
+-----------------+
-- rand
SELECT rand();
+------------------+
|            rand()|
+------------------+
|0.7211420708112387|
+------------------+
SELECT rand(0);
+------------------+
|           rand(0)|
+------------------+
|0.7604953758285915|
+------------------+
SELECT rand(null);
+------------------+
```

```
|        rand(NULL)|
+------------------+
|0.7604953758285915|
+------------------+
-- randn
SELECT randn();
+-------------------+
|             randn()|
+-------------------+
|-0.8175603217732732|
+-------------------+
SELECT randn(0);
+------------------+
|          randn(0)|
+------------------+
|1.6034991609278433|
+------------------+
SELECT randn(null);
+------------------+
|       randn(NULL)|
+------------------+
|1.6034991609278433|
+------------------+
-- random
SELECT random();
+-----------------+
|           rand()|
+-----------------+
|0.394205008255365|
+-----------------+
SELECT random(0);
+------------------+
|           rand(0)|
+------------------+
|0.7604953758285915|
+------------------+
SELECT random(null);
+------------------+
|        rand(NULL)|
+------------------+
|0.7604953758285915|
+------------------+
-- rint
SELECT rint(12.3456);
```

```
+-------------+
|rint(12.3456)|
+-------------+
|         12.0|
+-------------+
-- round
SELECT round(2.5, 0);
+-------------+
|round(2.5, 0)|
+-------------+
|            3|
+-------------+
-- sec
SELECT sec(0);
+------+
|SEC(0)|
+------+
|   1.0|
+------+
-- shiftleft
SELECT shiftleft(2, 1);
+---------------+
|shiftleft(2, 1)|
+---------------+
|              4|
+---------------+
-- sign
SELECT sign(40);
+--------+
|sign(40)|
+--------+
|     1.0|
+--------+
SELECT sign(INTERVAL -'100' YEAR);
+--------------------------+
|sign(INTERVAL '-100' YEAR)|
+--------------------------+
|                      -1.0|
+--------------------------+
-- signum
SELECT signum(40);
+----------+
|SIGNUM(40)|
+----------+
```

```
|       1.0|
+----------+
SELECT signum(INTERVAL -'100' YEAR);
+----------------------------+
|SIGNUM(INTERVAL '-100' YEAR)|
+----------------------------+
|                        -1.0|
+----------------------------+
-- sin
SELECT sin(0);
+------+
|SIN(0)|
+------+
|   0.0|
+------+
-- sinh
SELECT sinh(0);
+-------+
|SINH(0)|
+-------+
|    0.0|
+-------+
-- sqrt
SELECT sqrt(4);
+-------+
|SQRT(4)|
+-------+
|    2.0|
+-------+
-- tan
SELECT tan(0);
+------+
|TAN(0)|
+------+
|   0.0|
+------+
-- tanh
SELECT tanh(0);
+-------+
|TANH(0)|
+-------+
|    0.0|
+-------+
-- try_add
```

```
SELECT try_add(1, 2);
+-------------+
|try_add(1, 2)|
+-------------+
|            3|
+-------------+
SELECT try_add(2147483647, 1);
+----------------------+
|try_add(2147483647, 1)|
+----------------------+
|                  NULL|
+----------------------+
SELECT try_add(date'2021-01-01', 1);
+-----------------------------+
|try_add(DATE '2021-01-01', 1)|
+-----------------------------+
|                   2021-01-02|
+-----------------------------+
SELECT try_add(date'2021-01-01', interval 1 year);
+-------------------------------------------+
|try_add(DATE '2021-01-01', INTERVAL '1' YEAR)|
+-------------------------------------------+
|                                 2022-01-01|
+-------------------------------------------+
SELECT try_add(timestamp'2021-01-01 00:00:00', interval 1 day);
+-----------------------------------------------------------+
|try_add(TIMESTAMP '2021-01-01 00:00:00', INTERVAL '1' DAY)|
+-----------------------------------------------------------+
|                                    2021-01-02 00:00:00|
+-----------------------------------------------------------+
SELECT try_add(interval 1 year, interval 2 year);
+-----------------------------------------+
|try_add(INTERVAL '1' YEAR, INTERVAL '2' YEAR)|
+-----------------------------------------+
|                           INTERVAL '3' YEAR|
+-----------------------------------------+
-- try_divide
SELECT try_divide(3, 2);
+---------------+
|try_divide(3, 2)|
+---------------+
|             1.5|
+---------------+
SELECT try_divide(2L, 2L);
```

```
+----------------+
|try_divide(2, 2)|
+----------------+
|             1.0|
+----------------+
SELECT try_divide(1, 0);
+----------------+
|try_divide(1, 0)|
+----------------+
|            NULL|
+----------------+
SELECT try_divide(interval 2 month, 2);
+--------------------------------+
|try_divide(INTERVAL '2' MONTH, 2)|
+--------------------------------+
|             INTERVAL '0-1' YE...|
+--------------------------------+
SELECT try_divide(interval 2 month, 0);
+--------------------------------+
|try_divide(INTERVAL '2' MONTH, 0)|
+--------------------------------+
|                            NULL|
+--------------------------------+
-- try_multiply
SELECT try_multiply(2, 3);
+------------------+
|try_multiply(2, 3)|
+------------------+
|                 6|
+------------------+
SELECT try_multiply(-2147483648, 10);
+----------------------------+
|try_multiply(-2147483648, 10)|
+----------------------------+
|                        NULL|
+----------------------------+
SELECT try_multiply(interval 2 year, 3);
+---------------------------------+
|try_multiply(INTERVAL '2' YEAR, 3)|
+---------------------------------+
|              INTERVAL '6-0' YE...|
+---------------------------------+
-- try_subtract
SELECT try_subtract(2, 1);
```

```
+------------------+
|try_subtract(2, 1)|
+------------------+
|                 1|
+------------------+
SELECT try_subtract(-2147483648, 1);
+----------------------------+
|try_subtract(-2147483648, 1)|
+----------------------------+
|                        NULL|
+----------------------------+
SELECT try_subtract(date'2021-01-02', 1);
+---------------------------------+
|try_subtract(DATE '2021-01-02', 1)|
+---------------------------------+
|                       2021-01-01|
+---------------------------------+
SELECT try_subtract(date'2021-01-01', interval 1 year);
+-------------------------------------------------+
|try_subtract(DATE '2021-01-01', INTERVAL '1' YEAR)|
+-------------------------------------------------+
|                                       2020-01-01|
+-------------------------------------------------+
SELECT try_subtract(timestamp'2021-01-02 00:00:00', interval 1 day);
+-----------------------------------------------------------------+
|try_subtract(TIMESTAMP '2021-01-02 00:00:00', INTERVAL '1' DAY)|
+-----------------------------------------------------------------+
|                                              2021-01-01 00:00:00|
+-----------------------------------------------------------------+
SELECT try_subtract(interval 2 year, interval 1 year);
+-------------------------------------------------+
|try_subtract(INTERVAL '2' YEAR, INTERVAL '1' YEAR)|
+-------------------------------------------------+
|                                 INTERVAL '1' YEAR|
+-------------------------------------------------+
-- unhex
SELECT decode(unhex('53514C'), 'UTF-8');
+---------------------------------------+
|decode(unhex(53514C), UTF-8)|
+---------------------------------------+
|                                  SQL|
+---------------------------------------+
-- width_bucket
SELECT width_bucket(5.3, 0.2, 10.6, 5);
```

```
+------------------------------+
|width_bucket(5.3, 0.2, 10.6, 5)|
+------------------------------+
|                             3|
+------------------------------+
SELECT width_bucket(-2.1, 1.3, 3.4, 3);
+------------------------------+
|width_bucket(-2.1, 1.3, 3.4, 3)|
+------------------------------+
|                             0|
+------------------------------+
SELECT width_bucket(8.1, 0.0, 5.7, 4);
+------------------------------+
|width_bucket(8.1, 0.0, 5.7, 4)|
+------------------------------+
|                            5|
+------------------------------+
SELECT width_bucket(-0.9, 5.2, 0.5, 2);
+------------------------------+
|width_bucket(-0.9, 5.2, 0.5, 2)|
+------------------------------+
|                             3|
+------------------------------+
SELECT width_bucket(INTERVAL '0' YEAR, INTERVAL '0' YEAR, INTERVAL '10' YEAR, 10);
+-----------------------------------------------------------------------------+
|width_bucket(INTERVAL '0' YEAR, INTERVAL '0' YEAR, INTERVAL '10' YEAR, 10)|
+-----------------------------------------------------------------------------+
|                                                                            1|
+-----------------------------------------------------------------------------+
SELECT width_bucket(INTERVAL '1' YEAR, INTERVAL '0' YEAR, INTERVAL '10' YEAR, 10);
+-----------------------------------------------------------------------------+
|width_bucket(INTERVAL '1' YEAR, INTERVAL '0' YEAR, INTERVAL '10' YEAR, 10)|
+-----------------------------------------------------------------------------+
|                                                                            2|
+-----------------------------------------------------------------------------+
SELECT width_bucket(INTERVAL '0' DAY, INTERVAL '0' DAY, INTERVAL '10' DAY, 10);
+--------------------------------------------------------------------+
|width_bucket(INTERVAL '0' DAY, INTERVAL '0' DAY, INTERVAL '10' DAY, 10)|
+--------------------------------------------------------------------+
|                                                                   1|
+--------------------------------------------------------------------+
SELECT width_bucket(INTERVAL '1' DAY, INTERVAL '0' DAY, INTERVAL '10' DAY, 10);
+--------------------------------------------------------------------+
|width_bucket(INTERVAL '1' DAY, INTERVAL '0' DAY, INTERVAL '10' DAY, 10)|
```

```
+--------------------------------------------------------------------+
|                                                                   2|
+--------------------------------------------------------------------+
```

Fonctions du générateur

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge ces fonctions SQL, consultezthe section called “Commandes SQL prises en charge”.

Fonction	Description
exploser (expr)	Sépare les éléments du tableau « expr » en plusieurs lignes, ou les éléments de la carte « expr » en plusieurs lignes et colonnes. Sauf indication contraire, utilise le nom de colonne par défaut « col » pour les éléments du tableau ou « key » et « value » pour les éléments de la carte.
explode_outer (expr)	Sépare les éléments du tableau « expr » en plusieurs lignes, ou les éléments de la carte « expr » en plusieurs lignes et colonnes. Sauf indication contraire, utilise le nom de colonne par défaut « col » pour les éléments du tableau ou « key » et « value » pour les éléments de la carte.
en ligne (expr)	Fait exploser un tableau de structures. Utilise les noms de colonne col1, col2, etc. par défaut, sauf indication contraire.
inline_router (expr)	Fait exploser un tableau de structures. Utilise les noms de colonne col1, col2, etc. par défaut, sauf indication contraire.
posexplode (expr)	Sépare les éléments du tableau « expr » en plusieurs lignes avec des positions, ou les éléments de la carte « expr » en plusieurs lignes et colonnes avec des positions. Sauf indication contraire, utilise le nom de colonne `pos` pour la position, `col` pour les éléments du tableau ou `key` et `value` pour les éléments de la carte.

Fonction	Description
posexplode_outer (expr)	Sépare les éléments du tableau « expr » en plusieurs lignes avec des positions, ou les éléments de la carte « expr » en plusieurs lignes et colonnes avec des positions. Sauf indication contraire, utilise le nom de colonne `pos` pour la position, `col` pour les éléments du tableau ou `key` et `value` pour les éléments de la carte.
pile (n, expr1,..., exprk)	Sépare `expr1`,..., `exprok` en `n` lignes. Utilise les noms de colonne col0, col1, etc. par défaut, sauf indication contraire.

Exemples

```
-- explode
SELECT explode(array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

SELECT explode(collection => array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

SELECT * FROM explode(collection => array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

-- explode_outer
SELECT explode_outer(array(10, 20));
+---+
```

```
|col|
+---+
| 10|
| 20|
+---+

SELECT explode_outer(collection => array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

SELECT * FROM explode_outer(collection => array(10, 20));
+---+
|col|
+---+
| 10|
| 20|
+---+

-- inline
SELECT inline(array(struct(1, 'a'), struct(2, 'b')));
+----+----+
|col1|col2|
+----+----+
|   1|   a|
|   2|   b|
+----+----+

-- inline_outer
SELECT inline_outer(array(struct(1, 'a'), struct(2, 'b')));
+----+----+
|col1|col2|
+----+----+
|   1|   a|
|   2|   b|
+----+----+

-- posexplode
SELECT posexplode(array(10,20));
+---+---+
|pos|col|
```

```
+---+---+
|  0| 10|
|  1| 20|
+---+---+

SELECT * FROM posexplode(array(10,20));
+---+---+
|pos|col|
+---+---+
|  0| 10|
|  1| 20|
+---+---+

-- posexplode_outer
SELECT posexplode_outer(array(10,20));
+---+---+
|pos|col|
+---+---+
|  0| 10|
|  1| 20|
+---+---+

SELECT * FROM posexplode_outer(array(10,20));
+---+---+
|pos|col|
+---+---+
|  0| 10|
|  1| 20|
+---+---+

-- stack
SELECT stack(2, 1, 2, 3);
+----+----+
|col0|col1|
+----+----+
|   1|   2|
|   3|NULL|
+----+----+
```

Clause SELECT

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

OpenSearch SQL prend en charge une `SELECT` instruction utilisée pour récupérer des ensembles de résultats à partir d'une ou de plusieurs tables. La section suivante décrit la syntaxe globale des requêtes et les différentes constructions d'une requête.

Syntaxe

```
select_statement
[ { UNION | INTERSECT | EXCEPT } [ ALL | DISTINCT ] select_statement, ... ]
[ ORDER BY
    { expression [ ASC | DESC ] [ NULLS { FIRST | LAST } ]
    [ , ... ]
    }
]
[ SORT BY
    { expression [ ASC | DESC ] [ NULLS { FIRST | LAST } ]
    [ , ... ]
    }
]
[ WINDOW { named_window [ , WINDOW named_window, ... ] } ]
[ LIMIT { ALL | expression } ]
```

While `select_statement` est défini comme suit :

```
SELECT [ ALL | DISTINCT ] { [ [ named_expression ] [ , ... ] ] }
FROM { from_item [ , ... ] }
[ PIVOT clause ]
[ UNPIVOT clause ]
[ LATERAL VIEW clause ] [ ... ]
[ WHERE boolean_expression ]
[ GROUP BY expression [ , ... ] ]
[ HAVING boolean_expression ]
```

Paramètres

- TOUS

  Sélectionne toutes les lignes correspondantes dans la relation et est activé par défaut.

- DISTINCT

  Sélectionne toutes les lignes correspondantes dans la relation après avoir supprimé les doublons dans les résultats.

- expression_nommée

  Expression à laquelle un nom a été attribué. En général, il désigne une expression de colonne.

  Syntaxe : `expression [[AS] alias]`

- from_item

  Relation entre les tables

  Relation d'adhésion

  Relation pivot

  Relation dépivotante

  Fonction Table-value

  Tableau en ligne

  `[ LATERAL ] ( Subquery )`

- PIVOT

  La `PIVOT` clause est utilisée pour la perspective des données. Vous pouvez obtenir les valeurs agrégées en fonction d'une valeur de colonne spécifique.

- UNPIVOT

  La `UNPIVOT` clause transforme les colonnes en lignes. C'est l'inverse de`PIVOT`, sauf pour l'agrégation des valeurs.

- VUE LATÉRALE

  La `LATERAL VIEW` clause est utilisée conjointement avec des fonctions de génération telles que`EXPLODE`, qui généreront une table virtuelle contenant une ou plusieurs lignes.

`LATERAL VIEW`appliquera les lignes à chaque ligne de sortie d'origine.

- OÙ

  Filtre le résultat de la `FROM` clause en fonction des prédicats fournis.

- GROUPER PAR

  Spécifie les expressions utilisées pour regrouper les lignes.

  Ceci est utilisé conjointement avec des fonctions d'agrégation (`MIN`,`MAX`,`COUNT`,`SUM`,`AVG`,, etc.) pour regrouper les lignes en fonction des expressions de regroupement et des valeurs agrégées de chaque groupe.

  Lorsqu'une `FILTER` clause est attachée à une fonction d'agrégation, seules les lignes correspondantes sont transmises à cette fonction.

- AYANT

  Spécifie les prédicats selon lesquels les lignes produites par `GROUP BY` sont filtrées.

  La `HAVING` clause est utilisée pour filtrer les lignes une fois le regroupement effectué.

  Si elle `HAVING` est spécifiée sans`GROUP BY`, elle indique une expression `GROUP BY` sans regroupement (agrégat global).

- COMMANDEZ PAR

  Spécifie l'ordre des lignes du jeu de résultats complet de la requête.

  Les lignes de sortie sont ordonnées sur les partitions.

  Ce paramètre s'exclut mutuellement `SORT BY` `DISTRIBUTE BY` et ne peut pas être spécifié ensemble.

- TRIER PAR

  Spécifie l'ordre selon lequel les lignes sont ordonnées au sein de chaque partition.

  Ce paramètre s'exclut mutuellement `ORDER BY` et ne peut pas être spécifié ensemble.

- LIMITE

  Spécifie le nombre maximum de lignes pouvant être renvoyées par une instruction ou une sous-requête.

Cette clause est principalement utilisée conjointement avec `ORDER BY` pour produire un résultat déterministe.

- expression_booléenne

  Spécifie toute expression dont le type de résultat est un booléen.

  Deux expressions ou plus peuvent être combinées à l'aide des opérateurs logiques (`AND`,`OR`).

- expression

  Spécifie une combinaison d'une ou de plusieurs valeurs, opérateurs et fonctions SQL qui permet d'évaluer une valeur.

- fenêtre_nommée

  Spécifie des alias pour une ou plusieurs spécifications de fenêtre source.

  Les spécifications de la fenêtre source peuvent être référencées dans les définitions de fenêtre de la requête.

Clause WHERE

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

La `WHERE` clause est utilisée pour limiter les résultats de la `FROM` clause d'une requête ou d'une sous-requête en fonction de la condition spécifiée.

Syntaxe

```
WHERE boolean_expression
```

Paramètres

- expression_booléenne

  Spécifie toute expression dont le type de résultat est un booléen.

Deux expressions ou plus peuvent être combinées à l'aide des opérateurs logiques (AND,OR).

Exemples

```
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
(100, 'John', 30),
(200, 'Mary', NULL),
(300, 'Mike', 80),
(400, 'Dan',  50);

-- Comparison operator in `WHERE` clause.
SELECT * FROM person WHERE id > 200 ORDER BY id;
+---+----+---+
| id|name|age|
+---+----+---+
|300|Mike| 80|
|400| Dan| 50|
+---+----+---+

-- Comparison and logical operators in `WHERE` clause.
SELECT * FROM person WHERE id = 200 OR id = 300 ORDER BY id;
+---+----+----+
| id|name| age|
+---+----+----+
|200|Mary|null|
|300|Mike|  80|
+---+----+----+

-- IS NULL expression in `WHERE` clause.
SELECT * FROM person WHERE id > 300 OR age IS NULL ORDER BY id;
+---+----+----+
| id|name| age|
+---+----+----+
|200|Mary|null|
|400| Dan|  50|
+---+----+----+

-- Function expression in `WHERE` clause.
SELECT * FROM person WHERE length(name) > 3 ORDER BY id;
+---+----+----+
| id|name| age|
```

```
+---+----+----+
|100|John|  30|
|200|Mary|null|
|300|Mike|  80|
+---+----+----+

-- `BETWEEN` expression in `WHERE` clause.
SELECT * FROM person WHERE id BETWEEN 200 AND 300 ORDER BY id;
+---+----+----+
| id|name| age|
+---+----+----+
|200|Mary|null|
|300|Mike|  80|
+---+----+----+

-- Scalar Subquery in `WHERE` clause.
SELECT * FROM person WHERE age > (SELECT avg(age) FROM person);
+---+----+---+
| id|name|age|
+---+----+---+
|300|Mike| 80|
+---+----+---+

-- Correlated Subquery in `WHERE` clause.
SELECT id FROM person
WHERE exists (SELECT id FROM person where id = 200);
+---+----+----+
|id |name|age |
+---+----+----+
|200|Mary|null|
+---+----+----+
```

Clause GROUP BY

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

La GROUP BY clause est utilisée pour regrouper les lignes en fonction d'un ensemble d'expressions de regroupement spécifiées et pour calculer des agrégations sur le groupe de lignes en fonction d'une ou de plusieurs fonctions d'agrégation spécifiées.

Le système effectue également plusieurs agrégations pour le même ensemble d'enregistrements d'entrée via des ROLLUP clauses GROUPING SETSCUBE,. Les expressions de regroupement et les agrégations avancées peuvent être mélangées dans la GROUP BY clause et imbriquées dans une GROUPING SETS clause. Voir plus de détails dans la Mixed/Nested Grouping Analytics section.

Lorsqu'une FILTER clause est attachée à une fonction d'agrégation, seules les lignes correspondantes sont transmises à cette fonction.

Syntaxe

```
GROUP BY group_expression [ , group_expression [ , ... ] ] [ WITH { ROLLUP | CUBE } ]
GROUP BY { group_expression | { ROLLUP | CUBE | GROUPING SETS } (grouping_set
 [ , ...]) } [ , ... ]
```

Alors que les fonctions d'agrégation sont définies comme suit :

```
aggregate_name ( [ DISTINCT ] expression [ , ... ] ) [ FILTER ( WHERE
 boolean_expression ) ]
```

Paramètres

- expression_groupe

  Spécifie les critères selon lesquels les lignes sont regroupées. Le regroupement des lignes est effectué en fonction des valeurs de résultat des expressions de regroupement.

  Une expression de regroupement peut être un nom de colonneGROUP BY a, une position de colonne ou une expression similaireGROUP BY a + b. GROUP BY 0
- grouping_set

  Un ensemble de regroupement est spécifié par zéro ou plusieurs expressions séparées par des virgules entre parenthèses. Lorsque l'ensemble de regroupement ne comporte qu'un seul élément, les parenthèses peuvent être omises.

  Par exemple, GROUPING SETS ((a), (b)) est identique à GROUPING SETS (a, b).

Syntaxe : `{ ( [ expression [ , ... ] ] ) | expression }`

- ENSEMBLES DE REGROUPEMENT

  Regroupe les lignes pour chaque ensemble de regroupement spécifié par la suite`GROUPING SETS`.

  Par exemple, `GROUP BY GROUPING SETS ((warehouse), (product))` est sémantiquement équivalent à l'union des résultats de `GROUP BY warehouse` et. `GROUP BY product` Cette clause est un raccourci pour un UNION ALL dans lequel chaque étape de l'`UNION ALL`opérateur effectue l'agrégation de chaque ensemble de groupes spécifié dans la `GROUPING SETS` clause.

  De même, `GROUP BY GROUPING SETS ((warehouse, product), (product), ())` est sémantiquement équivalent à l'union des résultats d'`GROUP BY warehouse, product, GROUP BY product`un agrégat global.

- ROLLUP

  Spécifie plusieurs niveaux d'agrégation dans une seule instruction. Cette clause est utilisée pour calculer des agrégations basées sur plusieurs ensembles de regroupement. `ROLLUP`est un raccourci pour. `GROUPING SETS`

  Par exemple, `GROUP BY warehouse, product WITH ROLLUP or GROUP BY ROLLUP(warehouse, product)` équivaut à `GROUP BY GROUPING SETS((warehouse, product), (warehouse), ())`.

  `GROUP BY ROLLUP(warehouse, product, (warehouse, location))` est équivalent à `GROUP BY GROUPING SETS((warehouse, product, location), (warehouse, product), (warehouse), ())`.

  Les N éléments d'une spécification ROLLUP se traduisent par N+1 ENSEMBLES DE REGROUPEMENT.

- CUBE

  La clause CUBE est utilisée pour effectuer des agrégations basées sur une combinaison de colonnes de regroupement spécifiée dans la clause GROUP BY. CUBE est un raccourci pour GROUPER DES ENSEMBLES.

Par exemple, `GROUP BY warehouse, product WITH CUBE` or `GROUP BY CUBE(warehouse, product)` équivaut à `GROUP BY GROUPING SETS((warehouse, product), (warehouse), (product), ())`.

`GROUP BY CUBE(warehouse, product, (warehouse, location))` est équivalent à `GROUP BY GROUPING SETS((warehouse, product, location), (warehouse, product), (warehouse, location), (product, warehouse, location), (warehouse), (product), (warehouse, product), ())`. Les N éléments d'une `CUBE` spécification donnent `GROUPING SETS` 2^N.

- Analyse des regroupements mixtes/imbriqués

  Une `GROUP BY` clause peut inclure plusieurs group_expressions et plusieurs. `CUBE|ROLLUP|GROUPING SETS` `GROUPING SETS`peut également comporter des `CUBE|ROLLUP|GROUPING SETS` clauses imbriquées, telles que`GROUPING SETS(ROLLUP(warehouse, location),CUBE(warehouse, location))`, `GROUPING SETS(warehouse, GROUPING SETS(location, GROUPING SETS(ROLLUP(warehouse, location),CUBE(warehouse, location))))`.

  `CUBE|ROLLUP`est juste un sucre de syntaxe pour`GROUPING SETS`. Reportez-vous aux sections ci-dessus pour savoir comment traduire `CUBE|ROLLUP` en`GROUPING SETS`. `group_expression`peut être traité comme un seul groupe `GROUPING SETS` dans ce contexte.

  Pour un multiple `GROUPING SETS` dans la `GROUP BY` clause, nous générons un seul `GROUPING SETS` en faisant un produit croisé de l'original`GROUPING SETS`. Pour imbriquer `GROUPING SETS` dans la `GROUPING SETS` clause, il suffit de prendre ses ensembles de regroupement et de les supprimer.

  Par exemple, `GROUP BY warehouse, GROUPING SETS((product), ()), GROUPING SETS((location, size), (location), (size), ())` and `GROUP BY warehouse, ROLLUP(product), CUBE(location, size)` équivaut à `GROUP BY GROUPING SETS( (warehouse, product, location, size), (warehouse, product, location), (warehouse, product, size), (warehouse, product), (warehouse, location, size), (warehouse, location), (warehouse, size), (warehouse))`.

  `GROUP BY GROUPING SETS(GROUPING SETS(warehouse), GROUPING SETS((warehouse, product)))` est équivalent à `GROUP BY GROUPING SETS((warehouse), (warehouse, product))`.

- nom_agrégat

  Spécifie le nom d'une fonction d'agrégation (`MIN`,`MAX`,`COUNT`,`SUM`,`AVG`,,,, etc.).

- DISTINCT

  Supprime les doublons dans les lignes d'entrée avant qu'ils ne soient transmis aux fonctions d'agrégation.

- FILTRE

  Filtres : les lignes d'entrée pour lesquelles la `WHERE` clause `boolean_expression` in the est évaluée à true sont transmises à la fonction d'agrégation ; les autres lignes sont ignorées.

Exemples

```
CREATE TABLE dealer (id INT, city STRING, car_model STRING, quantity INT);
INSERT INTO dealer VALUES
(100, 'Fremont', 'Honda Civic', 10),
(100, 'Fremont', 'Honda Accord', 15),
(100, 'Fremont', 'Honda CRV', 7),
(200, 'Dublin', 'Honda Civic', 20),
(200, 'Dublin', 'Honda Accord', 10),
(200, 'Dublin', 'Honda CRV', 3),
(300, 'San Jose', 'Honda Civic', 5),
(300, 'San Jose', 'Honda Accord', 8);

-- Sum of quantity per dealership. Group by `id`.
SELECT id, sum(quantity) FROM dealer GROUP BY id ORDER BY id;
+---+-------------+
| id|sum(quantity)|
+---+-------------+
|100|           32|
|200|           33|
|300|           13|
+---+-------------+

-- Use column position in GROUP by clause.
SELECT id, sum(quantity) FROM dealer GROUP BY 1 ORDER BY 1;
+---+-------------+
| id|sum(quantity)|
+---+-------------+
|100|           32|
```

```
|200|           33|
|300|           13|
+---+-------------+

-- Multiple aggregations.
-- 1. Sum of quantity per dealership.
-- 2. Max quantity per dealership.
SELECT id, sum(quantity) AS sum, max(quantity) AS max FROM dealer GROUP BY id ORDER BY
 id;
+---+---+---+
| id|sum|max|
+---+---+---+
|100| 32| 15|
|200| 33| 20|
|300| 13|  8|
+---+---+---+

-- Count the number of distinct dealer cities per car_model.
SELECT car_model, count(DISTINCT city) AS count FROM dealer GROUP BY car_model;
+------------+-----+
|   car_model|count|
+------------+-----+
| Honda Civic|    3|
|   Honda CRV|    2|
|Honda Accord|    3|
+------------+-----+

-- Sum of only 'Honda Civic' and 'Honda CRV' quantities per dealership.
SELECT id, sum(quantity) FILTER (
WHERE car_model IN ('Honda Civic', 'Honda CRV')
) AS `sum(quantity)` FROM dealer
GROUP BY id ORDER BY id;
+---+-------------+
| id|sum(quantity)|
+---+-------------+
|100|           17|
|200|           23|
|300|            5|
+---+-------------+

-- Aggregations using multiple sets of grouping columns in a single statement.
-- Following performs aggregations based on four sets of grouping columns.
-- 1. city, car_model
-- 2. city
```

```
-- 3. car_model
-- 4. Empty grouping set. Returns quantities for all city and car models.
SELECT city, car_model, sum(quantity) AS sum FROM dealer
GROUP BY GROUPING SETS ((city, car_model), (city), (car_model), ())
ORDER BY city;
+---------+------------+---+
|     city|   car_model|sum|
+---------+------------+---+
|     null|        null| 78|
|     null| HondaAccord| 33|
|     null|    HondaCRV| 10|
|     null|  HondaCivic| 35|
|   Dublin|        null| 33|
|   Dublin| HondaAccord| 10|
|   Dublin|    HondaCRV|  3|
|   Dublin|  HondaCivic| 20|
|  Fremont|        null| 32|
|  Fremont| HondaAccord| 15|
|  Fremont|    HondaCRV|  7|
|  Fremont|  HondaCivic| 10|
| San Jose|        null| 13|
| San Jose| HondaAccord|  8|
| San Jose|  HondaCivic|  5|
+---------+------------+---+

-- Group by processing with `ROLLUP` clause.
-- Equivalent GROUP BY GROUPING SETS ((city, car_model), (city), ())
SELECT city, car_model, sum(quantity) AS sum FROM dealer
GROUP BY city, car_model WITH ROLLUP
ORDER BY city, car_model;
+---------+------------+---+
|     city|   car_model|sum|
+---------+------------+---+
|     null|        null| 78|
|   Dublin|        null| 33|
|   Dublin| HondaAccord| 10|
|   Dublin|    HondaCRV|  3|
|   Dublin|  HondaCivic| 20|
|  Fremont|        null| 32|
|  Fremont| HondaAccord| 15|
|  Fremont|    HondaCRV|  7|
|  Fremont|  HondaCivic| 10|
| San Jose|        null| 13|
| San Jose| HondaAccord|  8|
```

```
| San Jose|  HondaCivic|  5|
+---------+------------+---+

-- Group by processing with `CUBE` clause.
-- Equivalent GROUP BY GROUPING SETS ((city, car_model), (city), (car_model), ())
SELECT city, car_model, sum(quantity) AS sum FROM dealer
GROUP BY city, car_model WITH CUBE
ORDER BY city, car_model;
+---------+------------+---+
|     city|   car_model|sum|
+---------+------------+---+
|     null|        null| 78|
|     null| HondaAccord| 33|
|     null|    HondaCRV| 10|
|     null|  HondaCivic| 35|
|   Dublin|        null| 33|
|   Dublin| HondaAccord| 10|
|   Dublin|    HondaCRV|  3|
|   Dublin|  HondaCivic| 20|
|  Fremont|        null| 32|
|  Fremont| HondaAccord| 15|
|  Fremont|    HondaCRV|  7|
|  Fremont|  HondaCivic| 10|
| San Jose|        null| 13|
| San Jose| HondaAccord|  8|
| San Jose|  HondaCivic|  5|
+---------+------------+---+

--Prepare data for ignore nulls example
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
(100, 'Mary', NULL),
(200, 'John', 30),
(300, 'Mike', 80),
(400, 'Dan', 50);

--Select the first row in column age
SELECT FIRST(age) FROM person;
+--------------------+
| first(age, false)  |
+--------------------+
| NULL               |
+--------------------+
```

```
--Get the first row in column `age` ignore nulls,last row in column `id` and sum of
 column `id`.
SELECT FIRST(age IGNORE NULLS), LAST(id), SUM(id) FROM person;
+------------------+-----------------+----------+
| first(age, true)  | last(id, false)  | sum(id)  |
+------------------+-----------------+----------+
| 30                | 400              | 1000     |
+------------------+-----------------+----------+
```

Clause HAVING

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

La `HAVING` clause est utilisée pour filtrer les résultats produits par en `GROUP BY` fonction de la condition spécifiée. Il est souvent utilisé en conjonction avec une `GROUP BY` clause.

Syntaxe

```
HAVING boolean_expression
```

Paramètres

- expression_booléenne

  Spécifie toute expression dont le type de résultat est un booléen. Deux expressions ou plus peuvent être combinées à l'aide des opérateurs logiques (`AND`,`OR`).

  Remarque Les expressions spécifiées dans la `HAVING` clause peuvent uniquement faire référence à :

  1. Constantes
  2. Expressions qui apparaissent dans `GROUP BY`
  3. Fonctions d’agrégation

Exemples

```
CREATE TABLE dealer (id INT, city STRING, car_model STRING, quantity INT);
INSERT INTO dealer VALUES
(100, 'Fremont', 'Honda Civic', 10),
(100, 'Fremont', 'Honda Accord', 15),
(100, 'Fremont', 'Honda CRV', 7),
(200, 'Dublin', 'Honda Civic', 20),
(200, 'Dublin', 'Honda Accord', 10),
(200, 'Dublin', 'Honda CRV', 3),
(300, 'San Jose', 'Honda Civic', 5),
(300, 'San Jose', 'Honda Accord', 8);

-- `HAVING` clause referring to column in `GROUP BY`.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING city = 'Fremont';
+-------+---+
|   city|sum|
+-------+---+
|Fremont| 32|
+-------+---+

-- `HAVING` clause referring to aggregate function.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING sum(quantity) > 15;
+-------+---+
|   city|sum|
+-------+---+
| Dublin| 33|
|Fremont| 32|
+-------+---+

-- `HAVING` clause referring to aggregate function by its alias.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING sum > 15;
+-------+---+
|   city|sum|
+-------+---+
| Dublin| 33|
|Fremont| 32|
+-------+---+

-- `HAVING` clause referring to a different aggregate function than what is present in
-- `SELECT` list.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING max(quantity) > 15;
+------+---+
|  city|sum|
+------+---+
```

```
|Dublin| 33|
+------+---+

-- `HAVING` clause referring to constant expression.
SELECT city, sum(quantity) AS sum FROM dealer GROUP BY city HAVING 1 > 0 ORDER BY city;
+--------+---+
|    city|sum|
+--------+---+
|  Dublin| 33|
| Fremont| 32|
|San Jose| 13|
+--------+---+

-- `HAVING` clause without a `GROUP BY` clause.
SELECT sum(quantity) AS sum FROM dealer HAVING sum(quantity) > 10;
+---+
|sum|
+---+
| 78|
+---+
```

Clause ORDER BY

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

La `ORDER BY` clause est utilisée pour renvoyer les lignes de résultats de manière triée dans l'ordre spécifié par l'utilisateur. Contrairement à la clause SORT BY, cette clause garantit un ordre total dans la sortie.

Syntaxe

```
ORDER BY { expression [ sort_direction | nulls_sort_order ] [ , ... ] }
```

Paramètres

- COMMANDEZ PAR

Spécifie une liste d'expressions séparées par des virgules ainsi que des paramètres `sort_direction` `nulls_sort_order` facultatifs utilisés pour trier les lignes.

- direction_tri

  Spécifie éventuellement s'il faut trier les lignes par ordre croissant ou décroissant.

  Les valeurs valides pour le sens de tri sont `ASC` pour le sens ascendant et `DESC` pour le sens descendant.

  Si le sens de tri n'est pas explicitement spécifié, les lignes sont triées par défaut dans l'ordre croissant.

  Syntaxe : `[ ASC | DESC ]`

- nulls_sort_order

  Spécifie éventuellement si `NULL` les valeurs renvoyées sont des valeurs before/after non nulles.

  Si null_sort_order n'est pas spécifié, triez d'abord si l'ordre de `NULLs` tri est le cas `ASC` et NULLS trie en dernier si l'ordre de tri l'est. `DESC`

  1. Si cette `NULLS FIRST` option est spécifiée, les valeurs NULL sont renvoyées en premier, quel que soit l'ordre de tri.

  2. Si elle `NULLS LAST` est spécifiée, les valeurs NULL sont renvoyées en dernier, quel que soit l'ordre de tri.

  Syntaxe : `[ NULLS { FIRST | LAST } ]`

Exemples

```
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
(100, 'John', 30),
(200, 'Mary', NULL),
(300, 'Mike', 80),
(400, 'Jerry', NULL),
(500, 'Dan',  50);

-- Sort rows by age. By default rows are sorted in ascending manner with NULL FIRST.
SELECT name, age FROM person ORDER BY age;
```

```
+-----+----+
| name| age|
+-----+----+
|Jerry|null|
| Mary|null|
| John|  30|
|  Dan|  50|
| Mike|  80|
+-----+----+

-- Sort rows in ascending manner keeping null values to be last.
SELECT name, age FROM person ORDER BY age NULLS LAST;
+-----+----+
| name| age|
+-----+----+
| John|  30|
|  Dan|  50|
| Mike|  80|
| Mary|null|
|Jerry|null|
+-----+----+

-- Sort rows by age in descending manner, which defaults to NULL LAST.
SELECT name, age FROM person ORDER BY age DESC;
+-----+----+
| name| age|
+-----+----+
| Mike|  80|
|  Dan|  50|
| John|  30|
|Jerry|null|
| Mary|null|
+-----+----+

-- Sort rows in ascending manner keeping null values to be first.
SELECT name, age FROM person ORDER BY age DESC NULLS FIRST;
+-----+----+
| name| age|
+-----+----+
|Jerry|null|
| Mary|null|
| Mike|  80|
|  Dan|  50|
| John|  30|
```

```
+-----+----+

-- Sort rows based on more than one column with each column having different
-- sort direction.
SELECT * FROM person ORDER BY name ASC, age DESC;
+---+-----+----+
| id| name| age|
+---+-----+----+
|500|  Dan|  50|
|400|Jerry|null|
|100| John|  30|
|200| Mary|null|
|300| Mike|  80|
+---+-----+----+
```

Clause JOIN

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Une jointure SQL est utilisée pour combiner les lignes de deux relations en fonction de critères de jointure. La section suivante décrit la syntaxe globale des jointures et les différents types de jointures, ainsi que des exemples.

Syntaxe

```
relation INNER JOIN relation [ join_criteria ]
```

Paramètres

- relation

  Spécifie la relation à joindre.

- type de jointure

  Spécifie le type de jointure.

  Syntaxe : `INNER | CROSS | LEFT OUTER`

- join_criteria

  Spécifie comment les lignes d'une relation seront combinées avec les lignes d'une autre relation.

  Syntaxe : `ON boolean_expression | USING ( column_name [ , ... ] )`

- expression_booléenne

  Spécifie une expression dont le type de retour est booléen.

Types de jointure

- Jointure intérieure

  La jointure interne doit être spécifiée de manière explicite. Il sélectionne les lignes dont les valeurs correspondent dans les deux relations.

  Syntaxe : `relation INNER JOIN relation [ join_criteria ]`

- Jointure gauche

  Une jointure gauche renvoie toutes les valeurs de la relation de gauche et les valeurs correspondantes de la relation de droite, ou ajoute la valeur NULL en cas d'absence de correspondance. Elle est également appelée jointure extérieure gauche.

  Syntaxe : `relation LEFT OUTER JOIN relation [ join_criteria ]`

- Jointure croisée

  Une jointure croisée renvoie le produit cartésien de deux relations.

  Syntaxe : `relation CROSS JOIN relation [ join_criteria ]`

Exemples

```
-- Use employee and department tables to demonstrate different type of joins.
SELECT * FROM employee;
+---+-----+------+
| id| name|deptno|
+---+-----+------+
|105|Chloe|     5|
|103| Paul|     3|
|101| John|     1|
```

```
|102| Lisa|     2|
|104| Evan|     4|
|106|  Amy|     6|
+---+-----+------+
SELECT * FROM department;
+------+-----------+
|deptno|   deptname|
+------+-----------+
|     3|Engineering|
|     2|      Sales|
|     1|  Marketing|
+------+-----------+

-- Use employee and department tables to demonstrate inner join.
SELECT id, name, employee.deptno, deptname
FROM employee INNER JOIN department ON employee.deptno = department.deptno;
+---+-----+------+-----------|
| id| name|deptno|   deptname|
+---+-----+------+-----------|
|103| Paul|     3|Engineering|
|101| John|     1|  Marketing|
|102| Lisa|     2|      Sales|
+---+-----+------+-----------|

-- Use employee and department tables to demonstrate left join.
SELECT id, name, employee.deptno, deptname
FROM employee LEFT JOIN department ON employee.deptno = department.deptno;
+---+-----+------+-----------|
| id| name|deptno|   deptname|
+---+-----+------+-----------|
|105|Chloe|     5|       NULL|
|103| Paul|     3|Engineering|
|101| John|     1|  Marketing|
|102| Lisa|     2|      Sales|
|104| Evan|     4|       NULL|
|106|  Amy|     6|       NULL|
+---+-----+------+-----------|

-- Use employee and department tables to demonstrate cross join.
SELECT id, name, employee.deptno, deptname FROM employee CROSS JOIN department;
+---+-----+------+-----------|
| id| name|deptno|   deptname|
+---+-----+------+-----------|
|105|Chloe|     5|Engineering|
```

```
|105|Chloe|     5|  Marketing|
|105|Chloe|     5|      Sales|
|103| Paul|     3|Engineering|
|103| Paul|     3|  Marketing|
|103| Paul|     3|      Sales|
|101| John|     1|Engineering|
|101| John|     1|  Marketing|
|101| John|     1|      Sales|
|102| Lisa|     2|Engineering|
|102| Lisa|     2|  Marketing|
|102| Lisa|     2|      Sales|
|104| Evan|     4|Engineering|
|104| Evan|     4|  Marketing|
|104| Evan|     4|      Sales|
|106|  Amy|     4|Engineering|
|106|  Amy|     4|  Marketing|
|106|  Amy|     4|      Sales|
+---+-----+------+-----------|
```

Clause LIMIT

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

La `LIMIT` clause est utilisée pour limiter le nombre de lignes renvoyées par l'`SELECT`instruction. En général, cette clause est utilisée conjointement `ORDER BY` pour garantir que les résultats sont déterministes.

Syntaxe

```
LIMIT { ALL | integer_expression }
```

Paramètres

- TOUS

  Si elle est spécifiée, la requête renvoie toutes les lignes. En d'autres termes, aucune limite n'est appliquée si cette option est spécifiée.

- expression_entière

  Spécifie une expression pliable qui renvoie un entier.

Exemples

```
CREATE TABLE person (name STRING, age INT);
INSERT INTO person VALUES
('Jane Doe', 25),
('Pat C', 18),
('Nikki W', 16),
('John D', 25),
('Juan L', 18),
('Jorge S', 16);

-- Select the first two rows.
SELECT name, age FROM person ORDER BY name LIMIT 2;
+-------+---+
|   name|age|
+-------+---+
|  Pat C| 18|
|Jorge S| 16|
+------+---+

-- Specifying ALL option on LIMIT returns all the rows.
SELECT name, age FROM person ORDER BY name LIMIT ALL;
+--------+---+
|    name|age|
+--------+---+
|   Pat C| 18|
| Jorge S| 16|
|  Juan L| 18|
|  John D| 25|
| Nikki W| 16|
|Jane Doe| 25|
+--------+---+

-- A function expression as an input to LIMIT.
SELECT name, age FROM person ORDER BY name LIMIT length('OPENSEARCH');
+-------+---+
|   name|age|
+-------+---+
|  Pat C| 18|
```

```
|Jorge S| 16|
| Juan L| 18|
| John D| 25|
|Nikki W| 16|
+-------+---+
```

Clause CASE

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

La `CASE` clause utilise une règle pour renvoyer un résultat spécifique en fonction de la condition spécifiée, comme dans les instructions if/else dans d'autres langages de programmation.

Syntaxe

```
CASE [ expression ] { WHEN boolean_expression THEN then_expression } [ ... ]
[ ELSE else_expression ]
END
```

Paramètres

- expression_booléenne

  Spécifie toute expression dont le type de résultat est un booléen.

  Deux expressions ou plus peuvent être combinées à l'aide des opérateurs logiques (`AND`,`OR`).

- puis expression

  Spécifie l'expression alors en fonction de la condition boolean_expression.

  `then_expression`et `else_expression` doivent tous être du même type ou être soumis à un type commun.

- exprime_autre

  Spécifie l'expression par défaut.

`then_expression`et `else_expression` doivent tous être du même type ou être soumis à un type commun.

Exemples

```
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
(100, 'John', 30),
(200, 'Mary', NULL),
(300, 'Mike', 80),
(400, 'Dan', 50);
SELECT id, CASE WHEN id > 200 THEN 'bigger' ELSE 'small' END FROM person;
+------+------------------------------------------------------+
|  id  | CASE WHEN (id > 200) THEN bigger ELSE small END  |
+------+------------------------------------------------------+
| 100  | small                                                |
| 200  | small                                                |
| 300  | bigger                                               |
| 400  | bigger                                               |
+------+------------------------------------------------------+
SELECT id, CASE id WHEN 100 then 'bigger' WHEN  id > 300 THEN '300' ELSE 'small' END
 FROM person;
+------
+-------------------------------------------------------------------------------------------------------------
+
|  id  | CASE WHEN (id = 100) THEN bigger WHEN (id = CAST((id > 300) AS INT)) THEN 300
 ELSE small END  |
+------
+-------------------------------------------------------------------------------------------------------------
+
| 100  | bigger
                |
| 200  | small
                |
| 300  | small
                |
| 400  | small
                |
+------
+-------------------------------------------------------------------------------------------------------------
+
```

Expression de table commune

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Une expression de table commune (CTE) définit un jeu de résultats temporaire auquel un utilisateur peut éventuellement faire référence plusieurs fois dans le cadre d'une instruction SQL. Un CTE est principalement utilisé dans une `SELECT` déclaration.

Syntaxe

```
WITH common_table_expression [ , ... ]
```

While `common_table_expression` est défini comme suit :

```
Syntexpression_name [ ( column_name [ , ... ] ) ] [ AS ] ( query )
```

Paramètres

- nom_expression

  Spécifie le nom de l'expression de table commune.
- query

  Une `SELECT` déclaration.

Exemples

```
-- CTE with multiple column aliases
WITH t(x, y) AS (SELECT 1, 2)
SELECT * FROM t WHERE x = 1 AND y = 2;
+---+---+
|  x|  y|
+---+---+
|  1|  2|
+---+---+
```

```
-- CTE in CTE definition
WITH t AS (
WITH t2 AS (SELECT 1)
SELECT * FROM t2
)
SELECT * FROM t;
+---+
|  1|
+---+
|  1|
+---+

-- CTE in subquery
SELECT max(c) FROM (
WITH t(c) AS (SELECT 1)
SELECT * FROM t
);
+------+
|max(c)|
+------+
|     1|
+------+

-- CTE in subquery expression
SELECT (
WITH t AS (SELECT 1)
SELECT * FROM t
);
+----------------+
|scalarsubquery()|
+----------------+
|               1|
+----------------+

-- CTE in CREATE VIEW statement
CREATE VIEW v AS
WITH t(a, b, c, d) AS (SELECT 1, 2, 3, 4)
SELECT * FROM t;
SELECT * FROM v;
+---+---+---+---+
|  a|  b|  c|  d|
+---+---+---+---+
|  1|  2|  3|  4|
```

```
+---+---+---+---+
```

EXPLAIN

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

L'`EXPLAIN`instruction est utilisée pour fournir des plans logiques/physiques pour une instruction d'entrée. Par défaut, cette clause fournit uniquement des informations sur un plan physique.

Syntaxe

```
EXPLAIN [ EXTENDED | CODEGEN | COST | FORMATTED ] statement
```

Paramètres

- ÉTENDU

  Génère un plan logique analysé, un plan logique analysé, un plan logique optimisé et un plan physique.

  Le plan logique analysé est un plan non résolu extrait de la requête.

  Les plans logiques analysés transforment ce qui se traduit `unresolvedAttribute` `unresolvedRelation` en objets entièrement typés.

  Le plan logique optimisé est transformé par le biais d'un ensemble de règles d'optimisation, aboutissant au plan physique.
- CODEGEN

  Génère du code pour l'instruction, le cas échéant, ainsi qu'un plan physique.
- COÛT

  Si les statistiques des nœuds du plan sont disponibles, génère un plan logique et les statistiques.
- FORMATÉ

  Génère deux sections : le plan physique et les détails du nœud.

- déclaration

  Spécifie une instruction SQL à expliquer.

Exemples

```
-- Default Output
EXPLAIN select k, sum(v) from values (1, 2), (1, 3) t(k, v) group by k;
+----------------------------------------------------+
|                                                plan|
+----------------------------------------------------+
| == Physical Plan ==
*(2) HashAggregate(keys=[k#33], functions=[sum(cast(v#34 as bigint))])
+- Exchange hashpartitioning(k#33, 200), true, [id=#59]
+- *(1) HashAggregate(keys=[k#33], functions=[partial_sum(cast(v#34 as bigint))])
+- *(1) LocalTableScan [k#33, v#34]
|
+----------------------------------------------------

-- Using Extended
EXPLAIN EXTENDED select k, sum(v) from values (1, 2), (1, 3) t(k, v) group by k;
+----------------------------------------------------+
|                                                plan|
+----------------------------------------------------+
| == Parsed Logical Plan ==
'Aggregate ['k], ['k, unresolvedalias('sum('v), None)]
 +- 'SubqueryAlias `t`
+- 'UnresolvedInlineTable [k, v], [List(1, 2), List(1, 3)]

 == Analyzed Logical Plan ==
 k: int, sum(v): bigint
 Aggregate [k#47], [k#47, sum(cast(v#48 as bigint)) AS sum(v)#50L]
 +- SubqueryAlias `t`
    +- LocalRelation [k#47, v#48]

 == Optimized Logical Plan ==
 Aggregate [k#47], [k#47, sum(cast(v#48 as bigint)) AS sum(v)#50L]
 +- LocalRelation [k#47, v#48]

 == Physical Plan ==
 *(2) HashAggregate(keys=[k#47], functions=[sum(cast(v#48 as bigint))], output=[k#47,
 sum(v)#50L])
+- Exchange hashpartitioning(k#47, 200), true, [id=#79]
```

```
   +- *(1) HashAggregate(keys=[k#47], functions=[partial_sum(cast(v#48 as bigint))],
 output=[k#47, sum#52L])
    +- *(1) LocalTableScan [k#47, v#48]
|
+----------------------------------------------------+

-- Using Formatted
EXPLAIN FORMATTED select k, sum(v) from values (1, 2), (1, 3) t(k, v) group by k;
+----------------------------------------------------+
|                                                plan|
+----------------------------------------------------+
| == Physical Plan ==
 * HashAggregate (4)
 +- Exchange (3)
    +- * HashAggregate (2)
       +- * LocalTableScan (1)


 (1) LocalTableScan [codegen id : 1]
 Output: [k#19, v#20]

 (2) HashAggregate [codegen id : 1]
 Input: [k#19, v#20]

 (3) Exchange
 Input: [k#19, sum#24L]

 (4) HashAggregate [codegen id : 2]
 Input: [k#19, sum#24L]
|
+----------------------------------------------------+
```

Clause LATERAL SUBQUERY

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

`LATERAL SUBQUERY`est une sous-requête précédée du mot clé`LATERAL`. Il permet de référencer les colonnes de la `FROM` clause précédente. Sans le `LATERAL` mot clé, les sous-requêtes ne peuvent

faire référence qu'aux colonnes de la requête externe, mais pas à celles de la FROM clause. LATERAL SUBQUERYrend les requêtes complexes plus simples et plus efficaces.

Syntaxe

```
[ LATERAL ] primary_relation [ join_relation ]
```

Paramètres

- relation_primaire

  Spécifie la relation principale. Il peut s'agir de l'un des périphériques suivants :

  1. Relation entre les tables
  2. Requête aliasée

     Syntaxe : `( query ) [ [ AS ] alias ]`
  3. Relation aliasée

     `Syntax: ( relation ) [ [ AS ] alias ]`

Exemples

```
CREATE TABLE t1 (c1 INT, c2 INT);
INSERT INTO t1 VALUES (0, 1), (1, 2);
CREATE TABLE t2 (c1 INT, c2 INT);
INSERT INTO t2 VALUES (0, 2), (0, 3);
SELECT * FROM t1,
LATERAL (SELECT * FROM t2 WHERE t1.c1 = t2.c1);
+--------+-------+--------+-------+
|  t1.c1 | t1.c2 |  t2.c1 | t2.c2 |
+-------+--------+--------+-------+
|    0   |   1   |    0   |   3   |
|    0   |   1   |    0   |   2   |
+-------+--------+--------+-------+
SELECT a, b, c FROM t1,
LATERAL (SELECT c1 + c2 AS a),
LATERAL (SELECT c1 - c2 AS b),
LATERAL (SELECT a * b AS c);
+--------+-------+--------+
|    a   |   b   |    c   |
+-------+--------+--------+
```

```
|    3   |  -1   |   -3   |
|    1   |  -1   |   -1   |
+-------+--------+--------+
```

Clause de vue latérale

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

La `LATERAL VIEW` clause est utilisée conjointement avec des fonctions de génération telles que`EXPLODE`, qui généreront une table virtuelle contenant une ou plusieurs lignes. `LATERAL VIEW`appliquera les lignes à chaque ligne de sortie d'origine.

Syntaxe

```
LATERAL VIEW [ OUTER ] generator_function ( expression [ , ... ] ) [ table_alias ] AS
 column_alias [ , ... ]
```

Paramètres

- EXTÉRIEUR

  Si `OUTER` spécifié, renvoie null si une entrée array/map est vide ou nulle.
- fonction_générateur

  Spécifie une fonction de générateur (`EXPLODE`,`INLINE`, etc.).
- alias de table

  L'alias pour`generator_function`, qui est facultatif.
- alias de colonne

  Répertorie les alias de `generator_function` colonne qui peuvent être utilisés dans les lignes de sortie.

  Vous pouvez avoir plusieurs alias s'il `generator_function` comporte plusieurs colonnes de sortie.

Exemples

```
CREATE TABLE person (id INT, name STRING, age INT, class INT, address STRING);
INSERT INTO person VALUES
(100, 'John', 30, 1, 'Street 1'),
(200, 'Mary', NULL, 1, 'Street 2'),
(300, 'Mike', 80, 3, 'Street 3'),
(400, 'Dan', 50, 4, 'Street 4');
SELECT * FROM person
LATERAL VIEW EXPLODE(ARRAY(30, 60)) tableName AS c_age
LATERAL VIEW EXPLODE(ARRAY(40, 80)) AS d_age;
+------+-------+-------+--------+-----------+--------+--------+
|  id  | name  |  age  | class  |  address  | c_age  | d_age  |
+------+-------+-------+--------+-----------+--------+--------+
| 100  | John  | 30    | 1      | Street 1  | 30     | 40     |
| 100  | John  | 30    | 1      | Street 1  | 30     | 80     |
| 100  | John  | 30    | 1      | Street 1  | 60     | 40     |
| 100  | John  | 30    | 1      | Street 1  | 60     | 80     |
| 200  | Mary  | NULL  | 1      | Street 2  | 30     | 40     |
| 200  | Mary  | NULL  | 1      | Street 2  | 30     | 80     |
| 200  | Mary  | NULL  | 1      | Street 2  | 60     | 40     |
| 200  | Mary  | NULL  | 1      | Street 2  | 60     | 80     |
| 300  | Mike  | 80    | 3      | Street 3  | 30     | 40     |
| 300  | Mike  | 80    | 3      | Street 3  | 30     | 80     |
| 300  | Mike  | 80    | 3      | Street 3  | 60     | 40     |
| 300  | Mike  | 80    | 3      | Street 3  | 60     | 80     |
| 400  | Dan   | 50    | 4      | Street 4  | 30     | 40     |
| 400  | Dan   | 50    | 4      | Street 4  | 30     | 80     |
| 400  | Dan   | 50    | 4      | Street 4  | 60     | 40     |
| 400  | Dan   | 50    | 4      | Street 4  | 60     | 80     |
+------+-------+-------+--------+-----------+--------+--------+
SELECT c_age, COUNT(1) FROM person
LATERAL VIEW EXPLODE(ARRAY(30, 60)) AS c_age
LATERAL VIEW EXPLODE(ARRAY(40, 80)) AS d_age
GROUP BY c_age;
+--------+-----------+
| c_age  | count(1)  |
+--------+-----------+
| 60     | 8         |
| 30     | 8         |
+--------+-----------+
SELECT * FROM person
LATERAL VIEW EXPLODE(ARRAY()) tableName AS c_age;
+-----+-------+------+--------+----------+--------+
```

```
| id  | name  | age  | class  | address  | c_age  |
+-----+-------+------+--------+----------+--------+
+-----+-------+------+--------+----------+--------+
SELECT * FROM person
LATERAL VIEW OUTER EXPLODE(ARRAY()) tableName AS c_age;
+------+-------+-------+--------+-----------+--------+
|  id  | name  |  age  | class  |  address  | c_age  |
+------+-------+-------+--------+-----------+--------+
| 100  | John  | 30    | 1      | Street 1  | NULL   |
| 200  | Mary  | NULL  | 1      | Street 2  | NULL   |
| 300  | Mike  | 80    | 3      | Street 3  | NULL   |
| 400  | Dan   | 50    | 4      | Street 4  | NULL   |
+------+-------+-------+--------+-----------+--------+
```

Prédicat LIKE

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Un `LIKE` prédicat est utilisé pour rechercher un modèle spécifique. Ce prédicat prend également en charge plusieurs modèles avec des quantificateurs tels que `ANYSOME`, et. `ALL`

Syntaxe

```
[ NOT ] { LIKE search_pattern [ ESCAPE esc_char ] | [ RLIKE | REGEXP ] regex_pattern }
[ NOT ] { LIKE quantifiers ( search_pattern [ , ... ]) }
```

Paramètres

- modèle_de recherche

  Spécifie un modèle de chaîne à rechercher par la clause LIKE. Il peut contenir des caractères spéciaux correspondant à des modèles :

  - %correspond à zéro ou plusieurs caractères.
  - _correspond exactement à un caractère.
- esc_char

Spécifie le caractère échappe. Le caractère d'échappement par défaut est\.

- modèle_régulier

  Spécifie un modèle de recherche d'expressions régulières à rechercher par la `REGEXP` clause `RLIKE` or.

- quantificateurs

  Spécifie que les quantificateurs de prédicats incluent`ANY`, et`SOME`. `ALL`

  `ANY`ou `SOME` signifie que si l'un des modèles correspond à l'entrée, renvoie true.

  `ALL`signifie que si tous les modèles correspondent à l'entrée, renvoyez vrai.

Exemples

```
CREATE TABLE person (id INT, name STRING, age INT);
INSERT INTO person VALUES
(100, 'John', 30),
(200, 'Mary', NULL),
(300, 'Mike', 80),
(400, 'Dan',  50),
(500, 'Evan_w', 16);
SELECT * FROM person WHERE name LIKE 'M%';
+---+----+----+
| id|name| age|
+---+----+----+
|300|Mike|  80|
|200|Mary|null|
+---+----+----+
SELECT * FROM person WHERE name LIKE 'M_ry';
+---+----+----+
| id|name| age|
+---+----+----+
|200|Mary|null|
+---+----+----+
SELECT * FROM person WHERE name NOT LIKE 'M_ry';
+---+------+---+
| id|  name|age|
+---+------+---+
|500|Evan_W| 16|
|300|  Mike| 80|
```

```
|100|  John| 30|
|400|   Dan| 50|
+---+------+---+
SELECT * FROM person WHERE name RLIKE 'M+';
+---+----+----+
| id|name| age|
+---+----+----+
|300|Mike|  80|
|200|Mary|null|
+---+----+----+
SELECT * FROM person WHERE name REGEXP 'M+';
+---+----+----+
| id|name| age|
+---+----+----+
|300|Mike|  80|
|200|Mary|null|
+---+----+----+
SELECT * FROM person WHERE name LIKE '%\_%';
+---+------+---+
| id|  name|age|
+---+------+---+
|500|Evan_W| 16|
+---+------+---+
SELECT * FROM person WHERE name LIKE '%$_%' ESCAPE '$';
+---+------+---+
| id|  name|age|
+---+------+---+
|500|Evan_W| 16|
+---+------+---+
SELECT * FROM person WHERE name LIKE ALL ('%an%', '%an');
+---+----+----+
| id|name| age|
+---+----+----+
|400| Dan|  50|
+---+----+----+
SELECT * FROM person WHERE name LIKE ANY ('%an%', '%an');
+---+------+---+
| id|  name|age|
+---+------+---+
|400|   Dan| 50|
|500|Evan_W| 16|
+---+------+---+
SELECT * FROM person WHERE name LIKE SOME ('%an%', '%an');
+---+------+---+
```

```
| id|  name|age|
+---+------+---+
|400|   Dan| 50|
|500|Evan_W| 16|
+---+------+---+
SELECT * FROM person WHERE name NOT LIKE ALL ('%an%', '%an');
+---+----+----+
| id|name| age|
+---+----+----+
|100|John|  30|
|200|Mary|null|
|300|Mike|  80|
+---+----+----+
SELECT * FROM person WHERE name NOT LIKE ANY ('%an%', '%an');
+---+------+----+
| id|  name| age|
+---+------+----+
|100|  John|  30|
|200|  Mary|null|
|300|  Mike|  80|
|500|Evan_W|  16|
+---+------+----+
SELECT * FROM person WHERE name NOT LIKE SOME ('%an%', '%an');
+---+------+----+
| id|  name| age|
+---+------+----+
|100|  John|  30|
|200|  Mary|null|
|300|  Mike|  80|
|500|Evan_W|  16|
+---+------+----+
```

OFFSET

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

La OFFSET clause est utilisée pour spécifier le nombre de lignes à ignorer avant de commencer à renvoyer les lignes renvoyées par l'SELECTinstruction. En général, cette clause est utilisée conjointement ORDER BY pour garantir que les résultats sont déterministes.

Syntaxe

```
OFFSET integer_expression
```

Paramètres

- expression_entière

  Spécifie une expression pliable qui renvoie un entier.

Exemples

```
CREATE TABLE person (name STRING, age INT);
INSERT INTO person VALUES
('Jane Doe', 25),
('Pat C', 18),
('Nikki W', 16),
('Juan L', 25),
('John D', 18),
('Jorge S', 16);

-- Skip the first two rows.
SELECT name, age FROM person ORDER BY name OFFSET 2;
+-------+---+
|   name|age|
+-------+---+
| John D| 18|
| Juan L| 25|
|Nikki W| 16|
|Jane Doe| 25|
+-------+---+

-- Skip the first two rows and returns the next three rows.
SELECT name, age FROM person ORDER BY name LIMIT 3 OFFSET 2;
+-------+---+
|   name|age|
+-------+---+
| John D| 18|
```

```
| Juan L| 25|
|Nikki W| 16|
+-------+---+

-- A function expression as an input to OFFSET.
SELECT name, age FROM person ORDER BY name OFFSET length('WAGON');
+-------+---+
|   name|age|
+-------+---+
|Jane Doe| 25|
+-------+---+
```

Clause PIVOT

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

La `PIVOT` clause est utilisée pour la perspective des données. Nous pouvons obtenir les valeurs agrégées en fonction de valeurs de colonne spécifiques, qui seront transformées en plusieurs colonnes utilisées dans la `SELECT` clause. La `PIVOT` clause peut être spécifiée après le nom de la table ou la sous-requête.

Syntaxe

```
PIVOT ( { aggregate_expression [ AS aggregate_expression_alias ] } [ , ... ] FOR
 column_list IN ( expression_list ) )
```

Paramètres

- aggregate_expression

  Spécifie une expression agrégée (`SUM(a)COUNT(DISTINCT b)`, etc.).
- alias d'expression_agrégé

  Spécifie un alias pour l'expression agrégée.
- column_list

Contient des colonnes dans la FROM clause, qui spécifie les colonnes que vous souhaitez remplacer par de nouvelles colonnes. Vous pouvez utiliser des crochets pour entourer les colonnes, par exemple(c1, c2).

- expression_list

  Spécifie les nouvelles colonnes, qui sont utilisées pour faire correspondre les valeurs en column_list tant que condition d'agrégation. Vous pouvez également leur ajouter des alias.

Exemples

```
CREATE TABLE person (id INT, name STRING, age INT, class INT, address STRING);
INSERT INTO person VALUES
(100, 'John', 30, 1, 'Street 1'),
(200, 'Mary', NULL, 1, 'Street 2'),
(300, 'Mike', 80, 3, 'Street 3'),
(400, 'Dan', 50, 4, 'Street 4');
SELECT * FROM person
PIVOT (
SUM(age) AS a, AVG(class) AS c
FOR name IN ('John' AS john, 'Mike' AS mike)
);
+------+-----------+---------+---------+---------+---------+
|  id  |  address  | john_a  | john_c  | mike_a  | mike_c  |
+------+-----------+---------+---------+---------+---------+
| 200  | Street 2  | NULL    | NULL    | NULL    | NULL    |
| 100  | Street 1  | 30      | 1.0     | NULL    | NULL    |
| 300  | Street 3  | NULL    | NULL    | 80      | 3.0     |
| 400  | Street 4  | NULL    | NULL    | NULL    | NULL    |
+------+-----------+---------+---------+---------+---------+
SELECT * FROM person
PIVOT (
SUM(age) AS a, AVG(class) AS c
FOR (name, age) IN (('John', 30) AS c1, ('Mike', 40) AS c2)
);
+------+-----------+-------+-------+-------+-------+
|  id  |  address  | c1_a  | c1_c  | c2_a  | c2_c  |
+------+-----------+-------+-------+-------+-------+
| 200  | Street 2  | NULL  | NULL  | NULL  | NULL  |
| 100  | Street 1  | 30    | 1.0   | NULL  | NULL  |
| 300  | Street 3  | NULL  | NULL  | NULL  | NULL  |
| 400  | Street 4  | NULL  | NULL  | NULL  | NULL  |
```

```
+------+-----------+-------+-------+-------+-------+
```

Définir les opérateurs

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

Les opérateurs d'ensemble sont utilisés pour combiner deux relations d'entrée en une seule. OpenSearch SQL prend en charge trois types d'opérateurs d'ensemble :

- `EXCEPT` ou `MINUS`
- `INTERSECT`
- `UNION`

Les relations d'entrée doivent comporter le même nombre de colonnes et les mêmes types de données compatibles pour les colonnes respectives.

SAUF

`EXCEPT`et `EXCEPT ALL` renvoie les lignes trouvées dans une relation mais pas dans l'autre. `EXCEPT`(également,`EXCEPT DISTINCT`) ne prend que des lignes distinctes `EXCEPT ALL` sans supprimer les doublons des lignes de résultats. Notez qu'il `MINUS` s'agit d'un alias pour`EXCEPT`.

Syntaxe

```
 [ ( ] relation [ ) ] EXCEPT | MINUS [ ALL | DISTINCT ] [ ( ] relation [ ) ]
```

Exemples

```
-- Use table1 and table2 tables to demonstrate set operators in this page.
SELECT * FROM table1;
+---+
|  c|
+---+
|  3|
|  1|
|  2|
```

```
|  2|
|  3|
|  4|
+---+
SELECT * FROM table2;
+---+
|  c|
+---+
|  5|
|  1|
|  2|
|  2|
+---+
SELECT c FROM table1 EXCEPT SELECT c FROM table2;
+---+
|  c|
+---+
|  3|
|  4|
+---+
SELECT c FROM table1 MINUS SELECT c FROM table2;
+---+
|  c|
+---+
|  3|
|  4|
+---+
SELECT c FROM table1 EXCEPT ALL (SELECT c FROM table2);
+---+
|  c|
+---+
|  3|
|  3|
|  4|
+---+
SELECT c FROM table1 MINUS ALL (SELECT c FROM table2);
+---+
|  c|
+---+
|  3|
|  3|
|  4|
+---+
```

SE CROISER

`INTERSECT`et `INTERSECT ALL` renvoient les lignes présentes dans les deux relations. `INTERSECT`(également,`INTERSECT DISTINCT`) ne prend que des lignes distinctes `INTERSECT ALL` sans supprimer les doublons des lignes de résultats.

Syntaxe

```
 [ ( ] relation [ ) ] INTERSECT [ ALL | DISTINCT ] [ ( ] relation [ ) ]
```

Exemples

```
(SELECT c FROM table1) INTERSECT (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  2|
+---+
(SELECT c FROM table1) INTERSECT DISTINCT (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  2|
+---+
(SELECT c FROM table1) INTERSECT ALL (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  2|
|  2|
+---+
```

SYNDICAT

`UNION`et `UNION ALL` renvoient les lignes trouvées dans l'une ou l'autre relation. `UNION`(également,`UNION DISTINCT`) ne prend que des lignes distinctes `UNION ALL` sans supprimer les doublons des lignes de résultats.

Syntaxe

```
[ ( ] relation [ ) ] UNION [ ALL | DISTINCT ] [ ( ] relation [ ) ]
```

Exemples

```
(SELECT c FROM table1) UNION (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  3|
|  5|
|  4|
|  2|
+---+
(SELECT c FROM table1) UNION DISTINCT (SELECT c FROM table2);
+---+
|  c|
+---+
|  1|
|  3|
|  5|
|  4|
|  2|
+---+
SELECT c FROM table1 UNION ALL (SELECT c FROM table2);
+---+
|  c|
+---+
|  3|
|  1|
|  2|
|  2|
|  3|
|  4|
|  5|
|  1|
|  2|
|  2|
+---+
```

Clause TRIER PAR

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

La `SORT BY` clause est utilisée pour renvoyer les lignes de résultats triées dans chaque partition dans l'ordre spécifié par l'utilisateur. Lorsqu'il y a plus d'une partition, le résultat `SORT BY` peut être partiellement ordonné. Ceci est différent de la `ORDER BY` clause qui garantit un ordre total de la sortie.

Syntaxe

```
SORT BY { expression [ sort_direction | nulls_sort_order ] [ , ... ] }
```

Paramètres

- TRIER PAR

  Spécifie une liste d'expressions séparées par des virgules ainsi que les paramètres facultatifs sort_direction et nulls_sort_order qui sont utilisés pour trier les lignes au sein de chaque partition.
- direction_tri

  Spécifie éventuellement s'il faut trier les lignes par ordre croissant ou décroissant.

  Les valeurs valides pour le sens de tri sont `ASC` pour le sens ascendant et `DESC` pour le sens descendant.

  Si le sens de tri n'est pas explicitement spécifié, les lignes sont triées par défaut dans l'ordre croissant.

  Syntaxe : `[ ASC | DESC ]`
- nulls_sort_order

  Spécifie éventuellement si les valeurs NULL sont renvoyées avant/après les valeurs non NULL.

  Si `null_sort_order` ce n'est pas spécifié, NULLs triez d'abord si l'ordre de tri est le cas `ASC` et NULLS trie en dernier si c'est `DESC` le cas.

1. Si cette `NULLS FIRST` option est spécifiée, les valeurs NULL sont renvoyées en premier, quel que soit l'ordre de tri.

2. Si elle `NULLS LAST` est spécifiée, les valeurs NULL sont renvoyées en dernier, quel que soit l'ordre de tri.

Syntaxe : `[ NULLS { FIRST | LAST } ]`

Exemples

```
CREATE TABLE person (zip_code INT, name STRING, age INT);
INSERT INTO person VALUES
(94588, 'Shirley Rodriguez', 50),
(94588, 'Juan Li', 18),
(94588, 'Anil K', 27),
(94588, 'John D', NULL),
(94511, 'David K', 42),
(94511, 'Aryan B.', 18),
(94511, 'Lalit B.', NULL);
-- Sort rows by `name` within each partition in ascending manner
SELECT name, age, zip_code FROM person SORT BY name;
+------------------+----+--------+
|              name| age|zip_code|
+------------------+----+--------+
|            Anil K|  27|   94588|
|           Juan Li|  18|   94588|
|            John D|null|   94588|
| Shirley Rodriguez|  50|   94588|
|          Aryan B.|  18|   94511|
|           David K|  42|   94511|
|          Lalit B.|null|   94511|
+------------------+----+--------+
-- Sort rows within each partition using column position.
SELECT name, age, zip_code FROM person SORT BY 1;
+----------------+----+----------+
|              name| age|zip_code|
+------------------+----+--------+
|            Anil K|  27|   94588|
|           Juan Li|  18|   94588|
|            John D|null|   94588|
| Shirley Rodriguez|  50|   94588|
|          Aryan B.|  18|   94511|
```

```
|           David K|  42|   94511|
|          Lalit B.|null|   94511|
+------------------+----+--------+

-- Sort rows within partition in ascending manner keeping null values to be last.
SELECT age, name, zip_code FROM person SORT BY age NULLS LAST;
+----+------------------+--------+
| age|              name|zip_code|
+----+------------------+--------+
|  18|           Juan Li|   94588|
|  27|            Anil K|   94588|
|  50| Shirley Rodriguez|   94588|
|null|            John D|   94588|
|  18|          Aryan B.|   94511|
|  42|           David K|   94511|
|null|          Lalit B.|   94511|
+--------------+--------+--------+

-- Sort rows by age within each partition in descending manner, which defaults to NULL
 LAST.
SELECT age, name, zip_code FROM person SORT BY age DESC;
+----+------------------+--------+
| age|              name|zip_code|
+----+------------------+--------+
|  50|          Shirley Rodriguez|   94588|
|  27|            Anil K|   94588|
|  18|           Juan Li|   94588|
|null|            John D|   94588|
|  42|           David K|   94511|
|  18|          Aryan B.|   94511|
|null|          Lalit B.|   94511|
+----+------------------+--------+

-- Sort rows by age within each partition in descending manner keeping null values to
 be first.
SELECT age, name, zip_code FROM person SORT BY age DESC NULLS FIRST;
+----+------------------+--------+
| age|              name|zip_code|
+----+------------------+--------+
|null|            John D|   94588|
|  50| Shirley Rodriguez|   94588|
|  27|            Anil K|   94588|
|  18|           Juan Li|   94588|
|null|          Lalit B.|   94511|
```

```
|  42|           David K|   94511|
|  18|          Aryan B.|   94511|
+--------------+--------+--------+

-- Sort rows within each partition based on more than one column with each column
 having
-- different sort direction.
SELECT name, age, zip_code FROM person
SORT BY name ASC, age DESC;
+------------------+----+--------+
|              name| age|zip_code|
+------------------+----+--------+
|            Anil K|  27|   94588|
|           Juan Li|  18|   94588|
|            John D|null|   94588|
| Shirley Rodriguez|  50|   94588|
|          Aryan B.|  18|   94511|
|           David K|  42|   94511|
|          Lalit B.|null|   94511|
+------------------+----+--------+
```

UNPIVOT

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande SQL, consultezthe section called “Commandes SQL prises en charge”.

La `UNPIVOT` clause transforme plusieurs colonnes en plusieurs lignes utilisées dans la `SELECT` clause. La `UNPIVOT` clause peut être spécifiée après le nom de la table ou la sous-requête.

Syntaxe

```
UNPIVOT [ { INCLUDE | EXCLUDE } NULLS ] (
    { single_value_column_unpivot | multi_value_column_unpivot }
) [[AS] alias]

single_value_column_unpivot:
    values_column
    FOR name_column
    IN (unpivot_column [[AS] alias] [, ...])
```

```
multi_value_column_unpivot:
    (values_column [, ...])
    FOR name_column
    IN ((unpivot_column [, ...]) [[AS] alias] [, ...])
```

Paramètres

- unpivot_column

  Contient des colonnes dans la FROM clause, qui spécifie les colonnes que nous voulons défaire pivoter.

- nom_colonne

  Nom de la colonne contenant les noms des colonnes non pivotées.

- colonne_valeurs

  Nom de la colonne contenant les valeurs des colonnes non pivotées.

Exemples

```
CREATE TABLE sales_quarterly (year INT, q1 INT, q2 INT, q3 INT, q4 INT);
INSERT INTO sales_quarterly VALUES
(2020, null, 1000, 2000, 2500),
(2021, 2250, 3200, 4200, 5900),
(2022, 4200, 3100, null, null);
-- column names are used as unpivot columns
SELECT * FROM sales_quarterly
UNPIVOT (
sales FOR quarter IN (q1, q2, q3, q4)
);
+------+---------+-------+
| year | quarter | sales |
+------+---------+-------+
| 2020 | q2      | 1000  |
| 2020 | q3      | 2000  |
| 2020 | q4      | 2500  |
| 2021 | q1      | 2250  |
| 2021 | q2      | 3200  |
| 2021 | q3      | 4200  |
| 2021 | q4      | 5900  |
```

```
| 2022 | q1      | 4200  |
| 2022 | q2      | 3100  |
+------+---------+-------+
-- NULL values are excluded by default, they can be included
-- unpivot columns can be alias
-- unpivot result can be referenced via its alias
SELECT up.* FROM sales_quarterly
UNPIVOT INCLUDE NULLS (
sales FOR quarter IN (q1 AS Q1, q2 AS Q2, q3 AS Q3, q4 AS Q4)
) AS up;
+------+---------+-------+
| year | quarter | sales |
+------+---------+-------+
| 2020 | Q1      | NULL  |
| 2020 | Q2      | 1000  |
| 2020 | Q3      | 2000  |
| 2020 | Q4      | 2500  |
| 2021 | Q1      | 2250  |
| 2021 | Q2      | 3200  |
| 2021 | Q3      | 4200  |
| 2021 | Q4      | 5900  |
| 2022 | Q1      | 4200  |
| 2022 | Q2      | 3100  |
| 2022 | Q3      | NULL  |
| 2022 | Q4      | NULL  |
+------+---------+-------+
-- multiple value columns can be unpivoted per row
SELECT * FROM sales_quarterly
UNPIVOT EXCLUDE NULLS (
(first_quarter, second_quarter)
FOR half_of_the_year IN (
(q1, q2) AS H1,
(q3, q4) AS H2
)
);
+------+------------------+---------------+----------------+
|  id  | half_of_the_year | first_quarter | second_quarter |
+------+------------------+---------------+----------------+
| 2020 | H1               | NULL          | 1000           |
| 2020 | H2               | 2000          | 2500           |
| 2021 | H1               | 2250          | 3200           |
| 2021 | H2               | 4200          | 5900           |
| 2022 | H1               | 4200          | 3100           |
```

```
+------+-----------------+--------------+---------------+
```

Commandes PPL prises en charge

Les tableaux suivants indiquent les commandes PPL prises en charge par OpenSearch Dashboards pour interroger CloudWatch Logs, Amazon S3 ou Security Lake, ainsi que les commandes prises en charge par CloudWatch Logs Insights. CloudWatch Logs Insights utilise la même syntaxe PPL que les OpenSearch tableaux de bord lorsqu'il interroge les CloudWatch journaux, et les tables désignent les deux par le terme « journaux ». CloudWatch

> **Note**
>
> Lorsque vous analysez des données en dehors de OpenSearch Service, les commandes peuvent s'exécuter différemment de ce qu'elles font sur OpenSearch les index.

Rubriques

- Commandes
- Fonctions
- Informations supplémentaires pour les utilisateurs de CloudWatch Logs Insights utilisant OpenSearch PPL

Commandes

commande PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “fields”	Affiche un ensemble de champs devant être projetés.	S	S	S	`fields field1, field2`
the section called “où”	Filtre les données en fonction des	S	S	S	`where field1="success" | where field2 !`

commande PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
	conditions que vous spécifiez.				`= "i -023fe0a9 0929d8822 " | fields field3, col4, col5, col6 | head 1000`
the section called “stats”	Effectue des agrégations et des calculs.	S	S	S	`stats count(), count(`field1`), min(`field1`), max(`field1`), avg(`field1`) by field2 | head 1000`

commande PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “parse”	Extrait un modèle d'expression régulière (regex) d'une chaîne et affiche le modèle extrait. Le modèle extrait peut également être utilisé pour créer de nouveaux champs ou filtrer des données.	S	S	S	`parse `field1` ".*/(?<field2>[^/]+$)" | where field2 = "requestId" | fields field2, `field2` | head 1000`

commande PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “modèles”	Extrait les modèles de journal d'un champ de texte et ajoute les résultats aux résultats de recherche. Le regroupement des journaux en fonction de leurs modèles facilite l'agrégation des statistiques provenant de gros volumes de données de journaux à des fins d'analyse et de résolution des problèmes.	N pris en charge	S	S	`patterns new_field='no_numbers' pattern='[0-9]' message | fields message, no_numbers`

commande PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “sort”	Triez les résultats affichés par nom de champ. Utilisez le tri - FieldNamepour trier par ordre décroissant.	S	S	S	`stats count(), count(`field1`), min(`field1`) as field1Alias, max(`field1`), avg(`field1`) by field2 | sort - field1Alias | head 1000`

commande PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “eval”	Modifie ou traite la valeur d'un champ et la stocke dans un autre champ. Cela est utile pour modifier mathémati quement une colonne, appliquer des fonctions de chaîne à une colonne ou appliquer des fonctions de date à une colonne.	S	S	S	`eval field2 = `field1` * 2 | fields field1, field2 | head 20`
the section called “renommer”	Renomme un ou plusieurs champs dans les résultats de recherche.	S	S	S	`rename field2 as field1 | fields field1`
the section called “head”	Limite les résultats de requête affichés aux N premières lignes.	S	S	S	`fields `@message` | head 20`

commande PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “grok”	Analyse un champ de texte avec un modèle grok basé sur une expression régulière et ajoute les résultats aux résultats de recherche.	S	S	S	`grok email '.+@%{HOSTNAME:host}' | fields email`
the section called “top”	Recherche les valeurs les plus fréquentes pour un champ.	S	S	S	`top 2 Field1 by Field2`
the section called “dedup”	Supprime les entrées dupliquées en fonction des champs que vous spécifiez.	S	S	S	`dedup field1 | fields field1, field2, field3`
the section called “join”	Joint deux ensembles de données.	S	S	S	`source=customer | join ON c_custkey = o_custkey orders | head 10`

commande PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “chercher”	Enrichit vos données de recherche en ajoutant ou en remplaçant les données d'un index de recherche (table de dimensions). Vous pouvez étendre les champs d'un index avec les valeurs d'une table de dimensions, ajouter ou remplacer des valeurs lorsque la condition de recherche correspond	N pris en charge	S	S	`where orderType = 'Cancelled' | lookup account_list, mkt_id AS mkt_code replace amount, account_name as name | stats count(mkt_code), avg(amount) by name`

commande PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “sous-requête”	Exécute des requêtes complexes et imbriquées dans vos instructions PPL (Piped Processing Language).	S	S	S	`where id in [ subquery source=users | where user in [ subquery source=actions | where action="login" | fields user ] | fields uid ]`
the section called “rare”	Recherche les valeurs les moins fréquentes de tous les champs de la liste de champs.	S	S	S	`rare Field1 by Field2`
the section called “ligne de tendance”	Calcule les moyennes mobiles des champs.	S	S	S	`trendline sma(2, field1) as field1Alias`

commande PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “statistiques de l'événement”	Enrichissez les données de vos événements grâce à des statistiques récapitulatives calculées. Il analyse les champs spécifiés au sein de vos événements, calcule diverses mesures statistiques, puis ajoute ces résultats à chaque événement d'origine sous forme de nouveaux champs.	S (sauf `count()`)	S	S	`eventstats sum(field1) by field2`

commande PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “flatten”	Aplatit un champ. Le champ doit être de ce type : `struct<?,?> or array<struct<?,?>>`	S	S	S	`source=table | flatten field1`
the section called “résumé du champ”	Calcule les statistiques de base pour chaque champ (nombre, nombre distinct, min, max, avg, stddev et moyenne).	P en charge (un champ par requête)	S	S	`where field1 != 200 | fieldsummary includefields=field1 nulls=true`
the section called “fillnull”	Remplit les champs nuls avec la valeur que vous fournissez. Il peut être utilisé dans un ou plusieurs domaines.	S	S	S	`fields field1 | eval field2=field1 | fillnull value=0 field1`

commande PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “Développer”	Décompose un champ contenant plusieurs valeurs en lignes distinctes, en créant une nouvelle ligne pour chaque valeur du champ spécifié.	S	S	S	`expand employee | stats max(salary) as max by state, company`
the section called “describe”	Obtient des informations détaillées sur la structure et les métadonnées des tables, des schémas et des catalogues	N pris en charge	S	S	`describe schema.table`

Fonctions

Fonction PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “Chaîne” (CONCAT, CONCAT_WS ,	Fonctions intégrées dans PPL qui peuvent	S	S	S	`eval col1Len = LENGTH(col1) | fields col1Len`

Fonction PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
`LENGTH`, `LOWER`, `LTRIM`, `POSITION`, `REVERSE`, `RIGHT`, `RTRIM`, `SUBSTRING`, `TRIM`, `UPPER`)	manipuler et transformer des chaînes et des données de texte dans les requêtes PPL. Par exemple, convertir des majuscules, combiner des chaînes, extraire des parties et nettoyer du texte.				

Fonction PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “Date et heure” (DAY, DAYOFMONTH , DAY_OF_MONTH ,DAYOFWEEK , DAY_OF_WEEK , DAYOFYEAR , DAY_OF_YEAR , DAYNAME, FROM_UNIXTIME , HOUR, HOUR_OF_DAY , LAST_DAY, LOCALTIMESTAMP , LOCALTIME , MAKE_DATE , MINUTE, MINUTE_OF_HOUR , MONTH, MONTHNAME , MONTH_OF_YEAR , NOW, QUARTER, SECOND, SECOND_OF_MINUTE , SUBDATE, SYSDATE, TIMESTAMP , UNIX_TIMESTAMP , WEEK, WEEKDAY, WEEK_OF_YEAR , DATE_ADD,	Fonctions intégrées pour gérer et transformer les données de date et d'horodatage dans les requêtes PPL. Par exemple, date_add, date_format, datediff et current_date.	S	S	S	`eval newDate = ADDDATE(DATE('2020-08-26'), 1) | fields newDate`

Fonction PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
DATE_SUB, TIMESTAMP ADD , TIMESTAMP DIFF , UTC_TIMESTAMP , CURRENT_TIMEZONE)					
the section called “Condition” (EXISTS, IF, IFNULL, ISNOTNULL , ISNULL, NULLIF)	Fonctions intégrées qui effectuent des calculs sur plusieurs lignes pour produire une seule valeur résumée. Par exemple, sum, count, avg, max et min.	S	S	S	`eval field2 = isnull(col1) | fields field2, col1, field3`

Fonction PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “Mathématiques” (ABS, ACOS, ASIN, ATAN, ATAN2, CEIL, CEILING, CONV, COS, COT, CRC32, DEGREES, E, EXP, FLOOR, LN, LOG, LOG2, LOG10, MOD, PI. POW, POWER, RADIANS, RAND, ROUND, SIGN, SIN, SQRT, CBRT)	Fonctions intégrées pour effectuer des calculs mathématiques et des transformations dans les requêtes PPL. Par exemple : abs (valeur absolue), round (arrondit les nombres), sqrt (racine carrée), pow (calcul de puissance) et ceil (arrondit au nombre entier	S	S	S	`eval field2 = ACOS(col1) | fields col1`

Fonction PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
	le plus proche).				
the section called “Expressions” (Opérateurs arithmétiques (+,,*)-, Opérateurs de prédicat (>. <,) IN)	Les fonctions intégrées pour les expressions, en particulier les expressions de valeur, renvoient une valeur scalaire. Les expressions ont différents types et formes.	S	S	S	`where age > (25 + 5) | fields age`
the section called “Adresse IP” (CIDRMATCH)	Fonctions intégrées pour gérer les adresses IP telles que le CIDR.	S	S	S	`where cidrmatch (ip, '********** ***/24') | fields ip`

Fonction PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “JSON” (ARRAY_LENGTH , ARRAY_LENGTH , JSON, JSON_ARRAY , JSON_EXTRACT , JSON_KEYS , JSON_OBJECT , JSON_VALID , TO_JSON_STRING)	Fonctions intégrées pour gérer le JSON, notamment les tableaux, l'extraction et la validation.	S	S	S	`eval `json_extract('{"a":"b"}', '$.a')` = json_extract('{"a":"b"}', '$a')`
the section called “Lambda” (EXISTS, FILTER, REDUCE, TRANSFORM)	Fonctions intégrées pour gérer le JSON, notamment les tableaux, l'extraction et la validation.	N pris en charge	S	S	`eval array = json_array(1, -1, 2), result = filter(array, x -> x > 0) | fields result`

Fonction PPL	Description	CloudWatch Journaux	Amazon S3	Security Lake	Exemple de commande
the section called “Cryptographique” (MD5, SHA1, SHA2)	Des fonctions intégrées vous permettent de générer des empreintes digitales uniques de données, qui peuvent être utilisées à des fins de vérification, de comparaison ou dans le cadre de protocoles de sécurité plus complexes.	S	S	S	`eval` `MD5('hello')` = MD5('hello') | fields `MD5('hello')`

Informations supplémentaires pour les utilisateurs de CloudWatch Logs Insights utilisant OpenSearch PPL

Bien que CloudWatch Logs Insights prenne en charge la plupart des commandes et fonctions OpenSearch PPL, certaines commandes et fonctions ne le sont pas actuellement. Par exemple, il ne prend actuellement pas en charge les commandes Lookup dans PPL. Depuis le 2 juin 2025, CloudWatch Logs Insights prend désormais en charge les fonctions JOIN, subqueries, Flatten, Fillnull, Expand, Cidrmatch et JSON dans PPL. Pour obtenir la liste complète des commandes et fonctions de requête prises en charge, consultez les colonnes Amazon CloudWatch Logs dans les tableaux ci-dessus.

Exemples de requêtes et de quotas

Ce qui suit s'applique à la fois aux utilisateurs de CloudWatch Logs Insights et OpenSearch aux utilisateurs interrogeant CloudWatch des données.

Pour plus d'informations sur les limites applicables lors de l'interrogation de CloudWatch Logs from OpenSearch Service, consultez la section Quotas de CloudWatch journaux dans le guide de l'utilisateur Amazon CloudWatch Logs. Les limites concernent le nombre de groupes de CloudWatch journaux que vous pouvez interroger, le nombre maximal de requêtes simultanées que vous pouvez exécuter, le temps d'exécution maximal des requêtes et le nombre maximal de lignes renvoyées dans les résultats. Les limites sont les mêmes quel que soit le langage que vous utilisez pour interroger CloudWatch les journaux (à savoir, OpenSearch PPL, SQL et Logs Insights QL).

Commandes PPL

Rubriques

- comment
- commande de corrélation
- commande dedup
- décrire la commande
- commande eval
- commande eventstats
- commande d'extension
- expliquer la commande
- commande fillnull

- commande fields
- commande aplatir
- commande grok
- commande principale
- commande join
- commande de recherche
- commande d'analyse
- commande patterns
- commande rare
- renommer la commande
- commande de recherche
- commande de tri
- commande stats
- commande de sous-requête
- commande supérieure
- commande trendline
- où commande
- résumé du champ
- commande d'extension
- Fonctions PPL

comment

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

PPL prend en charge à la fois les commentaires de ligne et les commentaires de bloc. Le système n'évalue pas le texte des commentaires.

Commentaires sur les lignes

Les commentaires de ligne commencent par deux barres obliques//et se terminent par une nouvelle ligne.

Exemple :

```
os> source=accounts | top gender // finds most common gender of all the accounts
fetched rows / total rows = 2/2
+----------+
| gender   |
|----------|
| M        |
| F        |
+----------+
```

Bloquer les commentaires

Les commentaires en bloc commencent par une barre oblique suivie d'un astérisque \ * et se terminent par un astérisque suivi d'une barre oblique */.

Exemple :

```
os> source=accounts | dedup 2 gender /* dedup the document with gender field keep 2
 duplication */ | fields account_number, gender
fetched rows / total rows = 3/3
+------------------+----------+
| account_number   | gender   |
|------------------+----------|
| 1                | M        |
| 6                | M        |
| 13               | F        |
+------------------+----------+
```

commande de corrélation

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Vous pouvez corréler différentes sources de données selon des dimensions et des délais communs.

Cette corrélation est cruciale lorsque vous traitez de grandes quantités de données provenant de différents secteurs verticaux qui partagent les mêmes périodes mais ne sont pas officiellement synchronisées.

En corrélant ces différentes sources de données en fonction de délais et de dimensions similaires, vous pouvez enrichir vos données et découvrir des informations précieuses.

exemple

Le domaine de l'observabilité comporte trois sources de données distinctes :

- Journaux
- Métriques
- Suivis

Ces sources de données peuvent avoir des dimensions communes. Pour passer d'une source de données à une autre, vous devez les corréler correctement. À l'aide de conventions de dénomination sémantiques, vous pouvez identifier les éléments partagés dans les journaux, les traces et les métriques.

Exemple :

```
{
  "@timestamp": "2018-07-02T22:23:00.186Z",
  "aws": {
    "elb": {
      "backend": {
        "http": {
          "response": {
            "status_code": 500
          }
        },
        "ip": "********",
        "port": "80"
      },
      ...
     "target_port": [
        "10.0.0.1:80"
      ],
      "target_status_code": [
        "500"
```

```
      ],
      "traceId": "Root=1-58337262-36d228ad5d99923122bbe354",
      "type": "http"
    }
  },
  "cloud": {
    "provider": "aws"
  },
  "http": {
    "request": {
    ...
  },
  "communication": {
    "source": {
      "address": "**************",
      "ip": "**************",
      "port": 2817
    }
  },
  "traceId": "Root=1-58337262-36d228ad5d99923122bbe354"
}
```

Cet exemple montre un journal AWS ELB provenant d'un service résidant sur AWS. Il affiche une réponse HTTP du backend avec un code d'état de 500, indiquant une erreur. Cela peut déclencher une alerte ou faire partie de votre processus de surveillance habituel. La prochaine étape consiste à recueillir des données pertinentes sur cet événement afin de mener une enquête approfondie.

Bien que vous soyez tenté d'interroger toutes les données relatives à la période, cette approche peut être accablante. Vous pourriez vous retrouver avec trop d'informations et passer plus de temps à filtrer les données non pertinentes qu'à en identifier la cause première.

Vous pouvez plutôt utiliser une approche plus ciblée en corrélant les données provenant de différentes sources. Vous pouvez utiliser les dimensions suivantes à des fins de corrélation :

- Adresse IP - `"ip": "10.0.0.1" | "ip": "**************"`
- Port - `"port": 2817 | "target_port": "10.0.0.1:80"`

En supposant que vous avez accès à des traces et à des indices de mesures supplémentaires et que vous connaissez la structure de votre schéma, vous pouvez créer une requête de corrélation plus précise.

Voici un exemple de document d'index de trace contenant des informations HTTP que vous souhaiterez peut-être corréler :

```
{
  "traceId": "c1d985bd02e1dbb85b444011f19a1ecc",
  "spanId": "55a698828fe06a42",
  "traceState": [],
  "parentSpanId": "",
  "name": "mysql",
  "kind": "CLIENT",
  "@timestamp": "2021-11-13T20:20:39+00:00",
  "events": [
    {
      "@timestamp": "2021-03-25T17:21:03+00:00",
       ...
    }
  ],
  "links": [
    {
      "traceId": "c1d985bd02e1dbb85b444011f19a1ecc",
      "spanId": "55a698828fe06a42w2",
      },
      "droppedAttributesCount": 0
    }
  ],
  "resource": {
    "service@name": "database",
    "telemetry@sdk@name": "opentelemetry",
    "host@hostname": "ip-172-31-10-8.us-west-2.compute.internal"
  },
  "status": {
    ...
  },
  "attributes": {
    "http": {
      "user_agent": {
        "original": "Mozilla/5.0"
      },
      "network": {
         ...
        }
      },
      "request": {
```

```
          ...
        }
      },
      "response": {
        "status_code": "200",
        "body": {
          "size": 500
        }
      },
      "client": {
        "server": {
          "socket": {
            "address": "***********",
            "domain": "example.com",
            "port": 80
          },
          "address": "***********",
          "port": 80
        },
        "resend_count": 0,
        "url": {
          "full": "http://example.com"
        }
      },
      "server": {
        "route": "/index",
        "address": "***********",
        "port": 8080,
        "socket": {
          ...
        },
        "client": {
          ...
         }
        },
        "url": {
          ...
        }
      }
    }
  }
}
```

Dans cette approche, vous pouvez voir les `traceId` adresses HTTP et HTTP client/server `ip` qui peuvent être corrélées aux journaux ELB afin de mieux comprendre le comportement et l'état du système.

Nouvelle commande de requête de corrélation

Voici la nouvelle commande qui permettrait ce type d'investigation :

```
source alb_logs, traces | where alb_logs.ip="10.0.0.1" AND
 alb_logs.cloud.provider="aws"|
correlate exact fields(traceId, ip) scope(@timestamp, 1D) mapping(alb_logs.ip =
 traces.attributes.http.server.address, alb_logs.traceId = traces.traceId )
```

Voici ce que fait chaque partie de la commande :

1. `source alb_logs, traces`- Cela permet de sélectionner les sources de données que vous souhaitez corréler.
2. `where ip="10.0.0.1" AND cloud.provider="aws"`- Cela permet de réduire la portée de votre recherche.
3. `correlate exact fields(traceId, ip)`- Cela indique au système de corréler les données en fonction des correspondances exactes des champs suivants :
   - Le `ip` champ possède une condition de filtre explicite, il sera donc utilisé dans la corrélation pour toutes les sources de données.
   - Le `traceId` champ ne comporte aucun filtre explicite, il correspondra donc aux mêmes TraceID dans toutes les sources de données.

Les noms des champs indiquent la signification logique de la fonction dans la commande de corrélation. La condition de jointure réelle dépend de l'instruction de mappage que vous fournissez.

Le terme `exact` signifie que les instructions de corrélation nécessiteront que tous les champs correspondent afin de répondre à l'instruction de requête.

Le terme `approximate` tentera de correspondre dans le meilleur des cas et ne rejettera pas les lignes présentant des correspondances partielles.

Aborder différents types de mappage de terrain

Dans les cas où le même champ logique (tel que`ip`) porte des noms différents selon vos sources de données, vous devez fournir le mappage explicite des champs de chemin. Pour résoudre ce

problème, vous pouvez étendre vos conditions de corrélation pour qu'elles correspondent à différents noms de champs ayant des significations logiques similaires. Voici comment vous pouvez procéder :

```
alb_logs.ip = traces.attributes.http.server.address, alb_logs.traceId = traces.traceId
```

Pour chaque champ participant à la jointure par corrélation, vous devez fournir une instruction de mappage pertinente qui inclut toutes les tables à joindre par cette commande de corrélation.

exemple

Dans cet exemple, il existe 2 sources : `alb_logs, traces`

Il y a 2 champs : `traceId, ip`

Il existe deux instructions de mappage : `alb_logs.ip = traces.attributes.http.server.address, alb_logs.traceId = traces.traceId`

Définition des délais de corrélation

Pour simplifier le travail effectué par le moteur d'exécution (pilote), vous pouvez ajouter l'instruction scope. Cela indique explicitement à la requête de jointure l'heure à laquelle elle doit effectuer cette recherche.

`scope(@timestamp, 1D)je`

Dans cet exemple, le champ de recherche se concentre sur une base quotidienne, de sorte que les corrélations apparaissant le même jour sont regroupées. Ce mécanisme de cadrage simplifie et permet un meilleur contrôle des résultats, permettant une résolution de recherche incrémentielle en fonction de vos besoins.

Soutenir les conducteurs

La nouvelle commande de corrélation est en fait une commande de jointure « cachée ». Par conséquent, seuls les pilotes PPL suivants prennent en charge cette commande. Dans ces pilotes, la commande de corrélation sera directement traduite dans le plan logique Catalyst Join approprié.

exemple

```
source alb_logs, traces, metrics | where ip="10.0.0.1" AND
cloud.provider="aws"| correlate exact on (ip, port) scope(@timestamp,
2018-07-02T22:23:00, 1 D)
```

Plan logique :

```
'Project [*]
+- 'Join Inner, ('ip && 'port)
   :- 'Filter (('ip === "10.0.0.1" & 'cloud.provider === "aws") &
 inTimeScope('@timestamp, "2018-07-02T22:23:00", "1 D"))
      +- 'UnresolvedRelation [alb_logs]
   +- 'Join Inner, ('ip & 'port)
      :- 'Filter (('ip === "10.0.0.1" & 'cloud.provider === "aws") &
 inTimeScope('@timestamp, "2018-07-02T22:23:00", "1 D"))
         +- 'UnresolvedRelation [traces]
      +- 'Filter (('ip === "10.0.0.1" & 'cloud.provider === "aws") &
 inTimeScope('@timestamp, "2018-07-02T22:23:00", "1 D"))
         +- 'UnresolvedRelation [metrics]
```

Le moteur Catalyst optimise cette requête en fonction de l'ordre des jointures le plus efficace.

commande dedup

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la `dedup` commande pour supprimer des documents identiques de vos résultats de recherche en fonction des champs spécifiés.

Syntaxe

Utilisez la syntaxe suivante :

```
dedup [int] <field-list> [keepempty=<bool>] [consecutive=<bool>]
```

`int`

- Facultatif.
- La `dedup` commande conserve plusieurs événements pour chaque combinaison lorsque vous la spécifiez<int>. Le nombre pour <int>doit être supérieur à 0. Si vous ne spécifiez aucun chiffre, seul le premier événement est conservé. Tous les autres doublons sont supprimés des résultats.
- Valeur par défaut : 1

`keepempty`

- Facultatif.
- Si la valeur est vraie, conserve les documents pour lesquels un champ de la liste de champs possède une valeur NULL ou est MANQUANT.
- Valeur par défaut : false

`consecutive`

- Facultatif.
- Si la valeur est vraie, supprime uniquement les événements comportant des combinaisons de valeurs dupliquées consécutives.
- Valeur par défaut : false

`field-list`

- Obligatoire.
- Liste de champs séparés par des virgules. Au moins un champ est obligatoire.

Exemple 1 : Déduplication par un champ

Cet exemple montre comment dédupliquer des documents à l'aide du champ de genre.

Requête PPL :

```
os> source=accounts | dedup gender | fields account_number, gender;
fetched rows / total rows = 2/2
+------------------+----------+
| account_number   | gender   |
|------------------+----------|
| 1                | M        |
| 13               | F        |
+------------------+----------+
```

Exemple 2 : conserver 2 doublons de documents

L'exemple montre comment dédupliquer des documents avec le champ de genre, en conservant deux doublons.

Requête PPL :

```
os> source=accounts | dedup 2 gender | fields account_number, gender;
fetched rows / total rows = 3/3
+------------------+----------+
| account_number   | gender   |
|------------------+----------|
| 1                | M        |
| 6                | M        |
| 13               | F        |
+------------------+----------+
```

Exemple 3 : conserver ou ignorer le champ vide par défaut

L'exemple montre comment déduper le document en conservant le champ de valeur nulle.

Requête PPL :

```
os> source=accounts | dedup email keepempty=true | fields account_number, email;
fetched rows / total rows = 4/4
+------------------+-----------------------+
| account_number   | email                 |
+------------------+-----------------------+
| 1                | john_doe@example.com  |
| 6                | jane_doe@example.com  |
| 13               | null                  |
| 18               | juan_li@example.com   |
+------------------+-----------------------+
```

L'exemple montre comment déduper le document en ignorant le champ de valeur vide.

Requête PPL :

```
os> source=accounts | dedup email | fields account_number, email;
fetched rows / total rows = 3/3
+------------------+-----------------------+
| account_number   | email                 |
+------------------+-----------------------+
| 1                | john_doe@example.com  |
| 6                | jane_doe@example.com  |
| 18               | juan_li@example.com   |
+------------------+-----------------------+
```

Exemple 4 : Déduplication dans des documents consécutifs

L'exemple montre comment procéder à la déduplication dans des documents consécutifs.

Requête PPL :

```
os> source=accounts | dedup gender consecutive=true | fields account_number, gender;
fetched rows / total rows = 3/3
+------------------+----------+
| account_number   | gender   |
+------------------+----------+
| 1                | M        |
| 13               | F        |
| 18               | M        |
+------------------+----------+
```

Exemples supplémentaires

- `source = table | dedup a | fields a,b,c`
- `source = table | dedup a,b | fields a,b,c`
- `source = table | dedup a keepempty=true | fields a,b,c`
- `source = table | dedup a,b keepempty=true | fields a,b,c`
- `source = table | dedup 1 a | fields a,b,c`
- `source = table | dedup 1 a,b | fields a,b,c`
- `source = table | dedup 1 a keepempty=true | fields a,b,c`
- `source = table | dedup 1 a,b keepempty=true | fields a,b,c`
- `source = table | dedup 2 a | fields a,b,c`
- `source = table | dedup 2 a,b | fields a,b,c`
- `source = table | dedup 2 a keepempty=true | fields a,b,c`
- `source = table | dedup 2 a,b keepempty=true | fields a,b,c`
- `source = table | dedup 1 a consecutive=true| fields a,b,c`(la déduplication consécutive n'est pas prise en charge)

Limitation

- Pour `| dedup 2 a, b keepempty=false`

```
DataFrameDropColumns('_row_number_)
+- Filter ('_row_number_ <= 2) // allowed duplication = 2
   +- Window [row_number() windowspecdefinition('a, 'b, 'a ASC NULLS FIRST, 'b ASC
 NULLS FIRST, specifiedwindowframe(RowFrame, unboundedpreceding$(), currentrow$()))
 AS _row_number_], ['a, 'b], ['a ASC NULLS FIRST, 'b ASC NULLS FIRST]
      +- Filter (isnotnull('a) AND isnotnull('b)) // keepempty=false
         +- Project
            +- UnresolvedRelation
```

- Pour `| dedup 2 a, b keepempty=true`

```
Union
:- DataFrameDropColumns('_row_number_)
:  +- Filter ('_row_number_ <= 2)
:     +- Window [row_number() windowspecdefinition('a, 'b, 'a ASC NULLS FIRST, 'b ASC
 NULLS FIRST, specifiedwindowframe(RowFrame, unboundedpreceding$(), currentrow$()))
 AS _row_number_], ['a, 'b], ['a ASC NULLS FIRST, 'b ASC NULLS FIRST]
:        +- Filter (isnotnull('a) AND isnotnull('b))
:           +- Project
:              +- UnresolvedRelation
+- Filter (isnull('a) OR isnull('b))
   +- Project
      +- UnresolvedRelation
```

décrire la commande

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la `describe` commande pour obtenir des informations détaillées sur la structure et les métadonnées des tables, des schémas et des catalogues. Voici différents exemples et cas d'utilisation de la `describe` commande.

Describe

- `describe table`Cette commande est égale à la commande `DESCRIBE EXTENDED table` SQL

- `describe schema.table`
- `describe schema.\`table\``
- `describe catalog.schema.table`
- `describe catalog.schema.\`table\``
- `describe \`catalog\`.\`schema\`.\`table\``

commande eval

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

La `eval` commande évalue l'expression et ajoute le résultat au résultat de la recherche.

Syntaxe

Utilisez la syntaxe suivante :

```
eval <field>=<expression> ["," <field>=<expression> ]...
```

- `field`: Obligatoire. Si le nom du champ n'existe pas, un nouveau champ est ajouté. Si le nom du champ existe déjà, il sera remplacé.
- `expression`: Obligatoire. Toute expression prise en charge par le système.

Exemple 1 : créer le nouveau champ

Cet exemple montre comment créer un nouveau `doubleAge` champ pour chaque document. Le nouveau `doubleAge` est le résultat de l'évaluation de l'âge multiplié par 2.

Requête PPL :

```
os> source=accounts | eval doubleAge = age * 2 | fields age, doubleAge ;
fetched rows / total rows = 4/4
+-------+-------------+
| age   | doubleAge   |
|-------+-------------|
```

```
| 32    | 64          |
| 36    | 72          |
| 28    | 56          |
| 33    | 66          |
+-------+-------------+
```

Exemple 2 : remplacer le champ existant

Cet exemple montre comment remplacer le champ d'âge existant par l'âge plus 1.

Requête PPL :

```
os> source=accounts | eval age = age + 1 | fields age ;
fetched rows / total rows = 4/4
+-------+
| age   |
|-------|
| 33    |
| 37    |
| 29    |
| 34    |
+-------+
```

Exemple 3 : créer le nouveau champ avec le champ défini dans eval

Cet exemple montre comment créer un nouveau `ddAge` champ avec un champ défini dans la commande eval. Le nouveau champ `ddAge` est le résultat de l'évaluation `doubleAge` multiplié par 2, `doubleAge` défini dans la commande eval.

Requête PPL :

```
os> source=accounts | eval doubleAge = age * 2, ddAge = doubleAge * 2 | fields age,
 doubleAge, ddAge ;
fetched rows / total rows = 4/4
+-------+-------------+---------+
| age   | doubleAge   | ddAge   |
|-------+-------------+---------|
| 32    | 64          | 128     |
| 36    | 72          | 144     |
| 28    | 56          | 112     |
| 33    | 66          | 132     |
+-------+-------------+---------+
```

Hypothèses :a,b, c existe-t-il des champs dans `table`

Exemples supplémentaires

- `source = table | eval f = 1 | fields a,b,c,f`
- `source = table | eval f = 1`(champs de sortie a, b, c, f)
- `source = table | eval n = now() | eval t = unix_timestamp(a) | fields n,t`
- `source = table | eval f = a | where f > 1 | sort f | fields a,b,c | head 5`
- `source = table | eval f = a * 2 | eval h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = b | stats avg(f) by h`
- `source = table | eval f = ispresent(a)`
- `source = table | eval r = coalesce(a, b, c) | fields r`
- `source = table | eval e = isempty(a) | fields e`
- `source = table | eval e = isblank(a) | fields e`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one', a = 2, 'two', a = 3, 'three', a = 4, 'four', a = 5, 'five', a = 6, 'six', a = 7, 'se7en', a = 8, 'eight', a = 9, 'nine')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else 'unknown')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else concat(a, ' is an incorrect binary digit'))`
- `source = table | eval f = a in ('foo', 'bar') | fields f`
- `source = table | eval f = a not in ('foo', 'bar') | fields f`

Eval avec exemple de cas :

```
source = table | eval e = eval status_category =
case(a >= 200 AND a < 300, 'Success',
a >= 300 AND a < 400, 'Redirection',
a >= 400 AND a < 500, 'Client Error',
a >= 500, 'Server Error'
else 'Unknown')
```

Eval avec un autre exemple de cas :

Hypothèses :a,b, c existe-t-il des champs dans `table`

Exemples supplémentaires

- `source = table | eval f = 1 | fields a,b,c,f`
- `source = table | eval f = 1`(champs de sortie a, b, c, f)
- `source = table | eval n = now() | eval t = unix_timestamp(a) | fields n,t`
- `source = table | eval f = a | where f > 1 | sort f | fields a,b,c | head 5`
- `source = table | eval f = a * 2 | eval h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = b | stats avg(f) by h`
- `source = table | eval f = ispresent(a)`
- `source = table | eval r = coalesce(a, b, c) | fields r`
- `source = table | eval e = isempty(a) | fields e`
- `source = table | eval e = isblank(a) | fields e`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one', a = 2, 'two', a = 3, 'three', a = 4, 'four', a = 5, 'five', a = 6, 'six', a = 7, 'se7en', a = 8, 'eight', a = 9, 'nine')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else 'unknown')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else concat(a, ' is an incorrect binary digit'))`
- `source = table | eval f = a in ('foo', 'bar') | fields f`
- `source = table | eval f = a not in ('foo', 'bar') | fields f`

Eval avec exemple de cas :

```
source = table | eval e = eval status_category =
case(a >= 200 AND a < 300, 'Success',
```

```
a >= 300 AND a < 400, 'Redirection',
a >= 400 AND a < 500, 'Client Error',
a >= 500, 'Server Error'
else 'Unknown')
```

Eval avec un autre exemple de cas :

```
source = table |  where ispresent(a) |
eval status_category =
 case(a >= 200 AND a < 300, 'Success',
  a >= 300 AND a < 400, 'Redirection',
  a >= 400 AND a < 500, 'Client Error',
  a >= 500, 'Server Error'
  else 'Incorrect HTTP status code'
 )
 | stats count() by status_category
```

Limites

- Le remplacement de champs existants n'est pas pris en charge. Les requêtes qui tentent de le faire généreront des exceptions avec le message « La référence « a » est ambiguë ».

```
- `source = table | eval a = 10 | fields a,b,c`
- `source = table | eval a = a * 2 | stats avg(a)`
- `source = table | eval a = abs(a) | where a > 0`
- `source = table | eval a = signum(a) | where a < 0`
```

commande eventstats

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la `eventstats` commande pour enrichir les données de vos événements à l'aide de statistiques récapitulatives calculées. Il fonctionne en analysant des champs spécifiques au sein de vos événements, en calculant diverses mesures statistiques, puis en ajoutant ces résultats sous forme de nouveaux champs à chaque événement d'origine.

Principaux aspects des statistiques sur les événements

1. Il effectue des calculs sur l'ensemble des résultats ou au sein de groupes définis.
2. Les événements d'origine restent intacts et de nouveaux champs ont été ajoutés pour contenir les résultats statistiques.
3. La commande est particulièrement utile pour effectuer des analyses comparatives, identifier les valeurs aberrantes ou fournir un contexte supplémentaire à des événements individuels.

Différence entre les statistiques et les statistiques des événements

Les `eventstats` commandes `stats` and sont toutes deux utilisées pour calculer des statistiques, mais elles présentent des différences importantes dans leur mode de fonctionnement et dans ce qu'elles produisent.

Format de sortie

- `stats`: produit un tableau récapitulatif contenant uniquement les statistiques calculées.
- `eventstats`: ajoute les statistiques calculées sous forme de nouveaux champs aux événements existants, en préservant les données d'origine.

Rétention des événements

- `stats`: réduit le jeu de résultats au seul résumé statistique, en supprimant les événements individuels.
- `eventstats`: conserve tous les événements d'origine et ajoute de nouveaux champs avec les statistiques calculées.

Cas d’utilisation

- `stats`: Idéal pour créer des rapports de synthèse ou des tableaux de bord. Souvent utilisée comme commande finale pour résumer les résultats.
- `eventstats`: Utile lorsque vous devez enrichir des événements avec un contexte statistique pour une analyse ou un filtrage plus approfondis. Peut être utilisé en cours de recherche pour ajouter des statistiques qui pourront être utilisées dans les commandes suivantes.

Syntaxe

Utilisez la syntaxe suivante :

```
eventstats <aggregation>... [by-clause]
```

agrégation

- Obligatoire.
- Une fonction d'agrégation.
- L'argument de l'agrégation doit être un champ.

clause

- Facultatif.
- Syntaxe : `by [span-expression,] [field,]...`
- La clause by peut inclure des champs et des expressions tels que des fonctions scalaires et des fonctions d'agrégation. Vous pouvez également utiliser la clause span pour diviser un champ spécifique en compartiments à intervalles égaux. La commande eventstats effectue ensuite une agrégation en fonction de ces compartiments span.
- Par défaut : si vous ne spécifiez pas de clause by, la commande eventstats agrège l'ensemble des résultats.

étendre l'expression

- Facultatif, au plus un.
- Syntaxe : `span(field_expr, interval_expr)`
- L'unité de l'expression d'intervalle est l'unité naturelle par défaut. Toutefois, pour les champs de type date et heure, vous devez spécifier l'unité dans l'expression d'intervalle lorsque vous utilisez des unités de date/heure.

  Par exemple, pour diviser le champ `age` en compartiments sur 10 ans, utilisez`span(age, 10)`. Pour les champs temporels, vous pouvez diviser un `timestamp` champ en intervalles horaires à l'aide `span(timestamp, 1h)` de.

Unités de temps disponibles

Unités d'intervalle d'intervalle
milliseconde (ms)
seconde (s)
minute (m, distinction majuscules et minuscules)
heure (h)
jour (d)
semaine (s)
mois (M, distinction majuscules et minuscules)
trimestre (q)
année (y)

Fonctions d'agrégation

COUNT

COUNTrenvoie le nombre d'expr dans les lignes récupérées par une instruction SELECT.

Pour les CloudWatch journaux, les requêtes d'utilisation ne COUNT sont pas prises en charge.

Exemple :

```
os> source=accounts | eventstats count();
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------+--------------------
+------------+--------------------------+--------+-------+---------+
| account_number | balance  | firstname | lastname | age | gender | address
 | employer   | email                    | city   | state | count() |
+----------------+----------+-----------+----------+-----+--------+--------------------
+------------+--------------------------+--------+-------+---------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
 | AnyCorp    | janedoe@anycorp.com      | Brogan | IL    | 4       |
```

```
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
 | AnyCompany | marymajor@anycompany.com | Dante  | TN    | 4       |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
 | AnyOrg     |                          | Nogal  | VA    | 4       |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
 |            | juanli@exampleorg.com    | Orick  | MD    | 4       |
+----------------+----------+-----------+----------+-----+--------+--------------------
+------------+--------------------------+--------+-------+---------+
```

SUM

SUM(expr)renvoie la somme de expr.

Exemple :

```
os> source=accounts | eventstats sum(age) by gender;
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+-----------------------+------------+--------------------------+--------+-------
+--------------------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer   | email                    | city   | state | sum(age) by gender |
+----------------+----------+-----------+----------+-----+--------
+-----------------------+------------+--------------------------+--------+-------
+--------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp    | janedoe@anycorp.com      | Brogan | IL    | 101               |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | AnyCompany | marymajor@anycompany.com | Dante  | TN    | 101               |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg     |                          | Nogal  | VA    | 28                |
| 18             | 4180     | Juan      | Li       | 33  | M      | 467 Example Court
   |            | juanli@exampleorg.com    | Orick  | MD    | 101               |
+----------------+----------+-----------+----------+-----+--------
+-----------------------+------------+--------------------------+--------+-------
+--------------------+
```

AVG

AVG(expr)renvoie la valeur moyenne de expr.

Exemple :

```
os> source=accounts | eventstats avg(age) by gender;
```

```
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+------------+--------------------------+--------+-------
+--------------------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | avg(age) by gender |
+----------------+----------+-----------+----------+-----+--------
+----------------------+------------+--------------------------+--------+-------
+--------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 33.67             |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 33.67             |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 28.00             |
| 18             | 4180     | Juan      | Li       | 33  | M      | 467 Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 33.67             |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+--------------------+
```

MAX

`MAX(expr)`Renvoie la valeur maximale de expr.

exemple

```
os> source=accounts | eventstats max(age);
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | max(age)  |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 36        |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 36        |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 36        |
```

```
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 36        |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
```

MIN

`MIN(expr)`Renvoie la valeur minimale de expr.

exemple

```
os> source=accounts | eventstats min(age);
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | min(age)  |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 28        |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 28        |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | *** Any Street
   | AnyOrg      |                          | Nogal  | VA    | 28        |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 28        |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
```

STDDEV_SAMP

`STDDEV_SAMP(expr)`Renvoie l'écart type de l'échantillon de expr.

exemple

```
os> source=accounts | eventstats stddev_samp(age);
fetched rows / total rows = 4/4
```

```
+---------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | stddev_samp(age)       |
+---------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 3.304037933599835      |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 3.304037933599835      |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 3.304037933599835      |
| 18             | 4180     | Juan      | Li       | 33  | M      | 467 Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 3.304037933599835      |
+---------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
```

STDDEV_POP

`STDDEV_POP(expr)`Renvoie l'écart type de population de expr.

exemple

```
os> source=accounts | eventstats stddev_pop(age);
fetched rows / total rows = 4/4
+---------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | stddev_pop(age)        |
+---------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | 880 Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 2.****************     |
| 6              | 5686     | Mary      | Major    | 36  | M      | *** Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 2.****************     |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | *** Any Street
   | AnyOrg      |                          | Nogal  | VA    | 2.****************     |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 2.****************     |
```

```
+---------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+------------------------+
```

PERCENTILE ou PERCENTILE_APPROX

`PERCENTILE(expr, percent)`ou `PERCENTILE_APPROX(expr, percent)` Renvoie la valeur percentile approximative de expr au pourcentage spécifié.

pourcentage

- Le nombre doit être une constante comprise entre 0 et 100.

exemple

```
os> source=accounts | eventstats percentile(age, 90) by gender;
fetched rows / total rows = 4/4
+---------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+--------------------------------+
| account_number | balance  | firstname | lastname | age | gender | address
    | employer    | email                    | city   | state | percentile(age, 90) by
 gender  |
+---------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+--------------------------------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 36
       |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 36
       |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 28
       |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 36
       |
+---------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+--------------------------------+
```

Exemple 1 : calculer la moyenne, la somme et le nombre d'un champ par groupe

L'exemple montre comment calculer l'âge moyen, la somme de l'âge et le nombre d'événements de tous les comptes regroupés par sexe.

```
os> source=accounts | eventstats avg(age) as avg_age, sum(age) as sum_age, count() as
 count by gender;
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+-----------+-------+
| account_number | balance  | firstname | lastname | age | gender | address
    | employer    | email                    | city   | state | avg_age   | sum_age   |
 count |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+-----------+-------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
    | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 33.666667 | 101       |
 3     |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
    | Any Company | marymajor@anycompany.com | Dante  | TN    | 33.666667 | 101       |
 3     |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
    | AnyOrg      |                          | Nogal  | VA    | 28.000000 | 28        |
 1     |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
    |             | juanli@exampleorg.com    | Orick  | MD    | 33.666667 | 101       |
 3     |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+-----------+-------+
```

Exemple 2 : calculer le nombre par intervalle

L'exemple obtient le décompte de l'âge par intervalle de 10 ans.

```
os> source=accounts | eventstats count(age) by span(age, 10) as age_span
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+----------+
```

```
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | age_span |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 3        |
| 6              | 5686     | Mary      | Major    | 36  | M      | 671 Example Street
   | Any Company | marymajor@anycompany.com | Dante  | TN    | 3        |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | 789 Any Street
   | AnyOrg      |                          | Nogal  | VA    | 1        |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 3        |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----------+
```

Exemple 3 : calculer le nombre en fonction du sexe et de l'intervalle

L'exemple obtient le décompte de l'âge par intervalle de 5 ans et le groupe par sexe.

```
os> source=accounts | eventstats count() as cnt by span(age, 5) as age_span, gender
fetched rows / total rows = 4/4
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----+
| account_number | balance  | firstname | lastname | age | gender | address
   | employer    | email                    | city   | state | cnt |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----+
| 1              | 39225    | Jane      | Doe      | 32  | M      | *** Any Lane
   | AnyCorp     | janedoe@anycorp.com      | Brogan | IL    | 2   |
| 6              | 5686     | Mary      | Majo     | 36  | M      | 671 Example Street
   | Any Company | hattiebond@anycompany.com | Dante  | TN    | 1   |
| 13             | 32838    | Nikki     | Wolf     | 28  | F      | *** Any Street
   | AnyOrg      |                          | Nogal  | VA    | 1   |
| 18             | 4180     | Juan      | Li       | 33  | M      | *** Example Court
   |             | juanli@exampleorg.com    | Orick  | MD    | 2   |
+----------------+----------+-----------+----------+-----+--------
+----------------------+-------------+--------------------------+--------+-------
+-----+
```

Utilisation

- `source = table | eventstats avg(a)`
- `source = table | where a < 50 | eventstats avg(c)`
- `source = table | eventstats max(c) by b`
- `source = table | eventstats count(c) by b | head 5`
- `source = table | eventstats distinct_count(c)`
- `source = table | eventstats stddev_samp(c)`
- `source = table | eventstats stddev_pop(c)`
- `source = table | eventstats percentile(c, 90)`
- `source = table | eventstats percentile_approx(c, 99)`

Agrégations avec span

- `source = table | eventstats count(a) by span(a, 10) as a_span`
- `source = table | eventstats sum(age) by span(age, 5) as age_span | head 2`
- `source = table | eventstats avg(age) by span(age, 20) as age_span, country | sort - age_span | head 2`

Agrégations avec plage horaire (fonction de fenêtrage automatique)

- `source = table | eventstats sum(productsAmount) by span(transactionDate, 1d) as age_date | sort age_date`
- `source = table | eventstats sum(productsAmount) by span(transactionDate, 1w) as age_date, productId`

Les agrégations sont regroupées par plusieurs niveaux

- `source = table | eventstats avg(age) as avg_state_age by country, state | eventstats avg(avg_state_age) as avg_country_age by country`
- `source = table | eventstats avg(age) as avg_city_age by country, state, city | eval new_avg_city_age = avg_city_age - 1 | eventstats`

```
avg(new_avg_city_age) as avg_state_age by country, state |
where avg_state_age > 18 | eventstats avg(avg_state_age) as
avg_adult_country_age by country
```

commande d'extension

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la `expand` commande pour aplatir un champ de type :

- `Array<Any>`
- `Map<Any>`

Syntaxe

Utilisez la syntaxe suivante :

```
expand <field> [As alias]
```

field

- Le champ à étendre (exploser). Doit être d'un type compatible.

alias

- Facultatif. Le nom à utiliser à la place du nom de champ d'origine.

Utilisation

La `expand` commande produit une ligne pour chaque élément du tableau ou du champ de carte spécifié, où :

- Les éléments du tableau deviennent des lignes individuelles.

- Les paires clé-valeur de la carte sont divisées en lignes distinctes, chaque valeur-clé étant représentée par une ligne.

- Lorsqu'un alias est fourni, les valeurs éclatées sont représentées sous l'alias au lieu du nom du champ d'origine.
- Cela peut être utilisé en combinaison avec d'autres commandes, telles que `statseval`, et `parse` pour manipuler ou extraire des données après l'extension.

Exemples

- `source = table | expand employee | stats max(salary) as max by state, company`
- `source = table | expand employee as worker | stats max(salary) as max by state, company`
- `source = table | expand employee as worker | eval bonus = salary * 3 | fields worker, bonus`
- `source = table | expand employee | parse description '(?<email>.+@.+)' | fields employee, email`
- `source = table | eval array=json_array(1, 2, 3) | expand array as uid | fields name, occupation, uid`
- `source = table | expand multi_valueA as multiA | expand multi_valueB as multiB`

expliquer la commande

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

La `explain` commande vous aide à comprendre les plans d'exécution des requêtes, ce qui vous permet d'analyser et d'optimiser vos requêtes pour de meilleures performances. Cette introduction fournit un aperçu concis de l'objectif de la commande explain et de son importance dans l'optimisation des requêtes.

Comment

- `source=accounts | top gender // finds most common gender of all the accounts`(commentaire en ligne)
- `source=accounts | dedup 2 gender /* dedup the document with gender field keep 2 duplication */ | fields account_number, gender`(bloquer les commentaires)

Describe

- `describe table`Cette commande est égale à la commande `DESCRIBE EXTENDED table` SQL
- `describe schema.table`
- `describe schema.\`table\``
- `describe catalog.schema.table`
- `describe catalog.schema.\`table\``
- `describe \`catalog\`.\`schema\`.\`table\``

Expliquer

- `explain simple | source = table | where a = 1 | fields a,b,c`
- `explain extended | source = table`
- `explain codegen | source = table | dedup a | fields a,b,c`
- `explain cost | source = table | sort a | fields a,b,c`
- `explain formatted | source = table | fields - a`
- `explain simple | describe table`

Champs

- `source = table`
- `source = table | fields a,b,c`
- `source = table | fields + a,b,c`
- `source = table | fields - b,c`
- `source = table | eval b1 = b | fields - b1,c`

Résumé du champ

- `source = t | fieldsummary includefields=status_code nulls=false`
- `source = t | fieldsummary includefields= id, status_code, request_path nulls=true`
- `source = t | where status_code != 200 | fieldsummary includefields= status_code nulls=true`

Champ imbriqué

- `source = catalog.schema.table1, catalog.schema.table2 | fields A.nested1, B.nested1`
- `source = catalog.table | where struct_col2.field1.subfield > 'valueA' | sort int_col | fields int_col, struct_col.field1.subfield, struct_col2.field1.subfield`
- `source = catalog.schema.table | where struct_col2.field1.subfield > 'valueA' | sort int_col | fields int_col, struct_col.field1.subfield, struct_col2.field1.subfield`

Filtres

- `source = table | where a = 1 | fields a,b,c`
- `source = table | where a >= 1 | fields a,b,c`
- `source = table | where a < 1 | fields a,b,c`
- `source = table | where b != 'test' | fields a,b,c`
- `source = table | where c = 'test' | fields a,b,c | head 3`
- `source = table | where ispresent(b)`
- `source = table | where isnull(coalesce(a, b)) | fields a,b,c | head 3`
- `source = table | where isempty(a)`
- `source = table | where isblank(a)`
- `source = table | where case(length(a) > 6, 'True' else 'False') = 'True'`
- `source = table | where a not in (1, 2, 3) | fields a,b,c`
- `source = table | where a between 1 and 4`- Remarque : Cela renvoie a >= 1 et a <= 4, c'est-à-dire [1, 4]

- `source = table | where b not between '2024-09-10' and '2025-09-10'`- Remarque : cela renvoie b >= '**********' et b <= '2025-09-10'
- `source = table | where cidrmatch(ip, '***********/24')`
- `source = table | where cidrmatch(ipv6, '2003:db8::/32')`
- `source = table | trendline sma(2, temperature) as temp_trend`

Requêtes liées à l'IP

- `source = table | where cidrmatch(ip, '***************')`
- `source = table | where isV6 = false and isValid = true and cidrmatch(ipAddress, '***************')`
- `source = table | where isV6 = true | eval inRange = case(cidrmatch(ipAddress, '2003:***::/32'), 'in' else 'out') | fields ip, inRange`

Filtres complexes

```
source = table | eval status_category =
case(a >= 200 AND a < 300, 'Success',
    a >= 300 AND a < 400, 'Redirection',
    a >= 400 AND a < 500, 'Client Error',
    a >= 500, 'Server Error'
else 'Incorrect HTTP status code')
| where case(a >= 200 AND a < 300, 'Success',
    a >= 300 AND a < 400, 'Redirection',
    a >= 400 AND a < 500, 'Client Error',
    a >= 500, 'Server Error'
else 'Incorrect HTTP status code'
) = 'Incorrect HTTP status code'
```

```
source = table
| eval factor = case(a > 15, a - 14, isnull(b), a - 7, a < 3, a + 1 else 1)
| where case(factor = 2, 'even', factor = 4, 'even', factor = 6, 'even', factor = 8,
 'even' else 'odd') = 'even'
| stats count() by factor
```

Filtres avec conditions logiques

- `source = table | where c = 'test' AND a = 1 | fields a,b,c`
- `source = table | where c != 'test' OR a > 1 | fields a,b,c | head 1`
- `source = table | where c = 'test' NOT a > 1 | fields a,b,c`

Éval

Hypothèses :a,b, c existe-t-il des champs dans `table`

- `source = table | eval f = 1 | fields a,b,c,f`
- `source = table | eval f = 1`(champs de sortie a, b, c, f)
- `source = table | eval n = now() | eval t = unix_timestamp(a) | fields n,t`
- `source = table | eval f = a | where f > 1 | sort f | fields a,b,c | head 5`
- `source = table | eval f = a * 2 | eval h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = f * 2 | fields a,f,h`
- `source = table | eval f = a * 2, h = b | stats avg(f) by h`
- `source = table | eval f = ispresent(a)`
- `source = table | eval r = coalesce(a, b, c) | fields r`
- `source = table | eval e = isempty(a) | fields e`
- `source = table | eval e = isblank(a) | fields e`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one', a = 2, 'two', a = 3, 'three', a = 4, 'four', a = 5, 'five', a = 6, 'six', a = 7, 'se7en', a = 8, 'eight', a = 9, 'nine')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else 'unknown')`
- `source = table | eval f = case(a = 0, 'zero', a = 1, 'one' else concat(a, ' is an incorrect binary digit'))`
- `source = table | eval digest = md5(fieldName) | fields digest`
- `source = table | eval digest = sha1(fieldName) | fields digest`
- `source = table | eval digest = sha2(fieldName,256) | fields digest`

- `source = table | eval digest = sha2(fieldName,512) | fields digest`

commande fillnull

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Description

Utilisez la `fillnull` commande pour remplacer les valeurs nulles par une valeur spécifiée dans un ou plusieurs champs de vos résultats de recherche.

Syntaxe

Utilisez la syntaxe suivante :

```
fillnull [with <null-replacement> in <nullable-field>["," <nullable-field>]] | [using
 <source-field> = <null-replacement> [","<source-field> = <null-replacement>]]
```

- null-replacement : obligatoire. La valeur utilisée pour remplacer les valeurs nulles.
- champ nul : obligatoire. Référence de champ. Les valeurs nulles de ce champ seront remplacées par la valeur spécifiée dans null-replacement.

Exemple 1 : remplir un champ nul

L'exemple montre comment utiliser fillnull sur un seul champ :

```
os> source=logs | fields status_code | eval input=status_code | fillnull with 0 in
 status_code;
| input | status_code |
|-------|-------------|
| 403   | 403         |
| 403   | 403         |
| NULL  | 0           |
| NULL  | 0           |
| 200   | 200         |
```

```
| 404   | 404         |
| 500   | 500         |
| NULL  | 0           |
| 500   | 500         |
| 404   | 404         |
| 200   | 200         |
| 500   | 500         |
| NULL  | 0           |
| NULL  | 0           |
| 404   | 404         |
```

Exemple 2 : Fillnull appliqué à plusieurs champs

L'exemple montre que fillnull est appliqué à plusieurs champs.

```
os> source=logs | fields request_path, timestamp | eval
 input_request_path=request_path, input_timestamp = timestamp | fillnull with '???' in
 request_path, timestamp;
| input_request_path | input_timestamp       | request_path | timestamp             |
|-------------------------------------------------------------------------------------|
| /contact           | NULL                  | /contact     | ???                   |
| /home              | NULL                  | /home        | ???                   |
| /about             | 2023-10-01 10:30:00   | /about       | 2023-10-01 10:30:00   |
| /home              | 2023-10-01 10:15:00   | /home        | 2023-10-01 10:15:00   |
| NULL               | 2023-10-01 10:20:00   | ???          | 2023-10-01 10:20:00   |
| NULL               | 2023-10-01 11:05:00   | ???          | 2023-10-01 11:05:00   |
| /about             | NULL                  | /about       | ???                   |
| /home              | 2023-10-01 10:00:00   | /home        | 2023-10-01 10:00:00   |
| /contact           | NULL                  | /contact     | ???                   |
| NULL               | 2023-10-01 10:05:00   | ???          | 2023-10-01 10:05:00   |
| NULL               | 2023-10-01 10:50:00   | ???          | 2023-10-01 10:50:00   |
| /services          | NULL                  | /services    | ???                   |
| /home              | 2023-10-01 10:45:00   | /home        | 2023-10-01 10:45:00   |
| /services          | 2023-10-01 11:00:00   | /services    | 2023-10-01 11:00:00   |
| NULL               | 2023-10-01 10:35:00   | ???          | 2023-10-01 10:35:00   |
```

Exemple 3 : Fillnull appliqué à plusieurs champs avec différentes valeurs de remplacement nulles.

L'exemple montre fillnull avec différentes valeurs utilisées pour remplacer les valeurs nulles.

- `/error`sur le `request_path` terrain
- `1970-01-01 00:00:00`sur le `timestamp` terrain

```
os> source=logs | fields request_path, timestamp | eval
 input_request_path=request_path, input_timestamp = timestamp | fillnull using
 request_path = '/error', timestamp='1970-01-01 00:00:00';

| input_request_path | input_timestamp       | request_path | timestamp              |
|-------------------------------------------------------------------------------|
| /contact           | NULL                  | /contact     | 1970-01-01 00:00:00    |
| /home              | NULL                  | /home        | 1970-01-01 00:00:00    |
| /about             | 2023-10-01 10:30:00   | /about       | 2023-10-01 10:30:00    |
| /home              | 2023-10-01 10:15:00   | /home        | 2023-10-01 10:15:00    |
| NULL               | 2023-10-01 10:20:00   | /error       | 2023-10-01 10:20:00    |
| NULL               | 2023-10-01 11:05:00   | /error       | 2023-10-01 11:05:00    |
| /about             | NULL                  | /about       | 1970-01-01 00:00:00    |
| /home              | 2023-10-01 10:00:00   | /home        | 2023-10-01 10:00:00    |
| /contact           | NULL                  | /contact     | 1970-01-01 00:00:00    |
| NULL               | 2023-10-01 10:05:00   | /error       | 2023-10-01 10:05:00    |
| NULL               | 2023-10-01 10:50:00   | /error       | 2023-10-01 10:50:00    |
| /services          | NULL                  | /services    | 1970-01-01 00:00:00    |
| /home              | 2023-10-01 10:45:00   | /home        | 2023-10-01 10:45:00    |
| /services          | 2023-10-01 11:00:00   | /services    | 2023-10-01 11:00:00    |
| NULL               | 2023-10-01 10:35:00   | /error       | 2023-10-01 10:35:00    |
```

commande fields

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la `fields` commande pour conserver ou supprimer des champs dans les résultats de recherche.

Syntaxe

Utilisez la syntaxe suivante :

```
field [+|-] <field-list>
```

- `index`: Facultatif.

Si le signe plus (+) est utilisé, seuls les champs spécifiés dans la liste des champs seront conservés.

Si le signe moins (-) est utilisé, tous les champs spécifiés dans la liste des champs seront supprimés.

Par défaut : +

- `field list`: Obligatoire. Liste de champs séparés par des virgules à conserver ou à supprimer.

Exemple 1 : sélectionner les champs spécifiés dans le résultat

Cet exemple montre comment récupérer des `lastname` champs `account_numberfirstname`, et à partir des résultats de recherche.

Requête PPL :

```
os> source=accounts | fields account_number, firstname, lastname;
fetched rows / total rows = 4/4
+------------------+-------------+------------+
| account_number   | firstname   | lastname   |
|------------------+-------------+------------|
| 1                | Jane        | Doe        |
| 6                | John        | Doe        |
| 13               | Jorge       | Souza      |
| 18               | Juan        | Li         |
+------------------+-------------+------------+
```

Exemple 2 : Supprimer les champs spécifiés du résultat

Cet exemple montre comment supprimer le `account_number` champ des résultats de recherche.

Requête PPL :

```
os> source=accounts | fields account_number, firstname, lastname | fields -
 account_number ;
fetched rows / total rows = 4/4
+-------------+------------+
| firstname   | lastname   |
|-------------+------------|
| Jane        | Doe        |
| John        | Doe        |
```

```
| Jorge       | Souza      |
| Juan        | Li         |
+-------------+------------+
```

Exemples supplémentaires

- `source = table`
- `source = table | fields a,b,c`
- `source = table | fields + a,b,c`
- `source = table | fields - b,c`
- `source = table | eval b1 = b | fields - b1,c`

Exemple de champs imbriqués :

```
`source = catalog.schema.table1, catalog.schema.table2 | fields A.nested1, B.nested1`
`source = catalog.table | where struct_col2.field1.subfield > 'valueA' | sort int_col |
 fields  int_col, struct_col.field1.subfield, struct_col2.field1.subfield`
`source = catalog.schema.table | where struct_col2.field1.subfield > 'valueA' | sort
 int_col | fields  int_col, struct_col.field1.subfield, struct_col2.field1.subfield`
```

commande aplatir

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la commande aplatir pour développer les champs des types suivants :

- `struct<?,?>`
- `array<struct<?,?>>`

Syntaxe

Utilisez la syntaxe suivante :

```
flatten <field>
```

- champ : champ à aplatir. Le champ doit être d'un type compatible.

Schema (Schéma)

col_name	data_type
_heure	chaîne
ponts	<length:bigint, name:string>tableau <struc>
city	chaîne
couleur	structure<alt:bigint, lat:double, long:doub le>
country	chaîne

Données

_heure	ponts	city	couleur	country
13/09/2024 À 12:00:00	[{801, Tower Bridge}, {928, London Bridge}]	Londres	{35, 51,5074, -0,1278}	Angleterre
13/09/2024 À 12:00:00	[{232, Pont-Neuf}, {160, Pont Alexandre III}]	Paris	{35, 48,8566, 2,3522}	France
13/09/2024 À 12:00:00	[{48, Pont du	Venise	{2, 45,4408, 12,3155}	Italie

_heure	ponts	city	couleur	country
	Rialto}, {11, Pont des Soupirs}]			
13/09/2024 À 12:00:00	[{***, Pont Charles}, {343, Pont de la Légion}]	Prague	{200, 50,0755, 14,4378}	République tchèque
13/09/2024 À 12:00:00	[{375, Pont à chaînes}, {333, Pont de la Liberté}]	Budapest	{96, 47,4979, 19,0402}	Hongrie
13/09/1990 12:00:00	NULL	Varsovie	NULL	Pologne

Exemple 1 : aplatir la structure

Cet exemple montre comment aplatir un champ de structure.

Requête PPL :

```
source=table | flatten coor
```

_heure	ponts	city	country	alt	lat	long
13/09/2024 À 12:00:00	[{801, Tower Bridge}, {928, London Bridge}]	Londres	Angleterre	35	51,5074	-0,1278

_heure	ponts	city	country	alt	lat	long
13/09/2024 À 12:00:00	[{232, Pont-Neuf}, {160, Pont Alexandre III}]	Paris	France	35	48,8566	2,3522
13/09/2024 À 12:00:00	[{48, Pont du Rialto}, {11, Pont des Soupirs}]	Venise	Italie	2	45,4408	12,3155
13/09/2024 À 12:00:00	[{516, Pont Charles}, {343, Pont de la Légion}]	Prague	République tchèque	200	50,0755	14,4378
13/09/2024 À 12:00:00	[{375, Pont à chaînes}, {333, Pont de la Liberté}]	Budapest	Hongrie	96	47,4979	19,0402
13/09/1990 12:00:00	NULL	Varsovie	Pologne	NULL	NULL	NULL

Exemple 2 : aplatir un tableau

L'exemple montre comment aplatir un tableau de champs de structure.

Requête PPL :

```
source=table | flatten bridges
```

_heure	city	couleur	country	longueur	name
13/09/2024 À 12:00:00	Londres	{35, 51,5074, -0,1278}	Angleterre	801	Tower Bridge
13/09/2024 À 12:00:00	Londres	{35, 51,5074, -0,1278}	Angleterre	928	Pont de Londres
13/09/2024 À 12:00:00	Paris	{35, 48,8566, 2,3522}	France	232	Pont-Neuf
13/09/2024 À 12:00:00	Paris	{35, 48,8566, 2,3522}	France	160	Pont Alexandre III
13/09/2024 À 12:00:00	Venise	{2, 45,4408, 12,3155}	Italie	48	Pont du Rialto
13/09/2024 À 12:00:00	Venise	{2, 45,4408, 12,3155}	Italie	11	Pont des Soupirs
13/09/2024 À 12:00:00	Prague	{200, 50,0755, 14,4378}	République tchèque	516	Pont Charles
13/09/2024 À 12:00:00	Prague	{200, 50,0755, 14,4378}	République tchèque	343	Pont de la Legion
13/09/2024 À 12:00:00	Budapest	{96, 47,4979, 19,0402}	Hongrie	375	Pont à chaînes
13/09/2024 À 12:00:00	Budapest	{96, 47,4979, 19,0402}	Hongrie	333	Pont de la Liberté

_heure	city	couleur	country	longueur	name
13/09/1990 12:00:00	Varsovie	NULL	Pologne	NULL	NULL

Exemple 3 : aplatir un tableau et une structure

Cet exemple montre comment aplatir plusieurs champs.

Requête PPL :

```
source=table | flatten bridges | flatten coor
```

_heure	city	country	longueur	name	alt	lat	long
13/09/2024 À 12:00:00	Londres	Angleterre	801	Tower Bridge	35	51,5074	-0,1278
13/09/2024 À 12:00:00	Londres	Angleterre	928	Pont de Londres	35	51,5074	-0,1278
13/09/2024 À 12:00:00	Paris	France	232	Pont-Neuf	35	48,8566	2,3522
13/09/2024 À 12:00:00	Paris	France	160	Pont Alexandre III	35	48,8566	2,3522
13/09/2024 À 12:00:00	Venise	Italie	48	Pont du Rialto	2	45,4408	12,3155
13/09/2024 À 12:00:00	Venise	Italie	11	Pont des Soupirs	2	45,4408	12,3155

_heure	city	country	longueur	name	alt	lat	long
13/09/2024 À 12:00:00	Prague	République tchèque	516	Pont Charles	200	50,0755	14,4378
13/09/2024 À 12:00:00	Prague	République tchèque	343	Pont de la Legion	200	50,0755	14,4378
13/09/2024 À 12:00:00	Budape	Hongrie	375	Pont à chaînes	96	47,4979	19,0402
13/09/2024 À 12:00:00	Budape	Hongrie	333	Pont de la Liberté	96	47,4979	19,0402
13/09/1990 12:00:00	Varsovi	Pologne	NULL	NULL	NULL	NULL	NULL

commande grok

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

La `grok` commande analyse un champ de texte à l'aide d'un motif grok et ajoute les résultats au résultat de recherche.

Syntaxe

Utilisez la syntaxe suivante :

```
grok <field> <pattern>
```

field

- Obligatoire.
- Le champ doit être un champ de texte.

pattern

- Obligatoire.
- Le modèle grok utilisé pour extraire de nouveaux champs d'un champ de texte donné.
- Si un nouveau nom de champ existe déjà, il remplacera le champ d'origine.

Modèle grok

Le modèle grok est utilisé pour faire correspondre le champ de texte de chaque document afin d'extraire de nouveaux champs.

Exemple 1 : créer le nouveau champ

Cet exemple montre comment créer un nouveau champ `host` pour chaque document. `host`sera le nom d'hôte indiqué après `@` dans le `email` champ. L'analyse d'un champ nul renverra une chaîne vide.

```
os> source=accounts | grok email '.+@%{HOSTNAME:host}' | fields email, host ;
fetched rows / total rows = 4/4
+-------------------------+-------------+
| email                   | host        |
|-------------------------+-------------|
| jane_doe@example.com    | example.com |
| arnav_desai@example.net | example.net |
| null                    |             |
| juan_li@example.org     | example.org |
+-------------------------+-------------+
```

Exemple 2 : remplacer le champ existant

Cet exemple montre comment remplacer le `address` champ existant en supprimant le numéro de rue.

```
os> source=accounts | grok address '%{NUMBER} %{GREEDYDATA:address}' | fields address ;
```

```
fetched rows / total rows = 4/4
+------------------+
| address          |
|------------------|
| Example Lane     |
| Any Street       |
| Main Street      |
| Example Court    |
+------------------+
```

Exemple 3 : utilisation de grok pour analyser les journaux

Cet exemple montre comment utiliser grok pour analyser des journaux bruts.

```
os> source=apache | grok message '%{COMMONAPACHELOG}' | fields COMMONAPACHELOG,
 timestamp, response, bytes ;
fetched rows / total rows = 4/4
+-----------------------------------------------------------------------------------------------------------
+----------------------------+------------+---------+
| COMMONAPACHELOG
                                          | timestamp                  | response   |
 bytes   |
|-----------------------------------------------------------------------------------------------------------
+----------------------------+------------+---------|
| 177.95.8.74 - upton5450 [28/Sep/2022:10:15:57 -0700] "HEAD /e-business/mindshare
 HTTP/1.0" 404 19927                        | 28/Sep/2022:10:15:57 -0700 | 404        |
 19927   |
| 127.45.152.6 - pouros8756 [28/Sep/2022:10:15:57 -0700] "GET /architectures/
convergence/niches/mindshare HTTP/1.0" 100 28722 | 28/Sep/2022:10:15:57 -0700 | 100
    | 28722   |
| *************** - - [28/Sep/2022:10:15:57 -0700] "PATCH /strategize/out-of-the-box
 HTTP/1.0" 401 27439                       | 28/Sep/2022:10:15:57 -0700 | 401        |
 27439   |
| ************** - - [28/Sep/2022:10:15:57 -0700] "POST /users HTTP/1.1" 301 9481
                                       | 28/Sep/2022:10:15:57 -0700 | 301        | 9481
    |
+-----------------------------------------------------------------------------------------------------------
+----------------------------+------------+---------+
```

Limites

La commande grok a les mêmes limites que la commande parse.

commande principale

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la `head` commande pour renvoyer le premier nombre N de résultats spécifiés après un décalage facultatif dans l'ordre de recherche.

Syntaxe

Utilisez la syntaxe suivante :

```
head [<size>] [from <offset>]
```

<size>

- Entier facultatif.
- Le nombre de résultats à renvoyer.
- Par défaut: 10

<offset>

- Entier après facultatif`from`.
- Le nombre de résultats à ignorer.
- Par défaut : 0

Exemple 1 : Obtenir les 10 premiers résultats

Cet exemple montre comment récupérer un maximum de 10 résultats à partir de l'index des comptes.

Requête PPL :

```
os> source=accounts | fields firstname, age | head;
fetched rows / total rows = 4/4
```

```
+-------------+-------+
| firstname   | age   |
|-------------+-------|
| Jane        | 32    |
| John        | 36    |
| Jorge       | 28    |
| Juan        | 33    |
+-------------+-------+
```

Exemple 2 : obtenir les N premiers résultats

L'exemple montre les N premiers résultats de l'index des comptes.

Requête PPL :

```
os> source=accounts | fields firstname, age | head 3;
fetched rows / total rows = 3/3
+-------------+-------+
| firstname   | age   |
|-------------+-------|
| Jane        | 32    |
| John        | 36    |
| Jorge       | 28    |
+-------------+-------+
```

Exemple 3 : obtenir les N premiers résultats après le décalage M

Cet exemple montre comment récupérer les N premiers résultats après avoir ignoré M résultats de l'index des comptes.

Requête PPL :

```
os> source=accounts | fields firstname, age | head 3 from 1;
fetched rows / total rows = 3/3
+-------------+-------+
| firstname   | age   |
|-------------+-------|
| John        | 36    |
| Jorge       | 28    |
| Juan        | 33    |
+-------------+-------+
```

commande join

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

La commande join vous permet de combiner des données provenant de plusieurs sources sur la base de champs communs, ce qui vous permet d'effectuer des analyses complexes et d'obtenir des informations plus approfondies à partir de vos ensembles de données distribués

Schema

Il existe au moins deux indices, `otel-v1-apm-span-*` (grand) et `otel-v1-apm-service-map` (petit).

Champs pertinents issus des indices :

`otel-v1-apm-span-*`

- TraceID : identifiant unique pour une trace. Tous les spans d'une même trace partagent le même TraceID.
- SpanID : identifiant unique d'une plage au sein d'une trace, attribué lors de la création de la plage.
- parentSpanId - Le SpanID de l'espace parent de ce span. S'il s'agit d'une plage racine, ce champ doit être vide.
- durationInNanos - La différence en nanosecondes entre StartTime et EndTime. (c'est `latency` dans l'interface utilisateur)
- ServiceName : ressource d'où provient le span.
- TraceGroup : nom de l'intervalle racine de la trace.

`otel-v1-apm-service-map`

- ServiceName : nom du service qui a émis le span.
- destination.domain - Le ServiceName du service appelé par ce client.
- destination.resource - Le nom de l'intervalle (API, opération, etc.) appelé par ce client.

- target.domain - Le ServiceName du service appelé par un client.
- target.resource - Le nom de l'intervalle (API, opération, etc.) appelé par un client.
- traceGroupName - Le nom de l'intervalle de niveau supérieur qui a lancé la chaîne de demandes.

Exigence

Support join pour calculer les éléments suivants :

Pour chaque service, associez l'index span à l'index de la carte des services afin de calculer les métriques selon différents types de filtres.

Cet exemple de requête calcule le temps de latence lorsqu'il est filtré par groupe de traces `client_cancel_order` pour le `order` service.

```
SELECT avg(durationInNanos)
FROM `otel-v1-apm-span-000001` t1
WHERE t1.serviceName = `order`
  AND ((t1.name in
          (SELECT target.resource
           FROM `otel-v1-apm-service-map`
           WHERE serviceName = `order`
             AND traceGroupName = `client_cancel_order`)
        AND t1.parentSpanId != NULL)
       OR (t1.parentSpanId = NULL
           AND t1.name = `client_cancel_order`))
  AND t1.traceId in
    (SELECT traceId
     FROM `otel-v1-apm-span-000001`
     WHERE serviceName = `order`)
```

Migrer vers PPL

Syntaxe de la commande join

```
SEARCH source=<left-table>
| <other piped command>
| [joinType] JOIN
    [leftAlias]
    ON joinCriteria
    <right-table>
```

```
| <other piped command>
```

Réécriture

```
SEARCH source=otel-v1-apm-span-000001
| WHERE serviceName = 'order'
| JOIN left=t1 right=t2
    ON t1.traceId = t2.traceId AND t2.serviceName = 'order'
    otel-v1-apm-span-000001 -- self inner join
| EVAL s_name = t1.name -- rename to avoid ambiguous
| EVAL s_parentSpanId = t1.parentSpanId -- RENAME command would be better when it is
 supported
| EVAL s_durationInNanos = t1.durationInNanos
| FIELDS s_name, s_parentSpanId, s_durationInNanos -- reduce colunms in join
| LEFT JOIN left=s1 right=t3
    ON s_name = t3.target.resource AND t3.serviceName = 'order' AND t3.traceGroupName =
 'client_cancel_order'
    otel-v1-apm-service-map
| WHERE (s_parentSpanId IS NOT NULL OR (s_parentSpanId IS NULL AND s_name =
 'client_cancel_order'))
| STATS avg(s_durationInNanos) -- no need to add alias if there is no ambiguous
```

Type de joint

- Syntaxe : `INNER | LEFT OUTER | CROSS`
- Facultatif
- Type de jointure à effectuer. La valeur par défaut est `INNER` si elle n'est pas spécifiée.

Alias de gauche

- Syntaxe : `left = <leftAlias>`
- Facultatif
- Alias de sous-requête à utiliser avec le côté gauche de la jointure, afin d'éviter toute ambiguïté en matière de dénomination.

Critères d'adhésion

- Syntaxe : `<expression>`

- Obligatoire
- La syntaxe commence par`ON`. Il peut s'agir de n'importe quelle expression de comparaison. En général, les critères de jointure ressemblent à`<leftAlias>.<leftField>=<rightAlias>.<rightField>`.

  Par exemple : `l.id = r.id`. Si les critères de jointure contiennent plusieurs conditions, vous pouvez spécifier `AND` un `OR` opérateur entre chaque expression de comparaison. Par exemple, `l.id = r.id AND l.email = r.email AND (r.age > 65 OR r.age < 18)`.

Plus d'exemples

Migration depuis une requête SQL (TPC-H Q13) :

```
SELECT c_count, COUNT(*) AS custdist
FROM
  ( SELECT c_custkey, COUNT(o_orderkey) c_count
    FROM customer LEFT OUTER JOIN orders ON c_custkey = o_custkey
        AND o_comment NOT LIKE '%unusual%packages%'
    GROUP BY c_custkey
  ) AS c_orders
GROUP BY c_count
ORDER BY custdist DESC, c_count DESC;
```

Réécrit par une requête de jointure PPL :

```
SEARCH source=customer
| FIELDS c_custkey
| LEFT OUTER JOIN
    ON c_custkey = o_custkey AND o_comment NOT LIKE '%unusual%packages%'
    orders
| STATS count(o_orderkey) AS c_count BY c_custkey
| STATS count() AS custdist BY c_count
| SORT - custdist, - c_count
```

Limitation : les sous-recherches ne sont pas prises en charge dans le côté droit de jointure.

Si les sous-recherches sont prises en charge, vous pouvez réécrire la requête PPL ci-dessus comme suit :

```
SEARCH source=customer
```

```
| FIELDS c_custkey
| LEFT OUTER JOIN
   ON c_custkey = o_custkey
   [
      SEARCH source=orders
      | WHERE o_comment NOT LIKE '%unusual%packages%'
      | FIELDS o_orderkey, o_custkey
   ]
| STATS count(o_orderkey) AS c_count BY c_custkey
| STATS count() AS custdist BY c_count
| SORT - custdist, - c_count
```

commande de recherche

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la `lookup` commande pour enrichir vos données de recherche en ajoutant ou en remplaçant les données d'un index de recherche (table de dimensions). Cette commande permet d'étendre les champs d'un index avec des valeurs issues d'une table de dimensions. Vous pouvez également l'utiliser pour ajouter ou remplacer des valeurs lorsque les conditions de recherche sont remplies. La `lookup` commande est plus adaptée que la `Join` commande pour enrichir les données sources avec un ensemble de données statique.

Syntaxe

Utilisez la syntaxe suivante :

```
SEARCH source=<sourceIndex>
| <other piped command>
| LOOKUP <lookupIndex> (<lookupMappingField> [AS <sourceMappingField>])...
    [(REPLACE | APPEND) (<inputField> [AS <outputField>])...]
| <other piped command>
```

Index de recherche

- Obligatoire.

- Nom de l'index de recherche (table de dimensions).

lookupMappingField

- Obligatoire.
- Une clé de mappage dans l'index de recherche, analogue à une clé de jointure de la table de droite. Vous pouvez spécifier plusieurs champs, séparés par des virgules.

sourceMappingField

- Facultatif.
- Par défaut : **<** lookupMappingField **>**.
- Une clé de mappage issue de la requête source, analogue à une clé de jointure située sur le côté gauche.

Champ de saisie

- Facultatif.
- Par défaut : tous les champs de l'index de recherche contenant des valeurs correspondantes sont trouvées.
- Champ de l'index de recherche dans lequel les valeurs correspondantes sont appliquées au résultat en sortie. Vous pouvez spécifier plusieurs champs, séparés par des virgules.

Champ de sortie

- Facultatif.
- Par défaut: `<inputField>`.
- Un champ dans la sortie. Vous pouvez spécifier plusieurs champs de sortie. Si vous spécifiez un nom de champ existant à partir de la requête source, ses valeurs seront remplacées ou ajoutées par des valeurs correspondantes provenant d'InputField. Si vous spécifiez un nouveau nom de champ, il sera ajouté aux résultats.

REMPLACER | AJOUTER

- Facultatif.

- Par défaut : REPLACE
- Spécifie comment gérer les valeurs correspondantes. Si vous spécifiez REPLACE, les valeurs correspondantes dans <lookupIndex>le champ remplacent les valeurs du résultat. Si vous le spécifiezAPPEND, les valeurs correspondantes dans <lookupIndex>le champ ne sont ajoutées qu'aux valeurs manquantes dans le résultat.

Utilisation

- <lookupIndex>ID DE RECHERCHE EN TANT QUE CID REMPLACER LE courrier EN TANT QU'e-mail
- <lookupIndex>NOM DE RECHERCHE REMPLACER LE courrier PAR e-mail
- <lookupIndex>ID de recherche sous forme d'identifiant, nom, adresse d'ajout, e-mail
- <lookupIndex>ID LOOKUP

exemple

Voir les exemples suivantes.

```
SEARCH source=<sourceIndex>
| WHERE orderType = 'Cancelled'
| LOOKUP account_list, mkt_id AS mkt_code REPLACE amount, account_name AS name
| STATS count(mkt_code), avg(amount) BY name
```

```
SEARCH source=<sourceIndex>
| DEDUP market_id
| EVAL category=replace(category, "-", ".")
| EVAL category=ltrim(category, "dvp.")
| LOOKUP bounce_category category AS category APPEND classification
```

```
SEARCH source=<sourceIndex>
| LOOKUP bounce_category category
```

commande d'analyse

La `parse` commande analyse un champ de texte avec une expression régulière et ajoute le résultat au résultat de la recherche.

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Syntaxe

Utilisez la syntaxe suivante :

```
parse <field> <pattern>
```

`field`

- Obligatoire.
- Le champ doit être un champ de texte.

`pattern`

- Chaîne obligatoire.
- Il s'agit du modèle d'expression régulière utilisé pour extraire de nouveaux champs d'un champ de texte donné.
- Si un nouveau nom de champ existe déjà, il remplacera le champ d'origine.

Expression régulière

Le modèle d'expression régulière est utilisé pour faire correspondre l'ensemble du champ de texte de chaque document avec le moteur Java Regex. Chaque groupe de capture nommé dans l'expression deviendra un nouveau `STRING` champ.

Exemple 1 : créer un nouveau champ

L'exemple montre comment créer un nouveau champ `host` pour chaque document. `host`sera le nom d'hôte indiqué après `@` dans le `email` champ. L'analyse d'un champ nul renverra une chaîne vide.

Requête PPL :

```
os> source=accounts | parse email '.+@(?<host>.+)' | fields email, host ;
```

```
fetched rows / total rows = 4/4
+----------------------+-------------+
| email                | host        |
|----------------------+-------------|
| jane_doe@example.com  | example.com |
| john_doe@example.net  | example.net |
| null                 |             |
| juan_li@example.org   | example.org |
+----------------------+-------------+
```

Exemple 2 : remplacer un champ existant

L'exemple montre comment remplacer le `address` champ existant en supprimant le numéro de rue.

Requête PPL :

```
os> source=accounts | parse address '\d+ (?<address>.+)' | fields address ;
fetched rows / total rows = 4/4
+------------------+
| address          |
|------------------|
| Example Lane     |
| Example Street   |
| Example Avenue   |
| Example Court    |
+------------------+
```

Exemple 3 : Filtrer et trier par champ analysé casté

L'exemple montre comment trier les numéros de rue supérieurs à 500 dans le `address` champ.

Requête PPL :

```
os> source=accounts | parse address '(?<streetNumber>\d+) (?<street>.+)' | where
 cast(streetNumber as int) > 500 | sort num(streetNumber) | fields streetNumber,
 street ;
fetched rows / total rows = 3/3
+----------------+----------------+
| streetNumber   | street         |
|----------------+----------------|
| ***            | Example Street |
| ***            | Example Avenue |
```

```
| 880            | Example Lane   |
+----------------+----------------+
```

Limites

La commande parse comporte quelques limites :

- Les champs définis par analyse ne peuvent pas être analysés à nouveau.

  La commande suivante ne fonctionnera pas :

  ```
  source=accounts | parse address '\d+ (?<street>.+)' | parse street '\w+ (?<road>\w+)'
  ```

- Les champs définis par parse ne peuvent pas être remplacés par d'autres commandes.

  `where`ne correspondra à aucun document car il `street` ne peut pas être remplacé :

  ```
  source=accounts | parse address '\d+ (?<street>.+)' | eval street='1' | where
   street='1' ;
  ```

- Le champ de texte utilisé par l'analyse ne peut pas être remplacé.

  `street`ne sera pas analysé avec succès car il `address` est remplacé :

  ```
  source=accounts | parse address '\d+ (?<street>.+)' | eval address='1' ;
  ```

- Les champs définis par parse ne peuvent pas être filtrés ou triés après les avoir utilisés dans la `stats` commande.

  `where`dans la commande suivante ne fonctionnera pas :

  ```
  source=accounts | parse email '.+@(?<host>.+)' | stats avg(age) by host | where
   host=pyrami.com ;
  ```

commande patterns

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

La `patterns` commande extrait les modèles de journalisation d'un champ de texte et ajoute les résultats au résultat de recherche. Le regroupement des journaux en fonction de leurs modèles facilite l'agrégation des statistiques provenant de gros volumes de données de journaux à des fins d'analyse et de résolution des problèmes.

Syntaxe

Utilisez la syntaxe suivante :

```
patterns [new_field=<new-field-name>] [pattern=<pattern>] <field>
```

new-field-name

- Chaîne facultative.
- Il s'agit du nom du nouveau champ pour les modèles extraits.
- L'argument par défaut est `patterns_field`.
- Si le nom existe déjà, il remplacera le champ d'origine.

pattern

- Chaîne facultative.
- Il s'agit du modèle regex des caractères qui doivent être filtrés du champ de texte.
- En cas d'absence, le modèle par défaut est constitué de caractères alphanumériques (`[a-zA-Z \d]`).

field

- Obligatoire.
- Le champ doit être un champ de texte.

Exemple 1 : créer le nouveau champ

L'exemple montre comment utiliser des extraits de ponctuation `email` pour chaque document. L'analyse d'un champ nul renverra une chaîne vide.

Requête PPL :

```
os> source=accounts | patterns email | fields email, patterns_field ;
```

```
fetched rows / total rows = 4/4
+----------------------+------------------+
| email                | patterns_field   |
|----------------------+------------------|
| jane_doe@example.com | @.               |
| john_doe@example.net | @.               |
| null                 |                  |
| juan_li@example.org  | @.               |
+----------------------+------------------+
```

Exemple 2 : Extraire les modèles de log

L'exemple montre comment extraire les ponctuations d'un champ de journal brut à l'aide des modèles par défaut.

Requête PPL :

```
os> source=apache | patterns message | fields message, patterns_field ;
fetched rows / total rows = 4/4
+-------------------------------------------------------------------------------------------------------------
+---------------------------------+
| message
                                       | patterns_field                |
|-------------------------------------------------------------------------------------------------------------
+---------------------------------|
| 177.95.8.74 - upton5450 [28/Sep/2022:10:15:57 -0700] "HEAD /e-business/mindshare
 HTTP/1.0" 404 19927                         | ... -  [//::: -] " /-/ /."      |
| ************ - pouros8756 [28/Sep/2022:10:15:57 -0700] "GET /architectures/
convergence/niches/mindshare HTTP/1.0" 100 28722 | ... -  [//::: -] " //// /."     |
| *************** - - [28/Sep/2022:10:15:57 -0700] "PATCH /strategize/out-of-the-box
 HTTP/1.0" 401 27439                        | ... - - [//::: -] " //--- /."   |
| ************** - - [28/Sep/2022:10:15:57 -0700] "POST /users HTTP/1.1" 301 9481
                                       | ... - - [//::: -] " / /."       |
+-------------------------------------------------------------------------------------------------------------
+---------------------------------+
```

Exemple 3 : Extraire des modèles de journal avec un modèle de regex personnalisé

L'exemple montre comment extraire les ponctuations d'un champ de journal brut à l'aide de modèles définis par l'utilisateur.

Requête PPL :

```
os> source=apache | patterns new_field='no_numbers' pattern='[0-9]' message | fields
 message, no_numbers ;
fetched rows / total rows = 4/4
+-----------------------------------------------------------------------------------------------------------
+-----------------------------------------------------------------------------
+
| message
                                      | no_numbers
                                      |
|-----------------------------------------------------------------------------------------------------------
+-----------------------------------------------------------------------------|
| 177.95.8.74 - upton5450 [28/Sep/2022:10:15:57 -0700] "HEAD /e-business/mindshare
 HTTP/1.0" 404 19927                       | ... - upton [/Sep/::: -] "HEAD /e-
business/mindshare HTTP/."                          |
| 127.45.152.6 - pouros8756 [28/Sep/2022:10:15:57 -0700] "GET /architectures/
convergence/niches/mindshare HTTP/1.0" 100 28722 | ... - pouros [/Sep/::: -] "GET /
architectures/convergence/niches/mindshare HTTP/."   |
| *************** - - [28/Sep/2022:10:15:57 -0700] "PATCH /strategize/out-of-the-box
 HTTP/1.0" 401 27439                      | ... - - [/Sep/::: -] "PATCH /strategize/
out-of-the-box HTTP/."                         |
| ************** - - [28/Sep/2022:10:15:57 -0700] "POST /users HTTP/1.1" 301 9481
                                      | ... - - [/Sep/::: -] "POST /users HTTP/."
                                      |
+-----------------------------------------------------------------------------------------------------------
+-----------------------------------------------------------------------------
+
```

Limitation

La commande patterns présente les mêmes limites que la commande parse.

commande rare

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la `rare` commande pour rechercher le tuple de valeurs le moins courant de tous les champs de la liste de champs.

> **Note**
>
> Un maximum de 10 résultats sont renvoyés pour chaque tuple distinct de valeurs des champs groupés.

Syntaxe

Utilisez la syntaxe suivante :

```
rare [N] <field-list> [by-clause] rare_approx [N] <field-list> [by-clause]
```

liste de champs

- Obligatoire.
- Liste de noms de champs séparés par des virgules.

clause

- Facultatif.
- Un ou plusieurs champs par lesquels regrouper les résultats.

N

- Le nombre de résultats à renvoyer.
- Par défaut: 10

rare_approx

- Le nombre approximatif de (n) champs rares en utilisant la cardinalité estimée par l'algorithme HyperLogLog ++.

Exemple 1 : trouver les valeurs les moins courantes dans un champ

L'exemple permet de trouver le sexe le moins courant de tous les comptes.

Requête PPL :

```
os> source=accounts | rare gender;
os> source=accounts | rare_approx 10 gender;
os> source=accounts | rare_approx gender;
fetched rows / total rows = 2/2
+----------+
| gender   |
|----------|
| F        |
| M        |
+----------+
```

Exemple 2 : Trouvez les valeurs les moins courantes organisées par sexe

L'exemple permet de trouver l'âge le moins courant de tous les comptes regroupés par sexe.

Requête PPL :

```
os> source=accounts | rare 5 age by gender;
os> source=accounts | rare_approx 5 age by gender;
fetched rows / total rows = 4/4
+----------+-------+
| gender   | age   |
|----------+-------|
| F        | 28    |
| M        | 32    |
| M        | 33    |
| M        | 36    |
+----------+-------+
```

renommer la commande

Utilisez la `rename` commande pour modifier le nom d'un ou de plusieurs champs dans les résultats de recherche.

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Syntaxe

Utilisez la syntaxe suivante :

```
rename <source-field> AS <target-field>["," <source-field> AS <target-field>]...
```

champ-source

- Obligatoire.
- Il s'agit du nom du champ que vous souhaitez renommer.

champ-cible

- Obligatoire.
- Il s'agit du nom que vous souhaitez renommer.

Exemple 1 : renommer un champ

Cet exemple montre comment renommer un seul champ.

Requête PPL :

```
os> source=accounts | rename account_number as an | fields an;
fetched rows / total rows = 4/4
+------+
| an   |
|------|
| 1    |
| 6    |
| 13   |
| 18   |
+------+
```

Exemple 2 : renommer plusieurs champs

Cet exemple montre comment renommer plusieurs champs.

Requête PPL :

```
os> source=accounts | rename account_number as an, employer as emp | fields an, emp;
fetched rows / total rows = 4/4
+------+---------+
| an   | emp     |
```

```
|------+---------|
| 1    | Pyrami  |
| 6    | Netagy  |
| 13   | Quility |
| 18   | null    |
+------+---------+
```

Limites

- Le remplacement d'un champ existant n'est pas pris en charge :

```
source=accounts | grok address '%{NUMBER} %{GREEDYDATA:address}' | fields address
```

commande de recherche

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la `search` commande pour récupérer des documents à partir d'un index. La `search` commande ne peut être utilisée que comme première commande dans une requête PPL.

Syntaxe

Utilisez la syntaxe suivante :

```
search source=[<remote-cluster>:]<index> [boolean-expression]
```

search

- Facultatif.
- Mots clés de recherche, qui peuvent être omis.

index

- Obligatoire.

- La commande de recherche doit spécifier l'index à partir duquel effectuer la requête.
- Le nom de l'index peut être préfixé par `<cluster name>:` pour les recherches entre clusters.

expression booléenne

- Facultatif.
- Toute expression dont l'évaluation correspond à une valeur booléenne.

Exemple 1 : récupérer toutes les données

L'exemple montre comment récupérer tout le document depuis l'index des comptes.

Requête PPL :

```
os> source=accounts;
+------------------+-------------+----------------------+-----------+----------
+--------+----------------+---------+-------+-----------------------+------------+
| account_number   | firstname   | address              | balance   | gender   | city
 | employer       | state   | age   | email                 | lastname   |
|------------------+-------------+----------------------+-----------+----------
+--------+----------------+---------+-------+-----------------------+------------|
| 1                | Jorge       | *** Any Lane         | 39225     | M        | Brogan
 | ExampleCorp    | IL      | 32    | jane_doe@example.com  | Souza      |
| 6                | John        | *** Example Street   | 5686      | M        | Dante
 | AnyCorp        | TN      | 36    | john_doe@example.com  | Doe        |
| 13               | Jane        | *** Any Street       | *****     | F        | Nogal
 | ExampleCompany | VA      | 28    | null                  | Doe        |
| 18               | Juan        | *** Example Court    | 4180      | M        | Orick
 | null           | MD      | 33    | juan_li@example.org   | Li         |
+------------------+-------------+----------------------+-----------+----------
+--------+----------------+---------+-------+-----------------------+------------+
```

Exemple 2 : récupérer des données avec une condition

L'exemple montre comment récupérer tout le document depuis l'index des comptes avec.

Requête PPL :

```
os> SEARCH source=accounts account_number=1 or gender="F";
```

```
+------------------+-------------+--------------------+-----------+----------+--------
+----------------+---------+-------+-------------------------+------------+
| account_number   | firstname   | address            | balance   | gender   | city   |
 employer       | state   | age   | email                 -  | lastname   |
|------------------+-------------+--------------------+-----------+----------+--------
+----------------+---------+-------+-------------------------+------------|
| 1                | Jorge       | *** Any Lane       | *****     | M        | Brogan |
 ExampleCorp    | IL      | 32    | jorge_souza@example.com | Souza      |
| 13               | Jane        | *** Any Street     | *****     | F        | Nogal  |
 ExampleCompany | VA      | 28    | null                    | Doe        |
+------------------+-------------+--------------------+-----------+----------+--------
+-----------------+---------+-------+-------------------------+------------+
```

commande de tri

Utilisez la `sort` commande pour trier les résultats de recherche selon les champs spécifiés.

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Syntaxe

Utilisez la syntaxe suivante :

```
sort <[+|-] sort-field>...
```

[+|-]

- Facultatif.
- Le signe plus [+] représente l'ordre croissant avec NULL/MISSING les valeurs en premier.
- Le signe moins [-] représente l'ordre décroissant avec NULL/MISSING les dernières valeurs.
- Par défaut : ordre croissant avec NULL/MISSING les valeurs en premier.

champ de tri

- Obligatoire.
- Champ utilisé pour le tri.

Exemple 1 : Trier selon un champ

L'exemple montre comment trier le document avec le champ d'âge par ordre croissant.

Requête PPL :

```
os> source=accounts | sort age | fields account_number, age;
fetched rows / total rows = 4/4
+------------------+-------+
| account_number   | age   |
|------------------+-------|
| 13               | 28    |
| 1                | 32    |
| 18               | 33    |
| 6                | 36    |
+------------------+-------+
```

Exemple 2 : trier selon un champ et renvoyer tous les résultats

L'exemple montre comment trier le document avec le champ d'âge par ordre croissant.

Requête PPL :

```
os> source=accounts | sort age | fields account_number, age;
fetched rows / total rows = 4/4
+------------------+-------+
| account_number   | age   |
|------------------+-------|
| 13               | 28    |
| 1                | 32    |
| 18               | 33    |
| 6                | 36    |
+------------------+-------+
```

Exemple 3 : Trier par un champ par ordre décroissant

L'exemple montre comment trier le document avec le champ d'âge par ordre décroissant.

Requête PPL :

```
os> source=accounts | sort - age | fields account_number, age;
fetched rows / total rows = 4/4
```

```
+------------------+-------+
| account_number   | age   |
|------------------+-------|
| 6                | 36    |
| 18               | 33    |
| 1                | 32    |
| 13               | 28    |
+------------------+-------+
```

Exemple 4 : Trier selon plusieurs champs

L'exemple montre comment trier le document avec le champ de genre par ordre croissant et le champ d'âge par ordre décroissant.

Requête PPL :

```
os> source=accounts | sort + gender, - age | fields account_number, gender, age;
fetched rows / total rows = 4/4
+------------------+----------+-------+
| account_number   | gender   | age   |
|------------------+----------+-------|
| 13               | F        | 28    |
| 6                | M        | 36    |
| 18               | M        | 33    |
| 1                | M        | 32    |
+------------------+----------+-------+
```

Exemple 5 : le tri par champ inclut une valeur nulle

L'exemple montre comment trier le champ employeur selon l'option par défaut (ordre croissant et valeur nulle en premier). Le résultat indique que la valeur nulle se trouve dans la première ligne.

Requête PPL :

```
os> source=accounts | sort employer | fields employer;
fetched rows / total rows = 4/4
+------------+
| employer   |
|------------|
| null       |
| AnyCompany |
| AnyCorp    |
```

```
| AnyOrgty   |
+------------+
```

commande stats

Utilisez la `stats` commande pour calculer l'agrégation à partir des résultats de recherche.

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Gestion des valeurs NULL/MANQUANTES

Gestion des valeurs NULL/MANQUANTES

Fonction	NULL	MANQUANT
COUNT	Non compté	Non compté
SUM	Ignorer	Ignorer
AVG	Ignorer	Ignorer
MAX	Ignorer	Ignorer
MIN	Ignorer	Ignorer

Syntaxe

Utilisez la syntaxe suivante :

```
stats <aggregation>... [by-clause]
```

agrégation

- Obligatoire.
- Fonction d'agrégation appliquée à un champ.

clause

- Facultatif.
- Syntaxe : `by [span-expression,] [field,]...`
- Spécifie les champs et les expressions permettant de regrouper les résultats de l'agrégation. La clause secondaire vous permet de regrouper vos résultats d'agrégation à l'aide de champs et d'expressions. Vous pouvez utiliser des fonctions scalaires, des fonctions d'agrégation et même des expressions d'intervalle pour diviser des champs spécifiques en compartiments à intervalles égaux.
- Par défaut : si non `<by-clause>` est spécifié, la commande stats renvoie une seule ligne représentant l'agrégation sur l'ensemble de résultats.

étendre l'expression

- Facultatif, au plus un.
- Syntaxe : `span(field_expr, interval_expr)`
- L'unité de l'expression d'intervalle est l'unité naturelle par défaut. Si le champ est de type date et heure et que l'intervalle est exprimé en date/time unités, vous spécifiez l'unité dans l'expression d'intervalle.
- Par exemple, diviser le `age` champ en seaux d'ici 10 ans, on dirait. `span(age, 10)` Pour diviser un champ d'horodatage en intervalles horaires, utilisez. `span(timestamp, 1h)`

Unités de temps disponibles

Unités d'intervalle d'intervalle
milliseconde (ms)
seconde (s)
minute (m, distinction majuscules et minuscules)
heure (h)
jour (d)
semaine (s)

Unités d'intervalle d'intervalle
mois (M, distinction majuscules et minuscules)
trimestre (q)
année (y)

Fonctions d'agrégation

COUNT

Renvoie le nombre d'expr dans les lignes récupérées par une instruction SELECT.

Exemple :

```
os> source=accounts | stats count();
fetched rows / total rows = 1/1
+-----------+
| count()   |
|-----------|
| 4         |
+-----------+
```

SUM

`SUM(expr)`À utiliser pour renvoyer la somme de expr.

exemple

```
os> source=accounts | stats sum(age) by gender;
fetched rows / total rows = 2/2
+------------+----------+
| sum(age)   | gender   |
|------------+----------|
| 28         | F        |
| 101        | M        |
+------------+----------+
```

AVG

`AVG(expr)`À utiliser pour renvoyer la valeur moyenne de expr.

exemple

```
os> source=accounts | stats avg(age) by gender;
fetched rows / total rows = 2/2
+--------------------+----------+
| avg(age)           | gender   |
|--------------------+----------|
| 28.0               | F        |
| 33.666666666666664 | M        |
+--------------------+----------+
```

MAX

`MAX(expr)`À utiliser pour renvoyer la valeur maximale de expr.

exemple

```
os> source=accounts | stats max(age);
fetched rows / total rows = 1/1
+------------+
| max(age)   |
|------------|
| 36         |
+------------+
```

MIN

`MIN(expr)`À utiliser pour renvoyer la valeur minimale de expr.

exemple

```
os> source=accounts | stats min(age);
fetched rows / total rows = 1/1
+------------+
| min(age)   |
|------------|
| 28         |
+------------+
```

STDDEV_SAMP

`STDDEV_SAMP(expr)`À utiliser pour renvoyer l'écart type de l'échantillon de expr.

Exemple :

```
os> source=accounts | stats stddev_samp(age);
fetched rows / total rows = 1/1
+--------------------+
| stddev_samp(age)   |
|--------------------|
| 3.304037933599835  |
+--------------------+
```

STDDEV_POP

`STDDEV_POP(expr)`À utiliser pour renvoyer l'écart type de population de expr.

Exemple :

```
os> source=accounts | stats stddev_pop(age);
fetched rows / total rows = 1/1
+--------------------+
| stddev_pop(age)    |
|--------------------|
| 2.**************** |
+--------------------+
```

PRENDRE

`TAKE(field [, size])`À utiliser pour renvoyer les valeurs d'origine d'un champ. Il ne fournit aucune garantie quant à l'ordre des valeurs.

field

- Obligatoire.
- Le champ doit être un champ de texte.

size

- Entier facultatif.
- Le nombre de valeurs doit être renvoyé.

- La valeur par défaut est 10.

exemple

```
os> source=accounts | stats take(firstname);
fetched rows / total rows = 1/1
+-----------------------------+
| take(firstname)             |
|-----------------------------|
| [Jane, Mary, Nikki, Juan    |
+-----------------------------+
```

PERCENTILE ou PERCENTILE_APPROX

Utilisez `PERCENTILE(expr, percent)` ou `PERCENTILE_APPROX(expr, percent)` pour renvoyer la valeur percentile approximative de expr au pourcentage spécifié.

pourcentage

- Le nombre doit être une constante comprise entre 0 et 100.

exemple

```
os> source=accounts | stats percentile(age, 90) by gender;
fetched rows / total rows = 2/2
+-----------------------+----------+
| percentile(age, 90)   | gender   |
|-----------------------+----------|
| 28                    | F        |
| 36                    | M        |
+-----------------------+----------+
```

Exemple 1 : calculer le nombre d'événements

L'exemple montre comment calculer le nombre d'événements dans les comptes.

```
os> source=accounts | stats count();
fetched rows / total rows = 1/1
+-----------+
```

```
| count()   |
|-----------|
| 4         |
+-----------+
```

Exemple 2 : calculer la moyenne d'un champ

L'exemple montre comment calculer l'âge moyen de tous les comptes.

```
os> source=accounts | stats avg(age);
fetched rows / total rows = 1/1
+------------+
| avg(age)   |
|------------|
| 32.25      |
+------------+
```

Exemple 3 : calculer la moyenne d'un champ par groupe

L'exemple montre comment calculer l'âge moyen de tous les comptes, regroupés par sexe.

```
os> source=accounts | stats avg(age) by gender;
fetched rows / total rows = 2/2
+--------------------+----------+
| avg(age)           | gender   |
|--------------------+----------|
| 28.0               | F        |
| 33.666666666666664 | M        |
+--------------------+----------+
```

Exemple 4 : calculer la moyenne, la somme et le nombre d'un champ par groupe

L'exemple montre comment calculer l'âge moyen, l'âge total et le nombre d'événements pour tous les comptes, regroupés par sexe.

```
os> source=accounts | stats avg(age), sum(age), count() by gender;
fetched rows / total rows = 2/2
+--------------------+------------+-----------+----------+
| avg(age)           | sum(age)   | count()   | gender   |
|--------------------+------------+-----------+----------|
| 28.0               | 28         | 1         | F        |
```

```
| 33.666666666666664 | 101        | 3         | M        |
+--------------------+------------+-----------+----------+
```

Exemple 5 : Calculer le maximum d'un champ

L'exemple calcule l'âge maximum pour tous les comptes.

```
os> source=accounts | stats max(age);
fetched rows / total rows = 1/1
+------------+
| max(age)   |
|------------|
| 36         |
+------------+
```

Exemple 6 : Calculer le maximum et le minimum d'un champ par groupe

L'exemple calcule les valeurs d'âge maximum et minimum pour tous les comptes, regroupés par sexe.

```
os> source=accounts | stats max(age), min(age) by gender;
fetched rows / total rows = 2/2
+------------+------------+----------+
| max(age)   | min(age)   | gender   |
|------------+------------+----------|
| 28         | 28         | F        |
| 36         | 32         | M        |
+------------+------------+----------+
```

Exemple 7 : Calculer le nombre distinct d'un champ

Pour obtenir le nombre de valeurs distinctes d'un champ, vous pouvez utiliser la fonction `DISTINCT_COUNT` (ou`DC`) au lieu de`COUNT`. L'exemple calcule à la fois le nombre et le nombre distinct de champs de genre de tous les comptes.

```
os> source=accounts | stats count(gender), distinct_count(gender);
fetched rows / total rows = 1/1
+-----------------+--------------------------+
| count(gender)   | distinct_count(gender)   |
|-----------------+--------------------------|
| 4               | 2                        |
```

```
+-----------------+--------------------------+
```

Exemple 8 : Calculer le nombre par intervalle

L'exemple obtient le décompte de l'âge par intervalle de 10 ans.

```
os> source=accounts | stats count(age) by span(age, 10) as age_span
fetched rows / total rows = 2/2
+--------------+------------+
| count(age)   | age_span   |
|--------------+------------|
| 1            | 20         |
| 3            | 30         |
+--------------+------------+
```

Exemple 9 : Calculez le nombre en fonction du sexe et de l'intervalle

Cet exemple compte les enregistrements regroupés par sexe et par tranche d'âge de 5 ans.

```
os> source=accounts | stats count() as cnt by span(age, 5) as age_span, gender
fetched rows / total rows = 3/3
+-------+------------+----------+
| cnt   | age_span   | gender   |
|-------+------------+----------|
| 1     | 25         | F        |
| 2     | 30         | M        |
| 1     | 35         | M        |
+-------+------------+----------+
```

L'expression span apparaît toujours comme première clé de regroupement, quel que soit l'ordre spécifié dans la commande.

```
os> source=accounts | stats count() as cnt by gender, span(age, 5) as age_span
fetched rows / total rows = 3/3
+-------+------------+----------+
| cnt   | age_span   | gender   |
|-------+------------+----------|
| 1     | 25         | F        |
| 2     | 30         | M        |
| 1     | 35         | M        |
+-------+------------+----------+
```

Exemple 10 : Calculez le nombre et obtenez une liste d'e-mails par sexe et par durée

L'exemple obtient le décompte de l'âge par intervalle de 10 ans et le groupe par sexe. De plus, pour chaque ligne, obtenez une liste d'au plus 5 e-mails.

```
os> source=accounts | stats count() as cnt, take(email, 5) by span(age, 5) as age_span,
 gender
fetched rows / total rows = 3/3
+-------+-----------------------------------------------------+------------+----------+
| cnt   | take(email, 5)                                      | age_span   | gender   |
|-------+-----------------------------------------------------+------------+----------|
| 1     | []                                                  | 25         | F        |
| 2     | [janedoe@anycompany.com,juanli@examplecompany.org] | 30         | M        |
| 1     | [marymajor@examplecorp.com]                         | 35         | M        |
+-------+-----------------------------------------------------+------------+----------+
```

Exemple 11 : Calculer le percentile d'un champ

L'exemple montre comment calculer le 90e percentile d'âge de tous les comptes.

```
os> source=accounts | stats percentile(age, 90);
fetched rows / total rows = 1/1
+-----------------------+
| percentile(age, 90)   |
|-----------------------|
| 36                    |
+-----------------------+
```

Exemple 12 : Calculer le percentile d'un champ par groupe

L'exemple montre comment calculer le 90e percentile d'âge de tous les comptes regroupés par sexe.

```
os> source=accounts | stats percentile(age, 90) by gender;
fetched rows / total rows = 2/2
+-----------------------+----------+
| percentile(age, 90)   | gender   |
|-----------------------+----------|
| 28                    | F        |
| 36                    | M        |
+-----------------------+----------+
```

Exemple 13 : Calculer le percentile en fonction du sexe et de l'intervalle

L'exemple obtient le 90e percentile d'âge par intervalle de 10 ans et le groupe par sexe.

```
os> source=accounts | stats percentile(age, 90) as p90 by span(age, 10) as age_span,
 gender
fetched rows / total rows = 2/2
+-------+------------+----------+
| p90   | age_span   | gender   |
|-------+------------+----------|
| 28    | 20         | F        |
| 36    | 30         | M        |
+-------+------------+----------+
```

```
- `source = table | stats avg(a) `
- `source = table | where a < 50 | stats avg(c) `
- `source = table | stats max(c) by b`
- `source = table | stats count(c) by b | head 5`
- `source = table | stats distinct_count(c)`
- `source = table | stats stddev_samp(c)`
- `source = table | stats stddev_pop(c)`
- `source = table | stats percentile(c, 90)`
- `source = table | stats percentile_approx(c, 99)`
```

Agrégations avec span

```
- `source = table  | stats count(a) by span(a, 10) as a_span`
- `source = table  | stats sum(age) by span(age, 5) as age_span | head 2`
- `source = table  | stats avg(age) by span(age, 20) as age_span, country  | sort -
 age_span |  head 2`
```

Agrégations avec intervalle de fenêtre temporelle (fonction de fenêtrage Tumble)

```
- `source = table | stats sum(productsAmount) by span(transactionDate, 1d) as age_date
 | sort age_date`
- `source = table | stats sum(productsAmount) by span(transactionDate, 1w) as age_date,
 productId`
```

Les agrégations sont regroupées par plusieurs niveaux

```
- `source = table | stats avg(age) as avg_state_age by country, state | stats
 avg(avg_state_age) as avg_country_age by country`
- `source = table | stats avg(age) as avg_city_age by country, state, city | eval
 new_avg_city_age = avg_city_age - 1 | stats avg(new_avg_city_age) as avg_state_age
 by country, state | where avg_state_age > 18 | stats avg(avg_state_age) as
 avg_adult_country_age by country`
```

commande de sous-requête

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la `subquery` commande pour exécuter des requêtes complexes et imbriquées dans vos instructions PPL (Piped Processing Language).

```
source=logs | where field in [ subquery source=events | where condition | fields
 field ]
```

Dans cet exemple, la recherche principale (`source=logs`) est filtrée en fonction des résultats de la sous-requête (`source=events`).

La commande subquery prend en charge plusieurs niveaux d'imbrication pour des analyses de données complexes.

Exemple de sous-requête imbriquée

```
source=logs | where id in [ subquery source=users | where user in [ subquery
 source=actions | where action="login" | fields user] | fields uid ]
```

InSubquery Usage

- `source = outer | where a in [ source = inner | fields b ]`
- `source = outer | where (a) in [ source = inner | fields b ]`
- `source = outer | where (a,b,c) in [ source = inner | fields d,e,f ]`
- `source = outer | where a not in [ source = inner | fields b ]`

- `source = outer | where (a) not in [ source = inner | fields b ]`
- `source = outer | where (a,b,c) not in [ source = inner | fields d,e,f ]`
- `source = outer a in [ source = inner | fields b ]`(filtrage de recherche avec sous-requête)
- `source = outer a not in [ source = inner | fields b ]`(filtrage de recherche avec sous-requête)
- `source = outer | where a in [ source = inner1 | where b not in [ source = inner2 | fields c ] | fields b ]`(imbriqué)
- `source = table1 | inner join left = l right = r on l.a = r.a AND r.a in [ source = inner | fields d ] | fields l.a, r.a, b, c`(en tant que filtre de jointure)

Exemples de migration SQL avec PPL dans la sous-requête

TPC-H Q4 (sous-requête intégrée avec agrégation)

```
select
  o_orderpriority,
  count(*) as order_count
from
  orders
where
  o_orderdate >= date '1993-07-01'
  and o_orderdate < date '1993-07-01' + interval '3' month
  and o_orderkey in (
    select
      l_orderkey
    from
      lineitem
    where l_commitdate < l_receiptdate
  )
group by
  o_orderpriority
order by
  o_orderpriority
```

Réécrit par requête PPL InSubquery :

```
source = orders
```

```
| where o_orderdate >= "1993-07-01" and o_orderdate < "1993-10-01" and o_orderkey IN
  [ source = lineitem
    | where l_commitdate < l_receiptdate
    | fields l_orderkey
  ]
| stats count(1) as order_count by o_orderpriority
| sort o_orderpriority
| fields o_orderpriority, order_count
```

TPC-H Q20 (sous-requête imbriquée)

```
select
  s_name,
  s_address
from
  supplier,
  nation
where
  s_suppkey in (
    select
      ps_suppkey
    from
      partsupp
    where
      ps_partkey in (
        select
          p_partkey
        from
          part
        where
          p_name like 'forest%'
      )
  )
  and s_nationkey = n_nationkey
  and n_name = 'CANADA'
order by
  s_name
```

Réécrit par requête PPL InSubquery :

```
source = supplier
| where s_suppkey IN [
    source = partsupp
```

```
    | where ps_partkey IN [
        source = part
        | where like(p_name, "forest%")
        | fields p_partkey
      ]
    | fields ps_suppkey
  ]
| inner join left=l right=r on s_nationkey = n_nationkey and n_name = 'CANADA'
  nation
| sort s_name
```

ExistsSubquery utilisation

Hypothèses :a, b sont des champs de table externec,, d sont des champs de table internee, f sont des champs de table inner2.

- `source = outer | where exists [ source = inner | where a = c ]`
- `source = outer | where not exists [ source = inner | where a = c ]`
- `source = outer | where exists [ source = inner | where a = c and b = d ]`
- `source = outer | where not exists [ source = inner | where a = c and b = d ]`
- `source = outer exists [ source = inner | where a = c ]`(filtrage de recherche avec sous-requête)
- `source = outer not exists [ source = inner | where a = c ]`(filtrage de recherche avec sous-requête)
- `source = table as t1 exists [ source = table as t2 | where t1.a = t2.a ]`(l'alias de table est utile dans la sous-requête exists)
- `source = outer | where exists [ source = inner1 | where a = c and exists [ source = inner2 | where c = e ] ]`(imbriqué)
- `source = outer | where exists [ source = inner1 | where a = c | where exists [ source = inner2 | where c = e ] ]`(imbriqué)
- `source = outer | where exists [ source = inner | where c > 10 ]`(il existe une corrélation non corrélée)
- `source = outer | where not exists [ source = inner | where c > 10 ]`(il existe une corrélation non corrélée)
- `source = outer | where exists [ source = inner ] | eval l = "nonEmpty" | fields l`(il existe une corrélation spéciale non corrélée)

ScalarSubquery utilisation

Hypothèses :a, b les champs de la table sont-ils extérieursc, d sont-ils les champs de la table internese, f sont-ils des champs de la table imbriqués

Sous-requête scalaire non corrélée

Dans Select :

- `source = outer | eval m = [ source = inner | stats max(c) ] | fields m, a`
- `source = outer | eval m = [ source = inner | stats max(c) ] + b | fields m, a`

Où :

- `source = outer | where a > [ source = inner | stats min(c) ] | fields a`

Dans le filtre de recherche :

- `source = outer a > [ source = inner | stats min(c) ] | fields a`

Sous-requête scalaire corrélée

Dans Select :

- `source = outer | eval m = [ source = inner | where outer.b = inner.d | stats max(c) ] | fields m, a`
- `source = outer | eval m = [ source = inner | where b = d | stats max(c) ] | fields m, a`
- `source = outer | eval m = [ source = inner | where outer.b > inner.d | stats max(c) ] | fields m, a`

Où :

- `source = outer | where a = [ source = inner | where outer.b = inner.d | stats max(c) ]`
- `source = outer | where a = [ source = inner | where b = d | stats max(c) ]`

- `source = outer | where [ source = inner | where outer.b = inner.d OR inner.d = 1 | stats count() ] > 0 | fields a`

Dans le filtre de recherche :

- `source = outer a = [ source = inner | where b = d | stats max(c) ]`
- `source = outer [ source = inner | where outer.b = inner.d OR inner.d = 1 | stats count() ] > 0 | fields a`

Sous-requête scalaire imbriquée

- `source = outer | where a = [ source = inner | stats max(c) | sort c ] OR b = [ source = inner | where c = 1 | stats min(d) | sort d ]`
- `source = outer | where a = [ source = inner | where c = [ source = nested | stats max(e) by f | sort f ] | stats max(d) by c | sort c | head 1 ]`

Sous-requête (Relation)

`InSubquery`, `ExistsSubquery` et `ScalarSubquery` sont toutes des expressions de sous-requêtes. Mais il ne `RelationSubquery` s'agit pas d'une expression de sous-requête, c'est un plan de sous-requête couramment utilisé dans les clauses Join ou From.

- `source = table1 | join left = l right = r [ source = table2 | where d > 10 | head 5 ]`(sous-requête dans le côté droit de la jointure)
- `source = [ source = table1 | join left = l right = r [ source = table2 | where d > 10 | head 5 ] | stats count(a) by b ] as outer | head 1`

Contexte supplémentaire

`InSubqueryExistsSubquery`, et `ScalarSubquery` sont des expressions de sous-requêtes couramment utilisées dans les `where` clauses et les filtres de recherche.

Commande où :

```
| where <boolean expression> | ...
```

Filtre de recherche :

```
search source=* <boolean expression> | ...
```

Une expression de sous-requête peut être utilisée dans une expression booléenne :

```
| where orders.order_id in [ source=returns | where return_reason="damaged" | field
 order_id ]
```

`orders.order_id in [ source=... ]`C'est un`<boolean expression>`.

En général, nous appelons ce type de clause de sous-requête l'`InSubquery`expression. C'est un`<boolean expression>`.

Sous-requête avec différents types de jointure

Exemple d'utilisation de `ScalarSubquery` :

```
source=employees
| join source=sales on employees.employee_id = sales.employee_id
| where sales.sale_amount > [ source=targets | where target_met="true" | fields
 target_value ]
```

Contrairement à InSubquery, ExistsSubquery, et ScalarSubquery, a n' RelationSubqueryest pas une expression de sous-requête. Il s'agit plutôt d'un plan de sous-requête.

```
SEARCH source=customer
| FIELDS c_custkey
| LEFT OUTER JOIN left = c, right = o ON c.c_custkey = o.o_custkey
   [
      SEARCH source=orders
      | WHERE o_comment NOT LIKE '%unusual%packages%'
      | FIELDS o_orderkey, o_custkey
   ]
| STATS ...
```

commande supérieure

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la `top` commande pour rechercher le tuple de valeurs le plus courant de tous les champs de la liste de champs.

Syntaxe

Utilisez la syntaxe suivante :

```
top [N] <field-list> [by-clause] top_approx [N] <field-list> [by-clause]
```

N

- Le nombre de résultats à renvoyer.
- Par défaut: 10

liste de champs

- Obligatoire.
- Liste de noms de champs séparés par des virgules.

clause

- Facultatif.
- Un ou plusieurs champs par lesquels regrouper les résultats.

top_approx

- Un décompte approximatif des (n) premiers champs en utilisant la cardinalité estimée par l'algorithme HyperLogLog ++.

Exemple 1 : trouver les valeurs les plus courantes dans un champ

L'exemple permet de trouver le sexe le plus courant pour tous les comptes.

Requête PPL :

```
os> source=accounts | top gender;
os> source=accounts | top_approx gender;
fetched rows / total rows = 2/2
+----------+
```

```
| gender   |
|----------|
| M        |
| F        |
+----------+
```

Exemple 2 : Rechercher les valeurs les plus courantes dans un champ (limité à 1)

L'exemple permet de trouver le sexe le plus courant pour tous les comptes.

Requête PPL :

```
os> source=accounts | top_approx 1 gender;
fetched rows / total rows = 1/1
+----------+
| gender   |
|----------|
| M        |
+----------+
```

Exemple 3 : trouver les valeurs les plus courantes, groupées par sexe

L'exemple permet de trouver l'âge le plus courant pour tous les comptes, groupés par sexe.

Requête PPL :

```
os> source=accounts | top 1 age by gender;
os> source=accounts | top_approx 1 age by gender;
fetched rows / total rows = 2/2
+----------+-------+
| gender   | age   |
|----------+-------|
| F        | 28    |
| M        | 32    |
+----------+-------+
```

commande trendline

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la `trendline` commande pour calculer les moyennes mobiles des champs.

Syntaxe

Utilisez la syntaxe suivante

```
TRENDLINE [sort <[+|-] sort-field>] SMA(number-of-datapoints, field) [AS alias]
 [SMA(number-of-datapoints, field) [AS alias]]...
```

[+|-]

- Facultatif.
- Le signe plus [+] représente l'ordre croissant avec NULL/MISSING les valeurs en premier.
- Le signe moins [-] représente l'ordre décroissant avec NULL/MISSING les dernières valeurs.
- Par défaut : ordre croissant avec NULL/MISSING les valeurs en premier.

champ de tri

- Obligatoire lorsque le tri est utilisé.
- Champ utilisé pour le tri.

number-of-datapoints

- Obligatoire.
- Le nombre de points de données qui calculent la moyenne mobile.
- Doit être supérieur à zéro.

field

- Obligatoire.
- Nom du champ pour lequel la moyenne mobile doit être calculée.

alias

- Facultatif.
- Nom de la colonne résultante contenant la moyenne mobile.

Seul le type de moyenne mobile simple (SMA) est pris en charge. Il est calculé comme suit :

```
f[i]: The value of field 'f' in the i-th data-point
n: The number of data-points in the moving window (period)
t: The current time index

SMA(t) = (1/n) * Σ(f[i]), where i = t-n+1 to t
```

Exemple 1 : Calculer une moyenne mobile simple pour une série chronologique de températures

L'exemple calcule la moyenne mobile simple des températures à l'aide de deux points de données.

Requête PPL :

```
os> source=t | trendline sma(2, temperature) as temp_trend;
fetched rows / total rows = 5/5
+-----------+---------+--------------------+----------+
|temperature|device-id|           timestamp|temp_trend|
+-----------+---------+--------------------+----------+
|         12|     1492|2023-04-06 17:07:...|      NULL|
|         12|     1492|2023-04-06 17:07:...|      12.0|
|         13|      256|2023-04-06 17:07:...|      12.5|
|         14|      257|2023-04-06 17:07:...|      13.5|
|         15|      258|2023-04-06 17:07:...|      14.5|
+-----------+---------+--------------------+----------+
```

Exemple 2 : Calculez des moyennes mobiles simples pour une série chronologique de températures avec tri

L'exemple calcule deux moyennes mobiles simples sur les températures en utilisant deux et trois points de données triés en ordre décroissant par identifiant d'appareil.

Requête PPL :

```
os> source=t | trendline sort - device-id sma(2, temperature) as temp_trend_2 sma(3,
 temperature) as temp_trend_3;
fetched rows / total rows = 5/5
+-----------+---------+--------------------+------------+------------------+
|temperature|device-id|           timestamp|temp_trend_2|      temp_trend_3|
+-----------+---------+--------------------+------------+------------------+
|         15|      258|2023-04-06 17:07:...|        NULL|              NULL|
```

```
|         14|      257|2023-04-06 17:07:...|        14.5|               NULL|
|         13|      256|2023-04-06 17:07:...|        13.5|               14.0|
|         12|     1492|2023-04-06 17:07:...|        12.5|               13.0|
|         12|     1492|2023-04-06 17:07:...|        12.0|12.333333333333334|
+-----------+---------+--------------------+------------+------------------+
```

où commande

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

La `where` commande utilise une expression booléenne pour filtrer le résultat de la recherche. Il ne renvoie le résultat que lorsque l'expression booléenne est évaluée à true.

Syntaxe

Utilisez la syntaxe suivante :

```
where <boolean-expression>
```

expression booléenne

- Facultatif.
- Toute expression pouvant être évaluée à une valeur booléenne.

Exemple 1 : Filtrer un ensemble de résultats avec une condition

L'exemple montre comment extraire des documents de l'index des comptes qui répondent à des conditions spécifiques.

Requête PPL :

```
os> source=accounts | where account_number=1 or gender="F" | fields account_number,
 gender;
fetched rows / total rows = 2/2
+------------------+----------+
```

```
| account_number   | gender   |
|------------------+----------|
| 1                | M        |
| 13               | F        |
+------------------+----------+
```

Exemples supplémentaires

Filtres avec conditions logiques

- `source = table | where c = 'test' AND a = 1 | fields a,b,c`
- `source = table | where c != 'test' OR a > 1 | fields a,b,c | head 1`
- `source = table | where c = 'test' NOT a > 1 | fields a,b,c`
- `source = table | where a = 1 | fields a,b,c`
- `source = table | where a >= 1 | fields a,b,c`
- `source = table | where a < 1 | fields a,b,c`
- `source = table | where b != 'test' | fields a,b,c`
- `source = table | where c = 'test' | fields a,b,c | head 3`
- `source = table | where ispresent(b)`
- `source = table | where isnull(coalesce(a, b)) | fields a,b,c | head 3`
- `source = table | where isempty(a)`
- `source = table | where isblank(a)`
- `source = table | where case(length(a) > 6, 'True' else 'False') = 'True'`
- `source = table | where a between 1 and 4`- Remarque : Cela renvoie a >= 1 et a <= 4, c'est-à-dire [1, 4]
- `source = table | where b not between '2024-09-10' and '2025-09-10'`- Remarque : cela renvoie b >= '**********' et b <= '2025-09-10'
- `source = table | where cidrmatch(ip, '***********/24')`
- `source = table | where cidrmatch(ipv6, '2003:db8::/32')`

```
source = table | eval status_category =
    case(a >= 200 AND a < 300, 'Success',
```

```
    a >= 300 AND a < 400, 'Redirection',
    a >= 400 AND a < 500, 'Client Error',
    a >= 500, 'Server Error'
    else 'Incorrect HTTP status code')
    | where case(a >= 200 AND a < 300, 'Success',
    a >= 300 AND a < 400, 'Redirection',
    a >= 400 AND a < 500, 'Client Error',
    a >= 500, 'Server Error'
    else 'Incorrect HTTP status code'
    ) = 'Incorrect HTTP status code'
```

```
source = table
    | eval factor = case(a > 15, a - 14, isnull(b), a - 7, a < 3, a + 1 else 1)
    | where case(factor = 2, 'even', factor = 4, 'even', factor = 6, 'even', factor =
 8, 'even' else 'odd') = 'even'
    |  stats count() by factor
```

résumé du champ

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette commande PPL, consultez. the section called “Commandes”

Utilisez la `fieldsummary` commande pour calculer les statistiques de base pour chaque champ (nombre, nombre distinct, min, max, avg, stddev, moyenne) et déterminez le type de données de chaque champ. Cette commande peut être utilisée avec n'importe quel canal précédent et en tiendra compte.

Syntaxe

Utilisez la syntaxe suivante. Pour les cas d'utilisation des CloudWatch journaux, un seul champ d'une requête est pris en charge.

```
... | fieldsummary <field-list> (nulls=true/false)
```

inclure des champs

- Liste de toutes les colonnes à collecter avec des statistiques dans un ensemble de résultats unifié.

Valeurs null

- Facultatif.
- Si ce paramètre est défini sur true, incluez les valeurs nulles dans les calculs d'agrégation (remplacez null par zéro pour les valeurs numériques).

Exemple 1

Requête PPL :

```
os> source = t | where status_code != 200 | fieldsummary includefields= status_code
 nulls=true
+------------------+-------------+-----------+-----------+-----------+------------
+------------+-----------+----------------+
| Fields           | COUNT       | COUNT_DISTINCT    |  MIN  |  MAX   |  AVG   |  MEAN
  |        STDDEV       | NUlls | TYPEOF |
|------------------+-------------+-----------+-----------+-----------+------------
+------------+-----------+----------------|
| "status_code"    |      2      |         2         | 301   |   403  |  352.0 |  352.0
  |  72.12489168102785  |  0    | "int"  |
+------------------+-------------+-----------+-----------+-----------+------------
+------------+-----------+----------------|
```

Exemple 2

Requête PPL :

```
os> source = t | fieldsummary includefields= id, status_code, request_path nulls=true
+------------------+-------------+-----------+-----------+-----------+------------
+------------+-----------+----------------+
| Fields           | COUNT       | COUNT_DISTINCT    |  MIN  |  MAX   |  AVG   |  MEAN
  |        STDDEV       | NUlls | TYPEOF |
|------------------+-------------+-----------+-----------+-----------+------------
+------------+-----------+----------------|
|       "id"       |      6      |         6         | 1     |   6    |  3.5   |   3.5
 |  1.8708286933869707  |  0    | "int"  |
+------------------+-------------+-----------+-----------+-----------+------------
+------------+-----------+----------------|
| "status_code"    |      4      |         3         | 200   |   403  |  184.0 |  184.0
  |  161.16699413961905 |  2    | "int"  |
+------------------+-------------+-----------+-----------+-----------+------------
+------------+-----------+----------------|
```

```
| "request_path"   |      2      |         2         | /about| /home   |  0.0    |  0.0
    |      0            |  2    |"string"|
+-----------------+------------+-----------+-----------+-----------+-----------
+-----------+-----------+---------------|
```

commande d'extension

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette fonction PPL, consultez. the section called “Fonctions”

Utilisez la `expand` commande pour aplatir un champ de type Array <Any>ou Map<Any>, en produisant des lignes individuelles pour chaque élément ou paire clé-valeur.

Syntaxe

Utilisez la syntaxe suivante :

```
expand <field> [As alias]
```

field

- Le champ à étendre (exploser).
- Le champ doit être d'un type compatible.

alias

- Facultatif.
- Le nom à utiliser à la place du nom de champ d'origine.

Directives d'utilisation

La commande expand produit une ligne pour chaque élément du tableau ou du champ de carte spécifié, où :

- Les éléments du tableau deviennent des lignes individuelles.

- Les paires clé-valeur de la carte sont divisées en lignes distinctes, chaque valeur-clé étant représentée par une ligne.
- Lorsqu'un alias est fourni, les valeurs éclatées sont représentées sous l'alias au lieu du nom du champ d'origine.

Vous pouvez utiliser cette commande en combinaison avec d'autres commandes, telles que stats, eval et parse, pour manipuler ou extraire des données après l'expansion.

Exemples

- `source = table | expand employee | stats max(salary) as max by state, company`
- `source = table | expand employee as worker | stats max(salary) as max by state, company`
- `source = table | expand employee as worker | eval bonus = salary * 3 | fields worker, bonus`
- `source = table | expand employee | parse description '(?<email>.+@.+)' | fields employee, email`
- `source = table | eval array=json_array(1, 2, 3) | expand array as uid | fields name, occupation, uid`
- `source = table | expand multi_valueA as multiA | expand multi_valueB as multiB`

Vous pouvez utiliser la commande expand en combinaison avec d'autres commandes telles que eval, stats, etc. L'utilisation de plusieurs commandes d'extension créera un produit cartésien de tous les éléments internes de chaque tableau composite ou carte.

Requête SQL push-down efficace

La commande expand est traduite en une opération SQL équivalente à l'aide de LATERAL VIEW explode, ce qui permet une explosion efficace de tableaux ou de cartes au niveau de la requête SQL.

```
SELECT customer exploded_productId
FROM table
LATERAL VIEW explode(productId) AS exploded_productId
```

La commande explode offre les fonctionnalités suivantes :

- Il s'agit d'une opération de colonne qui renvoie une nouvelle colonne.
- Il crée une nouvelle ligne pour chaque élément de la colonne éclatée.
- Les valeurs nulles internes sont ignorées dans le cadre du champ éclaté (aucune ligne ne correspond created/exploded à une valeur nulle).

Fonctions PPL

Rubriques

- Fonctions de condition PPL
- Fonctions de hachage cryptographiques PPL
- Fonctions de date et d'heure PPL
- Expressions PPL
- Fonctions d'adresse IP PPL
- Fonctions JSON PPL
- Fonctions Lambda PPL
- Fonctions mathématiques PPL
- Fonctions de chaîne PPL
- Fonctions de conversion de type PPL

Fonctions de condition PPL

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette fonction PPL, consultez. the section called “Fonctions”

ISNULL

Description : `isnull(field)` renvoie vrai si le champ est nul.

Type d'argument :

- Tous les types de données pris en charge.

Type de retour :

- BOOLEAN

Exemple :

```
os> source=accounts | eval result = isnull(employer) | fields result, employer,
 firstname
fetched rows / total rows = 4/4
+----------+-------------+-------------+
| result   | employer    | firstname   |
|----------+-------------+-------------|
| False    | AnyCompany  | Mary        |
| False    | ExampleCorp | Jane        |
| False    | ExampleOrg  | Nikki       |
| True     | null        | Juan        |
+----------+-------------+-------------+
```

N'EST PAS NUL

Description : `isnotnull(field)` renvoie vrai si le champ n'est pas nul.

Type d'argument :

- Tous les types de données pris en charge.

Type de retour :

- BOOLEAN

Exemple :

```
os> source=accounts | where not isnotnull(employer) | fields account_number, employer
fetched rows / total rows = 1/1
+------------------+------------+
| account_number   | employer   |
|------------------+------------|
| 18               | null       |
+------------------+------------+
```

EXISTS

Exemple :

```
os> source=accounts | where exists(email) | fields account_number, email
fetched rows / total rows = 1/1
```

SI NUL

Description : `ifnull(field1, field2)` renvoie `field2` si la valeur `field1` est nulle.

Type d'argument :

- Tous les types de données pris en charge.
- Si les deux paramètres sont de types différents, la fonction échouera à la vérification sémantique.

Type de retour :

- N’importe quel compte

Exemple :

```
os> source=accounts | eval result = ifnull(employer, 'default') | fields result,
 employer, firstname
fetched rows / total rows = 4/4
+------------+------------+-------------+
| result     | employer   | firstname   |
|------------+------------+-------------|
| AnyCompany | AnyCompany | Mary        |
| ExampleCorp| ExampleCorp| Jane        |
| ExampleOrg | ExampleOrg | Nikki       |
| default    | null       | Juan        |
+------------+------------+-------------+
```

NULLIF

Description : `nullif(field1, field2)` renvoie null si deux paramètres sont identiques, sinon renvoie field1.

Type d'argument :

- Tous les types de données pris en charge.
- Si les deux paramètres sont de types différents, la fonction échouera à la vérification sémantique.

Type de retour :

- N'importe quel compte

Exemple :

```
os> source=accounts | eval result = nullif(employer, 'AnyCompany') | fields result,
 employer, firstname
fetched rows / total rows = 4/4
+----------------+----------------+-------------+
| result         | employer       | firstname   |
|----------------+----------------+-------------|
| null           | AnyCompany     | Mary        |
| ExampleCorp    | ExampleCorp    | Jane        |
| ExampleOrg     | ExampleOrg     | Nikki       |
| null           | null           | Juan        |
+----------------+----------------+-------------+
```

IF

Description : `if(condition, expr1, expr2)` renvoie `expr1` si la condition est vraie, sinon elle est renvoyée`expr2`.

Type d'argument :

- Tous les types de données pris en charge.
- Si les deux paramètres sont de types différents, la fonction échouera à la vérification sémantique.

Type de retour :

- N'importe quel compte

Exemple :

```
os> source=accounts | eval result = if(true, firstname, lastname) | fields result,
 firstname, lastname
fetched rows / total rows = 4/4
+----------+-------------+----------+
| result   | firstname | lastname   |
|----------+-------------+----------|
| Jane     | Jane      | Doe        |
| Mary     | Mary      | Major      |
| Pat      | Pat       | Candella   |
| Dale     | Jorge     | Souza      |
+----------+-----------+------------+

os> source=accounts | eval result = if(false, firstname, lastname) | fields result,
 firstname, lastname
fetched rows / total rows = 4/4
+----------+-------------+------------+
| result   | firstname   | lastname   |
|----------+-------------+------------|
| Doe      | Jane        | Doe        |
| Major    | Mary        | Major      |
| Candella | Pat         | Candella   |
| Souza    | Jorge       | Souza      |
+----------+-------------+------------+

os> source=accounts | eval is_vip = if(age > 30 AND isnotnull(employer), true, false) |
 fields is_vip, firstname, lastname
fetched rows / total rows = 4/4
+----------+-------------+------------+
| is_vip   | firstname   | lastname   |
|----------+-------------+------------|
| True     | Jane        | Doe        |
| True     | Mary        | Major      |
| False    | Pat         | Candella   |
| False    | Jorge       | Souza      |
+----------+-------------+------------+
```

Fonctions de hachage cryptographiques PPL

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette fonction PPL, consultez. the section called “Fonctions”

MD5

MD5 calcule le MD5 condensé et renvoie la valeur sous forme de chaîne hexadécimale de 32 caractères.

Utilisation : `md5('hello')`

Type d'argument :

- CHAÎNE

Type de retour :

- CHAÎNE

Exemple :

```
os> source=people | eval `MD5('hello')` = MD5('hello') | fields `MD5('hello')`
fetched rows / total rows = 1/1
+----------------------------------+
| MD5('hello')                     |
|----------------------------------|
| <32 character hex string>        |
+----------------------------------+
```

SHA1

SHA1 renvoie le résultat de la chaîne hexadécimale de SHA-1.

Utilisation : `sha1('hello')`

Type d'argument :

- CHAÎNE

Type de retour :

- CHAÎNE

Exemple :

```
os> source=people | eval `SHA1('hello')` = SHA1('hello') | fields `SHA1('hello')`
fetched rows / total rows = 1/1
+------------------------------------------+
| SHA1('hello')                            |
|------------------------------------------|
| <40-character SHA-1 hash result>         |
+------------------------------------------+
```

SHA2

SHA2 renvoie le résultat sous forme de chaîne hexadécimale de la famille de fonctions de hachage SHA-2 (SHA-224, SHA-256, SHA-384 et SHA-512). Le NumBits indique la longueur de bit souhaitée du résultat, qui doit avoir une valeur de 224, 256, 384, 512

Utilisation :

- `sha2('hello',256)`
- `sha2('hello',512)`

Type d'argument :

- CHAÎNE, ENTIER

Type de retour :

- CHAÎNE

Exemple :

```
os> source=people | eval `SHA2('hello',256)` = SHA2('hello',256) | fields
 `SHA2('hello',256)`
fetched rows / total rows = 1/1
+------------------------------------------------------------------+
| SHA2('hello',256)                                                |
|------------------------------------------------------------------|
| <64-character SHA-256 hash result>                               |
+------------------------------------------------------------------+

os> source=people | eval `SHA2('hello',512)` = SHA2('hello',512) | fields
 `SHA2('hello',512)`
```

```
fetched rows / total rows = 1/1
+------------------------------------------------------------------+
| SHA2('hello',512)                                                |
                                          |
|------------------------------------------------------------------|
| <128-character SHA-512 hash result>                              |
+------------------------------------------------------------------+
```

Fonctions de date et d'heure PPL

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette fonction PPL, consultez. the section called “Fonctions”

DAY

Utilisation : `DAY(date)` extrait le jour du mois pour une date comprise entre 1 et 31.

Type d'argument : STRING/DATE/TIMESTAMP

Type de retour : INTEGER

Synonymes :`DAYOFMONTH`, `DAY_OF_MONTH`

Exemple :

```
os> source=people | eval `DAY(DATE('2020-08-26'))` = DAY(DATE('2020-08-26')) | fields
 `DAY(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+---------------------------+
| DAY(DATE('2020-08-26'))   |
|---------------------------|
| 26                        |
+---------------------------+
```

DAYOFMONTH

Utilisation : `DAYOFMONTH(date)` extrait le jour du mois pour une date comprise entre 1 et 31.

Type d'argument : STRING/DATE/TIMESTAMP

Type de retour : INTEGER

Synonymes :DAY, DAY_OF_MONTH

Exemple :

```
os> source=people | eval `DAYOFMONTH(DATE('2020-08-26'))` =
 DAYOFMONTH(DATE('2020-08-26')) | fields `DAYOFMONTH(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+----------------------------------+
| DAYOFMONTH(DATE('2020-08-26'))   |
|----------------------------------|
| 26                               |
+----------------------------------+
```

DAY_OF_MONTH

Utilisation : DAY_OF_MONTH(DATE) extrait le jour du mois pour une date comprise entre 1 et 31.

Type d'argument : STRING/DATE/TIMESTAMP

Type de retour : INTEGER

Synonymes :DAY, DAYOFMONTH

Exemple :

```
os> source=people | eval `DAY_OF_MONTH(DATE('2020-08-26'))` =
 DAY_OF_MONTH(DATE('2020-08-26')) | fields `DAY_OF_MONTH(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-----------------------------------+
| DAY_OF_MONTH(DATE('2020-08-26'))   |
|-----------------------------------|
| 26                                |
+-----------------------------------+
```

DAYOFWEEK

Utilisation : DAYOFWEEK(DATE) renvoie l'index des jours de la semaine pour une date (1 = dimanche, 2 = lundi,..., 7 = samedi).

Type d'argument : STRING/DATE/TIMESTAMP

Type de retour : INTEGER

Synonymes : DAY_OF_WEEK

Exemple :

```
os> source=people | eval `DAYOFWEEK(DATE('2020-08-26'))` =
 DAYOFWEEK(DATE('2020-08-26')) | fields `DAYOFWEEK(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+---------------------------------+
| DAYOFWEEK(DATE('2020-08-26'))   |
|---------------------------------|
| 4                               |
+---------------------------------+
```

DAY_OF_WEEK

Utilisation : DAY_OF_WEEK(DATE) renvoie l'index des jours de la semaine pour une date (1 = dimanche, 2 = lundi,..., 7 = samedi).

Type d'argument : STRING/DATE/TIMESTAMP

Type de retour : INTEGER

Synonymes : DAYOFWEEK

Exemple :

```
os> source=people | eval `DAY_OF_WEEK(DATE('2020-08-26'))` =
 DAY_OF_WEEK(DATE('2020-08-26')) | fields `DAY_OF_WEEK(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+----------------------------------+
| DAY_OF_WEEK(DATE('2020-08-26'))  |
|----------------------------------|
| 4                                |
+----------------------------------+
```

DAYOFYEAR

Utilisation : DAYOFYEAR(DATE) renvoie le jour de l'année pour une date comprise entre 1 et 366.

Type d'argument : STRING/DATE/TIMESTAMP

Type de retour : INTEGER

Synonymes : DAY_OF_YEAR

Exemple :

```
os> source=people | eval `DAYOFYEAR(DATE('2020-08-26'))` =
 DAYOFYEAR(DATE('2020-08-26')) | fields `DAYOFYEAR(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+---------------------------------+
| DAYOFYEAR(DATE('2020-08-26'))   |
|---------------------------------|
| 239                             |
+---------------------------------+
```

DAY_OF_YEAR

Utilisation : DAY_OF_YEAR(DATE) renvoie le jour de l'année pour une date comprise entre 1 et 366.

Type d'argument : STRING/DATE/TIMESTAMP

Type de retour : INTEGER

Synonymes : DAYOFYEAR

Exemple :

```
os> source=people | eval `DAY_OF_YEAR(DATE('2020-08-26'))` =
 DAY_OF_YEAR(DATE('2020-08-26')) | fields `DAY_OF_YEAR(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+----------------------------------+
| DAY_OF_YEAR(DATE('2020-08-26'))  |
|----------------------------------|
| 239                              |
+----------------------------------+
```

DAYNAME

Utilisation : DAYNAME(DATE) renvoie le nom du jour de la semaine pour une date, y compris le lundi, le mardi, le mercredi, le jeudi, le vendredi, le samedi et le dimanche.

Type d'argument : STRING/DATE/TIMESTAMP

Type de retour : STRING

Exemple :

```
os> source=people | eval `DAYNAME(DATE('2020-08-26'))` = DAYNAME(DATE('2020-08-26')) |
 fields `DAYNAME(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+------------------------------+
| DAYNAME(DATE('2020-08-26'))  |
|------------------------------|
| Wednesday                    |
+------------------------------+
```

FROM_UNIXTIME

Utilisation : FROM_UNIXTIME renvoie une représentation de l'argument donné sous forme d'horodatage ou de valeur de chaîne de caractères. Cette fonction effectue une conversion inverse de la UNIX_TIMESTAMP fonction.

Si vous fournissez un deuxième argument, FROM_UNIXTIME utilisez-le pour formater le résultat de la même manière que la DATE_FORMAT fonction.

Si l'horodatage se situe en dehors de la plage 1970-01-01 00:00:00 à 30-01-01-18 23:59:59.999 999 (0 à 32536771199.999999 heure d'époque), la fonction revient. NULL

Type d'argument : DOUBLE, CHAÎNE

Carte des types de retour :

DOUBLE -> HORODATAGE

DOUBLE, CHAÎNE -> CHAÎNE

Exemples :

```
os> source=people | eval `FROM_UNIXTIME(1220249547)` = FROM_UNIXTIME(1220249547) |
 fields `FROM_UNIXTIME(1220249547)`
fetched rows / total rows = 1/1
+------------------------------+
| FROM_UNIXTIME(1220249547)    |
|------------------------------|
| 2008-09-01 06:12:27          |
+------------------------------+
```

```
os> source=people | eval `FROM_UNIXTIME(1220249547, 'HH:mm:ss')` =
 FROM_UNIXTIME(1220249547, 'HH:mm:ss') | fields `FROM_UNIXTIME(1220249547, 'HH:mm:ss')`
fetched rows / total rows = 1/1
+----------------------------------------+
| FROM_UNIXTIME(1220249547, 'HH:mm:ss')   |
|----------------------------------------|
| 06:12:27                               |
+----------------------------------------+
```

HOUR

Utilisation : `HOUR(TIME)` extrait la valeur horaire pour le temps.

Contrairement à une heure standard, la valeur horaire de cette fonction peut avoir une plage supérieure à 23. Par conséquent, la valeur de retour de `HOUR(TIME)` peut être supérieure à 23.

Type d'argument : STRING/TIME/TIMESTAMP

Type de retour : INTEGER

Synonymes : `HOUR_OF_DAY`

Exemple :

```
os> source=people | eval `HOUR(TIME('01:02:03'))` = HOUR(TIME('01:02:03')) | fields
 `HOUR(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+-------------------------+
| HOUR(TIME('01:02:03'))   |
|-------------------------|
| 1                       |
+-------------------------+
```

HOUR_OF_DAY

Utilisation : `HOUR_OF_DAY(TIME)` extrait la valeur de l'heure à partir de l'heure donnée.

Contrairement à une heure standard, la valeur horaire de cette fonction peut avoir une plage supérieure à 23. Par conséquent, la valeur de retour de `HOUR_OF_DAY(TIME)` peut être supérieure à 23.

Type d'argument : STRING/TIME/TIMESTAMP

Type de retour : INTEGER

Synonymes : HOUR

Exemple :

```
os> source=people | eval `HOUR_OF_DAY(TIME('01:02:03'))` =
 HOUR_OF_DAY(TIME('01:02:03')) | fields `HOUR_OF_DAY(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+---------------------------------+
| HOUR_OF_DAY(TIME('01:02:03'))   |
|---------------------------------|
| 1                               |
+---------------------------------+
```

LAST_DAY

Utilisation : LAST_DAY renvoie le dernier jour du mois sous forme de valeur DATE pour l'argument de date donné.

Type d'argument : DATE/STRING/TIMESTAMP/TIME

Type de retour : DATE

Exemple :

```
os> source=people | eval `last_day('2023-02-06')` = last_day('2023-02-06') | fields
 `last_day('2023-02-06')`
fetched rows / total rows = 1/1
+-------------------------+
| last_day('2023-02-06')   |
|-------------------------|
| 2023-02-28              |
+-------------------------+
```

LOCALTIMESTAMP

Utilisation : LOCALTIMESTAMP() est un synonyme deNOW().

Exemple :

```
> source=people | eval `LOCALTIMESTAMP()` = LOCALTIMESTAMP() | fields
 `LOCALTIMESTAMP()`
```

```
fetched rows / total rows = 1/1
+---------------------+
| LOCALTIMESTAMP()    |
|---------------------|
| 2022-08-02 15:54:19 |
+---------------------+
```

LOCALTIME

Utilisation : `LOCALTIME()` est un synonyme de`NOW()`.

Exemple :

```
> source=people | eval `LOCALTIME()` = LOCALTIME() | fields `LOCALTIME()`
fetched rows / total rows = 1/1
+---------------------+
| LOCALTIME()         |
|---------------------|
| 2022-08-02 15:54:19 |
+---------------------+
```

MAKE_DATE

Utilisation : `MAKE_DATE` renvoie une valeur de date basée sur les valeurs d'année, de mois et de jour données. Tous les arguments sont arrondis à des nombres entiers.

Spécifications : 1. MAKE_DATE (ENTIER, ENTIER, ENTIER) -> DATE

Type d'argument : INTEGER, INTEGER, INTEGER

Type de retour : DATE

Exemple :

```
os> source=people | eval `MAKE_DATE(1945, 5, 9)` = MAKEDATE(1945, 5, 9) | fields
 `MAKEDATE(1945, 5, 9)`
fetched rows / total rows = 1/1
+------------------------+
| MAKEDATE(1945, 5, 9)   |
|------------------------|
| 1945-05-09             |
+------------------------+
```

MINUTE

Utilisation : `MINUTE(TIME)` renvoie la composante minute de l'heure donnée, sous la forme d'un entier compris entre 0 et 59.

Type d'argument : STRING/TIME/TIMESTAMP

Type de retour : INTEGER

Synonymes : `MINUTE_OF_HOUR`

Exemple :

```
os> source=people | eval `MINUTE(TIME('01:02:03'))` =  MINUTE(TIME('01:02:03')) |
 fields `MINUTE(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+----------------------------+
| MINUTE(TIME('01:02:03'))   |
|----------------------------|
| 2                          |
+----------------------------+
```

MINUTE_OF_HOUR

Utilisation : `MINUTE_OF_HOUR(TIME)` renvoie la composante minute de l'heure donnée, sous la forme d'un entier compris entre 0 et 59.

Type d'argument : STRING/TIME/TIMESTAMP

Type de retour : INTEGER

Synonymes : `MINUTE`

Exemple :

```
os> source=people | eval `MINUTE_OF_HOUR(TIME('01:02:03'))` =
 MINUTE_OF_HOUR(TIME('01:02:03')) | fields `MINUTE_OF_HOUR(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+-----------------------------------+
| MINUTE_OF_HOUR(TIME('01:02:03'))   |
|-----------------------------------|
| 2                                 |
+-----------------------------------+
```

MONTH

Utilisation : MONTH(DATE) renvoie le mois de la date donnée sous forme de nombre entier, compris entre 1 et 12 (où 1 représente janvier et 12 représente décembre).

Type d'argument : STRING/DATE/TIMESTAMP

Type de retour : INTEGER

Synonymes : MONTH_OF_YEAR

Exemple :

```
os> source=people | eval `MONTH(DATE('2020-08-26'))` =  MONTH(DATE('2020-08-26')) |
 fields `MONTH(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-----------------------------+
| MONTH(DATE('2020-08-26'))   |
|-----------------------------|
| 8                           |
+-----------------------------+
```

MONTHNAME

Utilisation : MONTHNAME(DATE) renvoie le mois de la date donnée sous forme de nombre entier, compris entre 1 et 12 (où 1 représente janvier et 12 représente décembre).

Type d'argument : STRING/DATE/TIMESTAMP

Type de retour : INTEGER

Synonymes : MONTH_OF_YEAR

Exemple :

```
os> source=people | eval `MONTHNAME(DATE('2020-08-26'))` =
 MONTHNAME(DATE('2020-08-26')) | fields `MONTHNAME(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-------------------------------+
| MONTHNAME(DATE('2020-08-26'))   |
|-------------------------------|
| August                        |
+-------------------------------+
```

MONTH_OF_YEAR

Utilisation : MONTH_OF_YEAR(DATE) renvoie le mois de la date donnée sous forme de nombre entier, compris entre 1 et 12 (où 1 représente janvier et 12 représente décembre).

Type d'argument : STRING/DATE/TIMESTAMP

Type de retour : INTEGER

Synonymes : MONTH

Exemple :

```
os> source=people | eval `MONTH_OF_YEAR(DATE('2020-08-26'))` =
 MONTH_OF_YEAR(DATE('2020-08-26')) | fields `MONTH_OF_YEAR(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+-------------------------------------+
| MONTH_OF_YEAR(DATE('2020-08-26'))   |
|-------------------------------------|
| 8                                   |
+-------------------------------------+
```

NOW

Utilisation : NOW renvoie la date et l'heure actuelles sous forme de TIMESTAMP valeur au format « YYYY-MM-DD hh:mm:ss ». La valeur est exprimée dans le fuseau horaire du cluster.

> **Note**
>
> NOW( )renvoie un temps constant qui indique le début de l'exécution de l'instruction. Cela diffère deSYSDATE( ), qui renvoie l'heure exacte d'exécution.

Type de retour : TIMESTAMP

Spécification : NOW () -> TIMESTAMP

Exemple :

```
os> source=people | eval `value_1` = NOW(), `value_2` = NOW() | fields `value_1`,
 `value_2`
fetched rows / total rows = 1/1
+---------------------+---------------------+
```

```
| value_1             | value_2             |
|---------------------+---------------------|
| 2022-08-02 15:39:05 | 2022-08-02 15:39:05 |
+---------------------+---------------------+
```

QUARTER

Utilisation : QUARTER(DATE) renvoie le trimestre de l'année pour la date donnée sous forme de nombre entier, compris entre 1 et 4.

Type d'argument : STRING/DATE/TIMESTAMP

Type de retour : INTEGER

Exemple :

```
os> source=people | eval `QUARTER(DATE('2020-08-26'))` = QUARTER(DATE('2020-08-26')) |
 fields `QUARTER(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+------------------------------+
| QUARTER(DATE('2020-08-26'))  |
|------------------------------|
| 3                            |
+------------------------------+
```

SECOND

Utilisation : SECOND(TIME) renvoie la deuxième composante du temps donné sous forme d'entier, compris entre 0 et 59.

Type d'argument : STRING/TIME/TIMESTAMP

Type de retour : INTEGER

Synonymes : SECOND_OF_MINUTE

Exemple :

```
os> source=people | eval `SECOND(TIME('01:02:03'))` = SECOND(TIME('01:02:03')) | fields
 `SECOND(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+--------------------------+
| SECOND(TIME('01:02:03'))  |
```

```
|----------------------------|
| 3                          |
+----------------------------+
```

SECOND_OF_MINUTE

Utilisation : SECOND_OF_MINUTE(TIME) renvoie la deuxième composante du temps donné sous forme d'entier, compris entre 0 et 59.

Type d'argument : STRING/TIME/TIMESTAMP

Type de retour : INTEGER

Synonymes : SECOND

Exemple :

```
os> source=people | eval `SECOND_OF_MINUTE(TIME('01:02:03'))` =
 SECOND_OF_MINUTE(TIME('01:02:03')) | fields `SECOND_OF_MINUTE(TIME('01:02:03'))`
fetched rows / total rows = 1/1
+-------------------------------------+
| SECOND_OF_MINUTE(TIME('01:02:03'))  |
|-------------------------------------|
| 3                                   |
+-------------------------------------+
```

SUBDATE

Utilisation : SUBDATE(DATE, DAYS) soustrait le deuxième argument (tel que DATE ouDAYS) de la date donnée.

Type d'argument : DATE/TIMESTAMP, LONG

Carte des types de retour : (DATE, LONG) -> DATE

Antonymes : ADDDATE

Exemple :

```
os> source=people | eval `'2008-01-02' - 31d` = SUBDATE(DATE('2008-01-02'), 31),
 `'2020-08-26' - 1` = SUBDATE(DATE('2020-08-26'), 1), `ts '2020-08-26 01:01:01' -
 1` = SUBDATE(TIMESTAMP('2020-08-26 01:01:01'), 1) | fields `'2008-01-02' - 31d`,
 `'2020-08-26' - 1`, `ts '2020-08-26 01:01:01' - 1`
fetched rows / total rows = 1/1
```

```
+----------------------+--------------------+--------------------------------+
| '2008-01-02' - 31d   | '2020-08-26' - 1   | ts '2020-08-26 01:01:01' - 1   |
|----------------------+--------------------+--------------------------------|
| 2007-12-02 00:00:00  | 2020-08-25         | 2020-08-25 01:01:01            |
+----------------------+--------------------+--------------------------------+
```

SYSDATE

Utilisation : SYSDATE() renvoie la date et l'heure actuelles sous forme de TIMESTAMP valeur au format « YYYY-MM-DD hh:mm:ss.nnnnnn ».

SYSDATE()renvoie l'heure exacte à laquelle il s'exécute. Cela diffère de NOW (), qui renvoie un temps constant indiquant le moment où l'instruction a commencé à s'exécuter.

Type d'argument facultatif : INTEGER (0 à 6) - Spécifie le nombre de chiffres pour les fractions de secondes dans la valeur de retour.

Type de retour : TIMESTAMP

Exemple :

```
os> source=people | eval `SYSDATE()` = SYSDATE() | fields `SYSDATE()`
fetched rows / total rows = 1/1
+----------------------------+
| SYSDATE()                  |
|----------------------------|
| 2022-08-02 15:39:05.123456 |
+----------------------------+
```

TIMESTAMP

Utilisation : TIMESTAMP(EXPR) construit un type d'horodatage avec la chaîne d'entrée expr comme horodatage.

Avec un seul argument, TIMESTAMP(expr) construit un horodatage à partir de l'entrée. S'il s'expragit d'une chaîne, elle est interprétée comme un horodatage. Pour les arguments autres que des chaînes, la fonction est convertie expr en horodatage en utilisant le fuseau horaire UTC. Quand expr est une TIME valeur, la fonction applique la date du jour avant le casting.

Lorsqu'il est utilisé avec deux arguments, TIMESTAMP(expr1, expr2) ajoute l'expression temporelle (expr2) à l'expression de date ou d'horodatage (expr1) et renvoie le résultat sous forme de valeur d'horodatage.

Type d'argument : STRING/DATE/TIME/TIMESTAMP

Carte des types de retour :

(STRING/DATE/TIME/TIMESTAMP) -> HORODATAGE

(STRING/DATE/TIME/TIMESTAMP, STRING/DATE/TIME/TIMESTAMP) -> HORODATAGE

Exemple :

```
os> source=people | eval `TIMESTAMP('2020-08-26 13:49:00')` = TIMESTAMP('2020-08-26
 13:49:00'), `TIMESTAMP('2020-08-26 13:49:00', TIME('12:15:42'))` =
 TIMESTAMP('2020-08-26 13:49:00', TIME('12:15:42')) | fields `TIMESTAMP('2020-08-26
 13:49:00')`, `TIMESTAMP('2020-08-26 13:49:00', TIME('12:15:42'))`
fetched rows / total rows = 1/1
+-----------------------------------
+-----------------------------------------------------+
| TIMESTAMP('2020-08-26 13:49:00')   | TIMESTAMP('2020-08-26 13:49:00',
 TIME('12:15:42'))   |
|-----------------------------------
+-----------------------------------------------------|
| 2020-08-26 13:49:00                | 2020-08-27 02:04:42
     |
+-----------------------------------
+-----------------------------------------------------+
```

UNIX_TIMESTAMP

Utilisation : UNIX_TIMESTAMP convertit un argument de date donné en heure Unix (secondes depuis Epoch, qui a débuté début 1970). Si aucun argument n'est fourni, il renvoie l'heure Unix actuelle.

L'argument date peut être uneDATE, une TIMESTAMP chaîne ou un nombre dans l'un des formats suivants : YYMMDDYYMMDDhhmmss,YYYYMMDD, ouYYYYMMDDhhmmss. Si l'argument inclut une composante temporelle, il peut éventuellement inclure des fractions de secondes.

Si le format de l'argument n'est pas valide ou s'il se situe en dehors de la plage comprise entre 1970-01-01 00:00:00 et 30-01-01-18 23:59:59.999 999 (0 à 32536771199.999999 en temps d'époque), la fonction retourne. NULL

La fonction accepte DATETIMESTAMP, ou DOUBLE en tant que types d'arguments, ou aucun argument. Elle renvoie toujours une DOUBLE valeur représentant l'horodatage Unix.

Pour la conversion inverse, vous pouvez utiliser la fonction FROM_UNIXTIME.

Type d'argument :<NONE>/DOUBLE/DATE/TIMESTAMP

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `UNIX_TIMESTAMP(double)` = UNIX_TIMESTAMP(20771122143845),
 `UNIX_TIMESTAMP(timestamp)` = UNIX_TIMESTAMP(TIMESTAMP('1996-11-15 17:05:42')) |
 fields `UNIX_TIMESTAMP(double)`, `UNIX_TIMESTAMP(timestamp)`
fetched rows / total rows = 1/1
+-------------------------+----------------------------+
| UNIX_TIMESTAMP(double)  | UNIX_TIMESTAMP(timestamp)  |
|-------------------------+----------------------------|
| 3404817525.0            | 848077542.0                |
+-------------------------+----------------------------+
```

WEEK

Utilisation : `WEEK(DATE)` renvoie le numéro de semaine pour une date donnée.

Type d'argument : DATE/TIMESTAMP/STRING

Type de retour : INTEGER

Synonymes : `WEEK_OF_YEAR`

Exemple :

```
os> source=people | eval `WEEK(DATE('2008-02-20'))` = WEEK(DATE('2008-02-20')) | fields
 `WEEK(DATE('2008-02-20'))`
fetched rows / total rows = 1/1
+---------------------------+
| WEEK(DATE('2008-02-20'))  |
|---------------------------|
| 8                         |
+---------------------------+
```

WEEKDAY

Utilisation : `WEEKDAY(DATE)` renvoie l'index des jours de la semaine pour la date (0 = lundi, 1 = mardi,..., 6 = dimanche).

Elle est similaire à la `dayofweek` fonction, mais renvoie des index différents pour chaque jour.

Type d'argument : STRING/DATE/TIME/TIMESTAMP

Type de retour : INTEGER

Exemple :

```
os> source=people | eval `weekday(DATE('2020-08-26'))` = weekday(DATE('2020-08-26'))
 | eval `weekday(DATE('2020-08-27'))` = weekday(DATE('2020-08-27')) | fields
 `weekday(DATE('2020-08-26'))`, `weekday(DATE('2020-08-27'))`
fetched rows / total rows = 1/1
+------------------------------+------------------------------+
| weekday(DATE('2020-08-26'))  | weekday(DATE('2020-08-27'))  |
|------------------------------+------------------------------|
| 2                            | 3                            |
+------------------------------+------------------------------+
```

WEEK_OF_YEAR

Utilisation : `WEEK_OF_YEAR(DATE)` renvoie le numéro de semaine pour la date donnée.

Type d'argument : DATE/TIMESTAMP/STRING

Type de retour : INTEGER

Synonymes : `WEEK`

Exemple :

```
os> source=people | eval `WEEK_OF_YEAR(DATE('2008-02-20'))` = WEEK(DATE('2008-02-20'))|
 fields `WEEK_OF_YEAR(DATE('2008-02-20'))`
fetched rows / total rows = 1/1
+-----------------------------------+
| WEEK_OF_YEAR(DATE('2008-02-20'))  |
|-----------------------------------|
| 8                                 |
+-----------------------------------+
```

YEAR

Utilisation : `YEAR(DATE)` renvoie l'année pour la date, comprise entre 1000 et 9999, ou 0 pour la date « zéro ».

Type d'argument : STRING/DATE/TIMESTAMP

Type de retour : INTEGER

Exemple :

```
os> source=people | eval `YEAR(DATE('2020-08-26'))` = YEAR(DATE('2020-08-26')) | fields
 `YEAR(DATE('2020-08-26'))`
fetched rows / total rows = 1/1
+---------------------------+
| YEAR(DATE('2020-08-26'))   |
|---------------------------|
| 2020                      |
+---------------------------+
```

DATE_ADD

Utilisation : `DATE_ADD(date, INTERVAL expr unit)` ajoute l'intervalle spécifié à la date donnée.

Type d'argument : DATE, INTERVALLE

Type de retour : DATE

Antonymes : `DATE_SUB`

Exemple :

```
os> source=people | eval `'2020-08-26' + 1d` = DATE_ADD(DATE('2020-08-26'), INTERVAL 1
 DAY) | fields `'2020-08-26' + 1d`
fetched rows / total rows = 1/1
+---------------------+
| '2020-08-26' + 1d   |
|---------------------|
| 2020-08-27          |
+---------------------+
```

DATE_SUB

Utilisation : `DATE_SUB(date, INTERVAL expr unit)` soustrait l'intervalle expr de la date.

Type d'argument : DATE, INTERVALLE

Type de retour : DATE

Antonymes : `DATE_ADD`

Exemple :

```
os> source=people | eval `'2008-01-02' - 31d` = DATE_SUB(DATE('2008-01-02'), INTERVAL
 31 DAY) | fields `'2008-01-02' - 31d`
fetched rows / total rows = 1/1
+---------------------+
| '2008-01-02' - 31d  |
|---------------------|
| 2007-12-02          |
+---------------------+
```

`TIMESTAMPADD`

Utilisation : renvoie une `TIMESTAMP` valeur après avoir ajouté un intervalle de temps spécifié à une date donnée.

Arguments :

- intervalle : INTERVALLE (SECONDE, MINUTE, HEURE, JOUR, SEMAINE, MOIS, TRIMESTRE, ANNÉE)
- entier : ENTIER
- date : DATE, TIMESTAMP ou CHAÎNE

Si vous fournissez un `STRING` comme argument de date, formatez-le comme valide`TIMESTAMP`. La fonction convertit automatiquement un `DATE` argument en`TIMESTAMP`.

Exemples :

```
os> source=people | eval `TIMESTAMPADD(DAY, 17, '2000-01-01 00:00:00')` =
 TIMESTAMPADD(DAY, 17, '2000-01-01 00:00:00') | eval `TIMESTAMPADD(QUARTER, -1,
 '2000-01-01 00:00:00')` = TIMESTAMPADD(QUARTER, -1, '2000-01-01 00:00:00') | fields
 `TIMESTAMPADD(DAY, 17, '2000-01-01 00:00:00')`, `TIMESTAMPADD(QUARTER, -1, '2000-01-01
 00:00:00')`
fetched rows / total rows = 1/1
+-----------------------------------------------
+-----------------------------------------------+
```

```
| TIMESTAMPADD(DAY, 17, '2000-01-01 00:00:00') | TIMESTAMPADD(QUARTER, -1, '2000-01-01
 00:00:00') |
|----------------------------------------------
+----------------------------------------------|
| 2000-01-18 00:00:00                          | 1999-10-01 00:00:00
           |
+---------------------------------------------
+-----------------------------------------------+
```

TIMESTAMPDIFF

Utilisation : `TIMESTAMPDIFF(interval, start, end)` renvoie la différence entre le début et la fin date/times en unités d'intervalle spécifiées.

Arguments :

- intervalle : INTERVALLE (SECONDE, MINUTE, HEURE, JOUR, SEMAINE, MOIS, TRIMESTRE, ANNÉE)
- début : DATE, TIMESTAMP ou STRING
- fin : DATE, TIMESTAMP ou STRING

La fonction convertit automatiquement les arguments en arguments `TIMESTAMP` lorsque cela est approprié. `STRING`Formatez les arguments en tant que `TIMESTAMP` s valides.

Exemples :

```
os> source=people | eval `TIMESTAMPDIFF(YEAR, '1997-01-01 00:00:00', '2001-03-06
 00:00:00')` = TIMESTAMPDIFF(YEAR, '1997-01-01 00:00:00', '2001-03-06 00:00:00') |
 eval `TIMESTAMPDIFF(SECOND, timestamp('1997-01-01 00:00:23'), timestamp('1997-01-01
 00:00:00'))` = TIMESTAMPDIFF(SECOND, timestamp('1997-01-01 00:00:23'),
 timestamp('1997-01-01 00:00:00')) | fields `TIMESTAMPDIFF(YEAR, '1997-01-01 00:00:00',
 '2001-03-06 00:00:00')`, `TIMESTAMPDIFF(SECOND, timestamp('1997-01-01 00:00:23'),
 timestamp('1997-01-01 00:00:00'))`
fetched rows / total rows = 1/1
+-------------------------------------------------------------------
+-------------------------------------------------------------------------------------------
+
| TIMESTAMPDIFF(YEAR, '1997-01-01 00:00:00', '2001-03-06 00:00:00') |
 TIMESTAMPDIFF(SECOND, timestamp('1997-01-01 00:00:23'), timestamp('1997-01-01
 00:00:00')) |
|-------------------------------------------------------------------
+-------------------------------------------------------------------------------------------|
```

```
| 4                                                                  | -23
                                                                        |
+--------------------------------------------------------------------
+-------------------------------------------------------------------------------------------
+
```

UTC_TIMESTAMP

Utilisation : UTC_TIMESTAMP renvoie l'horodatage UTC actuel sous forme de valeur dans « YYYY-MM-DD hh:mm:ss ».

Type de retour : TIMESTAMP

Spécification : UTC_TIMESTAMP () -> TIMESTAMP

Exemple :

```
> source=people | eval `UTC_TIMESTAMP()` = UTC_TIMESTAMP() | fields `UTC_TIMESTAMP()`
fetched rows / total rows = 1/1
+---------------------+
| UTC_TIMESTAMP()     |
|---------------------|
| 2022-10-03 17:54:28 |
+---------------------+
```

CURRENT_TIMEZONE

Utilisation : CURRENT_TIMEZONE renvoie le fuseau horaire local actuel.

Type de retour : STRING

Exemple :

```
> source=people | eval `CURRENT_TIMEZONE()` = CURRENT_TIMEZONE() | fields
 `CURRENT_TIMEZONE()`
fetched rows / total rows = 1/1
+------------------------+
| CURRENT_TIMEZONE()     |
|------------------------|
| America/Chicago        |
+------------------------+
```

Expressions PPL

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette fonction PPL, consultez. the section called “Fonctions”

Les expressions, en particulier les expressions de valeur, renvoient une valeur scalaire. Les expressions ont différents types et formes. Par exemple, il existe des valeurs littérales sous forme d'expressions atomiques et des expressions arithmétiques, de prédicats et de fonctions basées sur celles-ci. Vous pouvez utiliser des expressions dans différentes clauses, par exemple en utilisant des expressions arithmétiques dans `Stats` les commandes `Filter` et.

Opérateurs

Une expression arithmétique est une expression formée de littéraux numériques et d'opérateurs arithmétiques binaires comme suit :

1. `+`: Ajouter.
2. `-`: Soustraire.
3. `*`: Multipliez.
4. `/`: Diviser (pour les entiers, le résultat est un entier dont la partie fractionnaire est ignorée)
5. `%`: Modulo (à utiliser uniquement avec des entiers ; le résultat est le reste de la division)

Priorité

Utilisez des parenthèses pour contrôler la priorité des opérateurs arithmétiques. Dans le cas contraire, les opérateurs de priorité supérieure sont exécutés en premier.

Conversion de type

Une conversion de type implicite est effectuée lors de la recherche de signatures d'opérateurs. Par exemple, un entier correspond à `+` un nombre réel à une signature`+(double,double)`, ce qui donne un nombre réel. Cette règle s'applique également aux appels de fonctions.

Exemple de différents types d'expressions arithmétiques :

```
os> source=accounts | where age > (25 + 5) | fields age ;
```

```
fetched rows / total rows = 3/3
+-------+
| age   |
|-------|
| 32    |
| 36    |
| 33    |
+-------+
```

Opérateurs de prédicat

Un opérateur de prédicat est une expression qui est évaluée comme vraie. La comparaison `MISSING` des `NULL` valeurs et suit les règles suivantes :

- Une `MISSING` valeur n'est égale qu'à une `MISSING` valeur et est inférieure aux autres valeurs.
- Une `NULL` valeur est égale à une `NULL` valeur, est supérieure à une `MISSING` valeur, mais inférieure à toutes les autres valeurs.

Opérateurs

Opérateurs de prédicat

Name (Nom)	Description
`>`	Supérieur à l'opérateur
`>=`	Opérateur supérieur ou égal
`<`	Moins qu'un opérateur
`!=`	Opérateur différent
`<=`	Opérateur inférieur ou égal
`=`	Opérateur égal
`LIKE`	Correspondance simple des motifs
`IN`	Test de valeur NULL
`AND`	Opérateur AND

Name (Nom)	Description
OR	Opérateur OR
XOR	Opérateur XOR
NOT	Test de valeur NON NULLE

Vous pouvez comparer les dates et les heures. Lorsque vous comparez différents types de date/heure (par exemple `DATE` et`TIME`), les deux sont convertis en`DATETIME`. Les règles suivantes s'appliquent à la conversion :

- `TIME`s'applique à la date d'aujourd'hui.
- `DATE`est interprété à minuit.

Opérateur de prédicat de base

Exemple d'opérateurs de comparaison :

```
os> source=accounts | where age > 33 | fields age ;
fetched rows / total rows = 1/1
+-------+
| age   |
|-------|
| 36    |
+-------+
```

`IN`

Exemple de champ de test de `IN` l'opérateur dans les listes de valeurs :

```
os> source=accounts | where age in (32, 33) | fields age ;
fetched rows / total rows = 2/2
+-------+
| age   |
|-------|
| 32    |
| 33    |
+-------+
```

OR

Exemple de l'ORopérateur :

```
os> source=accounts | where age = 32 OR age = 33 | fields age ;
fetched rows / total rows = 2/2
+-------+
| age   |
|-------|
| 32    |
| 33    |
+-------+
```

NOT

Exemple de l'NOTopérateur :

```
os> source=accounts | where age not in (32, 33) | fields age ;
fetched rows / total rows = 2/2
+-------+
| age   |
|-------|
| 36    |
| 28    |
+-------+
```

Fonctions d'adresse IP PPL

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette fonction PPL, consultez. the section called “Fonctions”

CIDRMATCH

Utilisation : `CIDRMATCH(ip, cidr)` vérifie si l'adresse IP spécifiée se situe dans la plage CIDR donnée.

Type d'argument :

- CHAÎNE, CHAÎNE
- Type de retour : BOOLEAN

Exemple :

```
os> source=ips | where cidrmatch(ip, '***********/24') | fields ip
fetched rows / total rows = 1/1
+--------------+
| ip           |
|--------------|
| ***********  |
+--------------+

os> source=ipsv6 | where cidrmatch(ip, '2003:db8::/32') | fields ip
fetched rows / total rows = 1/1
+-----------------------------------------+
| ip                                      |
|-----------------------------------------|
| 2003:0db8:****:****:****:****:****:0000 |
+-----------------------------------------+
```

> **Note**
>
> - `ip`peut être une IPv4 ou une IPv6 adresse.
> - `cidr`peut être un IPv4 ou un IPv6 bloc.
> - `ip`et `cidr` doit être soit les deux, IPv4 soit les deux IPv6.
> - `ip`et `cidr` doit être à la fois valide et non vide/non nul.

Fonctions JSON PPL

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette fonction PPL, consultez. the section called “Fonctions”

JSON

Utilisation : `json(value)` évalue si une chaîne peut être analysée au format JSON. La fonction renvoie la chaîne d'origine s'il s'agit d'un JSON valide, ou null s'il n'est pas valide.

Type d'argument : CHAÎNE

Type de retour : STRING/NULL. Expression STRING d'un format d'objet JSON valide.

Exemples :

```
os> source=people | eval `valid_json()` = json('[1,2,3,{"f1":1,"f2":[5,6]},4]') |
 fields valid_json
fetched rows / total rows = 1/1
+---------------------------------+
| valid_json                      |
+---------------------------------+
| [1,2,3,{"f1":1,"f2":[5,6]},4]   |
+---------------------------------+

os> source=people | eval `invalid_json()` = json('{"invalid": "json"') | fields
 invalid_json
fetched rows / total rows = 1/1
+----------------+
| invalid_json   |
+----------------+
| null           |
+----------------+
```

JSON_OBJECT

Utilisation : `json_object(<key>, <value>[, <key>, <value>]...)` renvoie un objet JSON à partir de membres de paires clé-valeur.

Type d'argument :

- A <key>doit être une chaîne.
- A <value>peut être n'importe quel type de données.

Type de retour : JSON_OBJECT. StructType Expression d'un objet JSON valide.

Exemples :

```
os> source=people | eval result = json_object('key', 123.45) | fields result
fetched rows / total rows = 1/1
+------------------+
| result           |
+------------------+
| {"key":123.45}   |
+------------------+

os> source=people | eval result = json_object('outer', json_object('inner', 123.45)) |
 fields result
fetched rows / total rows = 1/1
+------------------------------+
| result                       |
+------------------------------+
| {"outer":{"inner":123.45}}   |
+------------------------------+
```

JSON_ARRAY

Utilisation : `json_array(<value>...)` crée un TABLEAU JSON à l'aide d'une liste de valeurs.

Type d'argument : A `<value>` peut être n'importe quel type de valeur, telle qu'une chaîne, un nombre ou un booléen.

Type de retour : ARRAY. Un tableau de n'importe quel type de données pris en charge pour un tableau JSON valide.

Exemples :

```
os> source=people | eval `json_array` = json_array(1, 2, 0, -1, 1.1, -0.11)
fetched rows / total rows = 1/1
+------------------------------+
| json_array                   |
+------------------------------+
| [1.0,2.0,0.0,-1.0,1.1,-0.11] |
+------------------------------+

os> source=people | eval `json_array_object` = json_object("array", json_array(1, 2, 0,
 -1, 1.1, -0.11))
fetched rows / total rows = 1/1
+----------------------------------------+
| json_array_object                      |
+----------------------------------------+
```

```
| {"array":[1.0,2.0,0.0,-1.0,1.1,-0.11]} |
+----------------------------------------+
```

TO_JSON_STRING

Utilisation : `to_json_string(jsonObject)` renvoie une chaîne JSON avec une valeur d'objet JSON donnée.

Type d'argument : JSON_OBJECT

Type de retour : STRING

Exemples :

```
os> source=people | eval `json_string` = to_json_string(json_array(1, 2, 0, -1, 1.1,
 -0.11)) | fields json_string
fetched rows / total rows = 1/1
+--------------------------------+
| json_string                    |
+--------------------------------+
| [1.0,2.0,0.0,-1.0,1.1,-0.11]   |
+--------------------------------+

os> source=people | eval `json_string` = to_json_string(json_object('key', 123.45)) |
 fields json_string
fetched rows / total rows = 1/1
+-----------------+
| json_string     |
+-----------------+
| {'key', 123.45} |
+-----------------+
```

ARRAY_LENGTH

Utilisation : `array_length(jsonArray)` renvoie le nombre d'éléments du tableau le plus externe.

Type d'argument : ARRAY. Un objet ARRAY ou JSON_ARRAY.

Type de retour : INTEGER

Exemple :

```
os> source=people | eval `json_array` = json_array_length(json_array(1,2,3,4)),
 `empty_array` = json_array_length(json_array())
```

```
fetched rows / total rows = 1/1
+--------------+---------------+
| json_array   | empty_array   |
+--------------+---------------+
| 4            | 0             |
+--------------+---------------+
```

JSON_EXTRACT

Utilisation : `json_extract(jsonStr, path)` extrait un objet JSON d'une chaîne JSON en fonction du chemin JSON spécifié. La fonction renvoie null si la chaîne JSON d'entrée n'est pas valide.

Type d'argument : CHAÎNE, CHAÎNE

Type de retour : STRING

- Expression STRING d'un format d'objet JSON valide.
- NULLest renvoyé en cas de JSON non valide.

Exemples :

```
os> source=people | eval `json_extract('{"a":"b"}', '$.a')` = json_extract('{"a":"b"}',
 '$a')
fetched rows / total rows = 1/1
+----------------------------------+
| json_extract('{"a":"b"}', 'a')   |
+----------------------------------+
| b                                |
+----------------------------------+

os> source=people | eval `json_extract('{"a":[{"b":1},{"b":2}]}', '$.a[1].b')` =
 json_extract('{"a":[{"b":1},{"b":2}]}', '$.a[1].b')
fetched rows / total rows = 1/1
+------------------------------------------------------------+
| json_extract('{"a":[{"b":1.0},{"b":2.0}]}', '$.a[1].b')   |
+------------------------------------------------------------+
| 2.0                                                        |
+------------------------------------------------------------+

os> source=people | eval `json_extract('{"a":[{"b":1},{"b":2}]}', '$.a[*].b')` =
 json_extract('{"a":[{"b":1},{"b":2}]}', '$.a[*].b')
```

```
fetched rows / total rows = 1/1
+-------------------------------------------------------------+
| json_extract('{"a":[{"b":1.0},{"b":2.0}]}', '$.a[*].b')   |
+-------------------------------------------------------------+
| [1.0,2.0]                                                   |
+-------------------------------------------------------------+

os> source=people | eval `invalid_json` = json_extract('{"invalid": "json"')
fetched rows / total rows = 1/1
+----------------+
| invalid_json   |
+----------------+
| null           |
+----------------+
```

JSON_KEYS

Utilisation : `json_keys(jsonStr)` renvoie toutes les clés de l'objet JSON le plus externe sous forme de tableau.

Type d'argument : CHAÎNE. Expression STRING d'un format d'objet JSON valide.

Type de retour : ARRAY [STRING]. La fonction renvoie NULL toute autre chaîne JSON valide, une chaîne vide ou un JSON non valide.

Exemples :

```
os> source=people | eval `keys` = json_keys('{"f1":"abc","f2":{"f3":"a","f4":"b"}}')
fetched rows / total rows = 1/1
+------------+
| keus       |
+------------+
| [f1, f2]   |
+------------+

os> source=people | eval `keys` = json_keys('[1,2,3,{"f1":1,"f2":[5,6]},4]')
fetched rows / total rows = 1/1
+--------+
| keys   |
+--------+
| null   |
+--------+
```

JSON_VALID

Utilisation : `json_valid(jsonStr)` évalue si une chaîne JSON utilise une syntaxe JSON valide et renvoie VRAI ou FAUX.

Type d'argument : CHAÎNE

Type de retour : BOOLEAN

Exemples :

```
os> source=people | eval `valid_json` = json_valid('[1,2,3,4]'), `invalid_json` =
 json_valid('{"invalid": "json"') | feilds `valid_json`, `invalid_json`
fetched rows / total rows = 1/1
+--------------+----------------+
| valid_json   | invalid_json   |
+--------------+----------------+
| True         | False          |
+--------------+----------------+

os> source=accounts | where json_valid('[1,2,3,4]') and isnull(email) | fields
 account_number, email
fetched rows / total rows = 1/1
+------------------+---------+
| account_number   | email   |
|------------------+---------|
| 13               | null    |
+------------------+---------+
```

Fonctions Lambda PPL

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette fonction PPL, consultez. the section called “Fonctions”

EXISTS

Utilisation : `exists(array, lambda)` évalue si un prédicat Lambda est valable pour un ou plusieurs éléments du tableau.

Type d'argument : ARRAY, LAMBDA

Type de retour : BOOLEAN. Renvoie `TRUE` si au moins un élément du tableau répond au prédicat Lambda, sinon. `FALSE`

Exemples :

```
 os> source=people | eval array = json_array(1, -1, 2), result = exists(array, x -> x >
 0) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| true      |
+-----------+

os> source=people | eval array = json_array(-1, -3, -2), result = exists(array, x -> x
 > 0) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| false     |
+-----------+
```

`FILTER`

Utilisation : `filter(array, lambda)` filtre le tableau d'entrée à l'aide de la fonction Lambda donnée.

Type d'argument : ARRAY, LAMBDA

Type de retour : ARRAY. Un ARRAY qui contient tous les éléments du tableau d'entrée qui répondent au prédicat lambda.

Exemples :

```
 os> source=people | eval array = json_array(1, -1, 2), result = filter(array, x -> x >
 0) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
```

```
| [1, 2]    |
+-----------+

os> source=people | eval array = json_array(-1, -3, -2), result = filter(array, x -> x
 > 0) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| []        |
+-----------+
```

TRANSFORM

Utilisation : `transform(array, lambda)` transforme les éléments d'un tableau à l'aide de la fonction de transformation Lambda. Le deuxième argument implique l'indice de l'élément si vous utilisez la fonction Lambda binaire. Ceci est similaire `map` à une programmation fonctionnelle.

Type d'argument : ARRAY, LAMBDA

Type de retour : ARRAY. Un ARRAY qui contient le résultat de l'application de la fonction de transformation lambda à chaque élément du tableau d'entrée.

Exemples :

```
os> source=people | eval array = json_array(1, 2, 3), result = transform(array, x -> x
 + 1) | fields result
fetched rows / total rows = 1/1
+--------------+
| result       |
+--------------+
| [2, 3, 4]    |
+--------------+

os> source=people | eval array = json_array(1, 2, 3), result = transform(array, (x, i)
 -> x + i) | fields result
fetched rows / total rows = 1/1
+--------------+
| result       |
+--------------+
| [1, 3, 5]    |
+--------------+
```

REDUCE

Utilisation : `reduce(array, start, merge_lambda, finish_lambda)` réduit un tableau à une valeur unique en appliquant des fonctions lambda. La fonction applique le merge_lambda à la valeur de départ et à tous les éléments du tableau, puis l'applique au `finish_lambda` résultat.

Type d'argument : ARRAY, ANY, LAMBDA, LAMBDA

Type de retour : N'IMPORTE LEQUEL. Résultat final de l'application des fonctions Lambda à la valeur de départ et au tableau d'entrée.

Exemples :

```
 os> source=people | eval array = json_array(1, 2, 3), result = reduce(array, 0, (acc,
 x) -> acc + x) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| 6         |
+-----------+

os> source=people | eval array = json_array(1, 2, 3), result = reduce(array, 10, (acc,
 x) -> acc + x) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| 16        |
+-----------+

os> source=people | eval array = json_array(1, 2, 3), result = reduce(array, 0, (acc,
 x) -> acc + x, acc -> acc * 10) | fields result
fetched rows / total rows = 1/1
+-----------+
| result    |
+-----------+
| 60        |
+-----------+
```

Fonctions mathématiques PPL

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette fonction PPL, consultez. the section called “Fonctions”

ABS

Utilisation : `ABS(x)` calcule la valeur absolue de x.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : INTEGER/LONG/FLOAT/DOUBLE

Exemple :

```
os> source=people | eval `ABS(-1)` = ABS(-1) | fields `ABS(-1)`
fetched rows / total rows = 1/1
+-----------+
| ABS(-1)   |
|-----------|
| 1         |
+-----------+
```

ACOS

Utilisation : `ACOS(x)` calcule l'arc cosinus de x. Elle retourne `NULL` si x n'est pas compris entre -1 et 1.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `ACOS(0)` = ACOS(0) | fields `ACOS(0)`
fetched rows / total rows = 1/1
+--------------------+
| ACOS(0)            |
```

```
|--------------------|
| 1.5707963267948966 |
+--------------------+
```

ASIN

Utilisation : `asin(x)` calcule l'arc sinusoïdal de x. Elle retourne `NULL` si x n'est pas compris entre -1 et 1.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `ASIN(0)` = ASIN(0) | fields `ASIN(0)`
fetched rows / total rows = 1/1
+-----------+
| ASIN(0)   |
|-----------|
| 0.0       |
+-----------+
```

ATAN

Utilisation : `ATAN(x)` calcule la tangente d'arc de x. `atan(y, x)` Calcule la tangente d'arc de y/x, sauf que les signes des deux arguments déterminent le quadrant du résultat.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `ATAN(2)` = ATAN(2), `ATAN(2, 3)` = ATAN(2, 3) | fields
 `ATAN(2)`, `ATAN(2, 3)`
fetched rows / total rows = 1/1
+--------------------+--------------------+
| ATAN(2)            | ATAN(2, 3)         |
|--------------------+--------------------|
| 1.1071487177940904 | 0.5880026035475675 |
```

```
+-------------------+--------------------+
```

ATAN2

Utilisation : `ATAN2(y, x)` calcule l'arc tangent de y/ x, sauf que les signes des deux arguments déterminent le quadrant du résultat.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `ATAN2(2, 3)` = ATAN2(2, 3) | fields `ATAN2(2, 3)`
fetched rows / total rows = 1/1
+--------------------+
| ATAN2(2, 3)        |
|--------------------|
| 0.5880026035475675 |
+--------------------+
```

CBRT

Utilisation : CBRT calcule la racine cubique d'un nombre.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE :

INTEGER/LONG/FLOAT/DOUBLE-> DOUBLE

Exemple :

```
opensearchsql> source=location | eval `CBRT(8)` = CBRT(8), `CBRT(9.261)` = CBRT(9.261),
 `CBRT(-27)` = CBRT(-27) | fields `CBRT(8)`, `CBRT(9.261)`, `CBRT(-27)`;
fetched rows / total rows = 2/2
+-----------+---------------+-------------+
| CBRT(8)   | CBRT(9.261)   | CBRT(-27)   |
|-----------+---------------+-------------|
| 2.0       | 2.1           | -3.0        |
| 2.0       | 2.1           | -3.0        |
+-----------+---------------+-------------+
```

CEIL

Utilisation : alias de la `CEILING` fonction. `CEILING(T)`prend le plafond de valeur T.

Limitation : `CEILING` ne fonctionne comme prévu que lorsque le type double IEEE 754 affiche une décimale lors du stockage.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : LONG

Exemple :

```
os> source=people | eval `CEILING(0)` = CEILING(0), `CEILING(50.00005)` =
 CEILING(50.00005), `CEILING(-50.00005)` = CEILING(-50.00005) | fields `CEILING(0)`,
 `CEILING(50.00005)`, `CEILING(-50.00005)`
fetched rows / total rows = 1/1
+-------------+--------------------+---------------------+
| CEILING(0)  | CEILING(50.00005)  | CEILING(-50.00005)  |
|-------------+--------------------+---------------------|
| 0           | 51                 | -50                 |
+-------------+--------------------+---------------------+

os> source=people | eval `CEILING(3147483647.12345)` = CEILING(3147483647.12345),
 `CEILING(113147483647.12345)` = CEILING(113147483647.12345),
 `CEILING(3147483647.00001)` = CEILING(3147483647.00001) | fields
 `CEILING(3147483647.12345)`, `CEILING(113147483647.12345)`,
 `CEILING(3147483647.00001)`
fetched rows / total rows = 1/1
+----------------------------+-------------------------------
+----------------------------+
| CEILING(3147483647.12345)  | CEILING(113147483647.12345)  |
 CEILING(3147483647.00001)  |
|----------------------------+-------------------------------
+----------------------------|
| 3147483648                 | 113147483648                 | 3147483648
     |
+----------------------------+-------------------------------
+----------------------------+
```

CONV

Utilisation : `CONV(x, a, b)` convertit le nombre x d'une base en une base b.

Type d'argument : x : CHAÎNE, a : ENTIER, b : ENTIER

Type de retour : STRING

Exemple :

```
os> source=people | eval `CONV('12', 10, 16)` = CONV('12', 10, 16), `CONV('2C', 16,
 10)` = CONV('2C', 16, 10), `CONV(12, 10, 2)` = CONV(12, 10, 2), `CONV(1111, 2, 10)` =
 CONV(1111, 2, 10) | fields `CONV('12', 10, 16)`, `CONV('2C', 16, 10)`, `CONV(12, 10,
 2)`, `CONV(1111, 2, 10)`
fetched rows / total rows = 1/1
+---------------------+---------------------+-------------------
+---------------------+
| CONV('12', 10, 16)   | CONV('2C', 16, 10)   | CONV(12, 10, 2)   | CONV(1111, 2, 10)
 |
|---------------------+---------------------+-------------------
+---------------------|
| c                    | 44                   | 1100              | 15
 |
+---------------------+---------------------+-------------------
+---------------------+
```

COS

Utilisation : `COS(x)` calcule le cosinus de x, où x est donné en radians.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `COS(0)` = COS(0) | fields `COS(0)`
fetched rows / total rows = 1/1
+----------+
| COS(0)   |
|----------|
| 1.0      |
+----------+
```

COT

Utilisation : `COT(x)` calcule la cotangente de x. Elle renvoie out-of-range une erreur si x est égal à 0.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `COT(1)` = COT(1) | fields `COT(1)`
fetched rows / total rows = 1/1
+--------------------+
| COT(1)             |
|--------------------|
| 0.6420926159343306 |
+--------------------+
```

CRC32

Utilisation : CRC32 calcule une valeur de contrôle de redondance cyclique et renvoie une valeur non signée de 32 bits.

Type d'argument : CHAÎNE

Type de retour : LONG

Exemple :

```
os> source=people | eval `CRC32('MySQL')` = CRC32('MySQL') | fields `CRC32('MySQL')`
fetched rows / total rows = 1/1
+------------------+
| CRC32('MySQL')   |
|------------------|
| 3259397556       |
+------------------+
```

DEGREES

Utilisation : DEGREES(x) convertit x de radians en degrés.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `DEGREES(1.57)` = DEGREES(1.57) | fields `DEGREES(1.57)`
fetched rows / total rows  = 1/1
+------------------+
| DEGREES(1.57)    |
|------------------|
| 89.95437383553924 |
+------------------+
```

E

Utilisation : `E()` renvoie le numéro d'Euler.

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `E()` = E() | fields `E()`
fetched rows / total rows = 1/1
+------------------+
| E()              |
|------------------|
| 2.718281828459045 |
+------------------+
```

EXP

Utilisation : `EXP(x)` renvoie e élevé à la puissance de x.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `EXP(2)` = EXP(2) | fields `EXP(2)`
fetched rows / total rows = 1/1
+------------------+
| EXP(2)           |
|------------------|
| 7.38905609893065 |
+------------------+
```

FLOOR

Utilisation : FLOOR(T) prend le plancher de la valeur T.

Limitation : FLOOR ne fonctionne comme prévu que lorsque le type double IEEE 754 affiche une décimale lors du stockage.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : LONG

Exemple :

```
os> source=people | eval `FLOOR(0)` = FLOOR(0), `FLOOR(50.00005)` = FLOOR(50.00005),
 `FLOOR(-50.00005)` = FLOOR(-50.00005) | fields `FLOOR(0)`, `FLOOR(50.00005)`,
 `FLOOR(-50.00005)`
fetched rows / total rows = 1/1
+------------+------------------+--------------------+
| FLOOR(0)   | FLOOR(50.00005)   | FLOOR(-50.00005)   |
|------------+------------------+--------------------|
| 0          | 50                | -51                |
+------------+------------------+--------------------+

os> source=people | eval `FLOOR(3147483647.12345)` = FLOOR(3147483647.12345),
 `FLOOR(113147483647.12345)` = FLOOR(113147483647.12345), `FLOOR(3147483647.00001)`
 = FLOOR(3147483647.00001) | fields `FLOOR(3147483647.12345)`,
 `FLOOR(113147483647.12345)`, `FLOOR(3147483647.00001)`
fetched rows / total rows = 1/1
+--------------------------+----------------------------+--------------------------+
| FLOOR(3147483647.12345)   | FLOOR(113147483647.12345)   | FLOOR(3147483647.00001)   |
|--------------------------+----------------------------+--------------------------|
| 3147483647                | 113147483647                | 3147483647                |
+--------------------------+----------------------------+--------------------------+

os> source=people | eval `FLOOR(282474973688888.022)` = FLOOR(282474973688888.022),
 `FLOOR(9223372036854775807.022)` = FLOOR(9223372036854775807.022),
 `FLOOR(9223372036854775807.0000001)` = FLOOR(9223372036854775807.0000001)
 | fields `FLOOR(282474973688888.022)`, `FLOOR(9223372036854775807.022)`,
 `FLOOR(9223372036854775807.0000001)`
fetched rows / total rows = 1/1
+----------------------------+---------------------------------
+-------------------------------------+
| FLOOR(282474973688888.022)   | FLOOR(9223372036854775807.022)   |
 FLOOR(9223372036854775807.0000001)   |
```

```
|-----------------------------+-----------------------------------
+-------------------------------------|
| 282474973688888             | 9223372036854775807             | 9223372036854775807
                  |
+-----------------------------+-----------------------------------
+-------------------------------------+
```

LN

Utilisation : `LN(x)` renvoie le logarithme naturel de x.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `LN(2)` = LN(2) | fields `LN(2)`
fetched rows / total rows = 1/1
+--------------------+
| LN(2)              |
|--------------------|
| 0.6931471805599453 |
+--------------------+
```

LOG

Utilisation : `LOG(x)` renvoie le logarithme naturel de x qui est le logarithme en base e du x. log (B, x) est équivalent à log (x) /log (B).

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `LOG(2)` = LOG(2), `LOG(2, 8)` = LOG(2, 8) | fields `LOG(2)`,
 `LOG(2, 8)`
fetched rows / total rows = 1/1
+--------------------+-------------+
| LOG(2)             | LOG(2, 8)   |
|--------------------+-------------|
| 0.6931471805599453 | 3.0         |
```

```
+--------------------+-------------+
```

LOG2

Utilisation : `LOG2(x)` est équivalent à`log(x)/log(2)`.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `LOG2(8)` = LOG2(8) | fields `LOG2(8)`
fetched rows / total rows = 1/1
+-----------+
| LOG2(8)   |
|-----------|
| 3.0       |
+-----------+
```

LOG10

Utilisation : `LOG10(x)` est équivalent à`log(x)/log(10)`.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `LOG10(100)` = LOG10(100) | fields `LOG10(100)`
fetched rows / total rows = 1/1
+--------------+
| LOG10(100)   |
|--------------|
| 2.0          |
+--------------+
```

MOD

Utilisation : `MOD(n, m)` calcule le reste du nombre n divisé par m.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : type plus large entre les types de n et m si m est une valeur différente de zéro. Si m est égal à 0, renvoie NULL.

Exemple :

```
os> source=people | eval `MOD(3, 2)` = MOD(3, 2), `MOD(3.1, 2)` = MOD(3.1, 2) | fields
 `MOD(3, 2)`, `MOD(3.1, 2)`
fetched rows / total rows = 1/1
+-------------+---------------+
| MOD(3, 2)   | MOD(3.1, 2)   |
|-------------+---------------|
| 1           | 1.1           |
+-------------+---------------+
```

PI

Utilisation : `PI()` renvoie la constante pi.

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `PI()` = PI() | fields `PI()`
fetched rows / total rows = 1/1
+-------------------+
| PI()              |
|-------------------|
| 3.141592653589793 |
+-------------------+
```

POW

Utilisation : `POW(x, y)` calcule la valeur de x élevée à la puissance de y. Les entrées incorrectes renvoient un `NULL` résultat.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Synonymes : `POWER(_, _)`

Exemple :

```
os> source=people | eval `POW(3, 2)` = POW(3, 2), `POW(-3, 2)` = POW(-3, 2), `POW(3,
 -2)` = POW(3, -2) | fields `POW(3, 2)`, `POW(-3, 2)`, `POW(3, -2)`
fetched rows / total rows = 1/1
+------------+-------------+--------------------+
| POW(3, 2)  | POW(-3, 2)  | POW(3, -2)         |
|------------+-------------+--------------------|
| 9.0        | 9.0         | 0.1111111111111111 |
+------------+-------------+--------------------+
```

POWER

Utilisation : `POWER(x, y)` calcule la valeur de x élevée à la puissance de y. Les entrées incorrectes renvoient un `NULL` résultat.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Synonymes : `POW(_, _)`

Exemple :

```
os> source=people | eval `POWER(3, 2)` = POWER(3, 2), `POWER(-3, 2)` = POWER(-3, 2),
 `POWER(3, -2)` = POWER(3, -2) | fields `POWER(3, 2)`, `POWER(-3, 2)`, `POWER(3, -2)`
fetched rows / total rows = 1/1
+---------------+----------------+--------------------+
| POWER(3, 2)   | POWER(-3, 2)   | POWER(3, -2)       |
|---------------+----------------+--------------------|
| 9.0           | 9.0            | 0.1111111111111111 |
+---------------+----------------+--------------------+
```

RADIANS

Utilisation : `RADIANS(x)` convertit x de degrés en radians.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `RADIANS(90)` = RADIANS(90) | fields `RADIANS(90)`
```

```
fetched rows / total rows  = 1/1
+--------------------+
| RADIANS(90)        |
|--------------------|
| 1.5707963267948966 |
+--------------------+
```

RAND

Utilisation :`RAND()`/`RAND(N)`renvoie une valeur à virgule flottante aléatoire comprise entre 0 <= valeur < 1,0. Si vous spécifiez un entier N, la fonction initialise la graine avant son exécution. L'une des implications de ce comportement est qu'avec un argument N identique, `rand(N)` renvoie la même valeur à chaque fois, produisant une séquence répétable de valeurs de colonne.

Type d'argument : INTEGER

Type de retour : FLOAT

Exemple :

```
os> source=people | eval `RAND(3)` = RAND(3) | fields `RAND(3)`
fetched rows / total rows = 1/1
+------------+
| RAND(3)    |
|------------|
| 0.73105735 |
+------------+
```

ROUND

Utilisation : `ROUND(x, d)` arrondit l'argument x à d décimales. Si vous ne spécifiez pas d, la valeur par défaut est 0.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Carte des types de retour :

- (ENTIER/LONG [, ENTIER]) -> LONG
- (FLOAT/DOUBLE [, ENTIER]) -> LONG

Exemple :

```
os> source=people | eval `ROUND(12.34)` = ROUND(12.34), `ROUND(12.34, 1)` =
 ROUND(12.34, 1), `ROUND(12.34, -1)` = ROUND(12.34, -1), `ROUND(12, 1)` = ROUND(12, 1)
 | fields `ROUND(12.34)`, `ROUND(12.34, 1)`, `ROUND(12.34, -1)`, `ROUND(12, 1)`
fetched rows / total rows = 1/1
+----------------+-------------------+--------------------+----------------+
| ROUND(12.34)   | ROUND(12.34, 1)   | ROUND(12.34, -1)   | ROUND(12, 1)   |
|----------------+-------------------+--------------------+----------------|
| 12.0           | 12.3              | 10.0               | 12             |
+----------------+-------------------+--------------------+----------------+
```

SIGN

Utilisation : `SIGN` renvoie le signe de l'argument sous la forme -1, 0 ou 1, selon que le nombre est négatif, nul ou positif.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : INTEGER

Exemple :

```
os> source=people | eval `SIGN(1)` = SIGN(1), `SIGN(0)` = SIGN(0), `SIGN(-1.1)` =
 SIGN(-1.1) | fields `SIGN(1)`, `SIGN(0)`, `SIGN(-1.1)`
fetched rows / total rows = 1/1
+-----------+-----------+--------------+
| SIGN(1)   | SIGN(0)   | SIGN(-1.1)   |
|-----------+-----------+--------------|
| 1         | 0         | -1           |
+-----------+-----------+--------------+
```

SIN

Utilisation : `sin(x)` calcule le sinus de x, où x est exprimé en radians.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Type de retour : DOUBLE

Exemple :

```
os> source=people | eval `SIN(0)` = SIN(0) | fields `SIN(0)`
fetched rows / total rows = 1/1
```

```
+----------+
| SIN(0)   |
|----------|
| 0.0      |
+----------+
```

SQRT

Utilisation : SQRT calcule la racine carrée d'un nombre non négatif.

Type d'argument : INTEGER/LONG/FLOAT/DOUBLE

Carte des types de retour :

- (Non négatif) INTEGER/LONG/FLOAT/DOUBLE -> DOUBLE
- (Négatif) INTEGER/LONG/FLOAT/DOUBLE -> NULL

Exemple :

```
os> source=people | eval `SQRT(4)` = SQRT(4), `SQRT(4.41)` = SQRT(4.41) | fields
 `SQRT(4)`, `SQRT(4.41)`
fetched rows / total rows = 1/1
+-----------+--------------+
| SQRT(4)   | SQRT(4.41)   |
|-----------+--------------|
| 2.0       | 2.1          |
+-----------+--------------+
```

Fonctions de chaîne PPL

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette fonction PPL, consultez. the section called “Fonctions”

CONCAT

Utilisation : `CONCAT(str1, str2, ...., str_9)` additionne jusqu'à 9 chaînes.

Type d'argument :

- CHAÎNE, CHAÎNE,..., CHAÎNE
- Type de retour : STRING

Exemple :

```
os> source=people | eval `CONCAT('hello', 'world')` = CONCAT('hello', 'world'),
 `CONCAT('hello ', 'whole ', 'world', '!')` = CONCAT('hello ', 'whole ', 'world', '!')
 | fields `CONCAT('hello', 'world')`, `CONCAT('hello ', 'whole ', 'world', '!')`
fetched rows / total rows = 1/1
+----------------------------+---------------------------------------------+
| CONCAT('hello', 'world')   | CONCAT('hello ', 'whole ', 'world', '!')   |
|----------------------------+---------------------------------------------|
| helloworld                 | hello whole world!                          |
+----------------------------+---------------------------------------------+
```

CONCAT_WS

Utilisation : `CONCAT_WS(sep, str1, str2)` concatène deux chaînes ou plus en utilisant un séparateur spécifié entre elles.

Type d'argument :

- CHAÎNE, CHAÎNE,..., CHAÎNE
- Type de retour : STRING

Exemple :

```
os> source=people | eval `CONCAT_WS(',', 'hello', 'world')` = CONCAT_WS(',', 'hello',
 'world') | fields `CONCAT_WS(',', 'hello', 'world')`
fetched rows / total rows = 1/1
+-----------------------------------+
| CONCAT_WS(',', 'hello', 'world')   |
|-----------------------------------|
| hello,world                       |
+-----------------------------------+
```

LENGTH

Utilisation : `length(str)` renvoie la longueur de la chaîne d'entrée mesurée en octets.

Type d'argument :

- CHAÎNE
- Type de retour : INTEGER

Exemple :

```
os> source=people | eval `LENGTH('helloworld')` = LENGTH('helloworld') | fields
 `LENGTH('helloworld')`
fetched rows / total rows = 1/1
+------------------------+
| LENGTH('helloworld')   |
|------------------------|
| 10                     |
+------------------------+
```

LOWER

Utilisation : `lower(string)` convertit la chaîne d'entrée en minuscules.

Type d'argument :

- CHAÎNE
- Type de retour : STRING

Exemple :

```
os> source=people | eval `LOWER('helloworld')` = LOWER('helloworld'),
 `LOWER('HELLOWORLD')` = LOWER('HELLOWORLD') | fields `LOWER('helloworld')`,
 `LOWER('HELLOWORLD')`
fetched rows / total rows = 1/1
+----------------------+----------------------+
| LOWER('helloworld')   | LOWER('HELLOWORLD')   |
|----------------------+----------------------|
| helloworld            | helloworld            |
+----------------------+----------------------+
```

LTRIM

Utilisation : `ltrim(str)` supprime les espaces de début de la chaîne d'entrée.

Type d'argument :

- CHAÎNE
- Type de retour : STRING

Exemple :

```
os> source=people | eval `LTRIM('   hello')` = LTRIM('   hello'), `LTRIM('hello   ')` =
 LTRIM('hello   ') | fields `LTRIM('   hello')`, `LTRIM('hello   ')`
fetched rows / total rows = 1/1
+---------------------+---------------------+
| LTRIM('   hello')   | LTRIM('hello   ')   |
|---------------------+---------------------|
| hello               | hello               |
+---------------------+---------------------+
```

POSITION

Utilisation : `POSITION(substr IN str)` renvoie la position de la première occurrence d'une sous-chaîne dans une chaîne. Elle renvoie 0 si la sous-chaîne n'est pas dans la chaîne. Elle renvoie NULL si l'un des arguments est NULL.

Type d'argument :

- CHAÎNE, CHAÎNE
- Type de retour INTEGER

Exemple :

```
os> source=people | eval `POSITION('world' IN 'helloworld')` = POSITION('world'
 IN 'helloworld'), `POSITION('invalid' IN 'helloworld')`= POSITION('invalid' IN
 'helloworld')  | fields `POSITION('world' IN 'helloworld')`, `POSITION('invalid' IN
 'helloworld')`
fetched rows / total rows = 1/1
+-------------------------------------+---------------------------------------+
| POSITION('world' IN 'helloworld')   | POSITION('invalid' IN 'helloworld')   |
|-------------------------------------+---------------------------------------|
| 6                                   | 0                                     |
+-------------------------------------+---------------------------------------+
```

REVERSE

Utilisation : `REVERSE(str)` renvoie la chaîne inversée de la chaîne d'entrée.

Type d'argument :

- CHAÎNE
- Type de retour : STRING

Exemple :

```
os> source=people | eval `REVERSE('abcde')` = REVERSE('abcde') | fields
 `REVERSE('abcde')`
fetched rows / total rows = 1/1
+--------------------+
| REVERSE('abcde')   |
|--------------------|
| edcba              |
+--------------------+
```

RIGHT

Utilisation : `right(str, len)` renvoie les caractères les plus à droite de la chaîne d'entrée. Elle renvoie 0 si la sous-chaîne n'est pas dans la chaîne. Elle renvoie NULL si l'un des arguments est NULL.

Type d'argument :

- CHAÎNE, ENTIER
- Type de retour : STRING

Exemple :

```
os> source=people | eval `RIGHT('helloworld', 5)` = RIGHT('helloworld', 5),
 `RIGHT('HELLOWORLD', 0)` = RIGHT('HELLOWORLD', 0) | fields `RIGHT('helloworld', 5)`,
 `RIGHT('HELLOWORLD', 0)`
fetched rows / total rows = 1/1
+--------------------------+--------------------------+
| RIGHT('helloworld', 5)   | RIGHT('HELLOWORLD', 0)   |
|--------------------------+--------------------------|
```

```
| world                    |                          |
+--------------------------+--------------------------+
```

RTRIM

Utilisation : `rtrim(str)` supprime les espaces de fin de la chaîne d'entrée.

Type d'argument :

- CHAÎNE
- Type de retour : STRING

Exemple :

```
os> source=people | eval `RTRIM('   hello')` = RTRIM('   hello'), `RTRIM('hello   ')` =
 RTRIM('hello   ') | fields `RTRIM('   hello')`, `RTRIM('hello   ')`
fetched rows / total rows = 1/1
+----------------------+----------------------+
| RTRIM('   hello')    | RTRIM('hello   ')    |
|----------------------+----------------------|
|    hello             | hello                |
+----------------------+----------------------+
```

SUBSTRING

Utilisation : `substring(str, start)` or `substring(str, start, length)` renvoie une sous-chaîne de la chaîne d'entrée. Si aucune longueur n'est spécifiée, elle renvoie la chaîne entière à partir de la position de départ.

Type d'argument :

- CHAÎNE, ENTIER, ENTIER
- Type de retour : STRING

Exemple :

```
os> source=people | eval `SUBSTRING('helloworld', 5)` = SUBSTRING('helloworld',
 5), `SUBSTRING('helloworld', 5, 3)` = SUBSTRING('helloworld', 5, 3) | fields
 `SUBSTRING('helloworld', 5)`, `SUBSTRING('helloworld', 5, 3)`
fetched rows / total rows = 1/1
```

```
+------------------------------+----------------------------------+
| SUBSTRING('helloworld', 5)   | SUBSTRING('helloworld', 5, 3)    |
|------------------------------+----------------------------------|
| oworld                       | owo                              |
+------------------------------+----------------------------------+
```

TRIM

Utilisation : `trim(string)` supprime les espaces de début et de fin de la chaîne d'entrée.

Type d'argument :

- CHAÎNE
- Type de retour : STRING

Exemple :

```
os> source=people | eval `TRIM('   hello')` = TRIM('   hello'), `TRIM('hello   ')` =
 TRIM('hello   ') | fields `TRIM('   hello')`, `TRIM('hello   ')`
fetched rows / total rows = 1/1
+--------------------+--------------------+
| TRIM('   hello')   | TRIM('hello   ')   |
|--------------------+--------------------|
| hello              | hello              |
+--------------------+--------------------+
```

UPPER

Utilisation : `upper(string)` convertit la chaîne d'entrée en majuscules.

Type d'argument :

- CHAÎNE
- Type de retour : STRING

Exemple :

```
os> source=people | eval `UPPER('helloworld')` = UPPER('helloworld'),
 `UPPER('HELLOWORLD')` = UPPER('HELLOWORLD') | fields `UPPER('helloworld')`,
 `UPPER('HELLOWORLD')`
```

```
fetched rows / total rows = 1/1
+----------------------+----------------------+
| UPPER('helloworld')   | UPPER('HELLOWORLD')   |
|----------------------+----------------------|
| HELLOWORLD            | HELLOWORLD            |
+----------------------+----------------------+
```

Fonctions de conversion de type PPL

> **Note**
>
> Pour savoir quelles intégrations de sources de AWS données prennent en charge cette fonction PPL, consultez. the section called “Fonctions”

`TRIM`

Utilisation : `cast(expr as dateType)` convertit le `expr` en `dataType` et renvoie la valeur du`dataType`.

Les règles de conversion suivantes s'appliquent :

Règles de conversion de type

Src/Cible	CHAÎNE	NOMBRE	BOOLEAN	TIMESTAMP	DATE	TIME
CHAÎNE		Remarque 1	Remarque 1	HORODATAGE ()	DATE ()	HEURE ()
NOMBRE	Remarque 1		v ! =0	N/A	N/A	N/A
BOOLEAN	Remarque 1	v ? 1:0		N/A	N/A	N/A
TIMESTAMP	Remarque 1	N/A	N/A		DATE ()	HEURE ()
DATE	Remarque 1	N/A	N/A	N/A		N/A

Src/Cible	CHAÎNE	NOMBRE	BOOLEAN	TIMESTAMP	DATE	TIME
TIME	Remarque 1	N/A	N/A	N/A	N/A	

Exemple de conversion en chaîne :

```
os> source=people | eval `cbool` = CAST(true as string), `cint` = CAST(1 as string),
 `cdate` = CAST(CAST('2012-08-07' as date) as string) | fields `cbool`, `cint`, `cdate`
fetched rows / total rows = 1/1
+---------+--------+------------+
| cbool   | cint   | cdate      |
|---------+--------+------------|
| true    | 1      | 2012-08-07 |
+---------+--------+------------+
```

Exemple de conversion numérique :

```
os> source=people | eval `cbool` = CAST(true as int), `cstring` = CAST('1' as int) |
 fields `cbool`, `cstring`
fetched rows / total rows = 1/1
+---------+-----------+
| cbool   | cstring   |
|---------+-----------|
| 1       | 1         |
+---------+-----------+
```

Exemple de diffusion à ce jour :

```
os> source=people | eval `cdate` = CAST('2012-08-07' as date), `ctime` =
 CAST('01:01:01' as time), `ctimestamp` = CAST('2012-08-07 01:01:01' as timestamp) |
 fields `cdate`, `ctime`, `ctimestamp`
fetched rows / total rows = 1/1
+------------+----------+---------------------+
| cdate      | ctime    | ctimestamp          |
|------------+----------+---------------------|
| 2012-08-07 | 01:01:01 | 2012-08-07 01:01:01 |
+------------+----------+---------------------+
```

Exemple de casting enchaîné :

```
os> source=people | eval `cbool` = CAST(CAST(true as string) as boolean) | fields
 `cbool`
fetched rows / total rows = 1/1
+---------+
| cbool   |
|---------|
| True    |
+---------+
```

Surveillance des domaines Amazon OpenSearch Service

La surveillance joue un rôle important dans le maintien de la fiabilité, de la disponibilité et des performances d'Amazon OpenSearch Service et de vos autres AWS solutions. AWS fournit les outils suivants pour surveiller les ressources de votre OpenSearch service, signaler les problèmes et prendre des mesures automatiques le cas échéant :

Amazon CloudWatch

Amazon CloudWatch surveille les ressources OpenSearch de votre service en temps réel. Vous pouvez collecter et suivre les métriques, créer des tableaux de bord personnalisés, et définir des alarmes qui vous informent ou prennent des mesures lorsqu'une métrique atteint un certain seuil. Pour plus d'informations, consultez le guide de CloudWatch l'utilisateur Amazon.

Amazon CloudWatch Logs

Amazon CloudWatch Logs vous permet de surveiller, de stocker et d'accéder à vos fichiers OpenSearch journaux. CloudWatch Les journaux surveillent les informations contenues dans les fichiers journaux et peuvent vous avertir lorsque certains seuils sont atteints. Pour plus d'informations, consultez le guide de l'utilisateur d'Amazon CloudWatch Logs.

Amazon EventBridge

Amazon EventBridge fournit un flux d'événements système en temps quasi réel décrivant les modifications apportées à vos domaines OpenSearch de service. Vous pouvez créer des règles qui surveillent certains événements et déclencher des actions automatisées dans d'autres AWS services lorsque ces événements se produisent. Pour plus d'informations, consultez le guide de EventBridge l'utilisateur Amazon.

AWS CloudTrail

AWS CloudTrail capture les appels d'API de configuration adressés au OpenSearch service sous forme d'événements. Il peut livrer ces événements à un compartiment Amazon S3 que vous spécifiez. Grâce à ces informations, vous pouvez identifier les utilisateurs et les comptes qui ont effectué des demandes, l'adresse IP source à partir de laquelle elles ont été effectuées ainsi que le moment où elles ont été effectuées. Pour plus d'informations, consultez le guide de AWS CloudTrail l'utilisateur.

Rubriques

- Surveillance des métriques OpenSearch du cluster avec Amazon CloudWatch
- Surveillance des OpenSearch journaux avec Amazon CloudWatch Logs
- Surveillance des journaux d'audit dans Amazon OpenSearch Service
- Surveillance des événements OpenSearch liés au service avec Amazon EventBridge
- Surveillance des appels OpenSearch d'API Amazon Service avec AWS CloudTrail

Surveillance des métriques OpenSearch du cluster avec Amazon CloudWatch

Amazon OpenSearch Service publie les données de vos domaines sur Amazon CloudWatch. CloudWatch vous permet de récupérer des statistiques relatives à ces points de données sous forme d'ensemble classé de données en séries chronologiques, appelées métriques. OpenSearch Service envoie la plupart des métriques CloudWatch à des intervalles de 60 secondes. Si vous utilisez des volumes EBS magnétiques ou à usage général, les métriques correspondantes ne sont mises à jour que toutes les cinq minutes. Toutes les métriques cumulées (par exemple`ThreadpoolWriteRejected`,`ThreadpoolSearchRejected`) sont en mémoire et perdront leur état. Les métriques seront réinitialisées lors d'une chute d'un nœud, d'un rebond d'un nœud, d'un remplacement de nœud et d'un déploiement bleu/vert. Pour de plus amples informations sur Amazon CloudWatch, veuillez consulter le Guide de CloudWatch l'utilisateur Amazon.

La console OpenSearch Service affiche un ensemble de graphiques basés sur les données brutes envoyées par CloudWatch. En fonction de vos besoins, vous pouvez choisir d'afficher des données de cluster CloudWatch plutôt que des graphiques dans la console. Le service archive les métriques pendant deux semaines avant de les supprimer. Les métriques sont fournies sans frais supplémentaires, mais la création de tableaux de bord et d'alarmes CloudWatch reste facturée. Pour plus d'informations, consultez les CloudWatchtarifs Amazon.

OpenSearch Le service publie les métriques suivantes pour CloudWatch :

- the section called “Métriques du cluster”
- the section called “Métriques du nœud principal dédié”
- the section called “Métriques du volume EBS”
- the section called “Métriques des instances”
- the section called “UltraWarm métriques”

- the section called “Mesures dédiées concernant le nœud de coordinateur”
- the section called “Métriques de stockage à froid”
- the section called “Métriques d'alerte”
- the section called “Métriques de détection d'anomalies”
- the section called “Métriques de recherche asynchrone”
- the section called “Métriques SQL”
- the section called “Métriques k-NN”
- the section called “Métriques de recherche inter-clusters”
- the section called “Métriques de réplication inter-clusters (CCR)”
- the section called “Métriques Learning to Rank”
- the section called “Métriques du langage de traitement PPL (Piped Processing Language)”

Afficher les métriques dans CloudWatch

CloudWatch les métriques sont d'abord regroupées par espace de noms de service, puis par les différentes combinaisons de dimension au sein de chaque espace de noms.

Pour afficher des métriques à l'aide de la CloudWatch console

1. Ouvrez la CloudWatch console à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Dans le panneau de navigation de gauche, localisez Metrics (Métriques), puis choisissez All metrics (Toutes les métriques). Sélectionnez l'espace de OpenSearchService noms ES/.
3. Choisissez une dimension pour afficher les métriques correspondantes. Les métriques correspondant aux nœuds individuels se trouvent dans la dimension `ClientId, DomainName, NodeId`. Les métriques de cluster se trouvent dans la dimension `Per-Domain, Per-Client Metrics`. Certaines métriques de nœud sont agrégées au niveau du cluster et sont donc incluses dans les deux dimensions. Les métriques de partition se trouvent dans la dimension `ClientId, DomainName, NodeId, ShardRole`.

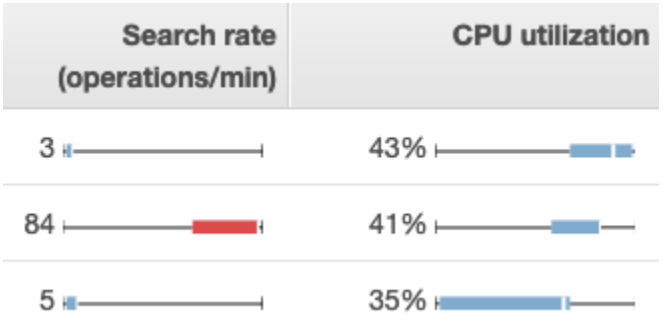
Pour afficher une liste de mesures à l'aide du AWS CLI

Exécutez la commande suivante :

```
aws cloudwatch list-metrics --namespace "AWS/ES"
```

Interprétation des graphiques d'état en OpenSearch service

Pour consulter les métriques dans OpenSearch Service, utilisez les onglets État du cluster et État de l'instance. L'onglet État de l'instance utilise des graphiques de zone pour fournir at-a-glance une visibilité de l'état de chaque OpenSearch nœud :



- Chaque zone colorée indique la plage de valeurs pour le nœud au cours de la période spécifiée.
- Les zones bleues représentent les valeurs qui sont cohérentes avec les autres nœuds. Les zones rouges représentent des valeurs hors normes.
- La ligne blanche dans chaque zone représente la valeur actuelle du nœud.
- Les « moustaches » des deux côtés de chaque zone présentent les valeurs minimale et maximale pour tous les nœuds au cours de la période.

Si vous modifiez la configuration de votre domaine, la liste des instances individuelles qui s'affiche dans les onglets État du cluster et État de l'instance double souvent de taille pour une courte période avant de revenir à sa taille appropriée. Pour obtenir une explication de ce comportement, consultez the section called “Configuration changes”.

Métriques du cluster

Amazon OpenSearch Service fournit les métriques suivantes pour les clusters.

Métrique	Description
`ClusterStatus.green`	Une valeur 1 indique que toutes les partitions d'index sont affectées aux nœuds du cluster. Statistiques pertinentes : Maximum

Métrique	Description
`ClusterStatus.yellow`	Une valeur 1 indique que les partitions principales pour tous les index sont attribuées aux nœuds d'un cluster, sauf pour les partitions de réplica d'au moins un index. Pour de plus amples informations, veuillez consulter the section called “Statut de cluster jaune”. Statistiques pertinentes : Maximum
`ClusterStatus.red`	Une valeur 1 indique que les partitions primaires et de réplica d'au moins un index ne sont pas allouées aux nœuds du cluster. Pour de plus amples informations, veuillez consulter the section called “Statut de cluster rouge”. Statistiques pertinentes : Maximum
`Shards.active`	Nombre total de partitions primaires et de partitions de réplica actives. Statistiques pertinentes : Maximum, Somme
`Shards.unassigned`	Nombre de partitions non allouées aux nœuds du cluster. Statistiques pertinentes : Maximum, Somme
`Shards.delayedUnassigned`	Nombre de partitions dont l'allocation de nœud a été retardée par les paramètres d'expiration. Statistiques pertinentes : Maximum, Somme
`Shards.activePrimary`	Nombre de partitions primaires actives. Statistiques pertinentes : Maximum, Somme
`Shards.initializing`	Nombre de partitions en cours d'initialisation. Statistiques pertinentes : somme

Métrique	Description
`Shards.relocating`	Nombre de partitions en cours de relocalisation. Statistiques pertinentes : somme
`Nodes`	Nombre de nœuds dans le cluster de OpenSearch services, y compris les nœuds principaux dédiés et UltraWarm les nœuds. Pour de plus amples informations, veuillez consulter the section called “Configuration changes”. Statistiques pertinentes : Maximum
`SearchableDocuments`	Nombre total de documents consultables sur tous les nœuds de données du cluster. Statistiques pertinentes : minimum, maximum, moyenne
`DeletedDocuments`	Nombre total de documents marqués pour suppression sur tous les nœuds de données du cluster. Ces documents ne s'affichent plus dans les résultats de recherche, mais OpenSearch retirent uniquement les documents supprimés du disque pendant les fusions de segments. Cette métrique augmente après les demandes de suppression et diminue après les fusions de segments. Statistiques pertinentes : minimum, maximum, moyenne
`CPUUtilization`	Pourcentage d'utilisation du processeur pour les nœuds de données du cluster. Maximum indique le nœud avec l'utilisation la plus élevée du processeur. La moyenne représente tous les nœuds du cluster. Cette métrique est également disponible pour les nœuds individuels. Statistiques pertinentes : Maximum, Moyenne

Métrique	Description
`FreeStorageSpace`	Espace libre pour les nœuds de données du cluster. `Sum` indique l'espace libre total pour le cluster, mais vous devez laisser la période à une minute pour obtenir une valeur précise. `Minimum` et `Maximum` indiquent les nœuds avec le moins et le plus d'espace libre, respectivement. Cette métrique est également disponibl e pour les nœuds individuels. OpenSearch Le service lance un `ClusterBlockException` lorsque cette métrique atteint`0`. Pour récupérer, vous devez supprimer des index, ajouter des instances plus grandes ou ajouter du stockage EBS aux instances existantes. Pour en savoir plus, veuillez consulter la section the section called “Manque d'espace de stockage disponible”. La console OpenSearch de service affiche cette valeur in Gio. La CloudWatch console Amazon l'affiche en Mio. **Note** `FreeStorageSpace` sera toujours inférieure aux valeurs fournies par le OpenSearch `_cluster/stats` et `_cat/allocation` APIs . OpenSearch Service réserve un pourcentage d'espace de stockage dans chaque instance pour les opérations internes. Pour plus d'informations, consultez Calcul des exigences de stockage. Statistiques pertinentes : Minimum, Maximum, Moyenne, Somme
`ClusterUsedSpace`	Espace total utilisé pour le cluster. Vous devez laisser la période à une minute pour obtenir une valeur précise. La console OpenSearch de service affiche cette valeur in Gio. La CloudWatch console Amazon l'affiche en Mio. Statistiques pertinentes : Minimum, Maximum

Métrique	Description
`ClusterIndexWritesBlocked`	Indique si votre cluster accepte ou bloque les demandes d'écriture entrantes. Une valeur de 0 signifie que le cluster accepte les demandes. Une valeur de 1 signifie qu'il bloque les demandes. Parmi les facteurs les plus fréquents, on retrouve les suivants : `FreeStorageSpace` est trop basse ou `JVMMemoryPressure` est trop élevée. Pour résoudre ce problème, nous vous conseillons d'ajouter de l'espace disque supplémentaire ou de redimensionner votre cluster. Statistiques pertinentes : Maximum
`JVMMemoryPressure`	Le pourcentage maximum du tas Java utilisé pour tous les nœuds de données dans le cluster. OpenSearch Le service utilise la moitié de la mémoire RAM d'une instance pour le tas Java, jusqu'à une taille de tas de 32 GiB. Vous pouvez mettre à l'échelle des instances verticalement jusqu'à 64 Gio de RAM, après quoi vous pouvez effectuer une mise à l'échelle horizontale en ajoutant des instances. Consultez the section called “ CloudWatch Alarmes recommandées”. Statistiques pertinentes : Maximum **Note** La logique de cette métrique a changé dans le logiciel de service R20220323. Pour plus d'informations, veuillez consulter les notes de mise à jour.
`OldGenJVMMemoryPressure`	Le pourcentage maximum du tas Java utilisé pour l'« ancienne génération » sur tous les nœuds de données dans le cluster. Cette métrique est également disponible au niveau du nœud. Statistiques pertinentes : Maximum

Métrique	Description
AutomatedSnapshotFailure	Nombre d'instantanés automatiques en échec pour le cluster. Une valeur de 1 indique qu'aucun instantané automatisé n'a été pris pour le domaine dans les 36 dernières heures. Statistiques pertinentes : Minimum, Maximum
CPUCreditBalance	Crédits UC restants disponibles pour des nœuds de données dans le cluster. Un crédit UC fournit les performances d'un cœur UC complet pendant une minute. Pour plus d'informations, consultez la section Crédits CPU dans le manuel Amazon EC2 Developer Guide. Cette métrique est disponible uniquement pour les types d'instance T2. Statistiques pertinentes : Minimum
OpenSearchDashboardsHealthyNodes	Surveillance de l'état des OpenSearch Tableaux de bord Si les statistiques minimales, maximales et moyennes sont toutes égales à 1, les Tableaux de bord se comporteront normalement. Si vous avez 10 nœuds avec un maximum de 1, un minimum de 0 et une moyenne de 0,7, cela signifie que 7 nœuds (70 %) sont sains et 3 nœuds (30%) sont non sains. Statistiques pertinentes : minimum, maximum, moyenne
OpensearchDashboardsReportingFailedRequestSysErrCount	Nombre de demandes visant à générer des rapports OpenSearch Tableaux de bord qui ont échoué en raison de problèmes de serveur ou de limitations de fonctions. Statistiques pertinentes : somme
OpensearchDashboardsReportingFailedRequestUserErrCount	Nombre de demandes visant à générer des rapports OpenSearch Tableaux de bord qui ont échoué en raison de problèmes client. Statistiques pertinentes : somme

Métrique	Description
OpensearchDashboardsReportingRequestCount	Nombre total de demandes visant à générer des rapports de OpenSearch tableaux de bord. Statistiques pertinentes : somme
OpensearchDashboardsReportingSuccessCount	Nombre de demandes réussies visant à générer des rapports OpenSearch Tableaux de bord. Statistiques pertinentes : somme
KMSKeyError	La valeur 1 indique que la AWS KMS clé utilisée pour chiffrer les données au repos a été désactivée. Pour revenir à un fonctionn ement normal du domaine, réactivez la clé. La console n'affiche cette métrique que pour les domaines qui chiffrent les données au repos. Statistiques pertinentes : Minimum, Maximum
KMSKeyInaccessible	La valeur 1 indique que la AWS KMS clé utilisée pour chiffrer les données au repos a été supprimée ou a révoqué ses autorisat ions à OpenSearch Service. Vous ne pouvez pas récupérer des domaines qui sont à cet état. Par contre, si vous disposez d'un instantané manuel, vous pouvez l'utiliser pour migrer les données du domaine vers un nouveau domaine. La console n'affiche cette métrique que pour les domaines qui chiffrent les données au repos. Statistiques pertinentes : Minimum, Maximum

Métrique	Description
`InvalidHostHeaderRequests`	Nombre de demandes HTTP faites au OpenSearch cluster incluant un en-tête d'hôte non valide (ou manquant). Les demandes valides incluent le nom d'hôte de domaine en tant que valeur de l'en-tête d'hôte. OpenSearch Le service rejette les demandes non valides pour les domaines d'accès public qui n'ont pas de stratégie d'accès restrictive. Nous recommandons d'appliquer une stratégie d'accès restrictive à tous les domaines. Si vous constatez que cette métrique présente des valeurs importantes, confirmez que vos clients OpenSearch incluent le nom d'hôte de domaine (et non, par exemple, son adresse IP) dans leurs demandes. Statistiques pertinentes : somme
`OpenSearchRequests (previously ElasticsearchRequests)`	Nombre de demandes faites au OpenSearch cluster. Statistiques pertinentes : somme
`2xx, 3xx, 4xx, 5xx`	Nombre de demandes adressées à un domaine ayant entraîné le code de réponse HTTP donné (2xx, 3xx, 4xx, 5xx). Statistiques pertinentes : somme

Métrique	Description
`ThroughputThrottle`	Indique si les disques ont été limités. L'étranglement se produit lorsque le débit combiné de `ReadThroughputMicroBursting` et `WriteThroughputMicroBursting` est supérieur au débit maximal,. `MaxProvisionedThroughput` `MaxProvisionedThroughput` est la valeur inférieure du débit de l'instance ou du débit du volume provisionné. Une valeur égale à 1 indique que les disques ont été limités. La valeur 0 indique un comportement normal. Pour plus d'informations sur le débit des instances, consultez la section Instances optimisées pour Amazon EBS. Pour plus d'informations sur le débit des volumes, consultez la section Types de volumes Amazon EBS. Statistiques pertinentes : Minimum, Maximum
`IopsThrottle`	Indique si le nombre d'opérations d'E/S par seconde (IOPS) sur le domaine a été limité. La régulation se produit lorsque les IOPS du nœud de données dépassent la limite maximale autorisée du volume EBS ou de l' EC2 instance du nœud de données. Pour plus d'informations sur les IOPS des instances, consultez la section Instances optimisées pour Amazon EBS. Pour plus d'informations sur les volumes IOPS, consultez la section Types de volumes Amazon EBS. Statistiques pertinentes : Minimum, Maximum
`HighSwapUsage`	La valeur 1 indique que l'échange dû à des erreurs de page a potentiellement provoqué des pics d'utilisation du disque sous-jacent au cours d'une période donnée. Statistiques pertinentes : Maximum

Métriques du nœud principal dédié

Amazon OpenSearch Service fournit les métriques suivantes pour les nœuds principaux dédiés.

Métrique	Description
`MasterCPUUtilization`	Pourcentage maximal de ressources UC utilisées par les nœuds principaux dédiés. Nous vous recommandons d'augmenter la taille du type d'instance lorsque cette métrique atteint 60 %. Statistiques pertinentes : Maximum
`MasterFreeStorageSpace`	Cette métrique n'est pas pertinente et peut être ignorée. Le service n'utilise pas de nœuds principaux comme nœuds de données.
`MasterJVMMemoryPressure`	Pourcentage maximal du tas Java utilisé pour tous les nœuds maîtres dédiés dans le cluster. Nous vous recommandons de migrer vers un type d'instance plus grand lorsque cette métrique atteint 85 %. Statistiques pertinentes : Maximum **Note** La logique de cette métrique a changé dans le logiciel de service R20220323. Pour plus d'informations, veuillez consulter les notes de mise à jour.
`MasterOldGenJVMMemoryPressure`	Le pourcentage maximum du tas Java utilisé pour l'« ancienne génération » par nœud principal. Statistiques pertinentes : Maximum
`MasterCPUCreditBalance`	Crédits UC restants disponibles pour les nœuds maîtres dédiés dans le cluster. Un crédit UC fournit les performances d'un cœur UC complet pendant une minute. Pour plus d'informations, consultez la section Crédits CPU dans le manuel Amazon EC2 Developer Guide. Cette métrique est disponible uniquement pour les types d'instance T2.

Métrique	Description
	Statistiques pertinentes : Minimum
`MasterReachableFromNode`	Vérification de l'état pour les exceptions `MasterNotDiscovered` . La valeur 1 indique un comportement normal. La valeur 0 indique que `/_cluster/health/` échoue. Les défaillances signifient que le nœud principal est inaccessible depuis le nœud source. Ils sont généralement la conséquence d'un problème de connectivité réseau ou de AWS dépendance. Statistiques pertinentes : Maximum
`MasterSysMemoryUtilization`	Pourcentage de mémoire du nœud principal actuellement utilisée. Statistiques pertinentes : Maximum

Mesures dédiées concernant le nœud de coordinateur

Amazon OpenSearch Service fournit les métriques suivantes pour les nœuds de coordination dédiés.

Métrique	Description
`CoordinatorCPUUtilization`	Le pourcentage maximum des ressources d'UC utilisées par les nœuds de coordination dédiés. Nous recommandons d'augmenter la taille du type d'instance lorsque cette métrique atteint 80 %. Statistiques pertinentes : Maximum
`CoordinatorJVMMemoryPressure`	Le pourcentage maximum du tas Java utilisé pour tous les nœuds de coordination dédiés dans le cluster. Nous vous recommandons de migrer vers un type d'instance plus grand lorsque cette métrique atteint 85 %. Statistiques pertinentes : Maximum
`CoordinatorOldGenJVMMemoryPressure`	Le pourcentage maximum du tas Java utilisé pour l'« ancienne génération » par nœud principal.

Métrique	Description
	Statistiques pertinentes : Maximum
`CoordinatorSysMemoryUtilization`	Le pourcentage de la mémoire du nœud coordinateur qui est en cours d'utilisation. Statistiques pertinentes : Maximum
`CoordinatorFreeStorageSpace`	Cette métrique indique que le service n'utilise pas de nœuds coordinateurs comme nœuds de données.

Métriques du volume EBS

Amazon OpenSearch Service fournit les métriques suivantes pour les volumes EBS.

Métrique	Description
`ReadLatency`	Latence, en secondes, pour les opérations de lecture sur les volumes EBS. Cette métrique est également disponible pour les nœuds individuels. Statistiques pertinentes : minimum, maximum, moyenne
`WriteLatency`	Latence, en secondes, pour les opérations d'écriture sur les volumes EBS. Cette métrique est également disponible pour les nœuds individuels. Statistiques pertinentes : minimum, maximum, moyenne
`ReadThroughput`	Débit, en octets par seconde, pour les opérations de lecture sur les volumes EBS. Cette métrique est également disponible pour les nœuds individuels. Statistiques pertinentes : minimum, maximum, moyenne
`ReadThroughputMicroBursting`	Débit, en octets par seconde, pour les opérations de lecture sur les volumes EBS lorsque le microbursting est pris en compte. Cette métrique est également disponible pour les nœuds individuels. Le micro-écl

Métrique	Description
	atement se produit lorsqu'un volume EBS enregistre des IOPS ou un débit élevés pendant des périodes nettement plus courtes (moins d'une minute). Statistiques pertinentes : minimum, maximum, moyenne
`WriteThroughput`	Débit, en octets par seconde, pour les opérations d'écriture sur les volumes EBS. Cette métrique est également disponible pour les nœuds individuels. Statistiques pertinentes : minimum, maximum, moyenne
`WriteThro ughputMic roBursting`	Débit, en octets par seconde, pour les opérations d'écriture sur des volumes EBS lorsque le microbursting est pris en compte. Cette métrique est également disponible pour les nœuds individuels. Le micro-écl atement se produit lorsqu'un volume EBS enregistre des IOPS ou un débit élevés pendant des périodes nettement plus courtes (moins d'une minute). Statistiques pertinentes : minimum, maximum, moyenne
`DiskQueueDepth`	Nombre de demandes d'entrée et de sortie (I/O) en attente pour un volume EBS. Statistiques pertinentes : minimum, maximum, moyenne
`ReadIOPS`	Nombre d'opérations d'entrée et de sortie (I/O) par seconde pour les opérations de lecture sur les volumes EBS. Cette métrique est également disponible pour les nœuds individuels. Statistiques pertinentes : minimum, maximum, moyenne

Métrique	Description
`ReadIOPSMicroBursting`	Nombre d'opérations d'entrée et de sortie (I/O) par seconde pour les opérations de lecture sur les volumes EBS lorsque le microbursting est pris en compte. Cette métrique est également disponible pour les nœuds individuels. Le micro-éclatement se produit lorsqu'un volume EBS enregistre des IOPS ou un débit élevés pendant des périodes nettement plus courtes (moins d'une minute). Statistiques pertinentes : minimum, maximum, moyenne
`WriteIOPS`	Nombre d'opérations d'entrée et de sortie (I/O) par seconde pour les opérations d'écriture sur les volumes EBS. Cette métrique est également disponible pour les nœuds individuels. Statistiques pertinentes : minimum, maximum, moyenne
`WriteIOPSMicroBursting`	Nombre d'opérations d'entrée et de sortie (I/O) par seconde pour les opérations d'écriture sur les volumes EBS lorsque le microbursting est pris en compte. Cette métrique est également disponible pour les nœuds individuels. Le micro-éclatement se produit lorsqu'un volume EBS enregistre des IOPS ou un débit élevés pendant des périodes nettement plus courtes (moins d'une minute). Statistiques pertinentes : minimum, maximum, moyenne
`BurstBalance`	Le pourcentage de crédits d'entrée et de sortie (E/S) restant dans le compartiment de débordement pour un volume EBS. Une valeur de 100 signifie que le volume a accumulé le nombre maximum de crédits. Si ce pourcentage tombe en dessous de 70 %, consultez the section called “Solde de débordement EBS faible”. Le solde de rafale reste à 0 pour les domaines avec des types de volumes gp3 et les domaines avec des volumes gp2 dont la taille de volume est supérieure à 1 000 Gio. Statistiques pertinentes : minimum, maximum, moyenne

Métrique	Description
`VolumeStalledIOcheck`	État de vos volumes EBS afin de déterminer à quel moment ils sont dégradés. La métrique est une valeur binaire qui renvoie un statut 0 (réussite) ou un statut 1 (échec) ou un statut 1 (échec) ou un statut 1 (échec) ou un statut 0 (réussite) ou un statut 1 (échec) ou un statut 1 (échec) ou un statut 1 (échec) ou un statut 1 (échec `VolumeStalledIOcheck` est également disponible pour les nœuds individuels. Statistiques pertinentes : minimum, maximum, moyenne

Métriques des instances

Amazon OpenSearch Service fournit les métriques suivantes pour chaque instance d'un domaine. OpenSearch Service agrège également ces métriques d'instance pour fournir des informations sur l'état global du cluster. Vous pouvez vérifier ce comportement à l'aide de la statistique Nombre d'échantillons dans la console. Notez que chaque métrique du tableau suivant inclut des statistiques concernant le nœud et le cluster.

> **Important**
>
> Les groupes de threads utilisés pour traiter les appels à l'API `_index` varient en fonction de la version d'Elasticsearch. Elasticsearch 1.5 et 2.3 utilisent le groupe de threads d'index. Elasticsearch 5. x, 6.0 et 6.2 utilisent le groupe de threads en bloc. OpenSearch et Elasticsearch 6.3 et versions ultérieures utilisent le groupe de threads d'écriture. Actuellement, la console OpenSearch de service n'inclut pas de graphique pour le groupe de threads en bloc.
> Utilisez `GET _cluster/settings?include_defaults=true` pour vérifier la taille du groupe de threads et de la file d'attente de votre cluster.

Métrique	Description
`FetchLatency`	Différence de temps total, en millisecondes, prise par toutes les opérations d'extraction de partitions dans un nœud entre la minute N et la minute (N - 1).

Métrique	Description
	Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum
`FetchRate`	Nombre total d'opérations de récupération de partitions par minute pour toutes les partitions d'un nœud de données. Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum, Somme
`ScrollTotal`	Nombre total d'opérations de défilement de partitions par minute pour toutes les partitions d'un nœud de données. Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum, Somme
`ScrollCurrent`	Nombre d'opérations de défilement de partitions qui sont en cours d'exécution. Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum, Somme
`OpenContexts`	Nombre de contextes de recherche ouverts Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum, Somme

Métrique	Description
ThreadCount	Nombre total de threads actuellement utilisés par le OpenSearch processus. Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum, Somme
ShardReactivateCount	Nombre total de fois où toutes les partitions ont été activées depuis un état inactif. Statistiques pertinentes concernant le nœud : somme Statistiques pertinentes concernant le cluster : somme
ConcurrentSearchRate	Nombre total de demandes de recherche utilisant une recherche par segment simultanée par minute pour toutes les partitions d'un nœud de données. Un même appel à l'API `_search` peut renvoyer les résultats de nombreuses partitions différentes. Si cinq de ces partitions se trouvent sur un même nœud, celui-ci renvoie 5 pour cette métrique, même si le client n'a effectué qu'une seule demande. Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum, Somme
ConcurrentSearchLatency	Différence de temps total, en millisecondes, prise par toutes les recherches utilisant une recherche par segment simultanée dans un nœud entre la minute N et la minute (N-1). Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum

Métrique	Description
`IndexingLatency`	Différence de temps total, en millisecondes, prise par toutes les opérations d'indexation dans un nœud entre la minute N et la minute (N-1). Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum
`IndexingRate`	Nombre d'opérations d'indexation par minute. Un seul appel à l'API `_bulk` qui ajoute deux documents et en met deux à jour compte pour quatre opérations, qui peuvent être étendues à un ou plusieurs nœuds. Si cet index possède une ou plusieurs répliques et se trouve sur un OpenSearch domaine sans instances optimisées, les autres nœuds du cluster enregistrent également un total de quatre opérations d'indexation. Pour les OpenSearch domaines dotés d'instances optimisées, les autres nœuds dotés de répliques n'enregistrent aucune opération. Les suppressions de documents ne sont pas prises en compte dans cette métrique. Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum, Somme
`SearchLatency`	Différence de temps total, en millisecondes, prise par toutes les recherches dans un nœud entre la minute N et la minute (N-1). Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum

Métrique	Description
SearchRate	Nombre total de demandes de recherche par minute pour toutes les partitions d'un nœud de données. Un même appel à l'API _search peut renvoyer les résultats de nombreuses partition s différentes. Si cinq de ces partitions se trouvent sur un même nœud, celui-ci renvoie 5 pour cette métrique, même si le client n'a effectué qu'une seule demande. Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum, Somme
SegmentCount	Nombre de segments sur un nœud de données. Plus vous avez de segments, plus la recherche prend du temps. OpenSearch fusionne parfois les segments plus petits en un segment plus grand. Statistiques pertinentes concernant le nœud : Maximum, Moyenne Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
SysMemoryUtilizati on	Pourcentage de mémoire de l'instance actuellement utilisée. Les valeurs élevées de cette métrique sont normales et ne représentent généralement pas un problème lié à votre cluster. Pour obtenir un meilleur indicateur des éventuels problèmes de performance et de stabilité, veuillez consulter la métrique JVMMemoryPressure . Statistiques pertinentes concernant le nœud : Minimum, Maximum, Moyenne Statistiques pertinentes concernant le cluster : Minimum, Maximum, Moyenne

Métrique	Description
JVMGCYoungCollectionCount	Nombre de fois que le nettoyage de la « jeune génération » a été exécuté. Un nombre important et évolutif d'exécutions est une part normale des opérations de cluster. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
JVMGCYoungCollectionTime	Temps, en millisecondes, que le cluster a consacré à l'exécution d'un nettoyage de la « jeune génération ». Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
JVMGCOldCollectionCount	Nombre de fois que le nettoyage de l'« ancienne génération » a été exécuté. Dans un cluster doté de ressources suffisantes, ce nombre doit rester faible et évoluer peu fréquemment. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
JVMGCOldCollectionTime	Temps, en millisecondes, que le cluster a consacré à l'exécution d'un nettoyage de l'« ancienne génération ». Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne

Métrique	Description
OpenSearchDashboardsConcurrentConnections	Nombre de connexions simultanées actives aux OpenSearch Tableaux de bord. Si ce nombre reste élevé, envisagez de mettre votre cluster à l'échelle. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
OpenSearchDashboardsHealthyNode	Surveillance de l'état du nœud OpenSearch Tableaux de bord individuel. La valeur 1 indique un comportement normal. La valeur 0 indique que les Tableaux de bord sont inaccessibles. Statistiques pertinentes concernant le nœud : Minimum Statistiques pertinentes concernant le cluster : Minimum, Maximum, Moyenne
OpenSearchDashboardsHeapTotal	Quantité de mémoire de tas allouée aux OpenSearch tableaux de bord, en Mio. Différents types d' EC2 instance peuvent affecter l'allocation de mémoire exacte. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
OpenSearchDashboardsHeapUsed	Quantité absolue de mémoire de tas utilisée par les OpenSearch tableaux de bord, en Mio. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne

Métrique	Description
OpenSearchDashboardsHeapUtilization	Le pourcentage maximum de mémoire de tas disponible utilisé par OpenSearch Tableaux de bord. Si cette valeur dépasse 80 %, envisagez de mettre votre cluster à l'échelle. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Minimum, Maximum, Moyenne
OpenSearchDashboardsOS1MinuteLoad	Charge moyenne de l'UC d'une minute pour les OpenSearch Tableaux de bord. La charge du processeur devrait idéalement rester inférieure à 1,00. Les pics temporaires n'ont rien d'inhabituel, mais nous vous recommandons d'augmenter la taille du type d'instance si cette métrique est systématiquement supérieure à 1,00. Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum
OpenSearchDashboardsRequestTotal	Nombre total de demandes HTTP faites aux OpenSearch Tableaux de bord. Si votre système est lent ou si vous constatez un nombre élevé de demandes des Tableaux de bord, envisagez d'augmenter la taille du type d'instance. Statistiques pertinentes concernant le nœud : Somme Statistiques pertinentes concernant le cluster : Somme
OpenSearchDashboardsResponseTimesMaxInMillis	Durée maximale, en millisecondes, nécessaire pour que les Tableaux de OpenSearch bord Dashboards répondent à une demande. Si les demandes mettent systématiquement beaucoup de temps à renvoyer des résultats, envisagez d'augmenter la taille du type d'instance. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Maximum, Moyenne

Métrique	Description
SearchTaskCancelled	Le nombre d'annulations de nœuds coordinateurs. Statistiques pertinentes concernant le nœud : Somme Statistiques pertinentes concernant le cluster : Somme
SearchShardTaskCancelled	Le nombre d'annulations de nœuds de données. Statistiques pertinentes concernant le nœud : Somme Statistiques pertinentes concernant le cluster : Somme
ThreadpoolForce_mergeQueue	Nombre de tâches mises en file d'attente dans le groupe de threads de fusion forcée. Si la taille de la file d'attente reste constamment élevée, envisagez de mettre votre cluster à l'échelle. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
ThreadpoolForce_mergeRejected	Nombre de tâches rejetées dans le groupe de threads de fusion forcée. Si ce nombre augmente constamment, envisagez de mettre votre cluster à l'échelle. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme
ThreadpoolForce_mergeThreads	Taille du groupe de threads de fusion forcée. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Moyenne, Somme

Métrique	Description
ThreadpoolIndexQueue	Nombre de tâches mises en file d'attente dans le groupe de threads d'index. Si la taille de la file d'attente reste constamment élevée, envisagez de mettre votre cluster à l'échelle. La taille maximale de la file d'attente d'index est de 200. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
ThreadpoolIndexRejected	Nombre de tâches rejetées dans le groupe de threads d'index. Si ce nombre augmente constamment, envisagez de mettre votre cluster à l'échelle. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme
ThreadpoolIndexThreads	Taille du groupe de threads d'index. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Moyenne, Somme
ThreadpoolSearchQueue	Nombre de tâches mises en file d'attente dans le groupe de threads de recherche. Si la taille de la file d'attente reste constamment élevée, envisagez de mettre votre cluster à l'échelle. La taille maximale de la file d'attente de recherche est de 1 000. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne

Métrique	Description
ThreadpoolSearchRejected	Nombre de tâches rejetées dans le groupe de threads de recherche. Si ce nombre augmente constamment, envisagez de mettre votre cluster à l'échelle. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme
ThreadpoolSearchThreads	Taille du groupe de threads de recherche. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Moyenne, Somme
Threadpoolsql-workerQueue	Nombre de tâches mises en file d'attente dans le groupe de threads de recherche SQL. Si la taille de la file d'attente reste constamment élevée, envisagez de mettre votre cluster à l'échelle. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
Threadpoolsql-workerRejected	Nombre de tâches rejetées dans le groupe de threads de recherche SQL. Si ce nombre augmente constamment, envisagez de mettre votre cluster à l'échelle. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme
Threadpoolsql-workerThreads	Taille du groupe de threads de recherche SQL. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Moyenne, Somme

Métrique	Description
ThreadpoolBulkQueue	Nombre de tâches mises en file d'attente dans le groupe de threads en bloc. Si la taille de la file d'attente reste constamment élevée, envisagez de mettre votre cluster à l'échelle. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
ThreadpoolBulkRejected	Nombre de tâches rejetées dans le groupe de threads en bloc. Si ce nombre augmente constamment, envisagez de mettre votre cluster à l'échelle. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme
ThreadpoolBulkThreads	Taille du groupe de threads en bloc. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Moyenne, Somme
ThreadpoolIndexSearcherQueue	Nombre de tâches mises en file d'attente dans le groupe de threads du chercheur d'index. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
ThreadpoolIndexSearcherRejected	Nombre de tâches rejetées dans le groupe de threads du chercheur d'index. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme

Métrique	Description
`ThreadpoolIndexSearcherThreads`	Taille du groupe de threads du chercheur d'index. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Moyenne, Somme
`ThreadpoolWriteThreads`	Taille du groupe de threads d'écriture. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Moyenne, Somme
`ThreadpoolWriteQueue`	Nombre de tâches mises en file d'attente dans le groupe de threads d'écriture. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Moyenne, Somme
`ThreadpoolWriteRejected`	Nombre de tâches rejetées dans le groupe de threads d'écriture. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Moyenne, Somme **Note** La taille de la file d'attente d'écriture par défaut ayant été augmentée de 200 à 10 000 dans la version 7.1, cette métrique n'est plus le seul indicateur de rejets de OpenSearch Service. Utilisez les métriques `CoordinatingWriteRejected` , `PrimaryWriteRejected` et `ReplicaWriteRejected` pour surveiller les rejets dans la version 7.1 et les versions ultérieures.

Métrique	Description
`CoordinatingWriteRejected`	Le nombre total de rejets s'est produit sur le nœud de coordination en raison de la pression d'indexation depuis le dernier démarrage du processus OpenSearch de service. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Moyenne, Somme Cette métrique est disponible dans la version 7.1 et les versions ultérieures.
`PrimaryWriteRejected`	Le nombre total de rejets s'est produit sur les partitions principales en raison de la pression d'indexation depuis le dernier démarrage du processus de OpenSearch service. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Moyenne, Somme Cette métrique est disponible dans la version 7.1 et les versions ultérieures.
`ReplicaWriteRejected`	Le nombre total de rejets s'est produit sur les partitions de réplica en raison de la pression d'indexation depuis le dernier démarrage du processus de OpenSearch service. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Moyenne, Somme Cette métrique est disponible dans la version 7.1 et les versions ultérieures.

Métrique	Description
`WorkloadManagementEnabled`	Indique si la fonctionnalité de gestion de charge de travail est activée. Une valeur de 1 signifie qu'il est activé, et une valeur de 0 signifie qu'il est désactivé. `monitor_only` Statistiques pertinentes concernant le nœud : Minimum Statistiques pertinentes concernant le cluster : Moyenne, Somme Cette métrique est disponible dans la version 7.1 et les versions ultérieures.
`SoftQueryGroupCount`	Nombre de groupes de requêtes en mode logiciel dans le domaine. Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum, Somme Cette métrique est disponible dans la version 7.1 et les versions ultérieures.
`EnforcedQueryGroupCount`	Nombre de groupes de requêtes en mode forcé dans le domaine. Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum, Somme Cette métrique est disponible dans la version 7.1 et les versions ultérieures.

UltraWarm métriques

Amazon OpenSearch Service fournit les métriques suivantes pour les UltraWarmnœuds.

Métrique	Description
`WarmCPUUtilization`	Pourcentage d'utilisation du processeur pour UltraWarm les nœuds du cluster. Maximum indique le nœud avec l'utilisation la plus élevée du processeur. Moyenne représente tous les UltraWarm nœuds du cluster. Cette métrique est également disponible pour les UltraWarm nœuds individuels. Statistiques pertinentes : Maximum, Moyenne
`WarmFreeStorageSpace`	Quantité d'espace de stockage à chaud gratuit en Mo. Parce UltraWarm qu'Amazon S3 plutôt que des disques attachés, `Sum` c'est la seule statistique pertinente. Vous devez laisser la période à une minute pour obtenir une valeur précise. Statistiques pertinentes : somme
`WarmSearchableDocuments`	Nombre total de documents consultables sur tous les index à chaud du cluster. Vous devez laisser la période à une minute pour obtenir une valeur précise. Statistiques pertinentes : somme
`WarmSearchLatency`	Différence de temps total, en millisecondes, prise par toutes les recherches UltraWarm entre la minute N et la minute (N-1). Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Moyenne, Maximum
`WarmSearchRate`	Nombre total de demandes de recherche par minute pour tous les partitions d'un UltraWarm nœud. Un même appel à l'API `_search` peut renvoyer les résultats de nombreuses partitions différentes. Si cinq de ces partitions se trouvent sur un même nœud, celui-ci renvoie 5 pour cette métrique, même si le client n'a effectué qu'une seule demande. Statistiques pertinentes concernant le nœud : Moyenne

Métrique	Description
	Statistiques pertinentes concernant le cluster : Moyenne, Maximum, Somme
`WarmStorageSpaceUtilization`	Volume total d'espace de stockage à chaud, en Mio, utilisé par le cluster. Statistiques pertinentes : Maximum
`HotStorageSpaceUtilization`	Volume total d'espace de stockage hot utilisé par le cluster. Statistiques pertinentes : Maximum
`WarmSysMemoryUtilization`	Pourcentage de mémoire du nœud à chaud actuellement utilisée. Statistiques pertinentes : Maximum
`HotToWarmMigrationQueueSize`	Nombre d'index actuellement en attente de migration du stockage hot vers le stockage à chaud. Statistiques pertinentes : Maximum
`WarmToHotMigrationQueueSize`	Nombre d'index actuellement en attente de migration du stockage à chaud vers le stockage hot. Statistiques pertinentes : Maximum
`HotToWarmMigrationFailureCount`	Nombre total de migrations hot vers à chaud ayant échoué. Statistiques pertinentes : somme
`HotToWarmMigrationForceMergeLatency`	Latence moyenne de l'étape de fusion forcée du processus de migration . Si cette étape se révèle particulièrement chronophage, envisagez d'augmenter `index.ultrawarm.migration.force_merge.max_num_segments` . Statistiques pertinentes : Moyenne

Métrique	Description
HotToWarmMigrationSnapshotLatency	Latence moyenne de l'étape d'instantané du processus de migration. Si cette étape se révèle particulièrement chronophage, assurez-vous que vos partitions sont correctement dimensionnées et distribuées dans tout le cluster. Statistiques pertinentes : Moyenne
HotToWarmMigrationProcessingLatency	Latence moyenne des migrations hot vers à chaud réussies, sans compter le temps passé dans la file d'attente. Cette valeur correspond à la durée nécessaire pour terminer les étapes de fusion forcée, d'instantané et de déplacement de partitions du processus de migration. Statistiques pertinentes : Moyenne
HotToWarmMigrationSuccessCount	Nombre total de migrations hot vers à chaud réussies. Statistiques pertinentes : somme
HotToWarmMigrationSuccessLatency	Latence moyenne des migrations hot vers à chaud, en comptant le temps passé dans la file d'attente. Statistiques pertinentes : Moyenne
WarmThreadpoolSearchThreads	Taille du groupe de threads UltraWarm de recherche. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Moyenne, Somme
WarmThreadpoolSearchRejected	Nombre de tâches rejetées dans le groupe UltraWarm de threads de recherche. Si ce nombre augmente continuellement, envisagez d'ajouter plus de UltraWarm nœuds. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme

Métrique	Description
`WarmThreadpoolSearchQueue`	Nombre de tâches mises en file d'attente dans le groupe de threads UltraWarm de recherche. Si la taille de la file d'attente reste constamment élevée, envisagez d'ajouter plus de UltraWarm nœuds. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
`WarmJVMMemoryPressure`	Le pourcentage maximum du tas Java utilisé pour les UltraWarm nœuds. Statistiques pertinentes : Maximum **Note** La logique de cette métrique a changé dans le logiciel de service R20220323. Pour plus d'informations, veuillez consulter les notes de mise à jour.
`WarmOldGenJVMMemoryPressure`	Le pourcentage maximum du tas Java utilisé pour l' « ancienne génération » par UltraWarm nœud. Statistiques pertinentes : Maximum
`WarmJVMGCYoungCollectionCount`	Nombre de fois que le nettoyage de la « jeune génération » a été exécuté sur UltraWarm des nœuds. Un nombre important et évolutif d'exécutions est une part normale des opérations de cluster. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne

Métrique	Description
`WarmJVMGCYoungCollectionTime`	Temps, en millisecondes, que le cluster a consacré à l'exécution d'un nettoyage de la « jeune génération » sur des nœuds. UltraWarm Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
`WarmJVMGCOldCollectionCount`	Nombre de fois que le nettoyage de l' « ancienne génération » a été exécuté sur UltraWarm des nœuds. Dans un cluster doté de ressources suffisantes, ce nombre doit rester faible et évoluer peu fréquemment. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
`WarmConcurrentSearchRate`	Nombre total de demandes de recherche utilisant une recherche par segment simultanée par minute pour toutes les partitions d'un UltraWarm nœud. Un même appel à l'API `_search` peut renvoyer les résultats de nombreuses partitions différentes. Si cinq de ces partitions se trouvent sur un même nœud, celui-ci renvoie 5 pour cette métrique, même si le client n'a effectué qu'une seule demande. Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
`WarmConcurrentSearchLatency`	Différence de temps total, en millisecondes, prise par toutes les recherches utilisant une recherche par segment simultanée dans un UltraWarm nœud entre la minute N et la minute (N-1). Statistiques pertinentes concernant le nœud : Moyenne Statistiques pertinentes concernant le cluster : Maximum, Moyenne

Métrique	Description
`WarmThreadpoolIndexSearcherQueue`	Nombre de tâches mises en file d'attente dans le groupe de threads du chercheur d' UltraWarm index. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme, Maximum, Moyenne
`WarmThreadpoolIndexSearcherRejected`	Nombre de tâches rejetées dans le groupe de threads de recherche d' UltraWarm index. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : Somme
`WarmThreadpoolIndexSearcherThreads`	Taille du groupe de threads du chercheur d' UltraWarm index. Statistiques pertinentes concernant le nœud : Maximum Statistiques pertinentes concernant le cluster : somme

Métriques de stockage à froid

Amazon OpenSearch Service fournit les métriques suivantes pour le stockage à froid.

Métrique	Description
`ColdStorageSpaceUtilization`	Volume total d'espace de stockage à froid, en Mio, utilisé par le cluster. Statistiques pertinentes : maximum
`ColdToWarmMigrationFailureCount`	Nombre total de migrations à froid vers à chaud ayant échoué. Statistiques pertinentes : somme
`ColdToWarmMigrationLatency`	Temps nécessaire pour mener à bien les migrations à froid vers à chaud.

Métrique	Description
	Statistiques pertinentes : Moyenne
`ColdToWarmMigrationQueueSize`	Nombre d'index actuellement en attente de migration du stockage à froid vers le stockage à chaud. Statistiques pertinentes : Maximum
`ColdToWarmMigrationSuccessCount`	Nombre total de migrations à froid vers à chaud réussies. Statistiques pertinentes : somme
`WarmToColdMigrationFailureCount`	Nombre total de migrations à chaud vers à froid ayant échoué. Statistiques pertinentes : somme
`WarmToColdMigrationLatency`	Temps nécessaire pour mener à bien les migrations à chaud vers à froid. Statistiques pertinentes : Moyenne
`WarmToColdMigrationQueueSize`	Nombre d'index actuellement en attente de migration du stockage à chaud vers le stockage à froid. Statistiques pertinentes : Maximum
`WarmToColdMigrationSuccessCount`	Nombre total de migrations à chaud vers à froid réussies. Statistiques pertinentes : somme

OR1 métriques

Amazon OpenSearch Service fournit les statistiques suivantes pour les OR1instances.

Métrique	Description
`RemoteStorageUsedSpace`	Volume total d'espace Amazon S3, en Mio, utilisé par le cluster. Statistiques pertinentes : somme

Métrique	Description
`RemoteStorageWriteRejected`	Nombre total de demandes rejetées sur les partitions principales en raison du stockage à distance et de la pression de réplication. Ceci est calculé à partir du dernier démarrage du processus de OpenSearch service. Statistiques pertinentes : somme
`ReplicationLagMaxTime`	Durée, en millisecondes, pendant laquelle les partitions de réplica restent derrière les partitions principales. Statistiques pertinentes : Maximum

Métriques d'alerte

Amazon OpenSearch Service fournit les métriques suivantes pour les alertes.

Métrique	Description
`AlertingDegraded`	Une valeur de 1 signifie que l'index d'alerte est rouge ou qu'un ou plusieurs nœuds ne sont pas prévus. La valeur 0 indique un comportement normal. Statistiques pertinentes : Maximum
`AlertingIndexExists`	Une valeur de 1 signifie que l'index `.opensearch-alerting-config` existe. Une valeur de 0 signifie que ce n'est pas le cas. Tant que vous n'utilisez pas la fonction d'alerte pour la première fois, cette valeur reste 0. Statistiques pertinentes : Maximum
`AlertingIndexStatus.green`	État de santé de l'index. Une valeur de 1 signifie vert. Une valeur de 0 signifie que l'index n'existe pas ou n'est pas vert. Statistiques pertinentes : Maximum

Métrique	Description
`AlertingIndexStatus.red`	État de santé de l'index. Une valeur de 1 signifie rouge. Une valeur de 0 signifie que l'index n'existe pas ou n'est pas rouge. Statistiques pertinentes : Maximum
`AlertingIndexStatus.yellow`	État de santé de l'index. Une valeur de 1 signifie jaune. Une valeur de 0 signifie que l'index n'existe pas ou n'est pas jaune. Statistiques pertinentes : Maximum
`AlertingNodesNotOnSchedule`	Une valeur de 1 signifie que certaines tâches ne sont pas exécutées dans les délais prévus. La valeur 0 signifie que tous les travaux d'alerte sont exécutés selon les prévisions (ou qu'il n'existe aucun travail d'alerte). Vérifiez la console OpenSearch de service ou adressez une `_nodes/stats` demande pour voir si les nœuds affichent une utilisation élevée des ressources. Statistiques pertinentes : Maximum
`AlertingNodesOnSchedule`	La valeur 1 signifie que toutes les tâches d'alerte sont exécutées selon les prévisions (ou qu'il n'existe pas de tâches d'alerte). Une valeur de 0 signifie que certaines tâches ne sont pas exécutées dans les délais prévus. Statistiques pertinentes : Maximum
`AlertingScheduledJobEnabled`	Une valeur de 1 signifie que le paramètre de cluster `opensearch.scheduled_jobs.enabled` a la valeur true. La valeur 0 correspond à la valeur « false » et signifie que les tâches planifiées sont désactivées. Statistiques pertinentes : Maximum

Métriques de détection d'anomalies

Amazon OpenSearch Service fournit les métriques suivantes pour la détection d'anomalies.

Métrique	Description
`ADPluginUnhealthy`	Une valeur de 1 signifie que le plugin de détection d'anomalies ne fonctionne pas correctement, soit en raison d'un nombre élevé de défaillances, soit parce que l'un des indices qu'il utilise est rouge. Une valeur de 0 indique que le plugin fonctionne comme prévu. Statistiques pertinentes : Maximum
`ADExecuteRequestCount`	Nombre de demandes pour détecter des anomalies. Statistiques pertinentes : somme
`ADExecuteFailureCount`	Nombre de demandes ayant échoué pour détecter des anomalies. Statistiques pertinentes : somme
`ADHCExecuteFailureCount`	Nombre de demandes visant à détecter des anomalies à cardinalité élevée ayant échoué. Statistiques pertinentes : somme
`ADHCExecuteRequestCount`	Nombre de demandes visant à détecter des anomalies à cardinalité élevée. Statistiques pertinentes : somme
`ADAnomalyResultsIndexStatusIndexExists`	Une valeur de 1 signifie l'index vers lequel l'alias `.opensearch-anomaly-results` pointe existe. Tant que vous n'avez pas utilisé la détection d'anomalies, cette valeur correspond à 0. Statistiques pertinentes : Maximum
`ADAnomalyResultsIndexStatus.red`	La valeur 1 signifie que l'index vers lequel l'alias `.opensearch-anomaly-results` pointe est rouge. Une valeur 0 signifie que ce n'est pas le cas. Tant que vous n'avez pas utilisé la détection d'anomalies, cette valeur correspond à 0. Statistiques pertinentes : Maximum

Métrique	Description
`ADAnomaly Detectors IndexStat usIndexExists`	Une valeur de 1 signifie que l'index `.opensearch-anomaly-detectors` existe. Une valeur de 0 signifie que ce n'est pas le cas. Tant que vous n'avez pas utilisé la détection d'anomalies, cette valeur correspond à 0. Statistiques pertinentes : Maximum
`ADAnomaly Detectors IndexStat us.red`	Une valeur de 1 signifie que l'index `.opensearch-anomaly-detectors` est rouge. Une valeur 0 signifie que ce n'est pas le cas. Tant que vous n'avez pas utilisé la détection d'anomalies, cette valeur correspond à 0. Statistiques pertinentes : Maximum
`ADModelsC heckpoint IndexStat usIndexExists`	Une valeur de 1 signifie que l'index `.opensearch-anomaly-checkpoints` existe. Une valeur de 0 signifie que ce n'est pas le cas. Tant que vous n'avez pas utilisé la détection d'anomalies, cette valeur correspond à 0. Statistiques pertinentes : Maximum
`ADModelsC heckpoint IndexStat us.red`	Une valeur de 1 signifie que l'index `.opensearch-anomaly-checkpoints` est rouge. Une valeur 0 signifie que ce n'est pas le cas. Tant que vous n'avez pas utilisé la détection d'anomalies, cette valeur correspond à 0. Statistiques pertinentes : Maximum

Métriques de recherche asynchrone

Amazon OpenSearch Service fournit les métriques suivantes pour la recherche asynchrone.

Statistiques de nœud coordinateur de recherche asynchrone (par nœud de coordinateur)

Métrique	Description
AsynchronousSearchSubmissionRate	Nombre de recherches asynchrones envoyées au cours de la dernière minute.
AsynchronousSearchInitializedRate	Nombre de recherches asynchrones initialisées au cours de la dernière minute.
AsynchronousSearchRunningCurrent	Nombre de recherches asynchrones en cours d'exécution.
AsynchronousSearchCompletionRate	Nombre de recherches asynchrones ayant abouti au cours de la dernière minute.
AsynchronousSearchFailureRate	Nombre de recherches asynchrones ayant abouti et échoué au cours de la dernière minute.
AsynchronousSearchPersistRate	Nombre de recherches asynchrones ayant perduré au cours de la dernière minute.
AsynchronousSearchPersistFailedRate	Nombre de recherches asynchrones n'ayant pas perduré au cours de la dernière minute.
AsynchronousSearchRejected	Nombre total de recherches asynchrones rejetées depuis le démarrage du nœud.

Métrique	Description
AsynchronousSearchCancelled	Nombre total de recherches asynchrones annulées depuis le démarrage du nœud.
AsynchronousSearchMaxRunningTime	Durée de la plus longue recherche asynchrone en cours d'exécution sur un nœud au cours de la dernière minute.

Statistiques concernant le cluster en matière de recherche asynchrone

Métrique	Description
AsynchronousSearchStoreHealth	État de santé du magasin dans l'index persistant (rouge/non-rouge) au cours de la dernière minute.
AsynchronousSearchStoreSize	Taille de l'index système de toutes les partitions au cours de la dernière minute.
AsynchronousSearchStoredResponseCount	Nombre de réponses stockées dans l'index système au cours de la dernière minute.

Auto-Tune

Amazon OpenSearch Service fournit les statistiques suivantes pour Auto-Tune.

Métrique	Description
AutoTuneChangesHistoryHeapSize	Historique des modifications en MiB pour les valeurs de réglage de la taille des tas.

Métrique	Description
`AutoTuneChangesHistoryJVMYoungGenArgs`	Historique des modifications pour les YongGen arguments JVM.
`AutoTuneFailed`	Un booléen qui indique si le changement Auto-Tune a échoué.
`AutoTuneSucceeded`	Un booléen qui indique si le changement Auto-Tune a réussi.
`AutoTuneValue`	L'historique des modifications de la file d'attente (nombre) et les réglages du cache changent l'historique des modifications (en MiB) pour des modifications non perturbatrices.

Multi-AZ avec métriques

Amazon OpenSearch Service fournit les mesures suivantes pour le mode Multi-AZ avec mode veille.

Mesures au niveau des nœuds pour les nœuds de données dans les zones de disponibilité actives

Métrique	Description
`CPUUtilization`	Pourcentage d'utilisation du processeur pour les nœuds de données du cluster. Maximum indique le nœud avec l'utilisation la plus élevée du processeur. La moyenne représente tous les nœuds du cluster. Cette métrique est également disponible pour les nœuds individuels.
`FreeStorageSpace`	Espace libre pour les nœuds de données du cluster. `Sum` indique l'espace libre total pour le cluster, mais vous devez laisser la période à une minute pour obtenir une valeur précise. `Minimum` et `Maximum` indiquent les nœuds avec le moins et le plus d'espace libre, respectivement. Cette métrique est également disponible pour les nœuds individuels. OpenSearch Le service lance un `ClusterBlockException` lorsque cette métrique atteint`0`. Pour récupérer, vous devez supprimer des index, ajouter des instances plus grandes ou ajouter du stockage EBS aux instances existantes. Pour en savoir plus, veuillez

Métrique	Description
	consulter la section the section called “Manque d'espace de stockage disponible”. La console OpenSearch de service affiche cette valeur in Gio. La CloudWatch console Amazon l'affiche en Mio.
`JVMMemory Pressure`	Le pourcentage maximum du tas Java utilisé pour tous les nœuds de données dans le cluster. OpenSearch Le service utilise la moitié de la mémoire RAM d'une instance pour le tas Java, jusqu'à une taille de tas de 32 GiB. Vous pouvez mettre à l'échelle des instances verticalement jusqu'à 64 Gio de RAM, après quoi vous pouvez effectuer une mise à l'échelle horizontale en ajoutant des instances. Consultez the section called “ CloudWatch Alarmes recommandées”.
`SysMemory Utilization`	Pourcentage de mémoire de l'instance actuellement utilisée. Les valeurs élevées de cette métrique sont normales et ne représentent généralem ent pas un problème lié à votre cluster. Pour obtenir un meilleur indicateu r des éventuels problèmes de performance et de stabilité, veuillez consulter la métrique `JVMMemoryPressure` .
`IndexingLatency`	Différence de temps total, en millisecondes, prise par toutes les opérations d'indexation dans un nœud entre la minute N et la minute (N-1).
`IndexingRate`	Nombre d'opérations d'indexation par minute.
`SearchLatency`	Différence de temps total, en millisecondes, prise par toutes les recherches dans un nœud entre la minute N et la minute (N-1).
`SearchRate`	Nombre total de demandes de recherche par minute pour toutes les partitions d'un nœud de données.
`Threadpoo lSearchQueue`	Nombre de tâches mises en file d'attente dans le groupe de threads de recherche. Si la taille de la file d'attente reste constamment élevée, envisagez de mettre votre cluster à l'échelle. La taille maximale de la file d'attente de recherche est de 1 000.

Métrique	Description
`ThreadpoolWriteQueue`	Nombre de tâches mises en file d'attente dans le groupe de threads d'écriture.
`ThreadpoolSearchRejected`	Nombre de tâches rejetées dans le groupe de threads de recherche. Si ce nombre augmente constamment, envisagez de mettre votre cluster à l'échelle.
`ThreadpoolWriteRejected`	Nombre de tâches rejetées dans le groupe de threads d'écriture.

Métriques de niveau cluster pour les clusters dans les zones de disponibilité actives

Métrique	Description
`DataNodes`	Nombre total de partitions actives et de partitions de secours.
`DataNodesShards.active`	Nombre total de partitions primaires et de partitions de réplica actives.
`DataNodesShards.unassigned`	Nombre de partitions non allouées aux nœuds du cluster.
`DataNodesShards.initializing`	Nombre de partitions en cours d'initialisation.
`DataNodesShards.relocating`	Nombre de partitions en cours de relocalisation.

Mesures de rotation des zones de disponibilité

Si c'est le `casActiveReads.`*`Availability-Zone`* `= 1`, la zone est active. Si c'est le `casActiveReads.`*`Availability-Zone`* `= 0`, la zone est en veille.

Métriques ponctuelles

Amazon OpenSearch Service fournit les statistiques suivantes pour les recherches ponctuelles (PIT).

Statistiques concernant le nœud de coordinateur

Métrique	Description
`CurrentPointInTime`	Nombre de contextes de recherche PIT actifs dans le nœud.
`TotalPointInTime`	Nombre de contextes de recherche PIT expirés depuis le démarrage du nœud.
`AvgPointInTimeAliveTime`	Maintien en vie des contextes de recherche PIT depuis le démarrage du nœud.
`HasActivePointInTime`	Une valeur de 1 indique qu'il existe des contextes PIT actifs sur les nœuds depuis leur disponibilité. Une valeur égale à 0 signifie qu'il n'y en a pas.
`HasUsedPointInTime`	Une valeur de 1 indique que des contextes PIT ont expiré sur les nœuds depuis leur disponibilité. Une valeur égale à 0 signifie qu'il n'y en a pas.

Métriques SQL

Amazon OpenSearch Service fournit les métriques suivantes pour la prise en charge de SQL.

Métrique	Description
`SQLFailedRequestCountByCusErr`	Nombre de demandes adressées à l'API `_sql` qui ont échoué en raison d'un problème client. Par exemple, une demande peut renvoyer le code d'état HTTP 400 en raison d'une exception `IndexNotFoundException` . Statistiques pertinentes : somme

Métrique	Description
`SQLFailedRequestCountBySysErr`	Nombre de demandes adressées à l'API `_sql` qui ont échoué en raison d'un problème de serveur ou d'une limitation de fonctionnalité. Par exemple, une demande peut renvoyer le code d'état HTTP 503 en raison d'une exception `VerificationException` . Statistiques pertinentes : somme
`SQLRequestCount`	Nombre de demandes adressées à l'API `_sql`. Statistiques pertinentes : somme
`SQLDefaultCursorRequestCount`	Similaire aux demandes de paginationSQLRequestCount , mais ne compte que les demandes de pagination. Statistiques pertinentes : somme
`SQLUnhealthy`	Une valeur de 1 indique que, en réponse à certaines demandes, le plug-in SQL renvoie 5xx codes de réponse ou passe une requête DSL non valide à OpenSearch. Les autres demandes devraient continuer à aboutir avec succès. La valeur 0 indique qu'il n'y a pas de défaillance récente. Si vous voyez une valeur soutenue de 1, résolvez les demandes adressées par vos clients au plugin. Statistiques pertinentes : Maximum

Métriques k-NN

Amazon OpenSearch Service inclut les métriques suivantes pour le plugin k-nearest neighbor (k-NN).

Métrique	Description
`KNNCacheCapacityReached`	Métrique par nœud permettant de déterminer si la capacité du cache a été atteinte. Cette métrique est uniquement pertinente dans le cadre d'une recherche k-NN approximative. Statistiques pertinentes : Maximum

Métrique	Description
KNNCircuitBreakerTriggered	Métrique par cluster permettant de déterminer si le disjoncteur de circuit est déclenché. Si des nœuds renvoient une valeur de 1 pour KNNCacheCapacityReached , cette valeur renvoie également 1. Cette métrique est uniquement pertinente dans le cadre d'une recherche k-NN approximative. Statistiques pertinentes : Maximum
KNNEvictionCount	Métrique par nœud du nombre de graphiques ayant été expulsés du cache en raison de contraintes de mémoire ou de temps d'inactivité. Les expulsions explicites se produisant en raison de la suppression d'index ne sont pas comptabilisées. Cette métrique est uniquement pertinente dans le cadre d'une recherche k-NN approximative. Statistiques pertinentes : somme
KNNGraphIndexErrors	Métrique par nœud du nombre de demandes d'ajout du champ knn_vector d'un document sur un graphe ayant généré une erreur. Statistiques pertinentes : somme
KNNGraphIndexRequests	Métrique par nœud du nombre de demandes d'ajout du champ knn_vector d'un document sur un graphe. Statistiques pertinentes : somme
KNNGraphMemoryUsage	Métrique par nœud de la taille actuelle du cache (taille totale de tous les graphes en mémoire) en kilo-octets. Cette métrique est uniquement pertinente dans le cadre d'une recherche k-NN approximative. Statistiques pertinentes : Moyenne

Métrique	Description
KNNGraphQueryErrors	Métrique par nœud du nombre de requêtes de graphe ayant généré une erreur. Statistiques pertinentes : somme
KNNGraphQueryRequests	Métrique par nœud du nombre de requêtes de graphe. Statistiques pertinentes : somme
KNNHitCount	Métrique par nœud du nombre d'accès au cache. Un accès au cache intervient lorsqu'un utilisateur interroge un graphe déjà chargé en mémoire. Cette métrique est uniquement pertinente dans le cadre d'une recherche k-NN approximative. Statistiques pertinentes : somme
KNNLoadExceptionCount	Métrique par nœud indiquant le nombre de fois où une exception s'est produite lors d'une tentative de chargement de graphe dans le cache. Cette métrique est uniquement pertinente dans le cadre d'une recherche k-NN approximative. Statistiques pertinentes : somme
KNNLoadSuccessCount	Métrique par nœud indiquant le nombre de fois où le plugin a chargé un graphe dans le cache. Cette métrique est uniquement pertinente dans le cadre d'une recherche k-NN approximative. Statistiques pertinentes : somme
KNNMissCount	Métrique par nœud du nombre d'échecs du cache. Un échec du cache intervient lorsqu'un utilisateur interroge un graphe pas encore chargé en mémoire. Cette métrique est uniquement pertinente dans le cadre d'une recherche k-NN approximative. Statistiques pertinentes : somme

Métrique	Description
KNNQueryRequests	Métrique par nœud du nombre de demandes de requête reçues par le plugin k-NN. Statistiques pertinentes : somme
KNNScriptCompilationErrors	Métrique par nœud du nombre d'erreurs lors d'une compilation de script. Cette statistique est uniquement pertinente pour la recherche de script de score k-NN. Statistiques pertinentes : somme
KNNScriptCompilations	Métrique par nœud indiquant le nombre de fois où le script k-NN a été compilé. Cette valeur doit généralement correspondre à 1 ou 0, mais si le cache contenant les scripts compilés est plein, le script k-NN peut être recompilé. Cette statistique est uniquement pertinente pour la recherche de script de score k-NN. Statistiques pertinentes : somme
KNNScriptQueryErrors	Métrique par nœud du nombre d'erreurs lors des requêtes de script. Cette statistique est uniquement pertinente pour la recherche de script de score k-NN. Statistiques pertinentes : somme
KNNScriptQueryRequests	Métrique par nœud du nombre total de requêtes de script. Cette statistique est uniquement pertinente pour la recherche de script de score k-NN. Statistiques pertinentes : somme
KNNTotalLoadTime	Délai, en nanosecondes, mis par k-NN pour charger les graphes dans le cache. Cette métrique est uniquement pertinente dans le cadre d'une recherche k-NN approximative. Statistiques pertinentes : somme

Métriques de recherche inter-clusters

Amazon OpenSearch Service fournit les métriques suivantes pour la recherche croisée.

Métriques de domaine source

Métrique	Dimension	Description
`CrossClusterOutboundConnections`	`ConnectionId`	Nombre de nœuds connectés. Si votre réponse inclut un ou plusieurs domaines ignorés, utilisez cette métrique pour suivre les connexions non saines. Si ce nombre chute jusqu'à 0, la connexion n'est pas saine.
`CrossClusterOutboundRequests`	`ConnectionId`	Nombre de demandes de recherche envoyées au domaine de destination. A utiliser pour vérifier si la charge de requêtes de recherche inter-clusters submerge votre domaine, corrélez n'importe quel pic de cette métrique avec n'importe quel pic JVM/CPU.

Métrique de domaine de destination

Métrique	Dimension	Description
`CrossClusterInboundRequests`	`ConnectionId`	Nombre de demandes de connexion entrantes reçues du domaine source.

Ajoutez une CloudWatch alarme à l'événement en cas de perte inattendue d'une connexion. Pour connaître les étapes de création d'une alarme, voir Création CloudWatch d'une alarme basée sur un seuil statique.

Métriques de réplication inter-clusters (CCR)

Amazon OpenSearch Service fournit les métriques suivantes pour la réplication croisée.

Métrique	Description
`ReplicationRate`	Le taux moyen d'opérations de réplication par seconde. Cette métrique est similaire à la métrique `IndexingRate` .
`LeaderCheckPoint`	Pour une connexion spécifique, la somme des valeurs des points de contrôle des principaux pour tous les index de réplication. Vous pouvez utiliser cette métrique pour mesurer la latence de réplication.
`FollowerCheckPoint`	Pour une connexion spécifique, la somme des valeurs des points de contrôle des suiveurs pour tous les index de réplication. Vous pouvez utiliser cette métrique pour mesurer la latence de réplication.
`ReplicationNumSyncingIndices`	Le nombre d'index qui ont un statut de réplication `SYNCING`.
`ReplicationNumBootstrappingIndices`	Le nombre d'index qui ont un statut de réplication `BOOTSTRAPPING` .
`ReplicationNumPausedIndices`	Le nombre d'index qui ont un statut de réplication `PAUSED`.
`ReplicationNumFailedIndices`	Le nombre d'index qui ont un statut de réplication `FAILED`.
`CrossClusterOutboundReplicationRequests`	Nombre de demandes de transport de réplication sur le domaine suiveur. Les demandes de transport sont internes et se produisent chaque fois qu'une opération d'API de réplication est appelée. Ils se produisent également lorsque le domaine suiveur interroge un changement par rapport au domaine leader.

Métrique	Description
`CrossClusterInboundReplicationRequests`	Nombre de demandes de transport de réplication sur le domaine principal. Les demandes de transport sont internes et se produisent chaque fois qu'une opération d'API de réplication est appelée.
`AutoFollowNumSuccessStartReplication`	Le nombre d'index suiveurs qui ont été créés avec succès par une règle de réplication pour une connexion spécifique.
`AutoFollowNumFailedStartReplication`	Le nombre d'index suiveurs qui n'ont pas pu être créés par une règle de réplication alors qu'il existait un modèle correspondant. Ce problème peut survenir en raison d'une avarie du réseau sur le cluster distant ou d'un problème de sécurité (c'est-à-dire que le rôle associé n'a pas l'autorisation de démarrer la réplication).
`AutoFollowLeaderCallFailure`	Indique si des requêtes ont échoué de l'index suiveur vers l'index principal pour extraire de nouvelles données. Une valeur de 1 signifie qu'il y a eu 1 ou plusieurs appels échoués au cours de la dernière minute.

Métriques Learning to Rank

Amazon OpenSearch Service fournit les métriques suivantes pour Learning to Rank.

Métrique	Description
`LTRRequestTotalCount`	Nombre total de demandes de classement.
`LTRRequestErrorCount`	Nombre total de demandes ayant échoué.
`LTRStatus.red`	Assure un suivi si l'un des index nécessaires à l'exécution du plugin est rouge.

Métrique	Description
`LTRMemoryUsage`	Mémoire totale utilisée par le plugin.
`LTRFeaturMemoryUsageInBytes`	Mémoire, en octets, utilisée par les champs des fonctions Learning to Rank.
`LTRFeaturesetMemoryUsageInBytes`	Mémoire, en octets, utilisée par tous les ensembles de fonctions Learning to Rank.
`LTRModelMemoryUsageInBytes`	Mémoire, en octets, utilisée par tous les modèles Learning to Rank.

Métriques du langage de traitement PPL (Piped Processing Language)

Amazon OpenSearch Service fournit les métriques suivantes pour le langage PPL.

Métrique	Description
`PPLFailedRequestCountByCusErr`	Nombre de demandes adressées à l'API `_ppl` qui ont échoué en raison d'un problème client. Par exemple, une demande peut renvoyer le code d'état HTTP 400 en raison d'une exception `IndexNotFoundException` .
`PPLFailedRequestCountBySysErr`	Nombre de demandes adressées à l'API `_ppl` qui ont échoué en raison d'un problème de serveur ou d'une limitation de fonctionnalité. Par exemple, une demande peut renvoyer le code d'état HTTP 503 en raison d'une exception `VerificationException` .
`PPLRequestCount`	Nombre de demandes adressées à l'API `_ppl`.

Surveillance des OpenSearch journaux avec Amazon CloudWatch Logs

Amazon OpenSearch Service expose les OpenSearch journaux suivants via Amazon CloudWatch Logs :

- Journaux des erreurs
- Journaux lents des demandes de recherche
- Partagez les journaux lents
- Journaux d'audit

Les journaux de lenteur des partitions de recherche, les journaux lents des partitions d'indexation et les journaux d'erreurs sont utiles pour résoudre les problèmes de performance et de stabilité. Les journaux d'audit permettent de suivre l'activité des utilisateurs à des fins de conformité. Par défaut, tous les journaux sont désactivés. Si cette option est activée, la CloudWatch tarification standard s'applique.

> **Note**
>
> Les journaux d'erreurs ne sont disponibles que pour les versions 5.1 OpenSearch et ultérieures d'Elasticsearch. Les journaux lents sont disponibles pour toutes les versions OpenSearch et pour Elasticsearch.

Pour ses journaux, OpenSearch utilise Apache Log4j 2 et ses niveaux de journalisation intégrés (du plus faible au plus sévère) de`TRACE`,`DEBUG`,, `INFO` `WARNERROR`, et. `FATAL`

Si vous activez les journaux d'erreurs, le OpenSearch Service publie des lignes de journal de `WARNERROR`, et `FATAL` vers CloudWatch. OpenSearch Le service publie également plusieurs exceptions par rapport au `DEBUG` niveau, notamment les suivantes :

- `org.opensearch.index.mapper.MapperParsingException`
- `org.opensearch.index.query.QueryShardException`
- `org.opensearch.action.search.SearchPhaseExecutionException`
- `org.opensearch.common.util.concurrent.OpenSearchRejectedExecutionException`
- `java.lang.IllegalArgumentException`

Les journaux d'erreurs favorisent la résolution des problèmes dans de nombreuses situations, y compris les suivantes :

- Problèmes de compilation de scripts Painless
- Requêtes non valides
- Indexation des problèmes
- Échecs d'instantané
- Échecs de migration d'Index State Management

> **Note**
>
> Toutes les erreurs ne sont pas signalées dans les journaux d'erreurs.

> **Note**
>
> OpenSearch Le service n'enregistre pas toutes les erreurs qui se produisent.

Rubriques

- Activation de la publication des journaux (console)
- Activation de la publication des journaux (AWS CLI)
- Activation de la publication des journaux (AWS SDKs)
- Activation de la publication des journaux (CloudFormation)
- Définition des seuils de lenteur de journalisation des demandes de recherche
- Définition des seuils de lenteur de journalisation des partitions
- Tester les journaux lents
- Affichage des journaux

Activation de la publication des journaux (console)

La console OpenSearch de service est le moyen le plus simple d'activer la publication de journaux sur CloudWatch.

Pour activer la publication du journal sur CloudWatch (console)

1. Accédez à aws.amazon.com, puis choisissez Se connecter et entrez vos informations d'identification.
2. Sous Analytics, sélectionnez Amazon OpenSearch Service.
3. Sélectionnez le domaine que vous souhaitez mettre à jour.
4. Dans l'onglet Logs (Journaux), sélectionnez un type de journal et choisissez Enable (Activer).
5. Créez un nouveau groupe de CloudWatch journaux ou choisissez-en un existant.

> **Note**
>
> Si vous prévoyez d'activer plusieurs journaux, il est recommandé de publier chacun d'eux dans son propre groupe de journaux. Cette séparation rend plus facile l'analyse des journaux.

6. Choisissez une stratégie d'accès qui contient les autorisations appropriées ou créez une stratégie à l'aide du code JSON que la console fournit :

   JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "es.amazonaws.com"
      },
      "Action": [
        "logs:PutLogEvents",
        "logs:CreateLogStream"
      ],
      "Resource": "cw_log_group_arn:*"
    }
  ]
}
```

Nous vous recommandons d'ajouter les clés de condition `aws:SourceAccount` et `aws:SourceArn` à la stratégie pour vous protéger contre le problème du député confus. Le compte source est le propriétaire du domaine et l'ARN source est l'ARN du domaine. Votre domaine doit être sur le logiciel de service R20211203 ou plus récent afin d'ajouter ces clés de condition.

Par exemple, vous pouvez ajouter la clé de condition suivante à la stratégie :

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account-id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:aws:es:region:account-id:domain/domain-name"
    }
}
```

> **Important**
>
> CloudWatch Logs prend en charge 10 politiques de ressources par région. Si vous envisagez d'activer les journaux pour plusieurs domaines de OpenSearch service, vous devez créer et réutiliser une politique plus large incluant plusieurs groupes de journaux afin d'éviter d'atteindre cette limite. Pour savoir comment mettre à jour votre politique , consultez the section called “Activation de la publication des journaux (AWS CLI)”.

7. Sélectionnez Activer.

   L'état de votre domaine passe de Actif à En cours de traitement. L'état doit revenir à Actif avant que la publication du journal ne soit activée. Cette modification prend généralement 30 minutes, mais peut prendre plus de temps en fonction de la configuration de votre domaine.

Si vous avez activé l'un des journaux lents des partitions, consultezthe section called “Définition des seuils de lenteur de journalisation des partitions”. Si vous avez activé les journaux d'audit, consultez the section called “Étape 2 : activer les journaux d'audit dans les OpenSearch tableaux de bord”. Si vous avez activé uniquement des journaux d'erreurs, vous n'avez pas besoin d'effectuer d'étapes de configuration supplémentaires.

Activation de la publication des journaux (AWS CLI)

Avant de pouvoir activer la publication des journaux, vous avez besoin d'un groupe de CloudWatch journaux. Si vous n'en possédez pas déjà un, vous pouvez en créer un à l'aide de la commande suivante :

```
aws logs create-log-group --log-group-name my-log-group
```

Entrez la commande suivante pour trouver l'ARN du groupe de journaux, puis notez-le :

```
aws logs describe-log-groups --log-group-name my-log-group
```

Vous pouvez désormais autoriser le OpenSearch service à écrire dans le groupe de journaux. Vous devez fournir l'ARN du groupe de journaux à proximité de la fin de la commande :

```
aws logs put-resource-policy \
  --policy-name my-policy \
  --policy-document '{ "Version": "2012-10-17", "Statement": [{ "Sid": "",
 "Effect": "Allow", "Principal": { "Service": "es.amazonaws.com"}, "Action":
[ "logs:PutLogEvents","logs:CreateLogStream"],"Resource": "cw_log_group_arn:*"}]}'
```

> **Important**
>
> CloudWatch Logs prend en charge 10 politiques de ressources par région. Si vous envisagez d'activer les journaux lents partiels pour plusieurs domaines de OpenSearch service, vous devez créer et réutiliser une politique plus large incluant plusieurs groupes de journaux afin d'éviter d'atteindre cette limite.

Si vous devez revoir cette politique ultérieurement, utilisez la commande `aws logs describe-resource-policies`. Pour mettre à jour la politique, exécutez la même commande `aws logs put-resource-policy` avec un nouveau document de politique.

Enfin, vous pouvez utiliser l'option `--log-publishing-options` pour activer la publication. La syntaxe pour l'option est la même pour les deux commandes `create-domain` et `update-domain-config`.

Paramètre	Valeurs valides
`--log-publishing-options`	`SEARCH_SLOW_LOGS={CloudWatchLogsLogGroupArn= cw_log_group_arn ,Enabled=true|false}`
	`INDEX_SLOW_LOGS={CloudWatchLogsLogGroupArn= cw_log_group_arn ,Enabled=true|false}`
	`ES_APPLICATION_LOGS={CloudWatchLogsLogGroupArn= cw_log_group_arn ,Enabled=true|false}`
	`AUDIT_LOGS={CloudWatchLogsLogGroupArn= cw_log_group_arn ,Enabled=true|false}`

> **Note**
>
> Si vous prévoyez d'activer plusieurs journaux, il est recommandé de publier chacun d'eux dans son propre groupe de journaux. Cette séparation rend plus facile l'analyse des journaux.

Exemple

L'exemple suivant permet de publier des journaux de ralentissement des partitions de recherche et d'indexation pour le domaine spécifié :

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --log-publishing-options
 "SEARCH_SLOW_LOGS={CloudWatchLogsLogGroupArn=arn:aws:logs:us-east-1:123456789012:log-group:my-log-
group,Enabled=true},INDEX_SLOW_LOGS={CloudWatchLogsLogGroupArn=arn:aws:logs:us-east-1:123456789012:log-group:my-other-log-group,Enabled=true}"
```

Pour désactiver la publication sur CloudWatch, exécutez la même commande avec`Enabled=false`.

Si vous avez activé l'un des journaux lents des partitions, consultezthe section called “Définition des seuils de lenteur de journalisation des partitions”. Si vous avez activé les journaux d'audit, consultez the section called “Étape 2 : activer les journaux d'audit dans les OpenSearch tableaux de bord”. Si

vous avez activé uniquement des journaux d'erreurs, vous n'avez pas besoin d'effectuer d'étapes de configuration supplémentaires.

Activation de la publication des journaux (AWS SDKs)

Avant de pouvoir activer la publication de journaux, vous devez d'abord créer un groupe de CloudWatch journaux, obtenir son ARN et autoriser le OpenSearch service à y écrire. Les opérations pertinentes sont documentées dans le manuel Amazon CloudWatch Logs API Reference :

- `CreateLogGroup`
- `DescribeLogGroup`
- `PutResourcePolicy`

Vous pouvez accéder à ces opérations à l'aide du AWS SDKs.

AWS SDKs (sauf Android et iOS SDKs) prennent en charge toutes les opérations définies dans le Amazon OpenSearch Service API Reference, y compris l'`--log-publishing-options`option pour `CreateDomain` et`UpdateDomainConfig`.

Si vous avez activé l'un des journaux lents des partitions, consultezthe section called “Définition des seuils de lenteur de journalisation des partitions”. Si vous avez activé uniquement des journaux d'erreurs, vous n'avez pas besoin d'effectuer d'étapes de configuration supplémentaires.

Activation de la publication des journaux (CloudFormation)

Dans cet exemple, nous avons l'habitude de CloudFormation créer un groupe de journaux appelé`opensearch-logs`, d'attribuer les autorisations appropriées, puis de créer un domaine dans lequel la publication des journaux est activée pour les journaux des applications, les journaux lents de recherche sur les partitions de recherche et l'indexation des journaux lents.

Avant de pouvoir activer la publication des journaux, vous devez créer un groupe de CloudWatch journaux :

```
Resources:
  OpenSearchLogGroup:
    Type: AWS::Logs::LogGroup
    Properties:
      LogGroupName: opensearch-logs
Outputs:
```

```
  Arn:
    Value:
      'Fn::GetAtt':
        - OpenSearchLogGroup
        - Arn
```

Le modèle génère l'ARN du groupe de journaux. Dans ce cas, l'ARN est `arn:aws:logs:us-east-1:123456789012:log-group:opensearch-logs`.

À l'aide de l'ARN, créez une politique de ressources qui autorise le OpenSearch service à écrire dans le groupe de journaux :

```
Resources:
 OpenSearchLogPolicy:
   Type: AWS::Logs::ResourcePolicy
   Properties:
     PolicyName: my-policy
     PolicyDocument: "{ \"Version\": \"2012-10-17\", \"Statement\": [{ \"Sid\": \"\",
 \"Effect\": \"Allow\", \"Principal\": { \"Service\": \"es.amazonaws.com\"}, \"Action
\":[ \"logs:PutLogEvents\",\"logs:CreateLogStream\"],\"Resource\": \"arn:aws:logs:us-
east-1:123456789012:log-group:opensearch-logs:*\"}]}"
```

Enfin, créez la CloudFormation pile suivante, qui génère un domaine OpenSearch de service avec publication de journaux. La politique d'accès permet Compte AWS à l'utilisateur d'envoyer toutes les requêtes HTTP au domaine.

```
Resources:
  OpenSearchServiceDomain:
    Type: "AWS::OpenSearchService::Domain"
    Properties:
      DomainName: my-domain
      EngineVersion: "OpenSearch_1.0"
      ClusterConfig:
        InstanceCount: 2
        InstanceType: "r6g.xlarge.search"
        DedicatedMasterEnabled: true
        DedicatedMasterCount: 3
        DedicatedMasterType: "r6g.xlarge.search"
      EBSOptions:
        EBSEnabled: true
        VolumeSize: 10
        VolumeType: "gp2"
```

```
      AccessPolicies:
        Version: "2012-10-17"
        Statement:
            Effect: "Allow"
            Principal:
                AWS: "arn:aws:iam::123456789012:user/es-user"
            Action: "es:*"
            Resource: "arn:aws:es:us-east-1:123456789012:domain/my-domain/*"
      LogPublishingOptions:
        ES_APPLICATION_LOGS:
          CloudWatchLogsLogGroupArn: "arn:aws:logs:us-east-1:123456789012:log-
group:opensearch-logs"
          Enabled: true
        SEARCH_SLOW_LOGS:
          CloudWatchLogsLogGroupArn: "arn:aws:logs:us-east-1:123456789012:log-
group:opensearch-logs"
          Enabled: true
        INDEX_SLOW_LOGS:
          CloudWatchLogsLogGroupArn: "arn:aws:logs:us-east-1:123456789012:log-
group:opensearch-logs"
          Enabled: true
```

Pour obtenir des informations détaillées sur la syntaxe, consultez les options de publication de journaux dans le Guide de l'utilisateur AWS CloudFormation .

Définition des seuils de lenteur de journalisation des demandes de recherche

Les journaux de lenteur des demandes de recherche sont disponibles pour les recherches sur les domaines de OpenSearch service exécutés sur les versions 2.13 et ultérieures. Les seuils de lenteur des demandes de recherche sont configurés pour le temps total pris par les demandes. Cela est différent des journaux de lenteur des demandes de partition, qui sont configurés en fonction du temps nécessaire à chaque partition.

Vous pouvez définir les journaux lents des demandes de recherche à l'aide des paramètres du cluster. Cela diffère des journaux partiels lents, que vous activez à l'aide des paramètres d'index. Par exemple, vous pouvez définir les paramètres suivants via l' OpenSearch API REST :

```
PUT domain-endpoint/_cluster/settings
{
  "transient": {
```

```
    "cluster.search.request.slowlog.threshold.warn": "5s",
    "cluster.search.request.slowlog.threshold.info": "2s"
  }
}
```

Définition des seuils de lenteur de journalisation des partitions

OpenSearch désactive les journaux lents des partitions par défaut. Une fois que vous avez activé la publication des journaux lents des partitions sur CloudWatch, vous devez toujours spécifier des seuils de journalisation pour chaque OpenSearch index. Ces seuils définissent précisément ce qui doit être journalisé et à quel niveau de journal.

Par exemple, vous pouvez définir les paramètres suivants via l' OpenSearch API REST :

```
PUT domain-endpoint/index/_settings
{
  "index.search.slowlog.threshold.query.warn": "5s",
  "index.search.slowlog.threshold.query.info": "2s"
}
```

Tester les journaux lents

Pour vérifier que les journaux lents des requêtes de recherche et des partitions sont publiés correctement, envisagez de commencer par des valeurs très faibles pour vérifier que les journaux apparaissent CloudWatch, puis d'augmenter les seuils à des niveaux plus utiles.

Si les journaux ne s'affichent pas, vérifiez les éléments suivants :

- Le groupe de CloudWatch logs existe-t-il ? Vérifiez la CloudWatch console.
- Le OpenSearch service est-il autorisé à écrire dans le groupe de journaux ? Vérifiez la console OpenSearch de service.
- Le domaine OpenSearch de service est-il configuré pour publier dans le groupe de journaux ? Vérifiez la console OpenSearch de service, utilisez l' AWS CLI `describe-domain-config`option ou appelez à `DescribeDomainConfig` l'aide de l'un des SDKs.
- Les seuils de OpenSearch journalisation sont-ils suffisamment bas pour que vos demandes les dépassent ?

  Pour revoir les seuils de lenteur de votre demande de recherche pour un domaine, utilisez la commande suivante :

```
GET domain-endpoint/_cluster/settings?flat_settings
```

Pour revoir les seuils de lenteur de journalisation de votre partition pour un index, utilisez la commande suivante :

```
GET domain-endpoint/index/_settings?pretty
```

Si vous souhaitez désactiver les journaux lents pour un index, rétablissez les seuils que vous avez modifiés aux valeurs par défaut de `-1`.

La désactivation de la publication pour CloudWatch utiliser la console de OpenSearch service AWS CLI n'arrête pas la génération OpenSearch de journaux ; elle arrête uniquement la publication de ces journaux. Assurez-vous de vérifier les paramètres de votre index si vous n'avez plus besoin des journaux lents des partitions, et les paramètres de votre domaine si vous n'avez plus besoin des journaux lents des demandes de recherche.

Affichage des journaux

L'affichage de l'application et la lenteur des connexions CloudWatch sont identiques à l'affichage de n'importe quel autre CloudWatch journal. Pour plus d'informations, consultez la section Afficher les données des CloudWatch journaux dans le guide de l'utilisateur Amazon Logs.

Voici quelques éléments à prendre en compte pour visualiser les journaux :

- OpenSearch Le service ne publie que les 255 000 premiers caractères de chaque ligne sur CloudWatch. Tout le contenu restant est tronqué. Pour les journaux d'audit, il s'agit de 10 000 caractères par message.
- Dans CloudWatch, les noms des flux de journaux ont les suffixes`-index-slow-logs`, `-search-slow-logs-application-logs`, et `-audit-logs` pour aider à identifier leur contenu.

Surveillance des journaux d'audit dans Amazon OpenSearch Service

Si votre domaine Amazon OpenSearch Service utilise un contrôle d'accès précis, vous pouvez activer les journaux d'audit pour vos données. Les journaux d'audit sont hautement personnalisables et

vous permettent de suivre l'activité des utilisateurs sur vos OpenSearch clusters, notamment les réussites et les échecs d'authentification, les demandes OpenSearch, les modifications d'index et les requêtes de recherche entrantes. La configuration par défaut permet de suivre un ensemble d'actions utilisateur courantes. Nous vous recommandons toutefois d'adapter les paramètres à vos besoins précis.

Tout comme les journaux OpenSearch d'application et les journaux lents, le OpenSearch service publie les journaux d'audit dans CloudWatch Logs. Si cette option est activée, la CloudWatch tarification standard s'applique.

> **Note**
>
> Pour activer les journaux d'audit, votre rôle d'utilisateur doit être associé au rôle `security_manager` qui vous donne accès à l'API REST OpenSearch `plugins/_security`. Pour en savoir plus, veuillez consulter la section the section called “Modification de l'utilisateur maître”.

Rubriques

- Limites
- Activation des journaux d'audit
- Activez la journalisation des audits à l'aide du AWS CLI
- Activation du journal d'audit à l'aide de l'API de configuration
- Couches et catégories de journaux d'audit
- Paramètres des journaux d'audit
- Exemples de journaux d'audit
- Configuration des journaux d'audit à l'aide de l'API REST

Limites

Les journaux d'audit présentent les limitations suivantes :

- Les journaux d'audit n'incluent pas les requêtes de recherche inter-clusters qui ont été rejetées par la politique d'accès au domaine de la destination.
- La longueur maximale de chaque message de journal d'audit est de 10 000 caractères. Si un message du journal d'audit dépasse cette limite, il est tronqué.

Activation des journaux d'audit

L'activation des journaux d'audit est un processus en deux étapes. Vous devez d'abord configurer votre domaine pour publier les journaux d'audit dans CloudWatch Logs. Vous activez ensuite les journaux d'audit dans les OpenSearch tableaux de bord et vous les configurez pour répondre à vos besoins.

> **Important**
>
> En cas d'erreur au cours de cette procédure, consultez the section called “Impossible d'activer les journaux d'audit” pour obtenir des informations de dépannage.

Étape 1 : activer les journaux d'audit et configurer une stratégie d'accès

Ces étapes décrivent comment activer les journaux d'audit à l'aide de la console. Vous pouvez également les activer à l'aide AWS CLI de l'API ou du OpenSearch service.

Pour activer les journaux d'audit pour un domaine de OpenSearch service (console)

1. Choisissez le domaine pour ouvrir sa configuration, puis allez à l'onglet Logs (Journaux).
2. Sélectionnez Audit logs (Journaux d'audit) et ensuite Enable (Activer).
3. Créez un groupe de CloudWatch journaux ou choisissez-en un existant.
4. Choisissez une stratégie d'accès qui contient les autorisations appropriées ou créez une stratégie à l'aide du code JSON que la console fournit :

   JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "es.amazonaws.com"
      },
      "Action": [
        "logs:PutLogEvents",
        "logs:CreateLogStream"
```

```
      ],
      "Resource": "cw_log_group_arn"
    }
  ]
}
```

Nous vous recommandons d'ajouter les clés de condition `aws:SourceAccount` et `aws:SourceArn` à la stratégie pour vous protéger contre le problème du député confus. Le compte source est le propriétaire du domaine et l'ARN source est l'ARN du domaine. Votre domaine doit être sur le logiciel de service R20211203 ou plus récent afin d'ajouter ces clés de condition.

Par exemple, vous pouvez ajouter la clé de condition suivante à la stratégie :

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account-id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:aws:es:region:account-id:domain/domain-name"
    }
}
```

5. Sélectionnez Activer.

Étape 2 : activer les journaux d'audit dans les OpenSearch tableaux de bord

Après avoir activé les journaux d'audit dans la console de OpenSearch service, vous devez également les activer dans les OpenSearch tableaux de bord et les configurer en fonction de vos besoins.

1. Ouvrez OpenSearch les tableaux de bord et choisissez Sécurité dans le menu de gauche.
2. Choisissez Journaux d'audit.
3. Choisissez Activer la journalisation d'audit.

L'interface utilisateur de Dashboards offre un contrôle total sur les paramètres des journaux d'audit sous General settings (Paramètres généraux) et Compliance settings (Paramètres de conformité).

Pour obtenir une description de toutes les options de configuration, consultez Paramètres des journaux d'audit.

Activez la journalisation des audits à l'aide du AWS CLI

La AWS CLI commande suivante active les journaux d'audit sur un domaine existant :

```
aws opensearch update-domain-config --domain-name my-domain --log-publishing-options
 "AUDIT_LOGS={CloudWatchLogsLogGroupArn=arn:aws:logs:us-east-1:123456789012:log-
group:my-log-group,Enabled=true}"
```

Vous pouvez également activer les journaux d'audit lorsque vous créez un domaine. Pour plus d'informations, consultez le Guide de référence des commandes AWS CLI.

Activation du journal d'audit à l'aide de l'API de configuration

Cette demande adressée à l'API de configuration active les journaux d'audit sur un domaine existant :

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/config
{
  "LogPublishingOptions": {
    "AUDIT_LOGS": {
      "CloudWatchLogsLogGroupArn":"arn:aws:logs:us-east-1:123456789012:log-
group1:sample-domain",
      "Enabled":true
    }
  }
}
```

Pour plus d'informations, consultez la référence de l'API Amazon OpenSearch Service.

Couches et catégories de journaux d'audit

La communication dans les clusters s'effectue sur deux couches distinctes : la couche REST et la couche de transport.

- La couche REST couvre la communication avec les clients HTTP tels que curl, Logstash, OpenSearch Dashboards, le client REST de haut niveau Java, la bibliothèque de requêtes Python, c'est-à-dire toutes les requêtes HTTP qui arrivent au cluster.
- La couche de transport couvre la communication entre les nœuds. Par exemple, une fois qu'une requête de recherche arrive au cluster (sur la couche REST), le nœud de coordination qui répond

à la requête envoie la requête aux autres nœuds, reçoit leurs réponses, réunit les documents nécessaires et les rassemble dans la réponse finale. Les opérations telles que l'allocation de partitions et le rééquilibrage se font également sur la couche de transport.

Vous pouvez activer ou désactiver les journaux d'audit pour des couches entières, ainsi que pour des catégories d'audit individuelles relatives à une couche. Le tableau suivant contient un récapitulatif des catégories d'audit et des couches pour lesquelles elles sont disponibles.

Catégorie	Description	Disponible pour la couche REST	Disponible pour la couche de transport
FAILED_LOGIN	Une demande contenait des informations d'identification valides et l'authentification a abouti.	Oui	Oui
MISSING_PRIVILEGES	Un utilisateur ne disposait pas des privilèges nécessaires pour effectuer la demande.	Oui	Oui
GRANTED_PRIVILEGES	Un utilisateur disposait des privilèges pour effectuer la demande.	Oui	Oui
OPENSEARCH_SECURITY_INDEX_ATTEMPT	Une demande a tenté de modifier l'index `.opendistro_security` .	Non	Oui
AUTHENTICATED	Une demande contenait des informations d'identification valides et l'authentification a abouti.	Oui	Oui

Catégorie	Description	Disponible pour la couche REST	Disponible pour la couche de transport
INDEX_EVENT	Une demande a effectué une opération administrative sur un index, comme la création d'un index, la définition d'un alias ou l'exécution d'une fusion forcée. La liste complète des `indices:admin/` actions incluses dans cette catégorie est disponible dans la OpenSearch documentation.	Non	Oui

Outre ces catégories standard, le contrôle précis des accès offre plusieurs catégories supplémentaires conçues pour répondre aux exigences de conformité des données.

Catégorie	Description
COMPLIANCE_DOC_READ	Une demande a exécuté un événement de lecture sur un document dans un index.
COMPLIANCE_DOC_WRITE	Une demande a exécuté un événement d'écriture sur un document dans un index.
COMPLIANCE_INTERNAL_CONFIG_READ	Une demande a exécuté un événement de lecture sur l'index `.opendistro_security` .

Catégorie	Description
COMPLIANCE_INTERNAL_CONFIG_WRITE	Une demande a exécuté un événement d'écriture sur l'index `.opendistro_security` .

Vous pouvez avoir n'importe quelle combinaison de catégories et d'attributs de message. Par exemple, si vous envoyez une demande REST pour indexer un document, les lignes suivantes peuvent apparaître dans les journaux d'audit :

- AUTHENTICATED sur la couche REST (authentification)
- GRANDED_PRIVILEGE sur la couche de transport (autorisation)
- COMPLIANCE_DOC_WRITE (document écrit dans un index)

Paramètres des journaux d'audit

Les journaux d'audit disposent de nombreuses options de configuration.

Paramètres généraux

Les paramètres généraux vous permettent d'activer ou de désactiver des catégories individuelles ou des couches entières. Nous vous recommandons vivement de maintenir GRANTED_PRIVILEGES et AUTHENTICATED comme catégories exclues. À défaut, ces catégories sont journalisées pour chaque demande valide adressée au cluster.

Nom	Paramètres backend	Description
Couche REST	enable_rest	Permet d'activer ou de désactiver les événements qui se produisent sur la couche REST.
Catégories désactivées sur la couche REST	disabled_rest_categories	Permet de spécifier les catégories d'audit à ignorer sur la couche REST. La modification de ces catégories peut considérablement augmenter la taille des journaux d'audit.
Couche de transport	enable_transport	Permet d'activer ou de désactiver les événements qui se produisent sur la couche de transport.

Nom	Paramètres backend	Description
Catégories désactivées sur la couche de transport	disabled_transport_categories	Permet de spécifier les catégories d'audit à ignorer sur la couche de transport. La modification de ces catégories peut considérablement augmenter la taille des journaux d'audit.

Les paramètres d'attribut vous permettent de personnaliser le niveau de détail de chaque ligne de journal.

Nom	Paramètres backend	Description
Demandes groupées	resolve_bulk_requests	L'activation de ce paramètre génère un journal pour chacun des documents associés à une demande groupée, ce qui peut considérablement augmenter la taille des journaux d'audit.
Corps de la demande	log_request_body	Permet d'inclure le corps des requêtes.
Résolution des index	resolve_indices	Permet de résoudre les alias en index.

Utilisez les paramètres d'exclusion pour exclure un ensemble d'utilisateurs ou de chemins d'API :

Nom	Paramètres backend	Description
Utilisateurs ignorés	ignore_users	Permet de spécifier les utilisateurs à exclure.
Demandes ignorées	ignore_requests	Permet de spécifier les modèles de demande à exclure.

Paramètres de conformité

Les paramètres de conformité vous permettent de définir l'accès au niveau de l'index, du document ou du champ.

Nom	Paramètres backend	Description
Journalisation de la conformité	enable_compliance	Permet d'activer ou de désactiver la journalisation de la conformité.

Vous pouvez spécifier les paramètres suivants pour la journalisation des événements de lecture et d'écriture.

Nom	Paramètres backend	Description
Journalisation de la configuration interne	internal_config	Activer ou désactiver la journalisation des événements sur l'index `.opendistro_security` .

Vous pouvez spécifier les paramètres suivants pour les événements de lecture.

Nom	Paramètres backend	Description
Métadonnées de lecture	read_metadata_only	Permet de n'inclure que les métadonnées des événements de lecture. Aucun champ de document n'est inclus.
Utilisateurs ignorés	read_ignore_users	Permet de ne pas inclure certains utilisateurs pour les événements de lecture.
Champs surveillés	read_watched_fields	Permet de spécifier les index et les champs à surveiller pour les événements de lecture. L'ajout de champs surveillés génère un journal par accès au document, ce qui peut considérablement augmenter la taille des journaux d'audit.

Nom	Paramètres backend	Description
		Les champs surveillés prennent en charge les modèles d'index et les modèles de champ : `{ "index-name-pattern": [ "field-name-pattern" ], "logs*": [ "message" ], "twitter": [ "id", "user*" ] }`

Vous pouvez spécifier les paramètres suivants pour les événements d'écriture.

Nom	Paramètres backend	Description
Métadonnées d'écriture	write_metadata_only	Permet de n'inclure que les métadonnées des événements d'écriture. Aucun champ de document n'est inclus.
Journalisation des différences	write_log_diffs	Si write_metadata_only est défini sur la valeur false, cela permet de n'inclure que les différences entre les événements d'écriture.
Utilisateurs ignorés	write_ignore_users	Permet de ne pas inclure certains utilisateurs pour les événements d'écriture.
Index surveillés	write_watched_indices	Permet de spécifier les index ou les modèles d'index à surveiller pour les événements d'écriture. L'ajout de champs surveillés génère un journal par accès au document, ce qui

Nom	Paramètres backend	Description
		peut considérablement augmenter la taille des journaux d'audit.

Exemples de journaux d'audit

Cette section comprend un exemple de configuration, de requête de recherche et le journal d'audit correspondant pour tous les événements de lecture et d'écriture d'un index.

Étape 1 : Configurer des journaux d'audit

Après avoir activé la publication des journaux d'audit dans un groupe de CloudWatch journaux, accédez à la page de journalisation OpenSearch des audits des tableaux de bord et choisissez Activer la journalisation des audits.

1. Dans Paramètres généraux, choisissez Configurer et vérifiez que la couche REST est activée.
2. Dans Paramètres de conformité, choisissez Configurer.
3. Sous Écriture, accédez à Champs surveillés et ajoutez à cet index le champ `accounts` de tous les événements d'écriture.
4. Sous Read (Lecture), accédez à Watched Fields (Champs surveillés) et ajoutez les champs `ssn` et `id-` à l'index `accounts` :

```
{
  "accounts-": [
    "ssn",
    "id-"
  ]
}
```

Étape 2 : Exécuter des événements de lecture et d'écriture

1. Accédez à OpenSearch Tableaux de bord, choisissez Outils de développement et indexez un exemple de document :

```
PUT accounts/_doc/0
```

```
{
  "ssn": "123",
  "id-": "456"
}
```

2. Pour tester un événement de lecture, envoyez la demande suivante :

```
GET accounts/_search
{
  "query": {
    "match_all": {}
  }
}
```

Étape 3 : Observer les journaux

1. Ouvrez la CloudWatch console à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Dans le panneau de navigation, choisissez Groupes de journaux.
3. Choisissez le groupe de journaux que vous avez spécifié lors de l'activation des journaux d'audit. Au sein du groupe de journaux, OpenSearch Service crée un flux de journal pour chaque nœud de votre domaine.
4. Dans Flux de journaux, choisissez Rechercher partout.
5. Pour les événements de lecture et d'écriture, consultez les journaux correspondants. Cinq secondes peuvent s'écouler avant l'apparition du journal.

   Exemple de journal d'audit d'écriture

```
{
  "audit_compliance_operation": "CREATE",
  "audit_cluster_name": "824471164578:audit-test",
  "audit_node_name": "be217225a0b77c2bd76147d3ed3ff83c",
  "audit_category": "COMPLIANCE_DOC_WRITE",
  "audit_request_origin": "REST",
  "audit_compliance_doc_version": 1,
  "audit_node_id": "3xNJhm4XS_yTzEgDWcGRjA",
  "@timestamp": "2020-08-23T05:28:02.285+00:00",
  "audit_format_version": 4,
  "audit_request_remote_address": "3.236.145.227",
  "audit_trace_doc_id": "lxnJGXQBqZSlDB91r_uZ",
  "audit_request_effective_user": "admin",
```

```
  "audit_trace_shard_id": 8,
  "audit_trace_indices": [
    "accounts"
  ],
  "audit_trace_resolved_indices": [
    "accounts"
  ]
}
```

Exemple de journal d'audit de lecture

```
{
  "audit_cluster_name": "824471164578:audit-docs",
  "audit_node_name": "806f6050cb45437e2401b07534a1452f",
  "audit_category": "COMPLIANCE_DOC_READ",
  "audit_request_origin": "REST",
  "audit_node_id": "saSevm9ASte0-pjAtYi2UA",
  "@timestamp": "2020-08-31T17:57:05.015+00:00",
  "audit_format_version": 4,
  "audit_request_remote_address": "54.240.197.228",
  "audit_trace_doc_id": "config:7.7.0",
  "audit_request_effective_user": "admin",
  "audit_trace_shard_id": 0,
  "audit_trace_indices": [
    "accounts"
  ],
  "audit_trace_resolved_indices": [
    "accounts"
  ]
}
```

Pour inclure le corps de la demande, revenez aux paramètres de conformité dans les OpenSearch tableaux de bord et désactivez l'option Écrire les métadonnées. Pour exclure les événements d'un utilisateur spécifique, ajoutez-le à Utilisateurs ignorés.

Pour obtenir une description de chacun des champs de journal d'audit, consultez Guide de référence des champs de journal d'audit. Pour plus d'informations sur la recherche et l'analyse des données de vos journaux d'audit, consultez Analyzing Log Data with CloudWatch Logs Insights dans le guide de l'utilisateur Amazon CloudWatch Logs.

Configuration des journaux d'audit à l'aide de l'API REST

Nous vous recommandons d'utiliser OpenSearch des tableaux de bord pour configurer les journaux d'audit, mais vous pouvez également utiliser l'API REST de contrôle d'accès précise. Cette section contient un exemple de requête. La documentation complète sur l'API REST est disponible dans la OpenSearch documentation.

```
PUT _opendistro/_security/api/audit/config
{
  "enabled": true,
  "audit": {
    "enable_rest": true,
    "disabled_rest_categories": [
      "GRANTED_PRIVILEGES",
      "AUTHENTICATED"
    ],
    "enable_transport": true,
    "disabled_transport_categories": [
      "GRANTED_PRIVILEGES",
      "AUTHENTICATED"
    ],
    "resolve_bulk_requests": true,
    "log_request_body": true,
    "resolve_indices": true,
    "exclude_sensitive_headers": true,
    "ignore_users": [
      "kibanaserver"
    ],
    "ignore_requests": [
      "SearchRequest",
      "indices:data/read/*",
      "/_cluster/health"
    ]
  },
  "compliance": {
    "enabled": true,
    "internal_config": true,
    "external_config": false,
    "read_metadata_only": true,
    "read_watched_fields": {
      "read-index-1": [
        "field-1",
        "field-2"
```

```
      ],
      "read-index-2": [
        "field-3"
      ]
    },
    "read_ignore_users": [
      "read-ignore-1"
    ],
    "write_metadata_only": true,
    "write_log_diffs": false,
    "write_watched_indices": [
      "write-index-1",
      "write-index-2",
      "log-*",
      "*"
    ],
    "write_ignore_users": [
      "write-ignore-1"
    ]
  }
}
```

Surveillance des événements OpenSearch liés au service avec Amazon EventBridge

Amazon OpenSearch Service s'intègre EventBridge à Amazon pour vous informer de certains événements qui affectent vos domaines. Les événements des AWS services sont fournis à presque EventBridge en temps réel. Les mêmes événements sont également envoyés à Amazon CloudWatch Events, le prédécesseur d'Amazon EventBridge. Vous pouvez écrire des règles simples pour indiquer quels événements vous intéressent et les actions automatisées à effectuer quand un événement correspond à une règle. Les actions pouvant être déclenchées automatiquement sont les suivantes :

- Appel d'une fonction AWS Lambda
- Appel d'une EC2 Run Command Amazon
- Relais de l'événement à Amazon Kinesis Data Streams
- Activation d'une machine d'état AWS Step Functions
- Notification d'une rubrique Amazon SNS ou d'une file d'attente Amazon SQS

Pour plus d'informations, consultez Commencer avec Amazon EventBridge dans le guide de EventBridge l'utilisateur Amazon.

Rubriques

- Événements de mise à jour du logiciel de service
- Événements Auto-Tune
- Événements relatifs à l'état du cluster
- Événements de point de terminaison d'un VPC
- Événements de retrait de nœuds
- Événements de retrait de nœuds dégradés
- Événements d'erreur de domaine
- Didacticiel : Écoute des EventBridge événements Amazon OpenSearch Service
- Tutoriel : Envoi d'alertes Amazon SNS pour les mises à jour logicielles disponibles

Événements de mise à jour du logiciel de service

OpenSearch Le service envoie des événements au EventBridge moment où l'un des événements de mise à jour du logiciel de service suivants se produit.

Mise à jour du logiciel de service disponible

OpenSearch Service envoie cet événement lorsqu'une mise à jour du logiciel de service est disponible.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
```

```
    "event": "Service Software Update",
    "status": "Available",
    "severity": "Informational",
    "description": "Service software update R20220928 available. Service Software
 Deployment Mechanism:
                    Blue/Green. For more information on deployment configuration,
 please
                    see: https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/managedomains-configuration-changes.html"
  }
}
```

Mise à jour du logiciel de service programmée

OpenSearch Service envoie cet événement lorsqu'une mise à jour du logiciel de service a été planifiée. Pour les mises à jour facultatives, vous recevez la notification à la date prévue et vous avez la possibilité de les replanifier à tout moment. Pour les mises à jour requises, vous recevez la notification trois jours avant la date prévue, et vous avez la possibilité de la replanifier dans la fenêtre obligatoire.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Scheduled",
    "severity": "High",
    "description": "A new service software update [R20200330-p1] has been scheduled at
 [21st May 2023 12:40 GMT].
                    Please see documentation for more information on scheduling
 software updates:
                    https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/service-software.html."
```

```
  }
}
```

Mise à jour du logiciel de service reprogrammée

OpenSearch Service envoie cet événement lorsqu'une mise à jour du logiciel de service optionnelle a été reprogrammée. Pour de plus amples informations, veuillez consulter the section called “Mises à jour facultatives ou obligatoires”.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Rescheduled",
    "severity": "High",
    "description": "The service software update [R20200330-p1], which was originally
 scheduled for
                    [21st May 2023 12:40 GMT], has been rescheduled to [23rd May 2023
 12:40 GMT].
                    Please see documentation for more information on scheduling
 software updates:
                    https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/service-software.html."
  }
}
```

La mise à jour du logiciel de service a débuté

OpenSearch Service envoie cet événement au début d'une mise à jour du logiciel de service.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Started",
    "severity": "Informational",
    "description": "Service software update [R20200330-p1] started.
  }
}
```

Mise à jour du logiciel de service terminée

OpenSearch Service envoie cet événement lorsqu'une mise à jour du logiciel de service est terminée.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Completed",
    "severity": "Informational",
    "description": "Service software update [R20200330-p1] completed."
  }
}
```

Mise à jour du logiciel de service annulée

OpenSearch Service envoie cet événement lorsqu'une mise à jour du logiciel de service a été annulée.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Cancelled",
    "severity": "Informational",
    "description": "The scheduled service software update [R20200330-p1] has been
 cancelled as a
                   newer update is available. Please schedule the latest update."
  }
}
```

Mise à jour du logiciel de service programmée

OpenSearch Service envoie cet événement lorsqu'une mise à jour du logiciel de service qui était précédemment planifiée pour le domaine a été annulée.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
```

```
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Cancelled",
    "severity": "Informational",
    "description": "The scheduled service software update [R20200330-p1] has been
 cancelled."
  }
}
```

Mise à jour du logiciel de service non exécutée

OpenSearch Le service envoie cet événement lorsqu'il ne peut pas lancer une mise à jour du logiciel de service.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Unexecuted",
    "severity": "Informational",
    "description": "The scheduled service software update [R20200330-p1] cannot be
 started. Reason: [reason]"
  }
}
```

Échec de la mise à jour du logiciel de service

OpenSearch Service envoie cet événement lorsqu'une mise à jour du logiciel de service a échoué.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
    "status": "Failed",
    "severity": "High",
    "description": "Installation of service software update [R20200330-p1] failed.
 [reason].
  }
}
```

Mise à jour du logiciel de service requise

OpenSearch Service envoie cet événement lorsqu'une mise à jour du logiciel de service est requise. Pour de plus amples informations, veuillez consulter the section called “Mises à jour facultatives ou obligatoires”.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Software Update Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2016-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Service Software Update",
```

```
    "status": "Required",
    "severity": "High",
    "description": "Service software update [R20200330-p1] available. Update
                    will be automatically installed after [21st May 2023] if no
                    action is taken. Service Software Deployment Mechanism: Blue/Green.
                    For more information on deployment configuration, please see:
                    https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/managedomains-configuration-changes.html"
  }
}
```

Événements Auto-Tune

OpenSearch Le service envoie des événements au EventBridge moment où l'un des événements de Auto-Tune suivants se produit.

Auto-Tune en attente

OpenSearch Service envoie cet événement lorsqu'Auto-Tune a identifié des recommandations de réglage pour améliorer les performances et la disponibilité du cluster. Cet événement s'applique uniquement aux domaines pour lesquels Auto-Tune est désactivé.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Pending",
    "description": "Auto-Tune recommends the following new settings for your
 domain: { JVM Heap size : 60%}. Enable Auto-Tune to improve cluster stability and
 performance.",
    "scheduleTime": "{iso8601-timestamp}"
```

```
  }
}
```

Auto-Tune démarré

OpenSearch Le service envoie cet événement lorsqu'Auto-Tune commence à appliquer de nouveaux paramètres à votre domaine.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Started",
    "scheduleTime": "{iso8601-timestamp}",
    "startTime": "{iso8601-timestamp}",
    "description" : "Auto-Tune is applying the following settings to your domain: { JVM
 Heap size : 60%}."
  }
}
```

Auto-Tune requiert un déploiement bleu/vert planifié

OpenSearch Service envoie cet événement lorsqu'Auto-Tune a identifié des recommandations de réglage qui nécessitent un déploiement bleu/vert planifié.

Exemple

Voici un exemple d'événement de ce type :

```
{
```

```
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Low",
    "status": "Pending",
    "startTime": "{iso8601-timestamp}",
    "description": "Auto-Tune has identified the following settings for your domain
 that require a blue/green deployment: { JVM Heap size : 60%}.
                   You can schedule the deployment for your preferred time."
  }
}
```

Auto-Tune annulé

OpenSearch Le service envoie cet événement lorsque la planification Auto-Tune a été annulée, car il n'y a aucune recommandation de réglage en attente.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Low",
    "status": "Cancelled",
    "scheduleTime": "{iso8601-timestamp}",
    "description": "Auto-Tune has cancelled the upcoming blue/green deployment."
```

```
  }
}
```

Auto-Tune terminé

OpenSearch Service envoie cet événement lorsqu'Auto-Tune a terminé le déploiement bleu/vert et que le cluster est opérationnel avec les nouveaux paramètres JVM en place.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Completed",
    "completionTime": "{iso8601-timestamp}",
    "description": "Auto-Tune has completed the blue/green deployment and successfully
 applied the following settings: { JVM Heap size : 60%}."
  }
}
```

Auto-Tune désactivé et modifications annulées

OpenSearch Le service envoie cet événement lorsqu'Auto-Tune a été désactivé et que les modifications appliquées ont été annulées.

Exemple

Voici un exemple d'événement de ce type :

```
{
```

```
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": [ "arn:aws:es:us-east-1:123456789012:domain/test-domain" ],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Completed",
    "description": "Auto-Tune is now disabled. All settings have been reverted. Auto-
Tune will continue to evaluate
                    cluster performance and provide recommendations.",
    "completionTime": "{iso8601-timestamp}"
  }
}
```

Auto-Tune désactivé et modifications conservées

OpenSearch Le service envoie cet événement lorsqu'Auto-Tune a été désactivé et que les modifications appliquées ont été conservées.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "3acb26c8-397c-4c89-a80a-ce672a864c55",
  "detail-type": "Amazon OpenSearch Service Auto-Tune Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2020-10-30T22:06:31Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Auto-Tune Event",
    "severity": "Informational",
    "status": "Completed",
    "description": "Auto-Tune is now disabled. The most-recent settings by Auto-Tune
 have been retained.
```

```
                    Auto-Tune will continue to evaluate cluster performance and provide
 recommendations.",
    "completionTime": "{iso8601-timestamp}"
  }
}
```

Événements relatifs à l'état du cluster

OpenSearch Le service envoie certains événements EventBridge lorsque l'état de votre cluster est compromis.

La récupération de cluster rouge a commencé

OpenSearch Le service envoie cet événement lorsque l'état de votre cluster est en rouge continu depuis plus d'une heure. Il tente de restaurer automatiquement un ou plusieurs index rouges à partir d'un instantané afin de corriger l'état du cluster.

Exemple

Voici un exemple d'événement de ce type :

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Started",
      "severity":"High",
      "description":"Your cluster status is red. We have started automatic snapshot
 restore for the red indices.
                    No action is needed from your side. Red indices [red-index-0, red-
index-1]"
   }
}
```

Récupération du cluster rouge partiellement terminée

OpenSearch Service envoie cet événement lorsqu'il n'a pu restaurer qu'un sous-ensemble d'index rouges à partir d'un instantané lors d'une tentative de réparation d'un état de cluster rouge.

Exemple

Voici un exemple d'événement de ce type :

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Partially Restored",
      "severity":"High",
      "description":"Your cluster status is red. We were able to restore the following
 Red indices from
                   snapshot: [red-index-0]. Indices not restored: [red-index-1].
 Please refer https://docs.aws.amazon.com/opensearch-service/latest/developerguide/
handling-errors.html#handling-errors-red-cluster-status for troubleshooting steps."
   }
}
```

La récupération de cluster rouge a échoué

OpenSearch Le service envoie cet événement lorsqu'il ne parvient pas à restaurer les index lors d'une tentative de réparation d'un état de cluster rouge.

Exemple

Voici un exemple d'événement de ce type :

```
{
   "version":"0",
```

```
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Failed",
      "severity":"High",
      "description":"Your cluster status is red. We were unable to restore the Red
 indices automatically.
                   Indices not restored: [red-index-0, red-index-1]. Please refer
 https://docs.aws.amazon.com/opensearch-service/latest/developerguide/handling-
errors.html#handling-errors-red-cluster-status for troubleshooting steps."
   }
}
```

Partitions à supprimer

OpenSearch Service envoie cet événement lorsqu'il a tenté de réparer automatiquement l'état de votre cluster rouge après qu'il ait été rouge en continu pendant 14 jours, mais qu'un ou plusieurs index restent rouges. Après 7 jours supplémentaires (21 jours de rouge continu), OpenSearch Service procède à la suppression des partitions non attribuées sur tous les index rouges.

Exemple

Voici un exemple d'événement de ce type :

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2022-04-09T10:36:48Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
```

```
   "detail":{
      "severity":"Medium",
      "description":"Your cluster status is red. Please fix the red indices as soon as
 possible.
                    If not fixed by 2022-04-12 01:51:47+00:00, we will delete all
 unassigned shards,
                    the unit of storage and compute, for these red indices to recover
 your domain and make it green.
                    Please refer to https://docs.aws.amazon.com/opensearch-service/
latest/developerguide/handling-errors.html#handling-errors-red-cluster-status for
 troubleshooting steps.
                    test_data, test_data1",
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Shard(s) to be deleted"
   }
}
```

Partitions supprimées

OpenSearch Le service envoie cet événement lorsque l'état de votre cluster est en rouge continu depuis plus de 21 jours. Il supprime les partitions non attribuées (stockage et calcul) sur tous les index rouges. Pour en savoir plus, consultez the section called “Correction automatique des clusters rouges”.

Exemple

Voici un exemple d'événement de ce type :

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Cluster Status Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2022-04-09T10:54:48Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "severity":"High",
      "description":"We have deleted unassinged shards, the unit of storage and
 compute, in
```

```
                    red indices: index-1, index-2 because these indices were red for
 more than
                    21 days and could not be restored with the automated restore
 process.
                    Please refer to https://docs.aws.amazon.com/opensearch-service/
latest/developerguide/handling-errors.html#handling-errors-red-cluster-status for
 troubleshooting steps.",
      "event":"Automatic Snapshot Restore for Red Indices",
      "status":"Shard(s) deleted"
   }
}
```

Avertissement sur le nombre élevé de partitions

OpenSearch Service envoie cet événement lorsque le nombre moyen de partitions sur vos nœuds de données chauds a dépassé 90 % de la limite recommandée par défaut de 1 000. Bien que les versions ultérieures d'Elasticsearch prennent en OpenSearch charge une limite configurable du nombre maximal de partitions par nœud, nous vous recommandons de ne pas avoir plus de 1 000 partitions par nœud. Consultez Choix du nombre de partitions.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2016-11-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"High Shard Count",
     "status":"Warning",
     "severity":"Low",
     "description":"One or more data nodes have close to 1000 shards. To ensure optimum
 performance and stability of your
                    cluster, please refer to the best practice guidelines - https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/sizing-domains.html#bp-
sharding."
  }
```

```
}
```

Limite du nombre de partitions dépassée

OpenSearch Service envoie cet événement lorsque le nombre moyen de partitions sur vos nœuds de données chauds a dépassé la limite recommandée par défaut de 1 000. Bien que les versions ultérieures d'Elasticsearch prennent en OpenSearch charge une limite configurable du nombre maximal de partitions par nœud, nous vous recommandons de ne pas avoir plus de 1 000 partitions par nœud. Consultez Choix du nombre de partitions.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2016-11-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"High Shard Count",
     "status":"Warning",
     "severity":"Medium",
     "description":"One or more data nodes have more than 1000 shards. To ensure
 optimum performance and stability of your
                   cluster, please refer to the best practice guidelines - https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/sizing-domains.html#bp-
sharding."
  }
}
```

Espace disque faible

OpenSearch Service envoie cet événement lorsqu'un ou plusieurs nœuds de votre cluster dispose de moins de 25 % de l'espace de stockage disponible, ou moins de 25 Go.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Low Disk Space",
     "status":"Warning",
     "severity":"Medium",
     "description":"One or more data nodes in your cluster has less than 25% of storage
 space or less than 25GB.
                  Your cluster will be blocked for writes at 20% or 20GB. Please refer
 to the documentation for more information - https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/handling-errors.html#troubleshooting-cluster-block"
  }
}
```

Faible violation du filigrane du disque

OpenSearch Service envoie cet événement lorsque tous les nœuds de votre cluster disposent de moins de 10 % de l'espace de stockage disponible, soit moins de 10 Go. Lorsque tous les nœuds franchissent le filigrane du disque bas, tout nouvel index produit un cluster jaune, et lorsque tous les nœuds tombent en dessous du filigrane du disque haut, un cluster rouge apparaît.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
```

```
  "detail":{
     "event":"Low Disk Watermark Breach",
     "status":"Warning",
     "severity":"Medium",
     "description":"Low Disk Watermark threshold is about to be breached. Once the
 threshold is breached, new index creation will be blocked on all
                   nodes to prevent the cluster status from turning red. Please
 increase disk size to suit your storage needs. For more information,
                   see https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/handling-errors.html#troubleshooting-cluster-block".
  }
}
```

Solde de débordement EBS inférieur à 70 %

OpenSearch Le service envoie cet événement lorsque le solde de débordement EBS sur un ou plusieurs nœuds de données tombe en dessous de 70 %. L'épuisement du solde de débordement EBS peut provoquer une indisponibilité généralisée du cluster et une limitation des demandes d'E/S, ce qui peut entraîner des latences élevées et des délais d'attente pour les demandes d'indexation et de recherche. Pour connaître les étapes permettant de résoudre ce problème, consultez the section called “Solde de débordement EBS faible”.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"EBS Burst Balance",
     "status":"Warning",
     "severity":"Medium",
     "description":"EBS burst balance on one or more data nodes is below 70%.
                   Follow https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/handling-errors.html#handling-errors-low-ebs-burst
```

```
                    to fix this issue."
  }
}
```

Solde de débordement EBS inférieur à 20 %

OpenSearch Le service envoie cet événement lorsque le solde de débordement EBS sur un ou plusieurs nœuds de données tombe en dessous de 20 %. L'épuisement du solde de débordement EBS peut provoquer une indisponibilité généralisée du cluster et une limitation des demandes d'E/S, ce qui peut entraîner des latences élevées et des délais d'attente pour les demandes d'indexation et de recherche. Pour connaître les étapes permettant de résoudre ce problème, consultez the section called “Solde de débordement EBS faible”.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"EBS Burst Balance",
     "status":"Warning",
     "severity":"High",
     "description":"EBS burst balance on one or more data nodes is below 20%.
                    Follow https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/handling-errors.html#handling-errors-low-ebs-burst
                    to fix this issue.
  }
}
```

Limitation de débit de disque

OpenSearch Service envoie cet événement lorsque les demandes de lecture et d'écriture envoyées à votre domaine sont réduites en raison des limites de débit de vos volumes ou de votre instance

EBS. EC2 Si vous recevez cette notification, envisagez d'augmenter la capacité de vos volumes ou instances en suivant les bonnes pratiques AWS recommandées par. Si votre type de volume est le gp2 même, augmentez la taille du volume. Si votre type de volume est le casgp3, augmentez le débit. Vous pouvez également vérifier que votre base d'instances et votre débit EBS maximal sont supérieurs ou égaux au débit du volume provisionné, et que vous pouvez les augmenter en conséquence.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Disk Throughput Throttle",
     "status":"Warning",
     "severity":"Medium",
     "description":"Your domain is experiencing throttling due to instance or volume
 throughput limitations.
                   Please consider scaling your domain to suit your throughput needs.
 In July 2023, we improved
                   the accuracy of throughput throttle calculation by replacing ‘Max
 volume throughput’ with
                   ‘Provisioned volume throughput’. Please refer to the documentation
 for more information."
   }
}
```

Grande taille de partition

OpenSearch Service envoie cet événement lorsqu'une ou plusieurs partitions de votre cluster a dépassé 50 ou plus de 50 ou plus de 50 ou plus de 65 Go. Pour garantir des performances et une stabilité optimales du cluster, réduisez la taille des partitions.

Pour plus d'informations, consultez les bonnes pratiques en matière de partitionnement.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Large Shard Size",
     "status":"Warning",
     "severity":"Medium",
     "description":"One or more shards are larger than 65GiB. To ensure optimum cluster
 performance and stability, reduce shard sizes.
                    For more information, see https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/monitoring-events.html#monitoring-events-large-shard-
size."
  }
}
```

Utilisation JVM élevée

OpenSearch Le service envoie cet événement lorsque la `JVMMemoryPressure` métrique de votre domaine dépasse 80 %. S'il dépasse 92 % pendant 30 minutes, toutes les opérations d'écriture sur votre cluster seront bloquées. Pour garantir une stabilité optimale du cluster, réduisez le trafic vers le cluster ou dimensionnez votre domaine afin de fournir suffisamment de mémoire pour votre charge de travail.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
```

```
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"High JVM Usage",
     "status":"Warning",
     "severity":"High",
     "description":"JVM memory pressure has exceeded 80%. If it exceeds 92% for 30
 minutes, all write operations to your cluster
                   will be blocked. To ensure optimum cluster stability, reduce
 traffic to the cluster or use larger instance types.
                   For more information, see https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/monitoring-events.html#monitoring-events-high-jvm."
  }
}
```

GC insuffisant

OpenSearch Le service envoie cet événement lorsque le maximum de JVM est supérieur à 70 % et que la différence entre le maximum et le minimum est inférieure à 30 %. Cela peut indiquer que la JVM n'est pas en mesure de récupérer suffisamment de mémoire pendant les cycles de collecte des déchets pour votre charge de travail. Cela peut entraîner des réponses de plus en plus lentes et des latences plus élevées ; et dans certains cas, même des pertes de nœuds en raison de l'expiration du délai imparti pour les tests de santé. Pour garantir une stabilité optimale du cluster, réduisez le trafic vers le cluster ou dimensionnez votre domaine afin de fournir suffisamment de mémoire pour votre charge de travail.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
```

```
     "event":"Insufficient GC",
     "status":"Warning",
     "severity":"Medium",
     "description":"Maximum JVM is above 70% and JVM range is less than 30%. This may
 indicate insufficient garbage collection for your workload.
                   For more information, see https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/monitoring-events.html#monitoring-events-insufficient-
gc."
  }
}
```

Avertissement de routage d'index personnalisé

OpenSearch Le service envoie cet événement lorsque votre domaine est en cours de traitement et contient des index avec des paramètres index.routing.allocation personnalisés, ce qui peut bloquer les déploiements bleu-vert. Vérifiez que les paramètres sont correctement appliqués.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Custom Index Routing Warning",
     "status":"Warning",
     "severity":"Medium",
     "description":"Your domain is in processing state and contains indice(s) with
 custom index.routing.allocation
                   settings which can cause blue-green deployments to get stuck.
 Verify settings are applied properly.
                   For more information, see https://docs.aws.amazon.com/opensearch-
service/latest/developerguide/monitoring-events.html#monitoring-events-index-routing."
  }
}
```

Mise hors service d'instance

OpenSearch Le service envoie cet événement lorsque votre domaine est défectueux en raison de partitions non attribuées avec. `[ShardLockObtainFailedException]` Pour plus d'informations, consultez Comment résoudre l'exception de verrouillage des partitions en mémoire dans Amazon OpenSearch Service ?

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2017-12-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Failed Shard Lock",
     "status":"Warning",
     "severity":"Medium",
     "description":"Your domain is unhealthy due to unassigned shards with
 [ShardLockObtainFailedException]. For more information,
                    see https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/monitoring-events.html#monitoring-events-failed-shard-lock."
}
```

Événements de point de terminaison d'un VPC

OpenSearch Le service envoie certains événements EventBridge liés aux points de terminaison AWS PrivateLink d'interface.

Échec de la création d'un point de terminaison d'un VPC

OpenSearch Le service envoie cet événement lorsqu'il n'est pas en mesure de créer un point de terminaison d'un VPC demandé. Cette erreur peut survenir parce que vous avez atteint la limite du nombre de points de terminaison d'un VPC autorisés au sein d'une région. Cette erreur s'affichera également si un sous-réseau ou un groupe de sécurité spécifié n'existe pas.

Exemple

Voici un exemple d'événement de ce type :

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service VPC Endpoint Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"VPC Endpoint Create Validation",
      "status":"Failed",
      "severity":"High",
      "description":"Unable to create VPC endpoint aos-0d4c74c0342343 for domain
                     arn:aws:es:eu-south-1:123456789012:domain/my-domain due to the
 following validation failures: You've reached the limit on the
                     number of VPC endpoints that you can create in the AWS Region."
   }
}
```

Échec de la mise à jour d'un point de terminaison d'un VPC

OpenSearch Le service envoie cet événement lorsqu'il n'est pas en mesure de supprimer un point de terminaison d'un VPC demandé.

Exemple

Voici un exemple d'événement de ce type :

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service VPC Endpoint Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
```

```
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"VPC Endpoint Update Validation",
      "status":"Failed",
      "severity":"High",
      "description":"Unable to update VPC endpoint aos-0d4c74c0342343 for domain
                    arn:aws:es:eu-south-1:123456789012:domain/my-domain due to the
 following validation failures: <failure message>."
   }
}
```

Échec de la suppression d'un point de terminaison d'un VPC

OpenSearch Le service envoie cet événement lorsqu'il n'est pas en mesure de supprimer un point de terminaison d'un VPC demandé.

Exemple

Voici un exemple d'événement de ce type :

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service VPC Endpoint Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"VPC Endpoint Delete Validation",
      "status":"Failed",
      "severity":"High",
      "description":"Unable to delete VPC endpoint aos-0d4c74c0342343 for domain
                    arn:aws:es:eu-south-1:123456789012:domain/my-domain due to the
 following validation failures: Specified subnet doesn't exist."
   }
}
```

Événements de retrait de nœuds

OpenSearch Le service envoie des événements au EventBridge moment où l'un des événements de retrait de nœuds suivants se produit.

Mise hors service d'un nœud programmée

OpenSearch Le service envoie cet événement lorsqu'un retrait d'un nœud a été programmé.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2023-04-07T10:07:33Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Node Retirement Notification",
    "status": "Scheduled",
    "severity": "Medium",
    "description": "An automated action to retire and replace a node has been scheduled
 on your domain.
                    The node will be replaced in the next off-peak window. For more
 information, see
                    https://docs.aws.amazon.com/opensearch-service/latest/
developerguide/monitoring-events.html."
  }
}
```

Mise hors service d'un nœud terminée

OpenSearch Le service envoie cet événement lorsqu'un retrait d'un nœud est terminé.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2023-04-07T10:07:33Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Node Retirement Notification",
    "status": "Completed",
    "severity": "Medium",
    "description": "The node has been retired and replaced with a new node."
  }
}
```

Échec du retrait du nœud

OpenSearch Le service envoie cet événement lorsqu'un retrait d'un nœud a échoué.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Amazon OpenSearch Service Notification",
  "source": "aws.es",
  "account": "123456789012",
  "time": "2023-04-07T10:07:33Z",
  "region": "us-east-1",
  "resources": ["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail": {
    "event": "Node Retirement Notification",
    "status": "Failed",
    "severity": "Medium",
    "description": "Node retirement failed. No actions are required from your end. We
 will automatically
                    retry replacing the node."
  }
}
```

Événements de retrait de nœuds dégradés

OpenSearch Le service envoie ces événements lorsqu'un remplacement de nœud est nécessaire en raison de la dégradation du matériel d'un nœud.

Notification de retrait d'un nœud dégradé

OpenSearch Le service envoie cet événement lorsque l'action automatique de retrait et de remplacement d'un nœud dégradé a été planifiée pour votre domaine.

Exemple

Voici un exemple d'événement de ce type :

```
{
   "version":"0",
   "id":"db233454-aad1-7676-3b15-10a84b052baa",
   "detail-type":"Amazon OpenSearch Service Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2024-01-11T08:16:06Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-node-replacement"
   ],
   "detail":{
      "severity":"Medium",
      "description":"An automated action to retire and replace a node has
 been scheduled on your domain. For more information, please see https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/monitoring-events.html.",
      "event":"Degraded Node Retirement Notification",
      "status":"Scheduled"
   }
}
```

Le retrait du nœud dégradé est terminé

OpenSearch Le service envoie cet événement lorsqu'un nœud dégradé a été retiré et remplacé par un nouveau nœud.

Exemple

Voici un exemple d'événement de ce type :

```
{
   "version":"0",
   "id":"7444215c-90f9-a52d-bcda-e85973a9a762",
   "detail-type":"Amazon OpenSearch Service Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2024-01-11T10:20:30Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-node-replacement"
   ],
   "detail":{
      "severity":"Medium",
      "description":"The node has been retired and replaced with a new node.",
      "event":"Degraded Node Retirement Notification",
      "status":"Completed"
   }
}
```


Le retrait du nœud dégradé a échoué

OpenSearch Le service envoie cet événement en cas d'échec du retrait du nœud dégradé.

Exemple

Voici un exemple d'événement de ce type :


```
{
   "version":"0",
   "id":"c328e9bb-93b9-c0b2-b17a-df527fdf96b6",
   "detail-type":"Amazon OpenSearch Service Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2024-01-11T08:31:38Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-node-replacement"
   ],
   "detail":{
      "severity":"Medium",
      "description":"Node retirement failed. No actions are required from your end. We
 will automatically re-try replacing the node.",
      "event":"Degraded Node Retirement Notification",
```

```
      "status":"Failed"
   }
}
```

Événements d'erreur de domaine

OpenSearch Le service envoie des événements EventBridge lorsque l'une des erreurs de domaine suivantes se produit.

Échec de validation de la mise à jour du domaine

OpenSearch Service envoie cet événement s'il rencontre un ou plusieurs échecs de validation lorsqu'il tente de mettre à jour ou d'effectuer un changement de configuration sur un domaine. Pour connaître les étapes de résolution de ces échecs, consultez the section called “Résolution des erreurs de validation”.

Exemple

Voici un exemple d'événement de ce type :

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-012345678901",
   "detail-type":"Amazon OpenSearch Service Domain Update Notification",
   "source":"aws.es",
   "account":"123456789012",
   "time":"2016-11-01T13:12:22Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:es:us-east-1:123456789012:domain/test-domain"
   ],
   "detail":{
      "event":"Domain Update Validation",
      "status":"Failed",
      "severity":"High",
      "description":"Unable to perform updates to your domain due to the following
 validation failures: <failures>
                    Please see the documentation for more information https://
docs.aws.amazon.com/opensearch-service/latest/developerguide/managedomains-
configuration-changes.html#validation"
   }
}
```

Clé KMS inaccessible

OpenSearch Le service envoie cet événement lorsqu'il ne peut pas accéder à votre AWS KMS clé.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Domain Error Notification",
  "source":"aws.es",
  "account":"123456789012",
  "time":"2016-11-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"KMS Key Inaccessible",
     "status":"Error",
     "severity":"High",
     "description":"The KMS key associated with this domain is inaccessible. You are at
 risk of losing access to your domain.
                    For more information, please refer to https://docs.aws.amazon.com/
opensearch-service/latest/developerguide/encryption-at-rest.html#disabled-key."
  }
}
```

Isolement de domaine

OpenSearch Le service envoie cet événement lorsque votre domaine est isolé et ne peut pas recevoir, lire ou écrire de demandes car il est inaccessible par le réseau.

Exemple

Voici un exemple d'événement de ce type :

```
{
  "version":"0",
  "id":"01234567-0123-0123-0123-012345678901",
  "detail-type":"Amazon OpenSearch Service Notification",
  "source":"aws.es",
```

```
  "account":"123456789012",
  "time":"2023-11-01T13:12:22Z",
  "region":"us-east-1",
  "resources":["arn:aws:es:us-east-1:123456789012:domain/test-domain"],
  "detail":{
     "event":"Domain Isolation Notification",
     "status":"Error",
     "severity":"High",
     "description":"Your OpenSearch Service domain has been isolated. An isolated
 domain is unreachable by network and cannot receive, read, or write requests. For more
 information and assistance, please contact AWS Support at https://docs.aws.amazon.com/
opensearch-service/latest/developerguide/encryption-at-rest.html#disabled-key."
  }
}
```

Didacticiel : Écoute des EventBridge événements Amazon OpenSearch Service

Dans le cadre de ce tutoriel, vous allez configurer une AWS Lambda fonction simple qui écoutera les événements Amazon OpenSearch Service et les écrira dans un flux de CloudWatch journaux.

Prérequis

Ce didacticiel part du principe que vous disposez d'un domaine OpenSearch de service existant. Si vous n'avez pas encore créé de domaine, suivez la procédure décrite dans Création et gestion des domaines pour en créer un.

Étape 1 : Créer la fonction Lambda

Au cours de cette procédure, vous allez créer une fonction Lambda simple à utiliser comme cible pour les messages d'événements OpenSearch de service.

Pour créer une fonction Lambda cible

1. Ouvrez la AWS Lambda console à l'adresse https://console.aws.amazon.com/lambda/.
2. Choisissez Create function (Créer une fonction) et Author from scratch (Créer à partir de zéro).
3. Dans le champ Function name (Nom de la fonction), saisissez event-handler.
4. Pour Environnement d’exécution, sélectionnez Python 3.8.
5. Sélectionnez Create function (Créer une fonction).

6. Dans la section Code de fonction, modifiez l'exemple de code selon l'exemple suivant :

```
import json

def lambda_handler(event, context):
    if event["source"] != "aws.es":
        raise ValueError("Function only supports input from events with a source
 type of: aws.es")

    print(json.dumps(event))
```

   Il s'agit d'une fonction Python 3.8 simple qui imprime les événements envoyés par OpenSearch Service. Si tout est correctement configuré, à la fin de ce tutoriel, les détails de l'événement apparaissent dans le flux de CloudWatch journaux qui est associé à cette fonction Lambda.
7. Choisissez Déployer.

Étape 2 : Enregistrer une règle d'événement

Au cours de cette étape, vous allez créer une EventBridge règle pour capturer les événements de vos domaines de OpenSearch service. Cette règle capturera tous les événements du compte où elle est définie. Les messages d'événement eux-mêmes contiennent des informations sur la source de l'événement, comme le domaine d'origine. Vous pouvez utiliser ces informations pour filtrer et trier les événements par programmation.

Pour créer une EventBridge règle

1. Ouvrez la EventBridge console à l'adresse https://console.aws.amazon.com/events/.
2. Choisissez Créer une règle.
3. Nommez la règle event-rule.
4. Choisissez Next (Suivant).
5. Pour le modèle d'événement, sélectionnez AWS services, Amazon OpenSearch Service et Tous les événements. Ce modèle s'applique à tous vos domaines de OpenSearch service et à chaque événement OpenSearch de service. Vous pouvez également créer un modèle plus spécifique pour filtrer certains résultats.
6. Appuyez sur Next (Suivant).
7. Pour la cible, choisissez Lambda Function (Fonction Lambda). Dans la liste déroulante des fonctions, choisissez event-handler (gestionnaire d'événements).

8. Appuyez sur Next (Suivant).
9. Ignorez les identifications et appuyez à nouveau sur Next (Suivant).
10. Vérifiez la configuration et choisissez Create rule (Créer une règle).

Étape 3 : Tester votre configuration

La prochaine fois que vous recevrez une notification dans la section Notifications de la console de OpenSearch service, si tout est correctement configuré, votre fonction Lambda sera déclenchée et écrira les données d'événement dans un flux de CloudWatch journaux journaux pour la fonction.

Pour tester votre configuration

1. Ouvrez la CloudWatch console à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Dans le panneau de navigation, choisissez Logs (Journaux) et sélectionnez le groupe de journaux pour votre fonction Lambda (par exemple,/aws/lambda/event-handler).
3. Sélectionnez un flux de journaux pour afficher les données d'événement.

Tutoriel : Envoi d'alertes Amazon SNS pour les mises à jour logicielles disponibles

Dans le cadre de ce tutoriel, vous allez configurer une règle d' EventBridge événement Amazon qui capturera les notifications de mises à jour disponibles du logiciel de OpenSearch service dans Amazon Service et vous enverra une notification par e-mail via Amazon Simple Notification Service (Amazon SNS).

Prérequis

Ce didacticiel part du principe que vous disposez d'un domaine OpenSearch de service existant. Si vous n'avez pas encore créé de domaine, suivez la procédure décrite dans Création et gestion des domaines pour en créer un.

Étape 1 : Créer une rubrique Amazon SNS et s'y abonner

Configurez une rubrique Amazon SNS à utiliser comme une cible de l'événement pour votre nouvelle règle d'événement.

Pour créer une cible Amazon SNS

1. Ouvrez la console Amazon SNS à partir de l'adresse v3/home. https://console.aws.amazon.com/sns/
2. Choisissez Topics (Rubriques) et Create topic (Créer une rubrique).
3. Pour le type de tâche, choisissez Standard, et nommez la tâche software-update.
4. Choisissez Create topic (Créer une rubrique).
5. Une fois la rubrique créée, choisissez Create subscription (Créer un abonnement).
6. Pour Protocole, choisissez E-mail. Dans le champ Endpoint (Point de terminaison), saisissez l'adresse e-mail à laquelle vous avez actuellement accès et choisissez Create subscription (Créer un abonnement).
7. Vérifiez votre compte de messagerie et attendez de recevoir un e-mail de confirmation de l'abonnement. Lorsque vous le recevez, choisissez Confirm subscription (Confirmer l'abonnement).

Étape 2 : Enregistrer une règle d'événement

Ensuite, enregistrez une règle d'événement pour ne capturer que les événements de mise à jour du logiciel de service.

Pour créer une règle d'événement

1. Ouvrez la EventBridge console à l'adresse https://console.aws.amazon.com/events/.
2. Choisissez Créer une règle.
3. Nommez la règle softwareupdate-rule.
4. Choisissez Next (Suivant).
5. Pour le modèle d'événement, sélectionnez AWS services, Amazon OpenSearch Service et Amazon OpenSearch Service Software Update Notification. Ce modèle correspond à tout événement de mise à jour du logiciel de OpenSearch service de Service. Pour plus d'informations sur les modèles d'événements, consultez la section Modèles EventBridge d'événements Amazon dans le guide de EventBridge l'utilisateur Amazon.
6. Le cas échéant, vous pouvez filtrer pour n'inclure que des sévérités spécifiques. Pour connaître la sévérité de chaque événement, veuillez consulter the section called “Événements de mise à jour du logiciel de service”.
7. Choisissez Next (Suivant).
8. Dans la cible, choisissez SNS topic (Rubrique SNS) et sélectionnez software-update (mise à jour du logiciel).

9. Choisissez Next (Suivant).
10. Ignorez les identifications et choisissez Next (Suivant).
11. Vérifiez la configuration de la règle et choisissez Create rule (Créer une règle).

La prochaine fois que vous recevrez une notification du OpenSearch service pour vous indiquer qu'une mise à jour du logiciel de service disponible, si tout est correctement configuré, Amazon SNS vous enverra une alerte par e-mail concernant la mise à jour.

Surveillance des appels OpenSearch d'API Amazon Service avec AWS CloudTrail

Amazon OpenSearch Service s'intègre AWS CloudTrailà un service qui fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un AWS service dans OpenSearch Service. CloudTrail capture tous les appels d'API de configuration pour le OpenSearch service sous forme d'événements.

> **Note**
>
> CloudTrail capture uniquement les appels à l'API de configuration, tels que `CreateDomain` et`GetUpgradeStatus`. CloudTrail ne capture pas les appels vers OpenSearch APIs, tels que `_search` et`_bulk`. Pour ces appels, consultez the section called “Surveillance des journaux d'audit”.

Les appels capturés incluent des appels provenant de la console de OpenSearch AWS CLI service ou d'un AWS SDK. Si vous créez un suivi, vous pouvez activer la diffusion continue des CloudTrail événements vers un compartiment Amazon S3, y compris des événements pour le OpenSearch service. Si vous ne configurez pas de suivi, vous pouvez toujours consulter les événements les plus récents sur la CloudTrail console dans Historique des événements. À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été faite au OpenSearch Service, l'adresse IP à partir de laquelle la demande a été faite, qui a fait la demande, quand elle a été faite et des détails supplémentaires.

Pour en savoir plus CloudTrail, consultez le guide de AWS CloudTrail l'utilisateur.

Informations sur OpenSearch le service Amazon dans CloudTrail

CloudTrail est activé sur votre compte Compte AWS lorsque vous créez le compte. Lorsqu'une activité se produit dans le OpenSearch service, cette activité est enregistrée dans un CloudTrail événement avec les autres événements du AWS service dans l'historique des événements. Vous pouvez afficher, rechercher et télécharger les événements récents dans votre compte Compte AWS . Pour plus d'informations, consultez la section Affichage des événements avec l'historique des CloudTrail événements.

Pour un enregistrement continu des événements de votre Compte AWS compte, y compris les événements liés au OpenSearch service, créez un parcours. Un suivi permet CloudTrail de fournir des fichiers journaux à un compartiment Amazon S3. Par défaut, lorsque vous créez un journal d'activité dans la console, il s'applique à toutes les régions Régions AWS. Le journal enregistre les événements de toutes les régions de la AWS partition et transmet les fichiers journaux au compartiment Amazon S3 que vous spécifiez. En outre, vous pouvez configurer d'autres AWS services pour analyser plus en détail les données d'événements collectées dans les CloudTrail journaux et agir en conséquence. Pour plus d'informations, consultez les ressources suivantes :

- Création d'un parcours pour votre Compte AWS
- AWS intégrations de services avec Logs CloudTrail
- Configuration des notifications Amazon SNS pour CloudTrail
- Réception de fichiers CloudTrail journaux de plusieurs régions et réception de fichiers CloudTrail journaux de plusieurs comptes

Toutes les actions de l'API de configuration des OpenSearch services sont enregistrées CloudTrail et documentées dans le Amazon OpenSearch Service API Reference.

Chaque événement ou entrée de journal contient des informations sur la personne ayant initié la demande. Les informations relatives à l'identité permettent de déterminer les éléments suivants :

- Si la demande a été faite avec les informations d'identification de l'utilisateur root ou AWS Identity and Access Management (IAM)
- Si la demande a été effectuée avec des informations d'identification de sécurité temporaires pour un rôle ou un utilisateur fédéré
- Si la demande a été faite par un autre AWS service

Pour plus d'informations, consultez la section Élément userIdentity CloudTrail .

Comprendre les entrées du fichier journal Amazon OpenSearch Service

Un suivi est une configuration qui permet de transmettre des événements sous forme de fichiers journaux à un compartiment Amazon S3 que vous spécifiez. CloudTrail les fichiers journaux contiennent une ou plusieurs entrées de journal. Un événement représente une demande unique provenant de n'importe quelle source et inclut des informations sur l'action demandée, la date et l'heure de l'action, les paramètres de la demande, etc. CloudTrail les fichiers journaux ne constituent pas une trace ordonnée des appels d'API publics, ils n'apparaissent donc pas dans un ordre spécifique.

L'exemple suivant montre une entrée de CloudTrail journal illustrant l'`CreateDomain`opération :


```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::123456789012:user/test-user",
    "accountId": "123456789012",
    "accessKeyId": "access-key",
    "userName": "test-user",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2018-08-21T21:59:11Z"
      }
    },
    "invokedBy": "signin.amazonaws.com"
  },
  "eventTime": "2018-08-21T22:00:05Z",
  "eventSource": "es.amazonaws.com",
  "eventName": "CreateDomain",
  "awsRegion": "us-west-1",
  "sourceIPAddress": "123.123.123.123",
  "userAgent": "signin.amazonaws.com",
  "requestParameters": {
    "engineVersion": "OpenSearch_1.0",
    "clusterConfig": {
      "instanceType": "m4.large.search",
      "instanceCount": 1
    },
    "snapshotOptions": {
```

```
      "automatedSnapshotStartHour": 0
    },
    "domainName": "test-domain",
    "encryptionAtRestOptions": {},
    "eBSOptions": {
      "eBSEnabled": true,
      "volumeSize": 10,
      "volumeType": "gp2"
    },
    "accessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":[\"123456789012\"]},\"Action\":[\"es:*\"],\"Resource\":
\"arn:aws:es:us-west-1:123456789012:domain/test-domain/*\"}]}",
    "advancedOptions": {
      "rest.action.multi.allow_explicit_index": "true"
    }
  },
  "responseElements": {
    "domainStatus": {
      "created": true,
      "clusterConfig": {
        "zoneAwarenessEnabled": false,
        "instanceType": "m4.large.search",
        "dedicatedMasterEnabled": false,
        "instanceCount": 1
      },
      "cognitoOptions": {
        "enabled": false
      },
      "encryptionAtRestOptions": {
        "enabled": false
      },
      "advancedOptions": {
        "rest.action.multi.allow_explicit_index": "true"
      },
      "upgradeProcessing": false,
      "snapshotOptions": {
        "automatedSnapshotStartHour": 0
      },
      "eBSOptions": {
        "eBSEnabled": true,
        "volumeSize": 10,
        "volumeType": "gp2"
      },
      "engineVersion": "OpenSearch_1.0",
```

```
      "processing": true,
      "aRN": "arn:aws:es:us-west-1:123456789012:domain/test-domain",
      "domainId": "123456789012/test-domain",
      "deleted": false,
      "domainName": "test-domain",
      "accessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:root\"},\"Action\":\"es:*\",
\"Resource\":\"arn:aws:es:us-west-1:123456789012:domain/test-domain/*\"}]}"
    }
  },
  "requestID": "12345678-1234-1234-1234-987654321098",
  "eventID": "87654321-4321-4321-4321-987654321098",
  "eventType": "AwsApiCall",
  "recipientAccountId": "123456789012"
}
```

Sécurité dans Amazon OpenSearch Service

La sécurité du cloud AWS est la priorité absolue. En tant que AWS client, vous bénéficiez d'un centre de données et d'une architecture réseau conçus pour répondre aux exigences des entreprises les plus sensibles en matière de sécurité.

La sécurité est une responsabilité partagée entre vous AWS et vous. Le modèle de responsabilité partagée décrit cette notion par les termes sécurité du cloud et sécurité dans le cloud :

- Sécurité du cloud : AWS est chargée de protéger l'infrastructure qui exécute les AWS services dans le AWS cloud. AWS vous fournit également des services que vous pouvez utiliser en toute sécurité. Des auditeurs tiers testent et vérifient régulièrement l'efficacité de notre sécurité dans le cadre des programmes de conformitéAWS. Pour en savoir plus sur les programmes de conformité qui s'appliquent à Amazon OpenSearch Service, consultez la section AWS Services concernés par programme de conformité.
- Sécurité dans le cloud — Votre responsabilité est déterminée par le AWS service que vous utilisez. Vous êtes également responsable d'autres facteurs, y compris de la sensibilité de vos données, des exigences de votre entreprise, ainsi que de la législation et de la réglementation applicables.

Cette documentation vous aide à comprendre comment appliquer le modèle de responsabilité partagée lors de l'utilisation OpenSearch du Service. Les rubriques suivantes expliquent comment configurer le OpenSearch service pour répondre à vos objectifs de sécurité et de conformité. Vous apprendrez également à utiliser d'autres AWS services qui vous aident à surveiller et à sécuriser les ressources OpenSearch de vos services.

Rubriques

- Protection des données dans Amazon OpenSearch Service
- Identity and Access Management dans Amazon OpenSearch Service
- Prévention du problème de l'adjoint confus entre services
- Contrôle d'accès précis dans Amazon Service OpenSearch
- Validation de conformité pour Amazon OpenSearch Service
- Résilience dans Amazon OpenSearch Service
- Authentification et autorisation JWT pour Amazon Service OpenSearch
- Sécurité de l'infrastructure dans Amazon OpenSearch Service
- Authentification SAML pour les tableaux de bord OpenSearch

- Configuration de l'authentification Amazon Cognito pour les tableaux de bord OpenSearch
- Utilisation des rôles liés à un service pour Amazon Service OpenSearch

Protection des données dans Amazon OpenSearch Service

Le modèle de responsabilité AWS partagée de s'applique à la protection des données dans Amazon OpenSearch Service. Comme décrit dans ce modèle, AWS est chargé de protéger l'infrastructure mondiale qui gère tous les AWS Cloud. La gestion du contrôle de votre contenu hébergé sur cette infrastructure relève de votre responsabilité. Vous êtes également responsable des tâches de configuration et de gestion de la sécurité des Services AWS que vous utilisez. Pour plus d'informations sur la confidentialité des données, consultez Questions fréquentes (FAQ) sur la confidentialité des données. Pour en savoir plus sur la protection des données en Europe, consultez le billet de blog Modèle de responsabilité partagée AWS et RGPD (Règlement général sur la protection des données) sur le Blog de sécuritéAWS .

À des fins de protection des données, nous vous recommandons de protéger les Compte AWS informations d'identification et de configurer les utilisateurs individuels avec AWS IAM Identity Center ou AWS Identity and Access Management (IAM). Ainsi, chaque utilisateur se voit attribuer uniquement les autorisations nécessaires pour exécuter ses tâches. Nous vous recommandons également de sécuriser vos données comme indiqué ci-dessous :

- Utilisez l'authentification multifactorielle (MFA) avec chaque compte.
- SSL/TLS À utiliser pour communiquer avec AWS les ressources. Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Configurez l'API et la journalisation de l'activité des utilisateurs avec AWS CloudTrail. Pour plus d'informations sur l'utilisation des CloudTrail sentiers pour capturer AWS des activités, consultez la section Utilisation des CloudTrail sentiers dans le guide de AWS CloudTrail l'utilisateur.
- Utilisez des solutions de AWS chiffrement, ainsi que tous les contrôles de sécurité par défaut qu'ils contiennent Services AWS.
- Utilisez des services de sécurité gérés avancés tels qu'Amazon Macie, qui contribuent à la découverte et à la sécurisation des données sensibles stockées dans Amazon S3.
- Si vous avez besoin de modules cryptographiques validés par la norme FIPS 140-3 pour accéder AWS via une interface de ligne de commande ou une API, utilisez un point de terminaison FIPS. Pour plus d'informations sur les points de terminaison FIPS disponibles, consultez Norme FIPS (Federal Information Processing Standard) 140-3.

Nous vous recommandons fortement de ne jamais placer d'informations confidentielles ou sensibles, telles que les adresses e-mail de vos clients, dans des balises ou des champs de texte libre tels que le champ Nom. Cela inclut lorsque vous travaillez avec le OpenSearch Service ou un autre Services AWS à l'aide de la console AWS CLI, de l'API ou AWS SDKs. Toutes les données que vous entrez dans des balises ou des champs de texte de forme libre utilisés pour les noms peuvent être utilisées à des fins de facturation ou dans les journaux de diagnostic. Si vous fournissez une adresse URL à un serveur externe, nous vous recommandons fortement de ne pas inclure d'informations d'identification dans l'adresse URL permettant de valider votre demande adressée à ce serveur.

Chiffrement des données au repos pour Amazon OpenSearch Service

OpenSearch Les domaines de service offrent le chiffrement des données au repos, une fonctionnalité de sécurité qui permet d'empêcher tout accès non autorisé à vos données. La fonctionnalité utilise AWS Key Management Service (AWS KMS) pour stocker et gérer vos clés de chiffrement et l'algorithme Advanced Encryption Standard avec des clés de 256 bits (AES-256) pour effectuer le chiffrement. Si cette option est activée, elle chiffre les aspects suivants d'un domaine :

- Tous les index (y compris ceux UltraWarm stockés)
- OpenSearch journaux
- Échangez les fichiers
- Toutes les autres données dans le répertoire de l'application
- Instantanés automatiques

Les services suivants ne sont pas chiffrées lorsque vous activez le chiffrement des données au repos, mais vous pouvez prendre des mesures supplémentaires afin de les protéger :

- Instantanés manuels : vous ne pouvez actuellement pas utiliser de AWS KMS clés pour chiffrer les instantanés manuels. Toutefois, vous pouvez utiliser le chiffrement côté serveur avec des clés gérées par S3 ou des clés KMS pour chiffrer le compartiment que vous utilisez comme référentiel d'instantanés. Pour obtenir des instructions, veuillez consulter the section called "Inscription d'un référentiel d'instantanés manuels".
- Journaux lents et journaux d'erreurs : si vous publiez des journaux et que vous souhaitez les chiffrer, vous pouvez chiffrer leur groupe de CloudWatch journaux à l'aide de la même AWS KMS clé que le domaine de OpenSearch service. Pour plus d'informations, consultez la section Chiffrer les données des CloudWatch journaux dans les journaux à l'aide AWS Key Management Service du guide de l'utilisateur Amazon CloudWatch Logs.

> **Note**
>
> Vous ne pouvez pas activer le chiffrement au repos sur un domaine existant si UltraWarm le stockage à froid est activé sur le domaine. Vous devez d'abord UltraWarm désactiver le stockage à froid, activer le chiffrement au repos, puis réactiver UltraWarm le stockage à froid. Si vous souhaitez conserver les index dans UltraWarm un stockage à froid, vous devez les déplacer vers un stockage à chaud avant de les désactiver UltraWarm ou de les stocker à froid.

OpenSearch Le service prend uniquement en charge les clés KMS de chiffrement symétriques, et non les clés asymétriques. Pour savoir comment créer des clés symétriques, consultez la section Création d'une clé KMS dans le guide du AWS Key Management Service développeur.

Que le chiffrement au repos soit activé ou non, tous les domaines chiffrent automatiquement les packages personnalisés à l'aide de clés AES-256 et OpenSearch de clés gérées par le service.

Autorisations

Pour utiliser la console de OpenSearch service afin de configurer le chiffrement des données au repos, vous devez disposer d'autorisations de lecture AWS KMS, telles que la politique basée sur l'identité suivante :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:List*",
        "kms:Describe*"
      ],
      "Resource": "*"
    }
  ]
}
```

Si vous souhaitez utiliser une clé autre que celle que vous AWS possédez, vous devez également être autorisé à créer des autorisations pour la clé. Ces autorisations se présentent généralement sous la forme d'une politique basée sur les ressources que vous indiquez lorsque vous créez la clé.

Si vous souhaitez conserver votre clé exclusive au OpenSearch Service, vous pouvez ajouter la ViaService condition kms : à cette politique clé :

```
"Condition": {
  "StringEquals": {
    "kms:ViaService": "es.us-west-1.amazonaws.com"
  },
  "Bool": {
    "kms:GrantIsForAWSResource": "true"
  }
}
```

Pour plus d'informations, consultez Politiques de clé dans AWS KMS dans le Guide du développeur AWS Key Management Service .

Activation du chiffrement de données au repos

Le chiffrement des données inactives sur les nouveaux domaines nécessite soit Elasticsearch 5.1, OpenSearch soit une version ultérieure. Son activation sur des domaines existants nécessite Elasticsearch 6.7 ou version ultérieure. OpenSearch

Pour activer le chiffrement des données au repos (console)

1. Ouvrez le domaine dans la AWS console, puis choisissez Actions et Modifier la configuration de sécurité.
2. Dans Encryption (Chiffrement), sélectionnez Enable encryption of data at rest (Activer le chiffrement des données au repos).
3. Choisissez la AWS KMS clé à utiliser, puis cliquez sur Enregistrer les modifications.

Vous pouvez également activer le chiffrement via l'API de configuration. La requête suivante permet le chiffrement des données au repos sur un domaine existant :

```
{
   "ClusterConfig":{
      "EncryptionAtRestOptions":{
```

```
        "Enabled": true,
        "KmsKeyId":"arn:aws:kms:us-east-1:123456789012:alias/my-key"
      }
   }
}
```

Clé KMS désactivée ou supprimée

Si vous désactivez ou supprimez la clé que vous avez utilisée pour chiffrer un domaine, celui-ci devient inaccessible. OpenSearch Le service vous envoie une notification vous informant qu'il ne peut pas accéder à la clé KMS. Réactivez immédiatement la clé pour accéder à votre domaine.

L'équipe du OpenSearch service ne peut pas vous aider à récupérer vos données si votre clé est supprimée. AWS KMS supprime les clés uniquement après une période d'attente d'au moins sept jours. Si votre clé est en attente de suppression, annulez la suppression ou effectuez un instantané manuel du domaine pour éviter toute perte de données.

Désactivation du chiffrement de données au repos

Une fois que vous avez configuré un domaine pour chiffrer les données au repos, vous ne pouvez pas désactiver le paramètre. Au lieu de cela, vous pouvez prendre un instantané manuel du domaine existant, créer un autre domaine, migrer vos données et supprimer l'ancien domaine.

Surveillance des domaines qui chiffrent les données au repos

Les domaines qui chiffrent des données au repos ont deux métriques supplémentaires : `KMSKeyError` et `KMSKeyInaccessible`. Ces métriques s'affichent uniquement si le domaine rencontre un problème avec votre clé de chiffrement. Pour une liste complète de ces métriques, consultez the section called “Métriques du cluster”. Vous pouvez les consulter à l'aide de la console OpenSearch Service ou de la CloudWatch console Amazon.

> **Tip**
>
> Chaque métrique représente un problème important pour un domaine. Nous vous recommandons donc de créer des CloudWatch alarmes pour les deux. Pour de plus amples informations, veuillez consulter the section called “ CloudWatch Alarmes recommandées”.

Autres considérations

- La rotation automatique des touches préserve les propriétés de vos AWS KMS clés, de sorte que la rotation n'a aucun effet sur votre capacité à accéder à vos OpenSearch données. Les domaines OpenSearch de service chiffrés ne prennent pas en charge la rotation manuelle des clés, qui implique la création d'une nouvelle clé et la mise à jour des références à l'ancienne clé. Pour en savoir plus, consultez la section Rotation AWS KMS des touches dans le guide du AWS Key Management Service développeur.
- Certains types d'instance ne prennent pas en charge le chiffrement des données au repos. Pour plus d'informations, consultez the section called “Types d’instance pris en charge”.
- Les domaines qui chiffrent les données au repos utilisent un nom de référentiel différent pour leurs instantanés automatiques. Pour plus d'informations, consultez the section called “Restauration des instantanés”.
- Bien que nous vous recommandions d'activer le chiffrement au repos, celui-ci peut entraîner des charges supplémentaires de processeur et quelques millisecondes de latence. La plupart des cas d'utilisation ne sont toutefois pas sensibles à ces différences, cependant l'ampleur de l'impact dépend de la configuration du cluster, des clients et du profil d'utilisation.

Node-to-node chiffrement pour Amazon OpenSearch Service

Node-to-node le chiffrement fournit un niveau de sécurité supplémentaire en plus des fonctionnalités par défaut d'Amazon OpenSearch Service.

Chaque domaine OpenSearch de service, qu'il utilise ou non un accès VPC, réside dans son propre VPC dédié. Cette architecture empêche les attaquants potentiels d'intercepter le trafic entre les OpenSearch nœuds et assure la sécurité du cluster. Par défaut, toutefois, le trafic au sein du VPC n'est pas chiffré. Node-to-nodele chiffrement permet le chiffrement TLS 1.2 pour toutes les communications au sein du VPC.

Si vous envoyez des données au OpenSearch Service via HTTPS, le node-to-node chiffrement permet de garantir que vos données restent cryptées lorsqu' OpenSearch elles sont distribuées (et redistribuées) dans le cluster. Si les données arrivent non chiffrées via HTTP, le OpenSearch service les chiffre une fois qu'elles ont atteint le cluster. Vous pouvez exiger que tout le trafic vers le domaine arrive via HTTPS à l'aide de la console ou de l'API de configuration. AWS CLI

Node-to-node le chiffrement est nécessaire si vous activez le contrôle d'accès détaillé.

Activation du node-to-node chiffrement

Node-to-node le chiffrement des nouveaux domaines nécessite n'importe quelle version d' OpenSearchElasticsearch 6.0 ou version ultérieure. L'activation du node-to-node chiffrement sur les domaines existants nécessite n'importe quelle version d' OpenSearchElasticsearch 6.7 ou version ultérieure. Choisissez le domaine existant dans la console AWS , Actions, et Edit security configuration (Modifier la configuration de la sécurité).

Vous pouvez également utiliser l'API de configuration AWS CLI or. Pour plus d'informations, consultez la référence des AWS CLI commandes et la référence OpenSearch de l'API de service.

Désactivation du chiffrement node-to-node

Une fois que vous avez configuré un domaine pour utiliser node-to-node le chiffrement, vous ne pouvez pas désactiver ce paramètre. Au lieu de cela, vous pouvez prendre un instantané manuel du domaine chiffré, créer un autre domaine, migrer vos données et supprimer l'ancien domaine.

Identity and Access Management dans Amazon OpenSearch Service

Amazon OpenSearch Service propose plusieurs méthodes pour contrôler l'accès à vos domaines. Cette rubrique présente différents types de stratégies, explique leurs interactions et indique comment créer vos propres stratégies personnalisées.

> **Important**
>
> Le support VPC introduit des considérations supplémentaires en matière de contrôle d'accès aux OpenSearch services. Pour de plus amples informations, veuillez consulter the section called “À propos des stratégies d'accès pour les domaines de VPC”.

Types de stratégies

OpenSearch Le service prend en charge trois types de politiques d'accès :

- the section called “Stratégies basées sur les ressources”
- the section called “Politiques basées sur l’identité”
- the section called “Stratégies basées sur l'IP”

Stratégies basées sur les ressources

Vous ajoutez une stratégie basée sur les ressources, souvent appelée stratégie d'accès au domaine, lorsque vous créez un domaine. Ces politiques spécifient quelles actions un principal peut effectuer sur les sous-ressources du domaine (à l'exception de la recherche entre clusters). Les sous-ressources incluent les OpenSearch index et. APIs L'élément de politique JSON principal dans IAM spécifie les comptes, les utilisateurs ou les rôles auxquels l'accès est autorisé. L'élément de politique Resource JSON indique à quelles sous-ressources ces principaux peuvent accéder.

Par exemple, la politique suivante basée sur les ressources accorde à `test-user` l'accès total (`es:*`) aux sous-ressources de `test-domain` :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:user/test-user"
        ]
      },
      "Action": [
        "es:*"
      ],
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/*"
    }
  ]
}
```

Deux considérations importantes s'appliquent à cette stratégie :

- Ces privilèges s'appliquent uniquement à ce domaine. Sauf si vous créez des stratégies similaires sur d'autres domaines,`test-user` peut uniquement accéder à `test-domain`.
- La barre oblique `/*` de l'élément `Resource` indique que les stratégies basées sur les ressources s'appliquent uniquement aux sous-ressources du domaine, et non au domaine lui-même. Dans les stratégies basées sur les ressources, l'action `es:*` équivaut à `es:ESHttp*`.

Par exemple, `test-user` peut envoyer des demandes concernant un index (`GET https://search-test-domain.us-west-1.es.amazonaws.com/test-index`), mais ne peut pas mettre à jour la configuration du domaine (`POST https://es.us-west-1.amazonaws.com/2021-01-01/opensearch/domain/test-domain/config`). Notez la différence entre les deux points de terminaison. L'accès à l'API de configuration nécessite une politique basée sur l'identité.

Vous pouvez spécifier un nom d'index partiel en ajoutant un caractère générique. Cet exemple identifie tous les index commençant par `commerce` :

```
arn:aws:es:us-west-1:987654321098:domain/test-domain/commerce*
```

Dans ce cas, le caractère générique signifie que `test-user` peut envoyer des requêtes aux index de `test-domain` dont le nom commence par `commerce`.

Pour restreindre davantage `test-user`, vous pouvez appliquer la stratégie suivante :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:user/test-user"
        ]
      },
      "Action": [
        "es:ESHttpGet"
      ],
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/commerce-data/_search"
    }
  ]
}
```

À présent, `test-user` ne peut effectuer qu'une seule opération : rechercher sur l'index `commerce-data`. Tous les autres index au sein du domaine sont inaccessibles et, sans autorisation d'utiliser les actions `es:ESHttpPut` ou `es:ESHttpPost`, `test-user` ne peut pas ajouter ou modifier des documents.

Ensuite, vous pouvez décider de configurer un rôle pour les utilisateurs avancés. Cette politique donne `power-user-role` accès aux méthodes HTTP GET et PUT pour tous les éléments URIs de l'index :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:role/power-user-role"
        ]
      },
      "Action": [
        "es:ESHttpGet",
        "es:ESHttpPut"
      ],
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/commerce-data/*"
    }
  ]
}
```

Si votre domaine se trouve dans un VPC ou utilise un contrôle précis des accès, vous pouvez utiliser une stratégie d'accès ouverte au domaine. Sinon, votre stratégie d'accès au domaine doit contenir certaines restrictions, soit par le principal, soit par l'adresse IP.

Pour plus d'informations sur les différentes actions disponibles, consultez the section called “Références des éléments de stratégie”. Pour un contrôle nettement plus précis de vos données, utilisez une stratégie d'accès ouverte au domaine avec contrôle précis des accès.

Politiques basées sur l’identité

Contrairement aux politiques basées sur les ressources, qui font partie de chaque domaine de OpenSearch service, vous associez des politiques basées sur l'identité aux utilisateurs ou aux rôles à l'aide du service AWS Identity and Access Management (IAM). Tout comme les stratégies basées sur une ressource, celles basées sur une identité déterminent qui est autorisé à accéder à un service, quelles actions peuvent être exécutées et, le cas échéant, les ressources concernées.

Bien que ce ne soit certainement pas nécessaire, les stratégies basées sur une identité ont tendance à être plus génériques. Bien souvent, elles ne régissent que les actions de l'API de configuration qu'un utilisateur peut effectuer. Une fois ces politiques en place, vous pouvez utiliser des politiques basées sur les ressources (ou un contrôle d'accès précis) dans OpenSearch Service pour permettre aux utilisateurs d'accéder aux index et. OpenSearch APIs

> **Note**
>
> Les utilisateurs dotés de la `AmazonOpenSearchServiceReadOnlyAccess` politique AWS gérée ne peuvent pas voir l'état de santé du cluster sur la console. Pour leur permettre de consulter l'état de santé du cluster (et d'autres OpenSearch données), ajoutez l'`es:ESHttpGet`action à une politique d'accès et associez-la à leurs comptes ou rôles.

Étant donné que les stratégies basées sur l'identité sont attachées à des utilisateurs ou des rôles (principaux), le JSON ne spécifie pas de principal. La stratégie suivante accorde l'accès à des actions commençant par `Describe` et `List`. Cette combinaison d'actions fournit un accès en lecture seule aux configurations de domaine, mais pas aux données stockées dans le domaine lui-même :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "es:Describe*",
        "es:List*"
      ],
      "Effect": "Allow",
      "Resource": "*"
```

```
    }
  ]
}
```

Un administrateur peut avoir un accès complet au OpenSearch service et à toutes les données stockées sur tous les domaines :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "es:*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Les stratégies basées sur l'identité vous permettent d'utiliser des balises pour contrôler l'accès à l'API de configuration. La stratégie suivante, par exemple, permet aux principaux attachés d'afficher et de mettre à jour la configuration d'un domaine si ce domaine dispose de la balise `team:devops` :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Action": [
      "es:UpdateDomainConfig",
      "es:DescribeDomain",
      "es:DescribeDomainConfig"
    ],
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
```

```
      "ForAnyValue:StringEquals": {
        "aws:ResourceTag/team": [
          "devops"
        ]
      }
    }
  }]
}
```

Vous pouvez également utiliser des balises pour contrôler l'accès à l' OpenSearch API. Les politiques basées sur des balises pour l' OpenSearch API ne s'appliquent qu'aux méthodes HTTP. Par exemple, la politique suivante permet aux entités associées d'envoyer des requêtes GET et PUT à l' OpenSearch API si le domaine possède la `environment:production` balise :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Action": [
      "es:ESHttpGet",
      "es:ESHttpPut"
    ],
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
      "ForAnyValue:StringEquals": {
        "aws:ResourceTag/environment": [
          "production"
        ]
      }
    }
  }]
}
```

Pour un contrôle plus précis de l' OpenSearch API, pensez à utiliser un contrôle d'accès précis.

> **Note**
>
> Après avoir ajouté une ou plusieurs balises OpenSearch APIs à une politique basée sur des balises, vous devez effectuer une seule opération de balise (telle que l'ajout, la suppression ou la modification d'une balise) pour que les modifications prennent effet sur un domaine. Vous devez utiliser le logiciel de service R20211203 ou version ultérieure pour inclure les opérations d' OpenSearch API dans les politiques basées sur des balises.

OpenSearch Le service prend en charge les clés de condition `TagKeys` globales `RequestTag` et les clés de condition pour l'API de configuration, et non pour l' OpenSearch API. Ces conditions s'appliquent uniquement aux appels d'API incluant des balises dans la demande, comme `CreateDomain`, `AddTags` et `RemoveTags`. La stratégie suivante permet aux principaux attachés de créer des domaines, mais uniquement s'ils disposent de la balise `team:it` dans la requête :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "es:CreateDomain",
      "es:AddTags"
    ],
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "aws:RequestTag/team": [
          "it"
        ]
      }
    }
  }
}
```

Pour plus d'informations sur l'utilisation de balises pour le contrôle d'accès et sur les différences entre les politiques basées sur les ressources et les politiques basées sur l'identité, consultez la section

Définir les autorisations en fonction des attributs avec autorisation ABAC dans le guide de l'utilisateur IAM.

Stratégies basées sur l'IP

Les stratégies basées sur l'IP limitent l'accès à un domaine à une ou plusieurs adresses IP ou à des blocs CIDR spécifiques. Techniquement, les stratégies basées sur l'adresse IP ne sont pas un type de stratégie distincte. Il s'agit plutôt de politiques basées sur les ressources qui spécifient un principal anonyme et incluent une condition spéciale. Pour plus d'informations, voir Éléments de politique IAM JSON : Condition dans le guide de l'utilisateur IAM.

Le principal avantage des politiques basées sur l'IP est qu'elles autorisent les requêtes non signées adressées à un domaine de OpenSearch service, ce qui vous permet d'utiliser des clients tels que curl et OpenSearch Dashboards ou d'accéder au domaine via un serveur proxy. Pour en savoir plus, veuillez consulter la section the section called “Utilisation d'un proxy pour accéder au OpenSearch service à partir de tableaux de bord”.

> **Note**
>
> Si vous avez activé un accès VPC pour votre domaine, vous ne pouvez pas configurer une stratégie basée sur l'IP. Vous pouvez plutôt l'utiliser `security groups` pour contrôler les adresses IP autorisées à accéder au domaine. Pour plus d’informations, consultez les rubriques suivantes :
>
> - the section called “À propos des stratégies d'accès pour les domaines de VPC”
> - Contrôlez le trafic vers vos AWS ressources à l'aide des groupes de sécurité décrits dans le guide de l'utilisateur Amazon VPC

La stratégie suivante permet à toutes les demandes en provenance de la plage d'adresses IP spécifiée d'accéder à `test-domain`:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

```
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Condition": {
        "IpAddress": {
          "aws:SourceIp": [
            "192.0.2.0/24"
          ]
        }
      },
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/*"
    }
  ]
}
```

Si votre domaine dispose d'un point de terminaison public et n'utilise pas le contrôle d'accès affiné, nous vous recommandons de combiner les entités IAM et les adresses IP. Cette stratégie n'accorde à `test-user` l'accès HTTP que si la demande provient de la plage IP spécifiée :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "AWS": [
        "arn:aws:iam::987654321098:user/test-user"
      ]
    },
    "Action": [
      "es:ESHttp*"
    ],
    "Condition": {
      "IpAddress": {
        "aws:SourceIp": [
          "192.0.2.0/24"
        ]
```

```
      }
    },
    "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/*"
  }]
}
```

Formulation et signature de demandes OpenSearch de service

Même si vous configurez une politique d'accès entièrement ouverte basée sur les ressources, toutes les demandes adressées à l'API de configuration du OpenSearch service doivent être signées. Si vos politiques spécifient des rôles ou des utilisateurs IAM, les demandes adressées doivent OpenSearch APIs également être signées à l'aide de AWS Signature Version 4. La méthode de signature varie en fonction de l'API :

- Pour appeler l'API de configuration du OpenSearch service, nous vous recommandons d'utiliser l'un des AWS SDKs. Cela simplifie SDKs considérablement le processus et peut vous faire gagner beaucoup de temps par rapport à la création et à la signature de vos propres demandes. Les points de terminaison de l'API de configuration utilisent le format suivant :

  ```
  es.region.amazonaws.com/2021-01-01/
  ```

  Par exemple, la demande suivante apporte une modification de configuration au domaine `movies`, mais vous devez la signer vous-même (non recommandé) :

  ```
  POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/movies/config
  {
    "ClusterConfig": {
      "InstanceType": "c5.xlarge.search"
    }
  }
  ```

  Si vous utilisez l'un d'entre eux SDKs, comme Boto 3, le SDK gère automatiquement la signature des demandes :

  ```
  import boto3

  client = boto3.client(es)
  response = client.update_domain_config(
    DomainName='movies',
  ```

```
  ClusterConfig={
    'InstanceType': 'c5.xlarge.search'
  }
)
```

Pour obtenir un exemple de code Java, consultez the section called “À l'aide du AWS SDKs”.

- Pour passer des appels au OpenSearch APIs, vous devez signer vos propres demandes. OpenSearch APIs Utilisez le format suivant :

```
domain-id.region.es.amazonaws.com
```

Par exemple, la demande suivante recherche l'index `movies` pour thor :

```
GET https://my-domain.us-east-1.es.amazonaws.com/movies/_search?q=thor
```

> **Note**
>
> Le service ignore les paramètres transmis URLs pour les requêtes HTTP POST signées avec Signature Version 4.

En cas de conflit entre plusieurs stratégies

Si les stratégies sont en désaccord ou ne mentionnent explicitement un utilisateur, cela crée des situations complexes. Comment fonctionne l'IAM dans le guide de l'utilisateur de l'IAM fournit un résumé concis de la logique d'évaluation des politiques :

- Par défaut, toutes les demandes sont refusées.
- Une autorisation explicite remplace ce fonctionnement par défaut.
- Un refus explicite remplace toute autorisation.

Par exemple, si une politique basée sur les ressources vous accorde l'accès à une sous-ressource de domaine (un OpenSearch index ou une API), mais qu'une politique basée sur l'identité vous en refuse l'accès, l'accès vous est refusé. Si une stratégie basée sur une identité accorde l'accès et celle basée sur une ressource ne spécifie rien concernant votre accès, vous êtes autorisé à accéder. Pour

un récapitulatif complet des issues possibles en matière de sous-ressources de domaine, consultez le tableau suivant des recoupements entre stratégies.

	Autorisé dans la stratégie basée sur une ressource	Refusé dans la stratégie basée sur une ressource	Ni autorisé ni refusé dans la stratégie basée sur une ressource
Allowed in identity-based policy	Autorisation	Refuser	Autorisation
Denied in identity-based policy	Refuser	Refuser	Refuser
Neither allowed nor denied in identity-based policy	Autorisation	Refuser	Refuser

Références des éléments de stratégie

OpenSearch Le service prend en charge la plupart des éléments de politique de la référence des éléments de stratégie IAM, à l'exception de`NotPrincipal`. Le tableau suivant indique les éléments les plus courants.

Élément de stratégie JSON	Récapitulatif
`Version`	La version actuelle du langage de la stratégie est `2012-10-17` . Toutes les stratégies d'accès doivent spécifier cette valeur.
`Effect`	Cet élément spécifie si la déclaration autorise ou refuse l'accès aux actions spécifiées. Les valeurs valides sont `Allow` ou `Deny`.
`Principal`	Cet élément indique le rôle Compte AWS ou l'utilisateur IAM autorisé ou refusé à une ressource et peut prendre plusieurs formes :

Élément de stratégie JSON	Récapitulatif
	• AWS comptes : `"Principal":{"AWS": ["123456789012"]}` ou `"Principal":{"AWS": ["arn:aws:iam::123456789012:root"]}` • Utilisateurs IAM : `"Principal":{"AWS": ["arn:aws:iam::123456789012:user/test-user"]}` • Rôles IAM : `"Principal":{"AWS": ["arn:aws:iam::123456789012:role/test-role"]}`

> **Important**
>
> L'indication du caractère générique * permet l'accès anonyme au domaine, ce que nous ne recommandons pas, sauf si vous ajoutez une condition basée sur IP, si vous utilisez la prise en charge du VPC ou si vous activez un contrôle d'accès affiné. En outre, examinez attentivement les politiques suivantes pour vous assurer qu'elles n'accordent pas un accès étendu :
>
> - Politiques basées sur l'identité associées aux principaux AWS correspondants (par exemple, rôles IAM)
> - Politiques basées sur les ressources associées aux AWS ressources associées (par exemple, clés AWS Key Management Service KMS)

Élément de stratégie JSON	Récapitulatif
`Action`	OpenSearch Le service utilise `ESHttp*` des actions pour les méthodes OpenSearch HTTP. Les autres actions s'appliquent à l'API de configuration. Certaines actions `es:` acceptent des autorisations au niveau des ressources. Par exemple, vous pouvez accorder à un utilisateur l'autorisation de supprimer un domaine particulier sans l'autoriser à supprimer n'importe quel domaine. D'autres actions s'appliquent uniquement au service lui-même. Par exemple, `es:ListDomainNames` n'a aucun sens dans le contexte d'un domaine unique et, par conséquent, nécessite un caractère générique. Pour obtenir la liste de toutes les actions disponibles et savoir si elles s'appliquent aux sous-ressources du domaine (`test-domain/*`), à la configuration du domaine (`test-domain`) ou uniquement au service (`*`), consultez la section Actions, ressources et clés de condition pour Amazon OpenSearch Service dans le Service Authorization Reference Les politiques basées sur les ressources diffèrent des autorisations de niveau ressource. Les stratégies basées sur une ressource sont des stratégies JSON complètes attachées à des domaines. Les autorisations au niveau des ressources vous permettent de limiter les actions à des domaines ou sous-ressources spécifiques. En pratique, vous pouvez envisager les autorisations au niveau des ressources comme une partie facultative d'une stratégie basée sur une ressource ou une identité. Bien que les autorisations au niveau des ressources pour `es:CreateDomain` peuvent sembler peu intuitives (pourquoi accorder à un utilisateur l'autorisation de créer un domaine qui existe déjà ?), l'utilisation d'un caractère générique vous permet d'appliquer une méthode simple de dénomination pour vos domaines telle que `"Resource": "arn:aws:es:us-west-1:987654321098:domain/my-team-name-*"` .

Élément de stratégie JSON	Récapitulatif
	Bien entendu, rien ne vous empêche d'inclure des actions aux côtés d'éléments de ressources moins restrictifs, comme dans l'exemple suivant : JSON `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "es:ESHttpGet", "es:DescribeDomain" ], "Resource": "*" } ] }` Pour en savoir plus sur l'appairage d'actions et de ressources, référez-vous à l'élément `Resource` dans ce tableau.

Élément de stratégie JSON	Récapitulatif
`Condition`	OpenSearch Le service prend en charge la plupart des conditions décrites dans les clés contextuelles des conditions AWS globales du guide de l'utilisateur IAM. Les exceptions notables incluent la `aws:PrincipalTag` clé, que le OpenSearch Service ne prend pas en charge. Lorsque vous configurez une stratégie basée sur l'IP, vous spécifiez les adresses IP ou blocs d'adresse CIDR en tant que condition comme suit : `"Condition": { "IpAddress": { "aws:SourceIp": [ "192.0.2.0/32" ] } }` Comme indiqué dansthe section called “Politiques basées sur l’identité”, les clés `aws:ResourceTag` `aws:RequestTag` , et de `aws:TagKeys` condition s'appliquent à l'API de configuration ainsi qu'à OpenSearch APIs.

Élément de stratégie JSON	Récapitulatif

`Resource`	OpenSearch Le service utilise `Resource` les éléments de trois manières fondamentales :
	
	• Pour les actions qui s'appliquent au OpenSearch Service lui-mêmees:ListDomainNames , comme ou pour autoriser un accès complet, utilisez la syntaxe suivante :
	
	`"Resource": "*"`
	
	• Pour les actions qui impliquent la configuration d'un domaine, comme `es:DescribeDomain` , vous pouvez utiliser la syntaxe suivante :
	
	`"Resource": "arn:aws:es:` *region*`:`*aws-account-id*`:domain/`*domain-name*`"`
	
	• Pour les actions concernant les sous-ressources d'un domaine, comme `es:ESHttpGet` , vous pouvez utiliser la syntaxe suivante :
	
	`"Resource": "arn:aws:es:` *region*`:`*aws-account-id*`:domain/`*domain-name*`/*"`
	
	Vous n'êtes pas obligé d'utiliser un joker. OpenSearch Le service vous permet de définir une politique d'accès différente pour chaque OpenSearch index ou API. Par exemple, vous pouvez limiter les autorisations d'un utilisateur à l'index `test-index` :
	
	`"Resource": "arn:aws:es:` *region*`:`*aws-account-id*`:domain/`*domain-name*`/test-index"`
	
	Au lieu d'un accès total à `test-index` , vous préférerez peut-être limiter la stratégie à l'API de recherche :
	
	`"Resource": "arn:aws:es:` *region*`:`*aws-account-id*`:domain/`*domain-name*`/test-index/_search"`

Élément de stratégie JSON	Récapitulatif
	Vous pouvez même contrôler l'accès à chaque document : `"Resource": "arn:aws:es:  region:aws-account-id:domain/domain-name /test-index/test-type/1"` Essentiellement, s'il OpenSearch exprime la sous-ressource sous forme d'URI, vous pouvez contrôler l'accès à celle-ci à l'aide d'une politique d'accès. Pour plus de contrôle sur les ressources auxquelle s un utilisateur peut accéder, veuillez consulter the section called “Contrôle précis des accès”. Pour plus d'informations sur les actions prenant en charge les autorisat ions au niveau des ressources, référez-vous à l'élément `Action` dans ce tableau.

Options avancées et considérations relatives aux API

OpenSearch Le service comporte plusieurs options avancées, dont l'une a des implications en matière de contrôle d'accès :`rest.action.multi.allow_explicit_index`. Avec sa configuration par défaut sur true (vrai), elle permet aux utilisateurs de contourner les autorisations au niveau des sous-ressources dans certaines circonstances.

Prenons l'exemple suivant de stratégie basée sur une ressource :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:user/test-user"
```

```
        ]
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": [
        "arn:aws:es:us-west-1:987654321098:domain/test-domain/test-index/*",
        "arn:aws:es:us-west-1:987654321098:domain/test-domain/_bulk"
      ]
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:user/test-user"
        ]
      },
      "Action": [
        "es:ESHttpGet"
      ],
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/
restricted-index/*"
    }
  ]
}
```

Cette politique accorde `test-user` un accès complet à l'API OpenSearch en bloc `test-index` et à celle-ci. Elle autorise également les demandes `GET` sur `restricted-index`.

L'exemple suivant de demande d'indexation échoue, comme vous pouviez vous y attendre, en raison d'une erreur d'autorisation :

```
PUT https://search-test-domain.us-west-1.es.amazonaws.com/restricted-index/movie/1
{
  "title": "Your Name",
  "director": "Makoto Shinkai",
  "year": "2016"
}
```

Contrairement à l'API index, l'API bulk vous permet de créer, mettre à jour et supprimer un grand nombre de documents en un seul appel. Toutefois, ces opérations sont généralement définies dans

le corps de la demande, plutôt que dans l'URL de la demande. Étant donné que le OpenSearch URLs service contrôle l'accès aux sous-ressources du domaine, il `test-user` peut en fait utiliser l'API en bloc pour apporter des modifications à`restricted-index`. Même si l'utilisateur n'a pas les autorisations POST pour l'index, la demande suivante aboutit :

```
POST https://search-test-domain.us-west-1.es.amazonaws.com/_bulk
{ "index" : { "_index": "restricted-index", "_type" : "movie", "_id" : "1" } }
{ "title": "Your Name", "director": "Makoto Shinkai", "year": "2016" }
```

Dans ce cas, la stratégie d'accès ne parvient pas à remplir sa fonction. Pour empêcher les utilisateurs de passer outre ce type de restrictions, vous pouvez remplacer la valeur de `rest.action.multi.allow_explicit_index` par false (faux). Si cette valeur est fausse, tous les appels aux commandes bulk, mget et msearch APIs qui spécifient les noms d'index dans le corps de la requête cessent de fonctionner. En d'autres termes, les appels `_bulk` ne fonctionnent plus, mais les appels `test-index/_bulk`, oui. Ce deuxième point de terminaison contient un nom d'index, de sorte que vous n'avez pas besoin d'en spécifier un dans le corps de la demande.

OpenSearch Les tableaux de bord reposent largement sur mget et msearch, il est donc peu probable qu'ils fonctionnent correctement après cette modification. Pour une correction partielle, vous pouvez laisser `rest.action.multi.allow_explicit_index` la valeur true et refuser à certains utilisateurs l'accès à une ou plusieurs d'entre elles APIs.

Pour plus d'informations sur la modification de ce paramètre, consultez the section called “Paramètres avancés du cluster”.

De même, la stratégie basée sur une ressource ci-après engendre deux problèmes subtils :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:user/test-user"
      },
      "Action": "es:ESHttp*",
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/*"
```

```
    },
    {
      "Effect": "Deny",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:user/test-user"
      },
      "Action": "es:ESHttp*",
      "Resource": "arn:aws:es:us-west-1:987654321098:domain/test-domain/
restricted-index/*"
    }
  ]
}
```

- Malgré le refus explicite, `test-user` peut continuer à effectuer des appels tels que `GET https://search-test-domain.us-west-1.es.amazonaws.com/_all/_search` et `GET https://search-test-domain.us-west-1.es.amazonaws.com/*/_search` pour accéder aux documents dans `restricted-index`.
- L'élément `Resource` référence `restricted-index/*`, si bien que `test-user` n'est pas autorisé à accéder directement aux documents de l'index. Toutefois, l'utilisateur a les autorisations requises pour supprimer l'ensemble de l'index. Pour empêcher l'accès et la suppression, la stratégie doit spécifier `restricted-index*`.

Plutôt que de combiner de vastes autorisations avec des refus ciblés, l'approche la plus sûre consiste à appliquer le principe du moindre privilège et à accorder uniquement les autorisations qui sont requises pour exécuter une tâche. Pour plus d'informations sur le contrôle de l'accès à des index ou à des OpenSearch opérations individuels, consultezthe section called “Contrôle précis des accès”.

> **Important**
>
> La spécification du caractère générique* permet un accès anonyme à votre domaine. Il n'est pas recommandé d'utiliser le caractère générique. En outre, examinez attentivement les politiques suivantes pour vous assurer qu'elles n'accordent pas un accès étendu :
>
> - Politiques basées sur l'identité associées aux AWS principaux associés (par exemple, rôles IAM)
> - Politiques basées sur les ressources associées aux AWS ressources associées (par exemple, clés AWS Key Management Service KMS)

Configuration des politiques d'accès

- Pour obtenir des instructions sur la création ou la modification de politiques basées sur les ressources et les adresses IP dans OpenSearch Service, consultezthe section called “Configuration des politiques d'accès”.
- Pour obtenir des instructions sur la création ou la modification de politiques basées sur l'identité dans IAM, voir Définir des autorisations IAM personnalisées avec des politiques gérées par le client dans le guide de l'utilisateur IAM.

Exemples de stratégies supplémentaires

Bien que ce chapitre contienne de nombreux exemples de politiques, le contrôle d' AWS accès est un sujet complexe qu'il est préférable de comprendre à l'aide d'exemples. Pour plus d'informations, consultez Exemples de politiques basées sur l'identité IAM dans le Guide de l'utilisateur IAM.

Référence des autorisations OpenSearch de l'API Amazon Service

Lorsque vous configurez le contrôle d'accès, vous rédigez des politiques d'autorisation que vous pouvez associer à une identité IAM (politiques basées sur l'identité). Pour des informations de référence détaillées, consultez les rubriques suivantes dans la Référence de l’autorisation de service :

- Actions, ressources et clés de condition pour le OpenSearch service.
- Actions, ressources et clés de condition pour OpenSearch l'ingestion.

Cette référence contient des informations sur les opérations d'API qui peuvent être utilisées dans une politique IAM. Il inclut également la AWS ressource pour laquelle vous pouvez accorder les autorisations, ainsi que les clés de condition que vous pouvez inclure pour un contrôle d'accès précis.

Vous spécifiez les actions dans le champ `Action` de la politique, la valeur de ressource dans le champ `Resource` de la politique, et les conditions dans le champ `Condition` de la politique. Pour spécifier une action pour OpenSearch Service, utilisez le `es:` préfixe suivi du nom de l'opération d'API (par exemple,`es:CreateDomain`). Pour spécifier une action pour OpenSearch Ingestion, utilisez le `osis:` préfixe suivi de l'opération d'API (par exemple,`osis:CreatePipeline`).

AWS politiques gérées pour Amazon OpenSearch Service

Une politique AWS gérée est une politique autonome créée et administrée par AWS. AWS les politiques gérées sont conçues pour fournir des autorisations pour de nombreux cas d'utilisation courants afin que vous puissiez commencer à attribuer des autorisations aux utilisateurs, aux groupes et aux rôles.

N'oubliez pas que les politiques AWS gérées peuvent ne pas accorder d'autorisations de moindre privilège pour vos cas d'utilisation spécifiques, car elles sont accessibles à tous les AWS clients. Nous vous recommandons de réduire encore les autorisations en définissant des politiques gérées par le client qui sont propres à vos cas d'utilisation.

Vous ne pouvez pas modifier les autorisations définies dans les politiques AWS gérées. Si les autorisations définies dans une politique AWS gérée sont AWS mises à jour, la mise à jour affecte toutes les identités principales (utilisateurs, groupes et rôles) auxquelles la politique est attachée. AWS est le plus susceptible de mettre à jour une politique AWS gérée lorsqu'une nouvelle politique Service AWS est lancée ou lorsque de nouvelles opérations d'API sont disponibles pour les services existants.

Pour plus d'informations, consultez Politiques gérées par AWS dans le Guide de l'utilisateur IAM.

AmazonOpenSearchDirectQueryGlueCreateAccess

Accorde à Amazon OpenSearch Service Direct Query Service l'accès aux `CreateDatabase`,`CreatePartition`,`CreateTable`, et `BatchCreatePartition` AWS Glue API.

Vous pouvez trouver la AmazonOpenSearchDirectQueryGlueCreateAccesspolitique dans la console IAM.

AmazonOpenSearchServiceFullAccess

Accorde un accès complet aux opérations et aux ressources de l'API de configuration du OpenSearch service pour un Compte AWS.

Vous pouvez trouver la AmazonOpenSearchServiceFullAccesspolitique dans la console IAM.

AmazonOpenSearchServiceReadOnlyAccess

Accorde un accès en lecture seule à toutes les ressources OpenSearch du service pour un. Compte AWS

Vous pouvez trouver la AmazonOpenSearchServiceReadOnlyAccesspolitique dans la console IAM.

AmazonOpenSearchServiceRolePolicy

Vous ne pouvez pas joindre de `AmazonOpenSearchServiceRolePolicy` à vos entités IAM. Cette politique est associée à un rôle lié au service qui permet au OpenSearch service d'accéder aux ressources du compte. Pour de plus amples informations, veuillez consulter the section called “Autorisations”.

Vous pouvez trouver la AmazonOpenSearchServiceRolePolicypolitique dans la console IAM.

AmazonOpenSearchServiceCognitoAccess

Fournit les autorisations minimales Amazon Cognito nécessaires pour activer l'authentification Cognito.

Vous pouvez trouver la AmazonOpenSearchServiceCognitoAccesspolitique dans la console IAM.

AmazonOpenSearchIngestionServiceRolePolicy

Vous ne pouvez pas joindre de `AmazonOpenSearchIngestionServiceRolePolicy` à vos entités IAM. Cette politique est associée à un rôle lié à un service qui permet à OpenSearch Ingestion d'activer l'accès VPC pour les pipelines d'ingestion, de créer des balises et de publier des statistiques relatives à l'ingestion sur votre compte CloudWatch . Pour de plus amples informations, veuillez consulter the section called “Utilisation des rôles liés à un service”.

Vous pouvez trouver la AmazonOpenSearchIngestionServiceRolePolicypolitique dans la console IAM.

OpenSearchIngestionSelfManagedVpcePolicy

Vous ne pouvez pas joindre de `OpenSearchIngestionSelfManagedVpcePolicy` à vos entités IAM. Cette politique est associée à un rôle lié à un service qui permet à OpenSearch Ingestion d'activer un accès VPC autogéré pour les pipelines d'ingestion, de créer des balises et de publier des statistiques relatives à l' CloudWatch ingestion sur votre compte. Pour de plus amples informations, veuillez consulter the section called “Utilisation des rôles liés à un service”.

Vous pouvez trouver la OpenSearchIngestionSelfManagedVpcePolicypolitique dans la console IAM.

AmazonOpenSearchIngestionFullAccess

Accorde un accès complet aux opérations et aux ressources de OpenSearch l'API d'ingestion pour un Compte AWS.

Vous pouvez trouver la AmazonOpenSearchIngestionFullAccesspolitique dans la console IAM.

AmazonOpenSearchIngestionReadOnlyAccess

Accorde un accès en lecture seule à toutes les ressources OpenSearch d'ingestion pour un. Compte AWS

Vous pouvez trouver la AmazonOpenSearchIngestionReadOnlyAccesspolitique dans la console IAM.

AmazonOpenSearchServerlessServiceRolePolicy

Fournit les Amazon CloudWatch autorisations minimales nécessaires pour envoyer des données métriques OpenSearch sans serveur à CloudWatch.

Vous pouvez trouver la AmazonOpenSearchServerlessServiceRolePolicypolitique dans la console IAM.

OpenSearch Mises à jour des services relatifs aux politiques AWS gérées

Consultez les détails des mises à jour des politiques AWS gérées pour le OpenSearch service depuis que ce service a commencé à suivre les modifications.

Modification	Description	Date
A mis à jour le `AmazonOpenSearchServiceRolePolicy`	La déclaration suivante a été ajoutée à la politique. Lorsqu'Amazon OpenSearch Service assume le rôle `AWSServiceRoleForAmazonOpenSearchService` lié au service, cette nouvelle déclaration de la politique permet de mettre OpenSearch à jour l'étendue d'accès de toute AWS IAM	31 mars 2025

Modification	Description	Date
	Identity Center application uniquement gérée par. OpenSearch `{ "Effect": "Allow", "Action": "sso:PutApplicationAccessScope", "Resource": "arn:aws:sso::*:application/*/*", "Condition": { "StringEquals": { "aws:ResourceOrgID": "${aws:PrincipalOrgID}" } } }`	
Mise à jour d'`AmazonOpenSearchServerlessServiceRolePolicy`	Le Sid a été ajouté `AllowAOSSCloudwatchMetrics` à la politique `AmazonOpenSearchServerlessServiceRolePolicy` . Un Sid est un identifiant de déclaration qui sert d'identifiant facultatif pour la déclaration de politique.	12 juillet 2024

Modification	Description	Date
Ajout de `OpenSearchIngestionSelfManagedVpcePolicy` .	Une nouvelle politique qui permet à OpenSearch Ingestion d'activer un accès VPC autogéré aux pipelines d'ingestion, de créer des balises et de publier des statistiques relatives à l' CloudWatch ingestion sur votre compte. Pour connaître le JSON de la politique, consultez Console IAM.	12 juin 2024
Ajouté `AmazonOpenSearchDirectQueryGlueCreateAccess`	Accorde à Amazon OpenSearch Service Direct Query Service l'accès aux `CreateDatabase` `CreatePartition` ,`CreateTable` , et `BatchCreatePartition` AWS Glue API.	6 mai 2024
Mises à jour : `AmazonOpenSearchServiceRolePolicy` et `AmazonElasticsearchServiceRolePolicy`	Ajout des autorisations nécessaires pour que le rôle lié au service puisse attribuer et annuler IPv6 l'attribution d'adresses. La politique Elasticsearch obsolète a également été mise à jour pour assurer une compatibilité descendante.	18 octobre 2023

Modification	Description	Date
Ajout de `AmazonOpenSearchIngestionServiceRolePolicy` .	Une nouvelle politique qui permet à OpenSearch Ingestion d'autoriser l'accès VPC aux pipelines d'ingestion, de créer des balises et de publier des statistiques relatives à l'ingestion sur votre compte CloudWatch . Pour connaître le JSON de la politique, consultez Console IAM.	26 avril 2023
Ajout de `AmazonOpenSearchIngestionFullAccess` .	Une nouvelle politique qui accorde un accès complet aux opérations et aux ressources de OpenSearch l'API d'ingestion pour un Compte AWS. Pour connaître le JSON de la politique, consultez Console IAM.	26 avril 2023
Ajout de `AmazonOpenSearchIngestionReadOnlyAccess` .	Une nouvelle politique qui accorde un accès en lecture seule à toutes les ressources d' OpenSearch ingestion pour un. Compte AWS Pour connaître le JSON de la politique, consultez Console IAM.	26 avril 2023

Modification	Description	Date
Ajout de `AmazonOpenSearchServerlessServiceRolePolicy` .	Une nouvelle politique qui fournit les autorisations minimales nécessaires pour envoyer des données métriques OpenSearch sans serveur à Amazon CloudWatch. Pour connaître le JSON de la politique, consultez Console IAM.	29 novembre 2022
Mises à jour : `AmazonOpenSearchServiceRolePolicy` et `AmazonElasticsearchServiceRolePolicy`	Ajout des autorisations nécessaires pour que le rôle lié au service crée des points de terminaison VPC OpenSearch gérés par le service. Certaines actions ne peuvent être effectuées que lorsque la requête contient la balise `OpenSearchManaged=true` . La politique Elasticsearch obsolète a également été mise à jour pour assurer une compatibilité descendante.	7 novembre 2022

Modification	Description	Date
Mises à jour : `AmazonOpenSearchServiceRolePolicy` et `AmazonElasticsearchServiceRolePolicy`	Ajout de la prise en charge de l'`PutMetricData` action, qui est nécessaire pour publier les métriques OpenSearch du cluster sur Amazon CloudWatch. La politique Elasticsearch obsolète a également été mise à jour pour assurer une compatibilité descendante. Pour connaître le JSON de la politique, consultez Console IAM.	12 septembre 2022
Mises à jour : `AmazonOpenSearchServiceRolePolicy` et `AmazonElasticsearchServiceRolePolicy`	Ajout de la prise en charge du type de ressource `acm`. La politique fournit l'autorisation minimale AWS Certificate Manager (ACM) en lecture seule nécessaire au rôle lié au service pour vérifier et valider les ressources ACM afin de créer et de mettre à jour des domaines personnalisés activés pour les points de terminaison. La politique Elasticsearch obsolète a également été mise à jour pour assurer une compatibilité descendante.	28 juillet 2022

Modification	Description	Date
Mises à jour : `AmazonOpenSearchServiceCognitoAccess` et `AmazonESCognitoAccess`	Ajout de la prise en charge de l'`UpdateUserPoolClient` action, qui est nécessaire pour définir la configuration du groupe d'utilisateurs de Cognito lors de la mise à niveau d'Elastic search vers. OpenSearch Autorisations corrigées pour l'action `SetIdentityPoolRoles` pour permettre l'accès à toutes les ressources. La politique Elasticsearch obsolète a également été mise à jour pour assurer une compatibilité descendante.	20 décembre 2021
Mise à jour d'`AmazonOpenSearchServiceRolePolicy`	Ajout de la prise en charge du type de ressource `security-group` . La politique fournit les autorisations Amazon EC2 et Elastic Load Balancing minimales nécessaires pour que le rôle lié au service autorise l'accès au VPC.	9 septembre 2021

Modification	Description	Date
• Ajout de `AmazonOpenSearchServiceFullAccess` . • Obsolète `AmazonESFullAccess`	Cette nouvelle politique a pour but de remplacer la précédente. Les deux politiques fournissent un accès complet à l'API OpenSearch de configuration du service et à toutes les méthodes HTTP pour le OpenSearch APIs. Le contrôle précis des accès et les stratégies basées sur les ressources peuvent cependant restreindre l'accès.	7 septembre 2021
• Ajout de `AmazonOpenSearchServiceReadOnlyAccess` . • Obsolète `AmazonESReadOnlyAccess`	Cette nouvelle politique a pour but de remplacer la précédente. Les deux politiques fournissent un accès en lecture seule à l'API de configuration du OpenSearch service (`es:Describe* es:List*`,, `etes:Get*`) et aucun accès aux méthodes HTTP pour le. OpenSearch APIs	7 septembre 2021

Modification	Description	Date
• Ajout de `AmazonOpenSearchServiceCognitoAccess` . • Obsolète `AmazonESCognitoAccess`	Cette nouvelle politique a pour but de remplacer la précédente. Les deux politiques fournissent les autorisations Amazon Cognito minimales nécessaires pour activer l'authentification Cognito.	7 septembre 2021
• Ajout de AmazonOpenSearchServiceRolePolicy. • Obsolète `AmazonElasticsearchServiceRolePolicy`	Cette nouvelle politique a pour but de remplacer la précédente. Les deux politiques fournissent les autorisations Amazon EC2 et Elastic Load Balancing minimales nécessaires pour que le rôle lié au service autorise l'accès au VPC.	7 septembre 2021
Démarrage du suivi des modifications	Amazon OpenSearch Service suit désormais les modifications apportées aux politiques AWS gérées.	7 septembre 2021

Prévention du problème de l’adjoint confus entre services

Le problème de député confus est un problème de sécurité dans lequel une entité qui n’est pas autorisée à effectuer une action peut contraindre une entité plus privilégiée à le faire. En AWS, l'usurpation d'identité interservices peut entraîner la confusion des adjoints. L’usurpation d’identité entre services peut se produire lorsqu’un service (le service appelant) appelle un autre service (le service appelé). Le service appelant peut être manipulé et ses autorisations utilisées pour agir sur les ressources d’un autre client auxquelles on ne serait pas autorisé d’accéder autrement. Pour éviter cela, AWS fournit des outils qui vous aident à protéger vos données pour tous les services avec des principaux de service qui ont eu accès aux ressources de votre compte.

Nous recommandons d'utiliser les clés de contexte de condition `aws:SourceAccount`globale `aws:SourceArn`et les clés contextuelles dans les politiques de ressources afin de limiter les autorisations qu'Amazon OpenSearch Service accorde à un autre service à la ressource. Si la valeur `aws:SourceArn` ne contient pas l'ID du compte, tel qu'un ARN de compartiment Amazon S3, vous devez utiliser les deux clés de contexte de condition globale pour limiter les autorisations. Si vous utilisez les deux clés de contexte de condition globale et que la valeur `aws:SourceArn` contient l'ID de compte, la valeur `aws:SourceAccount` et le compte dans la valeur `aws:SourceArn` doivent utiliser le même ID de compte lorsqu'ils sont utilisés dans la même instruction de politique. Utilisez `aws:SourceArn` si vous souhaitez qu'une seule ressource soit associée à l'accès entre services. Utilisez `aws:SourceAccount` si vous souhaitez autoriser l'association d'une ressource de ce compte à l'utilisation interservices.

La valeur de `aws:SourceArn` doit être l'ARN du domaine de OpenSearch service.

Le moyen le plus efficace de se protéger contre le problème de député confus consiste à utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'ARN complet de la ressource. Si vous ne connaissez pas l'ARN complet de la ressource ou si vous spécifiez plusieurs ressources, utilisez la clé de contexte de condition globale `aws:SourceArn` avec des caractères génériques (*) pour les parties inconnues de l'ARN. Par exemple, `arn:aws:es:*:123456789012:*`.

L'exemple suivant montre comment utiliser les clés contextuelles de condition `aws:SourceAccount` globale `aws:SourceArn` et les clés de contexte dans OpenSearch Service pour éviter le problème de confusion des adjoints.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":{
      "Sid":"ConfusedDeputyPreventionExamplePolicy",
      "Effect":"Allow",
      "Principal":{
         "Service":"es.amazonaws.com"
      },
      "Action":"sts:AssumeRole",
      "Condition":{
         "StringEquals":{
            "aws:SourceAccount":"123456789012"
         },
         "ArnLike":{
```

```
            "aws:SourceArn":"arn:aws:es:region:123456789012:domain/my-domain"
         }
      }
   }
}
```

Contrôle d'accès précis dans Amazon Service OpenSearch

Le contrôle d'accès précis offre des moyens supplémentaires de contrôler l'accès à vos données sur Amazon OpenSearch Service. Par exemple, selon l'auteur de la demande, vous pouvez souhaiter qu'une recherche renvoie les résultats d'un seul index. Vous pouvez masquer certains champs dans vos documents ou exclure certains documents.

Le contrôle précis des accès offre les avantages suivants :

- Contrôle d'accès basé sur les rôles
- Sécurité au niveau de l'index, du document et du champ
- OpenSearch Tableaux de bord mutualisés
- Authentification de base HTTP pour OpenSearch les OpenSearch tableaux de bord

Rubriques

- Vue d'ensemble : contrôle d'accès précis et OpenSearch sécurité des services
- Concepts clés
- À propos de l'utilisateur principal
- Activation du contrôle précis des accès
- Accès aux OpenSearch tableaux de bord en tant qu'utilisateur principal
- Gestion des autorisations
- Configurations recommandées
- Limites
- Modification de l'utilisateur maître
- Utilisateurs principaux supplémentaires
- Instantanés manuels
- Intégrations

- Différences d'API REST
- Didacticiel : configurer un domaine avec un utilisateur principal IAM et l'authentification Amazon Cognito
- Didacticiel : configurer un domaine avec la base de données utilisateur interne et l'authentification de base HTTP

Vue d'ensemble : contrôle d'accès précis et OpenSearch sécurité des services

La sécurité d'Amazon OpenSearch Service comporte trois niveaux principaux :

Réseau

La première couche de sécurité est le réseau, qui détermine si les demandes atteignent un domaine OpenSearch de service. Si vous choisissez Public access (Accès public) lorsque vous créez un domaine, les demandes de tout client connecté à Internet peuvent atteindre le point de terminaison du domaine. Si vous choisissez VPC Access (Accès VPC), les clients doivent se connecter au VPC (et les groupes de sécurité associés doivent l'autoriser) pour qu'une demande atteigne le point de terminaison. Pour plus d'informations, consultez the section called “Prise en charge de VPC”.

Stratégie d'accès au domaine

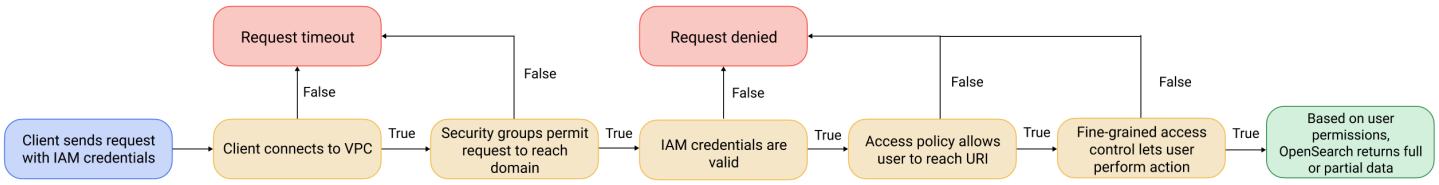
La deuxième couche de sécurité est la stratégie d'accès au domaine. Une fois qu'une requête atteint un point de terminaison de domaine, la stratégie d'accès basée sur les ressources autorise ou refuse l'accès de la demande à un URI donné. La stratégie d'accès accepte ou rejette les demandes au « bord » du domaine, avant qu'elles n'atteignent OpenSearch.

Contrôle précis des accès

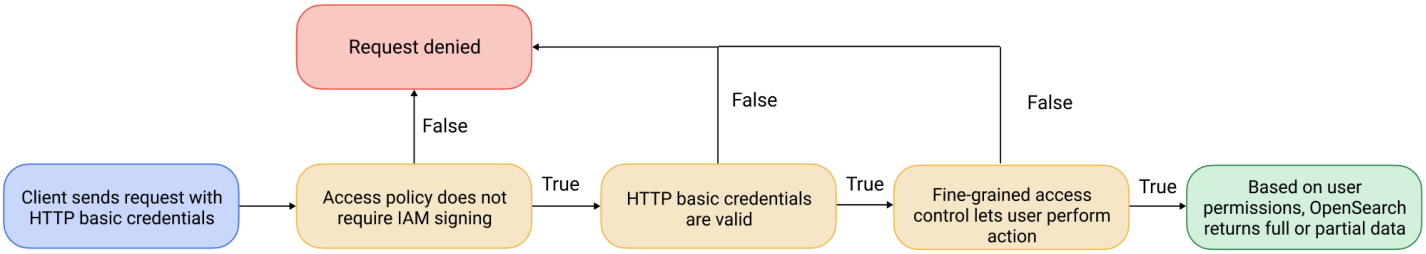
La troisième et dernière couche de sécurité est un contrôle précis des accès. Après qu'une stratégie d'accès basée sur les ressources autorise une demande à atteindre un point de terminaison de domaine, un contrôle précis des accès évalue les informations d'identification de l'utilisateur et authentifie l'utilisateur ou refuse la demande. Si le contrôle précis des accès authentifie l'utilisateur, il extrait tous les rôles mappés à cet utilisateur et utilise l'ensemble complet des autorisations pour déterminer comment traiter la demande.

> **Note**
>
> Si une politique d'accès basée sur les ressources contient des rôles ou des utilisateurs IAM, les clients doivent envoyer des demandes signées à l'aide de AWS Signature Version 4. Ainsi, les stratégies d'accès peuvent entrer en conflit avec le contrôle précis des accès, plus particulièrement si vous utilisez la base de données utilisateur interne et l'authentification de base HTTP. Vous ne pouvez pas signer une demande avec un nom d'utilisateur, un mot de passe et des informations d'identification IAM. En général, si vous activez le contrôle d'accès affiné, nous vous recommandons d'utiliser une stratégie d'accès au domaine qui ne nécessite pas de demandes signées.

Le diagramme suivant illustre une configuration courante : un domaine d'accès VPC avec contrôle précis des accès activé, une stratégie d'accès basée sur IAM et un utilisateur principal IAM.



Le diagramme suivant illustre une autre configuration courante : un domaine d'accès public avec contrôle précis des accès activé, une stratégie d'accès qui n'utilise pas d'entités IAM et un utilisateur principal dans la base de données utilisateur interne.



exemple

Considérez une demande `GET` adressée à `movies/_search?q=thor`. L'utilisateur dispose-t-il des autorisations pour effectuer une recherche dans l'index `movies` ? Si oui, l'utilisateur dispose-t-il des autorisations pour voir tous les documents qu'il contient ? La réponse devrait-elle omettre ou anonymiser des champs ? Pour l'utilisateur maître, la réponse peut se présenter comme suit :

```
{
  "hits": {
```

```
    "total": 7,
    "max_score": 8.772789,
    "hits": [{
        "_index": "movies",
        "_type": "_doc",
        "_id": "tt0800369",
        "_score": 8.772789,
        "_source": {
          "directors": [
            "Kenneth Branagh",
            "Joss Whedon"
          ],
          "release_date": "2011-04-21T00:00:00Z",
          "genres": [
            "Action",
            "Adventure",
            "Fantasy"
          ],
          "plot": "The powerful but arrogant god Thor is cast out of Asgard to
 live amongst humans in Midgard (Earth), where he soon becomes one of their finest
 defenders.",
          "title": "Thor",
          "actors": [
            "Chris Hemsworth",
            "Anthony Hopkins",
            "Natalie Portman"
          ],
          "year": 2011
        }
      },
      ...
    ]
  }
}
```

Si un utilisateur disposant d'autorisations plus limitées émet exactement la même demande, la réponse peut ressembler à ceci :

```
{
  "hits": {
    "total": 2,
    "max_score": 8.772789,
    "hits": [{
```

```
        "_index": "movies",
        "_type": "_doc",
        "_id": "tt0800369",
        "_score": 8.772789,
        "_source": {
          "year": 2011,
          "release_date":
 "3812a72c6dd23eef3c750c2d99e205cbd260389461e19d610406847397ecb357",
          "plot": "The powerful but arrogant god Thor is cast out of Asgard to
 live amongst humans in Midgard (Earth), where he soon becomes one of their finest
 defenders.",
          "title": "Thor"
        }
      },
      ...
    ]
  }
}
```

La réponse a moins d'accès et moins de champs pour chaque accès. En outre, le champ `release_date` est anonymisé. Si un utilisateur sans autorisation effectue la même demande, le cluster renvoie une erreur :

```
{
  "error": {
    "root_cause": [{
      "type": "security_exception",
      "reason": "no permissions for [indices:data/read/search] and User [name=limited-
user, roles=[], requestedTenant=null]"
    }],
    "type": "security_exception",
    "reason": "no permissions for [indices:data/read/search] and User [name=limited-
user, roles=[], requestedTenant=null]"
  },
  "status": 403
}
```

Si un utilisateur fournit des informations d'identification non valides, le cluster renvoie une exception `Unauthorized`.

Concepts clés

Lorsque vous vous lancez dans le contrôle d'accès détaillé, tenez compte des concepts suivants :

- Rôles : méthode de base pour utiliser un contrôle d'accès précis. Dans ce cas, les rôles sont distincts des rôles IAM. Les rôles contiennent n'importe quelle combinaison d'autorisations : à l'échelle du cluster, spécifique à l'index, au niveau du document et au niveau du champ.
- Cartographie : après avoir configuré un rôle, vous le mappez à un ou plusieurs utilisateurs. Par exemple, vous pouvez mapper trois rôles à un seul utilisateur : un rôle qui donne accès aux tableaux de bord, un qui fournit un accès à `index1` en lecture seule et un autre qui fournit un accès en écriture à `index2`. Vous pouvez également inclure toutes ces autorisations dans un seul rôle.
- Utilisateurs : personnes ou applications qui adressent des demandes au OpenSearch cluster. Les utilisateurs disposent d'informations d'identification, qu'il s'agisse de clés d'accès IAM ou d'un nom d'utilisateur et d'un mot de passe, qu'ils spécifient lorsqu'ils font des demandes.

À propos de l'utilisateur principal

L'utilisateur principal dans OpenSearch Service est soit une combinaison de nom d'utilisateur et de mot de passe, soit un utilisateur principal IAM disposant des autorisations complètes sur le OpenSearch cluster sous-jacent. Un utilisateur est considéré comme un utilisateur principal s'il a tous les accès au OpenSearch cluster et s'il a la possibilité de créer des utilisateurs internes, des rôles et des mappages de rôles dans les OpenSearch tableaux de bord.

Un utilisateur principal créé dans la console OpenSearch de service ou via la CLI est automatiquement mappé à deux rôles prédéfinis :

- `all_access`— Fournit un accès complet à toutes les opérations à l'échelle du cluster, l'autorisation d'écrire dans tous les index du cluster et l'autorisation d'écrire à tous les locataires.
- `security_manager`— Permet d'accéder au plugin de sécurité et de gérer les utilisateurs et les autorisations.

Ces deux rôles permettent à l'utilisateur d'accéder à l'onglet Sécurité des OpenSearch tableaux de bord, où il peut gérer les utilisateurs et les autorisations. Si vous créez un autre utilisateur interne et que vous le mappez uniquement au `all_access` rôle, l'utilisateur n'a pas accès à l'onglet Sécurité. Vous pouvez créer des utilisateurs principaux supplémentaires en les mappant explicitement aux

`security_manager` rôles `all_access` et. Pour obtenir des instructions, veuillez consulter the section called “Utilisateurs principaux supplémentaires”.

Lorsque vous créez un utilisateur principal pour votre domaine, vous pouvez spécifier un utilisateur principal IAM existant ou créer un utilisateur principal dans la base de données utilisateur interne. Tenez compte des points suivants lorsque vous décidez lequel utiliser :

- Principal IAM — Si vous choisissez un principal IAM pour votre utilisateur principal, toutes les demandes adressées au cluster doivent être signées à l'aide de AWS Signature Version 4.

  OpenSearch Le service ne prend en compte aucune des autorisations du principal IAM. L'utilisateur ou le rôle IAM sert uniquement à l'authentification. Les politiques relatives à cet utilisateur ou à ce rôle n'ont aucune incidence sur l'autorisation de l'utilisateur principal. L'autorisation est gérée par le biais des différentes autorisations du plugin OpenSearch de sécurité.

  Par exemple, vous pouvez n'attribuer aucune autorisation IAM à un principal IAM, et tant que la machine ou la personne peut s'authentifier auprès de cet utilisateur ou de ce rôle, elle dispose du pouvoir de l'utilisateur principal dans Service. OpenSearch

  Nous recommandons IAM si vous souhaitez utiliser les mêmes utilisateurs sur plusieurs clusters, si vous souhaitez utiliser Amazon Cognito pour accéder aux tableaux de bord ou si vous OpenSearch avez des clients qui prennent en charge la signature Signature version 4.
- Base de données utilisateur interne : si vous créez un maître dans la base de données utilisateur interne (avec une combinaison nom d'utilisateur et mot de passe), vous pouvez utiliser l'authentification HTTP de base (ainsi que les informations d'identification IAM) pour envoyer des demandes au cluster. La plupart des clients prennent en charge l'authentification de base, notamment curl, qui prend également en charge la version 4 de AWS Signature avec l'option --aws-sigv4. La base de données utilisateur interne est stockée dans un OpenSearch index, vous ne pouvez donc pas la partager avec d'autres clusters.

  Nous recommandons la base de données utilisateur interne si vous n'avez pas besoin de réutiliser les utilisateurs sur plusieurs clusters, si vous souhaitez utiliser l'authentification de base HTTP pour accéder aux tableaux de bord (plutôt qu'Amazon Cognito), ou si vous avez des clients qui prennent uniquement en charge l'authentification de base. La base de données utilisateur interne est le moyen le plus simple de démarrer avec OpenSearch Service.

Activation du contrôle précis des accès

Activez un contrôle d'accès précis à l'aide de la console ou de l' AWS CLI API de configuration. Pour les étapes, consultez Création et gestion des domaines.

Le contrôle d'accès détaillé nécessite Elasticsearch OpenSearch 6.7 ou version ultérieure. Il nécessite également le protocole HTTPS pour tout le trafic vers le domaine, le chiffrement des données au repos et le node-to-node chiffrement. Selon la façon dont vous configurez les fonctionnalités avancées du contrôle d'accès détaillé, le traitement supplémentaire de vos demandes peut nécessiter des ressources de calcul et de mémoire sur des nœuds de données individuels. Après avoir activé le contrôle précis des accès, vous ne pourrez pas le désactiver.

Activation du contrôle précis des accès sur des domaines existants

Vous pouvez activer un contrôle d'accès précis sur les domaines existants exécutant Elasticsearch OpenSearch 6.7 ou version ultérieure.

Pour activer le contrôle précis des accès sur un domaine existant (console)

1. Sélectionnez votre domaine et choisissez Actions et Edit security configuration (Modifier la configuration de sécurité).
2. Sélectionnez Enable fine-grained access control (Activer le contrôle précis des accès).
3. Choisissez comment créer l'utilisateur principal :
   - Si vous souhaitez utiliser IAM pour la gestion des utilisateurs, choisissez Set IAM ARN as master user (Définir l'ARN IAM en tant qu'utilisateur principal), puis indiquez l'ARN d'un rôle IAM.
   - Si vous souhaitez utiliser la base de données utilisateur interne, choisissez Create master user et spécifiez un nom d'utilisateur et un mot de passe.
4. (Facultatif) Sélectionnez Enable migration period for open/IP-based access policy (Activer la période de migration pour la stratégie d'accès Open/basée sur l'IP). Ce paramètre permet une période de transition de 30 jours pendant laquelle vos utilisateurs actuels peuvent continuer à accéder au domaine sans interruption. Les stratégies d'accès basées sur l'IP et Open existantes continueront à fonctionner avec votre domaine. Pendant cette période de migration, nous recommandons aux administrateurs de créer les rôles nécessaires et de les mapper aux utilisateurs pour le domaine. Si vous utilisez des stratégies basées sur l'identité au lieu d'une stratégie d'accès Open ou basée sur l'IP, vous pouvez désactiver ce paramètre.

Vous devez également mettre à jour vos clients pour qu'ils utilisent un contrôle précis des accès pendant la période de migration. Par exemple, si vous associez des rôles IAM à un contrôle d'accès précis, vous devez mettre vos clients à jour pour qu'ils commencent à signer les demandes avec AWS Signature Version 4. Si vous configurez l'authentification de base HTTP avec un contrôle précis des accès, vous devez mettre à jour vos clients pour qu'ils fournissent les informations d'identification d'authentification de base appropriées dans les demandes.

Pendant la période de migration, les utilisateurs qui accèdent au point de terminaison OpenSearch Dashboards pour le domaine arriveront directement sur la page de découverte plutôt que sur la page de connexion. Les administrateurs et les utilisateurs principaux peuvent choisir Login (Connexion) pour se connecter avec les informations d'identification de l'administrateur et configurer les mappages de rôles.

> **Important**
>
> OpenSearch Le service désactive automatiquement la période de migration après 30 jours. Nous vous recommandons de la terminer dès que vous aurez créé les rôles nécessaires et que vous les aurez mappés aux utilisateurs. Une fois la période de migration terminée, vous ne pouvez pas la réactiver.

5. Sélectionnez Enregistrer les modifications.

Le changement déclenche un déploiement bleu/vert pendant lequel l'état du cluster devient rouge, mais toutes les opérations du cluster ne sont pas affectées.

Pour activer le contrôle précis des accès sur un domaine existant (CLI)

Définissez `AnonymousAuthEnabled` sur `true` pour activer la période de migration avec un contrôle précis des accès :

```
aws opensearch update-domain-config --domain-name test-domain --region us-east-1 \
      --advanced-security-options '{ "Enabled": true,
 "InternalUserDatabaseEnabled":true, "MasterUserOptions": {"MasterUserName":"master-
username","MasterUserPassword":"master-password"},"AnonymousAuthEnabled": true}'
```

À propos du rôle default_role

Le contrôle précis des accès nécessite un mappage des rôles. Si votre domaine utilise des politiques d'accès basées sur l'identité, OpenSearch Service associe automatiquement vos utilisateurs à un nouveau rôle appelé default_role afin de vous aider à migrer correctement les utilisateurs existants. Ce mappage temporaire garantit que vos utilisateurs peuvent toujours envoyer avec succès des demandes GET et PUT signées par IAM jusqu'à ce que vous créiez vos propres mappages de rôles.

Le rôle n'ajoute aucune vulnérabilité ou faille de sécurité à votre domaine OpenSearch de service. Nous vous recommandons de supprimer le rôle par défaut dès que vous aurez configuré vos propres rôles et que vous les aurez mappés en conséquence.

Scénarios de migration

Le tableau suivant décrit le comportement de chaque méthode d'authentification avant et après l'activation du contrôle précis des accès sur un domaine existant, ainsi que les étapes que les administrateurs doivent suivre pour mapper correctement leurs utilisateurs aux rôles :

Méthode d'authentification	Avant l'activation du contrôle précis des accès	Après l'activation du contrôle précis des accès	Tâches de l'administrateur
Politiques basées sur l’identité	Tous les utilisateurs respectant la politique IAM peuvent accéder au domaine.	Vous n'avez pas besoin d'activer la période de migration. OpenSearch Le service mappe automatiquement tous les utilisateurs qui satisfont à la politique IAM au rôle default_role afin qu'ils puissent continuer à accéder au domaine.	1. Créez des mappages de rôles personnalisés sur le domaine. 2. Supprimez le rôle default_role.

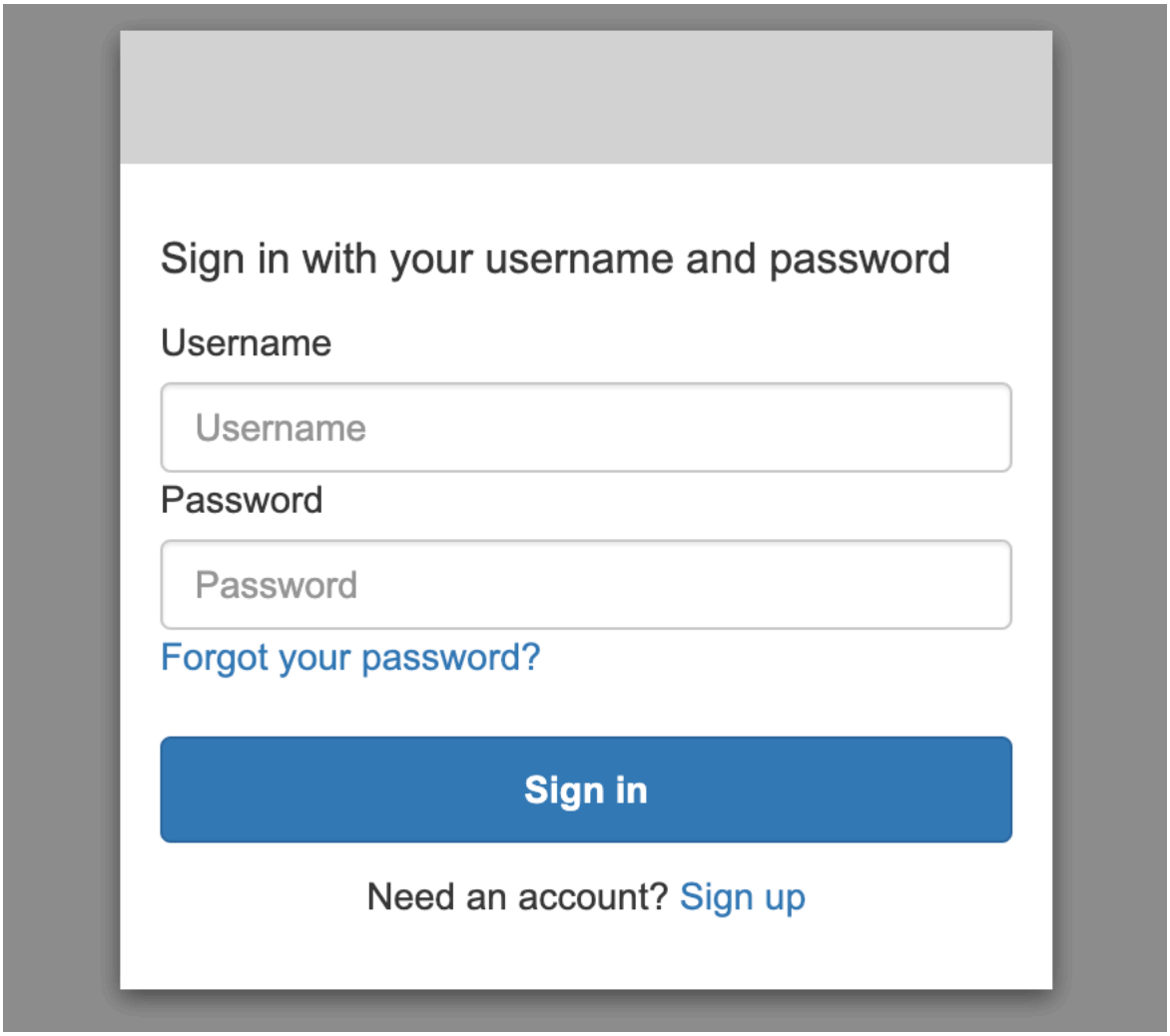
Méthode d'authentification	Avant l'activation du contrôle précis des accès	Après l'activation du contrôle précis des accès	Tâches de l'administrateur
Stratégies basées sur l'IP	Tous les utilisateurs des adresses IP ou des blocs d'adresses CIDR autorisés peuvent accéder au domaine.	Pendant la période de migration de 30 jours, tous les utilisateurs des adresses IP ou des blocs d'adresses CIDR autorisés peuvent continuer à accéder au domaine.	1. Créez des mappages de rôles personnalisés sur le domaine. 2. Mettez à jour vos clients pour qu'ils fournissent des informations d'identification d'authentification de base ou des informations d'identification IAM, selon la configuration de votre mappage de rôles. 3. Désactivez la période de migration. Les utilisateurs des adresses IP ou blocs d'adresses CIDR autorisés qui envoient des demandes sans authentification de base ou informations d'identification IAM perdront l'accès au domaine.
Stratégies d'accès Open	Tous les utilisateurs sur Internet peuvent accéder au domaine.	Pendant la période de migration de 30 jours, tous les utilisateurs sur Internet peuvent continuer à accéder au domaine.	1. Créez des mappages de rôles sur le domaine. 2. Mettez à jour vos clients pour qu'ils fournissent des informations d'identification d'authentification de base ou des informations d'identification IAM, selon la configuration de votre mappage de rôles. 3. Désactivez la période de migration. Les utilisateurs qui envoient des demandes sans authentification de base ou informations d'identification IAM perdront l'accès au domaine.

Accès aux OpenSearch tableaux de bord en tant qu'utilisateur principal

Le contrôle d'accès précis est doté d'un plugin OpenSearch Dashboards qui simplifie les tâches de gestion. Vous pouvez utiliser les tableaux de bord pour gérer les utilisateurs, les rôles, les mappages, les groupes d'actions et les locataires. La page de connexion OpenSearch aux tableaux de bord et la méthode d'authentification sous-jacente varient toutefois en fonction de la façon dont vous gérez les utilisateurs et configurez votre domaine.

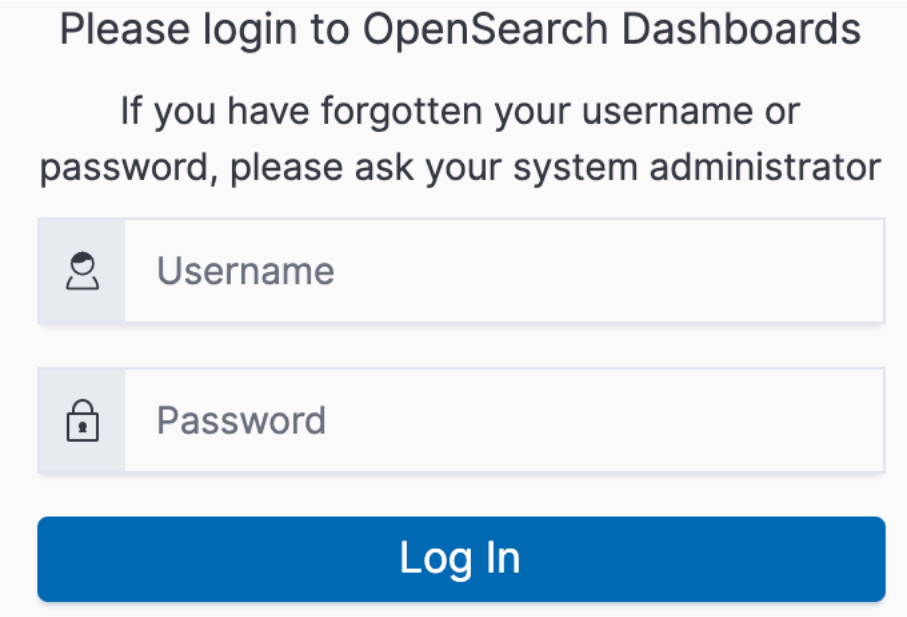
- Si vous souhaitez utiliser IAM pour la gestion des utilisateurs, utilisez the section called “Authentification Amazon Cognito pour les tableaux de bord OpenSearch ” pour accéder aux tableaux de bord. Sinon, les tableaux de bord afficheront une page de connexion non fonctionnelle. Consultez the section called “Limites”.

  Avec l'authentification Amazon Cognito, l'un des rôles assumés dans le pool d'identités doit correspondre au rôle IAM que vous avez spécifié pour l'utilisateur principal. Pour en savoir plus sur cette configuration, consultez the section called “(Facultatif) Configuration du contrôle précis des accès” et the section called “Didacticiel : contrôle précis des accès avec l'authentification Cognito”.



- Si vous choisissez d'utiliser la base de données utilisateur interne, vous pouvez vous connecter à Dashboards avec votre nom d'utilisateur et votre mot de passe principaux. Vous devez accéder aux tableaux de bord via HTTPS. L'authentification Amazon Cognito et SAML pour les tableaux de bord remplacent tous les deux cet écran de connexion.

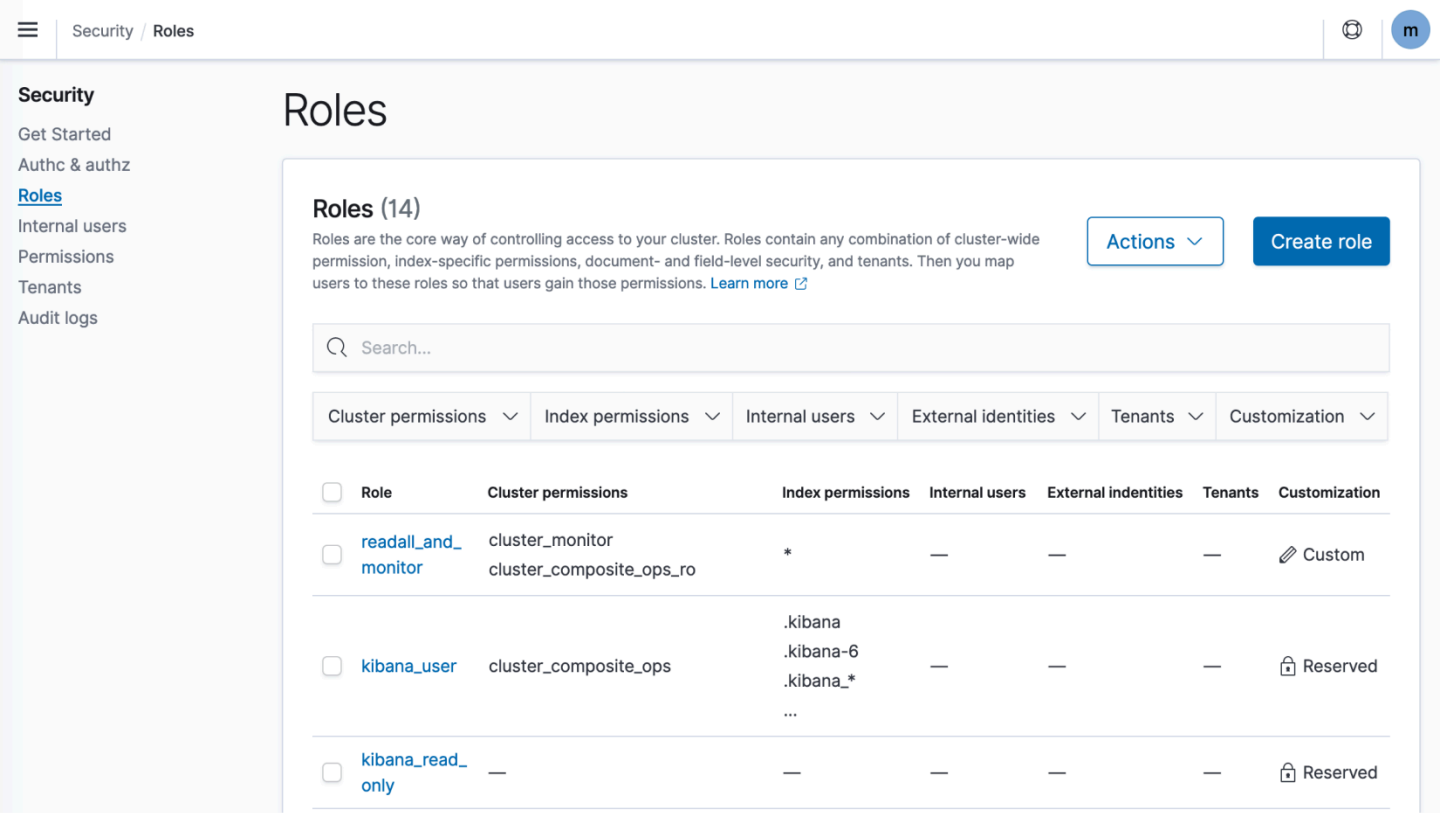
  Pour en savoir plus sur cette configuration, consultez the section called “Didacticiel : base de données utilisateur interne et authentification de base”.



- Si vous choisissez d'utiliser l'authentification SAML, vous pouvez vous connecter à l'aide des informations d'identification d'un fournisseur d'identité externe. Pour en savoir plus, consultez the section called “Authentification SAML pour les tableaux de bord OpenSearch ”.

Gestion des autorisations

Comme indiqué dans the section called “Concepts clés”, vous gérez les autorisations de contrôle précis des accès à l'aide de rôles, d'utilisateurs et de mappages. Cette section décrit comment créer et appliquer ces ressources. Nous vous recommandons de vous connecter aux tableaux de bord en tant qu'utilisateur principal pour exécuter ces opérations.



> **Note**
>
> Les autorisations que vous choisissez d'accorder aux utilisateurs varient considérablement en fonction du cas d'utilisation. Nous ne pouvons pas couvrir tous les scénarios de cette documentation. Lorsque vous déterminez les autorisations à accorder à vos utilisateurs, veillez à faire référence aux autorisations de OpenSearch cluster et d'index mentionnées dans les sections suivantes, et respectez toujours le principe du moindre privilège.

Créer des rôles

Vous pouvez créer de nouveaux rôles pour un contrôle d'accès précis à l'aide de OpenSearch tableaux de bord ou de l'`_plugins/_security`opération dans l'API REST. Pour en savoir plus, consultez Créer des rôles.

Le contrôle précis des accès inclut également un certain nombre de rôles prédéfinis. Les clients tels que OpenSearch Dashboards et Logstash envoient une grande variété de demandes OpenSearch, ce qui peut rendre difficile la création manuelle de rôles avec un minimum d'autorisations. Par exemple, le rôle `opensearch_dashboards_user` inclut les autorisations dont un utilisateur a

besoin pour créer des modèles d'index, des visualisations, des tableaux de bord et des locataires. Nous vous recommandons de le mapper à n'importe quel utilisateur ou rôle backend qui accède aux tableaux de bord, ainsi qu'aux rôles supplémentaires qui permettent d'accéder à d'autres index.

Amazon OpenSearch Service ne propose pas les OpenSearch rôles suivants :

- `observability_full_access`
- `observability_read_access`
- `reports_read_access`
- `reports_full_access`

Amazon OpenSearch Service propose plusieurs rôles qui ne sont pas disponibles avec OpenSearch :

- `ultrawarm_manager`
- `ml_full_access`
- `cold_manager`
- `notifications_full_access`
- `notifications_read_access`

Sécurité au niveau du cluster

Les autorisations au niveau du cluster permettent de faire des demandes étendues telles que `_mget`, `_msearch`, et `_bulk`, de surveiller l'état, de prendre des instantanés, etc. Gérez ces autorisations à l'aide de la section Autorisations de cluster lors de la création d'un rôle. Pour obtenir la liste complète des autorisations au niveau du cluster, consultez Autorisations de cluster.

Vous pouvez souvent atteindre la position de sécurité souhaitée à l'aide d'une combinaison des groupes d'actions par défaut, au lieu de le faire à l'aide d'autorisations individuelles. Pour obtenir la liste des groupes d'actions au niveau du cluster, consultez Niveau du cluster.

Sécurité au niveau de l'index

Les autorisations de niveau index incluent la possibilité de créer de nouveaux index, de rechercher des index, de lire et d'écrire des documents, de supprimer des documents, de gérer des alias, etc. Gérez ces autorisations à l'aide de la section Autorisations d'index lors de la création d'un rôle. Pour obtenir la liste complète des autorisations au niveau de l'indez, consultez Autorisations d'index.

Vous pouvez souvent atteindre la position de sécurité souhaitée à l'aide d'une combinaison des groupes d'actions par défaut, au lieu de le faire à l'aide d'autorisations individuelles. Pour obtenir la liste des groupes d'actions au niveau de l'index, consultez Niveau d'index.

Sécurité au niveau du document

La sécurité au niveau du document vous permet de restreindre les documents d'un index qu'un utilisateur peut consulter. Lorsque vous créez un rôle, spécifiez un modèle d'index et une OpenSearch requête. Tous les utilisateurs que vous mappez à ce rôle ne peuvent voir que les documents correspondant à la requête. La sécurité au niveau du document affecte le nombre d'accès que vous obtenez lorsque vous effectuez une recherche.

Pour en savoir plus, consultez Sécurité au niveau du document.

Sécurité au niveau du champ

La sécurité au niveau du champ vous permet de contrôler les champs de document qu'un utilisateur peut consulter. Lors de la création d'un rôle, ajoutez une liste de champs à inclure ou à exclure. Si vous incluez des champs, tous les utilisateurs que vous mappez à ce rôle ne peuvent voir que ces champs. Si vous excluez les champs, ils peuvent voir tous les champs sauf ceux exclus. La sécurité au niveau du champ affecte le nombre de champs inclus dans les appels lorsque vous effectuez une recherche.

Pour en savoir plus, consultez Sécurité au niveau du champ.

Masquage des champs

Le masquage des champs est une alternative à la sécurité au niveau du champ qui vous permet d'anonymiser les données d'un champ plutôt que de les supprimer complètement. Lors de la création d'un rôle, ajoutez une liste de champs à masquer. Le masquage des champs détermine si vous pouvez voir le contenu d'un champ lorsque vous effectuez une recherche.

> **Tip**
>
> Si vous appliquez le masquage standard à un champ, OpenSearch Service utilise un hachage aléatoire sécurisé qui peut entraîner des résultats d'agrégation inexacts. Pour effectuer des agrégations sur des champs masqués, utilisez plutôt un masquage basé sur un modèle.

Créer des utilisateurs

Si vous avez activé la base de données utilisateur interne, vous pouvez créer des utilisateurs à l'aide OpenSearch des tableaux de bord ou de l'`_plugins/_security`opération de l'API REST. Pour en savoir plus, consultez Créer des utilisateurs.

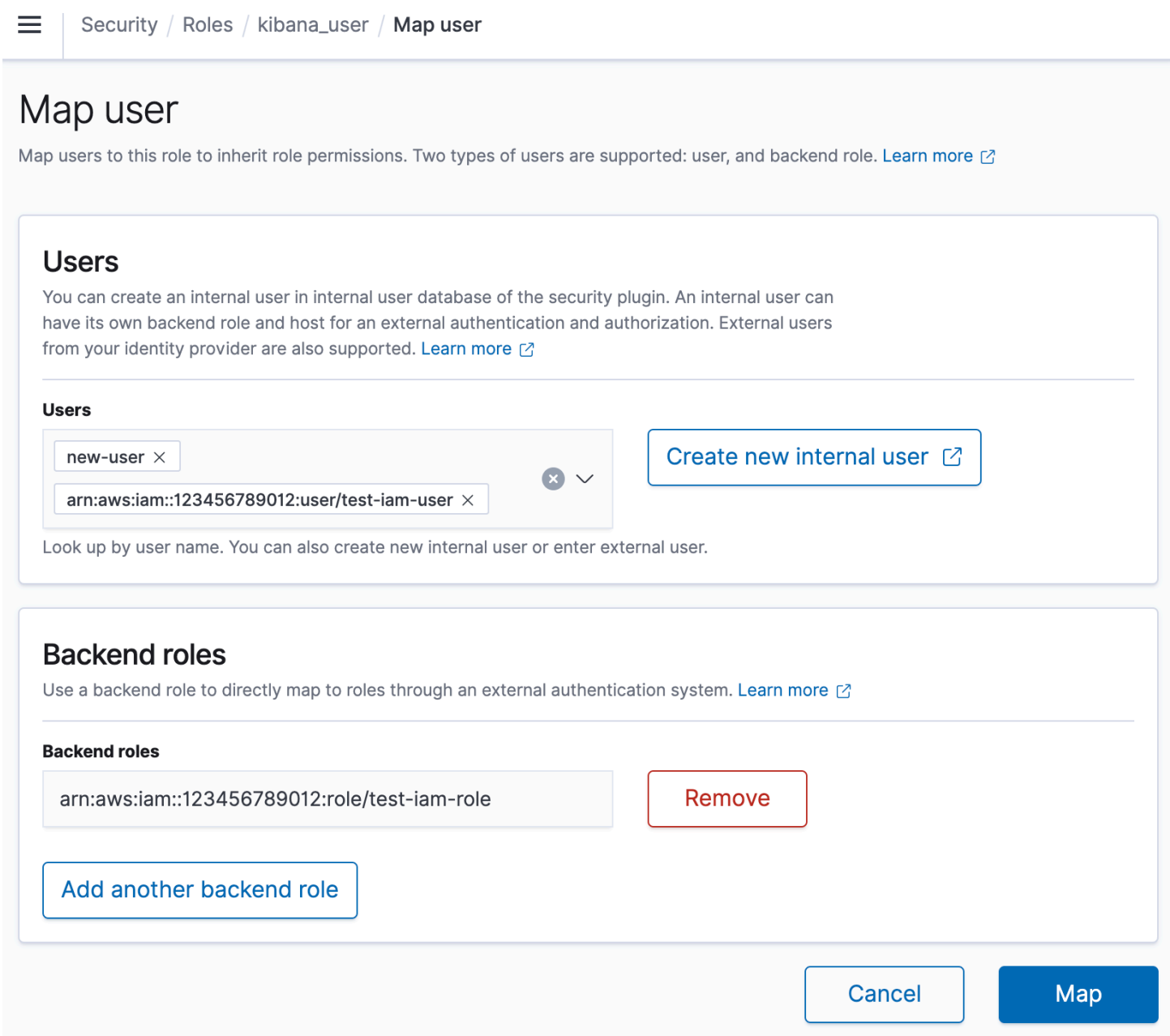
Si vous avez choisi IAM pour votre utilisateur principal, ignorez cette partie des tableaux de bord. Créez des rôles IAM à la place. Pour plus d'informations, consultez le Guide de l'utilisateur IAM.

Mappage des rôles aux utilisateurs

Le mappage des rôles est l'aspect le plus critique du contrôle précis des accès. Le contrôle précis des accès dispose de rôles prédéfinis pour vous aider à démarrer, mais à moins que vous ne mappiez des rôles à des utilisateurs, chaque demande adressée au cluster se termine par une erreur d'autorisation.

Les rôles principaux peuvent contribuer à simplifier le processus de mappage des rôles. Plutôt que de mapper le même rôle à 100 utilisateurs individuels, vous pouvez associer le rôle à un rôle principal partagé par les 100 utilisateurs. Les rôles backend peuvent correspondre à des rôles IAM ou à des chaînes arbitraires.

- Spécifiez les utilisateurs ARNs, les utilisateurs et les chaînes utilisateur Amazon Cognito dans la section Utilisateurs. Les chaînes utilisateur Cognito prennent la forme `Cognito/`*`user-pool-id`*`/`*`username`*.
- Spécifiez les rôles principaux et le rôle IAM ARNs dans la section Rôles principaux.



Vous pouvez associer les rôles aux utilisateurs à l'aide de OpenSearch tableaux de bord ou de l'`_plugins/_security`opération de l'API REST. Pour en savoir plus, consultez Mapper des utilisateurs à des rôles.

Créer des groupes d'actions

Les groupes d'actions sont des ensembles d'autorisations que vous pouvez réutiliser sur différentes ressources. Vous pouvez créer de nouveaux groupes d'actions à l'aide de OpenSearch tableaux de bord ou de l'`_plugins/_security`opération de l'API REST, bien que les groupes d'actions

par défaut soient suffisants dans la plupart des cas d'utilisation. Pour en savoir plus sur les groupes d'actions par défaut, consultez Groupes d'actions par défaut.

OpenSearch Tableaux de bord mutualisés

Les locataires sont des espaces permettant d'enregistrer des modèles d'index, des visualisations, des tableaux de bord et d'autres objets des tableaux de bord. La mutualisation des tableaux de bord vous permet de partager en toute sécurité votre travail avec d'autres utilisateurs de Dashboards (ou de le garder privé) et de configurer les locataires de manière dynamique. Vous pouvez contrôler quels rôles ont accès à un locataire et si ces rôles ont un accès en lecture ou en écriture. Le locataire global est le client par défaut. Pour en savoir plus, consultez la section OpenSearch Tableaux de bord mutualisés.

Pour afficher votre locataire actuel ou changer de locataire

1. Accédez aux OpenSearch tableaux de bord et connectez-vous.
2. Sélectionnez l'icône de votre utilisateur en haut à droite et choisissez Switch tenants (Changer les locataires).
3. Vérifiez votre locataire avant de créer des visualisations ou des tableaux de bord. Si vous souhaitez partager votre travail avec tous les autres utilisateurs des tableaux de bord, choisissez Global. Pour partager votre travail avec un sous-ensemble d'utilisateurs des tableaux de bord, choisissez un autre locataire partagé. Sinon, choisissez Private (Privé).

> **Note**
>
> OpenSearch Dashboards gère un index distinct pour chaque locataire et crée un modèle d'index appelé`tenant_template`. Ne supprimez ni ne modifiez l'`tenant_template`index, car cela pourrait entraîner un dysfonctionnement des OpenSearch tableaux de bord si le mappage de l'index des locataires est mal configuré.

Configurations recommandées

En raison de la façon dont le contrôle d'accès affiné interagit avec d'autres fonctionnalités de sécurité, nous recommandons plusieurs configurations de contrôle d'accès affiné qui fonctionnent bien dans la plupart des cas d'utilisation.

Description	Utilisateur maître	Stratégie d'accès au domaine
Utilisez les informations d'identification IAM pour les appels vers le OpenSearch APIs, et utilisez l'authentification SAML pour accéder aux tableaux de bord. Gérer les rôles de contrôle précis des accès à l'aide de l'API REST.	Rôle ou utilisateur IAM	JSON `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "AWS": "*" }, "Action": "es:ESHttp*", "Resource": " domain-arn /*" } ] }`
Utilisez les informations d'identification IAM ou l'authentification de base pour les appels vers le OpenSearch APIs. Gérer les rôles de contrôle précis des accès à l'aide de l'API REST. Cette configuration offre une grande flexibilité, en particulier si vous avez des OpenSearch clients qui ne prennent en charge que l'authentification de base. Si vous disposez d'un fournisseur d'identité	Nom d'utilisateur et mot de passe	JSON `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "AWS": "*" }, "Action": "es:ESHttp*", "Resource": " domain-arn /*" } ] }`

Description	Utilisateur maître	Stratégie d'accès au domaine
existant, utilisez l'authentification SAML pour accéder aux tableaux de bord. Dans le cas contraire, gérez les utilisateurs des tableaux de bord dans la base de données utilisateur interne.		
Utilisez les informations d'identification IAM pour les appels vers le OpenSearch APIs, et utilisez Amazon Cognito pour accéder aux tableaux de bord. Gérer les rôles de contrôle précis des accès à l'aide de l'API REST.	Rôle ou utilisateur IAM	JSON `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "AWS": "*" }, "Action": "es:ESHttp*", "Resource": " domain-arn /*" } ] }`

Description	Utilisateur maître	Stratégie d'accès au domaine
Utilisez les informations d'identification IAM pour les appels vers les OpenSearch APIs tableaux de bord et bloquez la plupart des accès à ceux-ci. Gérer les rôles de contrôle précis des accès à l'aide de l'API REST.	Rôle ou utilisateur IAM	JSON `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "AWS": "*" }, "Action": "es:ESHttp*", "Resource": " domain-arn /*" }, { "Effect": "Deny", "Principal": { "AWS": "*" }, "Action": "es:ESHttp*", "Resource": " domain-arn /_dashboards*" } ] }`

Limites

Le contrôle précis des accès présente plusieurs limites importantes :

- L'aspect `hosts` des mappages de rôles, qui mappe les rôles aux noms d'hôte ou aux adresses IP, ne fonctionne pas si le domaine se trouve dans un VPC. Vous pouvez toujours mapper les rôles avec les utilisateurs et les rôles backend.
- Si vous choisissez IAM pour l'utilisateur principal et que vous n'activez pas Amazon Cognito ou l'authentification SAML, les tableaux de bord afficheront une page de connexion non fonctionnelle.

- Si vous choisissez IAM pour l'utilisateur principal, vous pouvez toujours créer des utilisateurs dans la base de données utilisateur interne. Étant donné que l'authentification de base HTTP n'est pas activée dans cette configuration, toutes les demandes signées avec ces informations d'identification utilisateur sont rejetées.
- Si vous utilisez SQL pour interroger un index auquel vous n'avez pas accès, vous recevez une erreur « aucune autorisation ». Si l'index n'existe pas, vous recevez une erreur « no such index » (L'index n'existe pas). Cette différence dans les messages d'erreur signifie que vous pouvez confirmer l'existence d'un index si vous devinez son nom.

  Pour minimiser le problème, n'incluez pas d'informations sensibles dans les noms d'index. Pour refuser tout accès à SQL, ajoutez l'élément suivant à votre stratégie d'accès au domaine :

```
{
  "Effect": "Deny",
  "Principal": {
    "AWS": [
      "*"
    ]
  },
  "Action": [
    "es:*"
  ],
  "Resource": "arn:aws:es:us-east-1:123456789012:domain/my-domain/_plugins/_sql"
}
```

- Si la version de votre domaine est 2.3 ou supérieure et que le contrôle d'accès détaillé est activé, le réglage sur 1 `max_clause_count` entraîne des problèmes avec votre domaine. Nous vous recommandons de définir un nombre plus élevé pour ce compte.
- Si vous activez le contrôle d'accès détaillé dans un domaine où le contrôle d'accès détaillé n'est pas configuré, pour les sources de données créées pour une requête directe, vous devez configurer vous-même des rôles de contrôle d'accès précis. Pour plus d'informations sur la façon de configurer des rôles d'accès précis, consultez Création d'intégrations de sources de données Amazon OpenSearch Service avec Amazon S3.

Modification de l'utilisateur maître

Si vous oubliez les détails de l'utilisateur principal, vous pouvez le reconfigurer à l'aide de la console, de l' AWS CLI ou de l'API de configuration.

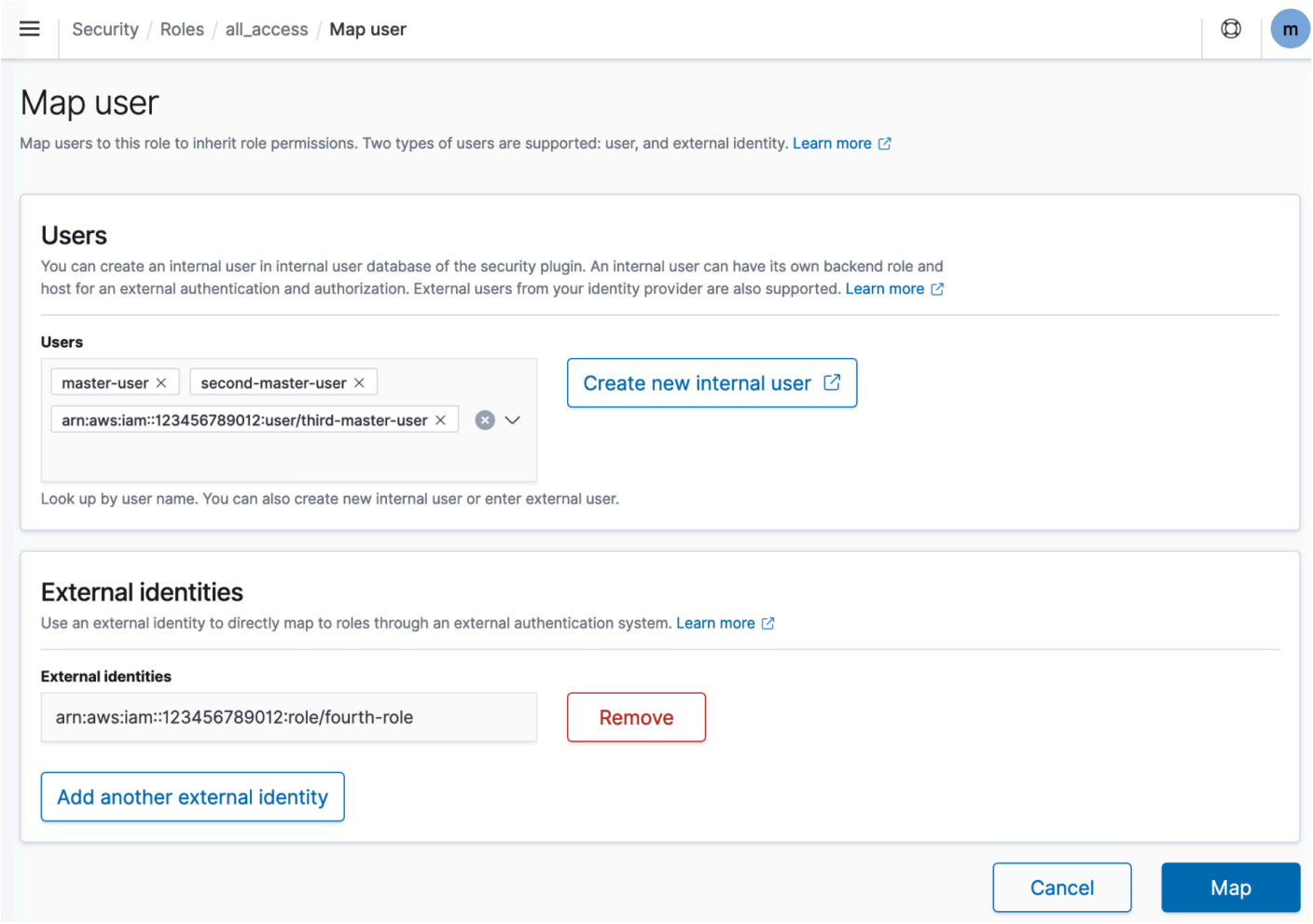
Pour modifier l'utilisateur maître (console)

1. Accédez à la console Amazon OpenSearch Service à l'adresse https://console.aws.amazon.com/aos/home/.
2. Sélectionnez votre domaine et choisissez Actions et Edit security configuration (Modifier la configuration de sécurité).
3. Choisissez Set IAM ARN as master user (Définir l'ARN IAM en tant qu'utilisateur principal) ou Create master user (Créer un nouvel utilisateur principal).
   - Si vous avez précédemment utilisé un utilisateur principal IAM, le contrôle précis des accès remappera le rôle `all_access` avec le nouvel ARN IAM que vous indiquerez.
   - Si vous avez précédemment utilisé la base de données utilisateur interne, le contrôle précis des accès créera un nouvel utilisateur principal. Vous pouvez utiliser le nouvel utilisateur principal pour supprimer l'ancien.
   - Le passage de la base de données utilisateur interne à un utilisateur principal IAM ne supprime aucun utilisateur de la base de données utilisateur interne. Cela permet simplement de désactiver l'authentification de base HTTP. Supprimez manuellement les utilisateurs de la base de données utilisateur interne ou conservez-les au cas où vous auriez besoin de réactiver l'authentification HTTP de base.
4. Sélectionnez Enregistrer les modifications.

Utilisateurs principaux supplémentaires

Vous désignez un utilisateur principal lorsque vous créez un domaine, mais si vous le souhaitez, vous pouvez utiliser cet utilisateur principal pour créer d'autres utilisateurs principaux. Deux options s'offrent à vous : les OpenSearch tableaux de bord ou l'API REST.

- Dans les tableaux de bord, choisissez Security (Sécurité), Roles (Rôles), puis mappez le nouvel utilisateur principal aux rôles `all_access` et `security_manager`.



- Pour utiliser l'API REST, envoyez les requêtes suivantes :

```
PUT _plugins/_security/api/rolesmapping/all_access
{
  "backend_roles": [
    "arn:aws:iam::123456789012:role/fourth-master-user"
  ],
  "hosts": [],
  "users": [
    "master-user",
    "second-master-user",
    "arn:aws:iam::123456789012:user/third-master-user"
  ]
}
```

```
PUT _plugins/_security/api/rolesmapping/security_manager
{
```

```
  "backend_roles": [
    "arn:aws:iam::123456789012:role/fourth-master-user"
  ],
  "hosts": [],
  "users": [
    "master-user",
    "second-master-user",
    "arn:aws:iam::123456789012:user/third-master-user"
  ]
}
```

Comme ces demandes remplacent les mappages de rôles actuels, exécutez d'abord les demandes `GET` afin que vous puissiez inclure tous les rôles actuels dans les demandes `PUT`. L'API REST est particulièrement utile si vous ne pouvez pas accéder aux tableaux de bord et que vous souhaitez mapper un rôle IAM Amazon Cognito au rôle `all_access`.

Instantanés manuels

Le contrôle précis des accès introduit quelques complications supplémentaires avec la prise des instantanés manuels. Pour enregistrer un référentiel d'instantanés, même si vous utilisez l'authentification de base HTTP à d'autres fins, vous devez mapper le rôle `manage_snapshots` à un rôle IAM disposant des autorisations `iam:PassRole` pour assumer `TheSnapshotRole`, comme défini dans the section called “Prérequis”.

Utilisez ensuite ce rôle IAM pour envoyer une demande signée au domaine, comme indiqué dans the section called “Inscription d'un référentiel d'instantanés manuels”.

Intégrations

Si vous utilisez d'autres AWS services avec OpenSearch Service, vous devez fournir les rôles IAM pour ces services avec les autorisations appropriées. Par exemple, les flux de diffusion Firehose utilisent souvent un rôle IAM appelé. `firehose_delivery_role` Dans Tableaux de bord, créez un rôle pour le contrôle précis des accès, puis mappez le rôle IAM à celui-ci. Dans ce cas, le nouveau rôle nécessite les autorisations suivantes :

```
{
  "cluster_permissions": [
    "cluster_composite_ops",
    "cluster_monitor"
  ],
```

```
  "index_permissions": [{
    "index_patterns": [
      "firehose-index*"
    ],
    "allowed_actions": [
      "create_index",
      "manage",
      "crud"
    ]
  }]
}
```

Les autorisations varient en fonction des actions effectuées par chaque service. Une AWS IoT règle ou une AWS Lambda fonction qui indexe des données nécessite probablement des autorisations similaires à celles de Firehose, tandis qu'une fonction Lambda qui effectue uniquement des recherches peut utiliser un ensemble plus limité.

Différences d'API REST

L'API REST à contrôle d'accès affiné diffère légèrement en fonction de votre version OpenSearch/Elasticsearch . Avant d'adresser une demande PUT, effectuez une demande GET pour vérifier le corps de requête attendu. Par exemple, une demande GET adressée à `_plugins/_security/api/user` renvoie tous les utilisateurs, que vous pouvez ensuite modifier et utiliser pour effectuer des demandes PUT valides.

Sur Elasticsearch 6x, les demandes de création d'utilisateurs se présentent comme suit :

```
PUT _opendistro/_security/api/user/new-user
{
  "password": "some-password",
  "roles": ["new-backend-role"]
}
```

Sur Elasticsearch 7.x OpenSearch ou sur Elasticsearch, les requêtes se présentent comme suit (remplacez `_plugins` par Elasticsearch `_opendistro` si vous utilisez Elasticsearch) :

```
PUT _plugins/_security/api/user/new-user
{
  "password": "some-password",
  "backend_roles": ["new-backend-role"]
}
```

En outre, les locataires sont les propriétés des rôles dans Elasticsearch 6.x :

```
GET _opendistro/_security/api/roles/all_access

{
  "all_access": {
    "cluster": ["UNLIMITED"],
    "tenants": {
      "admin_tenant": "RW"
    },
    "indices": {
      "*": {
        "*": ["UNLIMITED"]
      }
    },
    "readonly": "true"
  }
}
```

Dans OpenSearch Elasticsearch 7.x, ce sont des objets dotés de leur propre URI (`_pluginsremplacez-le _opendistro` si vous utilisez Elasticsearch) :

```
GET _plugins/_security/api/tenants

{
  "global_tenant": {
    "reserved": true,
    "hidden": false,
    "description": "Global tenant",
    "static": false
  }
}
```

Pour obtenir de la documentation sur l' OpenSearch API REST, consultez la référence de l'API du plugin de sécurité.

Tip

Si vous utilisez la base de données utilisateur interne, vous pouvez utiliser curl pour effectuer des demandes et tester votre domaine. Essayez les exemples de commande suivants :

```
curl -XGET -u 'master-user:master-user-password' 'domain-endpoint/_search'
```

```
curl -XGET -u 'master-user:master-user-password' 'domain-endpoint/_plugins/
_security/api/user'
```

Didacticiel : configurer un domaine avec un utilisateur principal IAM et l'authentification Amazon Cognito

Ce didacticiel couvre un cas d'utilisation courant d'Amazon OpenSearch Service pour le contrôle d'accès détaillé : un utilisateur principal IAM avec authentification Amazon Cognito pour les tableaux de bord. OpenSearch

Dans ce didacticiel, nous allons configurer un rôle IAM principal et un rôle IAM limité, que nous associerons ensuite aux utilisateurs dans Amazon Cognito. L'utilisateur principal peut ensuite se connecter aux OpenSearch tableaux de bord, associer l'utilisateur limité à un rôle et utiliser un contrôle d'accès précis pour limiter les autorisations de l'utilisateur.



Bien que ces étapes utilisent le pool d'utilisateurs Amazon Cognito pour l'authentification, ce même processus de base fonctionne pour tout fournisseur d'authentification Cognito qui vous permet d'attribuer différents rôles IAM à différents utilisateurs.

Dans le cadre de ce didacticiel, vous suivrez les étapes suivantes :

1. Créer les rôles IAM principal et limité
2. Créer un domaine avec authentification Cognito
3. Configuration d'un groupe d'utilisateurs et d'identités Cognito
4. Cartographier les rôles dans les OpenSearch tableaux de bord
5. Tester les autorisations

Étape 1 : créer les rôles IAM principal et limité

Accédez à la console AWS Identity and Access Management (IAM) et créez deux rôles distincts :

- `MasterUserRole` : l'utilisateur principal, qui dispose des autorisations complètes sur le cluster et gère les rôles et les mappages de rôles.
- `LimitedUserRole` : un rôle plus restreint, auquel vous accorderez un accès limité en tant qu'utilisateur principal.

Pour obtenir des instructions sur la création des rôles, consultez la section Création d'un rôle à l'aide de politiques de confiance personnalisées dans le guide de l'utilisateur IAM.

Les deux rôles doivent disposer de la politique d'approbation suivante, qui permet à votre groupe d'identités Cognito d'endosser les rôles :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "Federated": "cognito-identity.amazonaws.com"
    },
    "Action": "sts:AssumeRoleWithWebIdentity",
    "Condition": {
      "StringEquals": {
        "cognito-identity.amazonaws.com:aud": "{identity-pool-id}"
      },
      "ForAnyValue:StringLike": {
        "cognito-identity.amazonaws.com:amr": "authenticated"
      }
    }
  }]
}
```

> **Note**
>
> Remplacez `identity-pool-id` par l'identifiant unique de votre groupe d'identités Amazon Cognito. Par exemple, `us-east-1:0c6cdba7-3c3c-443b-a958-fb9feb207aa6`.

Étape 2 : créer un domaine avec authentification Cognito

Accédez à la console Amazon OpenSearch Service à l'https://console.aws.amazon.com/aos/adresse home/ et créez un domaine avec les paramètres suivants :

- OpenSearch 1.0 ou version ultérieure, ou Elasticsearch 7.8 ou version ultérieure
- Accès public
- Contrôle précis des accès activé avec `MasterUserRole` comme utilisateur principal (créé à l'étape précédente)
- Authentification Amazon Cognito activée pour OpenSearch les tableaux de bord. Pour obtenir des instructions sur l'activation de l'authentification Cognito et choisir un groupe d'utilisateurs et d'identités, consultez the section called “Configurer un domaine pour utiliser l'authentification Amazon Cognito”.
- La stratégie d'accès au domaine suivante :

  JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::{account-id}:root"
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:{region}:{account-id}:domain/{domain-name}/*"
    }
  ]
}
```

- HTTPS requis pour tout le trafic vers le domaine

- Node-to-node chiffrement
- Chiffrement de données au repos

Étape 3 : Configuration des utilisateurs de Cognito

Lors de la création de votre domaine, configurez les utilisateurs principaux et limités dans Amazon Cognito en suivant la procédure Créer un groupe d'utilisateurs dans le manuel Amazon Cognito Developer Guide. Enfin, configurez votre pool d'identités en suivant les étapes décrites dans Créer un pool d'identités dans Amazon Cognito. Le groupe d'utilisateurs et le groupe d'identités doivent se trouver dans la même Région AWS.

Étape 4 : Cartographier les rôles dans les OpenSearch tableaux de bord

Maintenant que vos utilisateurs sont configurés, vous pouvez vous connecter à OpenSearch Dashboards en tant qu'utilisateur principal et associer les utilisateurs à des rôles.

1. Retournez à la console OpenSearch de service et accédez à l'URL OpenSearch des tableaux de bord du domaine que vous avez créé. Le format de l'URL est le suivant : *domain-endpoint*`/_dashboards/`.
2. Connectez-vous à l'aide des informations d'identification `master-user`.
3. Choisissez Add sample data (Ajouter des exemples de données), puis ajoutez les exemples de données de vol.
4. Dans le panneau de navigation de gauche, choisissez Security (Sécurité), Roles (Rôles), Create role (Créer un rôle).
5. Nommez le rôle `new-role`.
6. Pour Index, spécifiez `opensearch_dashboards_sample_data_fli*` (`kibana_sample_data_fli*` sur les domaines Elasticsearch).
7. Pour Index permissions (Autorisations d'index), choisissez read (lire).
8. Pour Requête de sécurité au niveau du document, indiquez la requête suivante :

```
{
  "match": {
    "FlightDelay": true
  }
}
```

9. Pour la sécurité au niveau des champs, choisissez Exclude (Exclure) et indiquez `FlightNum`.
10. Pour Anonymisation, indiquez `Dest`.
11. Choisissez Create (Créer).
12. Choisissez Mapped users (Utilisateurs mappés), Manage mapping (Gérer le mappage). Ajoutez l'Amazon Resource Name (ARN) pour `LimitedUserRole` en tant qu'identité externe et choisissez Map (Mapper).
13. Revenez à la liste des rôles et choisissez opensearch_dashboards_user. Choisissez Mapped users (Utilisateurs mappés), Manage mapping (Gérer le mappage). Ajoutez l'ARN pour `LimitedUserRole` en tant que rôle backend, puis choisissez Map (Mapper).

Étape 5 : tester les autorisations

Lorsque vos rôles sont correctement mappés, vous pouvez vous connecter en tant qu'utilisateur limité et tester les autorisations.

1. Dans une nouvelle fenêtre de navigateur privée, accédez à l'URL OpenSearch des tableaux de bord du domaine, connectez-vous à l'aide des `limited-user` informations d'identification et choisissez Explorer par moi-même.
2. Accédez aux Outils de développement, puis exécutez la recherche par défaut :

   ```
   GET _search
   {
     "query": {
       "match_all": {}
     }
   }
   ```

   Notez l'erreur d'autorisation. `limited-user` n'a pas les autorisations nécessaires pour exécuter des recherches à l'échelle du cluster.
3. Exécutez une autre recherche :

   ```
   GET opensearch_dashboards_sample_data_flights/_search
   {
     "query": {
       "match_all": {}
     }
   }
   ```

Notez que tous les documents correspondants ont un champ `FlightDelay` ayant pour valeur `true`, un champ anonymisé `Dest` et aucun champ `FlightNum`.

4. Dans la fenêtre de votre navigateur d'origine, connecté en tant que `master-user`, choisissez Dev Tools (Outils de développement), puis effectuez les mêmes recherches. Notez la différence entre les autorisations, le nombre d'accès, les documents correspondants et les champs inclus.

Didacticiel : configurer un domaine avec la base de données utilisateur interne et l'authentification de base HTTP

Ce didacticiel couvre un autre cas d'utilisation courant du contrôle d'accès détaillé : un utilisateur principal dans la base de données utilisateur interne et l'authentification de base HTTP pour les OpenSearch tableaux de bord. L'utilisateur principal peut ensuite se connecter aux OpenSearch tableaux de bord, créer un utilisateur interne, associer l'utilisateur à un rôle et utiliser un contrôle d'accès précis pour limiter les autorisations de l'utilisateur.

Dans le cadre de ce didacticiel, vous suivrez les étapes suivantes :

1. Création d'un domaine avec un utilisateur principal
2. Configuration d'un utilisateur interne dans les OpenSearch tableaux de bord
3. Cartographier les rôles dans les OpenSearch tableaux de bord
4. Tester les autorisations

Étape 1 : Créer un domaine

Accédez à la console Amazon OpenSearch Service à l'https://console.aws.amazon.com/aos/adresse home/ et créez un domaine avec les paramètres suivants :

- OpenSearch 1.0 ou version ultérieure, ou Elasticsearch 7.9 ou version ultérieure
- Accès public
- Contrôle précis des accès avec un utilisateur principal dans la base de données utilisateur interne (`TheMasterUser` pour le reste du didacticiel)
- Authentification Amazon Cognito pour les tableaux de bord désactivés
- La stratégie d'accès suivante :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::{account-id}:root"
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:{region}:{account-id}:domain/{domain-name}/*"
    }
  ]
}
```

- HTTPS requis pour tout le trafic vers le domaine
- Node-to-node chiffrement
- Chiffrement de données au repos

Étape 2 : créer un utilisateur interne dans les OpenSearch tableaux de bord

Maintenant que vous avez un domaine, vous pouvez vous connecter à OpenSearch Dashboards et créer un utilisateur interne.

1. Retournez à la console OpenSearch de service et accédez à l'URL OpenSearch des tableaux de bord du domaine que vous avez créé. Le format de l'URL est le suivant : *domain-endpoint*`/_dashboards/`.
2. Connectez-vous à l'aide du`TheMasterUser`.
3. Choisissez Add sample data (Ajouter des exemples de données), puis ajoutez les exemples de données de vol.
4. Dans le volet de navigation de gauche, choisissez Sécurité, Utilisateurs internes, Créer un utilisateur interne.
5. Nommez l'utilisateur `new-user` et spécifiez un mot de passe. Ensuite, choisissez Create (Créer).

Étape 3 : Cartographier les rôles dans les OpenSearch tableaux de bord

Maintenant que votre utilisateur est configuré, vous pouvez le mapper à un rôle.

1. Restez dans la section Sécurité des OpenSearch tableaux de bord et choisissez Rôles, Créer un rôle.
2. Nommez le rôle `new-role`.
3. Pour Index, spécifiez `opensearch_dashboards_sample_data_fli*` (`kibana_sample_data_fli*`sur les domaines Elasticsearch) le modèle d'index.
4. Pour le groupe d'actions, choisissez read (lire).
5. Pour Requête de sécurité au niveau du document, indiquez la requête suivante :

```
{
  "match": {
    "FlightDelay": true
  }
}
```

6. Pour la sécurité au niveau des champs, choisissez Exclude (Exclure) et indiquez `FlightNum`.
7. Pour Anonymisation, indiquez `Dest`.
8. Choisissez Create (Créer).
9. Choisissez Mapped users (Utilisateurs mappés), Manage mapping (Gérer le mappage). Ensuite, ajoutez `new-user` à Users (Utilisateurs) et choisissez Map (Mapper).
10. Revenez à la liste des rôles et choisissez opensearch_dashboards_user. Choisissez Mapped users (Utilisateurs mappés), Manage mapping (Gérer le mappage). Ensuite, ajoutez `new-user` à Users (Utilisateurs) et choisissez Map (Mapper).

Étape 4 : tester les autorisations

Lorsque vos rôles sont correctement mappés, vous pouvez vous connecter en tant qu'utilisateur limité et tester les autorisations.

1. Dans une nouvelle fenêtre de navigateur privée, accédez à l'URL OpenSearch des tableaux de bord du domaine, connectez-vous à l'aide des `new-user` informations d'identification et choisissez Explorer par moi-même.
2. Accédez aux Outils de développement, puis exécutez la recherche par défaut :

```
GET _search
{
  "query": {
    "match_all": {}
  }
}
```

Notez l'erreur d'autorisation. `new-user` n'a pas les autorisations nécessaires pour exécuter des recherches à l'échelle du cluster.

3. Exécutez une autre recherche :

```
GET dashboards_sample_data_flights/_search
{
  "query": {
    "match_all": {}
  }
}
```

Notez que tous les documents correspondants ont un champ `FlightDelay` ayant pour valeur `true`, un champ anonymisé `Dest` et aucun champ `FlightNum`.

4. Dans la fenêtre de votre navigateur d'origine, connecté en tant que `TheMasterUser`, choisissez Dev Tools (Outils de développement) et effectuez les mêmes recherches. Notez la différence entre les autorisations, le nombre d'accès, les documents correspondants et les champs inclus.

Validation de conformité pour Amazon OpenSearch Service

Des auditeurs tiers évaluent la sécurité et la conformité d'Amazon OpenSearch Service dans le cadre de plusieurs programmes de AWS conformité. Il s'agit notamment des programmes SOC, PCI et HIPAA.

Si vous avez des exigences de conformité, envisagez d'utiliser n'importe quelle version d' OpenSearch Elasticsearch 6.0 ou version ultérieure. Les versions antérieures d'Elasticsearch ne proposent pas de combinaison de chiffrement des données au repos et de node-to-node chiffrement et il est peu probable qu'elles répondent à vos besoins. Vous pouvez également envisager d'utiliser n'importe quelle version d' OpenSearch Elasticsearch 6.7 ou version ultérieure si un contrôle d'accès précis est important pour votre cas d'utilisation. Quoi qu'il en soit, le choix d'une version particulière OpenSearch ou d'Elasticsearch lors de la création d'un domaine ne garantit pas la conformité.

Pour savoir si un programme Services AWS de conformité Service AWS s'inscrit dans le champ d'application de programmes de conformité spécifiques, consultez Services AWS la section de conformité et sélectionnez le programme de conformité qui vous intéresse. Pour des informations générales, voir Programmes de AWS conformité Programmes AWS de .

Vous pouvez télécharger des rapports d'audit tiers à l'aide de AWS Artifact. Pour plus d'informations, voir Téléchargement de rapports dans AWS Artifact .

Votre responsabilité en matière de conformité lors de l'utilisation Services AWS est déterminée par la sensibilité de vos données, les objectifs de conformité de votre entreprise et les lois et réglementations applicables. AWS fournit les ressources suivantes pour faciliter la mise en conformité :

- Conformité et gouvernance de la sécurité : ces guides de mise en œuvre de solutions traitent des considérations architecturales et fournissent les étapes à suivre afin de déployer des fonctionnalités de sécurité et de conformité.
- Référence des services éligibles HIPAA : liste les services éligibles HIPAA. Tous ne Services AWS sont pas éligibles à la loi HIPAA.
- AWS Ressources de https://aws.amazon.com/compliance/resources/ de conformité — Cette collection de classeurs et de guides peut s'appliquer à votre secteur d'activité et à votre région.
- AWS Guides de conformité destinés aux clients — Comprenez le modèle de responsabilité partagée sous l'angle de la conformité. Les guides résument les meilleures pratiques en matière de sécurisation Services AWS et décrivent les directives relatives aux contrôles de sécurité dans de nombreux cadres (notamment le National Institute of Standards and Technology (NIST), le Payment Card Industry Security Standards Council (PCI) et l'Organisation internationale de normalisation (ISO)).
- Évaluation des ressources à l'aide des règles du guide du AWS Config développeur : le AWS Config service évalue dans quelle mesure les configurations de vos ressources sont conformes aux pratiques internes, aux directives du secteur et aux réglementations.
- AWS Security Hub— Cela Service AWS fournit une vue complète de votre état de sécurité interne AWS. Security Hub utilise des contrôles de sécurité pour évaluer vos ressources AWS et vérifier votre conformité par rapport aux normes et aux bonnes pratiques du secteur de la sécurité. Pour obtenir la liste des services et des contrôles pris en charge, consultez Référence des contrôles Security Hub.
- Amazon GuardDuty — Cela Service AWS détecte les menaces potentielles qui pèsent sur vos charges de travail Comptes AWS, vos conteneurs et vos données en surveillant votre environnement pour détecter toute activité suspecte et malveillante. GuardDuty peut vous aider

à répondre à diverses exigences de conformité, telles que la norme PCI DSS, en répondant aux exigences de détection des intrusions imposées par certains cadres de conformité.

- AWS Audit Manager— Cela vous Service AWS permet d'auditer en permanence votre AWS utilisation afin de simplifier la gestion des risques et la conformité aux réglementations et aux normes du secteur.

Résilience dans Amazon OpenSearch Service

L'infrastructure AWS mondiale d'repose sur les Régions AWS et les zones de disponibilité. Régions AWS Les fournissent plusieurs zones de disponibilité physiquement séparées et isolées, reliées par un réseau à latence faible, à débit élevé et à forte redondance. Avec les zones de disponibilité, vous pouvez concevoir et exploiter des applications et des bases de données qui basculent automatiquement d'une zone de disponibilité à l'autre sans interruption. Les zones de disponibilité sont plus hautement disponibles, tolérantes aux pannes et évolutives que les infrastructures traditionnelles à un ou plusieurs centres de données.

Pour de plus amples informations sur Régions AWS les et les zones de disponibilité, veuillez consulter Infrastructure AWS mondiale d'.

Outre l'infrastructure AWS globale, OpenSearch Service propose plusieurs fonctions qui contribuent à la prise en charge des vos besoins en matière de résilience et de sauvegarde de données :

- Domaines multi-AZ et partitions de réplica
- Instantanés manuels et automatiques

Authentification et autorisation JWT pour Amazon Service OpenSearch

Amazon OpenSearch Service vous permet désormais d'utiliser des jetons Web JSON (JWTs) pour l'authentification et l'autorisation. JWTs sont des jetons d'accès basés sur JSON utilisés pour accorder un accès par authentification unique (SSO). Vous pouvez utiliser JWTs in OpenSearch Service pour créer des jetons d'authentification unique afin de valider les demandes adressées à votre domaine OpenSearch de service. Pour l'utiliser JWTs, le contrôle d'accès détaillé doit être activé et vous devez fournir une clé publique valide au format RSA ou ECDSA PEM. Pour plus d'informations sur le contrôle d'accès détaillé, consultez la section Contrôle d'accès détaillé dans Amazon Service. OpenSearch

Vous pouvez configurer les jetons Web JSON à l'aide de la console de OpenSearch service, du AWS Command Line Interface (AWS CLI) ou du AWS SDKs.

Considérations

Avant de l'utiliser JWTs avec Amazon OpenSearch Service, vous devez prendre en compte les points suivants :

- En raison de la taille des clés publiques RSA au format PEM, nous vous recommandons d'utiliser la AWS console pour configurer l'authentification et l'autorisation JWT.
- Vous devez fournir des utilisateurs et des rôles valides lorsque vous spécifiez les champs de sujets et de rôles pour vous JWTs, sinon les demandes seront refusées.
- OpenSearch 2.11 est la première version compatible pouvant être utilisée pour l'authentification JWT.

Modification de la stratégie d'accès au domaine

Avant de configurer votre domaine pour utiliser l'authentification et l'autorisation JWT, vous devez mettre à jour votre politique d'accès au domaine afin de permettre aux utilisateurs de JWT d'accéder au domaine. Dans le cas contraire, toutes les demandes autorisées entrantes de JWT sont refusées. La politique d'accès au domaine recommandée pour fournir un accès complet aux sous-ressources (/*) est la suivante :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
      {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:ESHttp*",
      "Resource": "domain-arn/*"
    }
  ]
}
```

Configuration de l'authentification et de l'autorisation JWT

Vous pouvez activer l'authentification et l'autorisation JWT pendant le processus de création du domaine ou en mettant à jour un domaine existant. Les étapes de configuration varient légèrement en fonction de l'option choisie.

Les étapes suivantes expliquent comment configurer un domaine existant pour l'authentification et l'autorisation JWT dans la console de OpenSearch service :

1. Sous Configuration du domaine, accédez à Authentification et autorisation JWT pour OpenSearch, sélectionnez Activer l'authentification et l'autorisation JWT.
2. Configurez la clé publique à utiliser pour votre domaine. Pour ce faire, vous pouvez soit télécharger un fichier PEM contenant une clé publique, soit le saisir manuellement.

   > **Note**
   >
   > Si la clé téléchargée ou saisie n'est pas valide, un avertissement apparaît au-dessus de la zone de texte indiquant le problème.

3. (Facultatif) Sous Paramètres supplémentaires, vous pouvez configurer les champs facultatifs suivants

   - Clé d'objet : vous pouvez laisser ce champ vide pour utiliser la `sub` clé par défaut pour votre JWTs.
   - Clé des rôles : vous pouvez laisser ce champ vide pour utiliser la `roles` clé par défaut pour votre JWTs.

   Une fois que vous avez apporté vos modifications, enregistrez votre domaine.

Utiliser un JWT pour envoyer une demande de test

Après avoir créé un nouveau JWT avec une paire sujet/rôle spécifiée, vous pouvez envoyer une demande de test. Pour ce faire, utilisez la clé privée pour signer votre demande via l'outil qui a créé le JWT. OpenSearch Le service est en mesure de valider la demande entrante en vérifiant cette signature.

> **Note**
>
> Si vous avez spécifié une clé de sujet ou une clé de rôle personnalisée pour votre JWT, vous devez utiliser les noms de réclamation corrects pour votre JWT.

Voici un exemple d'utilisation d'un jeton JWT pour accéder au OpenSearch service via le point de terminaison de recherche de votre domaine :

```
curl -XGET "$search_endpoint" -H "Authorization: Bearer <JWT>"
```

Configuration de l'authentification et de l'autorisation JWT ()AWS CLI

La AWS CLI commande suivante active l'authentification et l'autorisation JWT à OpenSearch condition que le domaine existe :

```
aws opensearch update-domain-config --domain-name <your_domain_name> --advanced-
security-options '{"JWTOptions":{"Enabled":true, "PublicKey": "<your_public_key>",
 "SubjectKey": "<your_subject_key>", "RolesKey": "<your_roles_key>"}}'
```

Configuration de l'authentification et de l'autorisation JWT (configuration via API)

La demande suivante adressée à l'API de configuration active l'authentification et l'autorisation JWT OpenSearch sur un domaine existant :

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/config
{
  "AdvancedSecurityOptions": {
    "JWTOptions": {
      "Enabled": true,
      "PublicKey": "public-key",
      "RolesKey": "optional-roles-key",
      "SubjectKey": "optional-subject-key"
   }
  }
}
```

Génération d'une paire de clés

JWTs Pour configurer votre OpenSearch domaine, vous devez fournir une clé publique au format PEM (Privacy-Enhanced Mail). Amazon OpenSearch Service prend actuellement en charge deux algorithmes de chiffrement asymétriques lors de l'utilisation JWTs : RSA et ECDSA.

Pour créer une paire de clés RSA à l'aide de la bibliothèque openssl commune, procédez comme suit :

1. `openssl genrsa -out privatekey.pem 2048`
2. `openssl rsa -in privatekey.pem -pubout -out publickey.pem`

Dans cet exemple, le `publickey.pem` fichier contient la clé publique à utiliser avec Amazon OpenSearch Service, tandis que `privatekey.pem` le fichier privé permet de signer la clé JWTs envoyée au service. De plus, vous avez la possibilité de convertir la clé privée dans le `pkcs8` format couramment utilisé si vous en avez besoin pour générer votre JWTs.

Si vous utilisez le bouton de téléchargement pour ajouter un fichier PEM directement à la console, le fichier doit avoir une `.pem` extension, d'autres extensions de fichier telles que `.crt.cert`, ou ne `.key` sont pas prises en charge pour le moment.

Sécurité de l'infrastructure dans Amazon OpenSearch Service

En tant que service géré, Amazon OpenSearch Service bénéficie de la sécurité offerte par le réseau AWS mondial. Pour plus d'informations sur les services AWS de sécurité et la manière dont AWS protège l'infrastructure, consultez la section Sécurité du AWS cloud. Pour concevoir votre AWS environnement en utilisant les meilleures pratiques en matière de sécurité de l'infrastructure, consultez la section Protection de l'infrastructure dans le Security Pillar Well-Architected Framework (Pilier de sécurité de l'infrastructure AWS Well-Architected

Vous utilisez les appels d'API AWS publiés par pour accéder au OpenSearch Service via le réseau. Les clients doivent prendre en charge les éléments suivants :

- Protocole TLS (Transport Layer Security). Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Ses suites de chiffrement PFS (Perfect Forward Secrecy) comme DHE (Ephemeral Diffie-Hellman) ou ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La plupart des systèmes modernes tels que Java 7 et les versions ultérieures prennent en charge ces modes.

En outre, les demandes doivent être signées à l'aide d'un ID de clé d'accès et d'une clé d'accès secrète associée à un principal IAM. Vous pouvez également utiliser AWS Security Token Service (AWS STS) pour générer des informations d'identification de sécurité temporaires et signer les demandes.

Vous utilisez les appels d'API AWS publiés par pour accéder à l'API de configuration du OpenSearch Service via le réseau. Pour configurer la version TLS minimale requise à accepter, spécifiez la valeur `TLSSecurityPolicy` dans les options de point de terminaison de domaine :

```
aws opensearch update-domain-config --domain-name my-domain --domain-endpoint-options
 '{"TLSSecurityPolicy": "Policy-Min-TLS-1-2-2019-07"}'
```

Pour plus d'informations, veuillez consulter la Référence des commandes de l'AWS CLI.

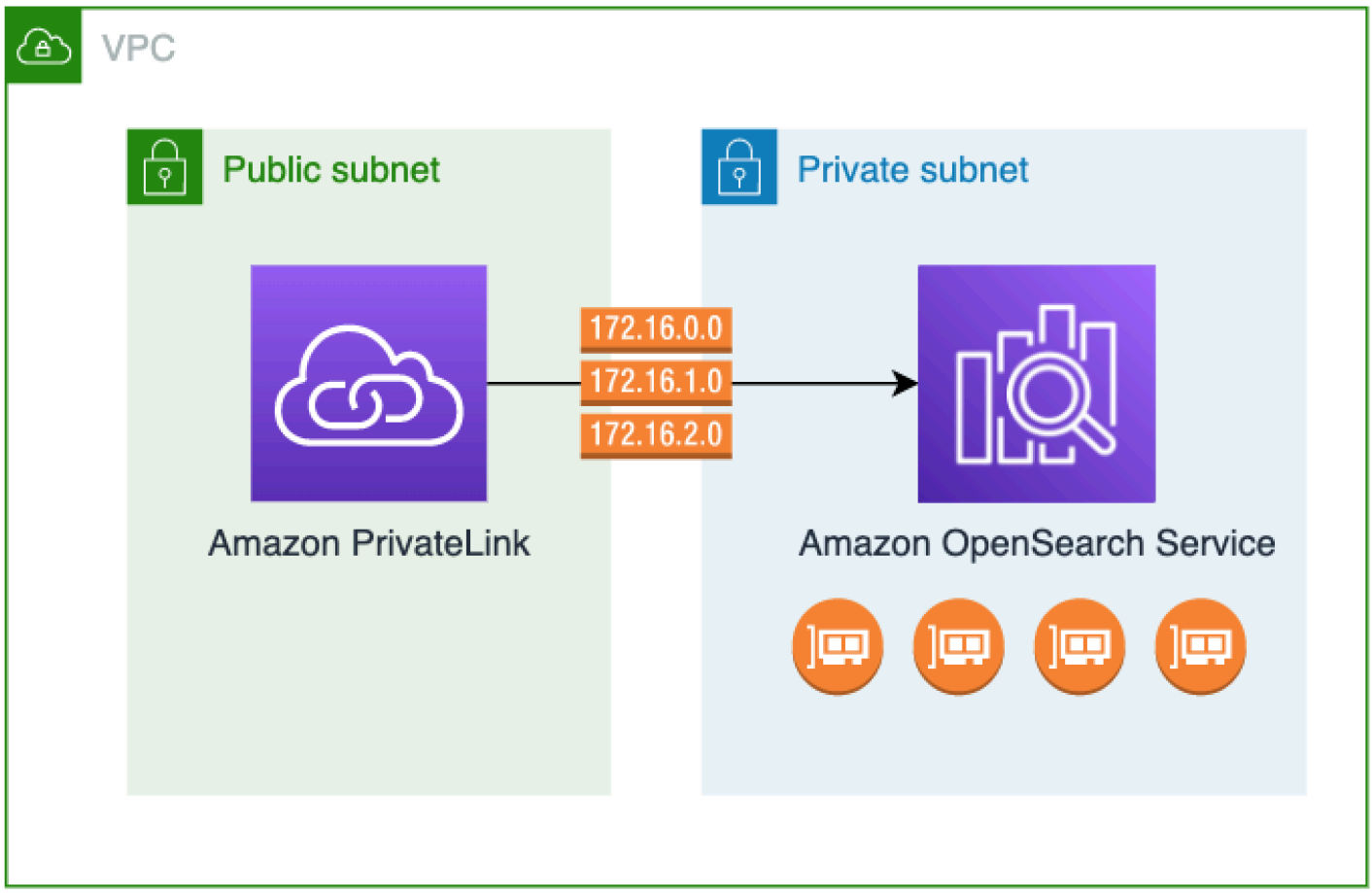
En fonction de la configuration de votre domaine, vous devrez peut-être également signer les demandes envoyées à OpenSearch APIs. Pour de plus amples informations, veuillez consulter the section called "Formulation et signature de demandes OpenSearch de service".

OpenSearch Le service prend en charge les domaines avec un accès public, qui peuvent recevoir des demandes de tout appareil connecté à Internet, ainsi que les domaines avec un accès VPC, qui sont isolés de l'Internet public.

Accéder à Amazon OpenSearch Service à l'aide d'un point de terminaison d'un OpenSearch VPC géré par le Service ()AWS PrivateLink

Vous pouvez accéder à un domaine Amazon OpenSearch Service en configurant un point de terminaison d'un OpenSearch VPC géré par le Service (à technologie). AWS PrivateLink Ces points de terminaison créent une connexion privée entre votre VPC et Amazon OpenSearch Service. Vous pouvez accéder aux domaines OpenSearch Service VPC comme s'ils se trouvaient dans votre VPC, sans passerelle Internet, périphérique NAT, connexion VPN ou connexion. AWS Direct Connect Les instances de votre VPC ne nécessitent pas d'adresses IP publiques pour accéder OpenSearch au Service.

Vous pouvez configurer les domaines OpenSearch de service pour exposer des points de terminaison supplémentaires s'exécutant sur des sous-réseaux publics ou privés au sein du même VPC, d'un VPC différent ou de différents. Comptes AWS Cela vous permet d'ajouter une couche de sécurité supplémentaire pour accéder à vos domaines, quel que soit leur emplacement d'exécution, sans aucune infrastructure à gérer. Le schéma suivant illustre les points de terminaison d' OpenSearch un VPC gérés par le Service au sein du même VPC :



Vous établissez cette connexion privée en créant un point de terminaison d'un VPC d' OpenSearch interface géré par le Service, à technologie. AWS PrivateLink Nous créons une interface réseau du point de terminaison dans chaque sous-réseau que vous activez pour le point de terminaison d'un VPC d'interface. Il s'agit d'interfaces réseau gérées par le service qui servent de point d'entrée pour le trafic destiné OpenSearch au Service. La tarification standard des points de terminaison d'AWS PrivateLink interface s'applique aux points de terminaison VPC gérés par le OpenSearch service facturés en vertu de. AWS PrivateLink

Vous pouvez créer des points de terminaison d'un VPC pour les domaines exécutant toutes les versions d' OpenSearch Elasticsearch héritées. Pour plus d'informations, consultez Accès aux Services AWS via AWS PrivateLink dans le Guide AWS PrivateLink .

Considérations et limitations relatives au OpenSearch Service

Avant de configurer un point de terminaison d'un VPC d'interface pour OpenSearch Service, consultez la section Accès à un AWS service à l'aide d'un point de terminaison d'un VPC d'interface pour Service dans le Guide.AWS PrivateLink

Lorsque vous utilisez des points de terminaison OpenSearch d'un VPC gérés par le Service, veuillez tenir compte des éléments suivants :

- Vous ne pouvez utiliser les points de terminaison d'un VPC d'interface que pour vous connecter à des domaines VPC. Les domaines publics ne sont pas pris en charge.
- Les points de terminaison d'un VPC ne peuvent se connecter qu'à des domaines au sein de la même Région AWS.
- Le protocole HTTPS est le seul protocole pris en charge pour les points de terminaison d'un VPC. Le protocole HTTP n'est pas autorisé.
- OpenSearch Le service prend en charge les appels vers toutes les opérations d' OpenSearch API prises en charge via un point de terminaison d'un VPC d'interface.
- Vous pouvez configurer jusqu'à 50 points de terminaison par compte et jusqu'à 10 points de terminaison par domaine. Un seul domaine peut disposer de 10 principaux autorisés au maximum.
- À l'heure actuelle, vous ne pouvez pas utiliser AWS CloudFormation pour créer des points de terminaison d'un VPC d'interface.
- Vous pouvez uniquement créer des points de terminaison d'un VPC d'interface via la console de OpenSearch service ou à l'aide de l'OpenSearch API de service. Vous ne pouvez pas créer de points de terminaison d'un VPC d'interface pour OpenSearch Service avec la console Amazon VPC.
- OpenSearch Les points de terminaison d'un VPC gérés par le service ne sont pas accessibles depuis Internet. Un point de terminaison de OpenSearch VPC géré par le Service est accessible uniquement au sein du VPC où le point de terminaison est alloué, ou au sein de tout VPC où le point de VPCs terminaison est alloué, comme le permettent les tables de routage et les groupes de sécurité.
- Les stratégies de point de terminaison d'un VPC ne sont pas prises en charge pour Service OpenSearch . Vous pouvez associer un groupe de sécurité aux interfaces réseau du point de terminaison afin de contrôler le trafic vers le OpenSearch Service via le point de terminaison d'un VPC d'interface.
- Votre rôle lié à un service doit figurer dans le même AWS compte que celui que vous utilisez pour créer le point de terminaison VPC.
- Pour créer, mettre à jour et supprimer le point de terminaison OpenSearch Service VPC, vous devez disposer des EC2 autorisations Amazon suivantes en plus de vos autorisations Amazon OpenSearch Service :
  - `ec2:CreateVpcEndpoint`

- `ec2:DescribeVpcEndpoints`
- `ec2:ModifyVpcEndpoint`
- `ec2:DeleteVpcEndpoints`
- `ec2:CreateTags`
- `ec2:DescribeTags`
- `ec2:DescribeSubnets`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeVpcs`

> **Note**
>
> Actuellement, vous ne pouvez pas limiter la création de points de terminaison VPC au OpenSearch service. Nous nous efforçons de rendre cela possible lors d'une future mise à jour.

Fournir un accès à un domaine

Si le VPC que vous souhaitez autoriser à accéder à votre domaine se trouve dans un autre Compte AWS, vous devez l'autoriser à partir du compte du propriétaire avant de pouvoir créer un point de terminaison d'un VPC d'interface.

Autoriser un VPC d'un autre à accéder Compte AWS à votre domaine

1. Ouvrez la console Amazon OpenSearch Service à l'adresse https://console.aws.amazon.com/aos/home/.
2. Dans le panneau de navigation, choisissez Domains (Domaines), puis ouvrez le domaine vers lequel vous souhaitez fournir un accès.
3. Accédez à l'onglet VPC endpoints (Points de terminaison d'un VPC) qui affiche les comptes et les correspondants VPCs ayant accès à votre domaine.
4. Choisissez Authorize principal (Autoriser le principal).
5. Saisissez l' Compte AWS ID du compte qui accèdera à votre domaine. Cette étape autorise le compte spécifié à créer des points de terminaison d'un VPC sur le domaine.
6. Choisissez Authorize (Autoriser).

Créer un point de terminaison d'un VPC d'interface pour un domaine VPC

Vous pouvez créer un point de terminaison d'un VPC d'interface pour OpenSearch Service avec la console OpenSearch Service ou l' AWS Command Line Interface ()AWS CLI.

Pour créer un point de terminaison d'un VPC d'interface pour un domaine de service OpenSearch

1. Ouvrez la console Amazon OpenSearch Service à l'adresse https://console.aws.amazon.com/aos/home/.
2. Dans le panneau de navigation de gauche, sélectionnez VPC endpoints (Points de terminaison d'un VPC).
3. Choisissez Créer un point de terminaison.
4. Choisissez si vous souhaitez connecter un domaine sur le actuel Compte AWS ou sur un autre Compte AWS.
5. Sélectionnez le domaine auquel vous vous connectez à l'aide de ce point de terminaison. Si le domaine se trouve sur le actuel Compte AWS, utilisez la liste déroulante pour le choisir. Si le domaine se trouve sur un autre compte, saisissez l'Amazon Resource Name (ARN) du domaine auquel vous connecter. Pour choisir un domaine sur un autre compte, le propriétaire doit vous donner accès au domaine.
6. Pour VPC, sélectionnez le VPC à partir duquel vous allez accéder à Service. OpenSearch
7. Pour Subnets (Sous-réseaux), sélectionnez un ou plusieurs sous-réseaux à partir desquels vous accéderez à OpenSearch Service.
8. Pour Security groups (Groupes de sécurité), sélectionnez les groupes de sécurité à associer aux interfaces réseau du point de terminaison. Il s'agit d'une étape essentielle au cours de laquelle vous devez limiter les ports, les protocoles et les sources de trafic entrant que vous autorisez dans votre point de terminaison. Les règles du groupe de sécurité doivent permettre aux ressources qui utiliseront le point de terminaison d'un VPC pour communiquer avec OpenSearch Service de communiquer avec l'interface réseau du point de terminaison.
9. Choisissez Créer un point de terminaison. Le point de terminaison sera actif au bout de deux à cinq minutes.

Utiliser les points de terminaison d'un OpenSearch VPC gérés par le Service à l'aide de l'API de configuration

Utilisez les opérations d'API suivantes pour créer et gérer des points de terminaison d'un OpenSearch VPC gérés par le Service.

- CreateVpcEndpoint
- ListVpcEndpoints
- UpdateVpcEndpoint
- DeleteVpcEndpoint

Utilisez les opérations d'API suivantes pour gérer l'accès des points de terminaison aux domaines VPC :

- AuthorizeVpcEndpointAccess
- ListVpcEndpointAccess
- ListVpcEndpointsForDomain
- RevokeVpcEndpointAccess

Authentification SAML pour les tableaux de bord OpenSearch

L'authentification SAML pour les OpenSearch tableaux de bord vous permet d'utiliser votre fournisseur d'identité existant pour proposer l'authentification unique (SSO) pour les tableaux de bord sur les domaines Amazon OpenSearch Service exécutant Elasticsearch 6.7 OpenSearch ou version ultérieure. Pour utiliser l'authentification SAML, vous devez activer le contrôle précis des accès.

Plutôt que de vous authentifier via Amazon Cognito ou la base de données utilisateur interne, l'authentification SAML OpenSearch pour les tableaux de bord vous permet de faire appel à des fournisseurs d'identité tiers pour vous connecter aux tableaux de bord, gérer un contrôle d'accès précis, effectuer des recherches dans vos données et créer des visualisations. OpenSearch Le service prend en charge les fournisseurs qui utilisent la norme SAML 2.0, tels qu'Okta, Keycloak, Active Directory Federation Services (ADFS), Auth0 et. AWS IAM Identity Center

L'authentification SAML pour les tableaux de bord permet uniquement d'accéder aux OpenSearch tableaux de bord via un navigateur Web. Vos informations d'identification SAML ne vous permettent pas de faire des requêtes HTTP directes vers les tableaux de bord OpenSearch ou les tableaux APIs de bord.

Présentation de la configuration SAML

Cette documentation suppose que vous disposez d'un fournisseur d'identité existant avec lequel vous êtes familier. Nous ne pouvons pas fournir d'étapes de configuration détaillées pour votre fournisseur exact, uniquement pour votre domaine OpenSearch de service.

Le flux de connexion OpenSearch aux tableaux de bord peut prendre l'une des deux formes suivantes :

- Fournisseur de services initié : vous accédez aux Tableaux de bord (par exemple, `https://my-domain.us-east-1.es.amazonaws.com/_dashboards`) , ce qui vous redirige vers l'écran de connexion. Une fois connecté, le fournisseur d'identité vous redirige vers Tableaux de bord.
- Initié par le fournisseur d'identité (IdP) : vous accédez à votre fournisseur d'identité, vous vous connectez et choisissez OpenSearch Dashboards dans un répertoire d'applications.

OpenSearch Le service fournit deux connexions uniques URLs, initiées par le SP et initiées par l'IdP, mais vous n'avez besoin que de celle qui correspond au flux de connexion aux tableaux de bord que vous souhaitez. OpenSearch

Quel que soit le type d'authentification que vous utilisez, l'objectif consiste à vous connecter via votre fournisseur d'identité et à recevoir une assertion SAML contenant votre nom d'utilisateur (obligatoire) et un rôle backend(facultatif, mais recommandé). Cette information permet au contrôle précis des accèsd'affecter des autorisations aux utilisateurs SAML. Dans les fournisseurs d'identité externes, les rôles backend sont généralement appelés « rôles » ou « groupes ».

Considérations

Prenez en compte les éléments suivants lorsque vous configurez l'authentification SAML :

- En raison de la taille du fichier de métadonnées IdP, nous vous recommandons vivement d'utiliser la AWS console pour configurer l'authentification SAML.
- Les domaines ne prennent en charge qu'une seule méthode d'authentification Tableaux de bord à la fois. Si l'authentification Amazon Cognito pour les OpenSearch tableaux de bord est activée, vous devez la désactiver avant de pouvoir activer l'authentification SAML.
- Si vous utilisez un équilibreur de charge réseau avec SAML, vous devez d'abord créer un point de terminaison personnalisé. Pour de plus amples informations, veuillez consulter ???.

- Les politiques de contrôle des services (SCP) ne seront pas applicables ni évaluées dans le cas d'identités non IAM (comme le SAML dans OpenSearch Amazon Serverless et SAML et l'autorisation utilisateur interne de base pour Amazon Service). OpenSearch

Authentification SAML pour les domaines VPC

SAML ne nécessite pas de communication directe entre votre fournisseur d'identité et votre fournisseur de services. Par conséquent, même si votre OpenSearch domaine est hébergé dans un VPC privé, vous pouvez toujours utiliser le protocole SAML tant que votre navigateur peut communiquer à la fois avec votre OpenSearch cluster et avec votre fournisseur d'identité. Votre navigateur joue essentiellement le rôle d'intermédiaire entre votre fournisseur d'identité et votre fournisseur de services. Pour un diagramme utile qui explique le flux d'authentification SAML, consultez la documentation d'Okta.

Modification de la stratégie d'accès au domaine

Avant de configurer l'authentification SAML, vous devez mettre à jour la stratégie d'accès au domaine afin d'autoriser les utilisateurs SAML à y accéder. Sinon, vous recevrez des erreurs d'accès refusé.

Nous recommandons la stratégie d'accès au domaine suivante, qui fournit un accès complet aux sous-ressources (`/*`) du domaine :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:ESHttp*",
      "Resource": "domain-arn/*"
    }
  ]
}
```

Pour rendre la politique plus restrictive, vous pouvez y ajouter une condition d'adresse IP. Cette condition limite l'accès uniquement à la plage d'adresses IP ou au sous-réseau spécifié. Par exemple, la politique suivante autorise l'accès uniquement depuis le sous-réseau 192.0.2.0/24 :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Condition": {
        "IpAddress": {
          "aws:SourceIp": [
            "192.0.2.0/24"
          ]
        }
      },
      "Resource": "domain-arn/*"
    }
  ]
}
```

> **Note**
>
> Une politique d'accès aux domaines ouverts nécessite l'activation d'un contrôle d'accès précis sur votre domaine. Dans le cas contraire, le message d'erreur suivant s'affiche : `To protect domains with public access, a restrictive policy or fine-grained access control is required.`
> Si vous avez un utilisateur principal ou un utilisateur interne configuré avec un mot de passe robuste, il peut être acceptable de maintenir la politique ouverte tout en utilisant un contrôle

d'accès précis du point de vue de la sécurité. Pour de plus amples informations, veuillez consulter ???.

Configuration de l'authentification initiée par le fournisseur de services ou le fournisseur d'identité

Ces étapes expliquent comment activer l'authentification SAML avec une authentification initiée par le SP ou par l'IdP pour les tableaux de bord. OpenSearch Pour l'étape supplémentaire nécessaire pour activer les deux, consultez Activer l'authentification initiée par le SP et l'IdP.

Étape 1 : activer l'authentification SAML

Vous pouvez activer l'authentification SAML soit lors de la création du domaine, soit en choisissant Actions, Edit security configuration (Modifier la configuration de sécurité) sur un domaine existant. Les étapes suivantes varient légèrement en fonction de celle que vous choisissez.

Dans la configuration du domaine, sous Authentification SAML pour les OpenSearch tableaux de bords/Kibana, sélectionnez Activer l'authentification SAML.

Étape 2 : configurer votre fournisseur d'identité

Suivez les étapes suivantes en fonction du moment où vous configurez l'authentification SAML.

En cas de création d'un nouveau domaine

Si vous êtes en train de créer un nouveau domaine, OpenSearch Service ne peut pas encore générer d'identifiant d'entité ou de SSO URLs pour le fournisseur de services. Votre fournisseur d'identité a besoin de ces valeurs afin d'activer correctement l'authentification SAML, mais elles ne peuvent être générées qu'après la création du domaine. Pour contourner cette interdépendance lors de la création du domaine, vous pouvez fournir des valeurs temporaires dans votre configuration IdP afin de générer les métadonnées requises, puis les mettre à jour une fois que votre domaine est actif.

Si vous utilisez un point de terminaison personnalisé, vous pouvez en déduire ce qu'il URLs sera. Par exemple, si votre point de terminaison personnalisé est `www.custom-endpoint.com`, l'ID d'entité du fournisseur de services sera `www.custom-endpoint.com`, l'adresse URL SSO initiée par l'IdP sera `www.custom-endpoint.com/_dashboards/_opendistro/_security/saml/acs/idpinitiated` et l'adresse URL SSO initiée par le SP sera `www.custom-endpoint.com/`

`_dashboards/_opendistro/_security/saml/acs`. Vous pouvez utiliser ces valeurs pour configurer votre fournisseur d'identité avant la création du domaine. Consultez la section suivante pour examiner des exemples.

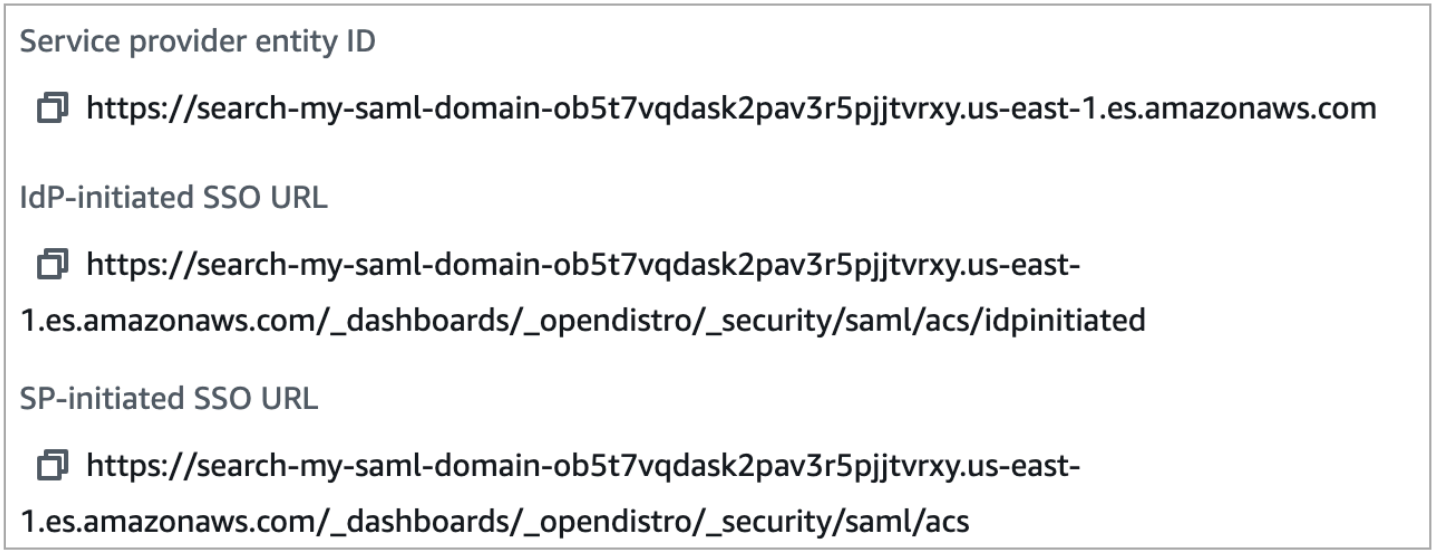
> **Note**
>
> Vous ne pouvez pas vous connecter avec un point de terminaison à double pile car le nom de domaine complet d'une requête HTTP est différent du nom de domaine complet d'une demande SAML. Un OpenSearch administrateur devra configurer un point de terminaison personnalisé et définir la valeur CNAME sur un point de terminaison à double pile si vous souhaitez vous connecter à l'aide d'un point de terminaison à double pile.

Si vous n'utilisez pas de point de terminaison personnalisé, vous pouvez saisir des valeurs temporaires dans votre IdP pour générer les métadonnées requises, puis les mettre à jour ultérieurement une fois le domaine actif.

Par exemple, dans Okta, vous pouvez saisir `https://temp-endpoint.amazonaws.com` dans les champs Single sign on URL (Adresse URL de l'authentification unique) et Audience URI (SP Entity ID) (URI de l'audience (ID d'entité du SP)), ce qui vous permet de générer les métadonnées. Ensuite, une fois le domaine actif, vous pouvez récupérer les valeurs correctes auprès de OpenSearch Service et les mettre à jour dans Okta. Pour obtenir des instructions, veuillez consulter the section called “Étape 6 : mettez à jour votre IdP URLs”.

En cas de modification d'un domaine existant

Si vous activez l'authentification SAML sur un domaine existant, copiez l'ID d'entité du fournisseur de services et l'un des URLs SSO. Pour obtenir des conseils sur l'adresse URL à utiliser, consultez the section called “Présentation de la configuration SAML”.



Utilisez les valeurs pour configurer votre fournisseur d'identité. Il s'agit-là de la partie la plus complexe du processus, et malheureusement, la terminologie de même que la procédure changent considérablement d'un fournisseur à l'autre. Consultez la documentation de votre fournisseur.

Dans Okta, par exemple, vous créez une application Web SAML 2.0. Pour Single sign on URL (Adresse URL de l'authentification unique), spécifiez l'adresse URL SSO. PourURI du public ciblé (ID d'entité du fournisseur de services), spécifiez l'ID d'entité du fournisseur de services.

Plutôt que d'utilisateurs et de rôles backend, Okta parle d'utilisateurs et de groupes. Pour Group Attribute Statements (Instructions d'attributs de groupe), nous vous recommandons d'ajouter `role` au champ Name (Nom) et l'expression régulière `.+` au champ Filter (Filtrer). Cette instruction indique au fournisseur d'identité Okta d'inclure tous les groupes d'utilisateurs sous le champ `role` de l'assertion SAML après l'authentification d'un utilisateur.

Dans IAM Identity Center, vous spécifiez l'ID de l'entité SP en tant qu'audience SAML de l'application. Vous devez également spécifier les mappages d'attributs suivants : `Subject=${user:subject}:format=unspecified` et`Role=${user:groups}:format=uri`.

Dans Auth0, vous créez une application Web normale et activez le module complémentaire SAML 2.0. Dans Keycloak, vous créez un client.

Étape 3 : importer les métadonnées IdP

Une fois votre fournisseur d'identité configuré, il génère un fichier de métadonnées de fournisseur d'identité. Ce fichier XML contient des informations sur le fournisseur, telles qu'un certificat TLS, des points de terminaison d'authentification unique et l'ID d'entité du fournisseur d'identité.

Copiez le contenu du fichier de métadonnées IdP et collez-le dans le champ Métadonnées depuis l'IdP de la console de service. OpenSearch Vous pouvez également choisir Importer depuis un fichier XML, puis charger le fichier. Le fichier de métadonnées doit se présenter comme suit :

```
<?xml version="1.0" encoding="UTF-8"?>
<md:EntityDescriptor entityID="entity-id"
 xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata">
  <md:IDPSSODescriptor WantAuthnRequestsSigned="false"
 protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
    <md:KeyDescriptor use="signing">
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:X509Data>
          <ds:X509Certificate>tls-certificate</ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>
    </md:KeyDescriptor>
    <md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified</
md:NameIDFormat>
    <md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress</
md:NameIDFormat>
    <md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
 Location="idp-sso-url"/>
    <md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
Redirect" Location="idp-sso-url"/>
  </md:IDPSSODescriptor>
</md:EntityDescriptor>
```

Étape 4 : configurer les champs SAML

Après avoir saisi les métadonnées de votre IdP, configurez les champs supplémentaires suivants dans la console de OpenSearch service :

- IdP entity ID (Identifiant d'entité du IdP) : copiez la valeur de la propriété `entityID` depuis votre fichier de métadonnées et collez-la dans ce champ. De nombreux fournisseurs d'identité affichent également cette valeur dans le cadre d'un résumé post-configuration. Certains fournisseurs l'appellent « auteur ».
- Nom d'utilisateur principal SAML et rôle principal de backend SAML : le rôle d'utilisateur and/or principal que vous spécifiez reçoit des autorisations complètes sur le cluster, équivalentes à celles d'un nouvel utilisateur principal, mais ne peut utiliser ces autorisations que dans les tableaux de bord. OpenSearch

Dans Okta, par exemple, il peut s'agir d'un utilisateur `jdoe`qui appartient au groupe`admins`. Si vous ajoutez `jdoe` au champ Nom d'utilisateur principal SAML, seul cet utilisateur reçoit des autorisations complètes. Si vous ajoutez `admins` au champ rôle backend principal SAML, tout utilisateur appartenant au groupe `admins` reçoit des autorisations complètes.

> **Note**
>
> Le contenu de l'assertion SAML doit correspondre exactement aux chaînes que vous utilisez pour le nom d'utilisateur principal SAML et le rôle principal SAML. Certains fournisseurs d'identité ajoutent un préfixe avant leur nom d'utilisateur, ce qui peut entraîner une hard-to-diagnose incompatibilité. Dans l'interface utilisateur du fournisseur d'identité, vous pouvez voir `jdoe`, mais l'assertion SAML peut contenir `auth0|jdoe`. Utilisez toujours la chaîne de l'assertion SAML.

De nombreux fournisseurs d'identité vous permettent d'afficher un exemple d'assertion lors du processus de configuration, et des outils tels que SAML-tracer peuvent vous aider à examiner et à résoudre le contenu des assertions. Les assertions se présentent comme suit :

```
<?xml version="1.0" encoding="UTF-8"?>
<saml2:Assertion ID="id67229299299259351343340162"
 IssueInstant="2020-09-22T22:03:08.633Z" Version="2.0"
  xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">
  <saml2:Issuer Format="urn:oasis:names:tc:SAML:2.0:nameid-format:entity">idp-issuer</
saml2:Issuer>
  <saml2:Subject>
    <saml2:NameID Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">username</saml2:NameID>
    <saml2:SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:bearer">
      <saml2:SubjectConfirmationData NotOnOrAfter="2020-09-22T22:08:08.816Z"
 Recipient="domain-endpoint/_dashboards/_opendistro/_security/saml/acs"/>
    </saml2:SubjectConfirmation>
  </saml2:Subject>
  <saml2:Conditions NotBefore="2020-09-22T21:58:08.816Z"
 NotOnOrAfter="2020-09-22T22:08:08.816Z">
    <saml2:AudienceRestriction>
      <saml2:Audience>domain-endpoint</saml2:Audience>
    </saml2:AudienceRestriction>
  </saml2:Conditions>
  <saml2:AuthnStatement AuthnInstant="2020-09-22T19:54:37.274Z">
```

```
    <saml2:AuthnContext>

 <saml2:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport<
saml2:AuthnContextClassRef>
    </saml2:AuthnContext>
  </saml2:AuthnStatement>
  <saml2:AttributeStatement>
    <saml2:Attribute Name="role" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:unspecified">
      <saml2:AttributeValue
        xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:type="xs:string">GroupName Match Matches regex ".+" (case-sensitive)
      </saml2:AttributeValue>
    </saml2:Attribute>
  </saml2:AttributeStatement>
</saml2:Assertion>
```

Étape 5 : (Facultatif) configurer des paramètres supplémentaires

Sous Additional settings (Paramètres supplémentaires), configurez les champs facultatifs suivants :

- Subject key (Clé d'objet) : vous pouvez laisser ce champ vide pour utiliser l'élément `NameID` de l'assertion SAML pour le nom d'utilisateur. Si votre assertion n'utilise pas cet élément standard et inclut plutôt le nom d'utilisateur comme attribut personnalisé, spécifiez cet attribut ici.
- Roles key (Clé de rôles) : si vous voulez utiliser des rôles backend (recommandé), spécifiez un attribut de l'assertion dans ce champ, tel que `role` ou `group`. Là encore, un outil tel que SAML-tracer peut vous être utile.
- Durée de vie de la session : par défaut, OpenSearch Dashboards déconnecte les utilisateurs au bout de 24 heures. Vous pouvez configurer cette valeur sur n'importe quel nombre compris entre 60 et 1 440 (24 heures) en spécifiant une nouvelle valeur.

Une fois que la configuration vous convient, enregistrez le domaine.

Étape 6 : mettez à jour votre IdP URLs

Si vous avez activé l'authentification SAML lors de la création d'un domaine, vous deviez spécifier une URLs valeur temporaire dans votre IdP afin de générer le fichier de métadonnées XML. Une fois que le statut du domaine est `Active` passé à, vous pouvez obtenir le bon URLs identifiant et modifier votre IdP.

Pour les récupérer URLs, sélectionnez le domaine et choisissez Actions, Modifier la configuration de sécurité. Dans le cadre de l'authentification SAML pour OpenSearch Dashboards/Kibana, vous pouvez trouver l'ID d'entité et le SSO du fournisseur de services corrects. URLs Copiez les valeurs et utilisez-les pour configurer votre fournisseur d'identité, en remplaçant le temporaire URLs que vous avez fourni à l'étape 2.

Étape 7 : associer les utilisateurs SAML aux rôles

Une fois que le statut de votre domaine est actif et que votre IdP est correctement configuré, accédez à OpenSearch Tableaux de bord.

- Si vous avez choisi l'URL initiée par le fournisseur de services, accédez à *domain-endpoint*`/_dashboards`. Pour vous connecter directement à un locataire spécifique, vous pouvez ajouter `?security_tenant=`*tenant-name* à l'adresse URL.
- Si vous avez choisi l'URL initiée par le fournisseur d'identité, accédez au répertoire d'applications de votre fournisseur d'identité.

Dans les deux cas, connectez-vous en tant qu'utilisateur principal SAML ou en tant qu'utilisateur appartenant au rôle backend SAML. Pour continuer l'exemple de l'étape 7, connectez-vous en tant que `jdoe` ou un membre du groupe `admins`.

Une fois OpenSearch les tableaux de bord chargés, choisissez Sécurité, Rôles. Mappez ensuite les rôles pour permettre aux autres utilisateurs d'accéder aux OpenSearch tableaux de bord.

Par exemple, vous pouvez mapper un collègue de confiance `jroe` aux rôles `all_access` et `security_manager`. Vous pouvez également mapper le rôle backend `analysts` aux rôles `readall` et `opensearch_dashboards_user`.

Si vous préférez utiliser l'API plutôt que les OpenSearch tableaux de bord, consultez l'exemple de demande suivant :

```
PATCH _plugins/_security/api/rolesmapping
[
  {
    "op": "add", "path": "/security_manager", "value": { "users": ["master-user",
 "jdoe", "jroe"], "backend_roles": ["admins"] }
  },
  {
    "op": "add", "path": "/all_access", "value": { "users": ["master-user", "jdoe",
 "jroe"], "backend_roles": ["admins"] }
```

```
  },
  {
    "op": "add", "path": "/readall", "value": { "backend_roles": ["analysts"] }
  },
  {
    "op": "add", "path": "/opensearch_dashboards_user", "value": { "backend_roles":
 ["analysts"] }
  }
]
```

Configuration de l'authentification initiée à la fois par le SP et l'IdP

Si vous souhaitez configurer l'authentification initiée par le fournisseur de services et le fournisseur d'identité, vous devez le faire via votre fournisseur d'identité. Par exemple, dans Okta, vous pouvez effectuer les étapes suivantes :

1. Dans votre application SAML, accédez à General (Général), SAML settings (Paramètres SAML).
2. Pour Single sign on URL (URL d'authentification unique), fournissez votre URL SSO initiée par l'IdP. Par exemple, `https://search-`*`domain-hash`*`/_dashboards/_opendistro/_security/saml/acs/`*`idpinitiated`*.
3. Activez Autoriser cette application à demander un autre SSO URLs.
4. Sous SSO requestable URLs, ajoutez un ou plusieurs SSO initiés par le SP. URLs Par exemple, `https://search-`*`domain-hash`*`/_dashboards/_opendistro/_security/saml/`*`acs`*.

Configuration de l'authentification SAML (AWS CLI)

La AWS CLI commande suivante active l'authentification SAML pour les OpenSearch tableaux de bord sur un domaine existant :

```
aws opensearch update-domain-config \
  --domain-name my-domain \
  --advanced-security-options '{"SAMLOptions":{"Enabled":true,"MasterUserName":"my-
idp-user","MasterBackendRole":"my-idp-group-or-role","Idp":{"EntityId":"entity-
id","MetadataContent":"metadata-content-with-quotes-escaped"},"RolesKey":"optional-
roles-key","SessionTimeoutMinutes":180,"SubjectKey":"optional-subject-key"}}'
```

Vous devez utiliser une séquence d'échappement sur tous les guillemets et caractères de nouvelle ligne dans le fichier XML des métadonnées. Par exemple, utilisez `<KeyDescriptor use=`

\"signing\">\n plutôt que <KeyDescriptor use="signing"> et un saut de ligne. Pour plus d'informations sur l'utilisation de la AWS CLI, consultezRéférence des commandes AWS CLI.

Configuration de l'authentification SAML (API de configuration)

La demande suivante adressée à l'API de configuration active l'authentification SAML pour les OpenSearch tableaux de bord sur un domaine existant :

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/domain/my-domain/config
{
  "AdvancedSecurityOptions": {
    "SAMLOptions": {
      "Enabled": true,
      "MasterUserName": "my-idp-user",
      "MasterBackendRole": "my-idp-group-or-role",
      "Idp": {
        "EntityId": "entity-id",
        "MetadataContent": "metadata-content-with-quotes-escaped"
      },
      "RolesKey": "optional-roles-key",
      "SessionTimeoutMinutes": 180,
      "SubjectKey": "optional-subject-key"
    }
  }
}
```

Vous devez utiliser une séquence d'échappement sur tous les guillemets et caractères de nouvelle ligne dans le fichier XML des métadonnées. Par exemple, utilisez <KeyDescriptor use= \"signing\">\n plutôt que <KeyDescriptor use="signing"> et un saut de ligne. Pour obtenir des informations détaillées sur l'utilisation de l'API de configuration, consultez la référence de l'API de OpenSearch service.

Résolution des problèmes SAML

Erreur	Détails
Votre demande : « */some/path* » n'est pas autorisée.	Vérifiez que vous avez fourni l'URL d'authentification unique qui convient (étape 3) à votre fournisseur d'identité.

Erreur	Détails
Fournissez un document de métadonnées de fournisseur d'identité valide pour activer SAML.	Le fichier de métadonnées de votre fournisseur d'identité n'est pas conforme à la norme SAML 2.0. Recherchez les erreurs à l'aide d'un outil de validation.
Les options de configuration SAML ne sont pas visibles dans la console.	Procédez à une mise à jour vers la dernière version du logiciel de service.
Erreur de configuration SAML : un problème est survenu lors de la récupération de la configuration SAML, vérifiez vos paramètres.	Cette erreur générique peut se produire pour de nombreuses raisons. • Vérifiez que vous avez fourni à votre fournisseur d'identité l'ID d'entité de fournisseur de services et l'URL d'authentification unique qui conviennent. • Générez à nouveau le fichier de métadonnées du fournisseur d'identité et vérifiez son l'ID d'entité. Ajoutez les métadonnées mises à jour dans la console AWS . • Vérifiez que votre politique d'accès au domaine autorise l'accès aux OpenSearch tableaux de bord et `_plugins/_security/*` . En général, nous recommandons une stratégie d'accès ouverte pour les domaines qui utilisent un contrôle précis des accès. • Consultez la documentation de votre fournisseur d'identité pour connaître la procédure de configuration de SAML.

Erreur	Détails
Rôle manquant : aucun rôle n'est disponible pour cet utilisateur, contactez votre administrateur système.	Vous vous êtes authentifié avec succès, mais le nom d'utilisateur et les rôles backend de l'assertion SAML ne sont mappés à aucun rôle et ne disposent donc aucune autorisation. Ces mappages sont sensibles à la casse. Votre administrateur système peut vérifier le contenu de votre assertion SAML à l'aide d'un outil tel que SAML-Tracer, puis vérifier le mappage de vos rôles à l'aide de la requête suivante : `GET _plugins/_security/api/rolesmapping`
Votre navigateur redirige ou reçoit en permanence des erreurs HTTP 500 lorsqu'il essaie d'accéder aux OpenSearch tableaux de bord.	Ces erreurs peuvent se produire si votre assertion SAML contient un grand nombre de rôles totalisant approximativement 1 500 caractères. Par exemple, si vous transmettez 80 rôles, dont la longueur moyenne est de 20 caractères, vous pouvez dépasser la limite de taille des cookies dans votre navigateur Web. À partir de OpenSearch la version 2.7, l'assertion SAML prend en charge les rôles de 5 000 caractères maximum.
Vous ne pouvez pas vous déconnecter d'ADFS.	ADFS exige que toutes les demandes de déconnexi on soient signées, ce que le OpenSearch service ne prend pas en charge. `<SingleLogoutService /` `>`Supprimez-le du fichier de métadonnées IdP pour obliger le OpenSearch service à utiliser son propre mécanisme de déconnexion interne.

Erreur	Détails
`Could not find entity descriptor for __PATH__.`	L'ID d'entité de l'IdP fourni dans les métadonnées XML à OpenSearch Service est différent de celui indiqué dans la réponse SAML. Pour résoudre ce problème, assurez-vous qu'ils correspondent. Activez les journaux d'erreurs d'application CW sur votre domaine pour trouver le message d'erreur permettant de résoudre le problème d'intégration SAML.
`Signature validation failed. SAML response rejected.`	OpenSearch Le service n'est pas en mesure de vérifier la signature dans la réponse SAML à l'aide du certificat de l'IdP fourni dans les métadonnées XML. Il peut s'agir d'une erreur manuelle ou d'une rotation de certificat de votre IdP. Mettez à jour le dernier certificat de votre IdP dans les métadonnées XML fournies au OpenSearch Service via le. AWS Management Console
`__PATH__ is not a valid audience for this response.`	Le champ d'audience de la réponse SAML ne correspond pas au point de terminaison du domaine. Pour corriger cette erreur, mettez à jour le champ d'audience SP pour qu'il corresponde au point de terminaison de votre domaine. Si vous avez activé les points de terminaison personnalisés, le champ d'audience doit correspondre à votre point de terminaison personnalisé. Activez les journaux d'erreurs d'application CW sur votre domaine pour trouver le message d'erreur permettant de résoudre le problème d'intégration SAML.
Votre navigateur reçoit une erreur HTTP 400 `Invalid Request Id` dans la réponse.	Cette erreur se produit généralement si vous avez configuré l'URL initiée par l'IdP avec le format. *`<DashboardsURL>`* `/_opendistro/_security/saml/acs` Configurez plutôt l'URL avec le format*`<DashboardsURL>`* `/_opendistro/_security/saml/acs/idpinitiated` .

Erreur	Détails
La réponse a été reçue à la __PATH__ place de__PATH__.	Le champ de destination de la réponse SAML ne correspond à aucun des formats d'URL suivants : • *<DashboardsURL>* /_opendistro/_security/saml/acs • *<DashboardsURL>* /_opendistro/_security/saml/acs/idpinitiated . Selon le flux de connexion que vous utilisez (initié par le SP ou par l'IDP), entrez dans un champ de destination correspondant à l'un des. OpenSearch URLs
La réponse comporte un InResponseTo attribut, alors qu'aucun attribut n'InResponseTo était attendu.	Vous utilisez l'URL initiée par l'IdP pour un flux de connexion initié par le SP. Utilisez plutôt l'URL initiée par le SP.

Désactivation de l'authentification SAML

Pour désactiver l'authentification SAML pour les OpenSearch tableaux de bord (console)

1. Choisissez le domaine, Actions, et Edit security configuration(Modifier la configuration de la sécurité).
2. Décochez Activer l'authentification SAML.
3. Sélectionnez Enregistrer les modifications.
4. Une fois le traitement terminé, vérifiez le mappage de rôles du contrôle précis des accès à l'aide de la demande suivante :

   ```
   GET _plugins/_security/api/rolesmapping
   ```

   La désactivation de l'authentification SAML pour les tableaux de bord ne supprime pas les mappages entre le nom d'utilisateur and/or principal SAML et le rôle principal principal SAML. Pour supprimer ces mappages, connectez-vous aux Tableaux de bord à l'aide de la base de données utilisateur interne (si elle est activée) ou utilisez l'API pour les supprimer :

```
PUT _plugins/_security/api/rolesmapping/all_access
{
  "users": [
    "master-user"
  ]
}
```

Support de propagation d'identité fiable d'IAM Identity Center pour Amazon Service OpenSearch

Vous pouvez désormais utiliser les principaux de votre centre d'identité AWS IAM configuré de manière centralisée (utilisateurs et groupes) via Trusted Identity Propagation pour accéder aux OpenSearch domaines via des applications de OpenSearch service. Pour activer la prise en charge d'IAM Identity Center pour Amazon OpenSearch Service, vous devez activer l'utilisation d'IAM Identity Center. Pour en savoir plus sur la procédure à suivre, consultez Qu'est-ce qu'IAM Identity Center ? . Voir Comment associer un OpenSearch domaine en tant que source de données dans les OpenSearch applications ? pour plus de détails.

Vous pouvez configurer IAM Identity Center à l'aide de la console de OpenSearch service, du AWS Command Line Interface (AWS CLI) ou du AWS SDKs.

> **Note**
>
> Les principaux centres d'identité IAM ne sont pas pris en charge par le biais de tableaux de bord (situés au même endroit que le cluster). Ils ne sont pris en charge que via une interface OpenSearch utilisateur centralisée (tableaux de bord).

Considérations

Avant d'utiliser IAM Identity Center avec Amazon OpenSearch Service, vous devez prendre en compte les points suivants :

- Le centre d'identité IAM est activé dans le compte.
- La version du OpenSearch domaine est 1.3 ou ultérieure.
- Le contrôle d'accès détaillé est activé sur le domaine.

- Le domaine doit se trouver dans la même région que l'instance IAM Identity Center.
- Le domaine et OpenSearch l'application doivent appartenir au même AWS compte.

Modification de la stratégie d'accès au domaine

Avant de configurer IAM Identity Center, vous devez mettre à jour la politique d'accès au domaine ou les autorisations du rôle IAM configuré dans les OpenSearch applications pour la propagation d'identités fiables.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "IAM Role configured in OpenSearch application"
            },
            "Action": "es:ESHttp*",
            "Resource": "domain-arn/*"
        },
        {
        ... // Any other permissions
        }
    ]
}
```

Configuration de l'authentification et de l'autorisation IAM Identity Center (console)

Vous pouvez activer l'authentification et l'autorisation IAM Identity Center pendant le processus de création du domaine ou en mettant à jour un domaine existant. Les étapes de configuration varient légèrement en fonction de l'option choisie.

Les étapes suivantes expliquent comment configurer un domaine existant pour l'authentification et l'autorisation IAM Identity Center dans la console Amazon OpenSearch Service :

1. Sous Configuration du domaine, accédez à Configuration de la sécurité, choisissez Modifier, accédez à la section Authentification du centre d'identité IAM, puis sélectionnez Activer l'accès à l'API authentifié auprès d'IAM Identity Center.

2. Sélectionnez la touche SubjectKey et Rôles comme suit.

   - Clé d'objet : choisissez l'un des attributs suivants UserId (par défaut) UserName et E-mail pour utiliser l'attribut correspondant comme principal d'accès au domaine.
   - Clé des rôles : choisissez l'un des rôles GroupId (par défaut) et GroupName utilisez les valeurs d'attribut correspondantes comme rôle principal fine-grained-access-controlpour tous les groupes associés au principal iDC.

Une fois que vous avez apporté vos modifications, enregistrez votre domaine.

Configuration d'un contrôle d'accès détaillé

Une fois que vous avez activé l'option IAM Identity Center sur votre OpenSearch domaine, vous pouvez configurer l'accès aux principaux IAM Identity Center en créant un mappage des rôles vers le rôle principal. La valeur du rôle principal pour le principal est basée sur l'appartenance au groupe du principal iDC et sur la RolesKey configuration de GroupId ou. GroupName

> **Note**
>
> Amazon OpenSearch Service peut prendre en charge jusqu'à 100 groupes pour un seul utilisateur. Si vous essayez d'utiliser un nombre d'instances supérieur au nombre autorisé, vous rencontrerez des incohérences dans le traitement de votre fine-grained-access-control autorisation et vous recevrez un message d'erreur 403.

Configuration de l'authentification et de l'autorisation (CLI) d'IAM Identity Center

```
  aws opensearch update-domain-config \
      --domain-name my-domain \
      --identity-center-options '{"EnabledAPIAccess": true,
 "IdentityCenterInstanceARN": "instance arn",  "SubjectKey": "UserId/UserName/
UserEmail" , "RolesKey": "GroupId/GroupName"}'
```

Désactivation de l'authentification IAM Identity Center sur le domaine

Pour désactiver IAM Identity Center sur votre OpenSearch domaine, procédez comme suit :

1. Choisissez le domaine, Actions, et Edit security configuration(Modifier la configuration de la sécurité).
2. Décochez Activer l'accès à l'API authentifié auprès d'IAM Identity Center.
3. Sélectionnez Enregistrer les modifications.
4. Une fois le traitement du domaine terminé, supprimez les mappages de rôles ajoutés pour les principaux iDC

Pour désactiver IAM Identity Center via la CLI, vous pouvez utiliser ce qui suit

```
aws opensearch update-domain-config \
    --domain-name my-domain \
    --identity-center-options '{"EnabledAPIAccess": false}'
```

Configuration de l'authentification Amazon Cognito pour les tableaux de bord OpenSearch

Vous pouvez authentifier et protéger votre installation par défaut des OpenSearch tableaux de bord Amazon OpenSearch Service à l'aide d'Amazon Cognito. L'authentification Amazon Cognito est facultative et disponible uniquement pour les domaines utilisant Elasticsearch OpenSearch 5.1 ou version ultérieure. Si vous ne configurez pas l'authentification Amazon Cognito, vous pouvez malgré tout protéger Dashboards à l'aide d'une stratégie d'accès basée sur l'adresse IP, d'un serveur proxy, de l'authentification HTTP de base ou de SAML.

La majeure partie du processus d'authentification se déroule dans Amazon Cognito, mais cette section fournit des directives et des exigences relatives à la configuration des ressources Amazon Cognito pour qu'elles fonctionnent OpenSearch avec les domaines de service. La tarification standard s'applique à toutes les ressources Amazon Cognito.

> **Tip**
>
> La première fois que vous configurez un domaine pour utiliser l'authentification Amazon Cognito pour les OpenSearch tableaux de bord, nous vous recommandons d'utiliser la console. Les ressources Amazon Cognito sont extrêmement personnalisables, et la console peut vous aider à identifier et comprendre les fonctions qui vous concernent.

Rubriques

- Prérequis
- Configurer un domaine pour utiliser l'authentification Amazon Cognito
- Autorisation du rôle authentifié
- Configuration des fournisseurs d'identité
- (Facultatif) Configuration du contrôle précis des accès
- (Facultatif) Personnalisation de la page de connexion
- (Facultatif) Configuration de la sécurité avancée
- Test
- Quotas
- Problèmes de configuration courants
- Désactivation de l'authentification Amazon Cognito pour les tableaux de bord OpenSearch
- Suppression de domaines utilisant l'authentification Amazon Cognito pour les tableaux de bord OpenSearch

Prérequis

Avant de pouvoir configurer l'authentification Amazon Cognito pour les OpenSearch tableaux de bord, vous devez remplir plusieurs conditions préalables. La console OpenSearch de service permet de rationaliser la création de ces ressources, mais la compréhension de l'objectif de chaque ressource facilite la configuration et le dépannage. L'authentification Amazon Cognito pour Dashboards nécessite les ressources suivantes :

- Groupe d'utilisateurs Amazon Cognito
- Groupes d'identités Amazon Cognito
- Rôle IAM auquel est attachée la politique `AmazonOpenSearchServiceCognitoAccess` (`CognitoAccessForAmazonOpenSearch`)

> **Note**
>
> Le groupe d'utilisateurs et le groupe d'identités doivent se trouver dans la même Région AWS. Vous pouvez utiliser le même groupe d'utilisateurs, le même pool d'identités et le même rôle IAM pour ajouter l'authentification Amazon Cognito pour les tableaux de bord à

OpenSearch plusieurs domaines de service. Pour en savoir plus, veuillez consulter la section the section called “Quotas”.

À propos du groupe d'utilisateurs

Les groupes d'utilisateurs ont deux fonctions principales : créer et gérer un annuaire d'utilisateurs et permettre l'inscription et la connexion des utilisateurs. Pour obtenir des instructions sur la création d'un groupe d'utilisateurs, consultez Getting started with user pools dans le manuel Amazon Cognito Developer Guide.

Lorsque vous créez un groupe d'utilisateurs à utiliser avec OpenSearch Service, tenez compte des points suivants :

- Votre groupe d'utilisateurs Amazon Cognito doit avoir un nom de domaine. OpenSearch Le service utilise ce nom de domaine pour rediriger les utilisateurs vers une page de connexion permettant d'accéder aux tableaux de bord. À part un nom de domaine, le groupe d'utilisateurs n'a pas besoin d'une configuration autre que celle par défaut.
- Vous devez spécifier les attributs standard obligatoires du groupe d'utilisateurs (par exemple : nom, date de naissance, adresse e-mail et numéro de téléphone). Vous ne pouvez pas modifier ces attributs une fois que vous avez créé le groupe d'utilisateurs. Vous devez donc choisir ceux qui vous concernent en ce moment.
- Lors de la création de votre groupe d'utilisateurs, choisissez si les utilisateurs peuvent créer leur propre compte, la fiabilité minimale des mots de passe des comptes et s'il convient d'activer l'authentification multi-facteurs. Si vous prévoyez d'utiliser un fournisseur d'identité externe, ces paramètres sont sans conséquence. Du point de vue technique, vous pouvez activer le groupe d'utilisateurs en tant que fournisseur d'identité et activer un fournisseur d'identité externe, mais la plupart des personnes préfèrent l'une ou l'autre méthode.

Le groupe d'utilisateurs IDs prend la forme de*`region_ID`*. Si vous prévoyez d'utiliser la AWS CLI ou un AWS SDK pour configurer le OpenSearch service, notez l'ID.

À propos du groupe d'identités

Les groupes d'identités vous permettent d'attribuer des rôles temporaires dotés de privilèges limités aux utilisateurs une fois qu'ils se sont connectés. Pour obtenir des instructions sur la création d'un pool d'identités, consultez la section Présentation de la console des pools d'identités dans le

manuel Amazon Cognito Developer Guide. Lorsque vous créez un pool d'identités à utiliser avec OpenSearch Service, tenez compte des points suivants :

- Si vous utilisez la console Amazon Cognito, vous devez cocher la case Activer l'accès aux identités non authentifiées pour créer le groupe d'identités. Après avoir créé le pool d'identités et configuré le domaine de OpenSearch service, Amazon Cognito désactive ce paramètre.
- Vous n'avez pas besoin d'ajouter de fournisseurs d'identités externes au groupe d'identités. Lorsque vous configurez le OpenSearch service pour utiliser l'authentification Amazon Cognito, il configure le groupe d'identités pour qu'il utilise le groupe d'utilisateurs que vous venez de créer.
- Une fois que vous avez créé le groupe d'identités, vous devez choisir des rôles IAM non authentifiés et authentifiés. Ces rôles spécifient les stratégies d'accès des utilisateurs avant et après qu'ils se soient connectés. Si vous utilisez la console Amazon Cognito, elle peut créer ces rôles à votre place. Une fois que vous avez créé le rôle authentifié, notez l'ARN, qui se présente sous la forme `arn:aws:iam::123456789012:role/Cognito_identitypoolnameAuth_Role`.

Le pool d'identités IDs prend la forme de`region:ID-ID-ID-ID-ID`. Si vous prévoyez d'utiliser la AWS CLI ou un AWS SDK pour configurer le OpenSearch service, notez l'ID.

À propos du rôle CognitoAccessForAmazonOpenSearch

OpenSearch Le service a besoin d'autorisations pour configurer les groupes d'utilisateurs et d'identités Amazon Cognito et les utiliser pour l'authentification. Vous pouvez utiliser`AmazonOpenSearchServiceCognitoAccess`, qui est une politique AWS gérée, à cette fin. `AmazonESCognitoAccess`est une ancienne politique qui a été remplacée `AmazonOpenSearchServiceCognitoAccess` lorsque le service a été renommé Amazon OpenSearch Service. Les deux politiques fournissent les autorisations Amazon Cognito minimales nécessaires pour activer l'authentification Amazon Cognito. Pour plus de détails sur les politiques, consultez AmazonOpenSearchServiceCognitoAccessle Guide de référence des politiques AWS gérées.

Si vous utilisez la console pour créer ou configurer votre domaine de OpenSearch service, elle crée un rôle IAM pour vous et associe la `AmazonOpenSearchServiceCognitoAccess` politique (ou la `AmazonESCognitoAccess` politique s'il s'agit d'un domaine Elasticsearch) au rôle. Le nom par défaut du rôle est `CognitoAccessForAmazonOpenSearch`.

Les politiques d'autorisation des rôles `AmazonOpenSearchServiceCognitoAccess` et `AmazonESCognitoAccess` les deux permettent au OpenSearch Service d'effectuer les actions suivantes sur tous les groupes d'identités et d'utilisateurs :

- Action : `cognito-idp:DescribeUserPool`
- Action : `cognito-idp:CreateUserPoolClient`
- Action : `cognito-idp:DeleteUserPoolClient`
- Action : `cognito-idp:UpdateUserPoolClient`
- Action : `cognito-idp:DescribeUserPoolClient`
- Action : `cognito-idp:AdminInitiateAuth`
- Action : `cognito-idp:AdminUserGlobalSignOut`
- Action : `cognito-idp:ListUserPoolClients`
- Action : `cognito-identity:DescribeIdentityPool`
- Action : `cognito-identity:SetIdentityPoolRoles`
- Action : `cognito-identity:GetIdentityPoolRoles`

Si vous utilisez le AWS CLI ou l'un des AWS SDKs, vous devez créer votre propre rôle, associer la politique et spécifier l'ARN de ce rôle lorsque vous configurez votre domaine de OpenSearch service. Le rôle doit avoir la relation d'approbation suivante :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "opensearchservice.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Pour obtenir des instructions, voir Créer un rôle pour déléguer des autorisations à un AWS service et Ajouter et supprimer des autorisations d'identité IAM dans le guide de l'utilisateur IAM.

Configurer un domaine pour utiliser l'authentification Amazon Cognito

Une fois les conditions requises remplies, vous pouvez configurer un domaine de OpenSearch service pour utiliser Amazon Cognito pour les tableaux de bord.

> **Note**
>
> Amazon Cognito n'est pas disponible du tout. Régions AWS Pour obtenir la liste des régions prises en charge, consultez la section Points de terminaison de service pour Amazon Cognito. Il n'est pas nécessaire d'utiliser la même région pour Amazon Cognito que pour OpenSearch le service.

Configuration de l'authentification Amazon Cognito (console)

Parce qu'elle crée le `CognitoAccessForAmazonOpenSearch` rôle qui vous convient, la console offre l'expérience de configuration la plus simple. Outre les autorisations de OpenSearch service standard, vous avez besoin de l'ensemble d'autorisations suivant pour utiliser la console afin de créer un domaine qui utilise l'authentification Amazon Cognito pour les OpenSearch tableaux de bord.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeVpcs",
        "cognito-identity:ListIdentityPools",
        "cognito-idp:ListUserPools",
        "iam:CreateRole",
        "iam:AttachRolePolicy"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
```

```
      "Action": [
        "iam:GetRole",
        "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::123456789012:role/service-
role/CognitoAccessForAmazonOpenSearch"
    }
  ]
}
```

Pour obtenir des instructions sur l'ajout d'autorisations à une identité (utilisateur, groupe d'utilisateurs ou rôle), consultez la section Ajout d'autorisations à une identité IAM (console).

Si `CognitoAccessForAmazonOpenSearch` existe déjà, vous avez besoin de moins d'autorisations :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeVpcs",
        "cognito-identity:ListIdentityPools",
        "cognito-idp:ListUserPools"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:GetRole",
        "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::123456789012:role/service-
role/CognitoAccessForAmazonOpenSearch"
    }
  ]
}
```

Pour configurer l'authentification Amazon Cognito pour Dashboards (console)

1. Ouvrez la console Amazon OpenSearch Service à l'adresse https://console.aws.amazon.com/aos/home/.
2. Sous Domains (Domaines), sélectionnez le domaine que vous souhaitez configurer.
3. Choisissez Actions, Edit security configuration (Modifier la configuration de sécurité).
4. Sélectionnez Enable Amazon Cognito authentication (Activer l'authentification Amazon Cognito).
5. Pour Région, sélectionnez celle Région AWS qui contient votre groupe d'utilisateurs et votre groupe d'identités Amazon Cognito.
6. Pour Cognito user pool (Groupe d'utilisateurs Cognito), sélectionnez un groupe d'utilisateurs ou créez-en un. Pour de plus amples informations, veuillez consulter the section called “À propos du groupe d'utilisateurs”.
7. Pour Cognito identity pool (Groupe d'identités Cognito), sélectionnez un groupe d'identités ou créez-en un. Pour de plus amples informations, veuillez consulter the section called “À propos du groupe d'identités”.

   > **Note**
   >
   > Les liens Créer un groupe d'utilisateurs et Créer un groupe d'identités vous dirigent vers la console Amazon Cognito pour créer ces ressources manuellement. Le processus n'est pas automatique. Pour de plus amples informations, veuillez consulter the section called “Prérequis”.

8. Pour nom de rôle IAM, utilisez la valeur par défaut `CognitoAccessForAmazonOpenSearch` (recommandé) ou entrez un nouveau nom. Pour de plus amples informations, veuillez consulter the section called “À propos du rôle CognitoAccessForAmazonOpenSearch”.
9. Sélectionnez Save Changes (Enregistrer les modifications).

Lorsque votre domaine a terminé le traitement, consultez les étapes de configuration supplémentaires dans the section called “Autorisation du rôle authentifié” et the section called “Configuration des fournisseurs d'identité”.

Configuration de l'authentification Amazon Cognito (AWS CLI)

Utilisez le `--cognito-options` paramètre pour configurer votre domaine OpenSearch de service. La syntaxe suivante est utilisée par les commandes `create-domain` et `update-domain-config` :

```
--cognito-options Enabled=true,UserPoolId="user-pool-id",IdentityPoolId="identity-pool-
id",RoleArn="arn:aws:iam::123456789012:role/CognitoAccessForAmazonOpenSearch"
```

Exemple

L'exemple suivant crée un domaine dans la région `us-east-1`, qui permet l'authentification Amazon Cognito pour Dashboards à l'aide du rôle `CognitoAccessForAmazonOpenSearch` et fournit un accès au domaine à `Cognito_Auth_Role` :

```
aws opensearch create-domain --domain-name my-domain --region us-east-1 --access-
policies '{ "Version":"2012-10-17", "Statement":[{"Effect":"Allow","Principal":{"AWS":
 ["arn:aws:iam::123456789012:role/
Cognito_Auth_Role"]},"Action":"es:ESHttp*","Resource":"arn:aws:es:us-
east-1:123456789012:domain/*" }]}' --engine-version "OpenSearch_1.0"
 --cluster-config InstanceType=m4.xlarge.search,InstanceCount=1
 --ebs-options EBSEnabled=true,VolumeSize=10 --cognito-options
 Enabled=true,UserPoolId="us-east-1_123456789",IdentityPoolId="us-
east-1:12345678-1234-1234-1234-123456789012",RoleArn="arn:aws:iam::123456789012:role/
CognitoAccessForAmazonOpenSearch"
```

Lorsque votre domaine a terminé le traitement, consultez les étapes de configuration supplémentaires dans the section called “Autorisation du rôle authentifié” et the section called “Configuration des fournisseurs d'identité”.

Configuration de l'authentification Amazon Cognito ()AWS SDKs

AWS SDKs (sauf Android et iOS SDKs) prennent en charge toutes les opérations définies dans le Amazon OpenSearch Service API Reference, y compris le `CognitoOptions` paramètre des `UpdateDomainConfig` opérations `CreateDomain` et. Pour plus d'informations sur l'installation et l'utilisation du AWS SDKs, consultez la section Kits de développement AWS logiciel.

Lorsque votre domaine a terminé le traitement, consultez les étapes de configuration supplémentaires dans the section called “Autorisation du rôle authentifié” et the section called “Configuration des fournisseurs d'identité”.

Autorisation du rôle authentifié

Par défaut, le rôle IAM authentifié que vous avez configuré en suivant les instructions the section called “À propos du groupe d'identités” ne dispose pas des privilèges nécessaires pour accéder OpenSearch aux tableaux de bord. Vous devez lui apporter des autorisations supplémentaires.

> **Note**
>
> Si vous avez configuré un contrôle d'accès détaillé et que vous utilisez une politique d'accès ouverte ou basée sur IP, vous pouvez ignorer cette étape.

Vous pouvez inclure ces autorisations dans une politique basée sur l'identité, mais à moins que vous ne souhaitiez que les utilisateurs authentifiés aient accès à tous les domaines du OpenSearch service, une stratégie basée sur les ressources attachée à un seul domaine est la meilleure approche.

Pour le `Principal`, spécifiez l'ARN du rôle authentifié Cognito que vous avez configuré conformément aux instructions figurant dans the section called “À propos du groupe d'identités”.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "AWS":[
               "arn:aws:iam::123456789012:role/Cognito_identitypoolnameAuth_Role"
            ]
         },
         "Action":[
            "es:ESHttp*"
         ],
         "Resource":"arn:aws:es:region:123456789012:domain/domain-name/*"
      }
   ]
}
```

Pour obtenir des instructions sur l'ajout d'une politique basée sur les ressources à un domaine OpenSearch de service, consultez. the section called “Configuration des politiques d'accès”

Configuration des fournisseurs d'identité

Lorsque vous configurez un domaine pour utiliser l'authentification Amazon Cognito pour les tableaux de bord, OpenSearch Service ajoute un client d'application au groupe d'utilisateurs et ajoute le groupe d'utilisateurs au pool d'identités en tant que fournisseur d'authentification.

> **Warning**
>
> Ne renommez pas et ne supprimez pas le client d'application.

Selon la manière dont vous avez configuré votre groupe d'utilisateurs, vous pouvez avoir besoin de créer des comptes d'utilisateur manuellement, ou les utilisateurs peuvent créer leur propre compte. Si ces paramètres sont acceptables, aucune action n'est requise de votre part. Toutefois, de nombreuses personnes préfèrent utiliser des fournisseurs d'identité externes.

Pour activer un fournisseur d'identité SAML 2.0, vous devez fournir un document de métadonnées SAML. Pour activer des fournisseurs d'identité sociaux tels que Login with Amazon, Facebook et Google, vous devez vous procurer un ID d'application et une clé secrète d'application auprès de ces fournisseurs. Vous pouvez activer n'importe quelle combinaison de fournisseurs d'identité.

Le moyen le plus simple de configurer votre groupe d'utilisateurs est d'utiliser la console Amazon Cognito. Pour obtenir des instructions, consultez les sections Connexion au groupe d'utilisateurs avec des fournisseurs d'identité tiers et Paramètres spécifiques à l'application avec le client d'application dans le guide du développeur Amazon Cognito.

(Facultatif) Configuration du contrôle précis des accès

Vous avez peut-être remarqué que les paramètres du pool d'identités par défaut attribuent à chaque utilisateur qui se connecte le même rôle IAM (`Cognito_`*`identitypool`*`Auth_Role`), ce qui signifie que chaque utilisateur peut accéder aux mêmes AWS ressources. Si vous souhaitez utiliser un contrôle précis des accès avec Amazon Cognito (par exemple, si vous souhaitez que les analystes de votre organisation disposent d'un accès en lecture seule à plusieurs index, mais que les développeurs disposent d'un accès en écriture à tous les index), vous avez deux options :

- Créez des groupes d'utilisateurs et configurez votre fournisseur d'identité pour choisir le rôle IAM en fonction du jeton d'authentification de l'utilisateur (recommandé).
- Configurez votre fournisseur d'identité pour choisir le rôle IAM en fonction d'une ou de plusieurs règles.

Pour obtenir une procédure pas à pas qui inclut un contrôle d'accès affiné, veuillez consulter the section called “Didacticiel : contrôle précis des accès avec l'authentification Cognito”.

> **Important**
>
> Tout comme le rôle par défaut, Amazon Cognito doit faire partie de la relation d'approbation de chaque rôle supplémentaire. Pour plus de détails, consultez la section Création de rôles pour le mappage des rôles dans le manuel Amazon Cognito Developer Guide.

Groupes d'utilisateurs et jetons

Lorsque vous créez un groupe d'utilisateurs, vous choisissez un rôle IAM pour les membres du groupe. Pour plus d'informations sur la création de groupes, consultez la section Ajouter des groupes à un groupe d'utilisateurs dans le manuel Amazon Cognito Developer Guide.

Une fois que vous avez créé un ou plusieurs groupes d'utilisateurs, vous pouvez configurer votre fournisseur d'authentification pour affecter les utilisateurs aux rôles de leurs groupes et non au rôle par défaut du groupe d'identités. Choisissez Choose role from token (Utiliser le rôle du jeton), puis Utiliser le rôle authentifié par défaut ou DENY (REFUSER) pour spécifier la façon dont le groupe d'identités doit gérer les utilisateurs qui ne font pas partie d'un groupe.

Règles

Les règles correspondent essentiellement à une série d'instructions `if` qu'Amazon Cognito évalue de manière séquentielle. Par exemple, si l'adresse e-mail d'un utilisateur contient `@corporate`, Amazon Cognito attribue le `Role_A` à cet utilisateur. Si l'adresse e-mail d'un utilisateur contient `@subsidiary`, il attribue `Role_B` à cet utilisateur. Sinon, il attribue à l'utilisateur le rôle authentifié par défaut.

Pour en savoir plus, consultez la section Utilisation du mappage basé sur des règles pour attribuer des rôles aux utilisateurs dans le manuel Amazon Cognito Developer Guide.

(Facultatif) Personnalisation de la page de connexion

Vous pouvez utiliser la console Amazon Cognito pour télécharger un logo personnalisé et apporter des modifications CSS à la page de connexion. Pour obtenir des instructions et une liste complète des propriétés CSS, consultez la section Personnalisation de l'image de marque (classique) de l'interface utilisateur hébergée dans le manuel Amazon Cognito Developer Guide.

(Facultatif) Configuration de la sécurité avancée

Les groupes d'utilisateurs Amazon Cognito prennent en charge les fonctionnalités de sécurité avancée, telles que l'authentification multifacteur, la vérification des informations d'identification compromises et l'authentification adaptative. Pour en savoir plus, consultez la section Utilisation des fonctionnalités de sécurité des groupes d'utilisateurs Amazon Cognito dans le manuel du développeur Amazon Cognito.

Test

Une fois que vous êtes satisfait de votre configuration, vérifiez que l'expérience utilisateur répond à vos attentes.

Pour accéder aux OpenSearch tableaux de bord

1. Accédez à `https://`*`opensearch-domain`*`/_dashboards` dans un navigateur web. Pour vous connecter directement à un locataire spécifique, ajoutez `?security_tenant=`*`tenant-name`* à l'URL.
2. Connectez-vous à l'aide de vos informations d'identification préférées.
3. Une fois OpenSearch les tableaux de bord chargés, configurez au moins un modèle d'index. Dashboards utilise ces modèles pour identifier les index à analyser. Entrez *, choisissez Next step (Étape suivante), puis Create index pattern (Créer un modèle d'index).
4. Pour explorer vos données, choisissez Discover (Découvrir).

Si une étape de ce processus échoue, consultez the section called “Problèmes de configuration courants” pour obtenir des informations de dépannage.

Quotas

Amazon Cognito comporte des limites souples sur un grand nombre de ses ressources. Si vous souhaitez activer l'authentification par tableau de bord pour un grand nombre de domaines de OpenSearch service, consultez les quotas dans Amazon Cognito et demandez des augmentations de la limite si nécessaire.

Chaque domaine OpenSearch de service ajoute un client d'application au groupe d'utilisateurs, ce qui ajoute un fournisseur d'authentification au pool d'identités. Si vous activez l'authentification par tableau de OpenSearch bord pour plus de 10 domaines, vous risquez de rencontrer la limite du

« nombre maximum de fournisseurs de pool d'utilisateurs Amazon Cognito par pool d'identités ». Si vous dépassez une limite, tous les domaines de OpenSearch service que vous essayez de configurer pour utiliser l'authentification Amazon Cognito pour les tableaux de bord peuvent rester bloqués dans un état de configuration en cours de traitement.

Problèmes de configuration courants

Les tableaux suivants répertorient les problèmes de configuration courants et les solutions correspondantes.

Configuration du OpenSearch service

Problème	Solution
`OpenSearch Service can't create the role` (console)	Vous ne disposez pas des autorisations IAM correctes . Ajoutez les autorisations spécifiées dans the section called “Configuration de l'authentification Amazon Cognito (console)”.
`User is not authorized to perform: iam:PassRole on resource CognitoAccessForAmazonOpenSearch` (console)	Vous n'avez pas `iam:PassRole` les autorisations nécessaires pour ce CognitoAccessForAmazonOpenSearchrôle. Attachez la politique suivante à votre compte : JSON `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "iam:PassRole" ], "Resource": "arn:aws:iam:: 123456789012:role/service-role/CognitoAccessForAmazonOpenSearch " } ] }`

Problème	Solution
	Vous pouvez également attacher la politique `IAMFullAccess` .
`User is not authorized to perform: cognito-identity:ListIdentityPools on resource`	Vous ne disposez pas des autorisations en lecture pour Amazon Cognito. Attachez la politique `AmazonCognitoReadOnly` à votre compte.
`An error occurred (ValidationException) when calling the CreateDomain operation: OpenSearch Service must be allowed to use the passed role`	OpenSearch Le service n'est pas spécifié dans la relation de confiance du `CognitoAccessForAmazonOpenSearch` rôle. Vérifiez que votre rôle utilise la relation d'approbation qui est spécifiée dans the section called “À propos du rôle CognitoAccessForAmazonOpenSearch”. Vous pouvez aussi utiliser la console pour configurer l'authentification Amazon Cognito. La console crée un rôle pour vous.
`An error occurred (ValidationException) when calling the CreateDomain operation: User is not authorized to perform: cognito-idp: action on resource: user pool`	Le rôle spécifié dans `--cognito-options` n'est pas autorisé à accéder aux ressources Amazon Cognito. Vérifiez que la `AmazonOpenSearchServiceCognitoAccess` politique AWS gérée est attachée au rôle. Vous pouvez aussi utiliser la console pour configurer l'authentification Amazon Cognito. La console crée un rôle pour vous.
`An error occurred (ValidationException) when calling the CreateDomain operation: User pool does not exist`	OpenSearch Le service ne trouve pas le groupe d'utilisateurs. Vérifiez que vous en avez créé un et qu'il a l'ID correct. Pour trouver l'ID, vous pouvez utiliser la console Amazon Cognito ou la commande suivante : AWS CLI `aws cognito-idp list-user-pools --max-results 60 --region region`

Problème	Solution
`An error occurred (ValidationException) when calling the CreateDomain operation : IdentityPool not found`	OpenSearch Le service ne trouve pas le pool d'identités. Vérifiez que vous en avez créé un et qu'il a l'ID correct. Pour trouver l'ID, vous pouvez utiliser la console Amazon Cognito ou la commande suivante : AWS CLI `aws cognito-identity list-identity-pools --max-results 60 --region region`
`An error occurred (ValidationException) when calling the CreateDomain operation : Domain needs to be specified for user pool`	Le groupe d'utilisateurs n'a pas de nom de domaine. Vous pouvez en configurer un à l'aide de la console Amazon Cognito ou de la commande AWS CLI suivante : `aws cognito-idp create-user-pool-domain --domain name --user-pool-id id`

Accès aux OpenSearch tableaux de bord

Problème	Solution
La page de connexion n'affiche pas mes fournisseurs d'identité préférés.	Vérifiez que vous avez activé le fournisseur d'identité pour le client OpenSearch Service app comme indiqué dansthe section called “Configuration des fournisseurs d'identité”.
La page de connexion ne semble pas associée à mon organisation.	Consultez the section called “(Facultatif) Personnalisation de la page de connexion”.
Mes informations d'identification de connexion ne fonctionnent pas.	Vérifiez que vous avez configuré le fournisseur d'identité de la façon spécifiée dans the section called “Configuration des fournisseurs d'identité”. Si vous utilisez le groupe d'utilisateurs comme fournisseur d'identité, vérifiez que le compte existe sur la console Amazon Cognito.

Problème	Solution
OpenSearch Les tableaux de bord ne se chargent pas du tout ou ne fonctionnent pas correctement.	Le rôle authentifié par Amazon Cognito nécessite les autorisations `es:ESHttp*` pour permettre au domaine (`/*`) d'accéder à Dashboards et de l'utiliser. Vérifiez que vous avez ajouté une stratégie d'accès, comme indiqué dans the section called “Autorisation du rôle authentifié”.
Lorsque je me déconnecte des OpenSearch tableaux de bord depuis un onglet, les autres onglets affichent un message indiquant que le jeton d'actualisation a été révoqué.	Lorsque vous vous déconnectez d'une session de OpenSearch tableaux de bord en utilisant l'authent ification Amazon Cognito OpenSearch , le service exécute AdminUserGlobalSignOutune opération qui vous déconnecte de toutes les sessions de tableaux de bord OpenSearch actives.

Problème	Solution
`Invalid identity pool configuration. Check assigned IAM roles for this pool.`	Amazon Cognito ne dispose pas des autorisations nécessaires pour assumer le rôle IAM au nom de l'utilisateur authentifié. Modifiez la relation d'approbation pour le rôle à inclure : JSON `{ "Version": "2012-10-17", "Statement": [{ "Effect": "Allow", "Principal": { "Federated": "cognito-identity.amazonaws.com" }, "Action": "sts:AssumeRoleWithWebIdentity", "Condition": { "StringEquals": { "cognito-identity.amazonaws.com:aud": " identity-pool-id " }, "ForAnyValue:StringLike": { "cognito-identity.amazonaws.com:amr": "authenticated" } } }] }`
`Token is not from a supported provider of this identity pool.`	Cette rare erreur peut se produire lorsque vous supprimez le client d'application client du groupe d'utilisateurs. Essayez d'ouvrir Dashboards dans une nouvelle session de navigateur.

Désactivation de l'authentification Amazon Cognito pour les tableaux de bord OpenSearch

Utilisez la procédure suivante pour désactiver l'authentification Amazon Cognito pour Dashboards.

Pour désactiver l'authentification Amazon Cognito pour Dashboards (console)

1. Ouvrez la console Amazon OpenSearch Service.
2. Sous Domains (Domaines), sélectionnez le domaine que vous souhaitez configurer.
3. Choisissez Actions, Edit security configuration (Modifier la configuration de sécurité).
4. Désélectionnez Enable Amazon Cognito authentication (Activer l'authentification Amazon Cognito).
5. Sélectionnez Enregistrer les modifications.

> **Important**
>
> Si vous n'avez plus besoin du groupe d'utilisateurs et du groupe d'identités Amazon Cognito, supprimez-les. Sinon, les frais continuent de vous être facturés.

Suppression de domaines utilisant l'authentification Amazon Cognito pour les tableaux de bord OpenSearch

Pour éviter que les domaines qui utilisent l'authentification Amazon Cognito pour les tableaux de bord ne restent bloqués dans un état de configuration de traitement, supprimez les domaines de OpenSearch service avant de supprimer les groupes d'utilisateurs et d'identités Amazon Cognito associés.

Utilisation des rôles liés à un service pour Amazon Service OpenSearch

Amazon OpenSearch Service utilise des rôles AWS Identity and Access Management liés à un service (IAM). Un rôle lié à un service est un type unique de rôle IAM lié directement à Service. OpenSearch Les rôles liés à un service sont prédéfinis par OpenSearch Service et incluent toutes les autorisations requises par le service pour appeler d'autres AWS services en votre nom.

Un rôle lié à un service simplifie la configuration du OpenSearch service, car vous n'avez pas besoin d'ajouter manuellement les autorisations requises. OpenSearch Service définit les autorisations de ses rôles liés à un service ; sauf définition contraire, seul OpenSearch Service peut endosser ses rôles. Les autorisations définies comprennent la politique d'approbation et la politique d'autorisation. De plus, cette politique d'autorisation ne peut pas être attachée à une autre entité IAM. Pour les mises à jour des rôles liés aux services et des politiques d'autorisation, consultez l'historique des documents pour Amazon OpenSearch Service.

Pour plus d'informations sur les autres services qui prennent en charge les rôles liés à un service, consultez AWS Services qui fonctionnent avec IAM et recherchez les services avec un Oui dans la colonne Rôles liés à un service. Sélectionnez un Oui ayant un lien pour consulter la documentation du rôle lié à un service, pour ce service.

Rubriques

- Utilisation des rôles liés à un service pour créer des domaines VPC et interroger directement les sources de données
- Utilisation des rôles liés à un service pour créer OpenSearch des collections sans serveur
- Utilisation des rôles liés à un service pour créer des pipelines d'ingestion OpenSearch

Utilisation des rôles liés à un service pour créer des domaines VPC et interroger directement les sources de données

Amazon OpenSearch Service utilise des rôles AWS Identity and Access Management liés à un service (IAM). Un rôle lié à un service est un type unique de rôle IAM lié directement à Service. OpenSearch Les rôles liés à un service sont prédéfinis par OpenSearch Service et incluent toutes les autorisations requises par le service pour appeler d'autres AWS services en votre nom.

OpenSearch Le service utilise le rôle lié à un service nommé AWSServiceRoleForAmazonOpenSearchService, qui fournit les autorisations minimales Amazon EC2 et Elastic Load Balancing nécessaires pour activer l'accès VPC d'un domaine ou d'une source de données de requête directe.

Rôle Elasticsearch hérité

Amazon OpenSearch Service utilise un rôle lié à un service appelé. `AWSServiceRoleForAmazonOpenSearchService` Vos comptes peuvent également contenir

un rôle lié à un service hérité appelé `AWSServiceRoleForAmazonElasticsearchService`, qui fonctionne avec les points de terminaison obsolètes de l'API Elasticsearch.

Si le rôle Elasticsearch hérité n'existe pas dans votre compte, OpenSearch Service crée automatiquement un rôle OpenSearch lié à un service la première fois que vous créez un domaine. OpenSearch Dans le cas contraire, votre compte continue d'utiliser le rôle Elasticsearch. Pour que cette création automatique aboutisse, vous devez avoir les autorisations permettant d'effectuer l'action `iam:CreateServiceLinkedRole`.

Autorisations

Le rôle lié à un service `AWSServiceRoleForAmazonOpenSearchService` approuve les services suivants pour endosser le rôle :

- `opensearchservice.amazonaws.com`

La politique d'autorisation du rôle nommée `AmazonOpenSearchServiceRolePolicy`permet à OpenSearch Service d'effectuer les actions suivantes sur les ressources spécifiées :

- Action : `acm:DescribeCertificate` sur *
- Action : `cloudwatch:PutMetricData` sur *
- Action : `ec2:CreateNetworkInterface` sur *
- Action : `ec2:DeleteNetworkInterface` sur *
- Action : `ec2:DescribeNetworkInterfaces` sur *
- Action : `ec2:ModifyNetworkInterfaceAttribute` sur *
- Action : `ec2:DescribeSecurityGroups` sur *
- Action : `ec2:DescribeSubnets` sur *
- Action : `ec2:DescribeVpcs` sur *
- Action : `ec2:CreateTags` sur l'ensemble des interfaces réseau et des points de terminaison d'un VPC
- Action : `ec2:DescribeTags` sur *
- Action : `ec2:CreateVpcEndpoint` sur tous les groupes de sécurité VPCs, sous-réseaux et tables de routage, ainsi que sur tous les points de terminaison VPC, lorsque la requête contient la balise `OpenSearchManaged=true`

- Action : `ec2:ModifyVpcEndpoint` sur tous les groupes de sécurité VPCs, sous-réseaux et tables de routage, ainsi que sur tous les points de terminaison VPC, lorsque la requête contient la balise `OpenSearchManaged=true`
- Action : `ec2:DeleteVpcEndpoints` sur tous les points de terminaison lorsque la requête contient la balise `OpenSearchManaged=true`
- Action : `ec2:AssignIpv6Addresses` sur *
- Action : `ec2:UnAssignIpv6Addresses` sur *
- Action : `elasticloadbalancing:AddListenerCertificates` sur *
- Action : `elasticloadbalancing:RemoveListenerCertificates` sur *

Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, un groupe ou un rôle) de créer, modifier ou supprimer un rôle lié à un service. Pour plus d'informations, consultez Autorisations de rôles liés à un service dans le Guide de l'utilisateur IAM.

Création du rôle lié à un service

Vous n'avez pas besoin de créer manuellement un rôle lié à un service. Lorsque vous créez un domaine compatible VPC ou une source de données de requête directe à l'aide de la AWS Management Console, OpenSearch Service crée pour vous le rôle lié à un service. Pour que cette création automatique aboutisse, vous devez avoir les autorisations permettant d'effectuer l'action `iam:CreateServiceLinkedRole`.

Vous pouvez également utiliser la console IAM, la CLI IAM ou l'API IAM pour créer manuellement un rôle lié à un service. Pour plus d'informations, consultez Création d'un rôle lié à un service dans le Guide de l'utilisateur IAM.

Modifier le rôle lié à un service

OpenSearch Service ne vous permet pas de modifier le rôle `AWSServiceRoleForAmazonOpenSearchService` lié à un service. Une fois que vous avez créé un rôle lié à un service, vous ne pouvez pas changer le nom du rôle, car plusieurs entités peuvent faire référence au rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour plus d'informations, consultez Modification d'un rôle lié à un service dans le IAM Guide de l'utilisateur.

Suppression du rôle lié à un service

Si vous n'avez plus besoin d'utiliser une fonctionnalité ou un service qui nécessite un rôle lié à un service, nous vous recommandons de supprimer ce rôle. De cette façon, vous n'avez aucune entité

inutilisée qui n'est pas surveillée ou gérée activement. Cependant, vous devez nettoyer votre rôle lié à un service avant de pouvoir le supprimer manuellement.

Nettoyage du rôle lié au service

Avant de pouvoir utiliser IAM pour supprimer un rôle lié à un service, vous devez d'abord vérifier qu'aucune session n'est active pour le rôle et supprimer toutes les ressources utilisées par le rôle.

Pour vérifier si une session est active pour le rôle lié à un service dans la console IAM

1. Connectez-vous à l' AWS Management Console et ouvrez la console IAM sur https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation de la console IAM, sélectionnez Roles (Rôles). Ensuite, sélectionnez le nom (et non la case à cocher) du rôle `AWSServiceRoleForAmazonOpenSearchService`.
3. Sur la page Récapitulatif du rôle sélectionné, choisissez l'onglet Access Advisor.
4. Dans l'onglet Access Advisor, consultez l'activité récente pour le rôle lié à un service.

> **Note**
>
> Si vous n'êtes pas certain que OpenSearch Service utilise le `AWSServiceRoleForAmazonOpenSearchService` rôle, vous pouvez essayer de supprimer le rôle. Si le service utilise le rôle, la suppression échoue et vous pouvez visualiser les ressources utilisant le rôle. Si le rôle est en cours d'utilisation, vous devez attendre la fin de la session avant de pouvoir supprimer le rôle, et/ou supprimer les ressources utilisant le rôle. Vous ne pouvez pas révoquer la session d'un rôle lié à un service.

Suppression manuelle d'un rôle lié à un service

Supprimez les rôles liés à un service depuis la console IAM, l'API ou la CLI. AWS Pour de plus amples informations, veuillez consulter Suppression d'un rôle lié à un service dans le Guide de l'utilisateur IAM.

Utilisation des rôles liés à un service pour créer OpenSearch des collections sans serveur

OpenSearch Utilise des rôles liés à un service AWS Identity and Access Management (IAM) sans serveur. Un rôle lié à un service est un type unique de rôle IAM lié directement à Service. OpenSearch Les rôles liés à un service sont prédéfinis par OpenSearch Service et incluent toutes les autorisations requises par le service pour appeler d'autres AWS services en votre nom.

OpenSearch sans serveur utilise le rôle lié à un service nommé AWSServiceRoleForAmazonOpenSearchServerless, qui fournit les autorisations nécessaires pour publier des métriques liées à la technologie sans serveur sur votre compte CloudWatch. La politique d'autorisations de rôle associée à AWSService RoleForAmazonOpenSearchServerless est nomméeAmazonOpenSearchServerlessServiceRolePolicy. Pour plus d'informations sur la politique, consultez AmazonOpenSearchServerlessServiceRolePolicyle Guide de référence des politiques AWS gérées.

Autorisations des rôles liés à un service pour sans serveur OpenSearch

OpenSearch sans serveur utilise le rôle lié à un service nommé AWSServiceRoleForAmazonOpenSearchServerless, ce qui permet à OpenSearch sans serveur d'appeler les AWS services en votre nom.

Le rôle AWSService RoleForAmazonOpenSearchServerless lié à un service approuve les services suivants pour endosser le rôle :

- `observability.aoss.amazonaws.com`

La politique d'autorisation du rôle nommée `AmazonOpenSearchServerlessServiceRolePolicy` permet à OpenSearch sans serveur d'effectuer les actions suivantes sur les ressources spécifiées :

- Action : `cloudwatch:PutMetricData` sur toutes les ressources AWS

> **Note**
>
> La politique inclut la clé de condition`{"StringEquals": {"cloudwatch:namespace": "AWS/AOSS"}}`, ce qui signifie que le rôle lié à un service peut uniquement envoyer des données métriques à l'`AWS/AOSS` CloudWatchespace de noms.

Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, un groupe ou un rôle) de créer, modifier ou supprimer un rôle lié à un service. Pour plus d'informations, consultez Autorisations de rôles liés à un service dans le Guide de l'utilisateur IAM.

Création du rôle lié à un service pour sans serveur OpenSearch

Vous n'avez pas besoin de créer manuellement un rôle lié à un service. Lorsque vous créez une collection OpenSearch sans serveur dans l' AWS Management Console, l'ou l' AWS API AWS CLI, OpenSearch sans serveur crée pour vous le rôle lié à un service.

> **Note**
>
> La première fois que vous créez une collection, le rôle `iam:CreateServiceLinkedRole` doit vous être attribué dans une politique basée sur l'identité.

Si vous supprimez ce rôle lié à un service et que vous avez ensuite besoin de le recréer, vous pouvez utiliser la même procédure pour recréer le rôle dans votre compte. Lorsque vous créez une collection OpenSearch sans serveur, OpenSearch sans serveur crée à nouveau le rôle lié à un service pour vous.

Vous pouvez également utiliser la console IAM pour créer un rôle lié à un service avec le cas d'utilisation Amazon OpenSearch sans serveur. Dans l'API AWS CLI ou dans l' AWS API, créez un rôle lié à un service avec le nom du `observability.aoss.amazonaws.com` service :

```
aws iam create-service-linked-role --aws-service-name
 "observability.aoss.amazonaws.com"
```

Pour plus d'informations, consultez Création d'un rôle lié à un service dans le Guide de l'utilisateur IAM. Si vous supprimez ce rôle lié à un service, vous pouvez utiliser ce même processus pour créer le rôle à nouveau.

Modification du rôle lié à un service pour sans serveur OpenSearch

OpenSearch sans serveur ne vous permet pas de modifier le rôle lié à un AWSService RoleForAmazonOpenSearchServerless service. Une fois que vous avez créé un rôle lié à un service, vous ne pouvez pas changer le nom du rôle, car plusieurs entités peuvent faire référence au rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour plus d'informations, consultez Modification d'un rôle lié à un service dans le IAM Guide de l'utilisateur.

Suppression du rôle lié à un service pour sans serveur OpenSearch

Si vous n'avez plus besoin d'utiliser une fonction ou un service qui nécessite un rôle lié à un service, nous vous recommandons de supprimer ce rôle. Cela vous évite d'avoir une entité inutilisée non surveillée ou non gérée activement. Cependant, vous devez nettoyer les ressources de votre rôle lié à un service avant de pouvoir les supprimer manuellement.

Pour supprimer le AWSServiceRoleForAmazonOpenSearchServerless, vous devez d'abord supprimer toutes les collections OpenSearch Serverless de votre Compte AWS.

> **Note**
>
> Si OpenSearch sans serveur utilise le rôle lorsque vous essayez de supprimer les ressources, la suppression peut échouer. Si cela se produit, patientez quelques minutes et réessayez.

Pour supprimer manuellement le rôle lié à un service à l'aide d'IAM

Utilisez la console IAM, le AWS CLI, ou l' AWS API pour supprimer le rôle lié au AWSService RoleForAmazonOpenSearchServerless service. Pour plus d'informations, consultez Suppression d'un rôle lié à un service dans le Guide de l'utilisateur IAM.

Régions prises en charge pour les rôles OpenSearch liés à un service sans serveur

OpenSearch sans serveur prend en charge l'utilisation du rôle AWSService RoleForAmazonOpenSearchServerless lié à un service dans toutes les régions où OpenSearch sans serveur est disponible. Pour obtenir la liste des régions prises en charge, consultez la section Points de terminaison et quotas Amazon OpenSearch Serverless dans le. Références générales AWS

Utilisation des rôles liés à un service pour créer des pipelines d'ingestion OpenSearch

Amazon OpenSearch Ingestion utilise des AWS Identity and Access Management rôles liés à un service (IAM). Un rôle lié à un service est un type unique de rôle IAM lié directement à Ingestion. OpenSearch Les rôles liés à un service sont prédéfinis OpenSearch par Ingestion et incluent toutes les autorisations requises par le service pour appeler d'autres AWS services en votre nom.

OpenSearch L'ingestion utilise le rôle lié au service nommé AWSServiceRoleForAmazonOpenSearchIngestionService, sauf lorsque vous

utilisez un VPC autogéré, auquel cas elle utilise le rôle lié au service nommé. AWSServiceRoleForOpensearchIngestionSelfManagedVpce La politique ci-jointe fournit les autorisations nécessaires pour créer un cloud privé virtuel (VPC) entre votre compte et OpenSearch Ingestion, ainsi que pour publier des CloudWatch métriques sur votre compte.

Autorisations

Le rôle lié à un service `AWSServiceRoleForAmazonOpenSearchIngestionService` approuve les services suivants pour endosser le rôle :

- `osis.amazon.com`

La politique d'autorisation du rôle nommée `AmazonOpenSearchIngestionServiceRolePolicy` permet OpenSearch à Ingestion d'effectuer les actions suivantes sur les ressources spécifiées :

- Action : `ec2:DescribeSubnets` sur *
- Action : `ec2:DescribeSecurityGroups` sur *
- Action : `ec2:DeleteVpcEndpoints` sur *
- Action : `ec2:CreateVpcEndpoint` sur *
- Action : `ec2:DescribeVpcEndpoints` sur *
- Action : `ec2:CreateTags` sur `arn:aws:ec2:*:*:network-interface/*`
- Action : `cloudwatch:PutMetricData` sur `cloudwatch:namespace": "AWS/OSIS"`

Le rôle lié à un service `AWSServiceRoleForOpensearchIngestionSelfManagedVpce` approuve les services suivants pour endosser le rôle :

- `self-managed-vpce.osis.amazon.com`

La politique d'autorisation du rôle nommée `OpenSearchIngestionSelfManagedVpcePolicy` permet OpenSearch à Ingestion d'effectuer les actions suivantes sur les ressources spécifiées :

- Action : `ec2:DescribeSubnets` sur *
- Action : `ec2:DescribeSecurityGroups` sur *
- Action : `ec2:DescribeVpcEndpoints` sur *
- Action : `cloudwatch:PutMetricData` sur `cloudwatch:namespace": "AWS/OSIS"`

Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, un groupe ou un rôle) de créer, modifier ou supprimer un rôle lié à un service. Pour plus d'informations, consultez Autorisations de rôles liés à un service dans le Guide de l'utilisateur IAM.

Création du rôle lié à un service pour Ingestion OpenSearch

Vous n'avez pas besoin de créer manuellement un rôle lié à un service. Lorsque vous créez un pipeline d' OpenSearch ingestion dans l'API AWS Management Console AWS CLI, le ou l' AWS API, OpenSearch Ingestion crée le rôle lié au service pour vous.

Si vous supprimez ce rôle lié à un service et que vous avez ensuite besoin de le recréer, vous pouvez utiliser la même procédure pour recréer le rôle dans votre compte. Lorsque vous créez un pipeline d' OpenSearch OpenSearch ingestion, Ingestion crée à nouveau le rôle lié à un service pour vous.

Modification du rôle lié à un service pour Ingestion OpenSearch

OpenSearch L'ingestion ne vous permet pas de modifier le rôle `AWSServiceRoleForAmazonOpenSearchIngestionService` lié à un service. Une fois que vous avez créé un rôle lié à un service, vous ne pouvez pas changer le nom du rôle, car plusieurs entités peuvent faire référence à ce rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour plus d'informations, consultez Modification d'un rôle lié à un service dans le IAM Guide de l'utilisateur.

Suppression du rôle lié à un service pour Ingestion OpenSearch

Si vous n'avez plus besoin d'utiliser une fonction ou un service qui nécessite un rôle lié à un service, nous vous recommandons de supprimer ce rôle. De cette façon, vous n'avez aucune entité inutilisée qui n'est pas surveillée ou gérée activement. Cependant, vous devez nettoyer les ressources de votre rôle lié à un service avant de pouvoir les supprimer manuellement.

Nettoyer un rôle lié à un service

Avant de pouvoir utiliser IAM pour supprimer un rôle lié à un service, vous devez supprimer toutes les ressources utilisées par le rôle.

> **Note**
>
> Si OpenSearch Ingestion utilise le rôle lorsque vous essayez de supprimer les ressources, la suppression peut échouer. Si cela se produit, patientez quelques minutes et réessayez.

Pour supprimer les ressources OpenSearch d'ingestion utilisées par le **`AWSServiceRoleForOpensearchIngestionSelfManagedVpce`** rôle **`AWSServiceRoleForAmazonOpenSearchIngestionService`** or

1. Accédez à la console Amazon OpenSearch Service et choisissez Ingestion.
2. Supprimez tous les pipelines. Pour obtenir des instructions, veuillez consulter the section called “Supprime des pipelines.”.

Supprimer le rôle lié à un service pour Ingestion OpenSearch

Vous pouvez utiliser la console OpenSearch Ingestion pour supprimer un rôle lié à un service.

Pour supprimer un rôle lié à un service (console)

1. Accédez à la Console IAM.
2. Choisissez Rôles et recherchez le AWSServiceRoleForOpensearchIngestionSelfManagedVpcerôle AWSServiceRoleForAmazonOpenSearchIngestionServiceou.
3. Sélectionnez le rôle, puis cliquez sur Supprimer.

Exemple de code pour Amazon OpenSearch Service

Ce chapitre contient un exemple de code courant pour l'utilisation d'Amazon OpenSearch Service : signature des requêtes HTTP dans divers langages de programmation, compression des corps des demandes HTTP et utilisation des AWS SDKs pour créer des domaines.

Rubriques

- Compatibilité des clients Elasticsearch
- Compression des requêtes HTTP dans Amazon OpenSearch Service
- Utilisation du AWS SDKs pour interagir avec Amazon OpenSearch Service

Compatibilité des clients Elasticsearch

Les dernières versions des clients Elasticsearch peuvent inclure des vérifications de licence ou de version qui entraînent une rupture artificielle de la compatibilité. Le tableau suivant contient des recommandations sur les versions de ces clients à utiliser pour une meilleure compatibilité avec OpenSearch Service.

> **Important**
>
> Ces versions du client sont obsolètes et ne sont pas mises à jour avec les dernières dépendances, notamment Log4j. Nous recommandons vivement d'utiliser les OpenSearch versions des clients lorsque cela est possible.

Client	Version recommandée
Client REST de bas niveau Java	7,13.4
Client REST de haut niveau Java	7,13.4
Client Elasticsearch Python	7,13.4
Client Ruby Elasticsearch	7.13.3
Client Node.js Elasticsearch	7,13.0

Compression des requêtes HTTP dans Amazon OpenSearch Service

Vous pouvez utiliser la compression gzip pour compresser les requêtes et les réponses HTTP dans Amazon OpenSearch Service. La compression gzip peut contribuer à réduire la taille de vos documents ainsi qu'à diminuer l'utilisation de la bande passante et la latence, ce qui permet d'améliorer les vitesses de transfert.

La compression gzip est prise en charge dans tous les domaines exécutant Elasticsearch 6.0 OpenSearch ou version ultérieure. Certains OpenSearch clients bénéficient d'une prise en charge intégrée de la compression gzip, et de nombreux langages de programmation disposent de bibliothèques qui simplifient le processus.

Activation de la compression gzip

À ne pas confondre avec OpenSearch les paramètres similaires, `http_compression.enabled` est spécifique au OpenSearch Service et active ou désactive la compression gzip dans un domaine. Domaines en cours d'exécution OpenSearch ou Elasticsearch 7. x, et activée par défaut dans les domaines gzip, et activée par défaut dans les domaines qui exécutent Elasticsearch 6. x l'ont désactivé par défaut.

Pour activer la compression gzip, envoyez la requête suivante :

```
PUT _cluster/settings
{
  "persistent" : {
    "http_compression.enabled": true
  }
}
```

Les requêtes adressées à `_cluster/settings` doivent être décompressées ; par conséquent, vous devrez peut-être utiliser un client distinct ou une requête HTTP standard pour mettre à jour les paramètres du cluster.

Pour vérifier que vous avez activé correctement la compression gzip, envoyez la requête suivante :

```
GET _cluster/settings?include_defaults=true
```

Assurez-vous que le paramètre suivant apparaît dans la réponse :

```
...
"http_compression": {
  "enabled": "true"
}
...
```

En-têtes obligatoires

Lorsque vous incluez un corps de requête compressé par gzip, conservez l'en-tête `Content-Type: application/json` standard, et ajoutez l'en-tête `Content-Encoding: gzip`. Pour accepter une réponse compressée par gzip, ajoutez également l'en-tête `Accept-Encoding: gzip`. Si un OpenSearch client prend en charge la compression gzip, il inclut probablement ces en-têtes automatiquement.

Exemple de code (Python 3)

L'exemple suivant utilise opensearch-py pour effectuer la compression et envoyer la requête. Ce code signe la demande à l'aide de vos informations d'identification IAM.

```
from opensearchpy import OpenSearch, RequestsHttpConnection
from requests_aws4auth import AWS4Auth
import boto3

host = '' # e.g. my-test-domain.us-east-1.es.amazonaws.com
region = '' # e.g. us-west-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

# Create the client.
search = OpenSearch(
    hosts = [{'host': host, 'port': 443}],
    http_auth = awsauth,
    use_ssl = True,
    verify_certs = True,
    http_compress = True, # enables gzip compression for request bodies
    connection_class = RequestsHttpConnection
)

document = {
```

```
  "title": "Moneyball",
  "director": "Bennett Miller",
  "year": "2011"
}

# Send the request.
print(search.index(index='movies', id='1', body=document, refresh=True))

# print(search.index(index='movies', doc_type='_doc', id='1', body=document,
 refresh=True))
```

Vous pouvez également spécifier les en-têtes appropriés, compresser vous-même le corps de la requête et utiliser une bibliothèque HTTP standard comme Requests. Ce code signe la requête en utilisant les informations d'identification HTTP de base, que votre domaine peut prendre en charge si vous utilisez le contrôle précis des accès.

```
import requests
import gzip
import json

base_url = '' # The domain with https:// and a trailing slash. For example, https://my-
test-domain.us-east-1.es.amazonaws.com/
auth = ('master-user', 'master-user-password') # For testing only. Don't store
 credentials in code.

headers = {'Accept-Encoding': 'gzip', 'Content-Type': 'application/json',
           'Content-Encoding': 'gzip'}

document = {
  "title": "Moneyball",
  "director": "Bennett Miller",
  "year": "2011"
}

# Compress the document.
compressed_document = gzip.compress(json.dumps(document).encode())

# Send the request.
path = 'movies/_doc?refresh=true'
url = base_url + path
response = requests.post(url, auth=auth, headers=headers, data=compressed_document)
print(response.status_code)
```

```
print(response.text)
```

Utilisation du AWS SDKs pour interagir avec Amazon OpenSearch Service

Cette section comprend des exemples d'utilisation de AWS SDKs pour interagir avec l'API de configuration Amazon OpenSearch Service. Ces exemples de code montrent comment créer, mettre à jour et supprimer des domaines OpenSearch de service.

Java

Cette section inclut des exemples pour les versions 1 et 2 de AWS SDK pour Java.

Version 2

Cet exemple utilise le OpenSearchClientBuilderconstructeur de la version 2 du AWS SDK pour Java pour créer un OpenSearch domaine, mettre à jour sa configuration et le supprimer. Supprimez la mise en commentaire des appels à `waitForDomainProcessing` (et mettez en commentaire les appels à `deleteDomain`) pour permettre au domaine d'être mis en ligne et utilisable.

```
package com.example.samples;

import java.util.concurrent.TimeUnit;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.opensearch.OpenSearchClient;
import software.amazon.awssdk.services.opensearch.model.ClusterConfig;
import software.amazon.awssdk.services.opensearch.model.EBSOptions;
import software.amazon.awssdk.services.opensearch.model.CognitoOptions;
import software.amazon.awssdk.services.opensearch.model.NodeToNodeEncryptionOptions;
import software.amazon.awssdk.services.opensearch.model.CreateDomainRequest;
import software.amazon.awssdk.services.opensearch.model.CreateDomainResponse;
import software.amazon.awssdk.services.opensearch.model.DescribeDomainRequest;
import software.amazon.awssdk.services.opensearch.model.UpdateDomainConfigRequest;
import software.amazon.awssdk.services.opensearch.model.UpdateDomainConfigResponse;
import software.amazon.awssdk.services.opensearch.model.DescribeDomainResponse;
import software.amazon.awssdk.services.opensearch.model.DeleteDomainRequest;
import software.amazon.awssdk.services.opensearch.model.DeleteDomainResponse;
import software.amazon.awssdk.services.opensearch.model.OpenSearchException;
import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
```

```
/**
 * Sample class demonstrating how to use the Amazon Web Services SDK for Java to
 create, update,
 * and delete Amazon OpenSearch Service domains.
 */

public class OpenSearchSample {

    public static void main(String[] args) {

     String domainName = "my-test-domain";

     // Build the client using the default credentials chain.
        // You can use the CLI and run `aws configure` to set access key, secret
        // key, and default region.

        OpenSearchClient client = OpenSearchClient.builder()
          // Unnecessary, but lets you use a region different than your default.
          .region(Region.US_EAST_1)
          // Unnecessary, but if desired, you can use a different provider chain.
          .credentialsProvider(DefaultCredentialsProvider.create())
                    .build();

        // Create a new domain, update its configuration, and delete it.
        createDomain(client, domainName);
        //waitForDomainProcessing(client, domainName);
        updateDomain(client, domainName);
        //waitForDomainProcessing(client, domainName);
        deleteDomain(client, domainName);
    }

    /**
     * Creates an Amazon OpenSearch Service domain with the specified options.
     * Some options require other Amazon Web Services resources, such as an Amazon
 Cognito user pool
     * and identity pool, whereas others require just an instance type or instance
     * count.
     *
     * @param client
     *            The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *            The name of the domain you want to create
     */
```

```
    public static void createDomain(OpenSearchClient client, String domainName) {

     // Create the request and set the desired configuration options

        try {

            ClusterConfig clusterConfig = ClusterConfig.builder()
                    .dedicatedMasterEnabled(true)
                    .dedicatedMasterCount(3)
                    // Small, inexpensive instance types for testing. Not
 recommended for production.
                    .dedicatedMasterType("t2.small.search")
                    .instanceType("t2.small.search")
                    .instanceCount(5)
                    .build();

            // Many instance types require EBS storage.
            EBSOptions ebsOptions = EBSOptions.builder()
                    .ebsEnabled(true)
                    .volumeSize(10)
                    .volumeType("gp2")
                    .build();

            NodeToNodeEncryptionOptions encryptionOptions =
 NodeToNodeEncryptionOptions.builder()
                    .enabled(true)
                    .build();

            CreateDomainRequest createRequest = CreateDomainRequest.builder()
                    .domainName(domainName)
                    .engineVersion("OpenSearch_1.0")
                    .clusterConfig(clusterConfig)
                    .ebsOptions(ebsOptions)
                    .nodeToNodeEncryptionOptions(encryptionOptions)
                    // You can uncomment this line and add your account ID, a
 username, and the
                    // domain name to add an access policy.
                    // .accessPolicies("{\"Version\":\"2012-10-17\",
\"Statement\":[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":
[\"arn:aws:iam::123456789012:user/user-name\"]},\"Action\":[\"es:*\"],\"Resource\":
\"arn:aws:es:region:123456789012:domain/domain-name/*\"}]}")
                    .build();

            // Make the request.
```

```
            System.out.println("Sending domain creation request...");
            CreateDomainResponse createResponse =
client.createDomain(createRequest);
            System.out.println("Domain status:
"+createResponse.domainStatus().toString());
            System.out.println("Domain ID:
"+createResponse.domainStatus().domainId());


        } catch (OpenSearchException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }

    /**
     * Updates the configuration of an Amazon OpenSearch Service domain with the
     * specified options. Some options require other Amazon Web Services resources,
such as an
     * Amazon Cognito user pool and identity pool, whereas others require just an
     * instance type or instance count.
     *
     * @param client
     *            The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *            The name of the domain to update
     */

    public static void updateDomain(OpenSearchClient client, String domainName) {

     // Updates the domain to use three data instances instead of five.
        // You can uncomment the Cognito line and fill in the strings to enable
Cognito
        // authentication for OpenSearch Dashboards.

        try {

            ClusterConfig clusterConfig = ClusterConfig.builder()
                    .instanceCount(5)
                    .build();

            CognitoOptions cognitoOptions = CognitoOptions.builder()
                    .enabled(true)
                    .userPoolId("user-pool-id")
```

```
                    .identityPoolId("identity-pool-id")
                    .roleArn("role-arn")
                    .build();

            UpdateDomainConfigRequest updateRequest =
UpdateDomainConfigRequest.builder()
                    .domainName(domainName)
                    .clusterConfig(clusterConfig)
                    //.cognitoOptions(cognitoOptions)
                    .build();

            System.out.println("Sending domain update request...");
            UpdateDomainConfigResponse updateResponse =
client.updateDomainConfig(updateRequest);
            System.out.println("Domain config:
"+updateResponse.domainConfig().toString());


        } catch (OpenSearchException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }

    /**
     * Deletes an Amazon OpenSearch Service domain. Deleting a domain can take
     * several minutes.
     *
     * @param client
     *             The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *             The name of the domain that you want to delete
     */

    public static void deleteDomain(OpenSearchClient client, String domainName) {

        try {

            DeleteDomainRequest deleteRequest = DeleteDomainRequest.builder()
                    .domainName(domainName)
                    .build();

            System.out.println("Sending domain deletion request...");
```

```
            DeleteDomainResponse deleteResponse =
client.deleteDomain(deleteRequest);
            System.out.println("Domain status: "+deleteResponse.toString());


        } catch (OpenSearchException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }

    /**
     * Waits for the domain to finish processing changes. New domains typically take
15-30 minutes
     * to initialize, but can take longer depending on the configuration. Most
updates to existing domains
     * take a similar amount of time. This method checks every 15 seconds and
finishes only when
     * the domain's processing status changes to false.
     *
     * @param client
     *            The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *            The name of the domain that you want to check
     */

    public static void waitForDomainProcessing(OpenSearchClient client, String
domainName) {
        // Create a new request to check the domain status.
        DescribeDomainRequest describeRequest = DescribeDomainRequest.builder()
                .domainName(domainName)
                .build();

        // Every 15 seconds, check whether the domain is processing.
        DescribeDomainResponse describeResponse =
client.describeDomain(describeRequest);
        while (describeResponse.domainStatus().processing()) {
            try {
                System.out.println("Domain still processing...");
                TimeUnit.SECONDS.sleep(15);
                describeResponse = client.describeDomain(describeRequest);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
```

```
        }

        // Once we exit that loop, the domain is available
        System.out.println("Amazon OpenSearch Service has finished processing
 changes for your domain.");
        System.out.println("Domain description: "+describeResponse.toString());
    }
}
```

Version 1

Cet exemple utilise le AWSElasticsearchClientBuilderconstructeur de la version 1 de AWS SDK pour Java pour créer un domaine Elasticsearch hérité, mettre à jour sa configuration et le supprimer. Supprimez la mise en commentaire des appels à `waitForDomainProcessing` (et mettez en commentaire les appels à `deleteDomain`) pour permettre au domaine d'être mis en ligne et utilisable.

```
package com.amazonaws.samples;

import java.util.concurrent.TimeUnit;
import com.amazonaws.auth.DefaultAWSCredentialsProviderChain;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.elasticsearch.AWSElasticsearch;
import com.amazonaws.services.elasticsearch.AWSElasticsearchClientBuilder;
import com.amazonaws.services.elasticsearch.model.CreateElasticsearchDomainRequest;
import com.amazonaws.services.elasticsearch.model.CreateElasticsearchDomainResult;
import com.amazonaws.services.elasticsearch.model.DeleteElasticsearchDomainRequest;
import com.amazonaws.services.elasticsearch.model.DeleteElasticsearchDomainResult;
import
 com.amazonaws.services.elasticsearch.model.DescribeElasticsearchDomainRequest;
import com.amazonaws.services.elasticsearch.model.DescribeElasticsearchDomainResult;
import com.amazonaws.services.elasticsearch.model.EBSOptions;
import com.amazonaws.services.elasticsearch.model.ElasticsearchClusterConfig;
import com.amazonaws.services.elasticsearch.model.ResourceNotFoundException;
import
 com.amazonaws.services.elasticsearch.model.UpdateElasticsearchDomainConfigRequest;
import
 com.amazonaws.services.elasticsearch.model.UpdateElasticsearchDomainConfigResult;
import com.amazonaws.services.elasticsearch.model.VolumeType;

/**
 * Sample class demonstrating how to use the Amazon Web Services SDK for Java to
 create, update,
```

```
 * and delete Amazon OpenSearch Service domains.
 */

public class OpenSearchSample {

    public static void main(String[] args) {

        final String domainName = "my-test-domain";

        // Build the client using the default credentials chain.
        // You can use the CLI and run `aws configure` to set access key, secret
        // key, and default region.
        final AWSElasticsearch client = AWSElasticsearchClientBuilder
                .standard()
                // Unnecessary, but lets you use a region different than your
 default.
                .withRegion(Regions.US_WEST_2)
                // Unnecessary, but if desired, you can use a different provider
 chain.
                .withCredentials(new DefaultAWSCredentialsProviderChain())
                .build();

        // Create a new domain, update its configuration, and delete it.
        createDomain(client, domainName);
        // waitForDomainProcessing(client, domainName);
        updateDomain(client, domainName);
        // waitForDomainProcessing(client, domainName);
        deleteDomain(client, domainName);
    }

    /**
     * Creates an Amazon OpenSearch Service domain with the specified options.
     * Some options require other Amazon Web Services resources, such as an Amazon
 Cognito user pool
     * and identity pool, whereas others require just an instance type or instance
     * count.
     *
     * @param client
     *            The client to use for the requests to Amazon OpenSearch Service
     * @param domainName
     *            The name of the domain you want to create
     */
    private static void createDomain(final AWSElasticsearch client, final String
 domainName) {
```

```
        // Create the request and set the desired configuration options
        CreateElasticsearchDomainRequest createRequest = new
 CreateElasticsearchDomainRequest()
                .withDomainName(domainName)
                .withElasticsearchVersion("7.10")
                .withElasticsearchClusterConfig(new ElasticsearchClusterConfig()
                        .withDedicatedMasterEnabled(true)
                        .withDedicatedMasterCount(3)
                        // Small, inexpensive instance types for testing. Not
 recommended for production
                        // domains.
                        .withDedicatedMasterType("t2.small.elasticsearch")
                        .withInstanceType("t2.small.elasticsearch")
                        .withInstanceCount(5))
                // Many instance types require EBS storage.
                .withEBSOptions(new EBSOptions()
                        .withEBSEnabled(true)
                        .withVolumeSize(10)
                        .withVolumeType(VolumeType.Gp2));
                // You can uncomment this line and add your account ID, a username,
 and the
                // domain name to add an access policy.
                // .withAccessPolicies("{\"Version\":\"2012-10-17\",
\"Statement\":[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":
[\"arn:aws:iam::123456789012:user/user-name\"]},\"Action\":[\"es:*\"],\"Resource\":
\"arn:aws:es:region:123456789012:domain/domain-name/*\"}]}")


        // Make the request.
        System.out.println("Sending domain creation request...");
        CreateElasticsearchDomainResult createResponse =
 client.createElasticsearchDomain(createRequest);
        System.out.println("Domain creation response from Amazon OpenSearch
 Service:");
        System.out.println(createResponse.getDomainStatus().toString());
    }

    /**
     * Updates the configuration of an Amazon OpenSearch Service domain with the
     * specified options. Some options require other Amazon Web Services resources,
 such as an
     * Amazon Cognito user pool and identity pool, whereas others require just an
     * instance type or instance count.
```

```
    *
    * @param client
    *            The client to use for the requests to Amazon OpenSearch Service
    * @param domainName
    *            The name of the domain to update
    */
   private static void updateDomain(final AWSElasticsearch client, final String
domainName) {
       try {
           // Updates the domain to use three data instances instead of five.
           // You can uncomment the Cognito lines and fill in the strings to enable
Cognito
           // authentication for OpenSearch Dashboards.
           final UpdateElasticsearchDomainConfigRequest updateRequest = new
UpdateElasticsearchDomainConfigRequest()
                   .withDomainName(domainName)
                   // .withCognitoOptions(new CognitoOptions()
                           // .withEnabled(true)
                           // .withUserPoolId("user-pool-id")
                           // .withIdentityPoolId("identity-pool-id")
                           // .withRoleArn("role-arn")
                   .withElasticsearchClusterConfig(new ElasticsearchClusterConfig()
                           .withInstanceCount(3));

           System.out.println("Sending domain update request...");
           final UpdateElasticsearchDomainConfigResult updateResponse = client
                   .updateElasticsearchDomainConfig(updateRequest);
           System.out.println("Domain update response from Amazon OpenSearch
Service:");
           System.out.println(updateResponse.toString());
       } catch (ResourceNotFoundException e) {
           System.out.println("Domain not found. Please check the domain name.");
       }
   }

   /**
    * Deletes an Amazon OpenSearch Service domain. Deleting a domain can take
    * several minutes.
    *
    * @param client
    *            The client to use for the requests to Amazon OpenSearch Service
    * @param domainName
    *            The name of the domain that you want to delete
    */
```

```
   private static void deleteDomain(final AWSElasticsearch client, final String
domainName) {
       try {
           final DeleteElasticsearchDomainRequest deleteRequest = new
DeleteElasticsearchDomainRequest()
                   .withDomainName(domainName);

           System.out.println("Sending domain deletion request...");
           final DeleteElasticsearchDomainResult deleteResponse =
client.deleteElasticsearchDomain(deleteRequest);
           System.out.println("Domain deletion response from Amazon OpenSearch
Service:");
           System.out.println(deleteResponse.toString());
       } catch (ResourceNotFoundException e) {
           System.out.println("Domain not found. Please check the domain name.");
       }
   }

   /**
    * Waits for the domain to finish processing changes. New domains typically take
15-30 minutes
    * to initialize, but can take longer depending on the configuration. Most
updates to existing domains
    * take a similar amount of time. This method checks every 15 seconds and
finishes only when
    * the domain's processing status changes to false.
    *
    * @param client
    *            The client to use for the requests to Amazon OpenSearch Service
    * @param domainName
    *            The name of the domain that you want to check
    */
   private static void waitForDomainProcessing(final AWSElasticsearch client, final
String domainName) {
       // Create a new request to check the domain status.
       final DescribeElasticsearchDomainRequest describeRequest = new
DescribeElasticsearchDomainRequest()
               .withDomainName(domainName);

       // Every 15 seconds, check whether the domain is processing.
       DescribeElasticsearchDomainResult describeResponse =
client.describeElasticsearchDomain(describeRequest);
       while (describeResponse.getDomainStatus().isProcessing()) {
           try {
```

```
                System.out.println("Domain still processing...");
                TimeUnit.SECONDS.sleep(15);
                describeResponse =
 client.describeElasticsearchDomain(describeRequest);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }

        // Once we exit that loop, the domain is available
        System.out.println("Amazon OpenSearch Service has finished processing
 changes for your domain.");
        System.out.println("Domain description response from Amazon OpenSearch
 Service:");
        System.out.println(describeResponse.toString());
    }
}
```

Python

Cet exemple utilise le client Python de OpenSearchServicebas niveau du AWS SDK for Python (Boto) pour créer un domaine, mettre à jour sa configuration et le supprimer.

```
import boto3
import botocore
from botocore.config import Config
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

my_config = Config(
    # Optionally lets you specify a region other than your default.
    region_name='us-west-2'
)

client = boto3.client('opensearch', config=my_config)

domainName = 'my-test-domain'  # The name of the domain
```

```
def createDomain(client, domainName):
    """Creates an Amazon OpenSearch Service domain with the specified options."""
    response = client.create_domain(
        DomainName=domainName,
        EngineVersion='OpenSearch_1.0',
        ClusterConfig={
            'InstanceType': 't2.small.search',
            'InstanceCount': 5,
            'DedicatedMasterEnabled': True,
            'DedicatedMasterType': 't2.small.search',
            'DedicatedMasterCount': 3
        },
        # Many instance types require EBS storage.
        EBSOptions={
            'EBSEnabled': True,
            'VolumeType': 'gp2',
            'VolumeSize': 10
        },
        AccessPolicies="{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":[\"arn:aws:iam::123456789012:user/user-name\"]},\"Action\":
[\"es:*\"],\"Resource\":\"arn:aws:es:us-west-2:123456789012:domain/my-test-domain/*
\"}]}",
        NodeToNodeEncryptionOptions={
            'Enabled': True
        }
    )
    print("Creating domain...")
    print(response)


def updateDomain(client, domainName):
    """Updates the domain to use three data nodes instead of five."""
    try:
        response = client.update_domain_config(
            DomainName=domainName,
            ClusterConfig={
                'InstanceCount': 3
            }
        )
        print('Sending domain update request...')
        print(response)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceNotFoundException':
```

```
            print('Domain not found. Please check the domain name.')
        else:
            raise error


def deleteDomain(client, domainName):
    """Deletes an OpenSearch Service domain. Deleting a domain can take several
 minutes."""
    try:
        response = client.delete_domain(
            DomainName=domainName
        )
        print('Sending domain deletion request...')
        print(response)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceNotFoundException':
            print('Domain not found. Please check the domain name.')
        else:
            raise error


def waitForDomainProcessing(client, domainName):
    """Waits for the domain to finish processing changes."""
    try:
        response = client.describe_domain(
            DomainName=domainName
        )
        # Every 15 seconds, check whether the domain is processing.
        while response["DomainStatus"]["Processing"] == True:
            print('Domain still processing...')
            time.sleep(15)
            response = client.describe_domain(
                DomainName=domainName)

        # Once we exit the loop, the domain is available.
        print('Amazon OpenSearch Service has finished processing changes for your
 domain.')
        print('Domain description:')
        print(response)

    except botocore.exceptions.ClientError as error:
        if error.response['Error']['Code'] == 'ResourceNotFoundException':
            print('Domain not found. Please check the domain name.')
```

```
        else:
            raise error


def main():
    """Create a new domain, update its configuration, and delete it."""
    createDomain(client, domainName)
    waitForDomainProcessing(client, domainName)
    updateDomain(client, domainName)
    waitForDomainProcessing(client, domainName)
    deleteDomain(client, domainName)
```

Nœud

Cet exemple utilise la version 3 du kit SDK pour JavaScript in Node.js pour OpenSearch créer un domaine, mettre à jour sa configuration et le supprimer.

```
var {
    OpenSearchClient,
    CreateDomainCommand,
    DescribeDomainCommand,
    UpdateDomainConfigCommand,
    DeleteDomainCommand
} = require("@aws-sdk/client-opensearch");
var sleep = require('sleep');

var client = new OpenSearchClient();

var domainName = 'my-test-domain'

// Create a new domain, update its configuration, and delete it.
createDomain(client, domainName)
waitForDomainProcessing(client, domainName)
updateDomain(client, domainName)
waitForDomainProcessing(client, domainName)
deleteDomain(client, domainName)

async function createDomain(client, domainName) {
    // Creates an Amazon OpenSearch Service domain with the specified options.
    var command = new CreateDomainCommand({
        DomainName: domainName,
        EngineVersion: 'OpenSearch_1.0',
        ClusterConfig: {
```

```
        'InstanceType': 't2.small.search',
        'InstanceCount': 5,
        'DedicatedMasterEnabled': 'True',
        'DedicatedMasterType': 't2.small.search',
        'DedicatedMasterCount': 3
        },
        EBSOptions:{
            'EBSEnabled': 'True',
            'VolumeType': 'gp2',
            'VolumeSize': 10
        },
        AccessPolicies: "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":[\"arn:aws:iam::123456789012:user/user-name\"]},\"Action\":
[\"es:*\"],\"Resource\":\"arn:aws:es:us-east-1:123456789012:domain/my-test-domain/*
\"}]}",
         NodeToNodeEncryptionOptions:{
            'Enabled': 'True'
        }
    });
    const response = await client.send(command);
    console.log("Creating domain...");
    console.log(response);
}

async function updateDomain(client, domainName) {
    // Updates the domain to use three data nodes instead of five.
    var command = new UpdateDomainConfigCommand({
        DomainName: domainName,
        ClusterConfig: {
        'InstanceCount': 3
        }
    });
    const response = await client.send(command);
    console.log('Sending domain update request...');
    console.log(response);
}

async function deleteDomain(client, domainName) {
    // Deletes an OpenSearch Service domain. Deleting a domain can take several
 minutes.
    var command = new DeleteDomainCommand({
        DomainName: domainName
    });
    const response = await client.send(command);
```

```
    console.log('Sending domain deletion request...');
    console.log(response);
}

async function waitForDomainProcessing(client, domainName) {
    // Waits for the domain to finish processing changes.
    try {
        var command = new DescribeDomainCommand({
            DomainName: domainName
        });
        var response = await client.send(command);

        while (response.DomainStatus.Processing == true) {
            console.log('Domain still processing...')
            await sleep(15000) // Wait for 15 seconds, then check the status again
            function sleep(ms) {
                return new Promise((resolve) => {
                    setTimeout(resolve, ms);
                });
            }
            var response = await client.send(command);
        }
        // Once we exit the loop, the domain is available.
        console.log('Amazon OpenSearch Service has finished processing changes for your
 domain.');
        console.log('Domain description:');
        console.log(response);

    } catch (error) {
        if (error.name === 'ResourceNotFoundException') {
            console.log('Domain not found. Please check the domain name.');
            }
    };
}
```

Indexation des données dans Amazon Service OpenSearch

Amazon OpenSearch Service utilisant une API REST, il existe de nombreuses méthodes pour indexer les documents. Vous pouvez utiliser des clients standard, tels que curl, ou n'importe quel langage de programmation capable d'envoyer des requêtes HTTP. Pour simplifier davantage le processus d'interaction avec celui-ci, OpenSearch Service a des clients pour de nombreux langages de programmation. Les utilisateurs avancés peuvent passer directement à l'étape the section called "Chargement de données de streaming dans le OpenSearch service".

Nous vous recommandons vivement d'utiliser Amazon OpenSearch Ingestion pour ingérer des données, un collecteur de données entièrement géré intégré au OpenSearch Service. Pour plus d'informations, consultez Amazon OpenSearch Ingestion.

Pour une introduction à l'indexation, consultez la OpenSearchdocumentation.

Restrictions de dénomination des index

OpenSearch Les index de service sont soumis aux restrictions de dénomination suivantes :

- Toutes les lettres doivent être en minuscules.
- Les noms d'index ne peuvent pas commencer par _ ni -.
- Les noms d'index ne peuvent pas contenir d'espaces, de virgules, :, ", *, +, /, \, |, ?, #, >, ni de <.

N'incluez pas d'informations sensibles dans les noms d'index, de type ou d'identifiant de document. OpenSearch Le service utilise ces noms dans ses Uniform Resource Identifiers (URIs). Les serveurs et les applications enregistrent souvent les requêtes HTTP, ce qui peut entraîner une exposition inutile des données si URIs elles contiennent des informations sensibles :

```
2018-10-03T23:39:43 198.51.100.14 200 "GET https://opensearch-domain/dr-jane-doe/flu-patients-2018/202-555-0100/ HTTP/1.1"
```

Même si vous ne disposez pas des autorisations nécessaires pour afficher le document JSON associé, vous pourriez déduire à partir de cette ligne de journal fictive que l'un des patients du Dr Untel, dont le numéro de téléphone est le 202-555-0100, a eu la grippe en 2018.

Si le OpenSearch Service détecte une adresse IP réelle ou perçue dans un nom d'index (par exemple,`my-index-12.34.56.78.91`), il masque l'adresse IP. Un appel à `_cat/indices` donne la réponse suivante :

```
green open my-index-x.x.x.x.91    soY19tBERoKo71WcEScidw 5 1 0 0   2kb  1kb
```

Pour éviter toute confusion inutile, évitez d'inclure des adresses IP dans les noms d'index.

Réduction de la taille des réponses

Les réponses du `_index` et `_bulk` APIs contiennent pas mal d'informations. Ces informations peuvent s'avérer utiles pour résoudre les problèmes liés aux demandes ou pour mettre en œuvre une logique de nouvelle tentative. Elles peuvent toutefois utiliser une grande quantité de bande passante. Dans cet exemple, l'indexation d'un document de 32 octets génère une réponse de 339 octets (en-têtes inclus) :

```
PUT opensearch-domain/more-movies/_doc/1
{"title": "Back to the Future"}
```

Réponse

```
{
  "_index": "more-movies",
  "_type": "_doc",
  "_id": "1",
  "_version": 4,
  "result": "updated",
  "_shards": {
    "total": 2,
    "successful": 2,
    "failed": 0
  },
  "_seq_no": 3,
  "_primary_term": 1
}
```

Cette taille de réponse peut sembler minime, mais si vous indexez 1 000 000 documents par jour (soit environ 11,5 documents par seconde), 339 octets par réponse représentent 10,17 Go de trafic de téléchargement par mois.

Si les coûts de transfert de données vous préoccupent, utilisez le `filter_path` paramètre pour réduire la taille de la réponse du OpenSearch service, mais veillez à ne pas filtrer les champs dont vous avez besoin pour identifier ou réessayer les demandes ayant échoué. Ces champs varient selon le client. Le `filter_path` paramètre fonctionne pour tous les OpenSearch services REST APIs, mais il est particulièrement utile APIs lorsque vous les appelez fréquemment, tels que le `_index` et `_bulk` APIs :

```
PUT opensearch-domain/more-movies/_doc/1?filter_path=result,_shards.total
{"title": "Back to the Future"}
```

Réponse

```
{
  "result": "updated",
  "_shards": {
    "total": 2
  }
}
```

Au lieu d'inclure des champs, vous pouvez exclure des champs à l'aide du préfixe `-`. `filter_path` prend également en charge les caractères génériques :

```
POST opensearch-domain/_bulk?filter_path=-took,-items.index._*
{ "index": { "_index": "more-movies", "_id": "1" } }
{"title": "Back to the Future"}
{ "index": { "_index": "more-movies", "_id": "2" } }
{"title": "Spirited Away"}
```

Réponse

```
{
  "errors": false,
  "items": [
    {
      "index": {
        "result": "updated",
        "status": 200
      }
    },
    {
      "index": {
```

```
        "result": "updated",
        "status": 200
      }
    }
  ]
}
```

Codecs d'index

Les codecs d'index déterminent la manière dont les champs stockés dans un index sont compressés et stockés sur disque. Le codec d'index est contrôlé par le `index.codec` paramètre statique, qui spécifie l'algorithme de compression. Ce paramètre a un impact sur la taille de la partition d'index et les performances opérationnelles.

Pour obtenir la liste des codecs pris en charge et leurs caractéristiques de performance, consultez la section Codecs pris en charge dans la documentation. OpenSearch

Lorsque vous choisissez un codec d'index, tenez compte des points suivants :

- Pour éviter les difficultés liées à la modification du paramètre de codec d'un index existant, testez une charge de travail représentative dans un environnement hors production avant d'utiliser un nouveau paramètre de codec. Pour plus d'informations, consultez la section Modification d'un codec d'index.
- Vous ne pouvez pas utiliser les codecs de compression Zstandard (`"index.codec": "zstd"`ou`"index.codec": "zstd_no_dict"`) pour les index K-nn ou Security Analytics.

Chargement de données de streaming dans Amazon OpenSearch Service

Vous pouvez utiliser OpenSearch Ingestion pour charger directement des données de streaming dans votre domaine Amazon OpenSearch Service, sans avoir besoin de recourir à des solutions tierces. Pour envoyer des données à OpenSearch Ingestion, vous configurez vos producteurs de données et le service fournit automatiquement les données au domaine ou à la collection que vous spécifiez. Pour commencer à utiliser OpenSearch Ingestion, voirthe section called “Tutoriel : Ingérer des données dans une collection”.

Vous pouvez toujours utiliser d'autres sources pour charger des données de streaming, telles qu'Amazon Data Firehose et Amazon CloudWatch Logs, qui disposent d'un support intégré pour

OpenSearch Service. D'autres, telles qu'Amazon S3, Amazon Kinesis Data Streams, et Amazon DynamoDB, utilisent des fonctions AWS Lambda comme gestionnaires d'événements. Les fonctions Lambda répondent aux nouvelles données en les traitant et en les diffusant dans votre domaine.

> **Note**
>
> Lambda prend en charge différents langages de programmation courants et est disponible dans la plupart des Régions AWS. Pour plus d'informations, consultez Getting started with Lambda dans le guide du AWS Lambda développeur et les points de terminaison AWS de service dans le. Références générales AWS

Chargement de données de streaming depuis OpenSearch Ingestion

Vous pouvez utiliser Amazon OpenSearch Ingestion pour charger des données dans un domaine OpenSearch de service. Vous configurez vos producteurs de données pour qu'ils envoient des données à OpenSearch Ingestion, qui les fournit automatiquement à la collection que vous spécifiez. Vous pouvez également configurer OpenSearch Ingestion pour transformer vos données avant de les livrer. Pour de plus amples informations, veuillez consulter OpenSearch Ingestion d'Amazon.

Chargement de données de streaming à partir d'Amazon S3

Vous pouvez utiliser Lambda pour envoyer des données vers votre domaine de OpenSearch service depuis Amazon S3. Les nouvelles données qui arrivent dans un compartiment S3 déclenchent l'envoi d'une notification d'événement à Lambda, qui exécute alors votre code personnalisé pour effectuer l'indexation.

Cette méthode de diffusion de données est extrêmement flexible. Vous pouvez indexer les métadonnées d'objet, ou si l'objet est en texte brut, analyser et indexer certains éléments du corps de l'objet. Cette section inclut des exemples de code Python simple qui utilisent des expressions régulières pour analyser un fichier journal et indexer les correspondances.

Prérequis

Avant de poursuivre, vous devez disposer des ressources suivantes.

Prérequis	Description
Compartiment Amazon S3	Pour en savoir plus, consultez Création de votre premier compartiment S3 dans le Guide de l'utilisateur Amazon Simple Storage Service. Le bucket doit résider dans la même région que votre domaine OpenSearch de service.
OpenSearch Domaine de service	Destination des données après leur traitement par votre fonction Lambda. Pour de plus amples informations, veuillez consulter the section called “ Création de domaines OpenSearch de service”.

Créer le package de déploiement Lambda

Les packages de déploiement sont des fichiers ZIP ou JAR qui contiennent votre code et ses dépendances. Cette section inclut des exemples de code Python. Pour les autres langages de programmation, consultez Packages de déploiement Lambda dans le Guide du développeur AWS Lambda .

1. Créez un répertoire. Dans cet exemple, nous utilisons le nom `s3-to-opensearch`.
2. Dans le répertoire, créez un fichier nommé `sample.py` :

```
import boto3
import re
import requests
from requests_aws4auth import AWS4Auth

region = '' # e.g. us-west-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

host = '' # the OpenSearch Service domain, e.g. https://search-mydomain.us-
west-1.es.amazonaws.com
index = 'lambda-s3-index'
datatype = '_doc'
url = host + '/' + index + '/' + datatype

headers = { "Content-Type": "application/json" }
```

```
s3 = boto3.client('s3')

# Regular expressions used to parse some simple log lines
ip_pattern = re.compile('(\d+\.\d+\.\d+\.\d+)')
time_pattern = re.compile('\[(\d+\/\w\w\w\/\d\d\d\d:\d\d:\d\d:\d\d\s-\d\d\d\d)\]')
message_pattern = re.compile('\"(.+)\"')

# Lambda execution starts here
def handler(event, context):
    for record in event['Records']:

        # Get the bucket name and key for the new file
        bucket = record['s3']['bucket']['name']
        key = record['s3']['object']['key']

        # Get, read, and split the file into lines
        obj = s3.get_object(Bucket=bucket, Key=key)
        body = obj['Body'].read()
        lines = body.splitlines()

        # Match the regular expressions to each line and index the JSON
        for line in lines:
            line = line.decode("utf-8")
            ip = ip_pattern.search(line).group(1)
            timestamp = time_pattern.search(line).group(1)
            message = message_pattern.search(line).group(1)

            document = { "ip": ip, "timestamp": timestamp, "message": message }
            r = requests.post(url, auth=awsauth, json=document, headers=headers)
```

Modifiez les variables des champs `region` et `host`.

3. Si vous ne l'avez pas encore fait, installez pip, puis installez les dépendances dans un nouveau répertoire `package` :

```
cd s3-to-opensearch

pip install --target ./package requests
pip install --target ./package requests_aws4auth
```

Boto3 est installé dans tous les environnements d'exécution Lambda et vous n'avez pas besoin de l'inclure dans votre package de déploiement.

4. Empaquetez le code d'application et les dépendances :

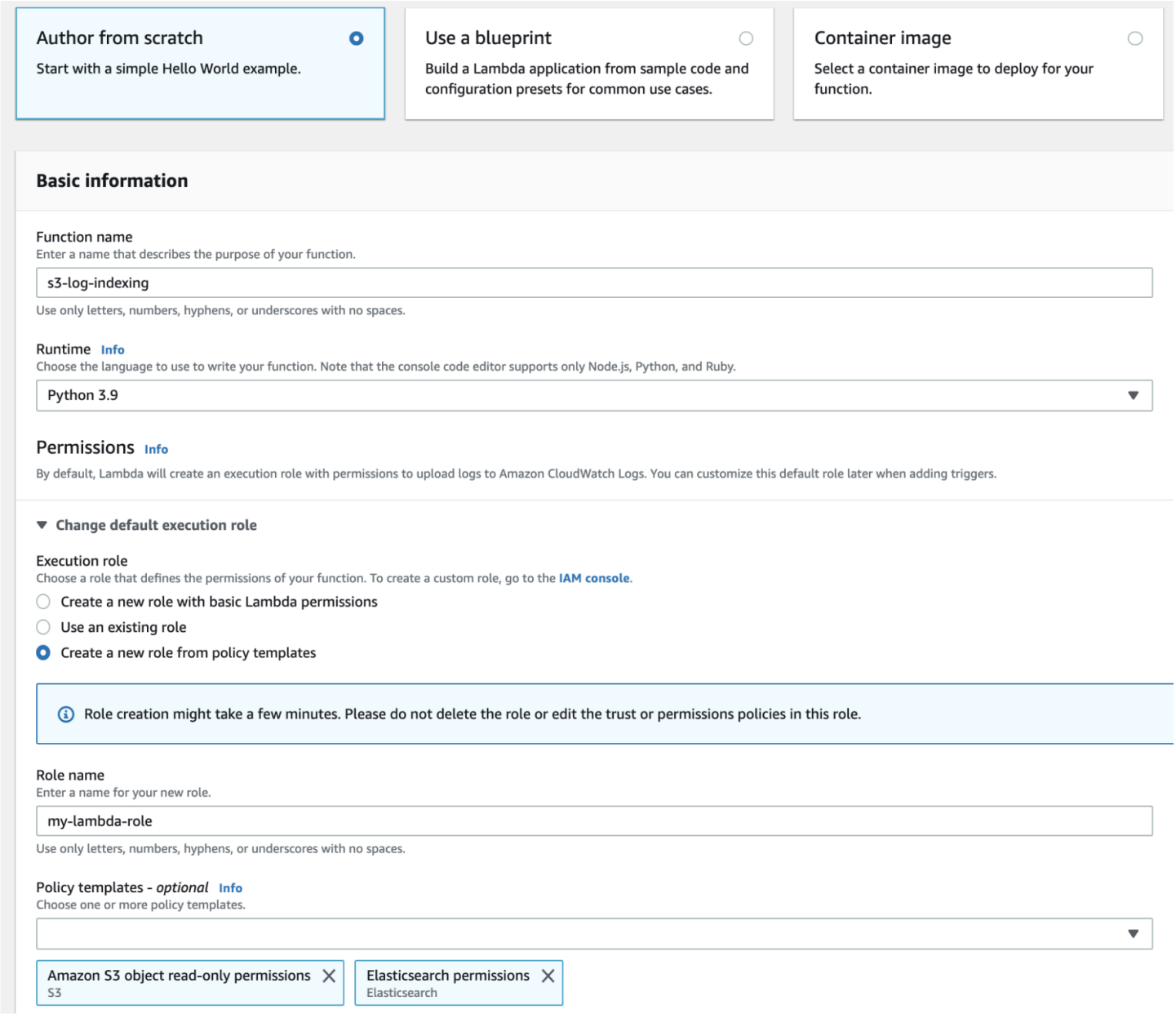
```
cd package
zip -r ../lambda.zip .

cd ..
zip -g lambda.zip sample.py
```

Créer la fonction Lambda

Après avoir créé le package de déploiement, vous pouvez créer la fonction Lambda. Lorsque vous créez une fonction, choisissez un nom, une exécution (par exemple, Python 3.8) et un rôle IAM. Le rôle IAM définit les autorisations pour votre fonction. Pour obtenir des instructions détaillées, consultez Création d'une fonction Lambda à l'aide de la console dans le Guide du développeur AWS Lambda .

Cet exemple suppose que vous utilisez la console. Choisissez Python 3.9 et un rôle doté des autorisations de lecture S3 et des autorisations d'écriture du OpenSearch service, comme illustré dans la capture d'écran suivante :



Une fois que vous avez créé la fonction, vous devez ajouter un déclencheur. Pour cet exemple, nous voulons que le code s'exécute chaque fois qu'un fichier journal arrive dans un compartiment S3 :

1. Choisissez Add trigger (Ajouter un déclencheur) et sélectionnez S3.
2. Choisissez votre compartiment.
3. Pour Event type (Type d'événement), choisissez PUT.
4. Pour Préfixe, tapez `logs/`.
5. Pour Suffixe, tapez `.log`.
6. Acceptez l'avertissement d'invocation récursive et choisissez Add (Ajouter).

Enfin, vous pouvez charger votre package de déploiement :

1. Choisissez Upload from (Charger à partir de) et .zip file (Fichier .zip), puis suivez les invites pour charger votre package de déploiement.
2. Au terme du chargement, modifiez les Paramètres d'exécution et remplacez le Gestionnaire par `sample.handler`. Ce paramètre indique à Lambda le fichier (`sample.py`) et la méthode (`handler`) à exécuter après un déclencheur.

À ce stade, vous disposez d'un ensemble complet de ressources : un compartiment pour les fichiers journaux, une fonction qui s'exécute chaque fois qu'un fichier journal est ajouté au compartiment, du code qui effectue l'analyse et l'indexation, et un domaine de OpenSearch service pour la recherche et la visualisation.

Test de la fonction Lambda

Après avoir créé la fonction, vous pouvez la tester en chargeant un fichier dans le compartiment Amazon S3. Créez un fichier nommé `sample.log` en utilisant les exemples de lignes de journal suivants :

```
12.345.678.90 - [10/Oct/2000:13:55:36 -0700] "PUT /some-file.jpg"
12.345.678.91 - [10/Oct/2000:14:56:14 -0700] "GET /some-file.jpg"
```

Chargez le fichier dans le dossier `logs` de votre compartiment S3. Pour obtenir des instructions, consultez Charger un objet dans votre compartiment dans le Guide de l'utilisateur Amazon Simple Storage Service.

Utilisez ensuite la console OpenSearch de service ou OpenSearch les tableaux de bord pour vérifier que l'`lambda-s3-index`index contient deux documents. Vous pouvez également effectuer une requête de recherche standard :

```
GET https://domain-name/lambda-s3-index/_search?pretty
{
  "hits" : {
    "total" : 2,
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "lambda-s3-index",
        "_type" : "_doc",
```

```
        "_id" : "vTYXaWIBJWV_TTkEuSDg",
        "_score" : 1.0,
        "_source" : {
          "ip" : "12.345.678.91",
          "message" : "GET /some-file.jpg",
          "timestamp" : "10/Oct/2000:14:56:14 -0700"
        }
      },
      {
        "_index" : "lambda-s3-index",
        "_type" : "_doc",
        "_id" : "vjYmaWIBJWV_TTkEuCAB",
        "_score" : 1.0,
        "_source" : {
          "ip" : "12.345.678.90",
          "message" : "PUT /some-file.jpg",
          "timestamp" : "10/Oct/2000:13:55:36 -0700"
        }
      }
    ]
  }
}
```

Chargement de données de streaming à partir d'Amazon Kinesis Data Streams

Vous pouvez charger des données de streaming depuis Kinesis Data Streams OpenSearch vers Service. Les nouvelles données qui arrivent dans le flux de données déclenchent l'envoi d'une notification d'événement à Lambda, qui exécute alors votre code personnalisé pour effectuer l'indexation. Cette section inclut des exemples de code Python simple.

Prérequis

Avant de poursuivre, vous devez disposer des ressources suivantes.

Prérequis	Description
Flux de données Amazon Kinesis	Source d'événement de votre fonction Lambda. Pour en savoir plus, consultez Kinesis Data Streams.

Prérequis	Description
OpenSearch Domaine de service	Destination des données après leur traitement par votre fonction Lambda. Pour plus d'informations, consultez the section called “ Création de domaines OpenSearch de service”

Prérequis	Description

Rôle IAM	Ce rôle doit disposer d'autorisations OpenSearch Service, Kinesis et Lambda de base, telles que les suivantes :
	
	JSON
	
	**{
	"Version": "2012-10-17",
	"Statement": [
	{
	"Effect": "Allow",
	"Action": [
	"es:ESHttpPost",
	"es:ESHttpPut",
	"logs:CreateLogGroup",
	"logs:CreateLogStream",
	"logs:PutLogEvents",
	"kinesis:GetShardIterator",
	"kinesis:GetRecords",
	"kinesis:DescribeStream",
	"kinesis:ListStreams"
	],
	"Resource": "*"
	}
	]
	}**
	
	Le rôle doit avoir la relation d'approbation suivante :
	
	JSON
	
	**{
	"Version": "2012-10-17",
	"Statement": [
	{
	"Effect": "Allow",
	"Principal": {
	"Service": "lambda.amazonaws.com"
	},**

Prérequis	Description
	"Action": "sts:AssumeRole" }] } Pour en savoir plus, consultez Création de rôles IAM dans le Guide de l'utilisateur IAM.

Créer la fonction Lambda

Suivez les instructions fournies dans the section called “Créer le package de déploiement Lambda”, mais créez un répertoire nommé `kinesis-to-opensearch` et utilisez le code suivant pour `sample.py` :

```
import base64
import boto3
import json
import requests
from requests_aws4auth import AWS4Auth

region = '' # e.g. us-west-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

host = '' # the OpenSearch Service domain, e.g. https://search-mydomain.us-
west-1.es.amazonaws.com
index = 'lambda-kine-index'
datatype = '_doc'
url = host + '/' + index + '/' + datatype + '/'

headers = { "Content-Type": "application/json" }

def handler(event, context):
    count = 0
    for record in event['Records']:
        id = record['eventID']
        timestamp = record['kinesis']['approximateArrivalTimestamp']
```

```
        # Kinesis data is base64-encoded, so decode here
        message = base64.b64decode(record['kinesis']['data'])

        # Create the JSON document
        document = { "id": id, "timestamp": timestamp, "message": message }
        # Index the document
        r = requests.put(url + id, auth=awsauth, json=document, headers=headers)
        count += 1
    return 'Processed ' + str(count) + ' items.'
```

Modifiez les variables des champs `region` et `host`.

Si vous ne l'avez pas encore fait, installez pip, puis utilisez les commandes suivantes pour installer vos dépendances :

```
cd kinesis-to-opensearch

pip install --target ./package requests
pip install --target ./package requests_aws4auth
```

Ensuite, suivez les instructions fournies dans the section called “Créer la fonction Lambda”, mais spécifiez le rôle IAM issu de the section called “Prérequis” et les paramètres suivants pour le déclencheur :

- Flux Kinesis : votre flux Kinesis
- Taille de lot : 100
- Position de départ : horizon Trim

Pour en savoir plus, consultez Présentation d'Amazon Kinesis Data Streams dans leGuide du développeur Amazon Kinesis Data Streams.

À ce stade, vous disposez d'un ensemble complet de ressources : un flux de données Kinesis, une fonction qui s'exécute une fois que le flux reçoit de nouvelles données et indexe ces données, et un domaine de OpenSearch service pour la recherche et la visualisation.

Tester la fonction Lambda

Après avoir créé la fonction, vous pouvez la tester en ajoutant un nouvel enregistrement dans le flux de données à l'aide de l' AWS CLI :

```
aws kinesis put-record --stream-name test --data "My test data." --partition-key
 partitionKey1 --region us-west-1
```

Utilisez ensuite la console OpenSearch de service ou OpenSearch les tableaux de bord pour vérifier qu'il `lambda-kine-index` contient un document. Vous pouvez également utiliser la demande suivante :

```
GET https://domain-name/lambda-kine-index/_search
{
  "hits" : [
    {
      "_index": "lambda-kine-index",
      "_type": "_doc",
      "_id":
 "shardId-000000000000:495835116157626994950129608214214566865294366804960870 42",
      "_score": 1,
      "_source": {
        "timestamp": 1523648740.051,
        "message": "My test data.",
        "id":
 "shardId-000000000000:495835116157626994950129608214214566865294366804960870 42"
      }
    }
  ]
}
```

Chargement de données de streaming à partir d'Amazon DynamoDB

Vous pouvez l'utiliser AWS Lambda pour envoyer des données vers votre domaine de OpenSearch service depuis Amazon DynamoDB. Les nouvelles données qui arrivent dans la table de la base de données déclenchent l'envoi d'une notification d'événement à Lambda, qui exécute ensuite votre code personnalisé pour effectuer l'indexation.

Prérequis

Avant de poursuivre, vous devez disposer des ressources suivantes.

Prérequis	Description
Tableau DynamoDB	Le tableau contient vos données sources. Pour plus d'informations, consultez Opérations de base sur les tables DynamoDBdans le Guide du développeur Amazon DynamoDB. La table doit résider dans la même région que votre domaine de OpenSearch service et avoir un flux défini sur Nouvelle image. Pour en savoir plus, consultez Activation d'un flux.
OpenSearch Domaine de service	Destination des données après leur traitement par votre fonction Lambda. Pour de plus amples informations, veuillez consulter the section called “ Création de domaines OpenSearch de service”.

Rôle IAM	Ce rôle doit disposer des autorisations d'exécution de base de OpenSearch service, DynamoDB et Lambda, telles que les suivantes :
	
	JSON
	
	{
	"Version": "2012-10-17",
	"Statement": [
	{
	"Effect": "Allow",
	"Action": [
	"es:ESHttpPost",
	"es:ESHttpPut",
	"dynamodb:DescribeStream",
	"dynamodb:GetRecords",
	"dynamodb:GetShardIterator",
	"dynamodb:ListStreams",
	"logs:CreateLogGroup",
	"logs:CreateLogStream",
	"logs:PutLogEvents"
	],
	"Resource": "*"
	}
	]
	}

Prérequis	Description
	Le rôle doit avoir la relation d'approbation suivante : JSON `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "Service": "lambda.amazonaws.com" }, "Action": "sts:AssumeRole" } ] }` Pour en savoir plus, consultez Création de rôles IAM dans le Guide de l'utilisateur IAM.

Créer la fonction Lambda

Suivez les instructions fournies dans the section called “Créer le package de déploiement Lambda”, mais créez un répertoire nommé `ddb-to-opensearch` et utilisez le code suivant pour `sample.py` :

```
import boto3
import requests
from requests_aws4auth import AWS4Auth

region = '' # e.g. us-east-1
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

host = '' # the OpenSearch Service domain, e.g. https://search-mydomain.us-
west-1.es.amazonaws.com
```

```
index = 'lambda-index'
datatype = '_doc'
url = host + '/' + index + '/' + datatype + '/'

headers = { "Content-Type": "application/json" }

def handler(event, context):
    count = 0
    for record in event['Records']:
        # Get the primary key for use as the OpenSearch ID
        id = record['dynamodb']['Keys']['id']['S']

        if record['eventName'] == 'REMOVE':
            r = requests.delete(url + id, auth=awsauth)
        else:
            document = record['dynamodb']['NewImage']
            r = requests.put(url + id, auth=awsauth, json=document, headers=headers)
        count += 1
    return str(count) + ' records processed.'
```

Modifiez les variables des champs `region` et `host`.

Si vous ne l'avez pas encore fait, installez pip, puis utilisez les commandes suivantes pour installer vos dépendances :

```
cd ddb-to-opensearch

pip install --target ./package requests
pip install --target ./package requests_aws4auth
```

Ensuite, suivez les instructions fournies dans the section called “Créer la fonction Lambda”, mais spécifiez le rôle IAM issu de the section called “Prérequis” et les paramètres suivants pour le déclencheur :

- Table : votre table DynamoDB
- Taille de lot : 100
- Position de départ : horizon Trim

Pour en savoir plus, consultez Traitement des nouveaux éléments avec DynamoDB Streams et Lambda dans le Guide du développeur Amazon DynamoDB.

À ce stade, vous disposez d'un ensemble complet de ressources : une table DynamoDB pour vos données source, un flux DynamoDB contenant les modifications apportées à la table, une fonction qui s'exécute après les modifications de vos données source et indexe ces modifications, et un domaine de service pour la recherche et la visualisation. OpenSearch

Test de la fonction Lambda

Après avoir créé la fonction, vous pouvez la tester en ajoutant un nouvel élément dans la table DynamoDB à l'aide de l'interface AWS CLI :

```
aws dynamodb put-item --table-name test --item '{"director": {"S": "Kevin
 Costner"},"id": {"S": "00001"},"title": {"S": "The Postman"}}' --region us-west-1
```

Utilisez ensuite la console OpenSearch de service ou OpenSearch les tableaux de bord pour vérifier qu'il `lambda-index` contient un document. Vous pouvez également utiliser la demande suivante :

```
GET https://domain-name/lambda-index/_doc/00001
{
    "_index": "lambda-index",
    "_type": "_doc",
    "_id": "00001",
    "_version": 1,
    "found": true,
    "_source": {
        "director": {
            "S": "Kevin Costner"
        },
        "id": {
            "S": "00001"
        },
        "title": {
            "S": "The Postman"
        }
    }
}
```

Chargement de données de streaming depuis Amazon Data Firehose

Firehose prend en charge le OpenSearch service en tant que destination de livraison. Pour savoir comment charger des données de streaming dans OpenSearch Service, consultez les sections

Creating a Kinesis Data Firehose Delivery Stream OpenSearch et Choose Service for Your Destination dans le manuel Amazon Data Firehose Developer Guide.

Avant de charger des données dans OpenSearch Service, vous devrez peut-être effectuer des transformations sur les données. Pour en savoir plus sur l'utilisation des fonctions Lambda pour effectuer cette tâche, consultez Transformation de données Amazon Kinesis Data Firehose dans le même guide.

Lorsque vous configurez un flux de diffusion, Firehose propose un rôle IAM « en un clic » qui lui donne l'accès aux ressources dont il a besoin pour envoyer des données au OpenSearch Service, sauvegarder des données sur Amazon S3 et transformer des données à l'aide de Lambda. En raison de la complexité du processus de création manuelle d'un tel rôle, nous vous recommandons d'utiliser le rôle fourni.

Chargement de données de streaming depuis Amazon CloudWatch

Vous pouvez charger des données de streaming depuis CloudWatch Logs vers votre domaine OpenSearch de service à l'aide d'un abonnement CloudWatch Logs. Pour plus d'informations sur les CloudWatch abonnements Amazon, consultez la section Traitement en temps réel des données de journal dans le cadre des abonnements. Pour obtenir des informations de configuration, consultez la section Streaming CloudWatch Logs to Amazon OpenSearch Service dans le manuel Amazon CloudWatch Developer Guide.

Chargement de données de streaming depuis AWS IoT

Vous pouvez envoyer des données à l' AWS IoT aide de règles. Pour en savoir plus, consultez l'OpenSearchaction décrite dans le guide du AWS IoT développeur.

Chargement de données dans Amazon OpenSearch Service avec Logstash

La version open source de Logstash (Logstash OSS) fournit un moyen pratique d'utiliser l'API en masse pour télécharger des données dans votre domaine Amazon Service. OpenSearch Le service prend en charge tous les plug-ins d'entrée Logstash standard, y compris le plug-in d'entrée Amazon S3. OpenSearch Le service prend en charge le plug-in logstash-output-opensearchde sortie, qui prend en charge à la fois l'authentification de base et les informations d'identification IAM. Le plugin fonctionne avec les versions 8.1 et inférieures de Logstash OSS.

Configuration

La configuration de Logstash varie en fonction du type d'authentification utilisé par votre domaine.

Quelle que soit la méthode d'authentification que vous utilisez, vous devez configurer `ecs_compatibility` comme `disabled` dans la section de sortie du fichier de configuration. Logstash 8.0 a introduit une modification novatrice dans laquelle tous les plugins sont exécutés dans Mode de compatibilité ECS par défaut. Vous devez remplacer la valeur par défaut pour conserver un comportement hérité.

Configuration du contrôle précis des accès

Si votre domaine OpenSearch de service utilise un contrôle d'accès précis avec authentification HTTP de base, la configuration est similaire à celle de tout autre OpenSearch cluster. L'entrée de cet exemple de fichier de configuration repose sur la version open source de Filebeat (Filebeat OSS) :

```
input {
  beats  {
    port => 5044
  }
}

output {
  opensearch {
    hosts       => "https://domain-endpoint:443"
    user        => "my-username"
    password    => "my-password"
    index       => "logstash-logs-%{+YYYY.MM.dd}"
    ecs_compatibility => disabled
    ssl_certificate_verification => false
  }
}
```

La configuration varie selon l'application Beats et le cas d'utilisation, mais votre configuration Filebeat OSS peut être semblable à la suivante :

```
filebeat.inputs:
- type: log
  enabled: true
  paths:
    - /path/to/logs/dir/*.log
```

```
filebeat.config.modules:
  path: ${path.config}/modules.d/*.yml
  reload.enabled: false
setup.ilm.enabled: false
setup.ilm.check_exists: false
setup.template.settings:
  index.number_of_shards: 1
output.logstash:
  hosts: ["logstash-host:5044"]
```

Configuration de l'IAM

Si votre domaine utilise une politique d'accès au domaine basée sur IAM ou un contrôle d'accès précis avec un utilisateur principal, vous devez signer toutes les demandes adressées au OpenSearch Service à l'aide des informations d'identification IAM. La politique basée sur l'identité suivante accorde toutes les requêtes HTTP aux sous-ressources de votre domaine.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:region:aws-account-id:domain/domain-name/*"
    }
  ]
}
```

Pour configurer votre configuration Logstash, modifiez votre fichier de configuration afin d'utiliser le plugin pour sa sortie. L'entrée de cet exemple de fichier de configuration repose sur les fichiers d'un compartiment S3 :

```
input {
  s3 {
    bucket => "amzn-s3-demo-"
    region => "us-east-1"
```

```
  }
}

output {
  opensearch {
    hosts => ["domain-endpoint:443"]
    auth_type => {
      type => 'aws_iam'
      aws_access_key_id => 'your-access-key'
      aws_secret_access_key => 'your-secret-key'
      region => 'us-east-1'
      }
      index  => "logstash-logs-%{+YYYY.MM.dd}"
      ecs_compatibility => disabled
  }
}
```

Si vous ne souhaitez pas fournir vos informations d'identification IAM dans le fichier de configuration, vous pouvez les exporter (ou exécuter `aws configure`) :

```
export AWS_ACCESS_KEY_ID="your-access-key"
export AWS_SECRET_ACCESS_KEY="your-secret-key"
export AWS_SESSION_TOKEN="your-session-token"
```

Si votre domaine de OpenSearch service se trouve dans un VPC, la machine Logstash OSS doit pouvoir se connecter au VPC et avoir accès au domaine via les groupes de sécurité VPC. Pour de plus amples informations, veuillez consulter the section called “À propos des stratégies d'accès pour les domaines de VPC”.

Recherche de données dans Amazon OpenSearch Service

Il existe plusieurs méthodes courantes pour rechercher des documents dans Amazon OpenSearch Service, notamment les recherches par URI et les recherches dans le corps des requêtes. OpenSearch Le service offre des fonctionnalités supplémentaires qui améliorent l'expérience de recherche, telles que les packages personnalisés, le support SQL et la recherche asynchrone. Pour une référence complète OpenSearch sur l'API de recherche, consultez la OpenSearchdocumentation.

> **Note**
>
> Les exemples de requêtes suivants fonctionnent avec OpenSearch APIs. Certaines demandes peuvent ne pas fonctionner avec des versions antérieures d'Elasticsearch.

Rubriques

- Recherches d'URI
- Recherches dans le corps de la demande
- Pagination des résultats de recherche
- Langage de requête Dashboards
- Importation et gestion de packages dans Amazon OpenSearch Service
- Interrogation de vos données Amazon OpenSearch Service avec SQL
- Recherche des k plus proches voisins (k-NN) dans Amazon Service OpenSearch
- Recherche entre clusters dans Amazon Service OpenSearch
- Learning to Rank pour Amazon OpenSearch Service
- Recherche asynchrone dans Amazon Service OpenSearch
- Recherche ponctuelle dans le temps dans Amazon OpenSearch Service
- Recherche sémantique dans Amazon Service OpenSearch
- Recherche par segment simultanée dans Amazon OpenSearch Service
- Génération de requêtes en langage naturel dans Amazon OpenSearch Service

Recherches d'URI

Les recherches d'URI (Universal Resource Identifier, Identificateur de ressource uniforme) constituent la méthode de recherche la plus simple. Dans une recherche d'URI, vous spécifiez la requête en tant que paramètre de demande HTTP :

```
GET https://search-my-domain.us-west-1.es.amazonaws.com/_search?q=house
```

Un exemple de réponse peut ressembler à ce qui suit :

```
{
  "took": 25,
  "timed_out": false,
  "_shards": {
    "total": 10,
    "successful": 10,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 85,
      "relation": "eq",
    },
    "max_score": 6.6137657,
    "hits": [
      {
        "_index": "movies",
        "_type": "movie",
        "_id": "tt0077975",
        "_score": 6.6137657,
        "_source": {
          "directors": [
            "John Landis"
          ],
          "release_date": "1978-07-27T00:00:00Z",
          "rating": 7.5,
          "genres": [
            "Comedy",
            "Romance"
          ],
```

```
          "image_url": "http://ia.media-imdb.com/images/M/
MV5BMTY2OTQxNTc1OF5BMl5BanBnXkFtZTYwNjA3NjI5._V1_SX400_.jpg",
          "plot": "At a 1962 College, Dean Vernon Wormer is determined to expel the
 entire Delta Tau Chi Fraternity, but those troublemakers have other plans for him.",
          "title": "Animal House",
          "rank": 527,
          "running_time_secs": 6540,
          "actors": [
            "John Belushi",
            "Karen Allen",
            "Tom Hulce"
          ],
          "year": 1978,
          "id": "tt0077975"
        }
      },
      ...
    ]
  }
}
```

Par défaut, cette requête recherche le mot house dans tous les champs de tous les index. Pour limiter la recherche, spécifiez un index (`movies`) et un champ de document (`title`) dans l'URI :

```
GET https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search?q=title:house
```

Vous pouvez inclure des paramètres supplémentaires dans la demande, mais les paramètres pris en charge ne fournissent qu'un petit sous-ensemble des options de OpenSearch recherche. La demande suivante renvoie 20 résultats (au lieu de la valeur par défaut de 10) et trie les résultats par année (plutôt que par `_score`) :

```
GET https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search?
q=title:house&size=20&sort=year:desc
```

Recherches dans le corps de la demande

Pour effectuer des recherches plus complexes, utilisez le corps de la demande HTTP et le langage dédié à un domaine (DSL) OpenSearch pour les requêtes. La requête DSL vous permet de spécifier l'ensemble des options de recherche OpenSearch .

> **Note**
>
> Vous ne pouvez pas inclure de caractères spéciaux Unicode dans la valeur d'un champ de texte, sinon la valeur sera analysée sous la forme de plusieurs valeurs séparées par le caractère spécial. Cette analyse incorrecte peut entraîner un filtrage involontaire des documents et potentiellement compromettre le contrôle de leur accès. Pour plus d'informations, consultez la section Remarque sur les caractères spéciaux Unicode dans les champs de texte de la OpenSearch documentation.

La requête `match` suivante est similaire à l'exemple de recherche d'URI final :

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
  "sort": {
    "year": {
      "order": "desc"
    }
  },
  "query": {
    "query_string": {
      "default_field": "title",
      "query": "house"
    }
  }
}
```

> **Note**
>
> L'API `_search` accepte les demandes `GET` et `POST` HTTP pour les recherches dans le corps des demandes, mais tous les clients HTTP ne prennent pas en charge l'ajout d'un corps de demande à une demande `GET`. `POST` constitue le choix le plus universel.

Dans de nombreux cas, il se peut que vous vouliez effectuer une recherche dans plusieurs champs, mais pas dans tous. Utilisez la requête `multi_match` :

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
```

```
  "size": 20,
  "query": {
    "multi_match": {
      "query": "house",
      "fields": ["title", "plot", "actors", "directors"]
    }
  }
}
```

Optimisation des champs

Vous pouvez améliorer la pertinence des recherches en « optimisant » certains champs à l'aide de multiplicateurs qui affectent plus de poids aux correspondances se trouvant dans un champ qu'à celles se trouvant dans d'autres champs. Dans l'exemple suivant, une correspondance pour john dans le champ `title` influence `_score` deux fois plus qu'une correspondance dans le champ `plot` et quatre fois plus qu'une correspondance dans les champs `actors` ou `directors`. Résultat : des films tels que John Wick et John Carter figurent dans les premiers résultats de recherche, tandis que des films avec John Travolta figurent dans les derniers résultats.

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
  "query": {
    "multi_match": {
      "query": "john",
      "fields": ["title^4", "plot^2", "actors", "directors"]
    }
  }
}
```

Mise en évidence des résultats de recherche

L'`highlight`option indique OpenSearch de renvoyer un objet supplémentaire à l'intérieur du `hits` tableau si la requête correspond à un ou plusieurs champs :

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
  "query": {
    "multi_match": {
      "query": "house",
```

```
      "fields": ["title^4", "plot^2", "actors", "directors"]
    }
  },
  "highlight": {
    "fields": {
      "plot": {}
    }
  }
}
```

Si la requête correspond au contenu du champ `plot`, le résultat peut ressembler à ce qui suit :

```
{
  "_index": "movies",
  "_type": "movie",
  "_id": "tt0091541",
  "_score": 11.276199,
  "_source": {
    "directors": [
      "Richard Benjamin"
    ],
    "release_date": "1986-03-26T00:00:00Z",
    "rating": 6,
    "genres": [
      "Comedy",
      "Music"
    ],
    "image_url": "http://ia.media-imdb.com/images/M/
MV5BMTIzODEzODE2OF5BMl5BanBnXkFtZTcwNjQ3ODcyMQ@@._V1_SX400_.jpg",
    "plot": "A young couple struggles to repair a hopelessly dilapidated house.",
    "title": "The Money Pit",
    "rank": 4095,
    "running_time_secs": 5460,
    "actors": [
      "Tom Hanks",
      "Shelley Long",
      "Alexander Godunov"
    ],
    "year": 1986,
    "id": "tt0091541"
  },
  "highlight": {
    "plot": [
```

```
      "A young couple struggles to repair a hopelessly dilapidated <em>house</em>."
    ]
  }
}
```

Par défaut, OpenSearch place la chaîne correspondante dans des balises, fournit jusqu'à 100 caractères de contexte autour de la correspondance et divise le contenu en phrases en identifiant les signes de ponctuation, les espaces, les tabulations et les sauts de ligne. Tous ces paramètres sont personnalisables :

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "size": 20,
  "query": {
    "multi_match": {
      "query": "house",
      "fields": ["title^4", "plot^2", "actors", "directors"]
    }
  },
  "highlight": {
    "fields": {
      "plot": {}
    },
    "pre_tags": "<strong>",
    "post_tags": "</strong>",
    "fragment_size": 200,
    "boundary_chars": ".,!? "
  }
}
```

API Count

Si vous n'êtes pas intéressé par le contenu de vos documents et voulez simplement connaître le nombre de correspondances, vous pouvez utiliser l'API `_count` au lieu de l'API `_search`. La demande suivante utilise la requête `query_string` pour identifier les comédies romantiques :

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_count
{
  "query": {
    "query_string": {
      "default_field": "genres",
```

```
      "query": "romance AND comedy"
    }
  }
}
```

Un exemple de réponse peut ressembler à ce qui suit :

```
{
  "count": 564,
  "_shards": {
    "total": 5,
    "successful": 5,
    "skipped": 0,
    "failed": 0
  }
}
```

Pagination des résultats de recherche

Si vous devez afficher un grand nombre de résultats de recherche, vous pouvez implémenter la pagination en utilisant différentes méthodes.

Point dans le temps

La fonction point dans le temps (PIT) est un type de recherche qui vous permet d'exécuter différentes requêtes sur un ensemble de données fixe dans le temps. Il s'agit de la méthode de pagination préférée OpenSearch, en particulier pour la pagination profonde. Vous pouvez utiliser PIT avec la version OpenSearch de service 2.5 ou ultérieure. Pour plus d'informations sur le PIT, voir???.

Les `size` paramètres `from` et

Le moyen le plus simple de paginer est d'utiliser les `size` paramètres `from` et. La demande suivante renvoie les résultats 20 à 39 de la liste des résultats de recherche indexée sur zéro :

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/movies/_search
{
  "from": 20,
  "size": 20,
  "query": {
    "multi_match": {
      "query": "house",
```

```
      "fields": ["title^4", "plot^2", "actors", "directors"]
    }
  }
}
```

Pour plus d'informations sur la pagination de recherche, voir Paginer les résultats dans la OpenSearch documentation.

Langage de requête Dashboards

Vous pouvez utiliser le langage de requête des tableaux de bord (DQL) pour rechercher des données et des visualisations dans les tableaux de bord. OpenSearch DQL utilise quatre principaux types de requêtes : termes, booléens, date et plage, et champ imbriqué.

Requête de termes

Une requête de termes nécessite que vous spécifiiez le terme que vous recherchez.

Pour effectuer une requête de termes, saisissez ce qui suit :

```
host:www.example.com
```

Requête booléenne

Vous pouvez utiliser les opérateurs booléens `AND`, `or`, et `not` pour combiner plusieurs requêtes.

Pour effectuer une requête booléenne, collez ce qui suit :

```
host.keyword:www.example.com and response.keyword:200
```

Requête de date et plage

Vous pouvez utiliser une requête de date et plage pour trouver une date avant ou après votre requête.

- > indique une recherche d'une date postérieure à la date spécifiée.
- < indique une recherche d'une date antérieure à la date spécifiée.

```
@timestamp > "2020-12-14T09:35:33"
```

Requête de champ imbriqué

Si vous avez un document avec des champs imbriqués, vous devez spécifier les parties du document que vous voulez récupérer. Voici un exemple de document qui contient des champs imbriqués :

```
{"NBA players":[
    {"player-name": "Lebron James",
      "player-position": "Power forward",
      "points-per-game": "30.3"
    },
    {"player-name": "Kevin Durant",
      "player-position": "Power forward",
      "points-per-game": "27.1"
    },
    {"player-name": "Anthony Davis",
      "player-position": "Power forward",
      "points-per-game": "23.2"
    },
    {"player-name": "Giannis Antetokounmpo",
      "player-position": "Power forward",
      "points-per-game":"29.9"
    }
  ]
}
```

Pour récupérer un champ spécifique à l'aide de DQL, collez ce qui suit :

```
NBA players: {player-name: Lebron James}
```

Pour récupérer plusieurs objets du document imbriqué, collez ce qui suit :

```
NBA players: {player-name: Lebron James} and NBA players: {player-name: Giannis
 Antetokounmpo}
```

Pour effectuer une recherche dans une plage, collez ce qui suit :

```
NBA players: {player-name: Lebron James} and NBA players: {player-name: Giannis
 Antetokounmpo and < 30}
```

Si votre document comporte un objet imbriqué dans un autre objet, vous pouvez toujours récupérer les données en spécifiant tous les niveaux. Pour ce faire, collez ce qui suit :

```
Top-Power-forwards.NBA players: {player-name:Lebron James}
```

Importation et gestion de packages dans Amazon OpenSearch Service

Amazon OpenSearch Service vous permet de charger des fichiers de dictionnaire personnalisés, comme des fichiers de mots vides et de synonymes, et d'associer des plug-ins à votre domaine. Ces plugins peuvent être préemballés, personnalisés ou tiers, ce qui vous donne la flexibilité d'étendre les fonctionnalités de votre domaine. Le terme générique désignant tous ces types de fichiers est « packages ».

- Les fichiers de dictionnaire aident à affiner les résultats des recherches en indiquant d' OpenSearch ignorer les mots fréquents ou de traiter des termes similaires, comme « crème glacée », « sorbet » et « glace » comme des équivalents. Ils peuvent également améliorer la recherche de radical, comme illustré avec le plugin d'analyse japonais (kuromoji).
- Les plugins préemballés fournissent des fonctionnalités intégrées, telles que le plugin Amazon Personalize pour des résultats de recherche personnalisés. Ces plugins utilisent le type de `ZIP-PLUGIN` package. Pour de plus amples informations, veuillez consulter the section called “Plugins par version du moteur”.
- Les plugins personnalisés et tiers vous permettent d'ajouter des fonctionnalités personnalisées ou de les intégrer à des systèmes externes, ce qui offre encore plus de flexibilité à votre domaine. Comme les plugins préemballés, vous téléchargez des plugins personnalisés sous forme de `ZIP-PLUGIN` packages. Pour les plug-ins tiers, vous devez également importer la licence et les fichiers de configuration du plug-in sous forme de packages distincts, puis les associer tous au domaine.

  Pour plus d’informations, consultez les rubriques suivantes :

  - the section called “Plug-ins personnalisés”
  - the section called “Plug-ins tiers”

> **Note**
>
> Vous pouvez associer un maximum de 20 plugins à un seul domaine. Cette limite inclut tous les types de plugins, qu'ils soient facultatifs, tiers ou personnalisés.

Rubriques

- Autorisations requises

- Chargement des packages dans Amazon S3
- Importation et association de packages
- Utilisation de packages avec OpenSearch
- Mise à jour de packages
- Mise à jour manuelle des index avec un nouveau dictionnaire
- Dissociation et suppression de packages
- Gestion des plugins personnalisés dans Amazon OpenSearch Service
- Installation de plugins tiers dans Amazon OpenSearch Service

Autorisations requises

Les utilisateurs sans accès administrateur doivent effectuer certaines actions AWS Identity and Access Management (IAM) afin de pouvoir gérer les packages :

- `es:CreatePackage`— Crée un package
- `es:DeletePackage`— Supprimer un package
- `es:AssociatePackage`— Associe un package à un domaine
- `es:DissociatePackage`— Dissociation d'un package d'un domaine

Vous devez également disposer d'autorisations sur le chemin du compartiment Amazon S3 ou l'objet où réside le package personnalisé.

Accordez toutes les autorisations dans IAM, et non dans la stratégie d'accès au domaine. Pour en savoir plus, consultez the section called “Gestion de l’identité et des accès”.

Chargement des packages dans Amazon S3

Cette section explique comment télécharger des packages de dictionnaires personnalisés, étant donné que des packages de plugins préemballés sont déjà installés. Pour pouvoir associer un dictionnaire personnalisé à votre domaine, vous devez le charger dans un compartiment Amazon S3. Pour en savoir plus, consultez Chargement d'objets dans le Guide de l'utilisateur Amazon Simple Storage Service. Les plug-ins pris en charge n'ont pas besoin d'être chargés.

Si votre dictionnaire contient des informations sensibles, configurez le chiffrement côté serveur avec des clés gérées par S3 lorsque vous le chargez. OpenSearch Le service n'a pas accès aux fichiers S3 que vous protégez à l'aide d'une AWS KMS clé dans S3.

Après avoir chargé le fichier, notez son chemin d'accès S3. Le format du chemin d'accès est `s3://amzn-s3-demo-bucket/file-path/file-name`.

Vous pouvez utiliser le fichier de synonymes suivant à des fins de test. Enregistrez-le sous `synonyms.txt`.

```
danish, croissant, pastry
ice cream, gelato, frozen custard
sneaker, tennis shoe, running shoe
basketball shoe, hightop
```

Certains dictionnaires, notamment les dictionnaires Hunspell, utilisent plusieurs fichiers et nécessitent leurs propres répertoires sur le système de fichiers. Actuellement, OpenSearch Service prend uniquement en charge les dictionnaires à fichier unique.

Importation et association de packages

La console constitue le moyen le plus simple d'importer un dictionnaire personnalisé dans OpenSearch Service. Lorsque vous importez un dictionnaire à partir d'Amazon S3, OpenSearch Service stocke sa propre copie du package et chiffre automatiquement cette copie à l'aide de l'algorithme AES-256 avec OpenSearch des clés gérées par le service.

Les plugins optionnels sont déjà préinstallés dans OpenSearch Service. Vous n'avez donc pas besoin de les télécharger vous-même, mais vous devez associer un plug-in à un domaine. Les plug-ins disponibles sont répertoriés sur l'écran Packages de la console.

Importation et association d'un package à un domaine

1. Dans la console Amazon OpenSearch Service, choisissez Packages.
2. Choisissez Import package (Importer un package).
3. Attribuez un nom descriptif au package.
4. Indiquez le chemin d'accès S3 au fichier, puis choisissez Import (Importer).
5. Revenez à l'écran Packages.
6. Lorsque le statut du package est Available (Disponible), sélectionnez-le.
7. Choisissez Associer à un domaine.
8. Sélectionnez un domaine, puis cliquez sur Suivant. Passez en revue les packages et choisissez Associer.
9. Dans le panneau de navigation, choisissez votre domaine, puis accédez à l'onglet Packages.

10. S'il s'agit d'un dictionnaire personnalisé, notez l'ID lorsque le package devient disponible. À utiliser `analyzers/id` comme chemin de fichier dans les demandes adressées à OpenSearch.

Utilisation de packages avec OpenSearch

Cette section explique comment utiliser les deux types de packages : les dictionnaires personnalisés et les plugins préemballés.

Utilisation de dictionnaires personnalisés

Après avoir associé un fichier à un domaine, vous pouvez l'utiliser dans des paramètres tels que `synonyms_pathstopwords_path`, et `user_dictionary` lorsque vous créez des tokenizers et des filtres de token. Le paramètre exact varie selon l'objet. Plusieurs objets prennent en charge `synonyms_path` et `stopwords_path`, mais `user_dictionary` est exclusif au plugin kuromoji.

Pour le plugin IK (Chinese) Analysis, vous pouvez charger un fichier de dictionnaire personnalisé en tant que package personnalisé et l'associer à un domaine, et le plugin le récupère automatiquement sans avoir besoin d'un paramètre `user_dictionary`. Si votre fichier est un fichier de synonymes, utilisez le paramètre `synonyms_path`.

L'exemple suivant ajoute un fichier de synonymes à un nouvel index :

```
PUT my-index
{
  "settings": {
    "index": {
      "analysis": {
        "analyzer": {
          "my_analyzer": {
            "type": "custom",
            "tokenizer": "standard",
            "filter": ["my_filter"]
          }
        },
        "filter": {
          "my_filter": {
            "type": "synonym",
            "synonyms_path": "analyzers/F111111111",
            "updateable": true
          }
        }
```

```
      }
    }
  },
  "mappings": {
    "properties": {
      "description": {
        "type": "text",
        "analyzer": "standard",
        "search_analyzer": "my_analyzer"
      }
    }
  }
}
```

Cette demande crée un analyseur personnalisé pour l'index qui utilise le tokenizer standard et un filtre de token « synonym ».

- Les tokenizers séparent les flux de caractères en tokens (des mots en règle générale) en fonction d'un ensemble de règles. L'exemple le plus simple est le tokenizer whitespace, qui sépare les caractères précédents en token chaque fois qu'il rencontre un caractère espace. Autre exemple plus complexe, le tokenizer standard, qui utilise un ensemble de règles basées sur la grammaire pour parcourir plusieurs langues.
- Les filtres de token ajoutent, modifient ou suppriment des tokens. Par exemple, le filtre de token « synonym » ajoute des tokens lorsqu'il trouve un mot figurant dans la liste des synonymes. Le filtre de token « stop » supprime les tokens lorsqu'il trouve un mot dans la liste des mots vides.

Cette demande ajoute également un champ de texte (`description`) au mappage et indique OpenSearch d'utiliser le nouvel analyseur comme analyseur de recherche. Vous pouvez constater qu'elle utilise toujours l'analyseur standard comme analyseur d'index.

Enfin, notez la ligne `"updateable": true` dans le filtre de jeton. Ce champ s'applique uniquement aux analyseurs de recherche, et non aux analyseurs d'index, et il est essentiel si vous souhaitez par la suite mettre automatiquement à jour l'analyseur de recherche.

À des fins de test, ajoutez des documents à l'index :

```
POST _bulk
{ "index": { "_index": "my-index", "_id": "1" } }
{ "description": "ice cream" }
{ "index": { "_index": "my-index", "_id": "2" } }
```

```
{ "description": "croissant" }
{ "index": { "_index": "my-index", "_id": "3" } }
{ "description": "tennis shoe" }
{ "index": { "_index": "my-index", "_id": "4" } }
{ "description": "hightop" }
```

Recherchez-les ensuite en utilisant un synonyme :

```
GET my-index/_search
{
  "query": {
    "match": {
      "description": "gelato"
    }
  }
}
```

Dans ce cas, OpenSearch renvoie la réponse suivante :

```
{
  "hits": {
    "total": {
      "value": 1,
      "relation": "eq"
    },
    "max_score": 0.99463606,
    "hits": [{
      "_index": "my-index",
      "_type": "_doc",
      "_id": "1",
      "_score": 0.99463606,
      "_source": {
        "description": "ice cream"
      }
    }]
  }
}
```

> **Tip**
>
> Les fichiers de dictionnaire utilisent l'espace de tas Java proportionnel à leur taille. Par exemple, un fichier de dictionnaire de 2 Go peut consommer 2 Go d'espace de tas sur un

nœud. Si vous utilisez des fichiers volumineux, assurez-vous que vos nœuds disposent d'un espace suffisant pour les accueillir. Surveillez la métrique `JVMMemoryPressure` et mettez votre cluster à l'échelle si nécessaire.

Utilisation de plugins préemballés

OpenSearch Le service vous permet d'associer des OpenSearch plugins optionnels préinstallés à utiliser avec votre domaine. Un package de plug-in préconfiguré est compatible avec une OpenSearch version spécifique et ne peut être associé qu'à des domaines dotés de cette version. La liste des packages disponibles pour votre domaine inclut tous les plugins compatibles avec la version de votre domaine. Une fois que vous avez associé un plug-in à un domaine, le processus d'installation sur le domaine commence. Ensuite, vous pouvez référencer et utiliser le plugin lorsque vous faites des demandes au OpenSearch Service.

Associer et dissocier un plugin nécessite un déploiement bleu/vert. Pour de plus amples informations, veuillez consulter the section called “Changements qui entraînent généralement des blue/green déploiements”.

Les plugins optionnels incluent des analyseurs de langue et des résultats de recherche personnalisés. Par exemple, le plugin Amazon Personalize Search Ranking utilise l'apprentissage automatique pour personnaliser les résultats de recherche pour vos clients. Pour plus d'informations sur ce plugin, consultez la section Personnalisation des résultats de recherche à partir de OpenSearch. Pour obtenir la liste de tous les plug-ins pris en charge, consultezthe section called “Plugins par version du moteur”.

Plugin Sudachi

Pour le plugin Sudachi, lorsque vous réassociez un fichier de dictionnaire, cela ne se répercute pas immédiatement sur le domaine. Le dictionnaire est actualisé lorsque le prochain déploiement bleu/vert s'exécute sur le domaine dans le cadre d'une modification de configuration ou d'une autre mise à jour. Vous pouvez également créer un nouveau package avec les données mises à jour, créer un nouvel index à l'aide de ce nouveau package, réindexer l'index existant dans le nouvel index, puis supprimer l'ancien index. Si vous préférez utiliser l'approche de réindexation, utilisez un alias d'index afin de ne pas perturber votre trafic.

De plus, le plugin Sudachi ne prend en charge que les dictionnaires Sudachi binaires, que vous pouvez télécharger à l'aide de l'CreatePackageAPI. Pour plus d'informations sur le dictionnaire

système prédéfini et le processus de compilation des dictionnaires utilisateur, consultez la documentation de Sudachi.

L'exemple suivant montre comment utiliser les dictionnaires système et utilisateur avec le tokenizer Sudachi. Vous devez télécharger ces dictionnaires sous forme de packages personnalisés avec type `TXT-DICTIONARY` et fournir leur package IDs dans les paramètres supplémentaires.

```
PUT sudachi_sample
{
  "settings": {
    "index": {
      "analysis": {
        "tokenizer": {
          "sudachi_tokenizer": {
            "type": "sudachi_tokenizer",
            "additional_settings": "{\"systemDict\": \"<system-dictionary-package-id>\",\"userDict\": [\"<user-dictionary-package-id>\"]}"
          }
        },
        "analyzer": {
          "sudachi_analyzer": {
            "filter": ["my_searchfilter"],
            "tokenizer": "sudachi_tokenizer",
            "type": "custom"
          }
        },
        "filter":{
          "my_searchfilter": {
            "type": "sudachi_split",
            "mode": "search"
          }
        }
      }
    }
  }
}
```

Mise à jour de packages

Cette section explique uniquement comment mettre à jour un package de dictionnaire personnalisé, car les packages de plugins préemballés sont déjà mis à jour pour vous. Le chargement d'une nouvelle version d'un dictionnaire sur Amazon S3 ne met pas automatiquement à jour le package

sur Amazon OpenSearch Service. OpenSearch Le service stocke sa propre copie du fichier. Par conséquent, si vous chargez une nouvelle version sur S3, vous devez la mettre à jour manuellement.

Chacun de vos domaines associés stocke également sa propre copie du fichier. Pour que le comportement de recherche reste prévisible, les domaines continuent à utiliser leur version actuelle du package jusqu'à ce que vous le mettiez explicitement à jour. Pour mettre à jour un package personnalisé, modifiez le fichier dans Amazon S3 Control, mettez le package à jour dans OpenSearch Service, puis appliquez la mise à jour.

console

1. Dans la console OpenSearch de service, choisissez Packages.
2. Choisissez un package et sélectionnez Update (Mettre à jour).
3. Indiquez un nouveau chemin d'accès S3 au fichier, puis choisissez Mettre le package à jour.
4. Revenez à l'écran Packages.
5. Lorsque le statut du package passe à Available (Disponible), sélectionnez-le. Choisissez ensuite un ou plusieurs domaines associés, sélectionnez Apply update (Appliquer la mise à jour) et confirmez. Attendez que le statut de l'association passe à Actif.
6. Les étapes suivantes varient en fonction de la façon dont vous avez configuré vos index :

   - Si votre domaine exécute OpenSearch ou Elasticsearch 7.8 ou version ultérieure, et n'utilise que des analyseurs de recherche avec le champ actualisable défini sur true, aucune action n'est requise de votre part. OpenSearch Le service met automatiquement à jour vos index à l'aide de l'API _plugins/_refresh_search_analyzers.
   - Si votre domaine exécute d'Elasticsearch 7.7 ou une version antérieure, utilise des analyseurs d'index ou n'utilise pas le `updateable` champ, consultez. the section called “Mise à jour manuelle des index avec un nouveau dictionnaire”

La console est la méthode la plus simple pour mettre à jour les packages de OpenSearch service AWS CLI SDKs, mais vous pouvez également utiliser la ou l'API de configuration. Pour plus d'informations, consultez le AWS CLI Command Reference et le Amazon OpenSearch Service API Reference.

AWS SDK

Au lieu de mettre à jour manuellement un package dans la console, vous pouvez utiliser le SDKs pour automatiser le processus de mise à jour. L'exemple de script Python suivant charge un nouveau

fichier de package sur Amazon S3, met à jour le package dans OpenSearch Service, puis applique le nouveau package au domaine spécifié. Après avoir vérifié le succès de la mise à jour, il passe un appel type OpenSearch démontrant que les nouveaux synonymes ont été appliqués.

Vous devez fournir des valeurs pour `host`, `region`, `file_name`, `bucket_name`, `s3_key`, `package_id`, `domain_name` et `query`.

```
from requests_aws4auth import AWS4Auth
import boto3
import requests
import time
import json
import sys

host = ''  # The OpenSearch domain endpoint with https:// and a trailing slash. For
 example, https://my-test-domain.us-east-1.es.amazonaws.com/
region = ''  # For example, us-east-1
file_name = ''  # The path to the file to upload
bucket_name = ''  # The name of the S3 bucket to upload to
s3_key = ''  # The name of the S3 key (file name) to upload to
package_id = ''  # The unique identifier of the OpenSearch package to update
domain_name = ''  # The domain to associate the package with
query = ''  # A test query to confirm the package has been successfully updated

service = 'es'
credentials = boto3.Session().get_credentials()
client = boto3.client('opensearch')
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key,
                   region, service, session_token=credentials.token)


def upload_to_s3(file_name, bucket_name, s3_key):
    """Uploads file to S3"""
    s3 = boto3.client('s3')
    try:
        s3.upload_file(file_name, bucket_name, s3_key)
        print('Upload successful')
        return True
    except FileNotFoundError:
        sys.exit('File not found. Make sure you specified the correct file path.')


def update_package(package_id, bucket_name, s3_key):
```

```
    """Updates the package in OpenSearch Service"""
    print(package_id, bucket_name, s3_key)
    response = client.update_package(
        PackageID=package_id,
        PackageSource={
            'S3BucketName': bucket_name,
            'S3Key': s3_key
        }
    )
    print(response)


def associate_package(package_id, domain_name):
    """Associates the package to the domain"""
    response = client.associate_package(
        PackageID=package_id, DomainName=domain_name)
    print(response)
    print('Associating...')


def wait_for_update(domain_name, package_id):
    """Waits for the package to be updated"""
    response = client.list_packages_for_domain(DomainName=domain_name)
    package_details = response['DomainPackageDetailsList']
    for package in package_details:
        if package['PackageID'] == package_id:
            status = package['DomainPackageStatus']
            if status == 'ACTIVE':
                print('Association successful.')
                return
            elif status == 'ASSOCIATION_FAILED':
                sys.exit('Association failed. Please try again.')
            else:
                time.sleep(10)  # Wait 10 seconds before rechecking the status
                wait_for_update(domain_name, package_id)


def sample_search(query):
    """Makes a sample search call to OpenSearch"""
    path = '_search'
    params = {'q': query}
    url = host + path
    response = requests.get(url, params=params, auth=awsauth)
    print('Searching for ' + '"' + query + '"')
```

```
print(response.text)
```

> **Note**
>
> Si vous recevez l'erreur « package introuvable » lorsque vous exécutez le script à l'aide de la commande AWS CLI, cela signifie probablement que Boto3 utilise la région spécifiée dans ~/.aws/config, qui n'est pas la région dans laquelle se trouve votre compartiment S3. Vous pouvez exécuter `aws configure` et spécifier la région qui convient, ou bien ajouter explicitement la région dans le client :
>
> ```
> client = boto3.client('opensearch', region_name='us-east-1')
> ```

Mise à jour manuelle des index avec un nouveau dictionnaire

Les mises à jour manuelles de l'index ne s'appliquent qu'aux dictionnaires personnalisés, et non aux plug-ins préemballés. Pour utiliser un dictionnaire mis à jour, vous devez mettre à jour manuellement les index si vous remplissez l'une des conditions suivantes :

- Votre domaine exécute Elasticsearch 7.7. ou une version antérieure.
- Vous utilisez des packages personnalisés comme analyseurs d'index.
- Vous utilisez des packages personnalisés comme analyseurs de recherche, mais n'incluez pas le champ actualisable.

Pour mettre à jour les analyseurs avec les nouveaux fichiers de package, deux options s'offrent à vous :

- Fermez et ouvrez les index que vous souhaitez mettre à jour :

```
POST my-index/_close
POST my-index/_open
```

- Réindexez les index. Commencez par créer un index qui utilise le fichier de synonymes mis à jour (ou un fichier entièrement nouveau). Notez que seul l'UTF-8 est pris en charge.

```
PUT my-new-index
{
  "settings": {
```

```
    "index": {
      "analysis": {
        "analyzer": {
          "synonym_analyzer": {
            "type": "custom",
            "tokenizer": "standard",
            "filter": ["synonym_filter"]
          }
        },
        "filter": {
          "synonym_filter": {
            "type": "synonym",
            "synonyms_path": "analyzers/F222222222"
          }
        }
      }
    }
  },
  "mappings": {
    "properties": {
      "description": {
        "type": "text",
        "analyzer": "synonym_analyzer"
      }
    }
  }
}
```

Réindexez ensuite l'ancien index par rapport à ce nouvel index :

```
POST _reindex
{
  "source": {
    "index": "my-index"
  },
  "dest": {
    "index": "my-new-index"
  }
}
```

Si vous mettez fréquemment à jour des analyseurs d'index, utilisez des alias d'index pour que le chemin d'accès à l'index le plus récent reste cohérent :

```
POST _aliases
{
  "actions": [
    {
      "remove": {
        "index": "my-index",
        "alias": "latest-index"
      }
    },
    {
      "add": {
        "index": "my-new-index",
        "alias": "latest-index"
      }
    }
  ]
}
```

Si vous n'avez pas besoin de l'ancien index, supprimez-le :

```
DELETE my-index
```

Dissociation et suppression de packages

La dissociation d'un package, qu'il s'agisse d'un dictionnaire personnalisé ou d'un plugin préemballé, d'un domaine implique que vous ne pouvez plus utiliser ce package lorsque vous créez de nouveaux index. Une fois qu'un package est dissocié, les index existants qui l'utilisaient ne peuvent plus l'utiliser. Vous devez supprimer le package de n'importe quel index avant de pouvoir le dissocier, sinon la dissociation échoue.

Pour dissocier un package d'un domaine et le supprimer du OpenSearch Service, la console est la méthode la plus simple. La suppression d'un package du OpenSearch service ne le supprime pas de son emplacement d'origine sur Amazon S3.

Dissociation d'un package d'un domaine

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation, choisissez Domains (Domaines).

3. Choisissez le domaine, puis accédez à l'onglet Packages.
4. Sélectionnez un package, Actions, puis choisissez Dissociate (Dissocier). Confirmez votre choix.
5. Attendez que le package disparaisse de la liste. Vous devrez peut-être actualiser votre navigateur.
6. Si vous souhaitez utiliser le package avec d'autres domaines, arrêtez le processus ici. Pour continuer à supprimer le package (s'il s'agit d'un dictionnaire personnalisé), choisissez Packages dans le volet de navigation.
7. Sélectionnez le package et choisissez Delete (Supprimer).

Vous pouvez également utiliser l'API de configuration AWS CLI SDKs, ou l'API de configuration pour dissocier et supprimer des packages. Pour plus d'informations, consultez le AWS CLI Command Reference et le Amazon OpenSearch Service API Reference.

Gestion des plugins personnalisés dans Amazon OpenSearch Service

À l'aide de plugins personnalisés pour OpenSearch Service, vous pouvez étendre les OpenSearch fonctionnalités dans des domaines tels que l'analyse linguistique, le filtrage personnalisé, le classement, etc., ce qui vous permet de créer des expériences de recherche personnalisées. Des plugins personnalisés pour OpenSearch peuvent être développés en étendant la `org.opensearch.plugins.Plugin` classe puis en l'empaquetant dans un `.zip` fichier.

Les extensions de plug-in suivantes sont actuellement prises en charge par Amazon OpenSearch Service :

- AnalysisPlugin— Étend les fonctionnalités d'analyse en ajoutant, par exemple, des analyseurs personnalisés, des tokeniseurs de caractères ou des filtres pour le traitement de texte.
- SearchPlugin— Améliore les capacités de recherche grâce à des types de requêtes personnalisés, à des algorithmes de similarité, à des options de suggestion et à des agrégations.
- MapperPlugin— Vous permet de créer des types de champs personnalisés et leurs configurations de mappage OpenSearch, ce qui vous permet de définir comment les différents types de données doivent être stockés et indexés.
- ScriptPlugin— Vous permet d'ajouter des fonctionnalités de script personnalisées, par exemple OpenSearch, à des scripts personnalisés pour des opérations telles que la notation, le tri et les transformations des valeurs des champs lors de la recherche ou de l'indexation.

Vous pouvez utiliser la console OpenSearch de service ou les commandes d'API existantes pour les packages personnalisés afin de télécharger et d'associer le plug-in au cluster Amazon OpenSearch Service. Vous pouvez également utiliser la DescribePackagescommande pour décrire tous les packages de votre compte et pour afficher des détails tels que les détails de OpenSearch la version et des erreurs. OpenSearch Le service valide le package de plug-in pour vérifier la compatibilité des versions, les failles de sécurité et les opérations de plug-in autorisées. Pour de plus amples informations sur les packages personnalisés, veuillez consulterthe section called “Packages”.

OpenSearch version et Région AWS support

Les plugins personnalisés sont pris en charge sur les domaines de OpenSearch service qui exécutent OpenSearch la version 2.15 dans les versions suivantes : Régions AWS

- USA Est (Ohio) (us-east-2)
- USA Est (Virginie du Nord) (us-east-1)
- USA Ouest (Oregon) (us-west-2)
- Asie-Pacifique (Mumbai) (ap-south-1)
- Asie-Pacifique (Séoul) (ap-northeast-2)
- Asie-Pacifique (Singapour) (ap-southeast-1)
- Asie-Pacifique (Sydney) (ap-southeast-2)
- Asie-Pacifique (Tokyo) (ap-northeast-1)
- Canada (Centre) (ca-central-1)
- Europe (Francfort) (eu-central-1)
- Europe (Irlande) (eu-west-1)
- Europe (Londres) (eu-west-2)
- Europe (Paris) (eu-west-3)
- Amérique du Sud (São Paulo) (sa-east-1)

> **Note**
>
> Les plugins personnalisés contiennent du code développé par l'utilisateur. Tout problème, y compris les violations des SLA, causé par le code développé par l'utilisateur ne donne pas droit à des crédits SLA. Pour plus d'informations, consultez Amazon OpenSearch Service - Contrat de niveau de service (SLA).

Rubriques

- Quotas de plug-in
- Prérequis
- Résolution des problèmes
- Installation d'un plugin personnalisé avec
- Gestion des plugins personnalisés à l'aide du AWS CLI
- AWS KMS Intégration du package personnalisé Amazon OpenSearch Service

Quotas de plug-in

- Vous pouvez créer jusqu'à 25 plugins personnalisés par compte et par région.
- La taille maximale non compressée pour un plugin est de 1 Go.
- Le nombre maximal de plug-ins pouvant être associés à un seul domaine est de 20. Ce quota s'applique à tous les types de plugins combinés : optionnels, tiers et personnalisés.
- Les plugins personnalisés sont pris en charge sur les domaines exécutant OpenSearch la version 2.15 ou ultérieure.
- Le `descriptor.properties` fichier de votre plugin doit prendre en charge une version du moteur similaire à la version 2.15.0 ou à toute version 2.x.x, où la version du correctif est définie sur zéro.
- Les fonctions suivantes ne sont pas disponibles lorsque votre domaine utilise des plugins personnalisés :
  - Recherche inter-clusters
  - Réplication inter-clusters (CCR)
  - Réindexation à distance
  - Auto-Tune
  - the section called “Multi-AZ avec mode veille”
  - Interface OpenSearch utilisateur centralisée

Prérequis

Avant d'installer un plugin personnalisé et de l'associer à un domaine, vérifiez que les conditions suivantes sont respectées :

- La version du moteur prise en charge pour le plugin dans le `descriptor.properties` fichier doit être similaire à `2.15.0` ou`2.x.0`. C'est-à-dire que la version du correctif doit être nulle.
- Les fonctionnalités suivantes doivent être activées sur votre domaine :
  - Node-to-node chiffrement
  - Chiffrement au repos
  - `EnforceHTTPS`est défini sur « vrai »

    Voir également opensearch-https-requiredle guide du AWS Config développeur.
  - Les clients doivent soutenir Policy-min-TLS-1-2-PFS-2023-10. Vous pouvez spécifier cette prise en charge à l'aide de la commande suivante. Remplacez *`placeholder value`* avec vos propres informations :

    ```
    aws opensearch update-domain-config \
        --domain-name domain-name \
        --domain-endpoint-options '{"TLSSecurityPolicy":"Policy-Min-TLS-1-2-
    PFS-2023-10" }'
    ```

    Pour plus d'informations, consultez DomainEndpointOptionsle manuel Amazon OpenSearch Service API Reference.

Résolution des problèmes

Si le système renvoie le message d'erreur`PluginValidationFailureReason : The provided plugin could not be loaded`, consultez the section called “L'installation du plugin personnalisé échoue en raison de la compatibilité des versions” les informations de dépannage.

Installation d'un plugin personnalisé avec

Pour associer un plugin tiers à un domaine, importez d'abord la licence et la configuration du plugin sous forme de packages.

Pour installer un plugin personnalisé

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, choisissez Packages.
3. Choisissez Import package (Importer un package).

4. Dans Nom, entrez un nom unique et facilement identifiable pour le plugin.
5. (Facultatif) Pour la description, fournissez tous les détails utiles sur le package ou son objectif.
6. Pour le type de package, choisissez Plugin.
7. Dans le champ Source du package, entrez le chemin ou accédez au fichier ZIP du plugin dans Amazon S3.
8. Pour la version OpenSearch du moteur, choisissez la version prise OpenSearch en charge par le plugin.
9. Pour le chiffrement du package, choisissez si vous souhaitez personnaliser la clé de chiffrement du package. Par défaut, OpenSearch Service chiffre le package du plugin avec un Clé détenue par AWS. Vous pouvez utiliser une clé gérée par le client à la place.
10. Choisissez Importer.

Après avoir importé le package du plugin, associez-le à un domaine. Pour obtenir des instructions, veuillez consulter the section called “Importation et association d'un package à un domaine”.

Gestion des plugins personnalisés à l'aide du AWS CLI

Vous pouvez utiliser le AWS CLI pour gérer un certain nombre de tâches personnalisées du plugin.

Tâches

- Installation d'un plugin personnalisé avec AWS CLI
- Mise à jour d'un plugin personnalisé avec AWS CLI
- Création ou mise à jour d'un plugin personnalisé avec une AWS KMS clé de sécurité
- Mise à niveau d'un domaine de OpenSearch service avec des plugins personnalisés vers une version ultérieure de OpenSearch l'utilisation du AWS CLI
- Désinstaller et afficher l'état de dissociation d'un plugin personnalisé

Installation d'un plugin personnalisé avec AWS CLI

Avant de commencer

Pour pouvoir associer un plugin personnalisé à votre domaine, vous devez le charger dans un compartiment Amazon Simple Storage Service (Amazon S3). Le compartiment doit se situer à l' Région AWS endroit où vous souhaitez utiliser le plug-in. Pour plus d'informations sur l'ajout d'un objet dans un compartiment S3, consultez la section Chargement d'objets dans le guide de l'utilisateur d'Amazon Simple Storage Service.

Si votre plugin contient des informations sensibles, configurez le chiffrement côté serveur avec des clés gérées par S3 lorsque vous le chargez. Après avoir chargé le fichier, notez son chemin d'accès S3. Le format du chemin d'accès est `s3://amzn-s3-demo-bucket/file-path/file-name`.

> **Note**
>
> Vous pouvez éventuellement sécuriser un plugin personnalisé lorsque vous le créez en spécifiant une clé AWS Key Management Service (AWS KMS). Pour plus d'informations, consultez the section called “Création ou mise à jour d'un plugin personnalisé avec une AWS KMS clé de sécurité”.

Pour installer un plugin personnalisé avec AWS CLI

1. Créez un nouveau package pour votre plugin personnalisé en exécutant la commande create-package suivante, en vous assurant que les conditions suivantes sont remplies :

   - L'emplacement du compartiment et de la clé doit pointer vers le `.zip` fichier du plugin dans un compartiment S3 du compte dans lequel vous exécutez les commandes.
   - Le compartiment S3 doit se situer dans la même région que celle dans laquelle le package est créé.
   - Seuls `.zip` les fichiers sont pris en charge pour les `ZIP-PLUGIN` packages.
   - Le contenu du `.zip` fichier doit suivre la structure du répertoire prévue par le plugin.
   - La valeur de `--engine-version` doit être au format`OpenSearch_{MAJOR}.{MINOR}`. Par exemple : **`OpenSearch_2.17`**.

   Remplacez *`placeholder values`* avec vos propres informations :

   ```
   aws opensearch create-package \
       --package-name package-name \
       --region region \
       --package-type ZIP-PLUGIN \
       --package-source S3BucketName=amzn-s3-demo-bucket,S3Key=s3-key \
       --engine-version opensearch-version
   ```

2. (Facultatif) Consultez l'état de l'`create-package`opération, y compris les éventuels résultats de validation et de vulnérabilité en matière de sécurité, à l'aide de la commande describe-packages. Remplacez *`placeholder values`* avec vos propres informations :

```
aws opensearch describe-packages \
    --region region  \
    --filters '[{"Name": "PackageType","Value": ["ZIP-PLUGIN"]}, {"Name":
 "PackageName","Value": ["package-name"]}]'
```

La commande renvoie des informations semblables à ce qui suit :

```
{
    "PackageDetailsList": [{
        "PackageID": "pkg-identifier",
        "PackageName": "package-name",
        "PackageType": "ZIP-PLUGIN",
        "PackageStatus": "VALIDATION_FAILED",
        "CreatedAt": "2024-11-11T13:07:18.297000-08:00",
        "LastUpdatedAt": "2024-11-11T13:10:13.843000-08:00",
        "ErrorDetails": {
            "ErrorType": "",
            "ErrorMessage": "PluginValidationFailureReason : Dependency Scan
 reported 3 vulnerabilities for the plugin: CVE-2022-23307, CVE-2019-17571,
 CVE-2022-23305"
        },
        "EngineVersion": "OpenSearch_2.15",
        "AllowListedUserList": [],
        "PackageOwner": "OWNER-XXXX"
    }]
}
```

> **Note**
>
> Pendant l'`create-package`opération, Amazon OpenSearch Service vérifie la `ZIP-PLUGIN` compatibilité des versions, les extensions de plug-in prises en charge et les failles de sécurité. Les failles de sécurité sont analysées à l'aide du service Amazon Inspector. Les résultats de ces vérifications sont affichés dans `ErrorDetails` le champ de la réponse de l'API.

3. Utilisez la commande associate-package pour associer le plugin au domaine de OpenSearch service de votre choix en utilisant l'ID du package créé à l'étape précédente.

> **Tip**
>
> Si vous avez plusieurs plug-ins, vous pouvez utiliser la commande associate-packages pour associer plusieurs packages à un domaine en une seule opération.

Remplacez *placeholder values* avec vos propres informations :

```
aws opensearch associate-package \
    --domain-name domain-name \
    --region region \
    --package-id package-id
```

> **Note**
>
> Le plugin est installé et désinstallé à l'aide d'un processus de déploiement bleu/vert.

4. (Facultatif) Utilisez la list-packages-for-domaincommande pour afficher le statut de l'association. Le statut de l'association change au fur et à mesure que le flux de travail passe de `ASSOCIATING` à`ACTIVE`. Le statut de l'association passe à ACTIVE une fois que l'installation du plugin est terminée et que le plugin est prêt à être utilisé.

   Remplacez *placeholder values* par vos propres informations.

```
aws opensearch list-packages-for-domain \
    --region region \
    --domain-name domain-name
```

Mise à jour d'un plugin personnalisé avec AWS CLI

Utilisez la commande update-package pour apporter des modifications à un plugin.

> **Note**
>
> Vous pouvez éventuellement sécuriser un plugin personnalisé lorsque vous le mettez à jour en spécifiant une clé AWS Key Management Service (AWS KMS). Pour plus d'informations,

consultez the section called “Création ou mise à jour d'un plugin personnalisé avec une AWS KMS clé de sécurité”.

Pour mettre à jour un plugin personnalisé avec AWS CLI

- Exécutez la commande suivante. Remplacez *placeholder values* par vos propres informations.

```
aws opensearch update-package \
    --region region \
    --package-id package-id \
    --package-source S3BucketName=amzn-s3-demo-bucket,S3Key=s3-key \
    --package-description description
```

Après avoir mis à jour un package, vous pouvez utiliser la commande associate-package ou associate-packages pour appliquer les mises à jour de package à un domaine.

> **Note**
>
> Vous pouvez auditer, créer, mettre à jour, associer et dissocier des opérations sur le plugin à l'aide AWS CloudTrail de. Pour de plus amples informations, veuillez consulter Surveillance des appels OpenSearch d'API Amazon Service avec AWS CloudTrail.

Création ou mise à jour d'un plugin personnalisé avec une AWS KMS clé de sécurité

Vous pouvez sécuriser un plugin personnalisé lorsque vous le créez ou le mettez à jour en spécifiant une AWS KMS clé. Pour ce faire, configurez `PackageEncryptionOptions` `true` et spécifiez l'Amazon Resource Name (ARN) de la clé, comme illustré dans les exemples suivants.

Exemple : création d'un plugin personnalisé avec AWS KMS clé de sécurité

```
aws opensearch create-package \
    --region us-east-2  --package-name my-custom-package \
    --package-type ZIP-PLUGIN \
    --package-source S3BucketName=amzn-s3-demo-bucket,S3Key=my-s3-key
    --engine-version OpenSearch_2.15
"PackageConfigOptions": {
```

```
     ...
  }
  "PackageEncryptionOptions": {
    "Enabled": true,
    "KmsKeyId":"arn:aws:kms:us-east-2:111222333444:key/2ba228d5-1d09-456c-ash9-
daf42EXAMPLE"
  }
```

Exemple : mise à jour d'un plugin personnalisé avec AWS KMS clé de sécurité

```
aws opensearch update-package \
    --region us-east-2  --package-name my-custom-package \
    --package-type ZIP-PLUGIN \
    --package-source S3BucketName=amzn-s3-demo-bucket,S3Key=my-s3-key
    --engine-version OpenSearch_2.15
"PackageConfigOptions": {
     ...
  }
  "PackageEncryptionOptions": {
    "Enabled": true,
    "KmsKeyId":"arn:aws:kms:us-east-2:111222333444:key/2ba228d5-1d09-456c-ash9-
daf42EXAMPLE"
  }
```

> **⚠ Important**
>
> Si la AWS KMS clé que vous spécifiez est désactivée ou supprimée, le cluster associé peut être inopérationnel.

Pour plus d'informations sur AWS KMS l'intégration avec des packages personnalisés,the section called “AWS KMS Intégration du package personnalisé Amazon OpenSearch Service”.

Mise à niveau d'un domaine de OpenSearch service avec des plugins personnalisés vers une version ultérieure de OpenSearch l'utilisation du AWS CLI

Lorsque vous devez mettre à niveau un domaine de OpenSearch service qui utilise des plugins personnalisés vers une version ultérieure de OpenSearch, effectuez les processus suivants.

Pour mettre à niveau un domaine de OpenSearch service avec des plugins personnalisés vers une version ultérieure de OpenSearch l'utilisation du AWS CLI

1. Utilisez la commande create-package pour créer un nouveau package pour votre plugin en spécifiant la nouvelle OpenSearch version.

   Assurez-vous que le nom du package est le même pour le plugin pour toutes les versions du moteur. La modification du nom du package entraîne l'échec du processus de mise à niveau du domaine lors du déploiement bleu/vert.
2. Mettez à niveau votre domaine vers la version supérieure en suivant les étapes décrites dansthe section called “Mise à niveau de domaines”.

   Au cours de ce processus, Amazon OpenSearch Service dissocie la version précédente du package de plug-in et installe la nouvelle version à l'aide d'un déploiement bleu/vert.

Désinstaller et afficher l'état de dissociation d'un plugin personnalisé

Pour désinstaller le plugin de n'importe quel domaine, vous pouvez utiliser la commande dissociate-package. L'exécution de cette commande supprime également tous les packages de configuration ou de licence associés. Vous pouvez ensuite utiliser la list-packages-for-domaincommande pour afficher l'état de la dissociation.

> **Tip**
>
> Vous pouvez également utiliser la commande dissociate-packages pour désinstaller plusieurs plugins d'un domaine en une seule opération.

Pour désinstaller et afficher l'état de dissociation d'un plugin personnalisé

1. Désactivez le plugin dans chaque index. Cela doit être fait avant de dissocier le package du plug-in.

   Si vous essayez de désinstaller un plugin avant de le désactiver dans tous les index, le processus de déploiement bleu/vert reste bloqué. `Processing`
2. Exécutez la commande suivante pour désinstaller le plug-in. Remplacez *`placeholder values`* par vos propres informations.

```
aws opensearch dissociate-package \
```

```
    --region region \
    --package-id plugin-package-id \
    --domain-name domain name
```

3. (Facultatif) Exécutez la list-packages-for-domaincommande pour afficher l'état de la dissociation.

AWS KMS Intégration du package personnalisé Amazon OpenSearch Service

Les packages personnalisés Amazon OpenSearch Service fournissent un cryptage par défaut pour protéger vos `ZIP-PLUGIN` packages au repos en utilisant Clés gérées par AWS.

- Clés détenues par AWS— Les packages personnalisés Amazon OpenSearch Service utilisent ces clés par défaut pour chiffrer automatiquement vos `ZIP-PLUGIN` colis. Vous ne pouvez pas afficher, gérer ou utiliser les clés, Clés détenues par AWS ou vérifier leur utilisation. Toutefois, vous n'avez pas besoin de prendre de mesure ou de modifier les programmes pour protéger les clés qui chiffrent les données. Pour plus d'informations, consultez Clés détenues par AWS dans le Guide du développeur AWS Key Management Service .
- Clés gérées par le client : vous pouvez ajouter une deuxième couche de chiffrement à l'existant Clés détenues par AWS en choisissant une clé gérée par le client lors de la création de votre package `ZIP-PLUGIN` personnalisé.

  Les packages personnalisés Amazon OpenSearch Service prennent en charge l'utilisation d'une clé symétrique gérée par le client que vous créez, détenez et gérez afin d'ajouter une deuxième couche de chiffrement au chiffrement existant AWS détenu par. Étant donné que vous avez le contrôle total de cette couche de chiffrement, vous pouvez effectuer les tâches suivantes :

  - Établir et maintenir des politiques clés
  - Établir et maintenir des politiques et des subventions AWS Identity and Access Management (IAM)
  - Activer et désactiver des stratégies de clé
  - Rotation des matériaux de chiffrement de clé
  - Ajout de balises
  - Créez des alias de clé
  - Planifier les clés pour la suppression

Pour plus d'informations, consultez Clés gérées par le client dans le Guide du développeur AWS Key Management Service (langue française non garantie).

> **Note**
>
> Les packages personnalisés Amazon OpenSearch Service activent automatiquement le chiffrement Clés détenues par AWS au repos sans frais. Toutefois, AWS KMS facture des frais liés à l'utilisation d'une clé gérée par le client. Pour de plus amples informations sur la tarification, veuillez consulter AWS Key Management Service Tarification.

Comment le OpenSearch service de packages personnalisés Amazon Service utilise les subventions dans AWS KMS

OpenSearch Les packages personnalisés de service nécessitent un octroi pour utiliser votre clé gérée par le client.

Lorsque vous créez un `ZIP-PLUGIN` package chiffré à l'aide d'une clé gérée par le client, le OpenSearch service de packages personnalisés Amazon Service crée un octroi en votre nom en envoyant une CreateGrantdemande à AWS KMS. Les autorisations AWS KMS permettent au OpenSearch Service d'accéder à une AWS KMS clé de votre compte. Les autorisations créées par les packages personnalisés du OpenSearch Service sont soumises à une contrainte qui autorise les opérations uniquement lorsque la demande inclut un contexte de chiffrement avec votre ID de package personnalisé.

Les packages personnalisés Amazon OpenSearch Service nécessitent l'octroi d'utiliser votre clé gérée par le client pour les opérations internes suivantes :

Opération	Description
`DescribeKey`	Envoie `DescribeKey` des requêtes AWS KMS à pour vérifier que l'ID de clé symétrique gérée par le client saisi lors de la création du package de plug-in est valide.
`GenerateDataKeyWithoutPlaintext`	Envoie `GenerateDataKeyWithoutPlaintext` des requêtes AWS KMS à pour générer des clés de données chiffrées par la clé gérée par le client.
`GenerateDataKey`	Envoie des `GenerateDataKey` demandes AWS KMS à pour générer des clés de données afin de chiffrer le package lors de sa copie interne.

Opération	Description
`Decrypt`	Envoie `Decrypt` des requêtes AWS KMS à pour déchiffrer les clés de données chiffrées afin qu'elles puissent être utilisées pour déchiffrer vos données.

Vous pouvez révoquer l'accès à l'octroi ou supprimer l'accès du service à la clé gérée par le client à tout moment. Dans ce cas, le OpenSearch Service ne pourra accéder à aucune des données chiffrées par la clé gérée par le client, ce qui affectera les opérations qui dépendent de ces données. Par exemple, si vous tentez d'associer un package de plug-in auquel OpenSearch Service ne peut pas accéder, l'opération renvoie une `AccessDeniedException` erreur.

Création d’une clé gérée par le client

Vous pouvez créer une clé symétrique gérée par le client à l'aide de la AWS Management Console ou de la AWS KMS APIs.

Pour créer une clé symétrique gérée par le client

- Suivez les étapes décrites dans la section Création d'une clé KMS dans le guide du AWS Key Management Service développeur.

Stratégie de clé

Les stratégies de clé contrôlent l’accès à votre clé gérée par le client. Chaque clé gérée par le client doit avoir exactement une stratégie de clé, qui contient des instructions qui déterminent les personnes pouvant utiliser la clé et comment elles peuvent l’utiliser. Lorsque vous créez votre clé gérée par le client, vous pouvez spécifier une stratégie de clé. Pour plus d’informations, consultez Politiques de clé dans AWS KMS dans le Guide du développeur AWS Key Management Service .

Pour utiliser votre clé gérée par le client avec les ressources de votre plugin, vous devez autoriser les opérations d'API suivantes dans la stratégie de clé :

- `kms:CreateGrant` : ajoute une attribution à une clé gérée par le client. Accorde un accès de contrôle à une AWS KMS clé spécifiée, permettant d'autoriser les opérations requises par les packages personnalisés du OpenSearch Service. Pour de plus amples informations sur l'utilisation des subventions, veuillez consulter le Guide du AWS KMS développeur.

  Cela permet au OpenSearch Service d'effectuer les opérations suivantes :

  - Appelez `GenerateDataKeyWithoutPlainText` pour générer une clé de données chiffrée et la stocker pour d'autres validations.
  - Appelez `GenerateDataKey` pour copier le package du plugin en interne.
  - Appelez `Decrypt` pour accéder au package du plugin en interne.
  - Configurez un directeur partant à la retraite pour permettre au service de`RetireGrant`.
- `kms:DescribeKey`: fournit les détails des clés gérées par le client pour permettre au OpenSearch service de valider la clé.
- `kms:GenerateDataKey`,`kms:GenerateDataKeyWithoutPlaintext`, `kms:Decrypt` — Permet aux packages personnalisés du OpenSearch Service d'utiliser ces opérations dans le cadre de la subvention.

Voici des exemples de déclarations de stratégie que vous pouvez ajouter pour les packages personnalisés de OpenSearch service :

```
"Statement" : [
  {
    "Sid" : "Allow access to principals authorized to use OpenSearch Service custom
 packages",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "*"
    },
    "Action" : [
      "kms:CreateGrant",
      "kms:GenerateDataKey",
      "kms:GenerateDataKeyWithoutPlaintext",
      "kms:Decrypt"
    ],
    "Resource" : "*",
    "Condition" : {
      "StringEquals" : {
        "kms:ViaService" : "custom-packages.region.amazonaws.com"
      },
      "StringEquals" : {
        "kms:EncryptionContext:packageId": "Id of the package"
      }
    }
  },
```

```
  {
    "Sid" : "Allow access to principals authorized to use Amazon OpenSearch Service
 custom packages",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "*"
    },
    "Action" : [
      "kms:DescribeKey"
    ],
    "Resource" : "*",
    "Condition" : {
      "StringEquals" : {
        "kms:ViaService" : "custom-packages.region.amazonaws.com"
      }
    }
  }
]
```

Pour plus d'informations sur la spécification des autorisations dans une politique, consultez la section Politiques clés du Guide du AWS Key Management Service développeur. AWS KMS

Pour plus d'informations sur la résolution des problèmes d'accès par clé, consultez la section Résolution AWS KMS des problèmes liés aux autorisations dans le Guide du AWS Key Management Service développeur.

Spécification d'une clé gérée par le client pour les packages personnalisés Amazon OpenSearch Service

Vous pouvez spécifier une clé gérée par le client en tant que deuxième couche de chiffrement pour vos `ZIP-PLUGIN` packages.

Lorsque vous créez un package de plug-in, vous pouvez spécifier la clé de données en saisissant un ID de AWS KMS clé, que les packages personnalisés du OpenSearch service utilisent pour chiffrer le package de plug-in.

AWS KMS ID de clé : un identifiant de clé pour une clé gérée par le AWS KMS client. Saisissez un ID de clé, un ARN de clé, un nom d'alias ou un ARN d'alias.

Contexte de chiffrement des packages personnalisés Amazon OpenSearch Service

Un contexte de chiffrement est un ensemble facultatif de paires clé-valeur qui contient des informations contextuelles supplémentaires sur les données.

AWS KMS utilise le contexte de chiffrement en tant que données authentifiées supplémentaires pour prendre en charge le chiffrement authentifié. Lorsque vous incluez un contexte de chiffrement dans une demande de chiffrement de données, AWS KMS lie le contexte de chiffrement aux données chiffrées. Pour déchiffrer les données, vous devez inclure le même contexte de chiffrement dans la demande.

Contexte de chiffrement des packages personnalisés Amazon OpenSearch Service

Les packages personnalisés Amazon OpenSearch Service utilisent le même contexte de chiffrement dans toutes les opérations AWS KMS cryptographiques, où la clé `packageId` et la valeur sont le package `package-id` de votre plugin.

Utiliser le contexte de chiffrement pour la surveillance

Lorsque vous utilisez une clé symétrique gérée par le client pour chiffrer votre package de plug-in, vous pouvez utiliser le contexte de chiffrement dans les enregistrements et les journaux d'audit pour identifier la manière dont la clé gérée par le client est utilisée. Le contexte de chiffrement apparaît également dans les journaux générés par AWS CloudTrail ou Amazon CloudWatch Logs.

Utilisation du contexte de chiffrement pour contrôler l'accès à votre clé gérée par le client

Vous pouvez utiliser le contexte de chiffrement dans les stratégies de clé et les politiques IAM en tant que conditions pour contrôler l'accès à votre clé symétrique gérée par le client. Vous pouvez également utiliser des contraintes de contexte de chiffrement dans un octroi.

OpenSearch Les packages personnalisés de service utilisent une contrainte de contexte de chiffrement dans les octrois pour contrôler l'accès à la clé gérée par le client dans votre compte ou région. La contrainte d'octroi exige que les opérations autorisées par l'octroi utilisent le contexte de chiffrement spécifié.

Vous trouverez ci-dessous des exemples de déclarations de stratégie de clé permettant d'accorder l'accès à une clé gérée par le client dans un contexte de chiffrement spécifique. La condition énoncée dans cette déclaration de stratégie exige que les octrois comportent une contrainte de contexte de chiffrement qui spécifie le contexte de chiffrement.

```
{
    "Sid": "Enable DescribeKey",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
    },
    "Action": "kms:DescribeKey",
    "Resource": "*"
},
{
    "Sid": "Enable OpenSearch Service custom packages to use the key",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
    },
    "Action" : [
         "kms:CreateGrant",
        "kms:GenerateDataKey",
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals" : {
            "kms:EncryptionContext:packageId": "ID of the package"
         }
    }
}
```

Surveillance de vos clés de chiffrement pour un service de packages OpenSearch personnalisés

Lorsque vous utilisez une clé gérée par le AWS KMS client avec les ressources du OpenSearch service de packages personnalisés de votre service, vous pouvez utiliser CloudTrail or CloudWatch Logs pour suivre les demandes auxquelles les packages OpenSearch personnalisés sont envoyés AWS KMS.

En savoir plus

Les ressources suivantes fournissent plus d'informations sur le chiffrement des données au repos.

- Pour plus d'informations sur AWS KMS les concepts de base, consultez AWS KMS keysle guide du AWS Key Management Service développeur.

- Pour plus d'informations sur les meilleures pratiques de sécurité pour AWS KMS, consultez le guide AWS prescriptif sur les AWS Key Management Service meilleures pratiques.

Installation de plugins tiers dans Amazon OpenSearch Service

Amazon OpenSearch Service prend en charge les plug-ins tiers de partenaires sélectionnés. Ces plugins peuvent améliorer votre OpenSearch configuration grâce à des fonctionnalités supplémentaires telles que des analyseurs personnalisés, des tokeniseurs ou des fonctionnalités de chiffrement. Suivez les instructions d'installation et de configuration spécifiques fournies par les développeurs tiers pour garantir une intégration correcte avec votre domaine OpenSearch de service.

> **Note**
>
> Vous devez obtenir et conserver des licences valides directement auprès des développeurs tiers. Il se peut que certains fournisseurs n'activent pas du tout leurs plug-ins Régions AWS, alors vérifiez auprès du fournisseur de plug-in pour en connaître la disponibilité.

Les Plug-ins tiers suivants peuvent être utilisés avec OpenSearch Service :

- Plug-in de chiffrement Portal26 (Titanium-LockBox) — Utilise le cryptage certifié NIST FIPS 140-2 pour chiffrer les données au fur et à mesure de leur indexation. Il inclut le support BYOK (Bring Your Own Key), qui vous permet de gérer vos clés de chiffrement pour une sécurité renforcée. Le plugin est fourni par Portal26 et nécessite la OpenSearch version 2.15 ou supérieure.
- Name Match (RNI) : fait correspondre les noms, les organisations, les adresses et les dates dans plus de 24 langues, ce qui améliore la sécurité et la conformité. Le plugin est fourni par Babel Street et nécessite la OpenSearch version 2.15 ou supérieure.

Rubriques

- Prérequis
- Installation de plug-in tiers
- Étapes suivantes

Prérequis

Avant d'installer un plug-in tiers, effectuez les opérations suivantes :

- J'ai obtenu les fichiers de configuration et de licence du plugin et chargez-les dans un compartiment Amazon S3. Le compartiment doit se situer dans le même Région AWS que le domaine.
- Un plugin tiers est un type de plugin personnalisé. Assurez-vous que le domaine répond aux conditions requises pour les plugins personnalisés.

Installation de plug-in tiers

Pour associer un plug-in tiers à un domaine de OpenSearch service, vous devez d'abord télécharger trois packages distincts : le package de licence, le package de configuration et le package de plug-in.

- Le package de licence inclut les informations de licence ou les métadonnées associées au plugin, au format .json ou .xml.
- Le package de configuration contient les fichiers de configuration du plugin ainsi que les ressources et paramètres associés. Ces fichiers définissent la manière dont le plugin se comporte ou s'intègre à OpenSearch celui-ci.
- Le package du plugin contient le binaire du plugin compilé, qui est le code exécutable qui OpenSearch s'exécute. C'est le cœur de la fonctionnalité du plugin.

Après avoir chargé les deux packages, vous pouvez associer le plugin et la licence à un domaine compatible.

console

Pour associer un plugin tiers à un domaine, importez d'abord la licence et la configuration du plugin sous forme de packages.

Pour installer un plugin tiers

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, choisissez Packages.
3. Importez d'abord le package de licence. Choisissez Import package (Importer un package).
4. Dans le champ Type de package, sélectionnez Licence.
5. Dans le champ Source du package, entrez le chemin d'accès au fichier JSON ou XML de licence dans Amazon S3.

6. Choisissez Importer. Le package apparaît dans l'onglet Licences de la page Packages.
7. Importez maintenant la configuration du plugin. Choisissez à nouveau Importer le package.
8. Pour Type de package, choisissez Configuration.
9. Dans le champ Source du package, entrez le chemin d'accès au fichier ZIP de configuration du plugin dans Amazon S3.
10. Choisissez Importer.
11. Enfin, importez le plugin lui-même. Choisissez Import package (Importer un package).
12. Pour le type de package, choisissez Plugin.
13. Dans le champ Source du package, entrez le chemin d'accès au fichier ZIP du plugin dans Amazon S3.
14. Sélectionnez la version OpenSearch du moteur prise en charge par le plugin.
15. Choisissez Importer.

Pour associer un plugin tiers à un domaine

1. Associez maintenant la licence et la configuration du plugin au domaine. Dans le volet de navigation de gauche, choisissez Domains (Domaines).
2. Choisissez le nom du domaine pour ouvrir sa configuration de cluster.
3. Accédez à l'onglet Plugins.
4. Choisissez Associer des packages, puis sélectionnez les packages de plug-in, de licence et de configuration que vous venez d'importer.
5. Choisissez Select (Sélectionner).
6. Choisissez Suivant. Passez en revue les packages à associer et choisissez Associer.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Tout d'abord, utilisez la commande create-package pour créer un nouveau package contenant la licence du plugin. Ils `S3Key` doivent pointer vers un fichier .json ou .xml dans Amazon S3 qui inclut le texte de licence ou les métadonnées.

```
aws opensearch create-package \
  --package-name plugin-license-package \
  --package-type PACKAGE-LICENSE \
  --package-source S3BucketName=my-bucket,S3Key=licenses/my-plugin-license.json
```

Utilisez à nouveau la commande create-package pour créer un package contenant la configuration du plugin. S3KeyIl doit pointer vers un fichier .zip dans Amazon S3 qui respecte la structure de répertoire attendue par le plugin.

```
aws opensearch create-package \
  --package-name plugin-config-package \
  --package-type PACKAGE-CONFIG \
  --package-source S3BucketName=my-bucket,S3Key=path/to/package.zip
```

Utilisez à nouveau la commande create-package pour créer un package contenant le plugin lui-même. Le S3Key doit pointer vers le fichier .zip du plug-in Amazon S3.

```
aws opensearch create-package \
  --package-name plugin-package \
  --package-type ZIP-PLUGIN \
  --package-source S3BucketName=my-bucket,S3Key=path/to/package.zip
```

Enfin, utilisez la commande associate-package pour lier le plugin, la licence et la configuration du partenaire à un domaine compatible en spécifiant le package IDs pour chacun. Spécifiez l'ID du plugin comme condition préalable pour les autres packages, ce qui signifie qu'il doit être associé au domaine avant les autres packages.

```
aws opensearch associate-packages \
  --domain-name my-domain \
  --package-list '[{"PackageID": "plugin-package-id"},{"PackageID": "license-package-id","PrerequisitePackageIDList":["plugin-package-id"]},{"PackageID":"config-package-id","PrerequisitePackageIDList":["plugin-package-id"]}]'
```

Étapes suivantes

Lorsque l'association est terminée, vous pouvez activer le plugin sur des index spécifiques ou le configurer selon vos besoins en fonction de vos besoins. Pour appliquer des fonctionnalités de plug-in tiers à des index spécifiques, modifiez les paramètres d'index lors de la création de l'index ou mettez à jour les index existants. Par exemple, si votre plug-in tiers inclut un analyseur personnalisé, référencez-le dans les paramètres d'index.

Pour appliquer les fonctionnalités du plugin de manière cohérente sur plusieurs index, utilisez des modèles d'index qui incluent les configurations du plugin. Consultez toujours la documentation du plugin pour comprendre comment configurer ses fonctionnalités pour votre OpenSearch installation.

Interrogation de vos données Amazon OpenSearch Service avec SQL

Vous pouvez utiliser SQL pour interroger Amazon OpenSearch Service, plutôt que d'utiliser le DSL de OpenSearch requête basé sur JSON. Interroger avec SQL est utile si vous êtes déjà familiarisé avec le langage ou que vous souhaitez intégrer votre domaine avec une application qui l'utilise. La prise en charge de SQL est disponible sur les domaines exécutant OpenSearch Elasticsearch 6.5 ou version ultérieure.

> **Note**
>
> Cette documentation décrit la compatibilité des versions entre OpenSearch Service et les différentes versions du plugin SQL, ainsi que les pilotes JDBC et ODBC. Consultez la OpenSearchdocumentation open source pour plus d'informations sur la syntaxe des requêtes de base et complexes, des fonctions, des requêtes de métadonnées et des fonctions d'agrégation.

Utilisez le tableau suivant pour trouver la version du plug-in SQL prise en charge par chaque OpenSearch version Elasticsearch.

OpenSearch

OpenSearch version	Version du plug-in SQL	Fonctionnalités notables
2.19.0	2.19.0.0	
2.18.0	2.18.0.0	
2.17.0	2.17.0.0	
2.15.0	2.15.0.0	
2.13.0	2.13.0.0	
2.11.0	2.11.0.0	Ajout de la prise en charge du langage PPL et des requêtes

OpenSearch version	Version du plug-in SQL	Fonctionnalités notables
2.9.0	2.9.0.0	Ajout du connecteur Spark et des fonctions de table de support et ProMQL
2.7.0	2.7.0.0	Ajouter une `datasource` API
2.5.0	2.5.0.0	
2.3.0	2.3.0.0	Ajouter des fonctions datetime `maketime` et `makedate`
1.3.0	1.3.0.0	Prise en charge de la taille limite par défaut des requêtes et de la clause IN pour effectuer une sélection dans une liste de valeurs
1.2.0	1.2.0.0	Ajout d'un nouveau protocole pour le format de réponse de visualisation
1.1.0	1.1.0.0	Prise en charge de la fonction de correspondance comme filtre dans SQL et PPL
1.0.0	1.0.0.0	Prise en charge de l'interrogation d'un flux de données

Open Distro for Elasticsearch

Version Elasticsearch.	Version du plug-in SQL	Fonctionnalités notables
7,10	1,13,0	NULL FIRST et LAST pour les fonctions de fenêtrage, fonction CAST (), commandes SHOW et DESCRIBE
7.9	1,11,0	Ajouter des fonctions de date/heure supplémentaires, le mot-clé ORDER BY
7.8	1,9.0	
7.7	1.8.0	

Version Elasticsearch.	Version du plug-in SQL	Fonctionnalités notables
7.3	1.3.0	Opérateurs de chaîne et de nombre multiples
7.1	1.1.0	

Exemple d'appel

Pour interroger vos données avec SQL, envoyez des requêtes HTTP à la fonction `_sql` en utilisant le format suivant :

```
POST domain-endpoint/_plugins/_sql
{
  "query": "SELECT * FROM my-index LIMIT 50"
}
```

> **Note**
>
> Si votre domaine exécute Elasticsearch plutôt que d'Elasticsearch OpenSearch, le format est. `_opendistro/_sql`

Remarques et différences

Les appels à la fonction `_plugins/_sql` incluent les noms d'index dans le corps de la requête et prennent donc en compte les mêmes considérations en matière de stratégie d'accès que les opérations bulk, mget et msearch. Comme toujours, suivez le principe du moindre privilège lorsque vous accordez des autorisations aux opérations d'API.

Pour de plus amples informations sur la sécurité concernant l'utilisation de SQL avec le contrôle précis des accès, consultez the section called “Contrôle précis des accès”.

Le plugin OpenSearch SQL inclut de nombreux paramètres réglables. Dans OpenSearch Service, utilisez le `_cluster/settings` chemin, et non le chemin des paramètres du plugin (`_plugins/_query/settings`) :

```
PUT _cluster/settings
{
```

```
  "transient" : {
    "plugins.sql.enabled" : true
  }
}
```

Pour les domaines Elasticsearch hérités, remplacez `plugins` avec `opendistro` :

```
PUT _cluster/settings
{
  "transient" : {
    "opendistro.sql.enabled" : true
  }
}
```

SQL Workbench

SQL Workbench est une interface utilisateur de OpenSearch tableaux de bord qui vous permet d'exécuter des requêtes SQL à la demande, de traduire SQL dans son équivalent REST et d'afficher et d'enregistrer les résultats aux formats texte, JSON, JDBC ou CSV. Pour plus d'informations, consultez Workbench de requête.

CLI SQL

L'interface CLI SQL est une application Python autonome que vous pouvez lancer avec la commande `opensearchsql`. Pour connaître les étapes d'installation, de configuration et d'utilisation, consultez CLI SQL.

Pilote JDBC

Le pilote Java Database Connectivity (JDBC) vous permet d'intégrer des domaines de OpenSearch service à vos applications de Business Intelligence (BI) favorites. Pour télécharger le pilote, cliquez ici. Pour plus d'informations, consultez le GitHubRéférentiel.

Pilote ODBC

Le pilote ODBC (Open Database Connectivity) est un pilote en lecture seule pour Windows et macOS qui vous permet de connecter des applications de business intelligence telles que Microsoft Excel au plug-in SQL.

Vous pouvez télécharger un exemple de fichier de pilote fonctionnel sur la page OpenSearch des artefacts. Pour plus d'informations sur l'installation du pilote, consultez le Référentiel SQL sur GitHub.

Recherche des k plus proches voisins (k-NN) dans Amazon Service OpenSearch

Abréviation de l'algorithme des k plus proches voisins (en anglais, k-nearest neighbors) associé, k-NN pour Amazon OpenSearch Service vous permet de rechercher des points dans un espace vectoriel et de trouver les « plus proches voisins » de ces points par distance euclidienne ou similarité cosinus. Les cas d'utilisation incluent des recommandations (par exemple, une fonctionnalité « autres chansons que vous pourriez aimer » dans une application musicale), la reconnaissance d'images et la détection des fraudes.

> **Note**
>
> Cette documentation fournit un bref aperçu du plug-in K-nn, ainsi que des limites liées à l'utilisation du plug-in avec un OpenSearch service géré. Pour une documentation complète du plug-in K-nn, y compris des exemples simples et complexes, des références de paramètres et la référence complète de l'API, consultez la OpenSearch documentation open source. La documentation open source couvre également le réglage des performances et les paramètres k-NN-specific du cluster.

Prise en main de k-NN

Pour utiliser k-NN, vous devez créer un index avec le paramètre `index.knn` et ajouter un ou plusieurs champs du type de données `knn_vector`.

```
PUT my-index

{
  "settings": {
    "index.knn": true
  },
  "mappings": {
    "properties": {
      "my_vector1": {
        "type": "knn_vector",
        "dimension": 2
      },
      "my_vector2": {
        "type": "knn_vector",
```

```
        "dimension": 4
      }
    }
  }
}
```

Le type de données `knn_vector` prend en charge une seule liste de 10 000 nombres à virgule flottante maximum, dont le nombre est défini par le paramètre `dimension` requis. Après avoir créé l'index, ajoutez-y des données.

```
POST _bulk

{ "index": { "_index": "my-index", "_id": "1" } }
{ "my_vector1": [1.5, 2.5], "price": 12.2 }
{ "index": { "_index": "my-index", "_id": "2" } }
{ "my_vector1": [2.5, 3.5], "price": 7.1 }
{ "index": { "_index": "my-index", "_id": "3" } }
{ "my_vector1": [3.5, 4.5], "price": 12.9 }
{ "index": { "_index": "my-index", "_id": "4" } }
{ "my_vector1": [5.5, 6.5], "price": 1.2 }
{ "index": { "_index": "my-index", "_id": "5" } }
{ "my_vector1": [4.5, 5.5], "price": 3.7 }
{ "index": { "_index": "my-index", "_id": "6" } }
{ "my_vector2": [1.5, 5.5, 4.5, 6.4], "price": 10.3 }
{ "index": { "_index": "my-index", "_id": "7" } }
{ "my_vector2": [2.5, 3.5, 5.6, 6.7], "price": 5.5 }
{ "index": { "_index": "my-index", "_id": "8" } }
{ "my_vector2": [4.5, 5.5, 6.7, 3.7], "price": 4.4 }
{ "index": { "_index": "my-index", "_id": "9" } }
{ "my_vector2": [1.5, 5.5, 4.5, 6.4], "price": 8.9 }
```

Vous pouvez alors rechercher les données en utilisant le type de requête `knn`.

```
GET my-index/_search
{
  "size": 2,
  "query": {
    "knn": {
      "my_vector2": {
        "vector": [2, 3, 5, 6],
        "k": 2
      }
```

```
    }
  }
}
```

Dans ce cas, k est le nombre de voisins que vous voulez que la requête renvoie, mais vous devez également inclure l'option `size`. Sinon, vous obtenez des résultats k pour chaque partition (et chaque segment) plutôt que des résultats k pour l'ensemble de la requête. k-NN prend en charge une valeur k maximale de 10 000.

Si vous combinez la requête `knn` avec d'autres clauses, vous pouvez recevoir moins de résultats que k. Dans cet exemple, la clause `post_filter` réduit le nombre de résultats de 2 à 1.

```
GET my-index/_search

{
  "size": 2,
  "query": {
    "knn": {
      "my_vector2": {
        "vector": [2, 3, 5, 6],
        "k": 2
      }
    }
  },
  "post_filter": {
    "range": {
      "price": {
        "gte": 6,
        "lte": 10
      }
    }
  }
}
```

Si vous devez gérer un volume important de requêtes tout en maintenant des performances optimales, vous pouvez utiliser l'_msearchAPI pour créer une recherche groupée avec JSON et envoyer une seule requête pour effectuer plusieurs recherches :

```
GET _msearch

{ "index": "my-index"}
{ "query": { "knn": {"my_vector2":{"vector": [2, 3, 5, 6],"k":2 }} } }
```

```
{ "index": "my-index", "search_type": "dfs_query_then_fetch"}
{ "query": { "knn": {"my_vector1":{"vector": [2, 3],"k":2 }} } }
```

La vidéo suivante montre comment configurer des recherches vectorielles groupées pour les requêtes k-NN.

Différences, réglage et limitations de k-NN

OpenSearch vous permet de modifier tous les paramètres K-nn à l'aide de l'`_cluster/settings`API. Sur OpenSearch Service, vous pouvez modifier tous les paramètres à l'exception de `knn.memory.circuit_breaker.enabled` et`knn.circuit_breaker.triggered`. Les statistiques k-NN sont incluses sous forme de CloudWatch métriques Amazon.

Comparez notamment la `KNNGraphMemoryUsage` métrique de chaque nœud de données avec la `knn.memory.circuit_breaker.limit` statistique et la RAM disponible pour le type d'instance. OpenSearch Le service utilise la moitié de la mémoire RAM d'une instance pour le tas Java (jusqu'à une taille de tas de 32 GiB). Par défaut, k-NN utilise jusqu'à 50 % de la moitié restante, de sorte qu'un type d'instance doté de 32 Gio de RAM peut accueillir 8 Gio de graphiques (32 * 0,5 * 0,5). Les performances peuvent baisser si l'utilisation de la mémoire graphique est supérieure à cette valeur.

Vous pouvez migrer un index K-nn créé sur la version 2.x ou ultérieure vers un domaine doté de la version 2.17 UltraWarmou ultérieure ou un stockage à froid sur un domaine.

L'API de suppression du cache et les API d'échauffement pour les indices K-nn sont bloquées pour les indices chauds. Lorsque la première requête est lancée pour l'index, elle télécharge les fichiers graphiques depuis Amazon S3 et charge le graphe en mémoire. De même, lorsque le TTL est expiré pour les graphes, les fichiers sont automatiquement expulsés de la mémoire.

Recherche entre clusters dans Amazon Service OpenSearch

La recherche entre clusters dans Amazon OpenSearch Service vous permet d'effectuer des requêtes et des agrégations sur plusieurs domaines connectés. Il est souvent plus judicieux d'utiliser plusieurs petits domaines plutôt qu'un seul grand domaine, en particulier lorsque vous exécutez plusieurs types de charges de travail.

Les domaines spécifiques à la charge de travail vous permettent d'effectuer les tâches suivantes :

- Optimiser chaque domaine en choisissant des types d'instance pour des charges de travail spécifiques.

- Établir des limites d'isolement des pannes entre les charges de travail. Cela signifie que si l'une de vos charges de travail échoue, l'erreur est contenue dans ce domaine spécifique et n'a aucun impact sur vos autres charges de travail.
- Passez plus facilement d'un domaine à l'autre.

La recherche entre clusters prend en charge les OpenSearch tableaux de bord, ce qui vous permet de créer des visualisations et des tableaux de bord pour tous vos domaines. Vous payez les frais de transfert de AWS données standard pour les résultats de recherche transférés entre domaines.

> **Note**
>
> L'open source propose OpenSearch également une documentation pour la recherche entre clusters. La configuration est très différente pour les clusters open source par rapport aux domaines Amazon OpenSearch Service gérés. Plus particulièrement, dans OpenSearch Service, vous configurez les connexions entre clusters à l'aide de cURL AWS Management Console plutôt que via cURL. En outre, le service géré utilise AWS Identity and Access Management (IAM) pour l'authentification entre clusters en plus d'un contrôle d'accès précis. Par conséquent, nous vous recommandons d'utiliser cette documentation, plutôt que la OpenSearch documentation open source, pour configurer la recherche entre clusters pour vos domaines.

Rubriques

- Limites
- Conditions préalables à la recherche inter-clusters
- Tarification de la recherche inter-clusters
- Configuration d'une connexion
- Suppression d'une connexion
- Configuration de la procédure de sécurité et d'exemples
- OpenSearch Tableaux de bord

Limites

La recherche inter-clusters comporte plusieurs limitations importantes :

- Vous ne pouvez pas connecter un domaine Elasticsearch à un OpenSearch domaine.
- Vous ne pouvez pas vous connecter à des OpenSearch/Elasticsearch clusters autogérés.
- Pour connecter des domaines entre différentes régions, les deux domaines doivent être sur Elasticsearch 7.10 ou version ultérieure ou. OpenSearch
- Un domaine peut avoir un maximum de 20 connexions sortantes. De même, un domaine peut avoir un maximum de 20 connexions entrantes. En d'autres termes, un domaine peut se connecter à un maximum de 20 autres domaines.
- Le domaine source doit se trouver sur une version identique ou supérieure à celle du domaine de destination. Si vous configurez une connexion bidirectionnelle entre deux domaines et que vous souhaitez mettre à niveau l'un d'entre eux ou les deux, vous devez d'abord supprimer l'une des connexions.
- Vous ne pouvez pas utiliser de dictionnaires personnalisés ou SQL avec la recherche inter-clusters.
- Vous ne pouvez pas l'utiliser AWS CloudFormation pour connecter des domaines.
- Vous ne pouvez pas utiliser la recherche inter-clusters sur des instances M3 ou les instances extensibles (T2 et T3).

Conditions préalables à la recherche inter-clusters

Avant de configurer la recherche inter-clusters, assurez-vous que vos domaines répondent aux exigences suivantes :

- Deux OpenSearch domaines, ou domaines Elasticsearch sur la version 6.7 ou ultérieure
- Contrôle précis des accès activé
- Node-to-node chiffrement activé

Tarification de la recherche inter-clusters

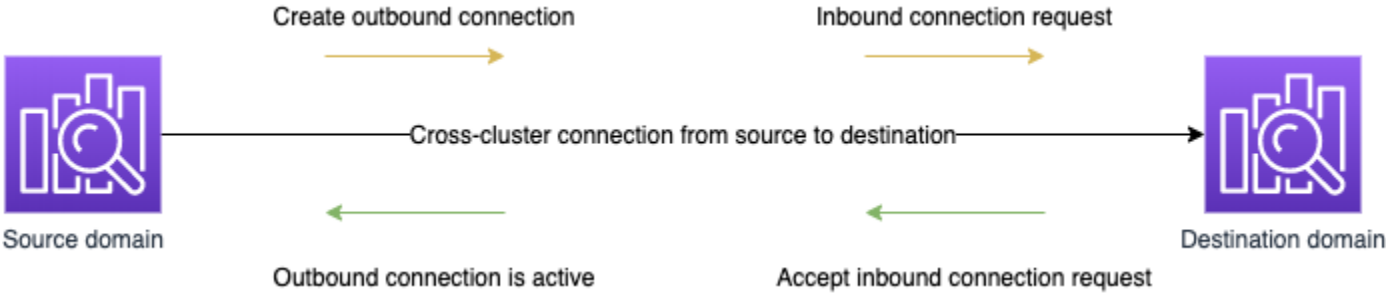
Pas de frais supplémentaire pour les recherches entre domaines.

Configuration d'une connexion

Le domaine « source » fait référence au domaine duquel provient une demande de recherche inter-clusters. En d'autres termes, le domaine source est celui auquel vous envoyez la demande de recherche initiale.

Le domaine « destination » est le domaine interrogé par le domaine source.

Une connexion inter-clusters est unidirectionnelle depuis la source vers le domaine de destination. Cela signifie que le domaine de destination ne peut pas interroger le domaine source. Cependant, vous pouvez configurer une autre connexion dans la direction opposée.



Le domaine source crée une connexion « sortante » vers le domaine de destination. Le domaine de destination reçoit une demande de connexion « entrante » depuis le domaine source.

Configurer une connexion

1. Dans le tableau de bord de votre domaine, choisissez un domaine, puis accédez à l'onglet Connections (Connexions).
2. Dans la section Outbound Connections (Connexions sortantes), choisissez Request (Demande).
3. Dans Alias de la connexion, saisissez un nom pour votre connexion.
4. Choisissez entre vous connecter à un domaine dans votre Compte AWS région ou dans un autre compte ou région.

   - Pour vous connecter à un cluster dans votre Compte AWS région, sélectionnez le domaine dans le menu déroulant et choisissez Request.
   - Pour vous connecter à un cluster dans une autre région Compte AWS ou dans une autre région, sélectionnez l'ARN du domaine distant et choisissez Request. Pour connecter des domaines entre différentes régions, les deux domaines doivent exécuter Elasticsearch version 7.10 ou ultérieure ou. OpenSearch
5. Pour ignorer les clusters non disponibles pour les requêtes de cluster, sélectionnez Ignorer les clusters non disponibles. Ce paramètre garantit que vos requêtes inter-clusters renvoient des résultats partiels malgré les échecs sur un ou plusieurs clusters distants.
6. La recherche inter-clusters valide d'abord la demande de connexion pour s'assurer que les conditions préalables sont remplies. Si les domaines s'avèrent incompatibles, la demande de connexion passera à l'état `Validation failed`.

7. Une fois la demande de connexion validée avec succès, elle est envoyée au domaine de destination, où elle doit être approuvée. Tant que cette approbation n'a pas eu lieu, la connexion reste dans l'état `Pending acceptance`. Lorsque la demande de connexion sera acceptée au niveau du domaine de destination, l'état passera à `Active` et le domaine de destination deviendra disponible pour les demandes.

   - La page du domaine affiche l'état global du domaine et les détails de l'état de l'instance de votre domaine de destination. Seuls les propriétaires de domaines ont la possibilité de créer, de visualiser, de supprimer et de surveiller les connexions vers ou depuis leurs domaines.

Une fois la connexion établie, tout le trafic qui circule entre les nœuds des domaines connectés est chiffré. Si vous connectez un domaine VPC à un domaine non-VPC et que le domaine non-VPC est un point de terminaison public pouvant recevoir du trafic depuis Internet, le trafic inter-clusters entre les domaines est toujours chiffré et sécurisé.

Suppression d'une connexion

La suppression d'une connexion arrête toute opération intercluster sur ses index.

1. Dans le tableau de bord de votre domaine, accédez à l'onglet Connexions.
2. Sélectionnez les connexions de domaine que vous souhaitez supprimer et choisissez Delete (Supprimer), puis confirmez la suppression.

Vous pouvez effectuer ces étapes sur le domaine source ou de destination pour supprimer la connexion. Une fois la connexion supprimée, elle restera visible en état `Deleted` pendant 15 jours.

Vous ne pouvez pas supprimer un domaine avec des connexions inter-clusters actives. Pour supprimer un domaine, commencez par supprimer toutes ses connexions entrantes et sortantes. Vous vous assurez ainsi de tenir compte des utilisateurs de domaines de clusters croisés avant de supprimer le domaine.

Configuration de la procédure de sécurité et d'exemples

1. Vous envoyez une demande de recherche inter-clusters vers le domaine source.
2. Le domaine source évalue cette demande en fonction de sa stratégie d'accès au domaine. Étant donné que la recherche inter-clusters nécessite un contrôle précis des accès, nous recommandons une stratégie d'accès ouvert sur le domaine source.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "*"
        ]
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:region:account:domain/src-domain/*"
    }
  ]
}
```

> **Note**
>
> Si vous incluez des index distants dans le chemin, vous devez encoder l'URI en URL dans l'ARN du domaine. Par exemple, utilisez `arn:aws:es:us-east-1:123456789012:domain/my-domain/local_index,dst%3Aremote_index` plutôt que `arn:aws:es:us-east-1:123456789012:domain/my-domain/local_index,dst:remote_index`.

Si vous choisissez d'utiliser une stratégie d'accès restrictive en plus d'un contrôle précis des accès, votre politique doit autoriser l'accès `es:ESHttpGet` au minimum.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
```

```
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:user/test-user"
        ]
      },
      "Action": "es:ESHttpGet",
      "Resource": "arn:aws:es:region:account:domain/src-domain/*"
    }
  ]
}
```

3. Le Contrôle précis des accès sur le domaine source évalue la demande :

   - La demande est-elle signée avec des informations d'identification de base IAM ou HTTP valides ?
   - Si c'est le cas, l'utilisateur a-t-il l'autorisation d'effectuer la recherche et d'accéder aux données ?

   Si la demande recherche uniquement des données sur le domaine de destination (par exemple, `dest-alias:dest-index/_search`), vous avez uniquement besoin d'autorisations sur le domaine de destination.

   Si la demande recherche des données sur les deux domaines (par exemple, `source-index,dest-alias:dest-index/_search`), vous avez besoin d'autorisations sur les deux domaines.

   Dans le cadre d'un contrôle d'accès précis, les utilisateurs doivent disposer de l'`indices:admin/shards/search_shards`autorisation en plus des autorisations standard `read` ou `search` des autorisations pour les index concernés.

4. Le domaine source transmet la demande au domaine de destination. Le domaine de destination évalue cette demande en fonction de sa stratégie d'accès au domaine. Vous devez inclure l'autorisation `es:ESCrossClusterGet` sur le domaine de destination :

   JSON

```
{
  "Version": "2012-10-17",
```

```
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:ESCrossClusterGet",
      "Resource": "arn:aws:es:region:account:domain/dst-domain"
    }
  ]
}
```

Assurez-vous que l'autorisation `es:ESCrossClusterGet` est appliquée pour `/dst-domain` et non pour `/dst-domain/*`.

Toutefois, cette politique minimale autorise uniquement les recherches inter-clusters. Pour effectuer d'autres opérations, notamment l'indexation de documents et l'exécution de recherches standard, vous avez besoin d'autorisations supplémentaires. Nous recommandons la politique suivante sur le domaine de destination :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "*"
        ]
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:region:account:domain/dst-domain/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
```

```
      },
      "Action": "es:ESCrossClusterGet",
      "Resource": "arn:aws:es:region:account:domain/dst-domain"
    }
  ]
}
```

> **Note**
>
> Toutes les demandes de recherche entre clusters entre domaines sont cryptées en transit par défaut dans le cadre du node-to-node chiffrement.

5. Le domaine de destination effectue la recherche et renvoie les résultats au domaine source.
6. Le domaine source combine ses propres résultats (le cas échéant) avec ceux du domaine de destination et vous les renvoie.
7. Nous recommandons Postman pour les demandes de test :

   - Sur le domaine de destination, indexez un document :

   ```
   POST https://dst-domain.us-east-1.es.amazonaws.com/books/_doc/1

   {
     "Dracula": "Bram Stoker"
   }
   ```

   - Pour interroger cet index à partir du domaine source, incluez l'alias de connexion du domaine de destination dans la requête.

   ```
   GET https://src-domain.us-east-1.es.amazonaws.com/<connection_alias>:books/
   _search

   {
       ...
     "hits": [
       {
         "_index": "source-destination:books",
         "_type": "_doc",
         "_id": "1",
         "_score": 1,
         "_source": {
   ```

```
        "Dracula": "Bram Stoker"
      }
    }
  ]
}
```

Vous trouverez l'alias de connexion dans l'onglet Connexions du tableau de bord de votre domaine.

- Si vous configurez une connexion entre `domain-a -> domain-b` avec un alias de connexion `cluster_b` et `domain-a -> domain-c` avec un alias de connexion `cluster_c`, recherchez `domain-a`, `domain-b` et `domain-c` comme suit :

```
GET https://src-domain.us-east-1.es.amazonaws.com/
local_index,cluster_b:b_index,cluster_c:c_index/_search
{
  "query": {
    "match": {
      "user": "domino"
    }
  }
}
```

Réponse

```
{
  "took": 150,
  "timed_out": false,
  "_shards": {
    "total": 3,
    "successful": 3,
    "failed": 0,
    "skipped": 0
  },
  "_clusters": {
    "total": 3,
    "successful": 3,
    "skipped": 0
  },
  "hits": {
    "total": 3,
    "max_score": 1,
```

```
    "hits": [
      {
        "_index": "local_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 1,
        "_source": {
          "user": "domino",
          "message": "This is message 1",
          "likes": 0
        }
      },
      {
        "_index": "cluster_b:b_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 2,
        "_source": {
          "user": "domino",
          "message": "This is message 2",
          "likes": 0
        }
      },
      {
        "_index": "cluster_c:c_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 3,
        "_source": {
          "user": "domino",
          "message": "This is message 3",
          "likes": 0
        }
      }
    ]
  }
}
```

Si vous n'avez pas choisi d'ignorer les clusters non disponibles dans la configuration de votre connexion, tous les clusters de destination que vous recherchez doivent être disponibles pour que votre demande de recherche s'exécute correctement. Sinon, la requête entière échoue. Même si l'un des domaines n'est pas disponible, aucun résultat de recherche n'est renvoyé.

OpenSearch Tableaux de bord

Vous pouvez visualiser les données de plusieurs domaines connectés de la même manière qu'à partir d'un seul domaine, sauf que vous devez accéder aux index distants à l'aide de `connection-alias:index`. Donc, votre modèle d'index doit correspondre à `connection-alias:index`.

Learning to Rank pour Amazon OpenSearch Service

OpenSearch utilise un cadre de classement probabiliste appelé BM-25 pour calculer les scores de pertinence. Si un mot-clé distinctif apparaît plus fréquemment dans un document, BM-25 attribue un score de pertinence plus élevé à ce document. Ce cadre ne tient cependant pas compte du comportement de l'utilisateur (comme les données de clics) qui peut améliorer la pertinence.

Learning to Rank est un plugin open source qui vous permet d'utiliser le machine learning et les données comportementales pour affiner la pertinence des documents. Il utilise des modèles issus des bibliothèques XGBoost et Ranklib pour réévaluer les résultats des recherches. Le plugin Elasticsearch LTR a été initialement développé par OpenSource Connections, avec des contributions significatives de Wikimedia Foundation, Snagajob Engineering, Bonsai, et Yelp Engineering. La OpenSearch version du plugin est dérivée du plugin Elasticsearch LTR.

Learning to Rank nécessite Elasticsearch 7.7 OpenSearch ou une version ultérieure. Pour utiliser le plugin Learning to Rank, vous devez disposer des autorisations d'administrateur complètes. Pour en savoir plus, veuillez consulter la section the section called “Modification de l'utilisateur maître”.

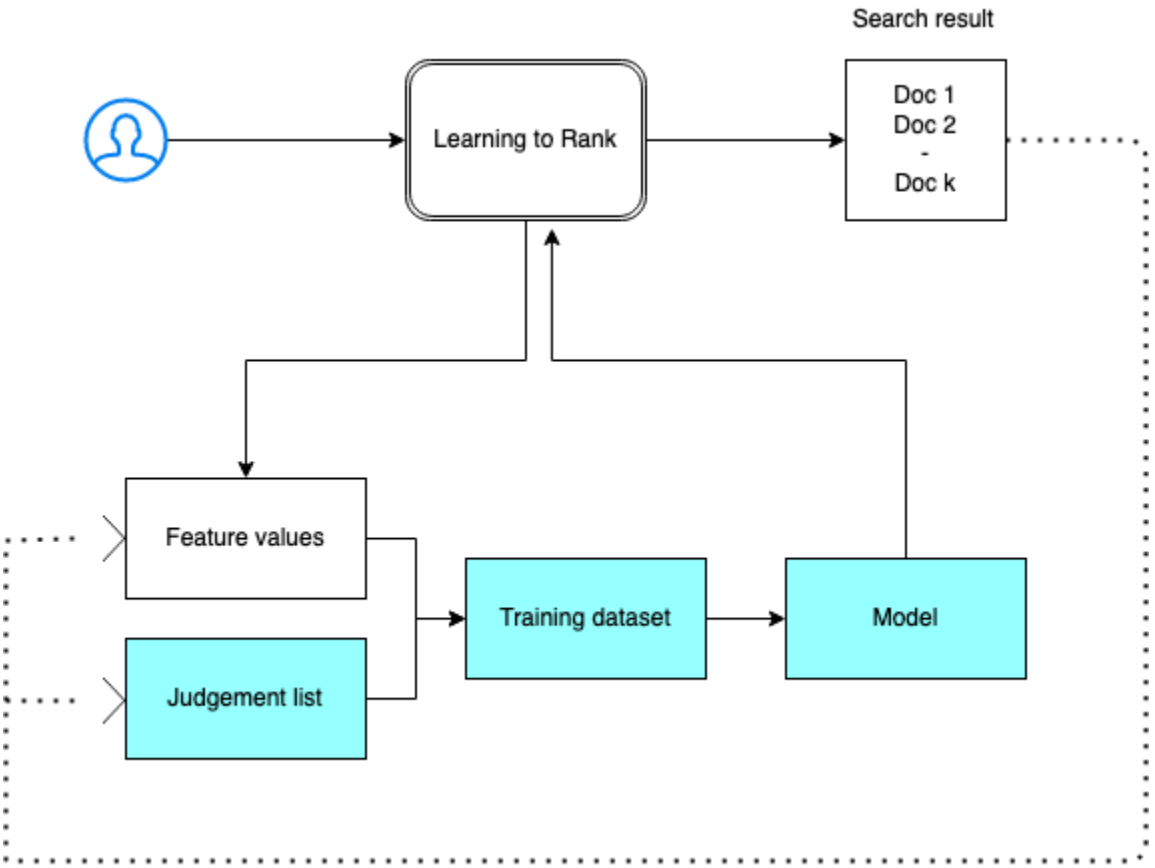
> **Note**
>
> Cette documentation fournit un aperçu général du plugin Learning to Rank et vous aide à commencer à l'utiliser. La documentation complète, avec les étapes détaillées et les descriptions des API, est disponible dans la documentation Learning to Rank.

Rubriques

- Prise en main de Learning to Rank
- API Learning to Rank

Prise en main de Learning to Rank

Vous devez fournir une liste de jugement, préparer un jeu de données d'entraînement et entraîner le modèle en dehors d'Amazon OpenSearch Service. Les parties signalées en bleu se font en dehors du OpenSearch Service :



Étape 1 : Initialiser le plugin

Pour initialiser le plugin Learning to Rank, envoyez la demande suivante à votre domaine OpenSearch Service :

```
PUT _ltr
```

```
{
  "acknowledged" : true,
  "shards_acknowledged" : true,
  "index" : ".ltrstore"
}
```

Cette commande crée un index `.ltrstore` caché qui stocke les informations de métadonnées, telles que les ensembles de fonctions et les modèles.

Étape 2 : Créer une liste de jugements

> **Note**
>
> Vous devez effectuer cette étape en dehors du OpenSearch Service.

Une liste de jugements est un ensemble d'exemples à partir desquels un modèle de machine learning apprend. Votre liste de jugements doit inclure les mots-clés importants pour vous et un ensemble de documents notés pour chaque mot-clé.

Dans cet exemple, nous disposons d'une liste de jugements pour un jeu de données de films. Une note de 4 indique une correspondance parfaite. Une note de 0 indique la pire correspondance.

Note	Mot clé	ID du document	Titre du film
4	rambo	7555	Rambo
3	rambo	1370	Rambo III
3	rambo	1369	Rambo 2 : La Mission
3	rambo	1368	La Mission

Préparez une liste de jugements au format suivant :

```
4 qid:1 # 7555 Rambo
3 qid:1 # 1370 Rambo III
3 qid:1 # 1369 Rambo: First Blood Part II
3 qid:1 # 1368 First Blood

where qid:1 represents "rambo"
```

Pour un exemple plus complet de liste de jugements, consultez jugements de films.

Vous pouvez créer cette liste de jugements manuellement avec l'aide d'annotateurs humains ou la déduire par programmation à partir de données analytiques.

Étape 3 : Créer un ensemble de fonctions

Une fonction est un champ qui correspond à la pertinence d'un document ; par exemple, `title`, `overview`, `popularity score` (nombre de vues), etc.

Créez un ensemble de fonctions avec un modèle Mustache pour chaque caractéristique. Pour plus d'informations sur les fonctions, consultez Utilisation des fonctions.

Dans cet exemple, nous créons un ensemble de caractéristiques `movie_features` avec les champs `title` et `overview` :

```
POST _ltr/_featureset/movie_features
{
  "featureset" : {
      "name" : "movie_features",
      "features" : [
        {
          "name" : "1",
          "params" : [
            "keywords"
          ],
          "template_language" : "mustache",
          "template" : {
            "match" : {
              "title" : "{{keywords}}"
            }
          }
        },
        {
          "name" : "2",
          "params" : [
            "keywords"
          ],
          "template_language" : "mustache",
          "template" : {
            "match" : {
              "overview" : "{{keywords}}"
            }
          }
        }
      ]
    }
}
```

Si vous interrogez l'index `.ltrstore` d'origine, vous récupérez votre ensemble de fonctions :

```
GET _ltr/_featureset
```

Étape 4 : Journaliser les valeurs des fonctions

Les valeurs de fonction sont les scores de pertinence calculés par BM-25 pour chaque fonction.

Combinez l'ensemble de caractéristiques et la liste de jugements pour journaliser les valeurs de caractéristique. Pour plus d'informations sur la journalisation des fonctions, consultez Journalisation des scores des fonctions.

Dans cet exemple, la requête `bool` récupère les documents notés avec le filtre, puis sélectionne l'ensemble de fonctions avec la requête `sltr`. La requête `ltr_log` combine les documents et les fonctions pour journaliser les valeurs de fonctions correspondantes :

```
POST tmdb/_search
{
  "_source": {
    "includes": [
      "title",
      "overview"
    ]
  },
  "query": {
    "bool": {
      "filter": [
        {
          "terms": {
            "_id": [
              "7555",
              "1370",
              "1369",
              "1368"
            ]
          }
        },
        {
          "sltr": {
            "_name": "logged_featureset",
            "featureset": "movie_features",
            "params": {
```

```
              "keywords": "rambo"
            }
          }
        }
      ]
    }
  },
  "ext": {
    "ltr_log": {
      "log_specs": {
        "name": "log_entry1",
        "named_query": "logged_featureset"
      }
    }
  }
}
```

Un exemple de réponse peut ressembler à ce qui suit :

```
{
  "took" : 7,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 4,
      "relation" : "eq"
    },
    "max_score" : 0.0,
    "hits" : [
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1368",
        "_score" : 0.0,
        "_source" : {
          "overview" : "When former Green Beret John Rambo is harassed by local law
 enforcement and arrested for vagrancy, the Vietnam vet snaps, runs for the hills and
```

```
rat-a-tat-tats his way into the action-movie hall of fame. Hounded by a relentless
sheriff, Rambo employs heavy-handed guerilla tactics to shake the cops off his tail.",
        "title" : "First Blood"
      },
      "fields" : {
        "_ltrlog" : [
          {
            "log_entry1" : [
              {
                "name" : "1"
              },
              {
                "name" : "2",
                "value" : 10.558305
              }
            ]
          }
        ]
      },
      "matched_queries" : [
        "logged_featureset"
      ]
    },
    {
      "_index" : "tmdb",
      "_type" : "movie",
      "_id" : "7555",
      "_score" : 0.0,
      "_source" : {
        "overview" : "When governments fail to act on behalf of captive missionaries,
ex-Green Beret John James Rambo sets aside his peaceful existence along the Salween
River in a war-torn region of Thailand to take action.  Although he's still haunted
by violent memories of his time as a U.S. soldier during the Vietnam War, Rambo can
hardly turn his back on the aid workers who so desperately need his help.",
        "title" : "Rambo"
      },
      "fields" : {
        "_ltrlog" : [
          {
            "log_entry1" : [
              {
                "name" : "1",
                "value" : 11.2569065
              },
```

```
              {
                "name" : "2",
                "value" : 9.936821
              }
            ]
          }
        ]
      },
      "matched_queries" : [
        "logged_featureset"
      ]
    },
    {
      "_index" : "tmdb",
      "_type" : "movie",
      "_id" : "1369",
      "_score" : 0.0,
      "_source" : {
        "overview" : "Col. Troutman recruits ex-Green Beret John Rambo for a highly
secret and dangerous mission. Teamed with Co Bao, Rambo goes deep into Vietnam to
rescue POWs. Deserted by his own team, he's left in a hostile jungle to fight for his
life, avenge the death of a woman and bring corrupt officials to justice.",
        "title" : "Rambo: First Blood Part II"
      },
      "fields" : {
        "_ltrlog" : [
          {
            "log_entry1" : [
              {
                "name" : "1",
                "value" : 6.334839
              },
              {
                "name" : "2",
                "value" : 10.558305
              }
            ]
          }
        ]
      },
      "matched_queries" : [
        "logged_featureset"
      ]
    },
```

```
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1370",
        "_score" : 0.0,
        "_source" : {
          "overview" : "Combat has taken its toll on Rambo, but he's finally begun to
 find inner peace in a monastery. When Rambo's friend and mentor Col. Trautman asks for
 his help on a top secret mission to Afghanistan, Rambo declines but must reconsider
 when Trautman is captured.",
          "title" : "Rambo III"
        },
        "fields" : {
          "_ltrlog" : [
            {
              "log_entry1" : [
                {
                  "name" : "1",
                  "value" : 9.425955
                },
                {
                  "name" : "2",
                  "value" : 11.262714
                }
              ]
            }
          ]
        },
        "matched_queries" : [
          "logged_featureset"
        ]
      }
    ]
  }
}
```

Dans l'exemple précédent, la première fonction n'a pas de valeur de fonction, car le mot-clé « rambo » n'apparaît pas dans le champ « title » du document dont l'ID est égal à 1368. Il s'agit d'une valeur de fonction manquante dans les données de formation.

Étape 5 : Créer un ensemble de données de formation

> **Note**
>
> Vous devez effectuer cette étape en dehors du OpenSearch Service.

L'étape suivante consiste à combiner la liste de jugements et les valeurs de fonction pour créer un jeu de données de formation. Si votre liste de jugements d'origine est semblable à ce qui suit :

```
4 qid:1 # 7555 Rambo
3 qid:1 # 1370 Rambo III
3 qid:1 # 1369 Rambo: First Blood Part II
3 qid:1 # 1368 First Blood
```

Convertissez-la en jeu de données de formation final comme celui-ci :

```
4 qid:1 1:12.318474 2:10.573917 # 7555 rambo
3 qid:1 1:10.357875 2:11.950391 # 1370 rambo
3 qid:1 1:7.010513 2:11.220095 # 1369 rambo
3 qid:1 1:0.0 2:11.220095 # 1368 rambo
```

Vous pouvez effectuer cette étape manuellement ou écrire un programme pour l'automatiser.

Étape 6 : Choisir un algorithme et créer le modèle

> **Note**
>
> Vous devez effectuer cette étape en dehors du OpenSearch Service.

Une fois le jeu de données de formation en place, l'étape suivante consiste à utiliser XGBoost les bibliothèques Ranklib pour créer un modèle. XGBoost et les bibliothèques Ranklib vous permettent de créer des modèles courants, tels que LambdaMART, Random Forests, etc.

Pour connaître les étapes à suivre XGBoost et Ranklib pour créer le modèle, consultez respectivement la RankLibdocumentation XGBoostet. Pour utiliser Amazon SageMaker pour créer le XGBoost modèle, consultez XGBoostAlgorithme.

Étape 7 : Déployer le modèle

Une fois le modèle créé, vous devez le déployer dans le plugin Learning to Rank. Pour plus d'informations sur le déploiement d'un modèle, consultez Téléchargement d'un modèle entraîné.

Dans cet exemple, nous créons un modèle `my_ranklib_model` à l'aide de la bibliothèque Ranklib :

```
POST _ltr/_featureset/movie_features/_createmodel?pretty
{
  "model": {
    "name": "my_ranklib_model",
    "model": {
      "type": "model/ranklib",
      "definition": """## LambdaMART
## No. of trees = 10
## No. of leaves = 10
## No. of threshold candidates = 256
## Learning rate = 0.1
## Stop early = 100

<ensemble>
   <tree id="1" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>0.0</threshold>
            <split pos="left">
               <output>-2.0</output>
            </split>
            <split pos="right">
               <feature>1</feature>
               <threshold>7.010513</threshold>
               <split pos="left">
                  <output>-2.0</output>
               </split>
               <split pos="right">
                  <output>-2.0</output>
               </split>
            </split>
         </split>
         <split pos="right">
```

```
            <output>2.0</output>
         </split>
      </split>
   </tree>
   <tree id="2" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>0.0</threshold>
            <split pos="left">
               <output>-1.67031991481781</output>
            </split>
            <split pos="right">
               <feature>1</feature>
               <threshold>7.010513</threshold>
               <split pos="left">
                  <output>-1.67031991481781</output>
               </split>
               <split pos="right">
                  <output>-1.6703200340270996</output>
               </split>
            </split>
         </split>
         <split pos="right">
            <output>1.6703201532363892</output>
         </split>
      </split>
   </tree>
   <tree id="3" weight="0.1">
      <split>
         <feature>2</feature>
         <threshold>10.573917</threshold>
         <split pos="left">
            <output>1.479954481124878</output>
         </split>
         <split pos="right">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
```

```
                  <output>-1.4799546003341675</output>
               </split>
               <split pos="right">
                  <output>-1.479954481124878</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.479954481124878</output>
            </split>
         </split>
      </split>
   </tree>
   <tree id="4" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>0.0</threshold>
            <split pos="left">
               <output>-1.3569872379302979</output>
            </split>
            <split pos="right">
               <feature>1</feature>
               <threshold>7.010513</threshold>
               <split pos="left">
                  <output>-1.3569872379302979</output>
               </split>
               <split pos="right">
                  <output>-1.3569872379302979</output>
               </split>
            </split>
         </split>
         <split pos="right">
            <output>1.3569873571395874</output>
         </split>
      </split>
   </tree>
   <tree id="5" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
```

```
            <threshold>0.0</threshold>
            <split pos="left">
               <output>-1.2721362113952637</output>
            </split>
            <split pos="right">
               <feature>1</feature>
               <threshold>7.010513</threshold>
               <split pos="left">
                  <output>-1.2721363306045532</output>
               </split>
               <split pos="right">
                  <output>-1.2721363306045532</output>
               </split>
            </split>
         </split>
         <split pos="right">
            <output>1.2721362113952637</output>
         </split>
      </split>
   </tree>
   <tree id="6" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.2110036611557007</output>
               </split>
               <split pos="right">
                  <output>-1.2110036611557007</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.2110037803649902</output>
            </split>
         </split>
         <split pos="right">
            <output>1.2110037803649902</output>
         </split>
```

```
      </split>
   </tree>
   <tree id="7" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.165616512298584</output>
               </split>
               <split pos="right">
                  <output>-1.165616512298584</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.165616512298584</output>
            </split>
         </split>
         <split pos="right">
            <output>1.165616512298584</output>
         </split>
      </split>
   </tree>
   <tree id="8" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.131177544593811</output>
               </split>
               <split pos="right">
                  <output>-1.131177544593811</output>
               </split>
```

```
            </split>
            <split pos="right">
               <output>-1.131177544593811</output>
            </split>
         </split>
         <split pos="right">
            <output>1.131177544593811</output>
         </split>
      </split>
   </tree>
   <tree id="9" weight="0.1">
      <split>
         <feature>2</feature>
         <threshold>10.573917</threshold>
         <split pos="left">
            <output>1.1046180725097656</output>
         </split>
         <split pos="right">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.1046180725097656</output>
               </split>
               <split pos="right">
                  <output>-1.1046180725097656</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.1046180725097656</output>
            </split>
         </split>
      </split>
   </tree>
   <tree id="10" weight="0.1">
      <split>
         <feature>1</feature>
         <threshold>10.357875</threshold>
         <split pos="left">
            <feature>1</feature>
            <threshold>7.010513</threshold>
            <split pos="left">
```

```
               <feature>1</feature>
               <threshold>0.0</threshold>
               <split pos="left">
                  <output>-1.0838804244995117</output>
               </split>
               <split pos="right">
                  <output>-1.0838804244995117</output>
               </split>
            </split>
            <split pos="right">
               <output>-1.0838804244995117</output>
            </split>
         </split>
         <split pos="right">
            <output>1.0838804244995117</output>
         </split>
      </split>
   </tree>
</ensemble>
"""
    }
  }
}
```

Pour voir le modèle, envoyez la requête suivante :

```
GET _ltr/_model/my_ranklib_model
```

Étape 8 : Effectuer une recherche avec Learning to Rank

Une fois le modèle déployé, vous pouvez effectuer une recherche.

Lancez la requête `sltr` avec les fonctions que vous utilisez et le nom du modèle que vous souhaitez exécuter :

```
POST tmdb/_search
{
  "_source": {
    "includes": ["title", "overview"]
  },
  "query": {
    "multi_match": {
```

```
      "query": "rambo",
      "fields": ["title", "overview"]
    }
  },
  "rescore": {
    "query": {
      "rescore_query": {
        "sltr": {
          "params": {
            "keywords": "rambo"
          },
          "model": "my_ranklib_model"
        }
      }
    }
  }
}
```

Avec Learning to Rank, « Rambo » est le premier résultat qui apparaît, car nous lui avons attribué la note la plus élevée dans la liste de jugements :

```
{
  "took" : 12,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 7,
      "relation" : "eq"
    },
    "max_score" : 13.096414,
    "hits" : [
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "7555",
        "_score" : 13.096414,
        "_source" : {
```

```
        "overview" : "When governments fail to act on behalf of captive missionaries,
ex-Green Beret John James Rambo sets aside his peaceful existence along the Salween
River in a war-torn region of Thailand to take action.  Although he's still haunted
by violent memories of his time as a U.S. soldier during the Vietnam War, Rambo can
hardly turn his back on the aid workers who so desperately need his help.",
        "title" : "Rambo"
      }
    },
    {
      "_index" : "tmdb",
      "_type" : "movie",
      "_id" : "1370",
      "_score" : 11.17245,
      "_source" : {
        "overview" : "Combat has taken its toll on Rambo, but he's finally begun to
find inner peace in a monastery. When Rambo's friend and mentor Col. Trautman asks for
his help on a top secret mission to Afghanistan, Rambo declines but must reconsider
when Trautman is captured.",
        "title" : "Rambo III"
      }
    },
    {
      "_index" : "tmdb",
      "_type" : "movie",
      "_id" : "1368",
      "_score" : 10.442155,
      "_source" : {
        "overview" : "When former Green Beret John Rambo is harassed by local law
enforcement and arrested for vagrancy, the Vietnam vet snaps, runs for the hills and
rat-a-tat-tats his way into the action-movie hall of fame. Hounded by a relentless
sheriff, Rambo employs heavy-handed guerilla tactics to shake the cops off his tail.",
        "title" : "First Blood"
      }
    },
    {
      "_index" : "tmdb",
      "_type" : "movie",
      "_id" : "1369",
      "_score" : 10.442155,
      "_source" : {
        "overview" : "Col. Troutman recruits ex-Green Beret John Rambo for a highly
secret and dangerous mission. Teamed with Co Bao, Rambo goes deep into Vietnam to
rescue POWs. Deserted by his own team, he's left in a hostile jungle to fight for his
life, avenge the death of a woman and bring corrupt officials to justice.",
```

```
          "title" : "Rambo: First Blood Part II"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "31362",
        "_score" : 7.424202,
        "_source" : {
          "overview" : "It is 1985, and a small, tranquil Florida town is being rocked
by a wave of vicious serial murders and bank robberies. Particularly sickening to the
authorities is the gratuitous use of violence by two “Rambo” like killers who dress
themselves in military garb. Based on actual events taken from FBI files, the movie
depicts the Bureau’s efforts to track down these renegades.",
          "title" : "In the Line of Duty: The F.B.I. Murders"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "13258",
        "_score" : 6.43182,
        "_source" : {
          "overview" : """Will Proudfoot (Bill Milner) is looking for an escape from
his family's stifling home life when he encounters Lee Carter (Will Poulter), the
school bully. Armed with a video camera and a copy of "Rambo: First Blood", Lee plans
to make cinematic history by filming his own action-packed video epic. Together, these
two newfound friends-turned-budding-filmmakers quickly discover that their imaginative
– and sometimes mishap-filled – cinematic adventure has begun to take on a life of its
own!""",
          "title" : "Son of Rambow"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "61410",
        "_score" : 3.9719706,
        "_source" : {
          "overview" : "It's South Africa 1990. Two major events are about to happen:
The release of Nelson Mandela and, more importantly, it's Spud Milton's first year
at an elite boys only private boarding school. John Milton is a boy from an ordinary
background who wins a scholarship to a private school in Kwazulu-Natal, South Africa.
Surrounded by boys with nicknames like Gecko, Rambo, Rain Man and Mad Dog, Spud has
```

```
 his hands full trying to adapt to his new home. Along the way Spud takes his first
 tentative steps along the path to manhood. (The path it seems could be a rather long
 road). Spud is an only child. He is cursed with parents from well beyond the lunatic
 fringe and a senile granny. His dad is a fervent anti-communist who is paranoid that
 the family domestic worker is running a shebeen from her room at the back of the
 family home. His mom is a free spirit and a teenager's worst nightmare, whether it's
 shopping for Spud's underwear in the local supermarket",
          "title" : "Spud"
        }
      }
    ]
  }
}
```

Si vous effectuez une recherche sans utiliser le plugin Learning to Rank, elle OpenSearch renvoie des résultats différents :

```
POST tmdb/_search
{
  "_source": {
    "includes": ["title", "overview"]
  },
  "query": {
    "multi_match": {
      "query": "Rambo",
      "fields": ["title", "overview"]
    }
  }
}
```

```
{
  "took" : 5,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 5,
      "relation" : "eq"
```

```
    },
    "max_score" : 11.262714,
    "hits" : [
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1370",
        "_score" : 11.262714,
        "_source" : {
          "overview" : "Combat has taken its toll on Rambo, but he's finally begun to
find inner peace in a monastery. When Rambo's friend and mentor Col. Trautman asks for
his help on a top secret mission to Afghanistan, Rambo declines but must reconsider
when Trautman is captured.",
          "title" : "Rambo III"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "7555",
        "_score" : 11.2569065,
        "_source" : {
          "overview" : "When governments fail to act on behalf of captive missionaries,
ex-Green Beret John James Rambo sets aside his peaceful existence along the Salween
River in a war-torn region of Thailand to take action.  Although he's still haunted
by violent memories of his time as a U.S. soldier during the Vietnam War, Rambo can
hardly turn his back on the aid workers who so desperately need his help.",
          "title" : "Rambo"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1368",
        "_score" : 10.558305,
        "_source" : {
          "overview" : "When former Green Beret John Rambo is harassed by local law
enforcement and arrested for vagrancy, the Vietnam vet snaps, runs for the hills and
rat-a-tat-tats his way into the action-movie hall of fame. Hounded by a relentless
sheriff, Rambo employs heavy-handed guerilla tactics to shake the cops off his tail.",
          "title" : "First Blood"
        }
      },
      {
```

```
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "1369",
        "_score" : 10.558305,
        "_source" : {
          "overview" : "Col. Troutman recruits ex-Green Beret John Rambo for a highly
 secret and dangerous mission. Teamed with Co Bao, Rambo goes deep into Vietnam to
 rescue POWs. Deserted by his own team, he's left in a hostile jungle to fight for his
 life, avenge the death of a woman and bring corrupt officials to justice.",
          "title" : "Rambo: First Blood Part II"
        }
      },
      {
        "_index" : "tmdb",
        "_type" : "movie",
        "_id" : "13258",
        "_score" : 6.4600153,
        "_source" : {
          "overview" : """Will Proudfoot (Bill Milner) is looking for an escape from
 his family's stifling home life when he encounters Lee Carter (Will Poulter), the
 school bully. Armed with a video camera and a copy of "Rambo: First Blood", Lee plans
 to make cinematic history by filming his own action-packed video epic. Together, these
 two newfound friends-turned-budding-filmmakers quickly discover that their imaginative
 – and sometimes mishap-filled – cinematic adventure has begun to take on a life of its
 own!""",
          "title" : "Son of Rambow"
        }
      }
    ]
  }
}
```

En fonction des performances estimées du modèle, ajustez la liste de jugements et les fonctions. Répétez ensuite les étapes 2 à 8 pour améliorer les résultats du classement au fil du temps.

API Learning to Rank

Utilisez les opérations Learning to Rank pour travailler par programmation avec des ensembles de fonctions et des modèles.

Créer un magasin

Crée un index `.ltrstore` caché qui stocke les informations de métadonnées, telles que les ensembles de fonctions et les modèles.

```
PUT _ltr
```

Supprimer le magasin

Supprime l'index `.ltrstore` caché et réinitialise le plugin.

```
DELETE _ltr
```

Créer un ensemble de fonctions

Crée un ensemble de fonctions.

```
POST _ltr/_featureset/<name_of_features>
```

Supprimer un ensemble de fonctions

Supprime un ensemble de fonctions.

```
DELETE _ltr/_featureset/<name_of_feature_set>
```

Obtenir un ensemble de fonctions

Récupère un ensemble de fonctions.

```
GET _ltr/_featureset/<name_of_feature_set>
```

Créer un modèle

Crée un modèle.

```
POST _ltr/_featureset/<name_of_feature_set>/_createmodel
```

Supprimer un modèle

Supprime un modèle.

```
DELETE _ltr/_model/<name_of_model>
```

Obtenir un modèle

Récupère un modèle.

```
GET _ltr/_model/<name_of_model>
```

Obtenir des statistiques

Fournit des informations sur le comportement du plugin.

```
GET _ltr/_stats
```

Vous pouvez également utiliser des filtres pour récupérer une statistique unique :

```
GET _ltr/_stats/<stat>
```

En outre, vous pouvez limiter les informations à un seul nœud du cluster :

```
GET _ltr/_stats/<stat>/nodes/<nodeId>

{
  "_nodes" : {
    "total" : 1,
    "successful" : 1,
    "failed" : 0
  },
  "cluster_name" : "873043598401:ltr-77",
  "stores" : {
    ".ltrstore" : {
      "model_count" : 1,
      "featureset_count" : 1,
      "feature_count" : 2,
      "status" : "green"
    }
  },
  "status" : "green",
  "nodes" : {
    "DjelK-_ZSfyzstO5dhGGQA" : {
```

```
      "cache" : {
        "feature" : {
          "eviction_count" : 0,
          "miss_count" : 0,
          "entry_count" : 0,
          "memory_usage_in_bytes" : 0,
          "hit_count" : 0
        },
        "featureset" : {
          "eviction_count" : 2,
          "miss_count" : 2,
          "entry_count" : 0,
          "memory_usage_in_bytes" : 0,
          "hit_count" : 0
        },
        "model" : {
          "eviction_count" : 2,
          "miss_count" : 3,
          "entry_count" : 1,
          "memory_usage_in_bytes" : 3204,
          "hit_count" : 1
        }
      },
      "request_total_count" : 6,
      "request_error_count" : 0
    }
  }
}
```

Les statistiques sont fournies à deux niveaux, nœud et cluster, comme indiqué dans les tableaux suivants :

Statistiques de niveau nœud

Nom de champ	Description
request_total_count	Nombre total de demandes de classement.
request_error_count	Nombre total de demandes qui ont échoué.
cache	Statistiques sur tous les caches (fonction s, ensembles de fonctions, modèles). Un accès au cache se produit lorsqu'un utilisate

Nom de champ	Description
	ur interroge le plugin et que le modèle est déjà chargé en mémoire.
cache.eviction_count	Nombre d'évictions du cache.
cache.hit_count	Nombre d'accès au cache.
cache.miss_count	Nombre d'échecs d'accès au cache. Un échec d'accès au cache se produit lorsqu'un utilisate ur interroge le plugin et que le modèle n'a pas encore été chargé en mémoire.
cache.entry_count	Nombre d'entrées dans le cache.
cache.memory_usage_in_bytes	Mémoire totale utilisée en octets.
cache.cache_capacity_reached	Indique si la limite de cache est atteinte.

Statistiques de niveau cluster

Nom de champ	Description
stores	Indique où sont stockés les ensembles de fonctions et les métadonnées du modèle. (Le magasin par défaut est « .ltrstore ». Sinon, le préfixe « .ltrstore_ » est ajouté à un nom fourni par l'utilisateur).
stores.status	Statut de l'index.
stores.feature_sets	Nombre d'ensembles de fonctions.
stores.features_count	Nombre de fonctions.
stores.model_count	Nombre de modèles.

Nom de champ	Description
status	Statut du plugin basé sur celui des index du Feature Store (rouge, jaune ou vert) et sur l'état du disjoncteur (ouvert ou fermé).
cache.cache_capacity_reached	Indique si la limite de cache est atteinte.

Obtenir les statistiques du cache

Renvoie des statistiques relatives à l'utilisation du cache et de la mémoire.

```
GET _ltr/_cachestats

{
    "_nodes": {
        "total": 2,
        "successful": 2,
        "failed": 0
    },
    "cluster_name": "opensearch-cluster",
    "all": {
        "total": {
            "ram": 612,
            "count": 1
        },
        "features": {
            "ram": 0,
            "count": 0
        },
        "featuresets": {
            "ram": 612,
            "count": 1
        },
        "models": {
            "ram": 0,
            "count": 0
        }
    },
    "stores": {
        ".ltrstore": {
```

```
            "total": {
                "ram": 612,
                "count": 1
            },
            "features": {
                "ram": 0,
                "count": 0
            },
            "featuresets": {
                "ram": 612,
                "count": 1
            },
            "models": {
                "ram": 0,
                "count": 0
            }
        }
    },
    "nodes": {
        "ejF6uutERF20wOFNOXB61A": {
            "name": "opensearch1",
            "hostname": "172.18.0.4",
            "stats": {
                "total": {
                    "ram": 612,
                    "count": 1
                },
                "features": {
                    "ram": 0,
                    "count": 0
                },
                "featuresets": {
                    "ram": 612,
                    "count": 1
                },
                "models": {
                    "ram": 0,
                    "count": 0
                }
            }
        },
        "Z2RZNWRLSveVcz2c6lHf5A": {
            "name": "opensearch2",
            "hostname": "172.18.0.2",
```

```
            "stats": {
                ...
            }
        }
    }
}
```

Effacer le cache

Efface le cache du plugin. Utilisez cette opération pour actualiser le modèle.

```
POST _ltr/_clearcache
```

Recherche asynchrone dans Amazon Service OpenSearch

La recherche asynchrone pour Amazon OpenSearch Service vous permet d'envoyer une requête de recherche qui sera exécutée en arrière-plan, de surveiller la progression de la demande et de récupérer les résultats ultérieurement. Vous pourrez récupérer des résultats partiels avant la fin de la recherche dès que ceux-ci seront disponibles. Au terme de la recherche, enregistrez les résultats pour les récupérer et les analyser ultérieurement.

La fonctionnalité de recherche asynchrone requiert la OpenSearch version ultérieure, ou Elasticsearch version 7.10 ou ultérieure.

Cette documentation fournit un bref aperçu de la recherche asynchrone. Il décrit également les limites de l'utilisation de la recherche asynchrone avec un domaine Amazon OpenSearch Service géré plutôt qu'un cluster open source OpenSearch . Pour une documentation complète sur la recherche asynchrone, y compris les paramètres disponibles, les autorisations et une référence d'API complète, consultez la section Recherche asynchrone dans la documentation. OpenSearch

Exemple d'appel de recherche

Pour effectuer une recherche asynchrone, envoyez des demandes HTTP à `_plugins/_asynchronous_search` en utilisant le format suivant :

```
POST opensearch-domain/_plugins/_asynchronous_search
```

> **Note**
>
> Si vous utilisez Elasticsearch 7.10 au lieu d'une OpenSearch version ultérieure, remplacez par `_opendistro` dans toutes les requêtes `_plugins` de recherche asynchrones.

Vous pouvez spécifier les options de recherche asynchrone suivantes :

Options	Description	Valeur par défaut	Obligatoire
`wait_for_completion_timeout`	Spécifie le délai d'attente des résultats. Vous pouvez voir tous les résultats obtenus au cours de cette période, comme pour une recherche normale. Vous pouvez interroger les résultats restants à partir d'un ID. La valeur maximale est de 300 secondes.	1 seconde	Non
`keep_on_completion`	Indique si vous souhaitez enregistrer les résultats dans le cluster une fois la recherche terminée. Vous pourrez examiner les résultats enregistrés ultérieurement.	`false`	Non
`keep_alive`	Spécifie la durée pendant laquelle le résultat est conservé dans le cluster. Par exemple, `2d` signifie que les résultats sont stockés dans le cluster pendant 48 heures. Passé ce délai, ou en cas d'annulation de la recherche, les résultats enregistrés sont supprimés. Notez que cela inclut l'exécution de la requête. En cas de dépassement de ce délai, le processus annule automatiquement cette requête.	12 heures	Non

Exemple de demande

```
POST _plugins/_asynchronous_search/?
pretty&size=10&wait_for_completion_timeout=1ms&keep_on_completion=true&request_cache=false
```

```
{
  "aggs": {
    "city": {
      "terms": {
        "field": "city",
        "size": 10
      }
    }
  }
}
```

> **Note**
>
> Tous les paramètres de demande qui s'appliquent à une requête `_search` standard sont pris en charge. Si vous utilisez Elasticsearch 7.10 au lieu d'une OpenSearch version ultérieure, remplacez-la par. `_plugins` `_opendistro`

Autorisations relatives à la recherche asynchrone

La recherche asynchrone prend en charge le contrôle précis des accès. Pour savoir comment panacher les autorisations et les adapter à votre cas d'utilisation, consultez Sécurité liée à la recherche asynchrone.

Pour les domaines où le contrôle précis des accès est activé, vous devez disposer des autorisations minimales suivantes pour un rôle :

```
# Allows users to use all asynchronous search functionality
asynchronous_search_full_access:
  reserved: true
  cluster_permissions:
    - 'cluster:admin/opensearch/asynchronous-search/*'
  index_permissions:
    - index_patterns:
        - '*'
      allowed_actions:
        - 'indices:data/read/search*'

# Allows users to read stored asynchronous search results
asynchronous_search_read_access:
  reserved: true
```

```
  cluster_permissions:
    - 'cluster:admin/opensearch/asynchronous-search/get'
```

Pour les domaines où le contrôle précis des accès est désactivé, utilisez votre accès IAM et votre clé secrète pour signer toutes les demandes. Vous pouvez accéder aux résultats à l'aide de l'ID de recherche asynchrone.

Paramètres de recherche asynchrone

OpenSearch vous permet de modifier tous les paramètres de recherche asynchrone disponibles à l'aide de l'`_cluster/settings`API. Dans OpenSearch Service, vous pouvez uniquement modifier les paramètres suivants :

- `plugins.asynchronous_search.node_concurrent_running_searches`
- `plugins.asynchronous_search.persist_search_failures`

Recherche croisée entre clusters

Vous pouvez effectuer une recherche asynchrone dans différents clusters avec les limitations mineures suivantes :

- Vous ne pouvez exécuter une recherche asynchrone que sur le domaine source.
- Vous ne pouvez pas minimiser les allers-retours réseau dans le cadre d'une requête de recherche croisée entre clusters.

Si vous configurez une connexion entre `domain-a -> domain-b` avec alias de connexion `cluster_b` et `domain-a -> domain-c` avec alias de connexion `cluster_c`, recherchez de manière asynchrone `domain-a`, `domain-b` et `domain-c` comme suit :

```
POST https://src-domain.us-east-1.es.amazonaws.com/
local_index,cluster_b:b_index,cluster_c:c_index/_plugins/_asynchronous_search/?
pretty&size=10&wait_for_completion_timeout=500ms&keep_on_completion=true&request_cache=false
{
  "size": 0,
  "_source": {
    "excludes": []
  },
  "aggs": {
    "2": {
```

```
      "terms": {
        "field": "clientip",
        "size": 50,
        "order": {
          "_count": "desc"
        }
      }
    }
  },
  "stored_fields": [
    "*"
  ],
  "script_fields": {},
  "docvalue_fields": [
    "@timestamp"
  ],
  "query": {
    "bool": {
      "must": [
        {
          "query_string": {
            "query": "status:404",
            "analyze_wildcard": true,
            "default_field": "*"
          }
        },
        {
          "range": {
            "@timestamp": {
              "gte": 1483747200000,
              "lte": 1488326400000,
              "format": "epoch_millis"
            }
          }
        }
      ],
      "filter": [],
      "should": [],
      "must_not": []
    }
  }
}
```

Réponse

```
{
  "id" :
 "Fm9pYzJyVG91U19xb0hIQUJnMHJfRFEAAAAAAAknghQ1OWVBczNZQjVEa2dMYTBXaTdEagAAAAAAAAAB",
  "state" : "RUNNING",
  "start_time_in_millis" : 1609329314796,
  "expiration_time_in_millis" : 1609761314796
}
```

Pour de plus amples informations, veuillez consulter the section called “Recherche croisée entre clusters”.

UltraWarm

Les recherches asynchrones avec les UltraWarm index continuent de fonctionner. Pour de plus amples informations, veuillez consulter the section called “UltraWarm rangement”.

> **Note**
>
> Vous pouvez surveiller les statistiques de recherche asynchrone dans. CloudWatch Pour accéder à une liste complète des métriques , veuillez consulter the section called “Métriques de recherche asynchrone”.

Recherche ponctuelle dans le temps dans Amazon OpenSearch Service

Le point in time (PIT) est un type de recherche qui vous permet d'exécuter différentes requêtes sur un ensemble de données fixe dans le temps. Généralement, lorsque vous exécutez la même requête sur le même index à différents moments, vous obtenez des résultats différents car les documents sont constamment indexés, mis à jour et supprimés. Avec PIT, vous pouvez effectuer des requêtes par rapport à un état constant de votre ensemble de données.

L'utilisation principale de la recherche PIT est de l'associer à des `search_after` fonctionnalités. Il s'agit de la méthode de pagination préférée OpenSearch, en particulier pour la pagination profonde, car elle fonctionne sur un ensemble de données figé dans le temps, elle n'est pas liée à une requête et elle permet une pagination cohérente en avant et en arrière. Vous pouvez utiliser PIT avec un domaine exécutant OpenSearch la version 2.5.

> **Note**
>
> Cette rubrique fournit une vue d'ensemble du PIT et de certains éléments à prendre en compte lors de son utilisation sur un domaine Amazon OpenSearch Service géré plutôt que sur un OpenSearch cluster autogéré. Pour une documentation complète du PIT, y compris une référence complète sur les API, voir Point in Time dans la OpenSearch documentation open source.

Considérations

Tenez compte des éléments suivants lorsque vous configurez vos recherches PIT :

- Si vous effectuez une mise à niveau depuis un domaine exécutant OpenSearch la version 2.3 et que vous avez besoin d'un contrôle d'accès précis pour les actions PIT, vous devez ajouter ces actions et rôles manuellement.
- Il n'y a aucune résilience pour le PIT. Le redémarrage du nœud, la fermeture du nœud, les déploiements bleu/vert et OpenSearch le redémarrage des processus entraînent la perte de toutes les données PIT.
- Si une partition est déplacée lors d'un déploiement bleu/vert, seuls les segments de données actifs sont transférés vers le nouveau nœud. Les segments de fragments détenus par le PIT (à la fois exclusivement et ceux partagés avec les données dynamiques) restent sur l'ancien nœud.
- Les recherches PIT ne fonctionnent actuellement pas avec la recherche asynchrone.

Créer un PIT

Pour exécuter une requête PIT, envoyez des requêtes HTTP en `_search/point_in_time` utilisant le format suivant :

```
POST opensearch-domain/my-index/_search/point_in_time?keep_alive=time
```

Vous pouvez spécifier les options PIT suivantes :

Options	Description	Valeur par défaut	Obligatoire
`keep_alive`	Durée de conservation du PIT. Chaque fois que vous accédez à un PIT avec une demande de recherche, la durée de vie du PIT est prolongée d'une durée égale au `keep_alive` paramètre . Ce paramètre de requête est obligatoire lorsque vous créez un PIT, mais facultatif dans une demande de recherche.		Oui
`preference`	Chaîne qui indique le nœud ou le fragment utilisé pour effectuer la recherche.	Aléatoire	Non
`routing`	Chaîne qui indique d'acheminer les demandes de recherche vers une partition spécifique.	Le document est `_id`	Non
`expand_wildcards`	Chaîne qui indique le type d'index qui peut correspondre au modèle générique. Prend en charge les valeurs séparées par des virgules. Les valeurs possibles sont les suivantes : • `all`: correspond à n'importe quel index ou flux de données, y compris les index cachés. • `open`: Faites correspondre les index ouverts, non masqués ou les flux de données non masqués. • `closed`: Faites correspondre les index fermés, non masqués ou les flux de données non masqués. • `hidden`: Faites correspondre les index ou les flux de données cachés. Doit être combiné avec ouvert, fermé ou à la fois ouvert et fermé. • `none`: Aucun modèle générique n'est accepté.	`open`	Non

Options	Description	Valeur par défaut	Obligatoire
`allow_partial_pit_creation`	Un booléen qui indique s'il faut créer un PIT avec des défaillances partielles.	`true`	Non

Exemple de réponse

```
{
    "pit_id":
 "o463QQEPbXktaW5kZXgtMDAwMDAxFnNOWU43ckt3U3IyaFVpbGE1UWEtMncAFjFyeXBsRGJmVFM2RTB6eVg1aVVqQncAA
    "_shards": {
        "total": 1,
        "successful": 1,
        "skipped": 0,
        "failed": 0
    },
    "creation_time": 1658146050064
}
```

Lorsque vous créez un PIT, vous recevez un identifiant PIT dans la réponse. Il s'agit de l'identifiant que vous utilisez pour effectuer des recherches avec le PIT.

Permissions ponctuelles

PIT prend en charge le contrôle d'accès affiné. Si vous effectuez une mise à niveau vers un domaine en OpenSearch version 2.5 et que vous avez besoin d'un contrôle d'accès précis, vous devez créer manuellement des rôles dotés des autorisations suivantes :

```
# Allows users to use all point in time search search functionality
point_in_time_full_access:
  reserved: true
  index_permissions:
    - index_patterns:
        - '*'
      allowed_actions:
        - "indices:data/read/point_in_time/create"
        - "indices:data/read/point_in_time/delete"
        - "indices:data/read/point_in_time/readall"
```

```
        - "indices:data/read/search"
        - "indices:monitor/point_in_time/segments"


# Allows users to use point in time search search functionality for specific index
# All type operations like list all PITs, delete all PITs are not supported in this
 case

point_in_time_index_access:
  reserved: true
  index_permissions:
    - index_patterns:
        - 'my-index-1'
      allowed_actions:
        - "indices:data/read/point_in_time/create"
        - "indices:data/read/point_in_time/delete"
        - "indices:data/read/search"
        - "indices:monitor/point_in_time/segments"
```

Pour les domaines dotés de OpenSearch la version 2.5 ou supérieure, vous pouvez utiliser le `point_in_time_full_access` rôle intégré. Pour plus d'informations, consultez la section Modèle de sécurité dans la OpenSearch documentation.

Réglages PIT

OpenSearch vous permet de modifier tous les paramètres PIT disponibles à l'aide de l'`_cluster/settings`API. Dans OpenSearch Service, vous ne pouvez actuellement pas modifier les paramètres.

Recherche croisée entre clusters

Vous pouvez créer PITs, rechercher avec PIT IDs PITs, répertorier et supprimer des PITs clusters avec les limitations mineures suivantes :

- Vous ne pouvez tout répertorier et tout supprimer PITs que sur le domaine source.
- Vous ne pouvez pas minimiser les allers-retours réseau dans le cadre d'une requête de recherche croisée entre clusters.

Pour de plus amples informations, veuillez consulter the section called “Recherche croisée entre clusters”.

UltraWarm

Les recherches PIT avec UltraWarm index continuent de fonctionner. Pour de plus amples informations, veuillez consulter the section called “UltraWarm rangement”.

> **Note**
>
> Vous pouvez surveiller les statistiques de recherche PIT dans CloudWatch. Pour accéder à une liste complète des métriques , veuillez consulter the section called “Métriques ponctuelles”.

Recherche sémantique dans Amazon Service OpenSearch

À partir de OpenSearch la version 2.9, vous pouvez utiliser la recherche sémantique pour mieux comprendre les requêtes de recherche et améliorer leur pertinence. Vous pouvez utiliser la recherche sémantique de deux manières : avec la recherche neuronale et avec la recherche K-Neighbor (K-nn).

Avec OpenSearch Service, vous pouvez configurer des connecteurs AI pour des services externes Services AWS et pour des services externes. À l'aide de la console, vous pouvez également créer un modèle ML à l'aide d'un AWS CloudFormation modèle. Pour de plus amples informations, veuillez consulter the section called “CloudFormation intégrations de modèles”.

Pour une documentation complète sur la recherche sémantique, y compris un step-by-step guide d'utilisation de la recherche sémantique, voir Recherche sémantique dans la documentation open source. OpenSearch

Recherche par segment simultanée dans Amazon OpenSearch Service

À partir de OpenSearch la version 2.17, la recherche par segments simultanés utilise un nouveau paramètre pour contrôler le comportement de recherche simultanée.

- Les nouveaux domaines créés avec la version 2.17 ont par défaut la recherche de segments simultanés définie en mode auto sur les nœuds de taille 2xl ou supérieure.
- Les domaines existants mis à niveau vers la version 2.17 ont par défaut la recherche par segments simultanés définie sur auto en fonction du type d'instance pour tous les nœuds de taille 2xl ou

supérieure, et si l'utilisation globale du processeur du cluster est inférieure à 45 % au cours de la dernière semaine.

- Pour plus d'informations, voir Recherche par segment simultanée version 2.17.

À partir de OpenSearch la version 2.13, vous pouvez utiliser la recherche par segment simultanée pour vous aider à rechercher des segments en parallèle pendant la phase de requête. Pour une documentation complète sur la recherche par segments simultanés, voir Recherche par segment simultanée dans la OpenSearch documentation open source. Pour plus d'informations sur CloudWatch les métriques Amazon relatives à la recherche de segments simultanés, consultez la section Mesures et UltraWarm métriques relatives aux instances.

Quelques restrictions supplémentaires s'appliquent lorsque vous utilisez la recherche par segment actuelle avec Amazon OpenSearch Service :

- Vous ne pouvez pas activer la recherche par segment simultanée au niveau de l'index dans OpenSearch Service.
- Par défaut, OpenSearch Service utilise un nombre de 2 tranches avec le mécanisme du nombre maximal de tranches.

Génération de requêtes en langage naturel dans Amazon OpenSearch Service

La fonctionnalité de génération de requêtes en langage naturel d'Amazon OpenSearch Service vous permet d'interroger les données de vos journaux opérationnels et de sécurité par le biais du langage naturel. OpenSearch est une option idéale pour explorer les données des journaux, car il s'agit d'un moteur d'analyse et de recherche de journaux hautement évolutif et performant. Vous pouvez désormais utiliser le langage naturel pour explorer ces journaux. Cette fonctionnalité vous permet d'identifier les problèmes sans recourir au langage PPL (OpenSearch Piped Processing Language) ou sans avoir à rechercher des définitions de données lorsque vous créez vos requêtes. Vous pouvez utiliser la fonctionnalité de génération de requêtes en langage naturel sur les domaines OpenSearch de service avec la version 2.13 et versions ultérieures. Le contrôle précis des accès doit être activé.

Cette fonctionnalité a été développée à l'aide de l'OpenSearch Assistant Toolkit. Si vous souhaitez créer des fonctionnalités similaires qui se connectent à vos grands modèles linguistiques, vous pouvez utiliser la boîte à outils pour configurer vos propres agents et outils.

Prérequis

Avant de pouvoir utiliser la fonctionnalité de génération de requêtes en langage naturel, votre domaine doit disposer des éléments suivants :

- Version 2.13 ou ultérieure.
- Logiciel de service R20240520-P4 ou supérieur.
- Contrôle précis des accès activé. Pour de plus amples informations, veuillez consulter the section called “Activation du contrôle précis des accès”.

Premiers pas

La génération de requêtes en langage naturel est activée par défaut sur tous les domaines créés avec la version 2.13 ou ultérieure pour lesquels le contrôle d'accès détaillé est activé.

Pour les autres domaines, activez-le en sélectionnant Activer la génération de requêtes en langage naturel et les fonctionnalités Amazon Q Developer.

Une fois que vous l'avez activée, accédez à la page Logs dans les OpenSearch tableaux de bord. Choisissez Event Explorer et posez une question à l'aide de l'assistant de requête.

Configurer des autorisations

Si vous activez la génération de requêtes en langage naturel sur un domaine OpenSearch Service préexistant, le rôle query_assistant_access peut ne pas y être défini. Les utilisateurs non-administrateurs doivent être mappés à ce rôle pour gérer les index à chaud des domaines utilisant le contrôle précis des accès. Pour créer manuellement le rôle query_assistant_access, procédez comme suit :

1. Dans les OpenSearch tableaux de bord, accédez à Sécurité, puis sélectionnez Rôles.
2. Choisissez Créer un rôle et configurez les autorisations de cluster suivantes :
   - `cluster:admin/opensearch/ml/config/get`
   - `cluster:admin/opensearch/ml/execute`
   - `cluster:admin/opensearch/ml/predict`
   - `cluster:admin/opensearch/ppl`
3. Nommez le rôle query_assistant_access.

4. Choisissez Créer un rôle. Le rôle query_assistant_access est désormais disponible.

> **Note**
>
> Vous devez également disposer des autorisations `indices:admin/mappings/get` et d'indexation pour les `read` index avec lesquels vous souhaitez utiliser des questions en langage naturel.

Automatisation des configurations

Flow Framework est un OpenSearch plugin qui fournit un moyen d'automatiser les OpenSearch configurations pour des cas d'utilisation tels que la génération de requêtes et le chat conversationnel. Étant donné que le plugin suit les ressources qui activent la fonctionnalité de génération de requêtes en langage naturel, l'index du framework de flux stocke un modèle pour chaque domaine qui utilise l'assistance aux requêtes.

Flow Framework vous permet de sélectionner parmi un ensemble de modèles prédéfinis ou de créer vos propres automatisations pour les connecteurs, outils, agents et autres composants d'apprentissage automatique destinés à servir de OpenSearch backend aux modèles génératifs.

Utilisation de OpenSearch tableaux de bord avec Amazon Service OpenSearch

OpenSearch Dashboards est un outil de visualisation open source conçu pour fonctionner avec. OpenSearch Amazon OpenSearch Service fournit une installation de tableaux de bord pour chaque domaine OpenSearch de service. Les tableaux de bord s'exécutent sur les nœuds de données actifs du domaine.

OpenSearch Les tableaux de bord sont un outil de visualisation permettant d'explorer et d'analyser les données au sein d'un même OpenSearch domaine. En revanche, l'interface OpenSearch utilisateur centralisée (également appelée OpenSearch application) est une interface utilisateur basée sur le cloud qui se connecte à plusieurs OpenSearch domaines, collections OpenSearch sans serveur et sources de AWS données. Il inclut des espaces de travail pour des cas d'utilisation spécifiques tels que l'observabilité et l'analyse de sécurité, et fournit une expérience unifiée entre les ensembles de données. Alors que les tableaux de bord sont liés à des domaines individuels, l'interface utilisateur centralisée permet l'intégration et l'analyse des données entre domaines. Pour de plus amples informations, veuillez consulter OpenSearch UI.

Vous trouverez un lien vers les OpenSearch tableaux de bord sur le tableau de bord de votre domaine dans la console OpenSearch de service. Pour les domaines en cours d'exécution OpenSearch, l'URL est*domain-endpoint*`/_dashboards/`. Pour les domaines exécutant l'ancienne version d'Elasticsearch, l'URL est. *domain-endpoint*`/_plugin/kibana`

Les requêtes utilisant cette installation par défaut de Dashboards ont un délai d'expiration de 300 secondes.

> **Note**
>
> Cette documentation décrit OpenSearch les tableaux de bord dans le contexte d'Amazon OpenSearch Service, y compris les différentes manières de s'y connecter. Pour une documentation complète, y compris un guide de démarrage, des instructions pour créer un tableau de bord, la gestion des tableaux de bord et le langage de requête des tableaux de bord (DQL), voir OpenSearch Tableaux de bord dans la documentation open source. OpenSearch

Contrôle de l'accès aux tableaux de bord

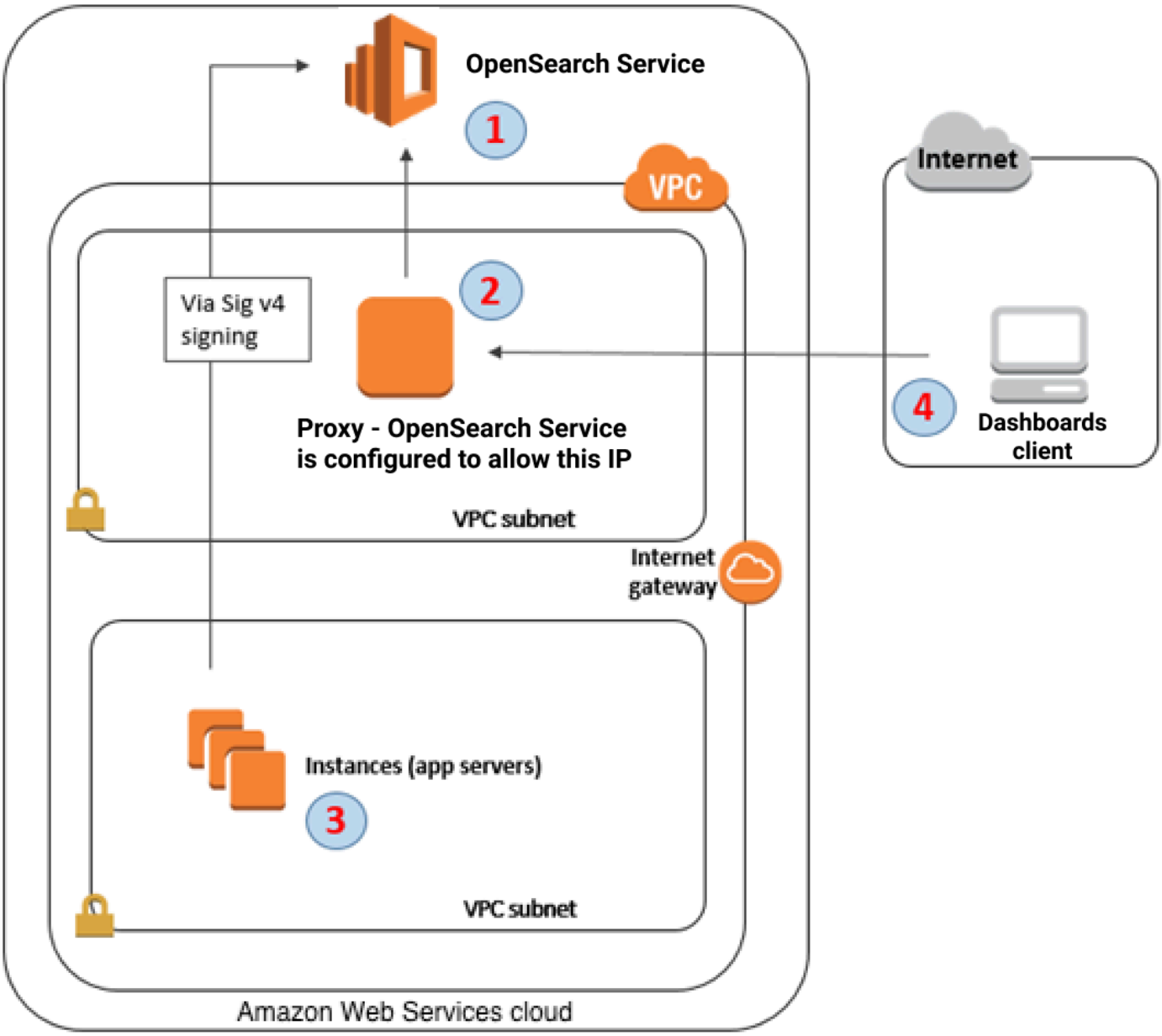
Les tableaux de bord ne prennent pas en charge nativement les utilisateurs et les rôles IAM, mais OpenSearch Service propose plusieurs solutions pour contrôler l'accès aux tableaux de bord :

- Activez l'authentification SAML pour Dashboards.
- Utilisez le contrôle précis des accès avec l'authentification HTTP de base.
- Configurez l'authentification Cognito pour Dashboards.
- Pour les domaines d'accès public, configurez une stratégie d'accès basée sur l'adresse IP utilisant ou non un serveur proxy.
- Pour les domaines d'accès VPC, utilisez une stratégie d'accès ouverte utilisant ou non un serveur proxy, ainsi que des groupes de sécurité pour contrôler l'accès. Pour en savoir plus, veuillez consulter la section the section called “À propos des stratégies d'accès pour les domaines de VPC”.

Utilisation d'un proxy pour accéder au OpenSearch service à partir de tableaux de bord

> **Note**
>
> Ce processus s'applique uniquement si votre domaine utilise l'accès public et que vous ne voulez pas utiliser l'authentification Cognito. Consultez the section called “Contrôle de l'accès aux tableaux de bord ”.

Dashboards étant une JavaScript application, les demandes proviennent de l'adresse IP de l'utilisateur. Le contrôle d'accès basé sur l'adresse IP peut ne pas convenir en raison du grand nombre d'adresses IP que vous devriez autoriser afin que chaque utilisateur ait accès à Dashboards. Une solution consiste à placer un serveur proxy entre les tableaux de bord et OpenSearch le service. Ensuite, vous pouvez ajouter une stratégie d'accès basée sur IP qui n'autorise les demandes qu'à partir d'une seule adresse IP, celle du proxy. Le schéma suivant illustre cette configuration.



1. Il s'agit de votre domaine de OpenSearch service. IAM fournit un accès autorisé à ce domaine. Une stratégie d'accès IP supplémentaire fournit l'accès au serveur proxy.
2. Il s'agit du serveur proxy qui s'exécute sur une EC2 instance Amazon.
3. D'autres applications peuvent utiliser le processus de signature Signature Version 4 pour envoyer des demandes authentifiées au OpenSearch Service.
4. Les clients des tableaux de bord se connectent à votre domaine de OpenSearch service via le proxy.

Pour activer ce type de configuration, il vous faut une politique basée sur les ressources, qui spécifie les rôles et les adresses IP. Voici un exemple de politique :

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Resource":"arn:aws:es:us-west-2:111111111111:domain/my-domain/*",
         "Principal":{
            "AWS":"arn:aws:iam::111111111111:role/allowedrole1"
         },
         "Action":[
            "es:ESHttpGet"
         ],
         "Effect":"Allow"
      },
      {
         "Effect":"Allow",
         "Principal":{
            "AWS":"*"
         },
         "Action":"es:*",
         "Condition":{
            "IpAddress":{
               "aws:SourceIp":[
                  "203.0.113.0/24",
                  "2001:DB8:1234:5678::/64"
               ]
            }
         },
         "Resource":"arn:aws:es:us-west-2:111111111111:domain/my-domain/*"
      }
   ]
}
```

Nous vous recommandons de configurer l' EC2 instance exécutant le serveur proxy avec une adresse IP élastique. De cette manière, vous pouvez remplacer l'instance si nécessaire et continuer de lui attacher la même adresse IP publique. Pour en savoir plus, consultez la section Adresses IP élastiques dans le guide de EC2 l'utilisateur Amazon.

Si vous utilisez un serveur proxy et l'authentification Cognito, il peut être nécessaire d'ajouter des paramètres pour Dashboards et Amazon Cognito afin d'éviter les erreurs `redirect_mismatch`. Consultez l'exemple `nginx.conf` suivant :

```
server {
    listen 443;
    server_name $host;
    rewrite ^/$ https://$host/_plugin/_dashboards redirect;

    ssl_certificate          /etc/nginx/cert.crt;
    ssl_certificate_key      /etc/nginx/cert.key;

    ssl on;
    ssl_session_cache  builtin:1000  shared:SSL:10m;
    ssl_protocols  TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers HIGH:!aNULL:!eNULL:!EXPORT:!CAMELLIA:!DES:!MD5:!PSK:!RC4;
    ssl_prefer_server_ciphers on;

    location /_plugin/_dashboards {
        # Forward requests to Dashboards
        proxy_pass https://$dashboards_host/_plugin/_dashboards;

        # Handle redirects to Cognito
        proxy_redirect https://$cognito_host https://$host;

        # Update cookie domain and path
        proxy_cookie_domain $dashboards_host $host;
        proxy_cookie_path / /_plugin/_dashboards/;

        # Response buffer settings
        proxy_buffer_size 128k;
        proxy_buffers 4 256k;
        proxy_busy_buffers_size 256k;
    }

    location ~ \/(log|sign|fav|forgot|change|saml|oauth2) {
        # Forward requests to Cognito
        proxy_pass https://$cognito_host;

        # Handle redirects to Dashboards
        proxy_redirect https://$dashboards_host https://$host;

        # Update cookie domain
```

```
        proxy_cookie_domain $cognito_host $host;
    }
}
```

> **Note**
>
> (Facultatif) Si vous choisissez de fournir un nœud de coordination dédié, il commencera automatiquement à héberger OpenSearch des tableaux de bord. Par conséquent, la disponibilité des ressources des nœuds de données, telles que le processeur et la mémoire, est accrue. Cette disponibilité accrue des ressources des nœuds de données peut contribuer à améliorer la résilience globale de votre domaine.

Configuration des tableaux de bord pour utiliser un serveur de carte WMS

L'installation par défaut de Dashboards for OpenSearch Service inclut un service de carte, à l'exception des domaines des régions de l'Inde et de la Chine. Le service de carte prend en charge jusqu'à 10 niveaux de zoom.

Quelle que soit votre région, vous pouvez configurer Dashboards pour qu'il utilise un autre serveur WMS (Web Map Service) pour les visualisations cartographiques de coordonnées. Les visualisations de carte de région prennent uniquement en charge le service de carte par défaut.

Pour configurer Dashboards afin d'utiliser un serveur de cartes WMS :

1. Ouvrir Dashboards
2. Choisissez Stack Management (Gestion des piles).
3. Choisissez Advanced Settings (Paramètres avancés).
4. Localiser la visualisation : TileMap :. WMSdefaults
5. Remplacez `enabled` par `true` et `url` par l'URL d'un serveur de cartes WMS valide.

```
{
  "enabled": true,
  "url": "wms-server-url",
  "options": {
    "format": "image/png",
    "transparent": true
```

```
  }
}
```

6. Sélectionnez Enregistrer les modifications.

Pour appliquer la nouvelle valeur par défaut aux visualisations, vous serez peut-être amené à recharger Dashboards. Si vous avez enregistré des visualisations, choisissez Options après l'ouverture de la visualisation. Vérifiez que le serveur de carte WMS est activé et que l'URL WMS contient votre serveur de carte préféré, puis choisissez Apply changes (Appliquer les modifications).

> **Note**
>
> Les services de cartes font souvent l'objet de frais ou de restrictions de licence. Vous devez tenir compte de tous ces points sur tous les serveurs de cartes que vous spécifiez. Les services de cartes disponibles dans l'U.S. Geological Survey peuvent être utiles à des fins de test.

Connexion d'un serveur de tableaux de bord local au service OpenSearch

Si vous avez déjà investi beaucoup de temps dans la configuration de votre propre instance de tableaux de bord, vous pouvez l'utiliser à la place (ou en complément) de l'instance de tableaux de bord par défaut fournie par OpenSearch Service. La procédure suivante s'applique aux domaines qui utilisent le contrôle précis des accès avec une stratégie d'accès ouverte.

Pour connecter un serveur de tableaux de bord local au service OpenSearch

1. Sur votre domaine OpenSearch de service, créez un utilisateur doté des autorisations appropriées :

   a. Dans Dashboards, accédez à Security (Sécurité), Internal users (Utilisateurs internes), puis choisissez Create internal user (Créer un utilisateur interne).

   b. Entrez un nom d'utilisateur et un mot de passe, puis choisissez Create (Créer).

   c. Accédez à Roles (Rôles) et choisissez un rôle.

   d. Sélectionnez Mapped users (Utilisateurs mappés) et choisissez Manage mapping (Gérer le mappage).

e. Dans le champ Users (Utilisateurs), ajoutez votre nom d'utilisateur et choisissez Map (Mapper).

2. Téléchargez et installez la version appropriée du plugin de OpenSearch sécurité sur votre installation autogérée de Dashboards OSS.

3. Sur votre serveur Dashboards local, ouvrez le `config/opensearch_dashboards.yml` fichier et ajoutez votre point de terminaison de OpenSearch service avec le nom d'utilisateur et le mot de passe que vous avez créés précédemment :

```
opensearch.hosts: ['https://domain-endpoint']
opensearch.username: 'username'
opensearch.password: 'password'
```

Vous pouvez utiliser l'exemple de fichier `opensearch_dashboards.yml` suivant :

```
server.host: '0.0.0.0'

opensearch.hosts: ['https://domain-endpoint']

opensearchDashboards.index: ".username"

opensearch.ssl.verificationMode: none # if not using HTTPS

opensearch_security.auth.type: basicauth
opensearch_security.auth.anonymous_auth_enabled: false
opensearch_security.cookie.secure: false # set to true when using HTTPS
opensearch_security.cookie.ttl: 3600000
opensearch_security.session.ttl: 3600000
opensearch_security.session.keepalive: false
opensearch_security.multitenancy.enabled: false
opensearch_security.readonly_mode.roles: ['opensearch_dashboards_read_only']
opensearch_security.auth.unauthenticated_routes: []
opensearch_security.basicauth.login.title: 'Please log in using your username and
 password'

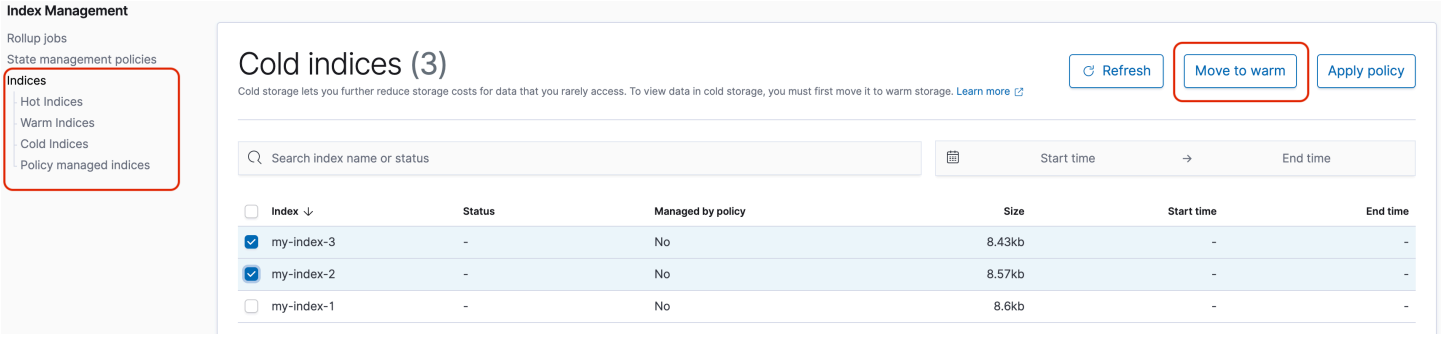
opensearch.username: 'username'
opensearch.password: 'password'
opensearch.requestHeadersWhitelist: [authorization, securitytenant,
 security_tenant]
```

Pour voir les index OpenSearch de vos services, démarrez votre serveur de tableaux de bord local, accédez à Dev Tools et exécutez la commande suivante :

```
GET _cat/indices
```

Gestion des index dans les tableaux de bord

L'installation des tableaux de bord sur votre domaine de OpenSearch service fournit une interface utilisateur utile pour gérer les index des différents niveaux de stockage de votre domaine. Choisissez Index Management dans le menu principal des tableaux de bord pour afficher tous les index stockés à chaud ou UltraWarmà froid, ainsi que les index gérés par les politiques ISM (Index State Management). Utilisez la gestion des index pour déplacer les index entre les niveaux de stockage tiède et à froid et pour surveiller les migrations entre les trois niveaux.



Notez que vous ne verrez pas les options d'indice chaud, chaud et froid à moins que vous n'ayez activé UltraWarm et/ou que le stockage à froid soit activé.

Fonctionnalités supplémentaires

L'installation par défaut des tableaux de bord sur chaque domaine OpenSearch de service comporte quelques fonctionnalités supplémentaires :

- Interfaces utilisateur pour les différents OpenSearchplugins
- Locataires
- Rapports

  Utilisez le menu Génération de rapports pour générer des rapports CSV à la demande à partir de la page Découvrir et des rapports PDF ou PNG des tableaux de bord ou des visualisations. Les rapports CSV ont une limite de 10 000 lignes.

- Diagrammes de Gantt
- Blocs-notes

Utilisation de l' OpenSearch interface utilisateur dans Amazon OpenSearch Service

OpenSearch L'interface utilisateur est une expérience d'analyse opérationnelle modernisée pour Amazon OpenSearch Service qui fournit une vue unifiée vous permettant d'interagir avec les données provenant de plusieurs sources. Contrairement aux OpenSearch tableaux de bord, qui ne fonctionnent qu'avec le domaine ou la collection qui les héberge, l' OpenSearch interface utilisateur est hébergée dans le AWS Cloud. Cela permet à l' OpenSearch interface utilisateur d'atteindre une haute disponibilité et de rester fonctionnelle pendant les mises à niveau du cluster, et de se connecter nativement à plusieurs sources de données. Pour plus d'informations sur les OpenSearch tableaux de bord, consultez OpenSearch Tableaux de bord.

Les principales fonctionnalités de l' OpenSearch interface utilisateur sont les suivantes :

- Support de sources de données multiples — L' OpenSearch interface utilisateur peut se connecter à plusieurs sources de données pour créer une vue complète. Cela inclut OpenSearch les domaines et les collections sans serveur, ainsi que les sources de AWS données intégrées telles qu'Amazon CloudWatch, Amazon Security Lake et Amazon Simple Storage Service (Amazon S3).
- Aucun temps d'arrêt lors des mises à niveau : l' OpenSearch interface utilisateur est hébergée dans le AWS Cloud. Cela signifie qu'il OpenSearch reste opérationnel et qu'il peut récupérer les données des clusters pendant les processus de mise à niveau.
- Espaces de travail : espaces conçus pour les collaborations d'équipe pour différents flux de travail, tels que l'observabilité, l'analyse de sécurité et la recherche. Vous pouvez définir les paramètres de confidentialité et gérer les autorisations des collaborateurs de votre espace de travail.
- Authentification unique : l' OpenSearch interface utilisateur fonctionne avec AWS IAM Identity Center la fédération SAML AWS Identity and Access Management (IAM) pour s'intégrer à vos fournisseurs d'identité afin de créer une expérience d'authentification unique pour vos utilisateurs finaux.
- Analyses basées sur OpenSearch Genai : l'interface utilisateur prend en charge la génération de requêtes en langage naturel afin de générer les bonnes requêtes pour votre analyse. OpenSearch L'interface utilisateur travaille également avec Amazon Q Developer pour fournir le chat Amazon Q et aider à générer des visualisations, un résumé des alertes, des informations et des détecteurs d'anomalies recommandés.

- Prise en charge de plusieurs langages de requête : l' OpenSearch interface utilisateur prend en charge le langage de traitement pipé (PPL), SQL, Lucene et le langage de requête DQL (Dashboards Query Language).
- Support entre régions et comptes : l' OpenSearch interface utilisateur peut utiliser la fonction de recherche entre clusters pour se connecter à des OpenSearch domaines appartenant à différents comptes et à différentes régions à des fins d'analyse et de visualisation agrégées.

Pour commencer et créer votre première OpenSearch interface utilisateur, suivez les instructions dethe section called “Premiers pas”.

Pour plus d'informations sur les dernières fonctionnalités publiées pour l' OpenSearch interface utilisateur, consultezthe section called “Historique de versions”.

Rubriques

- Historique des versions OpenSearch de l'interface utilisateur Amazon Service
- Commencer à utiliser l'interface OpenSearch utilisateur d'Amazon OpenSearch Service
- Activation de la fédération SAML avec AWS Identity and Access Management
- Gestion des associations de sources de données et des autorisations d'accès au Virtual Private Cloud
- Utilisation des espaces OpenSearch de travail Amazon Service
- Accès aux données entre régions et entre comptes grâce à la recherche entre clusters
- Gestion de l'accès à l' OpenSearch interface utilisateur depuis un point de terminaison VPC

Historique des versions OpenSearch de l'interface utilisateur Amazon Service

Le tableau suivant répertorie toutes les versions du support Amazon OpenSearch Service pour l' OpenSearch interface utilisateur ainsi que les fonctionnalités et améliorations incluses dans chaque version.

Modification	Date de publication	Description
Nouvelle fonctionnalité	16/04/2025	OpenSearch L'interface utilisateur fonctionne désormais avec la recherche entre clusters. Cela vous permet d'utiliser l' OpenSearch interface utilisateur d'une seule Région AWS pour accéder aux clusters d'une autre région. Cela se fait en le configurant en tant que cluster distant connecté à un cluster de la même région. Pour de plus amples informations, veuillez consulter the section called “Accès aux données entre régions et entre comptes grâce à la recherche entre clusters”.
Nouvelle fonctionnalité	31/03/2025	Amazon Q Developer est désormais généralement disponible sur Amazon OpenSearch Service. Pour plus d'informations, consultez Prise en charge d’Amazon Q.
Nouvelle fonctionnalité	05/02/2025	La fédération (Security Assertion Markup Language 2.0) (SAML) via AWS Identity and Access Management (IAM) fonctionne désormais avec l'interface utilisateur. OpenSearch Cela permet de créer une expérience d'authentification unique (SSO) initiée par le fournisseur d'identité pour vos utilisateurs finaux. Pour de plus amples informations, veuillez consulter the section called “Activation de la fédération SAML avec IAM”.
Nouvelle intégration	2024-12-01	L'intégration zéro ETL avec Amazon CloudWatch simplifie l'analyse et la visualisation des données de journal, réduisant ainsi les frais techniques et les coûts opérationnels. Pour plus d'informations, consultez New Amazon CloudWatch et Amazon OpenSearch Service lancent une expérience d'analyse intégrée sur le AWS News Blog.
Nouvelle intégration	2024-12-01	L'intégration Zero-ETL avec Amazon Security Lake permet aux entreprises de rechercher, d'analyser et d'obtenir des informations exploitables de manière efficace à partir de leurs données de sécurité. Pour plus d'informations, consultez la section Présentation de l'intégration d'Amazon OpenSearc

Modification	Date de publication	Description
		h Service et d'Amazon Security Lake pour simplifier les analyses de sécurité sur le blog d'AWS actualités.
Première version	07/11/2024	La version publique initiale de OpenSearch UI. Pour plus d'informations, consultez Amazon OpenSearch Service lance l' OpenSearch interface utilisateur de nouvelle génération sur le AWS Big Data Blog.

Commencer à utiliser l'interface OpenSearch utilisateur d'Amazon OpenSearch Service

Dans Amazon OpenSearch Service, une application est une instance de l'interface OpenSearch utilisateur (OpenSearch UI). Chaque application peut être associée à plusieurs sources de données, et une seule source peut être associée à plusieurs applications. Vous pouvez créer plusieurs applications pour différents administrateurs à l'aide des différentes options d'authentification prises en charge.

Utilisez les informations de cette rubrique pour vous guider tout au long du processus de création d'une application d' OpenSearch interface utilisateur à l'aide du AWS Management Console ou du AWS CLI.

Rubriques

- Autorisations requises pour créer des applications Amazon OpenSearch Service
- Création d'une application d' OpenSearch interface utilisateur
- Gestion des administrateurs d'applications

Autorisations requises pour créer des applications Amazon OpenSearch Service

Avant de créer une application, vérifiez que vous disposez des autorisations nécessaires pour la tâche. Contactez un administrateur de compte pour obtenir de l'aide si nécessaire.

Autorisations générales

Pour utiliser des applications dans OpenSearch Service, vous devez disposer des autorisations indiquées dans la politique suivante. Les autorisations répondent aux objectifs suivants :

- Les cinq `es:*Application` autorisations sont requises pour créer et gérer une application.
- Les trois `es:*Tags` autorisations sont requises pour ajouter, répertorier et supprimer des balises dans l'application.
- Les `es:GetDirectQueryDataSource` autorisations`aoss:BatchGetCollection`, `es:DescribeDomain` et sont requises pour associer des sources de données.
- Les `opensearch:*DirectQuery*` autorisations `aoss:APIAccessAlles:ESHttp*`,, et 4 sont requises pour accéder aux sources de données.
- Permet `iam:CreateServiceLinkedRole` à Amazon OpenSearch Service de créer un rôle lié à un service (SLR) dans votre compte. Ce rôle est utilisé et permet à l'application d' OpenSearch interface utilisateur de publier CloudWatch les métriques Amazon sur votre compte. Pour plus d'informations, consultez the section called “Autorisations” dans la rubrique Utilisation des rôles liés à un service pour créer des domaines VPC et interroger directement les sources de données.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "es:CreateApplication",
                "es:DeleteApplication",
                "es:GetApplication",
                "es:ListApplications",
                "es:UpdateApplication",
                "es:AddTags",
                "es:ListTags",
                "es:RemoveTags",
                "aoss:APIAccessAll",
                "es:ESHttp*",
                "opensearch:StartDirectQuery",
```

```
                "opensearch:GetDirectQuery",
                "opensearch:CancelDirectQuery",
                "opensearch:GetDirectQueryResult",
                "aoss:BatchGetCollection",
                "aoss:ListCollections",
                "es:DescribeDomain",
                "es:DescribeDomains",
                "es:ListDomainNames",
                "es:GetDirectQueryDataSource",
                "es:ListDirectQueryDataSources"
            ],
            "Resource": "*"
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/
opensearchservice.amazonaws.com/AWSServiceRoleForAmazonOpenSearchService"
        }
    ]
}
```

Autorisations pour créer une application qui utilise l'authentification IAM Identity Center (facultatif)

Par défaut, les applications de tableau de bord sont authentifiées à l'aide de AWS Identity and Access Management (IAM) pour gérer les autorisations des utilisateurs AWS des ressources. Toutefois, vous pouvez choisir de proposer une expérience d'authentification unique en utilisant IAM Identity Center, qui vous permet d'utiliser vos fournisseurs d'identité existants pour vous connecter aux applications d' OpenSearch interface utilisateur. Dans ce cas, vous allez sélectionner l'option Authentification avec IAM Identity Center dans la procédure décrite plus loin dans cette rubrique, puis accorder aux utilisateurs d'IAM Identity Center les autorisations nécessaires pour accéder à l'application d' OpenSearch interface utilisateur.)

Pour créer une application qui utilise l'authentification IAM Identity Center, vous devez disposer des autorisations suivantes. Remplacez *`placeholder values`* par vos propres informations. Contactez un administrateur de compte pour obtenir de l'aide si nécessaire.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "IDC_Permissions",
            "Effect": "Allow",
            "Action": [
                "es:CreateApplication",
                "es:DeleteApplication",
                "es:GetApplication",
                "es:ListApplications",
                "es:UpdateApplication",
                "es:AddTags",
                "es:ListTags",
                "es:RemoveTags",
                "aoss:BatchGetCollection",
                "aoss:ListCollections",
                "es:DescribeDomain",
                "es:DescribeDomains",
                "es:ListDomainNames",
                "es:GetDirectQueryDataSource",
                "es:ListDirectQueryDataSources",
                "sso:CreateApplication",
                "sso:DeleteApplication",
                "sso:PutApplicationGrant",
                "sso:PutApplicationAccessScope",
                "sso:PutApplicationAuthenticationMethod",
                "sso:ListInstances",
                "sso:DescribeApplicationAssignment",
                "sso:DescribeApplication",
                "sso:CreateApplicationAssignment",
                "sso:ListApplicationAssignments",
                "sso:DeleteApplicationAssignment",
                "sso-directory:SearchGroups",
                "sso-directory:SearchUsers",
                "sso:ListDirectoryAssociations",
                "identitystore:DescribeUser",
                "identitystore:DescribeGroup",
                "iam:ListRoles"
            ],
            "Resource": "*"
```

```
        },
        {
            "Sid": "SLR_Permission",
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/
opensearchservice.amazonaws.com/AWSServiceRoleForAmazonOpenSearchService"
        },
        {
            "Sid": "PassRole_Permission",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::account-id}:role/iam-role-for-identity-
center"
        }
    ]
}
```

Création d'une application d' OpenSearch interface utilisateur

Créez une application qui spécifie le nom de l'application, la méthode d'authentification et les administrateurs à l'aide de l'une des procédures suivantes.

Rubriques

- Création d'une application d' OpenSearch interface utilisateur utilisant l'authentification IAM dans la console
- Création d'une application d' OpenSearch interface utilisateur utilisant AWS IAM Identity Center l'authentification dans la console
- Création d'une application d' OpenSearch interface utilisateur utilisant AWS IAM Identity Center l'authentification à l'aide du AWS CLI

Création d'une application d' OpenSearch interface utilisateur utilisant l'authentification IAM dans la console

Pour créer une application d' OpenSearch interface utilisateur qui utilise l'authentification IAM dans la console

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, choisissez OpenSearch UI (Dashboards).
3. Choisissez Créer une application.
4. Dans Nom de l'application, entrez le nom de l'application.
5. Ne cochez pas la case Authentification avec IAM Identity Center. Pour plus d'informations sur la création d'une application utilisant l'authentification AWS IAM Identity Center, reportez-vous à the section called “Création d'une application d' OpenSearch interface utilisateur utilisant AWS IAM Identity Center l'authentification dans la console” la section suivante de cette rubrique.
6. (Facultatif) Vous êtes automatiquement ajouté en tant qu'administrateur de l'application que vous créez. Dans la zone de gestion des administrateurs de l'OpenSearch application, vous pouvez accorder des autorisations d'administrateur à d'autres utilisateurs.

   > **Note**
   >
   > Le rôle d'administrateur de l'application d' OpenSearch interface utilisateur autorise la modification et la suppression d'une application d' OpenSearch interface utilisateur. Les administrateurs d'applications peuvent également créer, modifier et supprimer des espaces de travail dans une application d' OpenSearch interface utilisateur.

   Pour accorder des autorisations d'administrateur à d'autres utilisateurs, choisissez l'une des options suivantes :

   - Accorder l'autorisation de l'administrateur à un ou plusieurs utilisateurs spécifiques : dans le champ Administrateurs de l'OpenSearchapplication, dans la liste contextuelle Propriétés, sélectionnez Utilisateurs IAM ou

     AWS IAM Identity Center utilisateurs, puis choisissez les utilisateurs individuels auxquels accorder des autorisations d'administrateur.

- Accorder des autorisations d'administrateur à tous les utilisateurs : tous les utilisateurs de votre organisation ou de votre compte reçoivent des autorisations d'administrateur.

7. (Facultatif) Dans la zone Balises, appliquez une ou plusieurs name/value paires de clés de balise à l'application.

   Les balises sont des métadonnées facultatives que vous affectez à une ressource. Les balises vous permettent de classer une ressource de différentes façons, par exemple, par objectif, par propriétaire ou par environnement.

8. Choisissez Créer.

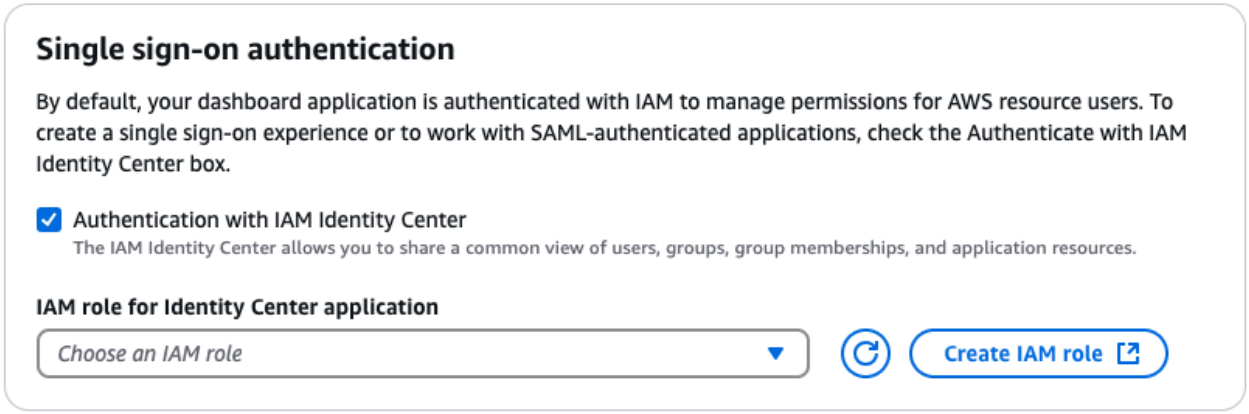
Création d'une application d' OpenSearch interface utilisateur utilisant AWS IAM Identity Center l'authentification dans la console

Pour créer une application d' OpenSearch interface utilisateur qui utilise AWS IAM Identity Center l'authentification, vous devez disposer des autorisations IAM décrites plus haut dans cette rubrique. the section called “Autorisations pour créer une application qui utilise l'authentification IAM Identity Center (facultatif)”

Pour créer une application d' OpenSearch interface utilisateur qui utilise AWS IAM Identity Center l'authentification dans la console

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, choisissez OpenSearch UI (Dashboards).
3. Choisissez Créer une application.
4. Dans Nom de l'application, entrez le nom de l'application.
5. (Facultatif) Pour activer l'authentification unique pour votre organisation ou votre compte, procédez comme suit :

   a. Cochez la case Authentification avec IAM Identity Center, comme illustré dans l'image suivante :



b. Effectuez l'une des actions suivantes :

- Dans la liste des applications Rôle IAM pour Identity Center, choisissez un rôle IAM existant qui fournit les autorisations requises pour permettre à IAM Identity Center d'accéder à l' OpenSearch interface utilisateur et aux sources de données associées. Consultez les politiques décrites dans le point suivant pour connaître les autorisations dont le rôle doit disposer.
- Créez un nouveau rôle avec les autorisations requises. Utilisez les procédures suivantes du guide de l'utilisateur IAM avec les options spécifiées pour créer un nouveau rôle et avec la politique d'autorisation et la politique de confiance nécessaires.
  - Procédure : création de politiques IAM (console)

    Au fur et à mesure que vous suivez les étapes de cette procédure, collez la politique suivante dans le champ JSON de l'éditeur de politiques :

    JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "IdentityStoreOpenSearchDomainConnectivity",
            "Effect": "Allow",
            "Action": [
                "identitystore:DescribeUser",
                "identitystore:ListGroupMembershipsForMember",
                "identitystore:DescribeGroup"
            ],
            "Resource": "*",
            "Condition": {
```

```
                "ForAnyValue:StringEquals": {
                    "aws:CalledViaLast": "es.amazonaws.com"
                }
            }
        },
        {
            "Sid": "OpenSearchDomain",
            "Effect": "Allow",
            "Action": [
                "es:ESHttp*"
            ],
            "Resource": "*"
        },
        {
            "Sid": "OpenSearchServerless",
            "Effect": "Allow",
            "Action": [
                "aoss:APIAccessAll"
            ],
            "Resource": "*"
        }
    ]
}
```

- Procédure : créer un rôle à l'aide de politiques de confiance personnalisées

  Au fur et à mesure que vous suivez les étapes de cette procédure, remplacez l'espace réservé JSON dans la zone Politique de confiance personnalisée par ce qui suit :

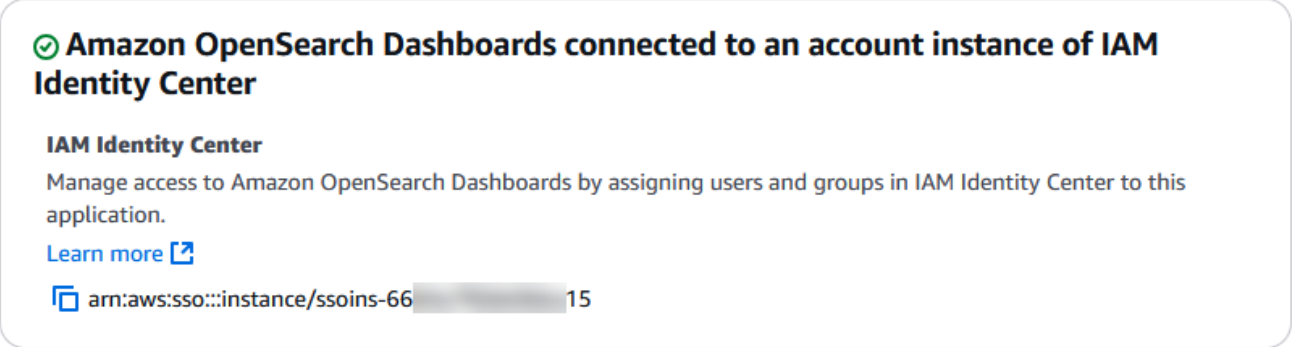
  > **Tip**
  >
  > Si vous ajoutez la politique de confiance à un rôle existant, ajoutez-la dans l'onglet Relation de confiance du rôle.

  JSON

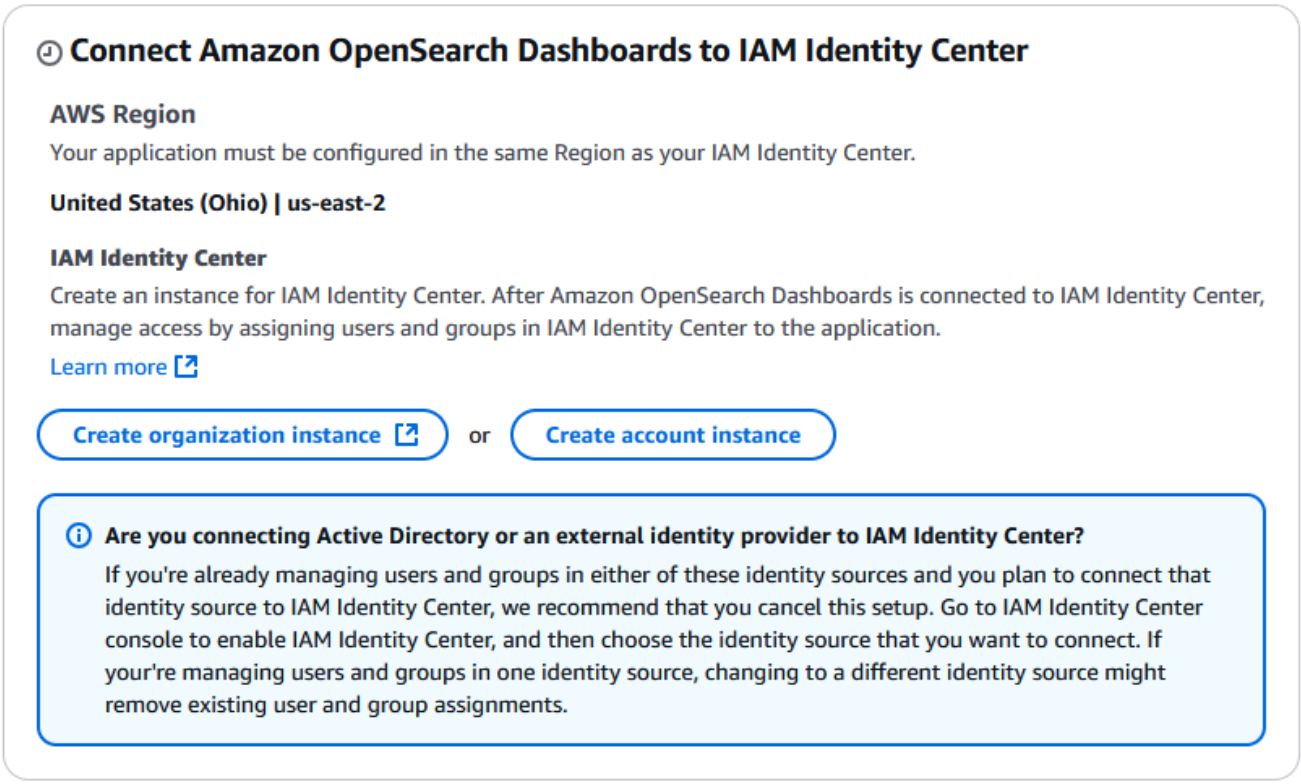
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
```

```
            "Principal": {
                "Service":
 "application.opensearchservice.amazonaws.com"
            },
            "Action": [
                "sts:AssumeRole",
                "sts:SetContext"
            ],
            "Condition": {
                "ForAllValues:ArnEquals": {
                    "sts:RequestContextProviders":
 "arn:aws:iam::aws:contextProvider/IdentityCenter"
                }
            }
        }
    ]
}
```

c. Si une instance IAM Identity Center a déjà été créée dans votre organisation ou votre compte, la console indique qu'Amazon OpenSearch Dashboards est déjà connecté à une instance d'organisation d'IAM Identity Center, comme illustré dans l'image suivante.



Si IAM Identity Center n'est pas encore disponible dans votre organisation ou votre compte, vous ou un administrateur disposant des autorisations nécessaires pouvez créer une instance d'organisation ou une instance de compte. La zone Connect Amazon OpenSearch Dashboards to IAM Identity Center propose des options pour les deux, comme le montre l'image suivante :

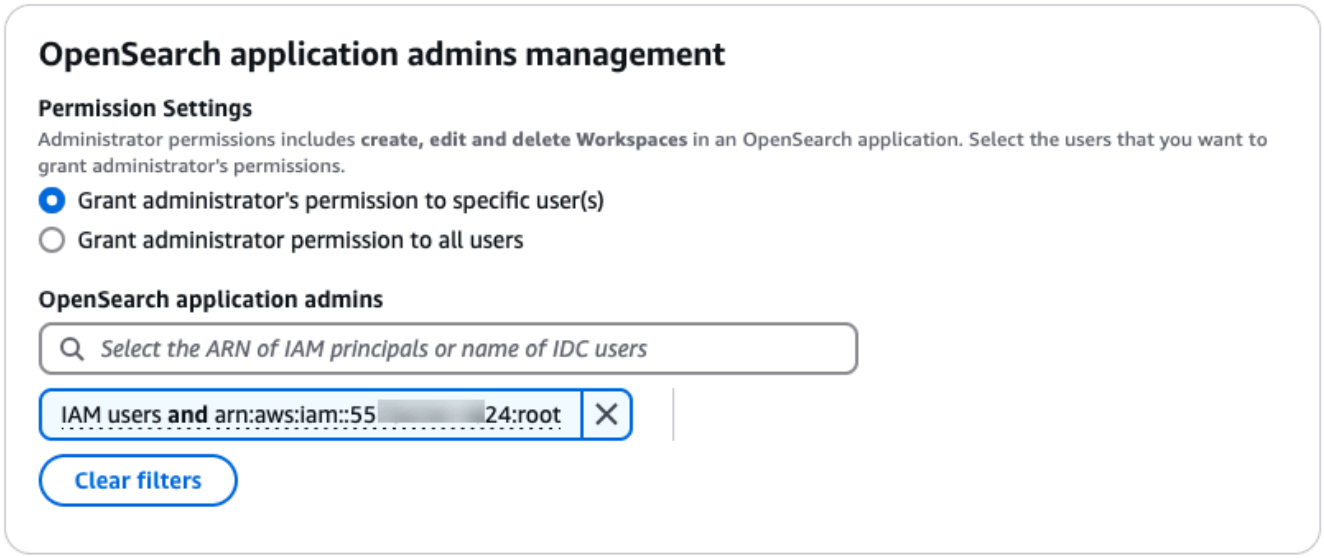


Dans ce cas, vous pouvez créer une instance de compte dans IAM Identity Center à des fins de test, ou demander à un administrateur de créer une instance organisationnelle dans IAM Identity Center. Pour plus d'informations, consultez les rubriques suivantes dans le AWS IAM Identity Center Guide de l'utilisateur :

> **Note**
>
> Actuellement, les applications d' OpenSearch interface utilisateur ne peuvent être créées qu'au même endroit Région AWS que votre instance organisationnelle IAM Identity Center. Pour plus d'informations sur l'accès aux sources de données de cette région après avoir créé l'application, consultezthe section called "Accès aux données entre régions et entre comptes grâce à la recherche entre clusters".

- Instances organisationnelles d'IAM Identity Center
- Instances de compte d'IAM Identity Center
- Activer AWS IAM Identity Center

6. (Facultatif) Vous êtes automatiquement ajouté en tant qu'administrateur de l'application que vous créez. Dans la zone de gestion des administrateurs de l'OpenSearch application, vous pouvez accorder des autorisations d'administrateur à d'autres utilisateurs, comme le montre l'image suivante :



> **Note**
>
> Le rôle d'administrateur de l'application d' OpenSearch interface utilisateur autorise la modification et la suppression d'une application d' OpenSearch interface utilisateur. Les administrateurs d'applications peuvent également créer, modifier et supprimer des espaces de travail dans une application d' OpenSearch interface utilisateur.

Pour accorder des autorisations d'administrateur à d'autres utilisateurs, choisissez l'une des options suivantes :

- Accorder l'autorisation de l'administrateur à un ou plusieurs utilisateurs spécifiques : dans le champ Administrateurs de l'OpenSearchapplication, dans la liste contextuelle Propriétés, sélectionnez Utilisateurs IAM ou

  AWS IAM Identity Center utilisateurs, puis choisissez les utilisateurs individuels auxquels accorder des autorisations d'administrateur.
- Accorder des autorisations d'administrateur à tous les utilisateurs : tous les utilisateurs de votre organisation ou de votre compte reçoivent des autorisations d'administrateur.

7. (Facultatif) Dans la zone Balises, appliquez une ou plusieurs name/value paires de clés de balise à l'application.

   Les balises sont des métadonnées facultatives que vous affectez à une ressource. Les balises vous permettent de classer une ressource de différentes façons, par exemple, par objectif, par propriétaire ou par environnement.
8. Choisissez Créer.

Création d'une application d' OpenSearch interface utilisateur utilisant AWS IAM Identity Center l'authentification à l'aide du AWS CLI

Pour créer une application d' OpenSearch interface utilisateur qui utilise l' AWS IAM Identity Center authentification à l'aide de AWS CLI, utilisez la commande create-application avec les options suivantes :

- `--name`— Le nom de l'application.
- `--iam-identity-center-options`— (Facultatif) L'instance IAM Identity Center et le rôle IAM qui OpenSearch seront utilisés pour l'authentification et le contrôle d'accès.

Remplacez *`placeholder values`* par vos propres informations.

```
aws opensearch create-application \
    --name application-name \
    --iam-identity-center-options "
          {
          \"enabled\":true,
          \"iamIdentityCenterInstanceArn\":\"arn:aws:sso:::instance/sso-instance\",
          \"iamRoleForIdentityCenterApplicationArn\":\"arn:aws:iam::account-
id:role/role-name\"
          }
    "
```

Gestion des administrateurs d'applications

Un administrateur d'application d' OpenSearch interface utilisateur est un rôle défini autorisé à modifier et à supprimer une application d' OpenSearch interface utilisateur.

Par défaut, en tant que créateur d'une application d' OpenSearch interface utilisateur, vous êtes le premier administrateur de l'application d' OpenSearch interface utilisateur.

Gestion des administrateurs de l' OpenSearch interface utilisateur à l'aide de la console

Vous pouvez ajouter des administrateurs supplémentaires à une application d' OpenSearch interface utilisateur dans le AWS Management Console, soit pendant le processus de création de l'application, soit sur la page d'édition une fois l'application créée.

Le rôle d'administrateur de l'application d' OpenSearch interface utilisateur autorise la modification et la suppression d'une application d' OpenSearch interface utilisateur. Les administrateurs d'applications peuvent également créer, modifier et supprimer des espaces de travail dans une application d' OpenSearch interface utilisateur.

Sur la page détaillée d'une application, vous pouvez rechercher le nom de ressource Amazon (ARN) d'un principal IAM ou le nom d'un utilisateur de l'IAM Identity Center.

Pour gérer les administrateurs de l' OpenSearch interface utilisateur à l'aide de la console

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, choisissez OpenSearch UI (Dashboards).
3. Dans la zone OpenSearch des applications, choisissez le nom d'une application existante.
4. Choisissez Modifier.
5. Pour accorder des autorisations d'administrateur à d'autres utilisateurs, choisissez l'une des options suivantes :

   - Accorder l'autorisation de l'administrateur à un ou plusieurs utilisateurs spécifiques : dans le champ Administrateurs de l'OpenSearchapplication, dans la liste contextuelle Propriétés, sélectionnez Utilisateurs IAM ou

     AWS IAM Identity Center utilisateurs, puis choisissez les utilisateurs individuels auxquels accorder des autorisations d'administrateur.
   - Accorder des autorisations d'administrateur à tous les utilisateurs : tous les utilisateurs de votre organisation ou de votre compte reçoivent des autorisations d'administrateur.
6. Choisissez Mettre à jour.

Vous pouvez supprimer des administrateurs supplémentaires, mais chaque application d' OpenSearch interface utilisateur doit conserver au moins un administrateur.

Gérer les administrateurs de l' OpenSearchinterface utilisateur à l'aide du AWS CLI

Vous pouvez créer et mettre à jour les administrateurs d'applications d' OpenSearch interface utilisateur à l'aide du AWS CLI.

Création d'administrateurs d' OpenSearch interface utilisateur à l'aide du AWS CLI

Vous trouverez ci-dessous des exemples d'ajout de responsables IAM et d'utilisateurs d'IAM Identity Center en tant qu'administrateurs lors de la création d'une OpenSearch application d'interface utilisateur.

Exemple 1 : création d'une application d' OpenSearch interface utilisateur qui ajoute un utilisateur IAM en tant qu'administrateur

Exécutez la commande suivante pour créer une application d' OpenSearch interface utilisateur qui ajoute un utilisateur IAM en tant qu'administrateur. Remplacez *placeholder values* par vos propres informations.

```
aws opensearch create-application \
    --name application-name \
    --app-configs "
        {
        \"key\":\"opensearchDashboards.dashboardAdmin.users\",
        \"value\":\"arn:aws:iam::account-id:user/user-id\"
        }
    "
```

Exemple 2 : créer une application d' OpenSearch interface utilisateur qui active IAM Identity Center et ajoute un ID utilisateur IAM Identity Center en tant qu'administrateur d'application d' OpenSearch interface utilisateur

Exécutez la commande suivante pour créer une application d' OpenSearch interface utilisateur qui active IAM Identity Center et ajoute un ID utilisateur IAM Identity Center en tant qu'administrateur d'application d' OpenSearch interface utilisateur. Remplacez *placeholder values* par vos propres informations.

`key`spécifie l'élément de configuration à définir, tel que le rôle d'administrateur pour l'application d' OpenSearch interface utilisateur. Les valeurs valides sont `opensearchDashboards.dashboardAdmin.users` et `opensearchDashboards.dashboardAdmin.groups`.

*xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx*représente la valeur attribuée à la clé, telle que l'Amazon Resource Name (ARN) d'un utilisateur IAM.

```
aws opensearch create-application \
    --name myapplication \
    --iam-identity-center-options "
        {
        \"enabled\":true,
        \"iamIdentityCenterInstanceArn\":\"arn:aws:sso:::instance/ssoins-instance-id\",
        \"iamRoleForIdentityCenterApplicationArn\":\"arn:aws:iam::account-id:role/role-
name\"
        }
    " \
    --app-configs "
        {
        \"key\":\"opensearchDashboards.dashboardAdmin.users\",
        \"value\":\"xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx\"
        }
    "
```

Mettre à jour les administrateurs de l' OpenSearch interface utilisateur en utilisant AWS CLI

Vous trouverez ci-dessous des exemples de mise à jour des principaux IAM et des utilisateurs de l'IAM Identity Center désignés en tant qu'administrateurs pour une application existante. OpenSearch

Exemple 1 : Ajouter un utilisateur IAM en tant qu'administrateur pour une application existante OpenSearch

Exécutez la commande suivante pour mettre à jour une application d' OpenSearch interface utilisateur afin d'ajouter un utilisateur IAM en tant qu'administrateur. Remplacez *placeholder values* par vos propres informations.

```
aws opensearch update-application \
    --id myapplication \
    --app-configs "
        {
        \"key\":\"opensearchDashboards.dashboardAdmin.users\",
        \"value\":\"arn:aws:iam::account-id:user/user-id\"
        }
    "
```

Exemple 2 : mettre à jour une application d' OpenSearch interface utilisateur pour ajouter un ID utilisateur IAM Identity Center en tant qu'administrateur d'application d' OpenSearch interface utilisateur

Exécutez la commande suivante pour mettre à jour une application d' OpenSearch interface utilisateur afin d'ajouter un ID utilisateur IAM Identity Center en tant qu'administrateur d'application d' OpenSearch interface utilisateur. Remplacez *placeholder values* par vos propres informations.

`key`spécifie l'élément de configuration à définir, tel que le rôle d'administrateur pour l'application d' OpenSearch interface utilisateur. Les valeurs valides sont `opensearchDashboards.dashboardAdmin.users` et `opensearchDashboards.dashboardAdmin.groups`.

*xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx*représente la valeur attribuée à la clé, telle que l'Amazon Resource Name (ARN) d'un utilisateur IAM.

```
aws opensearch update-application \
    --id myapplication \
    --app-configs "
        {
        \"key\":\"opensearchDashboards.dashboardAdmin.users\",
        \"value\":\"xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx\"
        }
    "
```

Activation de la fédération SAML avec AWS Identity and Access Management

OpenSearch L'interface utilisateur prend en charge le langage SAML (Security Assertion Markup Language 2.0), un standard ouvert utilisé par de nombreux fournisseurs d'identité. Cela permet la fédération d'identité avec AWS Identity and Access Management (IAM). Grâce à cette assistance, les utilisateurs de votre compte ou de votre organisation peuvent accéder directement à l' OpenSearch interface utilisateur en assumant les rôles IAM. Vous pouvez créer une expérience d'authentification unique initiée par le fournisseur d'identité (IdP) pour vos utilisateurs finaux, dans le cadre de laquelle ils peuvent s'authentifier auprès du fournisseur d'identité externe et être redirigés directement vers la page que vous avez définie dans l'interface utilisateur. OpenSearch Vous pouvez également mettre en œuvre un contrôle d'accès précis en configurant vos utilisateurs finaux ou vos groupes pour

qu'ils assument différents rôles IAM avec différentes autorisations d'accès à l' OpenSearch interface utilisateur et aux sources de données associées.

Cette rubrique présente des step-by-step instructions pour configurer l'utilisation de SAML avec l' OpenSearch interface utilisateur. Dans ces procédures, nous utilisons les étapes de configuration de l'application de gestion des identités et des accès Okta à titre d'exemple. Les étapes de configuration pour les autres fournisseurs d'identité, tels qu'Azure Active Directory et Ping, sont similaires.

Rubriques

- Étape 1 : configurer l'application du fournisseur d'identité (Okta)
- Étape 2 : Configuration de AWS la configuration pour Okta
- Étape 3 : Création de la politique d'accès à Amazon OpenSearch Service dans IAM
- Étape 4 : vérifier l'expérience d'authentification unique initiée par le fournisseur d'identité avec SAML

Étape 1 : configurer l'application du fournisseur d'identité (Okta)

Pour utiliser SAML avec l' OpenSearch interface utilisateur, la première étape consiste à configurer votre fournisseur d'identité.

Tâche 1 : créer des utilisateurs Okta

1. Connectez-vous à votre organisation Okta à l'adresse https://login.okta.com/en tant qu'utilisateur doté de privilèges administratifs.
2. Sur la console d'administration, sous Répertoire dans le volet de navigation, sélectionnez Personnes.
3. Choisissez Add person (Ajouter une personne).
4. Dans le champ Prénom, entrez le prénom de l'utilisateur.
5. Dans Nom de famille, entrez le nom de famille de l'utilisateur.
6. Dans Nom d'utilisateur, entrez le nom d'utilisateur de l'utilisateur au format e-mail.
7. Choisissez Je vais définir un mot de passe et entrez un mot de passe
8. (Facultatif) Décochez la case L'utilisateur doit changer de mot de passe lors de sa première connexion si vous ne souhaitez pas que l'utilisateur change le mot de passe lors de sa première connexion.
9. Choisissez Enregistrer.

Tâche 2 : créer et attribuer des groupes

1. Connectez-vous à votre organisation Okta à l'adresse https://login.okta.com/en tant qu'utilisateur doté de privilèges administratifs.
2. Sur la console d'administration, sous Répertoire dans le volet de navigation, choisissez Groups.
3. Choisissez Add Group (Ajouter un groupe).
4. Entrez un nom de groupe et choisissez Enregistrer.
5. Choisissez le groupe nouvellement créé, puis choisissez Affecter des personnes.
6. Choisissez le signe plus (+), puis cliquez sur OK.
7. (Facultatif) Répétez les étapes 1 à 6 pour ajouter d'autres groupes.

Tâche 3 : créer des applications Okta

1. Connectez-vous à votre organisation Okta à l'adresse https://login.okta.com/en tant qu'utilisateur doté de privilèges administratifs.
2. Sur la console d'administration, sous Applications dans le volet de navigation, sélectionnez Applications.
3. Choisissez Create App Integration (Créer une intégration d'appli).
4. Choisissez SAML 2.0 comme méthode de connexion, puis choisissez Next.
5. Entrez un nom pour l'intégration de votre application (par exemple,**`OpenSearch_UI`**), puis choisissez Next.
6. Entrez les valeurs suivantes dans l'application ; il n'est pas nécessaire de modifier les autres valeurs :

    a. 1. Pour l'URL d'authentification unique, entrez **`https://signin.aws.amazon.com/saml`** pour les AWS régions commerciales ou l'URL spécifique à votre région.

    b. 2. Pour l'URI d'audience (ID d'entité SP), entrez**`urn:amazon:webservices`**.

    c. 3. Pour le format Name ID, entrez**`EmailAddress`**.
7. Choisissez Suivant.
8. Choisissez Je suis un client Okta qui ajoute une application interne, puis choisissez Il s'agit d'une application interne que nous avons créée.
9. Choisissez Finish (Terminer).
10. Choisissez Affectations, puis Attribuer.

11. Choisissez Affecter aux groupes, puis sélectionnez Attribuer à côté des groupes que vous souhaitez ajouter.
12. Sélectionnez Exécuté.

Tâche 4 : configurer la configuration avancée d'Okta

Après avoir créé l'application SAML personnalisée, procédez comme suit :

1. Connectez-vous à votre organisation Okta à l'adresse https://login.okta.com/en tant qu'utilisateur doté de privilèges administratifs.

   Sur la console de l'administrateur, dans la zone Général, choisissez Modifier dans les paramètres SAML.
2. Choisissez Suivant.
3. Définissez l'état du relais par défaut sur le point de terminaison de l' OpenSearch interface utilisateur, en utilisant le format suivant :

   ```
   https://region.console.aws.amazon.com/aos/home?
   region=region#opensearch/applications/application-id/
   redirectToDashboardURL.
   ```

   Voici un exemple :

   ```
   https://us-east-2.console.aws.amazon.com/aos/home?region=us-
   east-2#opensearch/applications/abc123def4567EXAMPLE/
   redirectToDashboardURL
   ```
4. Sous Déclarations d'attribut (facultatif), ajoutez les propriétés suivantes :

   a. Indiquez le rôle IAM et le fournisseur d'identité séparés par des virgules à l'aide de l'attribut Role. Vous utiliserez ce même rôle IAM et ce même fournisseur d'identité lors d'une étape ultérieure lors de la configuration de AWS la configuration.

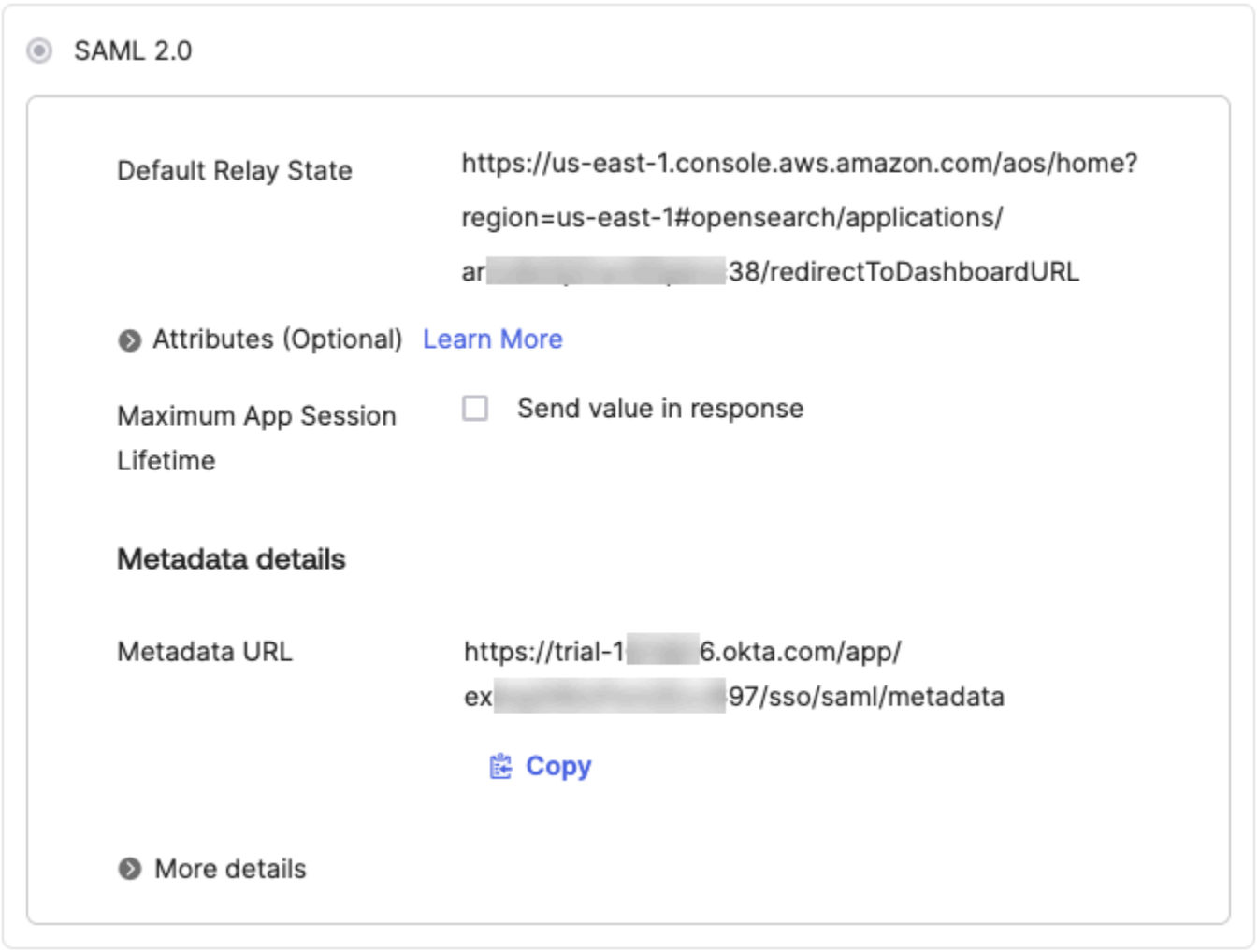
   b. Définissez user.login pour. RoleSessionName Il est utilisé comme identifiant pour les informations d'identification temporaires émises lorsque le rôle est assumé.

   À titre de référence :

Nom	Format du nom	Format	exemple
`https://aws.amazon.com/SAML/Attributes/Role`	Non précisé	`arn:aws:iam:: aws-account-id :role/role-name,arn:aws:iam:: aws-account-id :saml-provider/ provider-name`	`arn:aws:iam::111222333444:role/oktarole,arn:aws:iam::111222333444:saml-provider/oktaidp`
`https://aws.amazon.com/SAML/Attributes/RoleSessionName`	Non précisé	`user.login`	`user.login`

5. Après avoir ajouté les propriétés de l'attribut, cliquez sur Suivant, puis sur Terminer.

Le format de vos attributs doit être similaire à celui illustré dans l'image suivante. La valeur de l'état du relais par défaut est l'URL permettant de définir la page de destination pour les utilisateurs finaux de votre compte ou de votre organisation une fois qu'ils ont terminé la validation de l'authentification unique auprès d'Okta. Vous pouvez le définir sur n'importe quelle page de l' OpenSearch interface utilisateur, puis fournir cette URL aux utilisateurs finaux auxquels il est destiné.



Étape 2 : Configuration de AWS la configuration pour Okta

Effectuez les tâches suivantes pour configurer votre AWS configuration pour Okta.

Tâche 1 : recueillir des informations sur Okta

Pour cette étape, vous devez rassembler vos informations Okta afin de pouvoir les configurer ultérieurement. AWS

1. Connectez-vous à votre organisation Okta à l'adresse https://login.okta.com/en tant qu'utilisateur doté de privilèges administratifs.
2. Dans l'onglet Connexion, dans le coin inférieur droit de la page, choisissez Afficher les instructions de configuration SAML.

3. Prenez note de la valeur de l'URL d'authentification unique du fournisseur d'identité. Vous pouvez utiliser cette URL lorsque vous vous connectez à un client SQL tiers tel que SQL Workbench/J.
4. Utilisez les métadonnées du fournisseur d'identité dans le bloc 4, puis enregistrez le fichier de métadonnées au format .xml (par exemple,`metadata.xml`).

Tâche 2 : créer le fournisseur IAM

Pour créer votre fournisseur IAM, procédez comme suit :

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à https://console.aws.amazon.com/iam/l'adresse.
2. Dans le volet de navigation, sous Gestion des accès, sélectionnez Fournisseurs d'identité.
3. Choisissez Ajouter un fournisseur.
4. Pour le type de fournisseur, sélectionnez SAML.
5. Pour Nom du fournisseur, entrez un nom.
6. Pour le document de métadonnées, choisissez Choisir un fichier et chargez le fichier de métadonnées (.xml) que vous avez téléchargé précédemment.
7. Choisissez Ajouter un fournisseur.

Tâche 3 : créer un rôle IAM

Pour créer votre AWS Identity and Access Management rôle, procédez comme suit :

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à https://console.aws.amazon.com/iam/l'adresse.
2. Dans le volet de navigation, sous Gestion des accès, sélectionnez Rôles.
3. Choisissez Créer un rôle.
4. Pour le type d'entité fiable, sélectionnez la fédération SAML 2.0.
5. Pour le fournisseur basé sur SAML 2.0, choisissez le fournisseur d'identité que vous avez créé précédemment.
6. Sélectionnez Autoriser la programmation et l' AWS Management Console accès.
7. Choisissez Suivant.
8. Dans la liste des politiques d'autorisation, cochez les cases correspondant à la politique que vous avez créée précédemment et pour laquelle OpenSearchFullAccess.

9. Choisissez Suivant.
10. Dans la zone Révision, pour Nom du rôle, entrez le nom de votre rôle ; par exemple,**`oktarole`**.
11. (Facultatif) Dans Description, entrez une brève description de l'objectif du rôle.
12. Choisissez Créer un rôle.
13. Accédez au rôle que vous venez de créer, choisissez l'onglet Relations de confiance, puis choisissez Modifier la politique de confiance.
14. Dans le volet Modifier le relevé, sous Ajouter des actions pour STS, cochez la case correspondant à TagSession.
15. Choisissez Mettre à jour une politique.

Étape 3 : Création de la politique d'accès à Amazon OpenSearch Service dans IAM

Cette rubrique fournit des informations pour configurer vos rôles IAM avec accès aux OpenSearch services. Nous fournissons des exemples pour deux groupes `Alice` et `Bob` pour montrer comment obtenir un contrôle d'accès précis pour vos groupes d'utilisateurs à partir d'Okta.

Sample group: Alice

Requête :

```
GET _plugins/_security/api/roles/alice-group
```

Résultat:

```
{
  "alice-group": {
    "reserved": false,
    "hidden": false,
    "cluster_permissions": [
      "unlimited"
    ],
    "index_permissions": [
      {
        "index_patterns": [
          "alice*"
        ],
        "dls": "",
```

```
        "fls": [],
        "masked_fields": [],
        "allowed_actions": [
          "indices_all"
        ]
      }
    ],
    "tenant_permissions": [
      {
        "tenant_patterns": [
          "global_tenant"
        ],
        "allowed_actions": [
          "kibana_all_write"
        ]
      }
    ],
    "static": false
  }
}
```

Sample group: Bob

Requête :

```
GET _plugins/_security/api/roles/bob-group
```

Résultat:

```
{
  "bob-group": {
    "reserved": false,
    "hidden": false,
    "cluster_permissions": [
      "unlimited"
    ],
    "index_permissions": [
      {
        "index_patterns": [
          "bob*"
        ],
        "dls": "",
        "fls": [],
```

```
        "masked_fields": [],
        "allowed_actions": [
          "indices_all"
        ]
      }
    ],
    "tenant_permissions": [
      {
        "tenant_patterns": [
          "global_tenant"
        ],
        "allowed_actions": [
          "kibana_all_write"
        ]
      }
    ],
    "static": false
  }
}
```

Vous pouvez mapper les rôles de domaine Amazon OpenSearch Service aux rôles IAM à l'aide du mappage des rôles principaux, comme illustré dans l'exemple suivant :

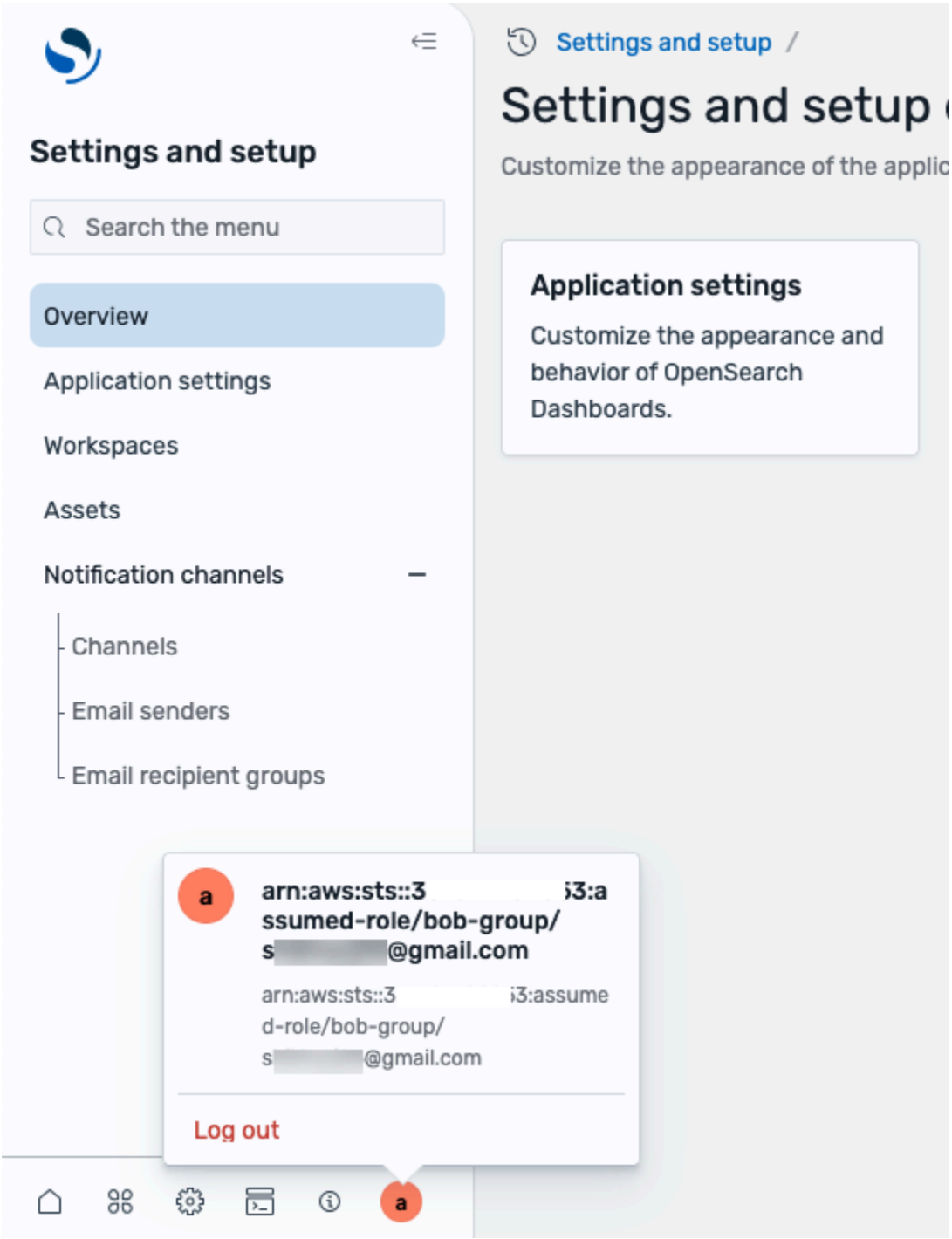
```
{
  "bob-group": {
    "hosts": [],
    "users": [],
    "reserved": false,
    "hidden": false,
    "backend_roles": [
      "arn:aws:iam::111222333444:role/bob-group"
    ],
    "and_backend_roles": []
  },
  "alice-group": {
    "hosts": [],
    "users": [],
    "reserved": false,
    "hidden": false,
    "backend_roles": [
      "arn:aws:iam::111222333444:role/alice-group"
    ],
```

```
    "and_backend_roles": []
  }
}
```

Étape 4 : vérifier l'expérience d'authentification unique initiée par le fournisseur d'identité avec SAML

Ouvrez l'URL de l'état du relais par défaut pour ouvrir la page d'authentification Okta. Entrez les informations d'identification d'un utilisateur final. Vous êtes automatiquement redirigé vers l' OpenSearch interface utilisateur.

Vous pouvez vérifier vos informations d'identification actuelles en cliquant sur l'icône utilisateur en bas du panneau de navigation, comme illustré dans l'image suivante :



Vous pouvez également vérifier les autorisations de contrôle d'accès détaillées accordées à l'utilisateur en accédant aux outils de développement en bas du panneau de navigation et en exécutant des requêtes dans la console. Vous trouverez ci-dessous des exemples de requêtes.

Example 1: Displays information about the current user

Requête :

```
GET _plugins/_security/api/account
```

Résultat:

```
{
  "user_name": "arn:aws:iam::XXXXXXXXXXX:role/bob-group",
  "is_reserved": false,
  "is_hidden": false,
  "is_internal_user": false,
  "user_requested_tenant": null,
  "backend_roles": [
    "arn:aws:iam::XXXXXXXXXXX:role/bob-group"
  ],
  "custom_attribute_names": [],
  "tenants": {
    "global_tenant": true,
    "arn:aws:iam::XXXXXXXXXXX:role/bob-group": true
  },
  "roles": [
    "bob-group"
  ]
}
```

Example 2: Displays actions permitted for a user

Requête :

```
GET bob-test/_search
```

Résultat:

```
{
  "took": 390,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
```

```
    "total": {
      "value": 1,
      "relation": "eq"
    },
    "max_score": 1,
    "hits": [
      {
        "_index": "bob-test",
        "_id": "ui01N5UBCIHpjO8Jlvfy",
        "_score": 1,
        "_source": {
          "title": "Your Name",
          "year": "2016"
        }
      }
    ]
  }
}
```

Example 3: Displays actions not permitted for a user

Requête :

```
GET alice-test
```

Résultat:

```
{
  "error": {
    "root_cause": [
      {
        "type": "security_exception",
        "reason": "no permissions for [indices:admin/get]
 and User [name=arn:aws:iam::111222333444:role/bob-group,
 backend_roles=[arn:aws:iam::111222333444:role/bob-group], requestedTenant=null]"
      }
    ],
    "type": "security_exception",
    "reason": "no permissions for [indices:admin/get]
 and User [name=arn:aws:iam::111222333444:role/bob-group,
 backend_roles=[arn:aws:iam::111222333444:role/bob-group], requestedTenant=null]"
  },
  "status": 403
```

```
}
```

Gestion des associations de sources de données et des autorisations d'accès au Virtual Private Cloud

Utilisez les procédures décrites dans cette section pour gérer les associations de sources de données et pour configurer les autorisations d'accès nécessaires pour un cloud privé virtuel (VPC).

Rubriques

- Associer une source de données à une application d' OpenSearch interface utilisateur
- Gestion de l'accès aux domaines dans un VPC
- Configuration de l'accès aux collections OpenSearch sans serveur dans un VPC

Associer une source de données à une application d' OpenSearch interface utilisateur

Après avoir créé une application d' OpenSearch interface utilisateur, vous pouvez utiliser la console ou l'associer AWS CLI à une ou plusieurs sources de données. Les utilisateurs finaux peuvent ensuite récupérer des données à partir de ces sources de données pour effectuer des recherches, travailler avec des tableaux de bord, etc.

Associer une source de données à une application d' OpenSearch interface utilisateur (console)

Pour associer une source de données à une application d' OpenSearch interface utilisateur à l'aide de la console

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Choisissez OpenSearch UI (Dashboards), puis le nom d'une application d' OpenSearch interface utilisateur.
3. Dans la zone Sources de données associées, choisissez Gérer les sources de données.
4. Choisissez parmi les OpenSearch domaines et les collections que vous souhaitez associer à l'application.

> **Tip**
>
> Si vous ne trouvez pas les sources de données que vous recherchez, contactez vos administrateurs pour qu'ils vous accordent les autorisations nécessaires. Pour de plus amples informations, veuillez consulter the section called “Autorisations pour créer une application qui utilise l'authentification IAM Identity Center (facultatif)”.

5. Choisissez Next, puis cliquez sur Enregistrer.

Une fois que vous avez associé une source de données à l'application, le bouton Lancer l'application est activé sur la page détaillée de l'application. Vous pouvez choisir Launch Application pour ouvrir la OpenSearch page Bienvenue sur laquelle vous pouvez créer et gérer des espaces de travail.

Pour plus d'informations sur l'utilisation des espaces de travail, consultezthe section called “Utilisation des espaces OpenSearch de travail Amazon Service”.

Gestion de l'accès aux domaines dans un VPC

Si un OpenSearch domaine d'un VPC était associé à l'application, un administrateur de VPC doit autoriser l'accès entre l'interface utilisateur OpenSearch et le VPC à l'aide de la console ou. AWS CLI

Gestion de l'accès aux domaines dans un VPC (console)

Pour configurer l'accès à un domaine VPC à l'aide de : AWS Management Console

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, choisissez Domains, puis choisissez le nom du domaine VPC.

   -ou-

   Choisissez Créer un domaine, puis configurez les détails du domaine.
3. Choisissez l'onglet Points de terminaison VPC, puis choisissez Authorize principal.
4. Dans la boîte de dialogue Autoriser les principaux, sélectionnez Autoriser les principaux depuis d'autres AWS services, puis choisissez des OpenSearch applications (tableau de bord) dans la liste.

5. Choisissez Authorize (Autoriser).

Gestion de l'accès aux domaines dans un VPC ()AWS CLI

Pour autoriser un domaine VPC à l'aide du AWS CLI

Pour autoriser le domaine VPC à l'aide de AWS CLI, exécutez la commande suivante. Remplacez *placeholder values* par vos propres informations.

```
aws opensearch authorize-vpc-endpoint-access \
 --domain-name domain-name \
 --service application.opensearchservice.amazonaws.com \
 --region region-id
```

Pour révoquer une association de domaine VPC à l'aide de la console

Lorsqu'une association n'est plus nécessaire, le propriétaire du domaine VPC peut révoquer l'accès en suivant la procédure suivante.

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, choisissez Domains, puis choisissez le nom du domaine VPC.
3. Choisissez l'onglet Points de terminaison VPC, puis sélectionnez le bouton correspondant à la ligne des OpenSearch applications (tableau de bord).
4. Choisissez Révoquer l'accès.

Pour révoquer une association de domaine VPC à l'aide du AWS CLI

Pour révoquer une association de domaine VPC avec OpenSearch l'application d'interface utilisateur, exécutez la commande suivante. Remplacez *placeholder values* par vos propres informations.

```
aws opensearch revoke-vpc-endpoint-access \
    --domain-name domain-name \
    --service application.opensearchservice.amazonaws.com \
    --region region-id
```

Configuration de l'accès aux collections OpenSearch sans serveur dans un VPC

Si une collection Amazon OpenSearch Serverless dans un VPC était associée à l'application, un administrateur VPC peut autoriser l'accès en créant une nouvelle politique réseau et en l'attachant à la collection.

Configuration de l'accès aux collections OpenSearch sans serveur dans un VPC (console)

Pour configurer l'accès aux collections OpenSearch sans serveur dans un VPC à l'aide de la console

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, sélectionnez Stratégies réseau, choisissez le nom de la stratégie réseau, puis sélectionnez Modifier.

   -ou-

   Choisissez Créer une stratégie réseau, puis configurez les détails de la stratégie.
3. Dans la zone Type d'accès, choisissez Privé (recommandé), puis sélectionnez Accès privé au AWS service.
4. Dans le champ de recherche, choisissez Service, puis choisissez`application.opensearchservice.amazonaws.com`.
5. Dans la zone Type de ressource, sélectionnez la case Activer l'accès au OpenSearch point de terminaison.
6. Pour Rechercher des collections, ou saisir des termes de préfixes spécifiques, dans le champ de recherche, sélectionnez Nom de la collection, puis entrez ou sélectionnez le nom des collections à associer à la politique réseau.
7. Choisissez Créer pour une nouvelle politique réseau ou Mettre à jour pour une politique réseau existante.

Configuration de l'accès aux collections OpenSearch sans serveur dans un AWS CLI VPC ()

Pour configurer l'accès aux collections OpenSearch sans serveur dans un VPC à l'aide du AWS CLI

1. Créez un fichier .json similaire au suivant. Remplacez *placeholder values* par vos propres informations.

```
{
    "Description" : "policy-description",
    "Rules": [{
       "ResourceType" : "collection",
        "Resource" : ["collection/collection_name"]
     }],
    "SourceServices" : [
          "application.opensearchservice.amazonaws.com"
      ],
      "AllowFromPublic" : false
}
```

2. Créez ou mettez à jour une politique réseau pour une collection dans un VPC afin qu'elle fonctionne avec les applications d' OpenSearch interface utilisateur.

   Create a network policy

   Exécutez la commande suivante. Remplacez *placeholder values* par vos propres informations.

```
aws opensearchserverless create-security-policy \
    --type network  \
    --region region \
    --endpoint-url endpoint-url \
    --name network-policy-name \
    --policy file:/path_to_network_policy_json_file
```

   La commande renvoie des informations semblables à ce qui suit :

```
{
    "securityPolicyDetail": {
        "createdDate": ******,
        "lastModifiedDate": ******,
```

```
        "name": "network-policy-name",
        "policy": [
            {
                "SourceVPCEs": [],
                "AllowFromPublic": false,
                "Description": "",
                "Rules": [
                    {
                        "Resource": [
                            "collection/network-policy-name"
                        ],
                        "ResourceType": "collection"
                    }
                ],
                "SourceServices": [
                    "application.opensearchservice.amazonaws.com"
                ]
            }
        ],
        "policyVersion": "******",
        "type": "network"
    }
}
```

Update a network policy

Exécutez la commande suivante. Remplacez *placeholder values* par vos propres informations.

```
aws opensearchserverless update-security-policy \
    --type network  \
    --region region \
    --endpoint-url endpoint-url \
    --name network-policy-name \
    --policy-version "policy_version_from_output_of_network_policy_creation" \
    --policy file:/path_to_network_policy_json_file
```

La commande renvoie des informations semblables à ce qui suit :

```
{
    "securityPolicyDetail": {
        "createdDate": ******,
```

```
        "lastModifiedDate": ******,
        "name": "network-policy-name",
        "policy": [
            {
                "SourceVPCEs": [],
                "AllowFromPublic": false,
                "Description": "",
                "Rules": [
                    {
                        "Resource": [
                            "collection/network-policy-name"
                        ],
                        "ResourceType": "collection"
                    }
                ],
                "SourceServices": [
                    "application.opensearchservice.amazonaws.com"
                ]
            }
        ],
        "policyVersion": "******",
        "type": "network"
    }
}
```

Utilisation des espaces OpenSearch de travail Amazon Service

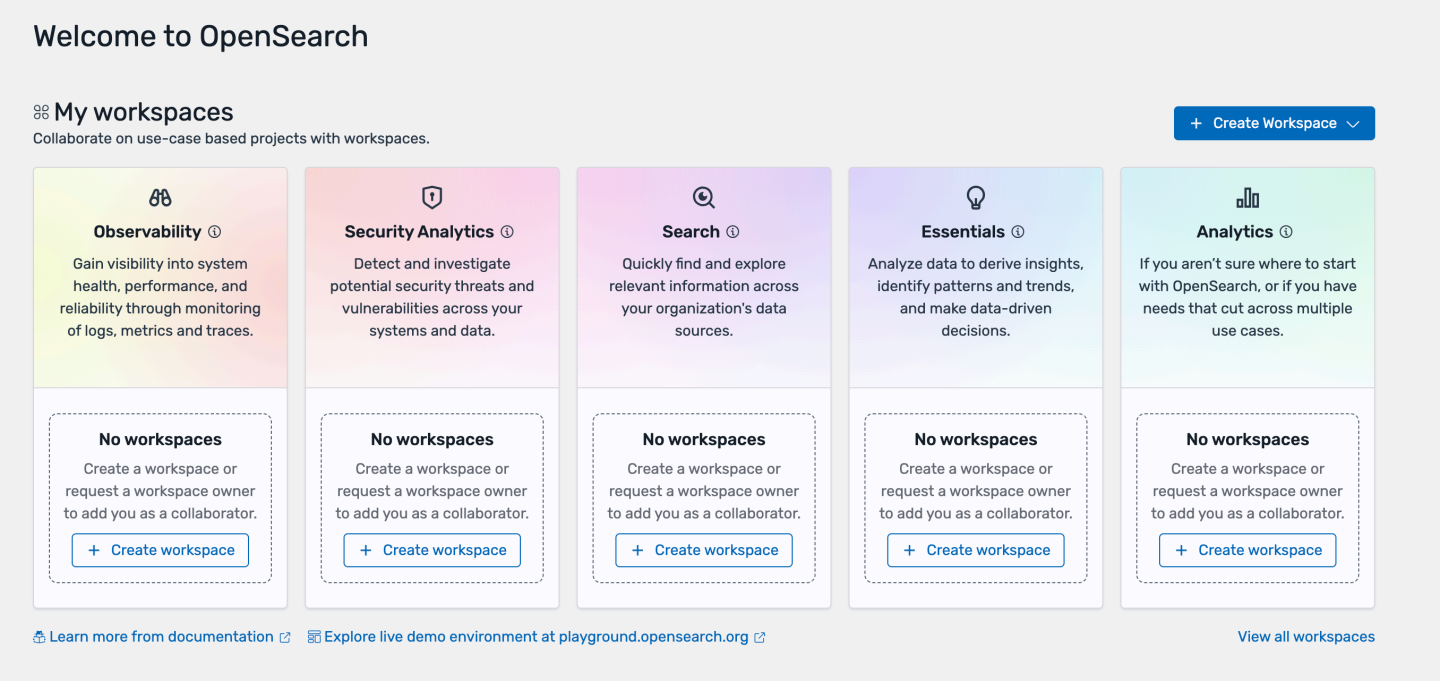
Amazon OpenSearch Service prend en charge la création de plusieurs espaces de travail spécifiques à des cas d'utilisation. Chaque espace de travail propose une expérience personnalisée adaptée aux cas d'utilisation courants tels que l'observabilité, l'analyse de sécurité et la recherche. Workspace prend également en charge la gestion des collaborateurs, de sorte que vous pouvez partager votre espace de travail uniquement avec les collaborateurs auxquels vous êtes destinés et gérer les autorisations de chacun d'entre eux.

Création d'espaces de travail d'applications d' OpenSearch interface utilisateur

Une fois qu'une application d' OpenSearch interface utilisateur a été créée et associée à des sources de données, et que les autorisations utilisateur ont été configurées pour l'application, vous pouvez lancer l'application d' OpenSearch interface utilisateur pour créer des espaces de travail.

Pour commencer à créer un espace de travail, vous pouvez sélectionner le bouton Lancer l'application sur la page détaillée de l'application ou utiliser l'URL de l'application d' OpenSearch interface utilisateur pour ouvrir la page d'accueil de l'application d' OpenSearch interface utilisateur dans une nouvelle fenêtre de navigateur.

L'application d' OpenSearch interface utilisateur fournit des options pour créer des espaces de travail et répertorie tous les espaces de travail existants sur la page d'accueil, classés par cas d'utilisation.



Pour plus d'informations sur les types d'espaces de travail pris en charge, consultezthe section called “Types d'espaces de travail”.

Confidentialité de l'espace de travail et collaborateurs

Vous pouvez définir un paramètre de confidentialité pour un espace de travail comme niveau d'autorisation par défaut pour tous les utilisateurs. Vous pouvez le faire lors de la création d'un espace de travail ou modifier un espace de travail existant (dans l'onglet Collaborateurs de l'espace de travail). Vous avez le choix entre trois options de confidentialité :

- Réservé aux collaborateurs : seuls les collaborateurs que vous ajoutez explicitement à l'espace de travail peuvent accéder à l'espace de travail. Vous pouvez définir des niveaux d'autorisation pour chaque collaborateur.

- Tout le monde peut consulter : toute personne ayant accès à l'application d' OpenSearch interface utilisateur peut accéder à l'espace de travail et consulter ses actifs, mais elle ne peut y apporter aucune modification.
- Tout le monde peut modifier : toute personne ayant accès à l'application d' OpenSearch interface utilisateur peut accéder à l'espace de travail, consulter les actifs qu'il contient et apporter des modifications aux actifs de l'espace de travail.

Dans l'onglet Collaborateurs de l'espace de travail, vous pouvez ajouter des utilisateurs ou des rôles IAM et des AWS IAM Identity Center utilisateurs en tant que collaborateurs dans un espace de travail. Les collaborateurs peuvent choisir entre trois niveaux d'autorisations :

- Lecture seule : le collaborateur peut uniquement consulter les actifs de l'espace de travail. Ce paramètre est remplacé si l'espace de travail est configuré pour utiliser le paramètre de confidentialité Tout le monde peut modifier.
- Lecture et écriture : le collaborateur peut consulter et modifier des ressources dans l'espace de travail. Si l'espace de travail est configuré pour utiliser le paramètre de confidentialité Tout le monde peut consulter, le collaborateur peut toujours le modifier.
- Administrateur — Le collaborateur peut mettre à jour les paramètres et supprimer l'espace de travail. Le collaborateur peut également modifier les paramètres de confidentialité de l'espace de travail et gérer les collaborateurs. L'utilisateur qui crée l'espace de travail est automatiquement désigné comme administrateur de l'espace de travail.

Types d'espaces de travail

Amazon OpenSearch Service propose cinq types d'espaces de travail, chacun doté de fonctionnalités différentes pour les différents cas d'utilisation :

- L'espace de travail Observability est conçu pour améliorer la visibilité sur l'état, les performances et la fiabilité du système grâce à la surveillance des journaux, des métriques et des traces.
- L'espace de travail Security Analytics est conçu pour détecter et étudier les menaces de sécurité et les vulnérabilités potentielles de vos systèmes et de vos données.
- L'espace de travail de recherche est conçu pour rechercher et explorer rapidement les informations pertinentes dans les sources de données de votre organisation.
- L'espace de travail Essentials est conçu pour le OpenSearch mode Serverless en tant que source de données et permet d'analyser les données pour en tirer des informations, identifier les

modèles et les tendances, et prendre rapidement des décisions basées sur les données. Vous pouvez rechercher et explorer les informations pertinentes dans les sources de données de votre organisation dans un espace de travail Essentials.

- L'espace de travail Analytics (toutes les fonctionnalités) est conçu pour des cas d'utilisation polyvalents et prend en charge toutes les fonctionnalités disponibles dans l'interface utilisateur du OpenSearch service (tableaux de bord).

Accès aux données entre régions et entre comptes grâce à la recherche entre clusters

À l'aide de la recherche entre clusters dans Amazon OpenSearch Serverless, vous pouvez effectuer des requêtes et des agrégations sur plusieurs domaines connectés.

La recherche entre clusters dans Amazon OpenSearch Serverless utilise les concepts de domaine source et de domaine de destination. Une demande de recherche entre clusters provient d'un domaine source. Le domaine de destination peut se trouver dans un domaine différent Compte AWS ou Région AWS (ou les deux) pour le domaine source à partir duquel effectuer la requête. À l'aide de la recherche entre clusters, vous pouvez configurer un domaine source à associer à votre OpenSearch interface utilisateur dans le même compte, puis créer des connexions aux domaines de destination. Par conséquent, vous pouvez utiliser l' OpenSearch interface utilisateur avec les données des domaines de destination, même s'ils appartiennent à un autre compte ou à une autre région.

Vous payez les frais de transfert de AWS données standard pour les données transférées vers et depuis Amazon OpenSearch Service. Les données transférées entre les nœuds de votre domaine de OpenSearch service ne vous sont pas facturées. Pour plus d'informations sur les frais d'entrée et de sortie de données, consultez la section Transfert de données sur la page de tarification d'Amazon EC2 On-Demand.

Vous pouvez utiliser la recherche entre clusters comme mécanisme permettant d'associer votre OpenSearch interface utilisateur à des clusters d'un autre compte ou d'une autre région. Les demandes entre domaines sont cryptées en transit par défaut dans le cadre du node-to-node chiffrement.

> **Note**
>
> L' OpenSearch outil open source documente également la recherche entre clusters. Notez que la configuration de l'outil open source est très différente pour les clusters open source par rapport aux domaines Amazon OpenSearch Serverless gérés.
> Plus particulièrement, dans Amazon OpenSearch Serverless, vous configurez les connexions entre clusters à l'aide de requêtes au AWS Management Console lieu d'utiliser`cURL`. Le service géré utilise AWS Identity and Access Management (IAM) pour l'authentification entre clusters en plus d'un contrôle d'accès précis.
> Par conséquent, nous vous recommandons d'utiliser le contenu de cette rubrique pour configurer la recherche entre clusters pour vos domaines plutôt que la OpenSearch documentation open source.

Différences fonctionnelles lors de l'utilisation de la recherche entre clusters

Par rapport aux domaines classiques, les domaines de destination créés à l'aide de la recherche entre clusters présentent les différences fonctionnelles et les exigences suivantes :

- Vous ne pouvez ni écrire ni exécuter de `PUT` commandes sur le cluster distant. Votre accès au cluster distant est en lecture seule.
- Le domaine source et le domaine de destination doivent tous deux être OpenSearch des domaines. Vous ne pouvez pas connecter un domaine Elasticsearch ou des clusters OpenSearch / Elasticsearch autogérés pour l'interface utilisateur. OpenSearch
- Un domaine peut avoir un maximum de 20 connexions à d'autres domaines. Cela inclut les connexions sortantes et entrantes.
- Le domaine source doit se trouver sur une version identique ou supérieure à OpenSearch celle du domaine de destination. Si vous souhaitez configurer des connexions bidirectionnelles entre deux domaines, les deux domaines doivent être dans la même version. Nous vous recommandons de mettre à jour les deux domaines vers la dernière version avant d'établir la connexion. Si vous devez mettre à jour des domaines après avoir configuré la connexion bidirectionnelle, vous devez d'abord supprimer la connexion, puis la recréer par la suite.
- Vous ne pouvez pas utiliser de dictionnaires personnalisés ou de code SQL avec les clusters distants.
- Vous ne pouvez pas l'utiliser AWS CloudFormation pour connecter des domaines.

- Vous ne pouvez pas utiliser la recherche inter-clusters sur des instances M3 ou les instances extensibles (T2 et T3).
- La recherche entre clusters ne fonctionne pas pour les collections Amazon OpenSearch Serverless.

Conditions requises pour la recherche entre clusters pour l'interface utilisateur OpenSearch

Avant de configurer la recherche entre clusters avec deux OpenSearch domaines, assurez-vous que ceux-ci répondent aux exigences suivantes :

- Le contrôle d'accès détaillé est activé pour les deux domaines
- Node-to-node le chiffrement est activé pour les deux domaines

Rubriques

- Configuration des autorisations d'accès pour l'accès aux données entre régions et entre comptes grâce à la recherche entre clusters
- Création d'une connexion entre les domaines
- Test de votre configuration de sécurité pour l'accès aux données entre régions et entre comptes grâce à la recherche entre clusters
- Suppression d’une connexion

Configuration des autorisations d'accès pour l'accès aux données entre régions et entre comptes grâce à la recherche entre clusters

Lorsque vous envoyez une demande de recherche entre clusters au domaine source, le domaine évalue cette demande par rapport à sa politique d'accès au domaine. La recherche entre clusters nécessite un contrôle d'accès précis. Voici un exemple de politique de libre accès sur le domaine source.

JSON

```
{
"Version": "2012-10-17",
"Statement": [
    {
```

```
        "Effect": "Allow",
        "Principal": {
          "AWS": [
            "*"
          ]
        },
        "Action": [
          "es:ESHttp*"
        ],
        "Resource": "arn:aws:es:region:111222333444:domain/src-domain/*"
      }
    ]
}
```

> **Note**
>
> Si vous incluez des index distants dans le chemin, vous devez encoder l'URI en URL dans l'ARN du domaine.
> Par exemple, utilisez le format ARN suivant : `:arn:aws:es:us-east-1:111222333444:domain/my-domain/local_index,dst%3Aremote_index`
> N'utilisez pas le format ARN suivant : `arn:aws:es:us-east-1:111222333444:domain/my-domain/local_index,dst:remote_index.`

Si vous choisissez d'utiliser une politique d'accès restrictive en plus d'un contrôle d'accès précis, votre politique doit au moins autoriser l'accès à. `es:ESHttpGet` Voici un exemple :

JSON

```
{
"Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::111222333444:user/john-doe"
```

```
        ]
      },
      "Action": "es:ESHttpGet",
      "Resource": "arn:aws:es:us-east-1:account:domain/my-domain/*"
    }
  ]
}
```

Le contrôle d'accès précis sur le domaine source évalue la demande afin de déterminer si elle est signée avec des informations d'identification de base IAM ou HTTP valides. Si tel est le cas, un contrôle d'accès précis évalue ensuite si l'utilisateur est autorisé à effectuer la recherche et à accéder aux données.

Les autorisations requises pour les recherches sont les suivantes :

- Si la demande recherche uniquement des données sur le domaine de destination (par exemple`dest-alias:dest-index/_search`), les autorisations ne sont requises que sur le domaine de destination).
- Si la demande recherche des données sur les deux domaines (par exemple`source-index,dest-alias:dest-index/_search`), des autorisations sont requises pour les deux domaines).
- Pour utiliser un contrôle d'accès précis, l'autorisation `indices:admin/shards/search_shards` est requise en plus des autorisations de lecture ou de recherche standard pour les index concernés.

Le domaine source transmet la demande au domaine de destination. Le domaine de destination évalue cette demande en fonction de sa stratégie d'accès au domaine. Pour prendre en charge toutes les fonctionnalités de l' OpenSearch interface utilisateur, telles que l'indexation de documents et l'exécution de recherches standard, des autorisations complètes doivent être définies. Voici un exemple de notre politique recommandée pour le domaine de destination :

JSON

```
{
"Version": "2012-10-17",
 "Statement": [
    {
```

```
       "Effect": "Allow",
       "Principal": {
         "AWS": [
           "*"
         ]
       },
       "Action": [
         "es:ESHttp*"
       ],
       "Resource": "arn:aws:es:us-east-2:111222333444:domain/my-destination-
domain/*"
    },
    {
    "Effect": "Allow",
        "Principal": {
    "AWS": "*"
      },
      "Action": "es:ESCrossClusterGet",
      "Resource": "arn:aws:es:us-east-2:111222333444:domain/"
    }
  ]
}
```

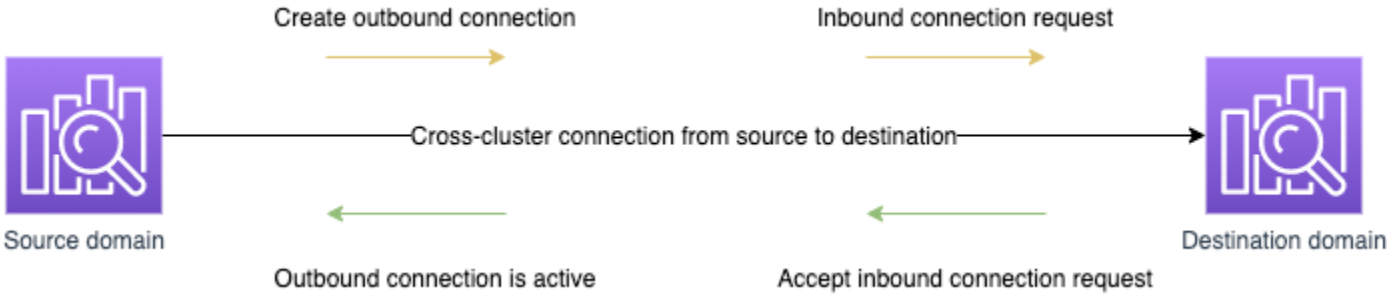
Si vous souhaitez effectuer uniquement des recherches de base, la politique minimale requise est que l'`es:ESCrossClusterGet`autorisation soit appliquée pour le domaine de destination sans prise en charge des caractères génériques. Par exemple, dans la politique précédente, vous devez spécifier le nom de domaine comme *`/my-destination-domain`* et non*`/my-destination-domain/*`*.

Dans ce cas, le domaine de destination effectue la recherche et renvoie les résultats au domaine source. Le domaine source combine ses propres résultats (le cas échéant) avec ceux du domaine de destination et vous les renvoie.

Création d'une connexion entre les domaines

Une connexion de recherche entre clusters est unidirectionnelle entre le domaine source et le domaine de destination. Cela signifie que les domaines de destination (dans un autre compte ou une autre région) ne peuvent pas interroger le domaine source, qui est local dans l' OpenSearch interface utilisateur. Le domaine source crée une connexion sortante vers le domaine de destination.

Le domaine de destination reçoit une demande de connexion entrante en provenance du domaine source.



Pour créer une connexion entre des domaines

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, sélectionnez Domaines.
3. Choisissez le nom du domaine à utiliser comme domaine source, puis cliquez sur l'onglet Connexions.
4. Dans la zone Connexions sortantes, sélectionnez Request.
5. Dans Alias de la connexion, saisissez un nom pour votre connexion. L'alias de connexion est utilisé dans l' OpenSearch interface utilisateur pour sélectionner les domaines de destination.
6. Pour le mode de connexion, choisissez Direct pour les recherches entre clusters ou la réplication.
7. Pour spécifier que la connexion doit ignorer les clusters non disponibles lors d'une recherche, cochez la case Ignorer les clusters non disponibles. Le choix de cette option garantit que vos requêtes inter-clusters renvoient des résultats partiels, indépendamment des défaillances survenant sur un ou plusieurs clusters distants.
8. Pour le cluster de destination, choisissez entre Se connecter à un cluster dans ce cas Compte AWS et Se connecter à un cluster dans un autre Compte AWS.
9. Pour l'ARN du domaine distant, entrez le nom de ressource Amazon (ARN) du cluster. L'ARN du domaine se trouve dans la zone Informations générales de la page détaillée du domaine.

   Le domaine doit répondre aux exigences suivantes :

   - L'ARN doit être au format`arn:`*`partition`*`:es:`*`regionaccount-id`*`:`*`type`*`/`*`domain-id`*. Par exemple :

     ```
     arn:aws:es:us-east-2:111222333444:domain/my-domain
     ```

- Le domaine doit être configuré pour utiliser la OpenSearch version 1.0 (ou ultérieure) ou la version 6.7 (ou ultérieure) d'Elasticsearch.
- Le contrôle d'accès détaillé doit être activé sur le domaine.
- Le domaine doit être en cours d'exécution OpenSearch.

10. Choisissez Request (Demander).

La recherche inter-clusters valide d'abord la demande de connexion pour s'assurer que les conditions préalables sont remplies. Si les domaines sont incompatibles, la demande de connexion entre dans l'`Validation failed`état.

Si la demande de connexion est validée avec succès, elle est envoyée au domaine de destination, où elle doit être approuvée. Jusqu'à ce que cette approbation soit donnée, la connexion reste en bon `Pending acceptance` état. Lorsque la demande de connexion sera acceptée au niveau du domaine de destination, l'état passera à `Active` et le domaine de destination deviendra disponible pour les demandes.

La page du domaine affiche l'état global du domaine et les détails de l'état de l'instance de votre domaine de destination. Seuls les propriétaires de domaines ont la possibilité de créer, de visualiser, de supprimer et de surveiller les connexions vers ou depuis leurs domaines.

Une fois la connexion établie, tout le trafic qui circule entre les nœuds des domaines connectés est chiffré. Lorsque vous connectez un domaine VPC à un domaine non VPC et que le domaine non VPC est un point de terminaison public pouvant recevoir du trafic en provenance d'Internet, le trafic inter-clusters entre les domaines reste crypté et sécurisé.

Test de votre configuration de sécurité pour l'accès aux données entre régions et entre comptes grâce à la recherche entre clusters

Après avoir configuré les autorisations d'accès pour l'accès aux données entre régions et entre comptes avec la recherche entre clusters, nous vous recommandons de tester la configuration à l'aide Postmand'une plateforme tierce pour le développement collaboratif d'API.

Pour configurer votre configuration de sécurité à l'aide de Postman

1. Dans le domaine de destination, indexez un document. Voici un exemple de demande :

```
POST https://dst-domain.us-east-1.es.amazonaws.com/books/_doc/1
{
```

```
"Dracula": "Bram Stoker"
}
```

2. Pour interroger cet index à partir du domaine source, incluez l'alias de connexion du domaine de destination dans la requête. Vous trouverez l'alias de connexion dans l'onglet Connexions du tableau de bord de votre domaine. Voici un exemple de demande et de réponse tronquée :

```
GET https://src-domain.us-east-1.es.amazonaws.com/connection_alias:books/_search
{
    ...
  "hits": [
    {
"_index": "source-destination:books",
      "_type": "_doc",
      "_id": "1",
      "_score": 1,
      "_source": {
"Dracula": "Bram Stoker"
      }
    }
  ]
}
```

3. (Facultatif) Vous pouvez créer une configuration qui inclut plusieurs domaines dans une seule recherche. Supposons, par exemple, que vous ayez configuré ce qui suit :

   Une connexion entre `domain-a` à`domain-b`, avec un alias de connexion nommé `cluster_b`

   Une connexion entre `domain-a` à`domain-c`, avec un alias de connexion nommé `cluster_c`

   Dans ce cas, vos recherches incluent le contenu `domain-a``domain-b`, et`domain-c`. Voici un exemple de demande et de réponse :

   Demande

```
GET https://src-domain.us-east-1.es.amazonaws.com/
local_index,cluster_b:b_index,cluster_c:c_index/_search
{
  "query": {
    "match": {
      "user": "domino"
    }
  }
```

```
}
```

Réponse :

```
{
"took": 150,
  "timed_out": false,
  "_shards": {
"total": 3,
    "successful": 3,
    "failed": 0,
    "skipped": 0
  },
  "_clusters": {
"total": 3,
    "successful": 3,
    "skipped": 0
  },
  "hits": {
"total": 3,
    "max_score": 1,
    "hits": [
      {
"_index": "local_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 1,
        "_source": {
"user": "domino",
          "message": "This is message 1",
          "likes": 0
        }
      },
      {
"_index": "cluster_b:b_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 2,
        "_source": {
"user": "domino",
          "message": "This is message 2",
          "likes": 0
        }
```

```
      },
      {
"_index": "cluster_c:c_index",
        "_type": "_doc",
        "_id": "0",
        "_score": 3,
        "_source": {
"user": "domino",
          "message": "This is message 3",
          "likes": 0
        }
      }
    ]
  }
}
```

Si vous n'avez pas choisi d'ignorer les clusters non disponibles dans la configuration de votre connexion, tous les clusters de destination que vous recherchez doivent être disponibles pour que votre demande de recherche s'exécute correctement. Sinon, la requête entière échoue. Même si l'un des domaines n'est pas disponible, aucun résultat de recherche n'est renvoyé.

Suppression d’une connexion

La suppression d'une connexion arrête toute opération de recherche entre clusters sur le domaine de destination.

Vous pouvez exécuter la procédure suivante sur le domaine source ou de destination pour supprimer la connexion. Une fois la connexion supprimée, elle reste visible avec un statut `Deleted` de 15 jours.

Vous ne pouvez pas supprimer un domaine avec des connexions inter-clusters actives. Pour supprimer un domaine, commencez par supprimer toutes ses connexions entrantes et sortantes. Vous vous assurez ainsi de tenir compte des utilisateurs de domaines de clusters croisés avant de supprimer le domaine.

Pour supprimer une connexion

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, sélectionnez Domaines.
3. Choisissez le nom du domaine à supprimer, puis cliquez sur l'onglet Connexions.

4. Sélectionnez le nom de la connexion à supprimer.
5. Choisissez Supprimer, puis confirmez la suppression.

Gestion de l'accès à l' OpenSearch interface utilisateur depuis un point de terminaison VPC

Vous pouvez créer une connexion privée entre votre VPC et l' OpenSearch interface utilisateur à l'aide de. AWS PrivateLink Grâce à cette connexion, vous pouvez accéder aux applications d' OpenSearch interface utilisateur comme si elles se trouvaient dans le même VPC. De cette façon, vous n'avez pas besoin de configurer une passerelle Internet, un périphérique NAT, une connexion VPN ou d' AWS Direct Connect établir la connexion. Les instances de votre VPC n'ont pas besoin d'adresses IP publiques pour accéder OpenSearch à l'interface utilisateur.

Pour établir cette connexion privée, vous devez d'abord créer un point de terminaison d'interface alimenté par AWS PrivateLink. Une interface réseau de point de terminaison est créée automatiquement dans chaque sous-réseau que vous spécifiez pour le point de terminaison de l'interface. Il s'agit d'interfaces réseau gérées par les demandeurs qui servent de point d'entrée pour le trafic destiné aux applications d' OpenSearch interface utilisateur.

Création d'une connexion privée entre un VPC et une interface utilisateur OpenSearch

Vous pouvez créer une connexion privée pour accéder à l' OpenSearch interface utilisateur depuis un VPC à l'aide du AWS Management Console ou. AWS CLI

Création d'une connexion privée entre un VPC et une OpenSearch interface utilisateur (console)

Pour créer une connexion privée entre un VPC et une OpenSearch interface utilisateur à l'aide de la console

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, sous Serverless, choisissez les points de terminaison VPC.
3. Choisissez Create VPC endpoint (Créer un point de terminaison d'un VPC).

4. Dans Nom, entrez le nom du point de terminaison.
5. Pour le VPC, sélectionnez le VPC à partir duquel vous allez accéder aux OpenSearch applications d'interface utilisateur.
6. Pour les sous-réseaux, sélectionnez un sous-réseau à partir duquel vous allez accéder aux applications d' OpenSearch interface utilisateur.

> **Note**
>
> L'adresse IP et le type DNS d'un point de terminaison sont basés sur le type de sous-réseau :
>
> - Double pile : si tous les sous-réseaux possèdent à la fois des plages d' IPv6 adresses IPv4 et des plages d'adresses.
> - IPv6: Si tous les sous-réseaux IPv6 ne sont que des sous-réseaux.
> - IPv4: si tous les sous-réseaux ont des plages d' IPv4 adresses.

7. Pour les groupes de sécurité, sélectionnez un ou plusieurs groupes de sécurité à associer aux interfaces réseau des terminaux.

> **Note**
>
> Au cours de cette étape, vous limitez les ports, les protocoles et les sources de trafic entrant que vous autorisez sur votre terminal. Assurez-vous que les règles du groupe de sécurité autorisent les ressources qui utiliseront le point de terminaison VPC pour communiquer avec les applications d' OpenSearch interface utilisateur à communiquer également avec l'interface réseau du point de terminaison.

8. 8. Choisissez Créer un point de terminaison.

Création d'une connexion privée entre un VPC et une OpenSearch interface utilisateur ()AWS CLI

Pour créer une connexion privée entre un VPC et une OpenSearch interface utilisateur à l'aide du AWS CLI

Exécutez la commande suivante. Remplacez *placeholder values* par vos propres informations.

```
aws opensearchserverless create-vpc-endpoint \
    --region region \
    --endpoint endpoint \
    --name vpc_endpoint_name \
    --vpc-id vpc_id \
    --subnet-ids subnet_ids
```

Mise à jour de la politique de point de terminaison du VPC pour autoriser l'accès à l' OpenSearch application d'interface utilisateur

Après avoir créé la connexion privée, mettez à jour la politique de point de terminaison du VPC pour autoriser l'accès à l'application d' OpenSearch interface utilisateur dans la politique du point de terminaison du VPC en spécifiant l'ID de l'application.

Pour plus d'informations sur la mise à jour d'une politique de point de terminaison VPC, consultez la section Mettre à jour une politique de point de terminaison de VPC dans le Guide.AWS PrivateLink

Assurez-vous que la politique de point de terminaison du VPC inclut l'énoncé suivant. Remplacez *placeholder value* par vos propres informations.

```
{
    "Statement": [{
        "Action": ["opensearch:*"],
        "Effect": "Allow",
        "Principal": "*",
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "opensearch:ApplicationId": ["opensearch-ui-application-id"]
            }
        }
    }]
}
```

Révocation de l'accès à l' OpenSearchinterface utilisateur dans une politique de point de terminaison VPC

OpenSearch L'interface utilisateur nécessite une autorisation explicite dans la politique de point de terminaison du VPC pour permettre aux utilisateurs d'accéder à l'application depuis le VPC. Si

vous ne souhaitez plus que les utilisateurs accèdent à l' OpenSearch interface utilisateur depuis le VPC, vous pouvez supprimer l'autorisation dans la politique du point de terminaison. Ensuite, les utilisateurs rencontrent un message d'`403 forbidden`erreur lorsqu'ils tentent d'accéder à l' OpenSearch interface utilisateur.

Pour plus d'informations sur la mise à jour d'une politique de point de terminaison VPC, consultez la section Mettre à jour une politique de point de terminaison de VPC dans le Guide.AWS PrivateLink

Voici un exemple de politique de point de terminaison du VPC qui refuse l'accès aux applications d'interface utilisateur depuis le VPC :

```
{
    "Statement": [{
        "Action": ["opensearch:*"],
        "Effect": "Allow",
        "Principal": "*",
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "opensearch:ApplicationId": [""]
            }
        }
    }]
}
```

Gestion des index dans Amazon Service OpenSearch

Après avoir ajouté des données à Amazon OpenSearch Service, vous devez souvent les réindexer, utiliser des alias d'index, déplacer un index vers un stockage plus rentable ou le supprimer complètement. Ce chapitre couvre le UltraWarm stockage, le stockage à froid et la gestion de l'état des index. Pour plus d'informations sur l' OpenSearch index APIs, consultez la OpenSearch documentation.

Rubriques

- UltraWarm stockage pour Amazon OpenSearch Service
- Stockage à froid pour Amazon OpenSearch Service
- OpenSearch stockage optimisé pour Amazon OpenSearch Service
- Index State Management dans Amazon OpenSearch Service
- Synthèse des index dans Amazon OpenSearch Service avec des cumulatifs d'index
- Transformations d'index dans Amazon Service OpenSearch
- Réplication entre clusters pour Amazon Service OpenSearch
- Migration des index Amazon OpenSearch Service à l'aide de la réindexation à distance
- Gestion des données en séries chronologiques dans Amazon OpenSearch Service avec des flux de données

UltraWarm stockage pour Amazon OpenSearch Service

UltraWarm offre un moyen économique de stocker de vastes quantités de données en lecture seule sur Amazon OpenSearch Service. Les nœuds de données standard utilisent un stockage « hot », qui prend la forme de magasins d'instance ou de volumes Amazon EBS attachés à chaque nœud. Le stockage hot offre les performances les plus rapides possibles pour l'indexation et la recherche de nouvelles données.

Plutôt que le stockage attaché, UltraWarm les nœuds utilisent Amazon S3 et une solution de mise en cache sophistiquée pour améliorer les performances. Pour les index dans lesquels vous n'écrivez pas activement, que vous interrogez moins fréquemment et desquels vous n'attendez pas les mêmes performances, UltraWarm offrent des coûts par Gio de données nettement inférieurs. Étant donné que les index à chaud (warm) sont en lecture seule, sauf si vous les rebasculez dans le stockage hot, UltraWarm ils conviennent particulièrement aux données immuables, telles que les journaux.

Dans OpenSearch, ces index à chaud se comportent comme n'importe quel autre index. Vous pouvez les interroger à l'aide de la même méthode APIs ou les utiliser pour créer des visualisations dans OpenSearch Dashboards.

Rubriques

- Prérequis
- UltraWarm exigences relatives aux performances et exigences relatives aux performances
- UltraWarm tarification
- Activant UltraWarm
- Migration des index vers le stockage UltraWarm
- Automatisation des migrations
- Réglage des migrations
- Annulation des migrations
- Liste des index hot et warm
- Rebasculement d'index à chaud vers le stockage hot
- Restauration des index à chaud à partir d'instantanés
- Instantanés manuels des index warm
- Migration d'index à chaud vers le stockage à froid
- Bonnes pratiques relatives aux index KNN
- Désactivation UltraWarm

Prérequis

UltraWarm comporte quelques prérequis importants :

- UltraWarm nécessite OpenSearch Elasticsearch 6.8 ou version ultérieure.
- Pour utiliser le stockage à chaud (warm), les domaines doivent disposer de nœuds principaux dédiés.
- Lorsque vous utilisez un domaine Multi-AZ avec veille, le nombre de nœuds chauds doit être un multiple du nombre de zones de disponibilité utilisées.
- Si votre domaine utilise un type d'instance T2 ou T3 pour vos nœuds de données, vous ne pouvez pas utiliser le stockage à chaud.

- Si votre index utilise le k-NN (`"index.knn":true`) approximatif, vous pouvez le déplacer vers le stockage à chaud à partir de la version 2.17 et des versions ultérieures. Les domaines des versions antérieures à 2.17 peuvent être mis à niveau vers la version 2.17 pour utiliser cette fonctionnalité, mais les index KNN créés sur des versions antérieures à 2.x ne peuvent pas migrer vers cette version. UltraWarm
- Si le domaine utilise le contrôle précis des accès, les utilisateurs doivent être mappés au `ultrawarm_manager` rôle dans OpenSearch Dashboards pour effectuer des appels d'API. UltraWarm

> **Note**
>
> Le `ultrawarm_manager` rôle peut ne pas être défini sur certains domaines de OpenSearch service préexistants. Si le rôle n'est pas disponible dans Dashboards, vous devez le créer manuellement.

Configurer des autorisations

Si vous l'activez UltraWarm sur un domaine de OpenSearch service préexistant, le `ultrawarm_manager` rôle peut ne pas y être défini. Les utilisateurs non-administrateurs doivent être mappés à ce rôle pour gérer les index à chaud des domaines utilisant le contrôle précis des accès. Pour créer manuellement le rôle `ultrawarm_manager`, procédez comme suit :

1. Dans les OpenSearch tableaux de bord, accédez à Sécurité, puis sélectionnez Autorisations.
2. Choisissez Create action group (Créer un groupe d'actions) et configurez les groupes suivants :

Nom du groupe	Autorisations
`ultrawarm_cluster`	• `cluster:admin/ultrawarm/migration/list` • `cluster:monitor/nodes/stats`
`ultrawarm_index_read`	• `indices:admin/ultrawarm/migration/get` • `indices:admin/get`
`ultrawarm_index_write`	• `indices:admin/ultrawarm/migration/warm` • `indices:admin/ultrawarm/migration/hot`

Nom du groupe	Autorisations
	• `indices:monitor/stats` • `indices:admin/ultrawarm/migration/cancel`

3. Choisissez Roles (Rôles), puis Create role (Créer un rôle).
4. Nommez le rôle ultrawarm_manager.
5. Pour Cluster permissions (Autorisations de cluster), sélectionnez `ultrawarm_cluster` et`cluster_monitor`.
6. Pour Index, saisissez *.
7. Pour Index permissions (Autorisations d'index), sélectionnez `ultrawarm_index_read`, `ultrawarm_index_write` et `indices_monitor`.
8. Choisissez Créer.
9. Après avoir créé le rôle, mappez-le à n'importe quel rôle utilisateur ou backend qui gèrera les UltraWarm index.

UltraWarm exigences relatives aux performances et exigences relatives aux performances

Comme indiqué dansthe section called “Calcul des exigences de stockage”, les données du stockage hot font l'objet d'une surcharge importante : les réplicas, l'espace réservé Linux et l'espace OpenSearch Service réservé. Par exemple, une partition principale de 20 Gio avec une partition de réplica nécessite environ 58 Gio de stockage hot.

Comme il utilise Amazon S3 UltraWarm , aucune de ces contraintes. Lors du calcul des exigences UltraWarm de stockage, vous ne prenez en compte que la taille des partitions principales. La durabilité des données dans S3 élimine le besoin de réplicas et S3 fait fi de toutes les considérations relatives au système d'exploitation ou au service. Cette même partition de 20 Gio nécessite 20 Gio de stockage à chaud (warm). Si vous provisionnez une instance `ultrawarm1.large.search`, vous pouvez utiliser les 20 Tio de son stockage maximale pour les partitions primaires. Reportez-vous à la section the section called “UltraWarm quotas de stockage” pour obtenir un résumé des types d'instance et la quantité maximale de stockage possible selon chaque cas.

Avec UltraWarm, nous recommandons toujours une taille de partition maximale de 50 GiB. Le nombre de cœurs CPU et quantité de RAM allouée à chaque type d' UltraWarm instance vous donne une idée du nombre de partitions qu'ils peuvent rechercher simultanément. Notez que même

si seules les partitions principales comptent pour le UltraWarm stockage dans S3, OpenSearch Dashboards indiquent `_cat/indices` toujours la taille de l' UltraWarm index en tant que total de toutes les partitions primaires et des partitions de réplica.

Par exemple, chaque instance `ultrawarm1.medium.search` deux cœurs CPU et peut traiter jusqu'à 1,5 Tio de stockage sur S3. Deux de ces instances présentent un stockage combiné de 3 Tio, ce qui correspond à environ 62 partitions si chaque partition est de 50 Gio. Si une requête adressée au cluster ne recherche que quatre de ces partitions, les performances peuvent être excellentes. Si la requête est importante et qu'elle recherche tous les 62 cœurs, les quatre cœurs de CPU peuvent avoir du mal à effectuer l'opération. Surveillez les `WarmJVMMemoryPressure` UltraWarm indicateurs `WarmCPUUtilization` et les indicateurs pour comprendre comment les instances gèrent vos charges de travail.

Si vos recherches sont volumineuses ou fréquentes, envisagez de conserver les index du stockage hot. Comme pour toute autre OpenSearch charge de travail, l'étape la plus importante pour déterminer si elle UltraWarm répond à vos besoins consiste à effectuer des tests clients représentatifs à l'aide d'un ensemble de données réalistes.

UltraWarm tarification

Avec le stockage hot, vous payez pour ce que vous provisionnez. Certaines instances nécessitent un volume Amazon EBS attaché, tandis que d'autres incluent un magasin d'instances. Que ce stockage soit vide ou plein, vous payez le même prix.

Avec le UltraWarm stockage, vous ne payez que ce que vous utilisez. Une instance `ultrawarm1.large.search` peut traiter jusqu'à 20 Tio de stockage sur S3, mais si vous stockez seulement 1 Tio de données, vous n'êtes facturé que pour 1 Tio de données. Comme pour tous les autres types de nœuds, un tarif horaire s'applique à chaque UltraWarm nœud. Pour de plus amples informations, veuillez consulter the section called “Tarification”.

Activant UltraWarm

La console est le moyen le plus simple de créer un domaine qui utilise un stockage à chaud (warm). Lors de la création du domaine, choisissez Enable warm data nodes (Activer les nœuds de données à chaud) et le nombre de nœuds à chaud (warm) souhaités. Le même processus de base fonctionne sur les domaines existants, à condition qu'ils répondent aux prérequis. Même après que l'état du domaine passe de Traitement à Actif, il UltraWarm peut ne pas être disponible pendant plusieurs heures.

Lorsque vous utilisez un domaine Multi-AZ avec veille, le nombre de nœuds chauds doit être un multiple du nombre de zones de disponibilité. Pour de plus amples informations, veuillez consulter the section called “Multi-AZ avec mode veille”.

Vous pouvez également utiliser l'API de configuration AWS CLIor pour activer UltraWarm, en particulierWarmEnabled, les WarmType optionsWarmCount, et dansClusterConfig.

> **Note**
>
> Les domaines prennent en charge un nombre maximum de nœuds à chaud. Pour plus d'informations, consultez the section called “Quotas”.

Exemple de commande CLI

La AWS CLI commande suivante crée un domaine constitué de trois nœuds de données, de trois nœuds principaux dédiés, de six nœuds à chaud et sur lequel le stockage à froid, avec contrôle précis des accès activé :

```
aws opensearch create-domain \
  --domain-name my-domain \
  --engine-version Opensearch_1.0 \
  --cluster-config
 InstanceCount=3,InstanceType=r6g.large.search,DedicatedMasterEnabled=true,DedicatedMasterType=
 \
  --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=11 \
  --node-to-node-encryption-options Enabled=true \
  --encryption-at-rest-options Enabled=true \
  --domain-endpoint-options EnforceHTTPS=true,TLSSecurityPolicy=Policy-Min-
TLS-1-2-2019-07 \
  --advanced-security-options
 Enabled=true,InternalUserDatabaseEnabled=true,MasterUserOptions='{MasterUserName=master-
user,MasterUserPassword=master-password}' \
  --access-policies '{"Version":"2012-10-17","Statement":
[{"Effect":"Allow","Principal":{"AWS":["123456789012"]},"Action":
["es:*"],"Resource":"arn:aws:es:us-west-1:123456789012:domain/my-domain/*"}]}' \
  --region us-east-1
```

Pour plus d'informations, consultez le Guide de référence des commandes AWS CLI.

Exemple de demande d'API de configuration

La demande suivante adressée à l'API de configuration crée un domaine avec trois nœuds de données, trois nœuds maîtres dédiés et six nœuds à chaud avec un contrôle précis des accès activé et une stratégie d'accès restrictive :

```
POST https://es.us-east-2.amazonaws.com/2021-01-01/opensearch/domain
{
  "ClusterConfig": {
    "InstanceCount": 3,
    "InstanceType": "r6g.large.search",
    "DedicatedMasterEnabled": true,
    "DedicatedMasterType": "r6g.large.search",
    "DedicatedMasterCount": 3,
    "ZoneAwarenessEnabled": true,
    "ZoneAwarenessConfig": {
      "AvailabilityZoneCount": 3
    },
    "WarmEnabled": true,
    "WarmCount": 6,
    "WarmType": "ultrawarm1.medium.search"
  },
  "EBSOptions": {
    "EBSEnabled": true,
    "VolumeType": "gp2",
    "VolumeSize": 11
  },
  "EncryptionAtRestOptions": {
    "Enabled": true
  },
  "NodeToNodeEncryptionOptions": {
    "Enabled": true
  },
  "DomainEndpointOptions": {
    "EnforceHTTPS": true,
    "TLSSecurityPolicy": "Policy-Min-TLS-1-2-2019-07"
  },
   "AdvancedSecurityOptions": {
    "Enabled": true,
    "InternalUserDatabaseEnabled": true,
    "MasterUserOptions": {
      "MasterUserName": "master-user",
      "MasterUserPassword": "master-password"
```

```
    }
  },
  "EngineVersion": "Opensearch_1.0",
  "DomainName": "my-domain",
  "AccessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":[\"123456789012\"]},\"Action\":[\"es:*\"],\"Resource\":
\"arn:aws:es:us-east-1:123456789012:domain/my-domain/*\"}]}"
}
```

Pour plus d'informations, consultez la référence d'API Amazon OpenSearch Service.

Migration des index vers le stockage UltraWarm

Si vous avez terminé d'écrire dans un index et que vous n'avez plus besoin des performances de recherche les plus rapides possibles, migrez-le de hot vers UltraWarm :

```
POST _ultrawarm/migration/my-index/_warm
```

Vérifiez ensuite l'état de la migration :

```
GET _ultrawarm/migration/my-index/_status

{
  "migration_status": {
    "index": "my-index",
    "state": "RUNNING_SHARD_RELOCATION",
    "migration_type": "HOT_TO_WARM",
    "shard_level_status": {
      "running": 0,
      "total": 5,
      "pending": 3,
      "failed": 0,
      "succeeded": 2
    }
  }
}
```

L'état de l'index doit être vert pour effectuer une migration. Si vous migrez plusieurs index en succession rapide, vous pouvez obtenir un résumé de toutes les migrations en texte clair, similaire à l'API `_cat` :

```
GET _ultrawarm/migration/_status?v
```

```
index    migration_type state
my-index HOT_TO_WARM    RUNNING_SHARD_RELOCATION
```

OpenSearch Le service migre un index à la fois vers UltraWarm. Vous pouvez avoir jusqu'à 200 migrations dans la file d'attente. Toute requête dépassant la limite sera rejetée. Pour vérifier le nombre actuel de migrations dans la file d'attente, consultez la métrique `HotToWarmMigrationQueueSize`. Les index restent disponibles tout au long du processus de migration, sans temps d'arrêt.

Le processus de migration comporte les états suivants :

```
PENDING_INCREMENTAL_SNAPSHOT
RUNNING_INCREMENTAL_SNAPSHOT
FAILED_INCREMENTAL_SNAPSHOT
PENDING_FORCE_MERGE
RUNNING_FORCE_MERGE
FAILED_FORCE_MERGE
PENDING_FULL_SNAPSHOT
RUNNING_FULL_SNAPSHOT
FAILED_FULL_SNAPSHOT
PENDING_SHARD_RELOCATION
RUNNING_SHARD_RELOCATION
FINISHED_SHARD_RELOCATION
```

Comme l'indiquent ces états, les migrations peuvent échouer pendant les instantanés, les relocations de partition ou les fusions forcées. Les échecs lors des instantanés ou de la relocalisation de partitions sont généralement dus à des défaillances de nœud ou à des problèmes de connectivité S3. Le manque d'espace disque est généralement la cause sous-jacente des échecs de fusion forcée.

Une fois la migration terminée, la même demande `_status` renvoie une erreur. Si vous vérifiez l'index à ce moment-là, vous pouvez voir certains paramètres qui sont propres aux index hot :

```
GET my-index/_settings

{
  "my-index": {
    "settings": {
      "index": {
        "refresh_interval": "-1",
        "auto_expand_replicas": "false",
```

```
        "provided_name": "my-index",
        "creation_date": "1599241458998",
        "unassigned": {
          "node_left": {
            "delayed_timeout": "5m"
          }
        },
        "number_of_replicas": "1",
        "uuid": "GswyCdR0RSq0SJYmzsIpiw",
        "version": {
          "created": "7070099"
        },
        "routing": {
          "allocation": {
            "require": {
              "box_type": "warm"
            }
          }
        },
        "number_of_shards": "5",
        "merge": {
          "policy": {
            "max_merge_at_once_explicit": "50"
          }
        }
      }
    }
  }
}
```

- `number_of_replicas`, dans ce cas, correspond au nombre de réplicas passifs qui ne consomment pas d'espace disque.
- `routing.allocation.require.box_type` spécifie que l'index doit utiliser des nœuds à chaud plutôt que des nœuds de données standard.
- `merge.policy.max_merge_at_once_explicit` spécifie le nombre de segments à fusionner simultanément pendant la migration.

Les index du stockage warm sont en lecture seule, sauf si vous les rebasculez dans le stockage hot, ce qui rend UltraWarm particulièrement adapté aux données immuables, telles que les journaux. Vous pouvez interroger les index et les supprimer, mais vous ne pouvez pas ajouter, mettre à jour

ou supprimer des documents individuels. Si vous essayez, il se peut que vous receviez l'erreur suivante :

```
{
  "error" : {
    "root_cause" : [
      {
        "type" : "cluster_block_exception",
        "reason" : "index [indexname] blocked by: [TOO_MANY_REQUESTS/12/disk usage
 exceeded flood-stage watermark, index has read-only-allow-delete block];"
      }
    ],
    "type" : "cluster_block_exception",
    "reason" : "index [indexname] blocked by: [TOO_MANY_REQUESTS/12/disk usage exceeded
 flood-stage watermark, index has read-only-allow-delete block];"
  },
  "status" : 429
}
```

Automatisation des migrations

Nous vous recommandons d'utiliser the section called “Gestion d'états des index” pour automatiser le processus de migration une fois qu'un index atteint un âge défini ou remplit d'autres conditions. Consultez l'exemple de politique qui illustre ce flux de travail.

Réglage des migrations

Les migrations d'index vers le UltraWarm stockage nécessitent une fusion forcée. Chaque OpenSearch index est composé d'un certain nombre de partitions, et chaque partition d'un certain nombre de segments Lucene. L'opération de fusion forcée purge les documents marqués pour suppression et conserve de l'espace disque. Par défaut, UltraWarm fusionne les index en un seul segment, à l'exception des indices kNN, où une valeur par défaut de 20 est utilisée.

Vous pouvez modifier cette valeur à hauteur de 1 000 segments à l'aide du paramètre `index.ultrawarm.migration.force_merge.max_num_segments`. Des valeurs plus élevées accélèrent le processus de migration, mais augmentent la latence de requête de l'index à chaud une fois la migration terminée. Pour modifier le paramètre, faites la demande suivante :

```
PUT my-index/_settings
{
  "index": {
```

```
    "ultrawarm": {
      "migration": {
        "force_merge": {
          "max_num_segments": 1
        }
      }
    }
  }
}
```

Pour vérifier combien de temps dure cette étape du processus de migration, consultez la métrique `HotToWarmMigrationForceMergeLatency`.

Annulation des migrations

UltraWarm gère les migrations de manière séquentielle, dans une file d'attente. Si une migration se trouve dans la file d'attente mais n'a pas encore démarré, vous pouvez la supprimer à l'aide de la demande suivante :

```
POST _ultrawarm/migration/_cancel/my-index
```

Si votre domaine utilise le contrôle précis des accès, vous devez disposer de l'autorisation `indices:admin/ultrawarm/migration/cancel` pour effectuer cette demande.

Liste des index hot et warm

UltraWarm ajoute deux options supplémentaires, similaires à `_all`, pour aider à gérer les index hot et warm. Pour obtenir la liste de tous les index hot ou warm, faites les demandes suivantes :

```
GET _warm
GET _hot
```

Vous pouvez utiliser ces options dans d'autres demandes qui spécifient des index, par exemple :

```
_cat/indices/_warm
_cluster/state/_all/_hot
```

Rebasculement d'index à chaud vers le stockage hot

Si vous devez écrire à nouveau dans un index, migrez-le vers le stockage hot :

```
POST _ultrawarm/migration/my-index/_hot
```

Vous pouvez effectuer jusqu'à 10 migrations en file d'attente du stockage à chaud vers le stockage hot à chaud à chaud à la fois. OpenSearch Le service traite les demandes de migration une par une, dans l'ordre dans lequel elles ont été mises en file d'attente. Pour vérifier le nombre actuel, consultez la métrique `WarmToHotMigrationQueueSize`.

Une fois la migration terminée, vérifiez les paramètres d'index pour vous assurer qu'ils répondent à vos besoins. Rebasculement des index dans le stockage hot avec un réplica.

Restauration des index à chaud à partir d'instantanés

En plus du référentiel standard pour les instantanés automatisés, UltraWarm ajoute un second référentiel pour les index à chaud,. `cs-ultrawarm` Chaque instantané de ce référentiel contient un seul index. Si vous supprimez un index chaud, son instantané reste dans le référentiel `cs-ultrawarm` pendant 14 jours, comme pour tout autre instantané automatique.

Lorsque vous restaurez un instantané à partir de `cs-ultrawarm`, il rebascule dans le stockage à chaud (warm), et non dans le stockage hot. Les instantanés des référentiels `cs-automated` et `cs-automated-enc` rebasculent dans le stockage hot.

Pour restaurer un UltraWarm instantané dans un espace de stockage chaud

1. Identifiez le dernier instantané qui contient l'index à restaurer :

```
GET _snapshot/cs-ultrawarm/_all?verbose=false

{
  "snapshots": [{
    "snapshot": "snapshot-name",
    "version": "1.0",
    "indices": [
      "my-index"
    ]
  }]
}
```

> **Note**
>
> Par défaut, l'`GET _snapshot/<repo>`opération affiche des informations détaillées telles que l'heure de début, l'heure de fin et la durée de chaque instantané d'un référentiel. L'`GET _snapshot/<repo>`opération extrait les informations des fichiers de chaque instantané contenu dans un référentiel. Si vous n'avez pas besoin de l'heure de début, de l'heure de fin et de la durée et que vous avez uniquement besoin du nom et des informations d'index d'un instantané, nous vous recommandons d'utiliser le `verbose=false` paramètre lors de la liste des instantanés afin de minimiser le temps de traitement et d'éviter les délais d'expiration.

2. Si l'index existe déjà, supprimez-le :

```
DELETE my-index
```

   Si vous ne souhaitez pas supprimer l'index, renvoyez-le dans le stockage hot et réindexez-le.

3. Restaurer l'instantané :

```
POST _snapshot/cs-ultrawarm/snapshot-name/_restore
```

   UltraWarm ignore tous les paramètres d'index que vous spécifiez dans cette demande de restauration, mais vous pouvez spécifier des options telles que `rename_pattern` et`rename_replacement`. Pour un résumé des options de restauration d' OpenSearch instantanés, consultez la OpenSearch documentation.

Instantanés manuels des index warm

Vous pouvez prendre des instantanés manuels des index warm, mais nous vous le déconseillons. Le référentiel `cs-ultrawarm` automatisé contient déjà un instantané pour chaque index à chaud pris lors de la migration, sans frais supplémentaires.

Par défaut, OpenSearch Service n'inclut pas d'index à chaud dans les instantanés manuels. Par exemple, l'appel suivant inclut uniquement des index hot :

```
PUT _snapshot/my-repository/my-snapshot
```

Si vous choisissez de prendre des instantanés manuels d'index warm, plusieurs considérations importantes s'appliquent.

- Vous ne pouvez pas mélanger les index hot et warm. Par exemple, la demande suivante échoue :

```
PUT _snapshot/my-repository/my-snapshot
{
  "indices": "warm-index-1,hot-index-1",
  "include_global_state": false
}
```

  Si elles combinent des index hot et warm, les instructions avec caractère générique (*) échouent également.

- Vous ne pouvez inclure qu'un index à chaud par instantané. Par exemple, la demande suivante échoue :

```
PUT _snapshot/my-repository/my-snapshot
{
  "indices": "warm-index-1,warm-index-2,other-warm-indices-*",
  "include_global_state": false
}
```

  Cette demande a abouti :

```
PUT _snapshot/my-repository/my-snapshot
{
  "indices": "warm-index-1",
  "include_global_state": false
}
```

- Les instantanés manuels sont toujours restaurés dans le stockage hot, même s'ils incluaient à l'origine un index à chaud.

Migration d'index à chaud vers le stockage à froid

Si vous disposez de données UltraWarm que vous interrogez rarement, envisagez de les migrer vers le stockage à froid. Le stockage à froid est destiné aux données auxquelles vous n'accédez qu'occasionnellement ou qui ne sont plus utilisées. Vous ne pouvez pas lire ou écrire dans des index cold, mais vous pouvez les migrer vers un stockage warm, sans frais, chaque fois que vous avez

besoin de les interroger. Pour obtenir des instructions, consultez la section Migration des index vers le stockage à froid.

Bonnes pratiques relatives aux index KNN

- Le niveau Ultrawarm/Cold est disponible pour tous les types de moteurs à indice KNN. Nous le recommandons pour les index KNN utilisant le moteur Lucene et la recherche vectorielle optimisée pour les disques, qui ne nécessitent pas de charger complètement les données du graphe dans une mémoire externe. Lorsque vous l'utilisez avec des moteurs en mémoire natifs tels que FAISS et NMSLIB, vous devez tenir compte de la taille du graphe de fragments sur lequel la recherche sera activement effectuée et provisionner les UltraWarm instances, de préférence du type instance, en conséquence. `uw.large` Par exemple, si les clients ont configuré deux `uw.large` instances, chacune d'entre elles disposera d'environ `knn.memory.circuit_breaker.limit * 61` GiB de mémoire externe disponible. Vous obtenez des performances optimales si toutes vos requêtes chaleureuses ciblent des fragments dont la taille cumulée du graphique ne dépasse pas la mémoire hors segment disponible. La latence est affectée si la mémoire disponible est inférieure à celle nécessaire pour charger le graphique en raison d'expulsions et de l'attente de la disponibilité de mémoire hors tas. C'est pourquoi nous ne recommandons pas d'utiliser des `uw.medium` instances pour les cas d'utilisation impliquant l'utilisation de moteurs en mémoire ou pour les cas de débit de recherche plus élevé, quels que soient les moteurs.
- Les index KNN vers lesquels la migration est effectuée ne UltraWarm seront pas fusionnés de force en un seul segment. Cela évite tout impact sur les nœuds chauds et chauds qui rencontrent des problèmes d'OOM en raison de la taille du graphe devenue trop importante pour les moteurs en mémoire. En raison de l'augmentation du nombre de segments par partition, cela peut entraîner une consommation plus importante d'espace de cache local et permettre à un moins grand nombre d'index de migrer vers le niveau chaud. Vous pouvez choisir de fusionner de force les index vers un seul segment en utilisant le paramètre existant et en le remplaçant avant de migrer les index vers le niveau chaud. Pour de plus amples informations, veuillez consulter the section called “Réglage des migrations”.
- Si vous avez un cas d'utilisation dans lequel les index sont rarement recherchés et ne répondent pas à une charge de travail sensible à la latence, vous pouvez choisir de migrer ces index vers le niveau. UltraWarm Cela vous aidera à réduire le nombre d'instances de calcul de niveau élevé et à laisser le UltraWarm niveau de calcul gérer les requêtes sur des index de faible priorité. Cela peut également aider à isoler les ressources consommées entre les requêtes des index de faible priorité et de haute priorité afin qu'elles n'aient pas d'impact mutuel.

Désactivation UltraWarm

La console constitue le moyen le plus simple de le désactiver UltraWarm. Choisissez le domaine, Actions, et Edit cluster configuration (Modifier la configuration de cluster). Désélectionnez Activer les nœuds de données chauds et choisissez Enregistrer les modifications. Vous pouvez également utiliser l'option `WarmEnabled` dans l' AWS CLI et dans l'API de configuration.

Avant de procéder à la désactivation UltraWarm, vous devez supprimer tous les index à chaud ou les migrer à nouveau vers le stockage hot. Une fois que le stockage warm est vide, attendez cinq minutes avant de tenter de le désactiver UltraWarm.

Stockage à froid pour Amazon OpenSearch Service

Le stockage à froid vous permet de stocker n'importe quelle quantité de données rarement consultées ou historiques sur votre domaine Amazon OpenSearch Service et de les analyser à la demande, à un coût inférieur à celui des autres niveaux de stockage. Le stockage à froid vous conviendra si vous devez effectuer des recherches périodiques ou des analyses sur vos anciennes données. Parmi les exemples pratiques de données stockables à froid figurent les journaux rarement consultés, les données à conserver pour satisfaire des exigences de conformité ou les journaux qui ont une valeur historique.

Comme le UltraWarmstockage, le stockage à froid repose sur Amazon S3. Lorsque vous devez interroger des données froides, vous pouvez les attacher de manière sélective à des UltraWarm nœuds existants. Vous pouvez gérer la migration et le cycle de vie de vos données froides manuellement ou à l'aide de politiques ISM (Index State Management).

Rubriques

- Prérequis
- Stockage à froid : exigences et considérations relatives aux performances
- Tarification du stockage à froid
- Activation du stockage à froid
- Gestion des index froids dans Dashboards OpenSearch
- Migration des index vers le stockage à froid
- Automatisation des migrations vers le stockage à froid
- Annulation des migrations vers le stockage à froid
- Liste des index froids

- Migration d'index à froid vers le stockage à chaud
- Restauration des index à froid à partir d'instantanés
- Annulation des migrations du stockage à froid vers le stockage à chaud
- Mise à jour des métadonnées des index froids
- Suppression d'index froids
- Désactivation du stockage à froid

Prérequis

Les prérequis suivants s'appliquent au stockage à froid :

- Le stockage à froid requiert OpenSearch Elasticsearch 7.9 ou une version ultérieure.
- Pour activer le stockage à froid sur un domaine de OpenSearch service, vous devez également activer le stockage à chaud sur le même domaine.
- Pour utiliser le stockage à froid, les domaines doivent disposer de nœuds principaux dédiés.
- Si votre domaine utilise un type d'instance T2 ou T3 pour vos nœuds de données, vous ne pouvez pas utiliser le stockage à froid.
- Si votre index utilise le k-NN (`"index.knn":true`) approximatif, vous pouvez le déplacer vers le stockage à froid à partir de la version 2.17 et versions ultérieures. Les domaines des versions antérieures à 2.17 peuvent passer à la version 2.17 pour utiliser cette fonctionnalité, mais les index KNN créés sur des versions antérieures à 2.x ne peuvent pas migrer vers Cold.
- Si le domaine utilise le contrôle précis des accès, les utilisateurs non administrateurs doivent être mappés au `cold_manager` rôle dans OpenSearch Dashboards afin de gérer les index froids.

> **Note**
>
> Le `cold_manager` rôle peut ne pas exister sur certains domaines de OpenSearch service préexistants. Si le rôle n'est pas disponible dans Dashboards, vous devez le créer manuellement.

Configurer des autorisations

Si vous activez le stockage à froid sur un domaine OpenSearch Service préexistant, le `cold_manager` rôle peut ne pas y être défini. Si le domaine utilise le contrôle précis des accès, les

utilisateurs non administrateurs doivent être mappés à ce rôle pour gérer les index froids. Pour créer manuellement le rôle `cold_manager`, procédez comme suit :

1. Dans les OpenSearch tableaux de bord, accédez à Sécurité, puis sélectionnez Autorisations.
2. Choisissez Create action group (Créer un groupe d'actions) et configurez les groupes suivants :

Nom du groupe	Autorisations
`cold_cluster`	• `cluster:monitor/nodes/stats` • `cluster:admin/ultrawarm*` • `cluster:admin/cold/*`
`cold_index`	• `indices:monitor/stats` • `indices:data/read/minmax` • `indices:admin/ultrawarm/migration/get` • `indices:admin/ultrawarm/migration/cancel`

3. Choisissez Roles (Rôles), puis Create role (Créer un rôle).
4. Nommez le rôle cold_manager.
5. Dans le champ Cluster permissions (Autorisations de cluster), choisissez le groupe `cold_cluster` que vous avez créé.
6. Dans le champ Index, saisissez `*`.
7. Dans le champ Index permissions (Autorisations d'index), choisissez le groupe `cold_index` que vous avez créé.
8. Choisissez Créer.
9. Après avoir créé le rôle, mappez-le à n'importe quel rôle utilisateur ou backend qui gère les index froids.

Stockage à froid : exigences et considérations relatives aux performances

Le stockage à froid utilise Amazon S3, il n'entraîne aucun des frais liés au stockage à chaud (réplicas, espace réservé Amazon S3, etc. OpenSearch). Le stockage à froid ne nécessite aucun type d'instance spécifique, car aucune capacité de calcul n'y est attachée. Vous pouvez stocker n'importe quelle quantité de données dans le stockage à froid. Contrôlez la

`ColdStorageSpaceUtilization` métrique sur Amazon CloudWatch pour voir la quantité d'espace de stockage à froid que vous utilisez.

Tarification du stockage à froid

Comme pour le UltraWarm stockage, avec le stockage à froid, vous ne payez que le stockage des données. Il n'y a pas de frais de calcul liés aux données froides et rien ne vous est facturé en l'absence de de données dans le stockage à froid.

Vous ne payez pas de frais de transfert lorsque vous déplacez des données entre le stockage à froid et le stockage à chaud. Pendant la migration d'index entre le stockage à chaud et le stockage à froid, vous continuez à ne payer qu'un seul exemplaire de l'index. Une fois la migration terminée, l'index est facturé en fonction du niveau de stockage vers lequel il a été migré. Pour plus d'informations sur la tarification du stockage à froid, consultez la tarification d'Amazon OpenSearch Service.

Activation du stockage à froid

La console est le moyen le plus simple de créer un domaine qui utilise le stockage à froid. Lorsque vous créez le domaine, choisissez d'abord Enable warm data nodes, car vous devez activer le stockage à chaud sur le même domaine. Choisissez ensuite Activer le stockage à froid.

Le même processus s'applique aux domaines existants, à condition de satisfaire les prérequis. Une fois l'état du domaine passé de Traitement en cours à Actif, le stockage à froid peut malgré tout rester indisponible pendant plusieurs heures.

Vous pouvez également activer le stockage à froid à l'aide de l'interface AWS CLI ou de l'API de configuration.

Exemple de commande de l'interface CLI

La AWS CLI commande suivante crée un domaine constitué de trois nœuds de données et de trois nœuds principaux dédiés, et sur lequel le stockage à froid et le contrôle précis des accès sont activés :

```
aws opensearch create-domain \
  --domain-name my-domain \
  --engine-version Opensearch_1.0 \
  --cluster-
config ColdStorageOptions={Enabled=true},WarmEnabled=true,WarmCount=4,WarmType=ultrawarm1.mediu
 \
  --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=11 \
```

```
  --node-to-node-encryption-options Enabled=true \
  --encryption-at-rest-options Enabled=true \
  --domain-endpoint-options EnforceHTTPS=true,TLSSecurityPolicy=Policy-Min-
TLS-1-2-2019-07 \
  --advanced-security-options
 Enabled=true,InternalUserDatabaseEnabled=true,MasterUserOptions='{MasterUserName=master-
user,MasterUserPassword=master-password}' \
  --region us-east-2
```

Pour plus d'informations, consultez le Guide de référence des commandes AWS CLI.

Exemple de demande d'API de configuration

La requête suivante adressée à l'API de configuration crée un domaine constitué de trois nœuds de données et de trois nœuds principaux dédiés, et sur lequel le stockage à froid et le contrôle précis des accès sont activés :

```
POST https://es.us-east-2.amazonaws.com/2021-01-01/opensearch/domain
{
  "ClusterConfig": {
    "InstanceCount": 3,
    "InstanceType": "r6g.large.search",
    "DedicatedMasterEnabled": true,
    "DedicatedMasterType": "r6g.large.search",
    "DedicatedMasterCount": 3,
    "ZoneAwarenessEnabled": true,
    "ZoneAwarenessConfig": {
      "AvailabilityZoneCount": 3
     },
    "WarmEnabled": true,
    "WarmCount": 4,
    "WarmType": "ultrawarm1.medium.search",
    "ColdStorageOptions": {
       "Enabled": true
     }
  },
  "EBSOptions": {
    "EBSEnabled": true,
    "VolumeType": "gp2",
    "VolumeSize": 11
  },
  "EncryptionAtRestOptions": {
    "Enabled": true
```

```
  },
  "NodeToNodeEncryptionOptions": {
    "Enabled": true
  },
  "DomainEndpointOptions": {
    "EnforceHTTPS": true,
    "TLSSecurityPolicy": "Policy-Min-TLS-1-2-2019-07"
  },
   "AdvancedSecurityOptions": {
    "Enabled": true,
    "InternalUserDatabaseEnabled": true,
    "MasterUserOptions": {
      "MasterUserName": "master-user",
      "MasterUserPassword": "master-password"
    }
  },
  "EngineVersion": "Opensearch_1.0",
  "DomainName": "my-domain"
}
```

Pour plus d'informations, consultez la référence d'Amazon OpenSearch Service API.

Gestion des index froids dans Dashboards OpenSearch

Vous pouvez gérer les index chauds, tièdes et froids avec l'interface Dashboards existante de votre domaine OpenSearch Service. Dashboards vous permet de migrer des index entre le stockage à chaud et le stockage à froid, et de surveiller le statut de migration des index, sans utiliser l'interface CLI ni l'API de configuration. Pour de plus amples informations, veuillez consulter la section Gestion des index dans les OpenSearch tableaux de bord.

Migration des index vers le stockage à froid

Lorsque vous migrez des index vers le stockage à froid, vous indiquez une plage de temps pour les données afin de faciliter la découverte. Vous pouvez sélectionner un champ d'horodatage basé sur les données de votre index, fournir manuellement un horodatage de début et de fin, ou choisir de ne pas en spécifier.

Paramètre	Valeur prise en charge	Description
`timestamp_field`	Champ de date/heure du mappage d'index.	Les valeurs minimales et maximales du champ fourni

Paramètre	Valeur prise en charge	Description
		sont calculées et stockées en tant que métadonnées `start_time` et `end_time` pour l'index froid.
`start_time` et `end_time`	Un des formats suivants : • strict_date_optional_time. Par exemple : `yyyy-MM-dd'T'HH:mm:ss.SSSZ` ou `yyyy-MM-dd` • Époque, en millisecondes	Les valeurs fournies sont stockées en tant que métadonnées `start_time` et `end_time` pour l'index froid.

Si vous ne souhaitez pas spécifier d'horodatage, ajoutez `?ignore=timestamp` à la requête.

La requête suivante migre un index chaud vers le stockage à froid, et fournit les heures de début et de fin des données de cet index :

```
POST _ultrawarm/migration/my-index/_cold
  {
    "start_time": "2020-03-09",
    "end_time": "2020-03-09T23:00:00Z"
  }
```

Vérifiez ensuite l'état de la migration :

```
GET _ultrawarm/migration/my-index/_status

{
  "migration_status": {
    "index": "my-index",
    "state": "RUNNING_METADATA_RELOCATION",
    "migration_type": "WARM_TO_COLD"
  }
}
```

OpenSearch Le service migre un index à la fois vers le stockage à froid. Vous pouvez avoir jusqu'à 100 migrations dans la file d'attente. Toute requête dépassant la limite sera rejetée. Pour vérifier le nombre actuel de migrations dans la file d'attente, consultez la métrique `WarmToColdMigrationQueueSize`. Le processus de migration comporte les états suivants :

```
ACCEPTED_COLD_MIGRATION - Migration request is accepted and queued.
RUNNING_METADATA_MIGRATION - The migration request was selected for execution and
 metadata is migrating to cold storage.
FAILED_METADATA_MIGRATION - The attempt to add index metadata has failed and all
 retries are exhausted.
PENDING_INDEX_DETACH - Index metadata migration to cold storage is completed. Preparing
 to detach the warm index state from the local cluster.
RUNNING_INDEX_DETACH - Local warm index state from the cluster is being removed. Upon
 success, the migration request will be completed.
FAILED_INDEX_DETACH - The index detach process failed and all retries are exhausted.
```

Automatisation des migrations vers le stockage à froid

Nous vous recommandons d'utiliser ISM (Index State Management) pour automatiser le processus de migration une fois qu'un index atteint un âge défini ou remplit d'autres conditions. Consultez l'exemple de politique, qui montre comment migrer automatiquement les index du stockage à chaud vers le stockage UltraWarm à froid.

> **Note**
>
> Un `timestamp_field` explicite est nécessaire pour déplacer les index vers le stockage à froid à l'aide d'une politique ISM (Index State Management).

Annulation des migrations vers le stockage à froid

En cas de mise en file d'attente ou d'échec d'une migration vers le stockage à froid, vous pouvez annuler la migration à l'aide de la requête suivante :

```
POST _ultrawarm/migration/_cancel/my-index

{
  "acknowledged" : true
}
```

Si votre domaine utilise le contrôle précis des accès, vous devez disposer de l'autorisation `indices:admin/ultrawarm/migration/cancel` pour effectuer cette requête.

Liste des index froids

Avant de lancer une requête, vous pouvez répertorier les index contenus dans le stockage à froid afin de déterminer lesquels migrer UltraWarm pour une analyse plus approfondie. La requête suivante répertorie tous les index froids, triés par nom d'index :

```
GET _cold/indices/_search
```

Exemple de réponse

```
{
  "pagination_id" : "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
  "total_results" : 3,
  "indices" : [
    {
      "index" : "my-index-1",
      "index_cold_uuid" : "hjEoh26mRRCFxRIMdgvLmg",
      "size" : 10339,
      "creation_date" : "2021-06-28T20:23:31.206Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    },
    {
      "index" : "my-index-2",
      "index_cold_uuid" : "0vIS2n-oROmOWDFmwFIgdw",
      "size" : 6068,
      "creation_date" : "2021-07-15T19:41:18.046Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    },
    {
      "index" : "my-index-3",
      "index_cold_uuid" : "EaeXOBodTLiDYcivKsXVLQ",
      "size" : 32403,
      "creation_date" : "2021-07-08T00:12:01.523Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    }
  ]
```

```
}
```

Le filtrage

Vous pouvez filtrer les index froids en fonction d'un modèle d'index basé sur un préfixe et de décalages horaires.

La requête suivante répertorie les index qui correspondent au modèle de préfixe `event-*` :

```
GET _cold/indices/_search
 {
   "filters":{
      "index_pattern": "event-*"
   }
 }
```

Exemple de réponse

```
{
  "pagination_id" : "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
  "total_results" : 1,
  "indices" : [
    {
      "index" : "events-index",
      "index_cold_uuid" : "4eFiab7rRfSvp3slrIsIKA",
      "size" : 32263273,
      "creation_date" : "2021-08-18T18:25:31.845Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    }
  ]
}
```

La requête suivante renvoie les index dont les champs de métadonnées `start_time` et `end_time` sont compris entre `2019-03-01` et `2020-03-01` :

```
GET _cold/indices/_search
{
  "filters": {
    "time_range": {
      "start_time": "2019-03-01",
```

```
      "end_time": "2020-03-01"
    }
  }
}
```

Exemple de réponse

```
{
  "pagination_id" : "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
  "total_results" : 1,
  "indices" : [
    {
      "index" : "my-index",
      "index_cold_uuid" : "4eFiab7rRfSvp3slrIsIKA",
      "size" : 32263273,
      "creation_date" : "2021-08-18T18:25:31.845Z",
      "start_time" : "2019-05-09T00:00Z",
      "end_time" : "2019-09-09T23:00Z"
    }
  ]
}
```

Tri

Vous pouvez trier les index froids par champs de métadonnées, tels que le nom ou la taille de l'index. La requête suivante répertorie tous les index triés par taille dans l'ordre décroissant :

```
GET _cold/indices/_search
 {
 "sort_key": "size:desc"
 }
```

Exemple de réponse

```
{
  "pagination_id" : "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
  "total_results" : 5,
  "indices" : [
    {
      "index" : "my-index-6",
      "index_cold_uuid" : "4eFiab7rRfSvp3slrIsIKA",
```

```
      "size" : 32263273,
      "creation_date" : "2021-08-18T18:25:31.845Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    },
    {
      "index" : "my-index-9",
      "index_cold_uuid" : "mbD3ZRVDRI6ONqgEOsJyUA",
      "size" : 57922,
      "creation_date" : "2021-07-07T23:41:35.640Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    },
    {
      "index" : "my-index-5",
      "index_cold_uuid" : "EaeXOBodTLiDYcivKsXVLQ",
      "size" : 32403,
      "creation_date" : "2021-07-08T00:12:01.523Z",
      "start_time" : "2020-03-09T00:00Z",
      "end_time" : "2020-03-09T23:00Z"
    }
  ]
}
```

Les autres clés de tri valides sont `start_time:asc/desc`, `end_time:asc/desc` et `index_name:asc/desc`.

Pagination

Vous pouvez paginer une liste d'index froids. Configurez le nombre d'index à renvoyer par page à l'aide du paramètre `page_size` (la valeur par défaut est 10). Chaque requête `_search` effectuée sur vos index froids renvoie un `pagination_id` que vous pouvez utiliser pour les appels suivants.

La requête suivante pagine les résultats d'une requête `_search` de vos index froids et affiche les 100 résultats suivants :

```
GET _cold/indices/_search?page_size=100
{
"pagination_id": "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY"
}
```

Migration d'index à froid vers le stockage à chaud

Après avoir réduit votre liste d'index froids avec les critères de filtrage de la section précédente, migrez-les vers l' UltraWarm endroit où vous pourrez interroger les données et les utiliser pour créer des visualisations.

La requête suivante migre deux index froids vers le stockage à chaud :

```
POST _cold/migration/_warm
 {
 "indices": "my-index1,my-index2"
 }


{
  "acknowledged" : true
}
```

Pour vérifier le statut de la migration et récupérer l'ID de migration, envoyez la requête suivante :

```
GET _cold/migration/_status
```

Exemple de réponse

```
{
  "cold_to_warm_migration_status" : [
    {
      "migration_id" : "tyLjXCA-S76zPQbPVHkOKA",
      "indices" : [
        "my-index1,my-index2"
      ],
      "state" : "RUNNING_INDEX_CREATION"
    }
  ]
}
```

Pour obtenir des informations de migration spécifiques à un index, incluez le nom de l'index :

```
GET _cold/migration/my-index/_status
```

Plutôt que de spécifier un index, vous pouvez répertorier les index en fonction de leur statut de migration actuel. Les valeurs valides sont `_failed`, `_accepted` et `_all`.

La commande suivante permet d'obtenir le statut de tous les index en une seule requête de migration :

```
GET _cold/migration/_status?migration_id=my-migration-id
```

Récupérez l'ID de migration à l'aide de la requête de statut. Pour obtenir des informations détaillées sur la migration, ajoutez `&verbose=true`.

Vous pouvez migrer des index du stockage à froid vers le stockage tiède par lots de 10, avec un maximum de 100 requêtes migrées de manière simultanée. Toute requête dépassant la limite sera rejetée. Pour vérifier le nombre actuel de migrations en cours, consultez la métrique `ColdToWarmMigrationQueueSize`. Le processus de migration comporte les états suivants :

```
ACCEPTED_MIGRATION_REQUEST - Migration request is accepted and queued.
RUNNING_INDEX_CREATION - Migration request is picked up for processing and will create
 warm indexes in the cluster.
PENDING_COLD_METADATA_CLEANUP - Warm index is created and the migration service will
 attempt to clean up cold metadata.
RUNNING_COLD_METADATA_CLEANUP - Cleaning up cold metadata from the indexes migrated to
 warm storage.
FAILED_COLD_METADATA_CLEANUP - Failed to clean up metadata in the cold tier.
FAILED_INDEX_CREATION - Failed to create an index in the warm tier.
```

Restauration des index à froid à partir d'instantanés

Si vous devez restaurer un index de froid supprimé, vous pouvez le restaurer au niveau chaud en suivant les instructions de la section, the section called “Restauration des index à chaud à partir d'instantanés” puis en faisant à nouveau migrer l'index vers le niveau froid. Vous ne pouvez pas restaurer un index de froid supprimé directement dans le niveau de froid. OpenSearch Le service conserve les index froids pendant 14 jours après leur suppression.

Annulation des migrations du stockage à froid vers le stockage à chaud

Si une migration d'index du stockage à froid vers le stockage à chaud est mise en file d'attente ou échoue, vous pouvez l'annuler à l'aide de la demande suivante :

```
POST _cold/migration/my-index/_cancel
```

```
{
  "acknowledged" : true
}
```

Pour annuler la migration d'un lot d'index (10 maximum à la fois), spécifiez l'ID de la migration :

```
POST _cold/migration/_cancel?migration_id=my-migration-id

{
  "acknowledged" : true
}
```

Récupérez l'ID de migration à l'aide de la requête de statut.

Mise à jour des métadonnées des index froids

Vous pouvez mettre à jour les champs `start_time` et `end_time` d'un index froid :

```
PATCH _cold/my-index
 {
 "start_time": "2020-01-01",
 "end_time": "2020-02-01"
 }
```

Vous ne pouvez pas mettre à jour le champ `timestamp_field` d'un index dans le stockage à froid.

> **Note**
>
> OpenSearch Dashboards ne prend pas pas pas pas pas en charge la méthode PATCH. Utilisez Curl, Postman ou une autre méthode pour mettre à jour les métadonnées froides.

Suppression d'index froids

Si vous n'utilisez pas de politique ISM, vous pouvez supprimer les index froids manuellement. La demande suivante supprime un index froid :

```
DELETE _cold/my-index

{
```

```
  "acknowledged" : true
}
```

Désactivation du stockage à froid

La console OpenSearch Service constitue le moyen le plus simple de désactiver le stockage à froid. Sélectionnez le domaine et choisissez Actions,Edit cluster configuration (Modifier la configuration de cluster), puis désélectionnez Enable cold storage (Activer le stockage à froid).

Pour utiliser la AWS CLI ou l'API de configuration, sous`ColdStorageOptions`, définissez`"Enabled"="false"`.

Avant de désactiver le stockage à froid, vous devez supprimer tous les index froids ou les migrer à nouveau vers le stockage à chaud, sinon la désactivation échouera.

OpenSearch stockage optimisé pour Amazon OpenSearch Service

La famille d'instances OpenSearch optimisée pour Amazon OpenSearch Service est une solution rentable pour le stockage de gros volumes de données. Un domaine doté d' OR1 instances utilise Amazon Elastic Block Store (Amazon EBS`gp3`) `io1` ou des volumes pour le stockage principal, les données étant copiées de manière synchrone sur Amazon S3 à leur arrivée. Cette structure de stockage fournit un débit d'indexation accru avec une durabilité élevée. La famille d'instances OpenSearch optimisée prend également en charge la restauration automatique des données en cas de panne. Pour plus d'informations sur les options de type d' OR1 instance, consultezthe section called “Types d'instance de la génération actuelle”.

Si vous exécutez de lourdes charges de travail d'analyse opérationnelle liées à l'indexation, telles que l'analyse des journaux, l'observabilité ou l'analyse de sécurité, vous pouvez bénéficier des performances et de l'efficacité de calcul améliorées des instances. OR1 De plus, la récupération automatique des données proposée par OR1 les instances améliore la fiabilité globale de votre domaine.

OpenSearch Le service envoie des OR1 métriques relatives au stockage à Amazon. CloudWatch Pour obtenir la liste des métriques disponibles, consultez la section ???.

OR1 les instances sont disponibles à la demande ou à la tarification des instances réservées, avec un tarif horaire pour les instances et le stockage fournis dans Amazon EBS et Amazon S3.

Rubriques

- Limites

- Réglage pour un meilleur débit d'ingestion
- En quoi les instances OpenSearch optimisées diffèrent-elles des autres instances
- En quoi OR1 diffère-t-il du UltraWarm stockage
- Approvisionnement d'un domaine avec des instances OR1

Limites

Tenez compte des limites suivantes lorsque vous utilisez OR1 des instances pour votre domaine.

- Les domaines nouvellement créés doivent exécuter la OpenSearch version 2.11 ou versions 2.11 ultérieures.
- Les domaines existants doivent exécuter la OpenSearch version 2.15 ou supérieure.
- Le chiffrement au repos doit être activé sur votre domaine. Pour de plus amples informations, veuillez consulter ???.
- Si votre domaine utilise des nœuds maîtres dédiés, ils doivent utiliser des instances Graviton. Pour plus d'informations sur les nœuds maîtres dédiés, consultez???.
- L'intervalle d'actualisation des index sur les OR1 instances doit être de 10 secondes ou plus. L'intervalle d'actualisation par défaut pour les OR1 instances est de 10 secondes.

Réglage pour un meilleur débit d'ingestion

Pour optimiser le débit d'indexation de vos OR1 instances, nous vous recommandons de procéder comme suit :

- Utilisez de gros volumes pour améliorer l'utilisation de la mémoire tampon. La taille recommandée ne doit pas dépasser 10 Mo.
- Utilisez plusieurs clients pour améliorer les performances du traitement parallèle.
- Définissez le nombre de partitions principales actives pour qu'il corresponde au nombre de nœuds de données afin d'optimiser l'utilisation des ressources.

En quoi les instances OpenSearch optimisées diffèrent-elles des autres instances

OpenSearch les instances optimisées diffèrent des instances non optimisées de la manière suivante :

- Pour les instances OpenSearch optimisées, l'indexation n'est effectuée que sur les partitions principales.
- Si des instances OpenSearch optimisées sont configurées avec des répliques, le taux d'indexation peut sembler inférieur à ce qu'il est réellement. Par exemple, s'il existe une partition principale et une partition répliquée, le taux d'indexation peut être de 1 000 alors que le taux d'indexation réel est de 2 000.
- OpenSearch les instances optimisées effectuent des opérations de mise en mémoire tampon avant d'envoyer à une source distante. Cela se traduit par des temps d'ingestion plus élevés.

> **Note**
>
> La `IndexingLatency` métrique n'est pas affectée, car elle n'inclut pas le temps nécessaire à la synchronisation du translog.

- Les fragments de réplique peuvent se trouver à quelques secondes de retard sur les fragments principaux. Vous pouvez surveiller le décalage à l'aide de la CloudWatch métrique `ReplicationLagMaxTime` Amazon

En quoi OR1 diffère-t-il du UltraWarm stockage

OpenSearch Le service offre des UltraWarm instances qui offrent un moyen économique de stocker de vastes quantités de données en lecture seule. Les deux OR1 UltraWarm instances stockent les données localement dans Amazon EBS et à distance dans Amazon S3. Cependant, OR1 les UltraWarm cas diffèrent de plusieurs manières importantes :

- OR1 les instances conservent une copie des données dans votre magasin local et distant. Dans UltraWarm certains cas, les données sont principalement conservées dans un magasin distant afin de réduire les coûts de stockage. En fonction de vos habitudes d'utilisation, les données peuvent être déplacées vers le stockage local.
- OR1 les instances sont actives et peuvent accepter des opérations de lecture et d'écriture, tandis que les données des UltraWarm instances sont en lecture seule jusqu'à ce que vous les redéplaciez manuellement vers le stockage à chaud.
- UltraWarm s'appuie sur des instantanés d'index pour la durabilité des données. OR1 les instances, en comparaison, effectuent la réplication et la restauration en arrière-plan. En cas d'index rouge, les OR1 instances restaurent automatiquement les fragments manquants de votre espace de

stockage distant dans Amazon S3. Le temps de récupération varie selon le volume de données à restaurer.

Pour plus d'informations sur UltraWarm le stockage, consultez???.

Approvisionnement d'un domaine avec des instances OR1

Vous pouvez sélectionner OR1 des instances pour vos nœuds de données lorsque vous créez un nouveau domaine avec le AWS Management Console ou le AWS Command Line Interface (AWS CLI). Vous pouvez ensuite indexer et interroger les données à l'aide de vos outils existants.

console

1. Accédez à la console Amazon OpenSearch Service à l'adressehttps://console.aws.amazon.com/aos/.
2. Dans le volet de navigation de gauche, choisissez Domains (Domaines).
3. Choisissez Create domain (Créer un domaine).
4. Dans la section Nombre de nœuds de données, développez le menu de la famille d'instances et choisissez OpenSearch Optimized.
5. Choisissez le type d'instance et les autres paramètres de stockage.
6. Dans la section Chiffrement, assurez-vous que l'option Activer le chiffrement des données au repos est sélectionnée.
7. Configurez le reste de votre domaine et choisissez Create.

AWS CLI

Pour provisionner un domaine qui utilise le OR1 stockage à l'aide du AWS CLI, vous devez fournir la valeur de la taille du type d' OR1 instance spécifique dans le`InstanceType`.

L'exemple suivant crée un domaine avec des OR1 instances de taille `2xlarge` et active le chiffrement au repos.

```
aws opensearch create-domain \
  --domain-name test-domain \
  --engine-version OpenSearch_2.11 \
  --cluster-config
 "InstanceType=or1.2xlarge.search,InstanceCount=3,DedicatedMasterEnabled=true,DedicatedMasterTy
 \
```

```
  --ebs-options "EBSEnabled=true,VolumeType=gp3,VolumeSize=200" \
  --encryption-at-rest-options Enabled=true \
  --advanced-security-options
 "Enabled=true,InternalUserDatabaseEnabled=true,MasterUserOptions={MasterUserName=test-
user,MasterUserPassword=test-password}" \
  --node-to-node-encryption-options Enabled=true \
  --domain-endpoint-options EnforceHTTPS=true \
  --access-policies '{"Version":"2012-10-17","Statement":
[{"Effect":"Allow","Principal":{"AWS":"*"},"Action":"es:*","Resource":"arn:aws:es:us-
east-1:account-id:domain/test-domain/*"}]}'
```

Index State Management dans Amazon OpenSearch Service

La gestion d'états des index (ISM) dans Amazon OpenSearch Service vous permet de définir des politiques de gestion personnalisées qui automatisent les tâches de routine et les appliquer aux indices et aux index. Vous n'avez plus besoin de configurer et de gérer des processus externes pour exécuter vos opérations d'index.

Chaque politique contient un état par défaut et une liste d'états pour la transition de l'index. Dans chaque état, vous pouvez définir une liste d'actions à effectuer et les conditions qui déclenchent ces transitions. Un cas d'utilisation typique consiste à supprimer périodiquement les anciens index après un certain temps. Par exemple, vous pouvez définir une politique qui déplace votre index dans un état `read_only` après 30 jours, puis les supprime après 90 jours.

Après avoir attaché une politique à un index, ISM crée une tâche qui s'exécute toutes les 5 à 8 minutes (ou 30 à 48 minutes pour les clusters antérieurs à 1.3) pour effectuer des actions de la politique, vérifier les conditions et faire passer l'index par différents états. En général, cette tâche est exécutée toutes les 5 minutes, plus une instabilité aléatoire de 0 à 60 % y est ajoutée pour vous assurer de ne pas voir un pic d'activité de tous vos indices en même temps. ISM n'exécute pas de tâches si l'état du cluster est rouge.

ISM requiert OpenSearch Elasticsearch 6.8 ou version ultérieure.

> **Note**
>
> Cette documentation fournit un bref aperçu de l'ISM ainsi que plusieurs exemples de politiques. Il explique également en quoi ISM pour les domaines Amazon OpenSearch Service diffère d'ISM sur les OpenSearch clusters autogérés. Pour une documentation complète d'ISM, y compris une référence complète des paramètres, une description de

chaque paramètre et une référence d'API, consultez la section Gestion de l'état des index dans la OpenSearch documentation.

Important

Vous ne pouvez plus utiliser de modèles d'index pour appliquer des stratégies ISM aux index nouvellement créés. Vous pouvez continuer à gérer automatiquement les index nouvellement créés via le champ Modèle ISM. Cette mise à jour introduit une modification majeure qui affecte les CloudFormation modèles existants utilisant ce paramètre.

Créer une politique ISM

Pour commencer à utiliser Index State Management

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Choisissez le domaine pour lequel vous voulez créer une stratégie ISM.
3. Depuis le tableau de bord du domaine, accédez à l'URL OpenSearch Tableaux de bord et connectez-vous avec votre nom d'utilisateur et votre mot de passe principaux. Le format de l'URL est le suivant :

   ```
   domain-endpoint/_dashboards/
   ```

4. Ouvrez le panneau de navigation de gauche dans OpenSearch Tableaux de bord et choisissez Index Management (Gestion des index), puis Create policy (Créer une politique).
5. Vous pouvez utiliser l'éditeur visuel ou l'éditeur JSON pour créer des stratégies. Nous vous recommandons d'utiliser l'éditeur visuel, car il offre un moyen plus structuré de définir les politiques. Afin d'obtenir de l'aide pour la création de stratégies, consultez les exemples de stratégies suivants.
6. Après avoir créé une stratégie, attachez-la à un ou plusieurs index :

   ```
   POST _plugins/_ism/add/my-index
   {
     "policy_id": "my-policy-id"
   }
   ```

> **Note**
>
> Si votre domaine exécute une ancienne version d'Elasticsearch, utilisez `_opendistro` au lieu de `_plugins`.

Vous pouvez également sélectionner l'index dans OpenSearch Tableaux de bord et choisir Appliquer la politique.

Exemples de politiques

Les exemples de politique suivants montrent comment automatiser les cas d'utilisation ISM courants.

Stockage à chaud > stockage UltraWarm > stockage à froid

Cet exemple de politique déplace un index du stockage à chaud vers UltraWarm, et éventuellement vers stockage à froid. Ensuite, il supprime l'index.

L'état initial de l'index est `hot`. Après dix jours, ISM le fait passer à l'état `warm`. 80 jours plus tard, l'index a plus de 90 jours et passe à l'état `cold`. Après un an, le service envoie une notification à une salle Amazon Chime pour indiquer que l'index est en cours de suppression, puis le supprime définitivement.

Notez que les index froids requièrent l'opération `cold_delete` plutôt que l'opération `delete` normale. Notez également qu'un `timestamp_field` explicite est nécessaire dans vos données afin de gérer les index froids avec ISM.

```
{
  "policy": {
    "description": "Demonstrate a hot-warm-cold-delete workflow.",
    "default_state": "hot",
    "schema_version": 1,
    "states": [{
        "name": "hot",
        "actions": [],
        "transitions": [{
          "state_name": "warm",
          "conditions": {
            "min_index_age": "10d"
          }
```

```
        }]
      },
      {
        "name": "warm",
        "actions": [{
          "warm_migration": {},
          "retry": {
            "count": 5,
            "delay": "1h"
          }
        }],
        "transitions": [{
          "state_name": "cold",
          "conditions": {
            "min_index_age": "90d"
          }
        }]
      },
      {
        "name": "cold",
        "actions": [{
            "cold_migration": {
              "timestamp_field": "<your timestamp field>"
            }
          }
        ],
        "transitions": [{
          "state_name": "delete",
          "conditions": {
             "min_index_age": "365d"
          }
        }]
      },
      {
        "name": "delete",
        "actions": [{
          "notification": {
            "destination": {
              "chime": {
                "url": "<URL>"
              }
            },
            "message_template": {
              "source": "The index {{ctx.index}} is being deleted."
```

```
            }
          }
        },
        {
          "cold_delete": {}
        }]
      }
    ]
  }
}
```

Réduire le nombre de réplicas

Cet exemple de politique réduit le nombre de réplicas à zéro au bout de sept jours pour économiser de l'espace disque, puis supprime l'index au bout de 21 jours. Cette politique suppose que votre index n'est pas critique et ne reçoit plus de demandes d'écriture, et qu'aucun réplica ne comporte un risque de perte de données.

```
{
  "policy": {
    "description": "Changes replica count and deletes.",
    "schema_version": 1,
    "default_state": "current",
    "states": [{
        "name": "current",
        "actions": [],
        "transitions": [{
          "state_name": "old",
          "conditions": {
            "min_index_age": "7d"
          }
        }]
      },
      {
        "name": "old",
        "actions": [{
          "replica_count": {
            "number_of_replicas": 0
          }
        }],
        "transitions": [{
          "state_name": "delete",
          "conditions": {
```

```
            "min_index_age": "21d"
          }
        }]
      },
      {
        "name": "delete",
        "actions": [{
          "delete": {}
        }],
        "transitions": []
      }
    ]
  }
}
```

Prendre un instantané d'index

Cet exemple de politique utilise l'opération `snapshot` pour prendre un instantané d'un index dès qu'il contient au moins un document. `repository` correspond au nom du référentiel d'instantanés manuels que vous avez enregistré dans Amazon S3. `snapshot` correspond au nom de l'instantané. Pour connaître les prérequis relatifs aux instantanés ainsi que la procédure à suivre pour enregistrer un référentiel, consultez the section called “Création d'instantanés d'index”.

```
{
  "policy": {
    "description": "Takes an index snapshot.",
    "schema_version": 1,
    "default_state": "empty",
    "states": [{
        "name": "empty",
        "actions": [],
        "transitions": [{
          "state_name": "occupied",
          "conditions": {
            "min_doc_count": 1
          }
        }]
      },
      {
        "name": "occupied",
        "actions": [{
          "snapshot": {
            "repository": "<my-repository>",
```

```
            "snapshot": "<my-snapshot>"
            }
          }],
          "transitions": []
      }
    ]
  }
}
```

Modèles ISM

Vous pouvez configurer un champ `ism_template` dans une politique afin que, lorsque vous créez un index qui correspond au modèle, la politique soit automatiquement attachée à cet index. Dans cet exemple, tout index créé dont le nom commence par « log » est automatiquement mis en correspondance avec la politique ISM `my-policy-id` :

```
PUT _plugins/_ism/policies/my-policy-id
{
  "policy": {
    "description": "Example policy.",
    "default_state": "...",
    "states": [...],
    "ism_template": {
      "index_patterns": ["log*"],
      "priority": 100
    }
  }
}
```

Pour un exemple plus détaillé, consultez Exemple de politique avec modèle ISM pour la substitution automatique.

Différences

ISM pour OpenSearch Amazon OpenSearch Service présente plusieurs différences par rapport à Elasticsearch.

Opérations ISM

- OpenSearch Le service prend en charge trois opérations ISM uniques `warm_migrationcold_migration`,, et `cold_delete` :

- Si votre domaine est UltraWarmactivé, l'`warm_migration`action fait passer l'index en stockage hot.
- Si le stockage à froid est activé sur votre domaine, l'action `cold_migration` fait passer l'index au stockage à froid et l'action `cold_delete` supprime l'index du stockage à froid.

  Même si l'une de ces actions ne se termine pas dans le délai d'attente défini, la migration ou la suppression des index se poursuit. Définir une notification d'erreur error_notification pour l'une des actions ci-dessus vous permettra de savoir que l'action a échoué si elle ne s'est pas terminée dans le délai imparti, elle est toutefois uniquement destinée à votre propre référence. L'opération proprement dite n'a pas dépassé le délai d'attente et continue de s'exécuter jusqu'à ce qu'elle réussisse ou échoue.

- Si votre domaine exécute Elasticsearch 7.4 OpenSearch ou version ultérieure, OpenSearch Service prendra en charge l'ISM `open` et `close` les opérations.

- Si votre domaine exécute Elasticsearch 7.7 OpenSearch ou version ultérieure, OpenSearch Service prendra en charge l'opération ISM`snapshot`.

Opérations ISM de stockage à froid

Pour les index à froid, vous devez indiquer un `?type=_cold` paramètre lorsque vous utilisez l'ISM APIs suivant :

- Ajouter une stratégie
- Supprimer une stratégie
- Mettre à jour une stratégie
- Réessayer une indexation qui a échoué
- Expliquer un index

Ces index APIs à froid présentent les différences supplémentaires suivantes :

- Les opérateurs de caractères génériques ne sont pas pris en charge, à moins que vous les utilisiez à la fin. Par exemple, `_plugins/_ism/<add, remove, change_policy, retry, explain>/logstash-*` est pris en charge, mais `_plugins/_ism/<add, remove, change_policy, retry, explain>/iad-*-prod` ne l'est pas.

- Les modèles et noms à plusieurs index ne sont pas pris en charge. Par exemple, `_plugins/_ism/<add, remove, change_policy, retry, explain>/app-logs` est pris en charge, mais `_plugins/_ism/<add, remove, change_policy, retry, explain>/app-logs,sample-data` ne l'est pas.

Paramètres ISM

OpenSearch et Elasticsearch vous permettent de modifier tous les paramètres ISM disponibles à l'aide de l'`_cluster/settings`API. Sur Amazon OpenSearch Service, vous pouvez uniquement modifier les paramètres ISM suivants :

- Paramètres au niveau du cluster :
  - `plugins.index_state_management.enabled`
  - `plugins.index_state_management.history.enabled`
- Paramètres au niveau de l'index :
  - `plugins.index_state_management.rollover_alias`

Didacticiel : automatisation des processus de gestion de l'état des index

Ce didacticiel montre comment implémenter une politique ISM qui automatise les tâches de routine de gestion des index et les applique aux index et aux modèles d'index.

La gestion d'états des index (ISM) dans Amazon OpenSearch Service vous permet d'automatiser les activités récurrentes de gestion des index, afin d'éviter d'utiliser des outils supplémentaires pour gérer les cycles de vie des index. Vous pouvez créer une politique qui automatise ces opérations en fonction de l'âge, de la taille et d'autres conditions de l'index, le tout depuis votre domaine Amazon OpenSearch Service.

OpenSearch Le service prend en charge trois niveaux de stockage : l'état « hot » par défaut pour l'écriture active et les analyses à faible latence, UltraWarm pour les données en lecture seule jusqu'à trois pétaoctets et le stockage cold pour l'archivage illimité à long terme.

Ce didacticiel présente un exemple de cas d'utilisation de la gestion de données de séries temporelles dans des index quotidiens. Dans ce didacticiel, vous configurez une politique qui prend un instantané automatisé de chaque index attaché après 24 heures. Elle fait ensuite migrer l'index de

l'état hot par défaut vers un UltraWarm stockage après deux jours, un stockage cold après 30 jours, et enfin supprime l'index après 60 jours.

Prérequis

- Votre domaine OpenSearch de service doit exécuter Elasticsearch version 6.8 ou ultérieure.
- Le stockage à froid doit UltraWarmêtre activé sur votre domaine.
- Vous devez inscrire un référentiel d'instantanés manuels pour votre domaine.
- Votre rôle d'utilisateur doit disposer d'autorisations suffisantes pour accéder à la console OpenSearch de service. Si nécessaire, validez et configurez l'accès à votre domaine.

Étape 1 : Configurer la politique ISM

Tout d'abord, configurez une politique ISM dans OpenSearch Tableaux de bord.

1. Depuis le tableau de bord de votre domaine dans la console de OpenSearch service, accédez à l'URL OpenSearch Dashboards et connectez-vous avec votre nom d'utilisateur et votre mot de passe principaux. Le format de l'URL est le suivant : *domain-endpoint*`/_dashboards/`.
2. Dans OpenSearch Tableaux de bord, choisissez Add sample data (Ajouter des données exemples) et ajoutez un ou plusieurs des index exemples à votre domaine.
3. Ouvrez le panneau de navigation de gauche et choisissez Index Management (Gestion des index), puis Create policy (Créer une politique).
4. Nommez la stratégie `ism-policy-example`.
5. Remplacez la stratégie par défaut par la stratégie suivante :

```
{
  "policy": {
    "description": "Move indexes between storage tiers",
    "default_state": "hot",
    "states": [
      {
        "name": "hot",
        "actions": [],
        "transitions": [
          {
            "state_name": "snapshot",
            "conditions": {
              "min_index_age": "24h"
```

```
            }
          }
        ]
      },
      {
        "name": "snapshot",
        "actions": [
          {
            "retry": {
              "count": 5,
              "backoff": "exponential",
              "delay": "30m"
            },
            "snapshot": {
              "repository": "snapshot-repo",
              "snapshot": "ism-snapshot"
            }
          }
        ],
        "transitions": [
          {
            "state_name": "warm",
            "conditions": {
              "min_index_age": "2d"
            }
          }
        ]
      },
      {
        "name": "warm",
        "actions": [
          {
            "retry": {
              "count": 5,
              "backoff": "exponential",
              "delay": "1h"
            },
            "warm_migration": {}
          }
        ],
        "transitions": [
          {
            "state_name": "cold",
            "conditions": {
```

```
            "min_index_age": "30d"
          }
        }
      ]
    },
    {
      "name": "cold",
      "actions": [
        {
          "retry": {
            "count": 5,
            "backoff": "exponential",
            "delay": "1h"
          },
          "cold_migration": {
            "start_time": null,
            "end_time": null,
            "timestamp_field": "@timestamp",
            "ignore": "none"
          }
        }
      ],
      "transitions": [
        {
          "state_name": "delete",
          "conditions": {
            "min_index_age": "60d"
          }
        }
      ]
    },
    {
      "name": "delete",
      "actions": [
        {
          "cold_delete": {}
        }
      ],
      "transitions": []
    }
  ],
  "ism_template": [
    {
      "index_patterns": [
```

```
          "index-*"
        ],
        "priority": 100
      }
    ]
  }
}
```

> **Note**
>
> Le champ `ism_template` attache automatiquement la politique à tout index nouvellement créé qui correspond à l'un des `index_patterns` spécifiés. Dans ce cas, tous les index qui commencent par `index-`. Vous pouvez modifier ce champ pour qu'il corresponde à un format d'index dans votre environnement. Pour plus d'informations, consultez Modèles ISM.

6. Dans la section `snapshot` de la politique, remplacez *`snapshot-repo`* par le nom du référentiel d'instantanés que vous avez inscrit pour votre domaine. Vous pouvez également remplacer facultativement *`ism-snapshot`*, qui sera le nom de l'instantané lorsqu'il sera créé.
7. Sélectionnez Create (Créer). La politique est maintenant visible sur la page State management policies (Politiques de gestion des états).

Étape 2 : Attacher la politique à un ou plusieurs index

Maintenant que vous avez créé votre politique, attachez-la à un ou plusieurs index dans votre cluster.

1. Accédez à l'onglet Hot indices (Index hot) et recherchez `opensearch_dashboards_sample`, qui répertorie tous les exemples d'index que vous avez ajoutés à l'étape 1.
2. Sélectionnez tous les index et choisissez Appliquer la politique, puis choisissez la ism-policy-examplepolitique que vous venez de créer.
3. Choisissez Appliquer.

Vous pouvez surveiller les index à mesure qu'ils passent par les différents états sur la page Policy managed indices (Index gérés par politique).

Synthèse des index dans Amazon OpenSearch Service avec des cumulatifs d'index

Dans Amazon OpenSearch Service, les cumulatifs d'index vous permettent de réduire les coûts de stockage en cumulant périodiquement les anciennes données dans des index synthétisés.

Vous sélectionnez les champs qui vous intéressent et utilisez un cumulatif d'index pour créer un nouvel index avec uniquement ces champs agrégés dans des compartiments de temps plus sommaires. Vous pouvez stocker des mois ou des années de données historiques à une fraction du coût avec les mêmes performances de requête.

Les cumulatifs d'index nécessitent Elasticsearch 7.9 OpenSearch ou version ultérieure.

> **Note**
>
> Cette documentation vous aide à démarrer avec la création d'une tâche de cumul d'index dans Amazon OpenSearch Service. Pour une documentation complète, y compris une liste de tous les paramètres disponibles et une référence complète de l'API, voir Index cumulés dans la OpenSearch documentation.

Création d'une tâche de cumulatif d'index

Pour commencer, choisissez Gestion des index dans OpenSearch Dashboards. Sélectionnez Tâches de cumulatif et choisissez Créer une tâche de cumulatif.

Étape 1 : Configurer les index

Configurez les index source et cible. L'index source correspond à l'index que vous souhaitez cumuler. L'index cible correspond à l'emplacement où les résultats du cumulatif d'index sont enregistrés.

Après avoir créé une tâche de cumulatif d'index, vous ne pouvez pas modifier vos sélections d'index.

Étape 2 : Définir des agrégations et des métriques

Sélectionnez les attributs avec les agrégations (termes et histogrammes) et les métriques (moyenne, somme, max, min et nombre de valeurs) que vous souhaitez cumuler. Assurez-vous de ne pas ajouter trop d'attributs hautement granulaires car vous n'économiseriez pas beaucoup d'espace.

Étape 3 : Spécifier des planifications

Spécifiez une planification pour cumuler vos index au fur et à mesure de leur ingestion. La tâche de cumulatif d'index est activée par défaut.

Étape 4 : vérifier et créer

Vérifiez votre configuration et sélectionnez Créer.

Étape 5 : Rechercher dans l'index cible

Vous pouvez utiliser l'API standard `_search` pour effectuer une recherche dans l'index cible. Dans l'index cible, vous ne pouvez pas accéder à la structure interne des données, car le plug-in réécrit automatiquement la requête en arrière-plan pour l'adapter à l'index cible. Ainsi, vous pouvez utiliser la même requête pour l'index source et l'index cible.

Pour interroger l'index cible, définissez `size` sur 0 :

```
GET target_index/_search
{
  "size": 0,
  "query": {
    "match_all": {}
  },
  "aggs": {
    "avg_cpu": {
      "avg": {
        "field": "cpu_usage"
      }
    }
  }
}
```

> **Note**
>
> OpenSearch les versions 2.2 et ultérieures prennent en charge la recherche de plusieurs index cumulatifs en une seule requête. OpenSearch les versions antérieures à 2.2 et les anciennes versions d'Elasticsearch OSS ne prennent en charge qu'un seul index cumulatif par recherche.

Transformations d'index dans Amazon Service OpenSearch

Alors que les tâches de cumulatif d'index vous permettent de réduire la granularité des données en regroupant les anciennes données dans des index condensés, les tâches de transformation vous permettent de créer une vue différente et résumée de vos données, centrée sur certains champs, afin que vous puissiez visualiser ou analyser les données de différentes manières.

Les transformations d'index disposent d'une interface utilisateur OpenSearch Dashboards et d'une API REST. Cette fonction nécessite une OpenSearch version 1.0 ou une version ultérieure.

> **Note**
>
> Cette documentation fournit un bref aperçu des transformations d'index pour vous aider à commencer à les utiliser sur un domaine Amazon OpenSearch Service. Pour une documentation complète et une référence sur l'API REST, voir les transformations d'index dans la OpenSearch documentation open source.

Création d'une tâche de transformation d'index

Si vous n'avez pas de données dans votre cluster, utilisez l'échantillon de données de vol dans OpenSearch Dashboards pour tester les tâches de transformation. Après avoir ajouté les données, lancez OpenSearch Dashboards. Choisissez ensuite Index Management (Gestion des index), Transform Jobs (Tâches de transformation) et Create Transform Job (Créer une tâche de transformation).

Étape 1 : Choix des index

Dans la section Indices (Index), sélectionnez l'index source et l'index cible. Vous pouvez soit sélectionner un index cible existant, soit en créer un nouveau en lui donnant un nom.

Si vous voulez transformer seulement un sous-ensemble de votre index source, choisissez Add Data Filter (Ajouter un filtre de données) et utilisez la OpenSearch requête DSL pour spécifier un sous-ensemble de votre index source.

Étape 2 : Choix des champs

Après avoir choisi vos index, choisissez les champs que vous voulez utiliser dans votre tâche de transformation, ainsi que l'utilisation de groupements ou d'agrégations.

- Vous pouvez utiliser des regroupements pour placer vos données dans des compartiments séparés dans votre index transformé. Par exemple, si vous souhaitez regrouper toutes les destinations d'aéroport dans les données de vol de l'échantillon, regroupez le `DestAirportID` champ dans un champ ou un champ cible, et vous pourrez trouver l'aéroport groupé IDs dans votre index transformé une fois la tâche de transformation terminée. `DestAirportID_terms`
- D'autre part, les agrégations vous permettent d'effectuer des calculs simples. Par exemple, vous pouvez inclure une agrégation dans votre tâche de transformation pour définir un nouveau champ de `sum_of_total_ticket_price` qui calcule la somme de tous les billets d'avion. Vous pouvez ensuite analyser les nouvelles données dans votre index transformé.

Étape 3 : Spécifier une planification

Les tâches de transformation sont activées par défaut et s'exécutent selon des planifications. Pour transform execution interval (intervalle d'exécution de transformation), spécifiez un intervalle en minutes, heures ou jours.

Étape 4 : Vérifier et surveiller

Vérifiez votre configuration et sélectionnez Créer. Surveillez ensuite la colonne Transform job status (État de la tâche de transformation).

Étape 5 : Rechercher dans l'index cible

Une fois la tâche terminée, vous pouvez utiliser l'API standard `_search` pour effectuer une recherche dans l'index cible.

Par exemple, après avoir exécuté une tâche de transformation qui transforme les données de vol en fonction du champ `DestAirportID`, vous pouvez exécuter la requête suivante pour renvoyer tous les champs dont la valeur est `SFO` :

```
GET target_index/_search
{
  "query": {
    "match": {
      "DestAirportID_terms" : "SFO"
    }
  }
}
```

Réplication entre clusters pour Amazon Service OpenSearch

Grâce à la réplication entre clusters dans Amazon OpenSearch Service, vous pouvez répliquer les index utilisateur, les mappages et les métadonnées d'un domaine de OpenSearch service à un autre. L'utilisation de la réplication entre clusters contribue à assurer la reprise après sinistre en cas l'interruption de service et vous permet de répliquer des données dans des centres de données géographiquement éloignés afin de réduire la latence. Vous payez les frais de transfert de AWS données standard pour les données transférées entre les domaines.

La réplication entre clusters suit un modèle de réplication actif-passif dans lequel l'index local ou suiveur extrait les données de l'index distant ou de l'index leader. L'indice leader fait référence à la source des données, ou à l'index à partir duquel vous souhaitez répliquer les données. L'index suiveur fait référence à la cible des données, ou à l'index vers lequel vous souhaitez répliquer les données.

La réplication entre clusters est disponible sur les domaines exécutant Elasticsearch 7.10 ou 1.1 ou OpenSearch version ultérieure.

> **Note**
>
> Cette documentation explique comment configurer la réplication entre clusters du point de vue d'Amazon OpenSearch Service. Cela inclut l'utilisation de AWS Management Console pour configurer des connexions entre clusters, ce qui n'est pas possible sur un cluster autogéré OpenSearch . Pour une documentation complète, y compris une référence de paramètres et une référence d'API complète, consultez la section Réplication entre clusters dans la OpenSearch documentation.

Rubriques

- Limites
- Prérequis
- Conditions d'autorisation
- Configurer une connexion inter-clusters
- Démarrer la réplication
- Confirmer la réplication
- Mettre en pause et reprendre la réplication

- Arrêter la réplication
- Suivi automatique
- Mise à niveau des domaines connectés

Limites

La réplication inter-clusters (CCR) présente les limitations suivantes :

- Vous ne pouvez pas répliquer des données entre des domaines Amazon OpenSearch Service et des clusters autogérés OpenSearch ou Elasticsearch.
- Vous ne pouvez pas répliquer un index d'un domaine suiveur vers un autre domaine suiveur. Si vous souhaitez répliquer un index vers plusieurs domaines abonnés, vous ne pouvez le répliquer qu'à partir du seul domaine leader.
- Un domaine peut être connecté, via une combinaison de connexions entrantes et sortantes, à un maximum de 20 autres domaines.
- Lorsque vous configurez initialement une connexion entre clusters, le domaine principal doit se trouver sur une version identique ou supérieure à celle du domaine suiveur.
- Vous ne pouvez pas l'utiliser AWS CloudFormation pour connecter des domaines.
- Vous ne pouvez pas utiliser la réplication inter-clusters (CCR) sur des instances M3 ou les instances extensibles (T2 et T3).
- Vous ne pouvez pas répliquer des données entre des index UltraWarm ou des index à froid. Les deux index doivent être dans un stockage à chaud.
- Lorsque vous supprimez un index du domaine leader, l'index correspondant du domaine suiveur n'est pas automatiquement supprimé.

Prérequis

Avant de configurer la réplication inter-clusters (CCR), vérifiez que vos domaines répondent aux exigences suivantes :

- Elasticsearch 7.10 ou 1.1 ou version ultérieure OpenSearch
- Contrôle précis des accès activé
- Node-to-node chiffrement activé

Conditions d'autorisation

Pour commencer la réplication, vous devez inclure l'autorisation `es:ESCrossClusterGet` dans le domaine distant (leader). Nous recommandons la politique IAM suivante pour le domaine distant. Cette politique vous permet également d'effectuer d'autres opérations, telles que l'indexation de documents et la réalisation de recherches standard :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "*"
        ]
      },
      "Action": [
        "es:ESHttp*"
      ],
      "Resource": "arn:aws:es:region:account:domain/leader-domain/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "es:ESCrossClusterGet",
      "Resource": "arn:aws:es:region:account:domain/leader-domain"
    }
  ]
}
```

Assurez-vous que l'autorisation `es:ESCrossClusterGet` est appliquée pour `/leader-domain` et non pour `/leader-domain/*`.

Pour que les utilisateurs non-administrateurs puissent effectuer des activités de réplication, ils doivent également être mappés aux autorisations appropriées. La plupart des autorisations correspondent

à des opérations d'API REST spécifiques. Par exemple, l'autorisation `indices:admin/plugins/replication/index/_resume` vous permet de reprendre la réplication d'un index. Pour obtenir la liste complète des autorisations, consultez la section Autorisations de réplication dans la OpenSearch documentation.

> **Note**
>
> Les commandes permettant de démarrer la réplication et de créer une règle de réplication sont des cas particuliers. Comme ils invoquent des processus en arrière-plan sur les domaines leader et suiveur, vous devez transmettre un `leader_cluster_role` et `follower_cluster_role` dans la demande. OpenSearch Le service utilise ces rôles dans toutes les tâches de réplication du backend. Pour plus d'informations sur le mappage et l'utilisation de ces rôles, voir Mapper les rôles du cluster leader et suiveur dans la OpenSearch documentation.

Configurer une connexion inter-clusters

Pour répliquer des index d'un domaine à un autre, vous devez configurer une connexion inter-clusters entre les domaines. La manière la plus simple de connecter des domaines consiste à utiliser l'onglet Connexions du tableau de bord du domaine. Vous pouvez également utiliser l'API de configuration ou la CLI AWS. Étant donné que la réplication inter-clusters (CCR) suit un modèle « pull », vous initiez des connexions à partir du domaine suiveur.

> **Note**
>
> Si vous avez déjà connecté deux domaines pour effectuer des recherches inter-clusters, vous ne pouvez pas utiliser cette même connexion pour la réplication. La connexion est marquée comme `SEARCH_ONLY` dans la console. Afin d'effectuer une réplication entre deux domaines précédemment connectés, vous devez supprimer la connexion et la recréer. Une fois cette opération effectuée, la connexion est disponible pour la recherche inter-clusters et la réplication inter-clusters.

Configurer une connexion

1. Dans la console Amazon OpenSearch Service, sélectionnez le domaine abonné, accédez à l'onglet Connections, puis choisissez Request.

2. Dans Alias de la connexion, saisissez un nom pour votre connexion.
3. Choisissez entre vous connecter à un domaine dans votre Compte AWS région ou dans un autre compte ou région.

   - Pour vous connecter à un domaine dans votre Compte AWS région, sélectionnez le domaine et choisissez Request.
   - Pour vous connecter à un domaine situé dans une autre région Compte AWS ou dans une autre région, spécifiez l'ARN du domaine distant et choisissez Request.

OpenSearch Le service valide la demande de connexion. Si les domaines sont incompatibles, la connexion échouera. Si la validation réussit, elle sera envoyée au domaine de destination pour approbation. Lorsque le domaine de destination approuvera la demande, vous pourrez commencer la réplication.

La réplication entre clusters prend en charge la réplication bidirectionnelle. Cela signifie que vous pouvez créer une connexion sortante entre le domaine A et le domaine B et une autre connexion sortante entre le domaine B et le domaine A. Vous pouvez ensuite configurer la réplication de telle sorte que le domaine A suive un index dans le domaine B et que le domaine B suive un index dans le domaine A.

Démarrer la réplication

Après avoir établi une connexion inter-clusters, vous pourrez commencer à répliquer des données. Commencez par créer un index dans le domaine leader à répliquer :

```
PUT leader-01
```

Pour répliquer cet index, envoyez cette commande au domaine suiveur :

```
PUT _plugins/_replication/follower-01/_start
{
   "leader_alias": "connection-alias",
   "leader_index": "leader-01",
   "use_roles":{
      "leader_cluster_role": "all_access",
      "follower_cluster_role": "all_access"
   }
}
```

Vous trouverez l'alias de connexion dans l'onglet Connexions du tableau de bord de votre domaine.

Cet exemple suppose qu'un administrateur émette la demande et utilise `all_access` pour le `leader_cluster_role` et le `follower_cluster_role` pour plus de simplicité. Dans les environnements de production, cependant, nous vous recommandons de créer des utilisateurs de réplication à la fois sur les index leader et suiveur et de les mapper en conséquence. Les noms d'utilisateur doivent être identiques. Pour plus d'informations sur ces rôles et sur la manière de les mapper, voir Mapper les rôles du cluster leader et suiveur dans la OpenSearch documentation.

Confirmer la réplication

Pour confirmer que la réplication est en cours, obtenez l'état de la réplication :

```
GET _plugins/_replication/follower-01/_status

{
  "status" : "SYNCING",
  "reason" : "User initiated",
  "leader_alias" : "connection-alias",
  "leader_index" : "leader-01",
  "follower_index" : "follower-01",
  "syncing_details" : {
    "leader_checkpoint" : -5,
    "follower_checkpoint" : -5,
    "seq_no" : 0
  }
}
```

Les valeurs du point de contrôle du leader et du suiveur commencent par des entiers négatifs et reflètent le nombre de partitions dont vous disposez (-1 pour une seule partition, -5 pour cinq partitions, etc.). Les valeurs sont incrémentées en entiers positifs à chaque modification apportée. Si les valeurs sont identiques, cela signifie que les index sont parfaitement synchronisés. Vous pouvez utiliser ces valeurs de point de contrôle pour mesurer la latence de réplication entre vos domaines.

Pour mieux valider la réplication, ajoutez un document à l'index leader :

```
PUT leader-01/_doc/1
{
   "Doctor Sleep":"Stephen King"
}
```

Et confirmez qu'il s'affiche dans l'index suiveur :

```
GET follower-01/_search

{
    ...
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "follower-01",
        "_type" : "_doc",
        "_id" : "1",
        "_score" : 1.0,
        "_source" : {
          "Doctor Sleep" : "Stephen King"
        }
      }
    ]
  }
}
```

Mettre en pause et reprendre la réplication

Vous pouvez suspendre temporairement la réplication si vous devez résoudre des problèmes ou réduire la charge sur le domaine leader. Envoyez cette demande au domaine suiveur. Veillez à inclure un corps de requête vide :

```
POST _plugins/_replication/follower-01/_pause
{}
```

Obtenez ensuite l'état pour vous assurer que la réplication est suspendue :

```
GET _plugins/_replication/follower-01/_status

{
  "status" : "PAUSED",
  "reason" : "User initiated",
  "leader_alias" : "connection-alias",
  "leader_index" : "leader-01",
  "follower_index" : "follower-01"
}
```

Lorsque les modifications seront terminées, reprenez la réplication. Envoyez cette demande au domaine suiveur. Veillez à inclure un corps de requête vide :

```
POST _plugins/_replication/follower-01/_resume
{}
```

Vous ne pouvez pas reprendre la réplication après qu'elle a été mise en pause pendant plus de 12 heures. Vous devez arrêter la réplication, supprimer l'index suiveur et redémarrer la réplication du leader.

Arrêter la réplication

Quand vous arrêtez complètement la réplication, l'index suiveur cessera de suivre le leader et deviendra un index standard. Vous ne pouvez pas redémarrer une réplication après l'avoir arrêtée.

Arrêtez la réplication à partir du domaine suiveur. Veillez à inclure un corps de requête vide :

```
POST _plugins/_replication/follower-01/_stop
{}
```

Suivi automatique

Vous pouvez définir un ensemble de règles de réplication sur un domaine leader unique qui réplique automatiquement les index correspondant à un modèle spécifié. Lorsqu'un index du domaine leader correspond à l'un des modèles (par exemple,`books*`), un index suiveur correspondant est créé sur le domaine suiveur. OpenSearch Le service réplique tous les index existants qui correspondent au modèle, ainsi que les nouveaux index que vous créez. Il ne réplique pas les index qui existent déjà sur le domaine suiveur.

Pour répliquer tous les index (à l'exception des index créés par le système et ceux qui existent déjà sur le domaine suiveur), utilisez un modèle générique (`*`).

Créer une règle de réplication.

Créez une règle de réplication sur le domaine suiveur et indiquez le nom de la connexion inter-clusters :

```
POST _plugins/_replication/_autofollow
{
   "leader_alias" : "connection-alias",
   "name": "rule-name",
```

```
   "pattern": "books*",
   "use_roles":{
      "leader_cluster_role": "all_access",
      "follower_cluster_role": "all_access"
   }
}
```

Vous trouverez l'alias de connexion dans l'onglet Connexions du tableau de bord de votre domaine.

Cet exemple suppose qu'un administrateur émet la demande et utilise `all_access` en tant que rôles de domaine leader et suiveur pour plus de simplicité. Dans les environnements de production, cependant, nous vous recommandons de créer des utilisateurs de réplication à la fois sur les index leader et suiveur et de les mapper en conséquence. Les noms d'utilisateur doivent être identiques. Pour plus d'informations sur ces rôles et sur la manière de les mapper, voir Mapper les rôles du cluster leader et suiveur dans la OpenSearch documentation.

Pour récupérer la liste des règles de réplication existantes sur un domaine, utilisez l'opération d'API des statistiques de suivi automatique.

Pour tester la règle, créez un index correspondant au modèle sur le domaine leader :

```
PUT books-are-fun
```

Et vérifiez que son réplica s'affiche sur le domaine suiveur :

```
GET _cat/indices

health status index          uuid                   pri rep docs.count docs.deleted
 store.size pri.store.size
green  open   books-are-fun  ldfHO78xYYdxRMULuiTvSQ     1   1          0            0
     208b           208b
```

Supprimer une règle de réplication

Lorsque vous supprimez une règle de réplication, le OpenSearch service arrête de répliquer les nouveaux index correspondant au modèle, mais poursuit l'activité de réplication existante jusqu'à ce que vous arrêtiez la réplication de ces index.

Supprimez les règles de réplication du domaine suiveur :

```
DELETE _plugins/_replication/_autofollow
```

```
{
   "leader_alias" : "connection-alias",
   "name": "rule-name"
}
```

Mise à niveau des domaines connectés

Afin de mettre à niveau la version du moteur de deux domaines dotés d'une connexion entre clusters, mettez d'abord à niveau le domaine suiveur, puis le domaine leader. Ne supprimez pas la connexion entre eux, sinon la réplication s'interrompt et vous ne pourrez pas la reprendre.

Migration des index Amazon OpenSearch Service à l'aide de la réindexation à distance

La réindexation à distance vous permet de copier des index d'un domaine Amazon OpenSearch Service vers un autre. Vous pouvez migrer des index depuis n'importe quel domaine de OpenSearch service ou depuis des clusters autogérés OpenSearch ou Elasticsearch.

Un domaine et un index distants font référence à la source des données, ou au domaine et à l'index à partir desquels vous souhaitez copier les données. Un domaine et un index locaux font référence à la cible des données, ou au domaine et à l'index vers lesquels vous souhaitez copier les données.

La réindexation à distance nécessite la OpenSearch version 1.0 ou ultérieure, ou Elasticsearch 6.7 ou version ultérieure, sur le domaine local. Le domaine distant doit présenter une version inférieure ou la même version majeure que le domaine local. Les versions d'Elasticsearch sont considérées comme inférieures aux OpenSearch versions, ce qui signifie que vous pouvez réindexer les données des domaines Elasticsearch vers des domaines. OpenSearch Dans la même version majeure, le domaine distant peut correspondre à n'importe quelle version mineure. Par exemple, la réindexation à distance d'Elasticsearch 7.10.x vers 7.9 est prise en charge, mais pas de la version OpenSearch 1.0 vers Elasticsearch 7.10.x.

> **Note**
>
> Cette documentation explique comment réindexer les données entre les domaines Amazon OpenSearch Service. Pour une documentation complète de l'`reindex`opération, y compris les étapes détaillées et les options prises en charge, voir le document Reindex dans la OpenSearch documentation.

Rubriques

- Prérequis
- Réindexer les données entre les domaines Internet OpenSearch du service
- Réindexer les données entre les domaines OpenSearch de service lorsque la télécommande se trouve dans un VPC
- Réindexer les données entre les domaines non liés OpenSearch aux services
- Réindexer des jeux de données volumineux
- Paramètres de réindexation à distance

Prérequis

La réindexation à distance présente les exigences suivantes :

- Le domaine distant doit être accessible à partir du domaine local. Pour un domaine distant résidant au sein d'un VPC, le domaine local doit avoir accès au VPC. Ce processus varie en fonction de la configuration du réseau, mais implique probablement la connexion à un VPN ou à un réseau géré, ou l'utilisation de la connexion de point de terminaison VPC native. Pour en savoir plus, veuillez consulter la section the section called “Prise en charge de VPC”.
- La demande doit être autorisée par le domaine distant comme toute autre demande REST. Si le contrôle d'accès détaillé est activé dans le domaine distant, vous devez être autorisé à effectuer une réindexation sur le domaine distant et à lire l'index sur le domaine local. Pour obtenir plus d’informations de sécurité, consultez the section called “Contrôle précis des accès”.
- Nous vous recommandons de créer un index avec le paramètre souhaité sur votre domaine local avant de lancer le processus de réindexation.
- Si votre domaine utilise un type d'instance T2 ou T3 pour vos nœuds de données, vous ne pouvez pas utiliser la réindexation à distance.

Réindexer les données entre les domaines Internet OpenSearch du service

Le scénario le plus simple est que l'index distant se trouve dans le même Région AWS que votre domaine local avec un point de terminaison accessible au public et que vous avez signé des informations d'identification IAM.

Dans le domaine distant, spécifiez l'index distant à partir duquel vous souhaitez réindexer et l'index local à réindexer :

```
POST _reindex
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443"
    },
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

Vous devez ajouter 443 à la fin du point de terminaison du domaine distant à des fins de vérification de validation.

Pour vérifier que l'index est copié sur le domaine local, envoyez cette demande au domaine local :

```
GET local_index/_search
```

Si l'index distant se trouve dans une région différente de votre domaine local, transmettez son nom de région, comme dans cet exemple de demande :

```
POST _reindex
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443",
      "region": "eu-west-1"
    },
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

Dans le cas d'une région isolée comme une région AWS GovCloud (US) ou une région de Chine, le point de terminaison peut ne pas être accessible car votre utilisateur IAM n'est pas reconnu dans ces régions.

Si le domaine distant est sécurisé par une authentification de base, spécifiez le nom d'utilisateur et le mot de passe :

```
POST _reindex
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443",
      "username": "username",
      "password": "password"
    },
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

Réindexer les données entre les domaines OpenSearch de service lorsque la télécommande se trouve dans un VPC

Chaque domaine OpenSearch de service est constitué de sa propre infrastructure interne de cloud privé virtuel (VPC). Lorsque vous créez un nouveau domaine dans un OpenSearch Service VPC existant, une interface Elastic Network est créée pour chaque nœud de données du VPC.

Comme l'opération de réindexation à distance est effectuée à partir du domaine de OpenSearch service distant, et donc au sein de son propre VPC privé, vous devez disposer d'un moyen d'accéder au VPC du domaine local. Vous pouvez le faire soit en utilisant la fonctionnalité intégrée de connexion des points de terminaison VPC pour établir une connexion AWS PrivateLink, soit en configurant un proxy.

Si votre domaine local utilise OpenSearch la version 1.0 ou ultérieure, vous pouvez utiliser la console ou le AWS CLI pour créer une AWS PrivateLink connexion. Une AWS PrivateLink connexion permet aux ressources du VPC local de se connecter de manière privée aux ressources du VPC distant au sein de celui-ci. Région AWS

Pour créer une connexion de point de terminaison VPC, le domaine source à réindexer doit se trouver dans un VPC local, et les domaines source et de destination doivent être identiques. Région AWS

Réindexez les données avec le AWS Management Console

Vous pouvez utiliser la réindexation à distance avec la console pour copier des index entre deux domaines partageant une connexion de point de terminaison VPC.

1. Accédez à la console Amazon OpenSearch Service à l'adressehttps://console.aws.amazon.com/aos/.
2. Dans le volet de navigation de gauche, choisissez Domains (Domaines).
3. Sélectionnez le domaine local ou le domaine dans lequel vous souhaitez copier les données. Cette action ouvre la page des détails du domaine. Cliquez sur l'onglet Connexions sous les informations générales, puis sélectionnez Demander.
4. Sur la page Demander une connexion, sélectionnez VPC Endpoint Connection pour votre mode de connexion et entrez les autres informations pertinentes. Ces détails incluent le domaine distant, qui est le domaine à partir duquel vous souhaitez copier des données. Choisissez ensuite Request (Demander).
5. Accédez à la page de détails du domaine distant, choisissez l'onglet Connexions et recherchez le tableau des connexions entrantes. Sélectionnez la case à cocher située à côté du nom du domaine à partir duquel vous venez de créer la connexion (le domaine local). Choisissez Approve (Approuver).
6. Revenez au domaine local, choisissez l'onglet Connections (Connexions) et recherchez le tableau des connexions sortantes. Une fois la connexion entre les deux domaines active, un point de terminaison devient disponible dans la colonne Endpoint (Point de terminaison) du tableau. Copiez le point de terminaison.
7. Ouvrez le tableau de bord du domaine local et sélectionnez Dev Tools (Outils du développeur) dans le menu de navigation de gauche. Pour vérifier que l'index de domaine distant n'existe pas encore sur votre domaine local, exécutez la requête GET suivante. *remote-domain-index-name*Remplacez-le par votre propre nom d'index.

   ```
   GET remote-domain-index-name/_search
   {
      "query":{
         "match_all":{}
      }
   }
   ```

   Dans le résultat, vous devriez voir une erreur indiquant que l'index n'a pas été trouvé.

8. Sous votre demande GET, créez une demande POST et utilisez votre point de terminaison comme hôte distant, comme suit.

```
POST _reindex
{
   "source":{
      "remote":{
         "host":"connection-endpoint",
         "username":"username",
         "password":"password"
      },
      "index":"remote-domain-index-name"
   },
   "dest":{
      "index":"local-domain-index-name"
   }
}
```

   Exécutez cette demande.

9. Exécutez à nouveau la demande GET. La sortie devrait maintenant indiquer que l'index local existe. Vous pouvez interroger cet index pour vérifier que toutes les données de l'index distant ont OpenSearch été copiées.

Réindexer les données avec les opérations de l'API OpenSearch de service

Vous pouvez utiliser la réindexation à distance avec l'API pour copier des index entre deux domaines partageant une connexion de point de terminaison VPC.

1. Utilisez l'opération CreateOutboundConnectionAPI pour demander une nouvelle connexion entre votre domaine local et votre domaine distant.

```
POST https://es.region.amazonaws.com/2021-01-01/opensearch/cc/outboundConnection

{
   "ConnectionAlias": "remote-reindex-example",
   "ConnectionMode": "VPC_ENDPOINT",
   "LocalDomainInfo": {
      "AWSDomainInformation": {
         "DomainName": "local-domain-name",
         "OwnerId": "aws-account-id",
         "Region": "region"
```

```
      }
   },
   "RemoteDomainInfo": {
      "AWSDomainInformation": {
         "DomainName": "remote-domain-name",
         "OwnerId": "aws-account-id",
         "Region": "region"
      }
   }
}
```

Vous recevez un `ConnectionId` dans la réponse. Enregistrez cet identifiant pour l'utiliser à l'étape suivante.

2. Utilisez l'opération AcceptInboundConnectionAPI avec votre identifiant de connexion pour approuver la demande provenant du domaine local.

```
PUT https://es.region.amazonaws.com/2021-01-01/opensearch/cc/
inboundConnection/ConnectionId/accept
```

3. Utilisez l'opération DescribeOutboundConnectionsAPI pour récupérer le point de terminaison de votre domaine distant.

```
{
    "Connections": [
        {
            "ConnectionAlias": "remote-reindex-example",
            "ConnectionId": "connection-id",
            "ConnectionMode": "VPC_ENDPOINT",
            "ConnectionProperties": {
                "Endpoint": "connection-endpoint"
            },
            ...
        }
    ]
}
```

Enregistrez le *connection-endpoint* pour l'utiliser à l'étape 5.

4. Pour vérifier que l'index de domaine distant n'existe pas encore sur votre domaine local, exécutez la requête GET suivante. *remote-domain-index-name*Remplacez-le par votre propre nom d'index.

```
GET local-domain-endpoint/remote-domain-index-name/_search
{
   "query":{
      "match_all":{}
   }
}
```

Dans le résultat, vous devriez voir une erreur indiquant que l'index n'a pas été trouvé.

5. Créez une requête POST et utilisez votre point de terminaison comme hôte distant, comme suit.

```
POST local-domain-endpoint/_reindex
{
   "source":{
      "remote":{
         "host":"connection-endpoint",
         "username":"username",
         "password":"password"
      },
      "index":"remote-domain-index-name"
   },
   "dest":{
      "index":"local-domain-index-name"
   }
}
```

Exécutez cette demande.

6. Exécutez à nouveau la demande GET. La sortie devrait maintenant indiquer que l'index local existe. Vous pouvez interroger cet index pour vérifier que toutes les données de l'index distant ont OpenSearch été copiées.

Si le domaine distant est hébergé dans un VPC et que vous ne souhaitez pas utiliser la fonctionnalité de connexion du point de terminaison du VPC, vous devez configurer un proxy avec un point de terminaison accessible au public. Dans ce cas, le OpenSearch service nécessite un point de terminaison public car il n'est pas en mesure d'envoyer du trafic vers votre VPC.

Lorsque vous exécutez un domaine en mode VPC, un ou plusieurs points de terminaison sont placés dans votre VPC. Toutefois, ces points de terminaison ne sont destinés qu'au trafic entrant dans le domaine au sein du VPC, et ils n'autorisent pas le trafic à entrer dans le VPC lui-même.

La commande de réindexation à distance est exécutée depuis le domaine local, de sorte que le trafic d'origine ne peut pas utiliser ces points de terminaison pour accéder au domaine distant. C'est pourquoi un proxy est requis dans ce cas d'utilisation. Le domaine proxy doit disposer d'un certificat signé par une autorité de certification publique (CA). Les certificats auto-signés ou signés par une autorité de certification privée ne sont pas pris en charge.

Réindexer les données entre les domaines non liés OpenSearch aux services

Si l'index distant est hébergé en dehors de OpenSearch Service, par exemple dans une EC2 instance autogérée, définissez le `external` paramètre sur : `true`

```
POST _reindex
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443",
      "username": "username",
      "password": "password",
      "external": true
    },
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

Dans ce cas, seule l'authentification de base avec un nom d'utilisateur et un mot de passe est prise en charge. Le domaine distant doit disposer d'un point de terminaison accessible au public (même s'il se trouve dans le même VPC que le domaine de OpenSearch service local) et d'un certificat signé par une autorité de certification publique. Les certificats autosignés ou signés par une autorité de certification privée ne sont pas pris en charge.

Réindexer des jeux de données volumineux

La réindexation à distance envoie une demande de défilement au domaine distant avec les valeurs par défaut suivantes :

- Contexte de recherche de 5 minutes

- Délai d'attente de socket de 30 secondes
- Taille de lot de 1 000

Nous vous recommandons de régler ces paramètres en fonction de vos données. Pour les documents volumineux, envisagez une taille de lot plus petite, un délai d'attente and/or plus long. Pour plus d'informations, voir Paginer les résultats.

```
POST _reindex?pretty=true&scroll=10h&wait_for_completion=false
{
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443",
      "socket_timeout": "60m"
    },
    "size": 100,
    "index": "remote_index"
  },
  "dest": {
    "index": "local_index"
  }
}
```

Nous vous recommandons également d'ajouter les paramètres suivants à l'index local pour de meilleures performances :

```
PUT local_index
{
  "settings": {
    "refresh_interval": -1,
    "number_of_replicas": 0
  }
}
```

Une fois le processus de réindexation terminé, vous pouvez définir le nombre de réplicas souhaité et supprimer le paramètre d'intervalle d'actualisation.

Pour réindexer uniquement un sous-ensemble de documents sélectionnés par le biais d'une requête, envoyez cette demande au domaine local :

```
POST _reindex
{
```

```
  "source": {
    "remote": {
      "host": "https://remote-domain-endpoint:443"
    },
    "index": "remote_index",
    "query": {
      "match": {
        "field_name": "text"
      }
    }
  },
  "dest": {
    "index": "local_index"
  }
}
```

La réindexation à distance ne prend pas en charge le découpage et dès lors, vous ne pouvez pas effectuer plusieurs opérations de défilement pour la même demande en parallèle.

Paramètres de réindexation à distance

Outre les options de réindexation standard, OpenSearch Service prend en charge les options suivantes :

Options	Valeurs valides	Description	Obligatoire
Externe	Booléenne	Si le domaine distant n'est pas un domaine de OpenSearc h service ou si vous effectuez une réindexation entre deux domaines VPC, spécifiez comme. `true`	Non
Région	Chaîne	Si le domaine distant se trouve dans une autre région, spécifiez le nom de la région.	Non

Gestion des données en séries chronologiques dans Amazon OpenSearch Service avec des flux de données

Un flux de travail typique pour gérer les données en séries chronologiques comporte plusieurs étapes, telles que la création d'un alias d'index de substitution, la définition d'un index d'écriture et la définition de mappages et de paramètres communs pour les index de support.

Les flux de données dans Amazon OpenSearch Service aident à simplifier ce processus de configuration initial. Les flux de données sont prêts à l'emploi pour les données temporelles telles que les journaux d'application qui sont généralement de nature append-only.

Les flux de données nécessitent 1.0 ou une OpenSearch version ultérieure.

> **Note**
>
> Cette documentation fournit des étapes de base pour vous aider à démarrer avec les flux de données sur un domaine Amazon OpenSearch Service. Pour une documentation complète, consultez la section Flux de données dans la OpenSearch documentation.

Premiers pas avec les flux de données

Un flux de données est composé en interne de plusieurs index de support. Les requêtes de recherche sont acheminées vers tous les index de support, tandis que les requêtes d'indexation sont acheminées vers le dernier index d'écriture.

Étape 1 : Créer un modèle d'index

Pour créer un flux de données, vous devez d'abord créer un modèle d'index qui configure un ensemble d'index en tant que flux de données. L'objet `data_stream` indique qu'il s'agit d'un flux de données et non d'un modèle d'index ordinaire. Le modèle d'index correspond au nom du flux de données :

```
PUT _index_template/logs-template
{
  "index_patterns": [
    "my-data-stream",
    "logs-*"
  ],
```

```
  "data_stream": {},
  "priority": 100
}
```

Dans ce cas, chaque document ingéré doit avoir un champ `@timestamp`. Vous pouvez également définir votre propre champ d'horodatage personnalisé comme une propriété de l'objet `data_stream` :

```
PUT _index_template/logs-template
{
  "index_patterns": "my-data-stream",
  "data_stream": {
    "timestamp_field": {
      "name": "request_time"
    }
  }
}
```

Étape 2 : Création d'un flux de données

Après avoir créé un modèle d'index, vous pouvez directement commencer à ingérer des données sans créer de flux de données.

Comme nous avons un modèle d'index correspondant avec un `data_stream` objet, il crée OpenSearch automatiquement le flux de données :

```
POST logs-staging/_doc
{
  "message": "login attempt failed",
  "@timestamp": "2013-03-01T00:00:00"
}
```

Étape 3 : Ingestion des données dans le flux de données

Pour ingérer des données dans un flux de données, vous pouvez utiliser l'indexation APIs habituelle. Assurez-vous que chaque document que vous indexez possède un champ d'horodatage. Si vous essayez d'ingérer un document qui ne possède pas de champ d'horodatage, vous obtenez une erreur.

```
POST logs-redis/_doc
```

```
{
  "message": "login attempt",
  "@timestamp": "2013-03-01T00:00:00"
}
```

Étape 4 : Recherche dans un flux de données

Vous pouvez effectuer une recherche dans un flux de données de la même manière que dans un index ordinaire ou un alias d'index. L'opération de recherche s'applique à tous les index de support (toutes les données présentes dans le flux).

```
GET logs-redis/_search
{
  "query": {
    "match": {
      "message": "login"
    }
  }
}
```

Étape 5 : Substitution d'un flux de données

Vous pouvez configurer une politique ISM (Index State Management) afin d'automatiser le processus de substitution pour le flux de données. La politique ISM est appliquée aux index de support au moment de leur création. Lorsque vous associez une politique à un flux de données, elle n'affecte que les futurs index de support de ce flux de données. Vous n'avez pas non plus besoin de fournir le paramètre `rollover_alias`, car la politique ISM déduit cette information de l'index de support.

> **Note**
>
> Si vous sustituez un index de support vers un stockage à froid, OpenSearch supprimez cet index du flux de données. Même si vous déplacez à nouveau l'index vers UltraWarm, il reste indépendant et ne fait pas partie du flux de données d'origine. Une fois qu'un index a été supprimé du flux de données, la recherche par rapport au flux ne renvoie aucune donnée de l'index.

> **Warning**
>
> L'index d'écriture d'un flux de données ne peut pas être migré vers un stockage à froid. Si vous souhaitez migrer les données de votre flux de données vers un stockage à froid, vous devez transférer le flux de données avant la migration.

Étape 6 : Gestion des flux de données dans OpenSearch Dashboards

Pour gérer les flux de données à partir de OpenSearch tableaux de bord, ouvrez les OpenSearchtableaux de bord, choisissez Gestion des index, sélectionnez Indices ou Indices gérés par des politiques.

Étape 7 : Suppression d'un flux de données

L'opération de suppression supprime d'abord les index de support d'un flux de données, puis supprime le flux de données lui-même.

Pour supprimer un flux de données et tous ses index de support cachés :

```
DELETE _data_stream/name_of_data_stream
```

Surveillance des données dans Amazon OpenSearch Service

Surveillez vos données de manière proactive dans Amazon OpenSearch Service grâce aux alertes et à la détection des anomalies. Configurez des alertes pour recevoir des notifications lorsque vos données dépassent certains seuils. La détection des anomalies utilise le machine learning pour détecter automatiquement les valeurs hors normes dans vos données de streaming. Vous pouvez associer la détection des anomalies à des alertes pour être averti dès qu'une anomalie est détectée.

Rubriques

- Configuration des alertes dans Amazon OpenSearch Service
- Détection des anomalies dans Amazon OpenSearch Service OpenSearch

Configuration des alertes dans Amazon OpenSearch Service

Configurez des alertes dans Amazon OpenSearch Service pour être averti lorsque les données d'un ou de plusieurs index répondent à certaines conditions. Par exemple, vous pouvez souhaiter recevoir un e-mail si votre application journalise plus de cinq erreurs HTTP 503 en une heure, ou souhaiter appeler un développeur si aucun nouveau document n'a été indexé au cours des 20 dernières minutes.

Les alertes nécessitent Elasticsearch 6.2 OpenSearch ou version ultérieure.

> **Note**
>
> Cette documentation fournit un bref aperçu des alertes et explique en quoi les alertes sur un domaine Amazon OpenSearch Service diffèrent des alertes sur un cluster open source. OpenSearch Pour obtenir une documentation complète sur les alertes, y compris une référence complète sur les API, une liste des champs de requête disponibles pour les moniteurs composites et les descriptions des variables de déclenchement et d'action disponibles, consultez la section Alertes dans la OpenSearch documentation.

Rubriques

- Autorisations relatives aux alertes

- Démarrer avec les alertes
- Notifications
- Différences

Autorisations relatives aux alertes

La fonctionnalité d'alerte prend en charge le contrôle précis des accès. Pour en savoir plus sur le mixage et l'appariement des autorisations en fonction de votre cas d'utilisation, consultez la section Sécurité des alertes dans la OpenSearch documentation.

Pour accéder à la page d'alerte dans les OpenSearch tableaux de bord, vous devez au moins être mappé au rôle `alerting_read_access` prédéfini ou disposer d'autorisations équivalentes. Ce rôle accorde des autorisations pour afficher les alertes, les destinations et les moniteurs, mais pas pour accuser réception des alertes ou modifier les destinations ou les moniteurs.

Démarrer avec les alertes

Pour créer une alerte, vous configurez un moniteur, c'est-à-dire une tâche exécutée selon un calendrier défini et interrogeant OpenSearch des index. Vous configurez également un ou plusieurs déclencheurs, qui définissent les conditions qui génèrent des événements. Enfin, vous configurez des actions, c'est-à-dire ce qu'il se passe lorsqu'une alerte se déclenche.

Pour commencer à utiliser des alertes

1. Choisissez Alertes dans le menu principal OpenSearch des tableaux de bord, puis choisissez Créer un moniteur.
2. Créez un moniteur par requête, par compartiment, par métrique de cluster ou par document. Pour plus d'instructions, veuillez consulter la rubrique Créer un moniteur (langue française non garantie).
3. Dans Triggers (Déclencheurs), créez un ou plusieurs déclencheurs. Pour consulter les instructions, veuillez consulter la rubrique Créer des déclencheurs (langue française non garantie).
4. Pour Actions, configurez un canal de notification pour l'alerte. Choisissez Slack, Amazon Chime, un webhook personnalisé ou Amazon SNS. Comme vous pouvez vous en douter, les notifications nécessitent une connectivité au canal. Par exemple, pour notifier une chaîne Slack ou envoyer un webhook personnalisé à un serveur tiers, votre domaine de OpenSearch service

a besoin d'une connectivité réseau appropriée. Le webhook personnalisé doit avoir une adresse IP publique pour qu'un domaine de OpenSearch service puisse lui envoyer des alertes.

> **Tip**
>
> Une fois qu'une action a réussi à envoyer un message, la sécurisation de l'accès à ce message (par exemple, l'accès à un canal Slack) est de votre responsabilité. Si votre domaine contient des données sensibles, vous pouvez utiliser des déclencheurs sans actions et rechercher périodiquement les alertes dans Dashboards.

Notifications

Les alertes s'intègrent aux notifications, qui constituent un système unifié de OpenSearch notifications. Les notifications vous permettent de configurer le service de communication que vous souhaitez utiliser et de consulter les statistiques pertinentes et les informations de dépannage. Pour une documentation complète, voir Notifications dans la OpenSearch documentation.

Votre domaine doit exécuter OpenSearch la version 2.3 ou ultérieure pour utiliser les notifications.

> **Note**
>
> OpenSearch les notifications sont distinctes des notifications de OpenSearch service, qui fournissent des détails sur les mises à jour du logiciel de service, les améliorations apportées à Auto-Tune et d'autres informations importantes au niveau du domaine. OpenSearch les notifications sont spécifiques au plugin.

Les canaux de notification ont remplacé les destinations d'alerte à partir de OpenSearch la version 2.0. Les destinations sont officiellement obsolètes et toutes les notifications d'alerte seront désormais gérées via des canaux.

Lorsque vous mettez à niveau vos domaines vers la version 2.3 ou ultérieure (puisque le support de OpenSearch service pour 2.x commence avec 2.3), vos destinations existantes sont automatiquement migrées vers les canaux de notification. Si la migration d'une destination échoue, le moniteur continuera à l'utiliser jusqu'à ce qu'il soit migré vers un canal de notification. Pour plus d'informations, consultez la section Questions sur les destinations dans la OpenSearch documentation.

Pour commencer à utiliser les notifications, connectez-vous aux OpenSearch tableaux de bord et choisissez Notifications, Canaux et Créer un canal.

Amazon Simple Notification Service (Amazon SNS) est un type de canal pris en charge pour les notifications. Afin d'authentifier les utilisateurs, vous devez soit leur fournir un accès complet à Amazon SNS, soit les laisser endosser un rôle IAM autorisé à accéder à Amazon SNS. Pour obtenir des instructions, veuillez consulter la rubrique Amazon SNS en tant que type de canal (langue française non garantie).

Différences

Par rapport à la version open source de OpenSearch, les alertes d'Amazon OpenSearch Service présentent des différences notables.

Paramètres d'alerte

OpenSearch Le service vous permet de modifier les paramètres d'alerte suivants :

- `plugins.scheduled_jobs.enabled`
- `plugins.alerting.alert_history_enabled`
- `plugins.alerting.alert_history_max_age`
- `plugins.alerting.alert_history_max_docs`
- `plugins.alerting.alert_history_retention_period`
- `plugins.alerting.alert_history_rollover_period`
- `plugins.alerting.filter_by_backend_roles`

Tous les autres paramètres utilisent les valeurs par défaut que vous ne pouvez pas modifier.

Pour désactiver les alertes, envoyez la requête suivante :

```
PUT _cluster/settings
{
  "persistent" : {
    "plugins.scheduled_jobs.enabled" : false
  }
}
```

La demande suivante configure les alertes pour supprimer automatiquement les index d'historique après sept jours, au lieu des 30 jours par défaut :

```
PUT _cluster/settings
{
  "persistent": {
    "plugins.alerting.alert_history_retention_period": "7d"
  }
}
```

Si vous avez déjà créé des moniteurs et que vous souhaitez arrêter la création d'index d'alertes quotidiens, supprimez tous les index de l'historique des alertes :

```
DELETE .plugins-alerting-alert-history-*
```

Pour réduire le nombre de partitions pour les index historiques, créez un modèle d'index. La requête suivante définit les index d'historique pour les alertes sur une partition et un réplica :

```
PUT _index_template/template-name
{
  "index_patterns": [".opendistro-alerting-alert-history-*"],
  "template": {
    "settings": {
      "number_of_shards": 1,
      "number_of_replicas": 1
    }
  }
}
```

En fonction de votre tolérance à la perte de données, vous pouvez même envisager de n'utiliser aucun réplica. Pour plus d'informations sur la création et la gestion de modèles d'index, consultez la section Modèles d'index dans la OpenSearch documentation.

Détection des anomalies dans Amazon OpenSearch Service OpenSearch

La fonctionnalité de détection des anomalies d'Amazon OpenSearch OpenSearch Service détecte automatiquement et en temps quasi réel les anomalies OpenSearch présentes dans vos données OpenSearch à l'aide de l'algorithme Random Cut Forest (RCF). RCF est un algorithme de machine learning non supervisé qui modélise un croquis de votre flux de données entrant. Cet algorithme calcule une valeur `anomaly grade` et `confidence score` pour chaque point de données entrant.

La fonctionnalité de détection des anomalies utilise ces valeurs pour différencier une anomalie des variations normales de vos données.

Vous pouvez jumeler le plugin de détection des anomalies avec le plugin pour vous avertir dès la détection des anomalies.

La fonctionnalité de détection des anomalies prend en charge le contrôle précis des accès, ou n'importe quelle OpenSearch version d'OpenSearch. Tous les types d'instances prennent en charge la détection des anomalies excepté `t2.micro` et `t2.small`.

> **Note**
>
> Cette documentation fournit un bref aperçu de la détection des anomalies dans le contexte d'Amazon OpenSearch Service. Pour une documentation complète, y compris les étapes détaillées, une référence d'API, une référence de tous les paramètres disponibles et les étapes de création de visualisations et de tableaux de bord, consultez la section Détection des anomalies dans la documentation open source. OpenSearch

Prérequis

Les prérequis suivants s'appliquent à la fonctionnalité de détection des anomalies :

- La fonctionnalité de détection des anomalies requiert Elasticsearch 7.4 OpenSearch ou version ultérieure.
- La fonctionnalité de détection des anomalies prend en charge le contrôle précis des accès à partir d'Elasticsearch versions 7.9 et ultérieures, et de toutes les versions d'OpenSearch. OpenSearch Avant la version 7.9 d'Elasticsearch, seuls les utilisateurs administrateurs peuvent créer, afficher et gérer les détecteurs.
- Si votre domaine utilise le contrôle précis des accès, les utilisateurs non administrateurs doivent être mappés au `anomaly_read_access` rôle anomaly_access dans OpenSearch les tableaux de bord OpenSearch afin de créer et de `anomaly_full_access` gérer les détecteurs.

Mise en route avec la détection des anomalies

Pour commencer, choisissez Anomaly Detection (Détection des anomalies) dans OpenSearch OpenSearch Dashboards.

Étape 1 : Créer un détecteur

Un détecteur est une tâche individuelle de détection d'anomalie. Vous pouvez créer plusieurs détecteurs, lesquels peuvent fonctionner simultanément, chacun d'entre eux analysant des données provenant de sources différentes.

Étape 2 : Ajouter des fonctionnalités à votre détecteur

Le terme « fonctionnalité » désigne le champ de votre index dans lequel vous recherchez les anomalies. Un détecteur peut détecter des anomalies sur une ou plusieurs fonctionnalités. Vous devez choisir une des agrégations suivantes pour chaque fonctionnalité : `average()`, `sum()`, `count()`, `min()` ou `max()`.

> **Note**
>
> La méthode `count()` d'agrégation n'est disponible que dans OpenSearch OpenSearch et à partir de la version 7.7 d'Elasticsearch. Pour Elasticsearch version 7.4, utilisez une expression personnalisée comme celle-ci :
>
> ```
> {
>   "aggregation_name": {
>      "value_count": {
>         "field": "field_name"
>      }
>   }
> }
> ```

La méthode d'agrégation détermine ce qui constitue une anomalie. Par exemple, si vous choisissez `min()`, le détecteur se concentre sur la recherche d'anomalies en se basant sur les valeurs minimales de votre fonctionnalité. Si vous choisissez `average()`, le détecteur détecte des anomalies en se basant sur les valeurs moyennes de votre fonctionnalité. Vous pouvez ajouter un maximum de cinq fonctionnalités par détecteur.

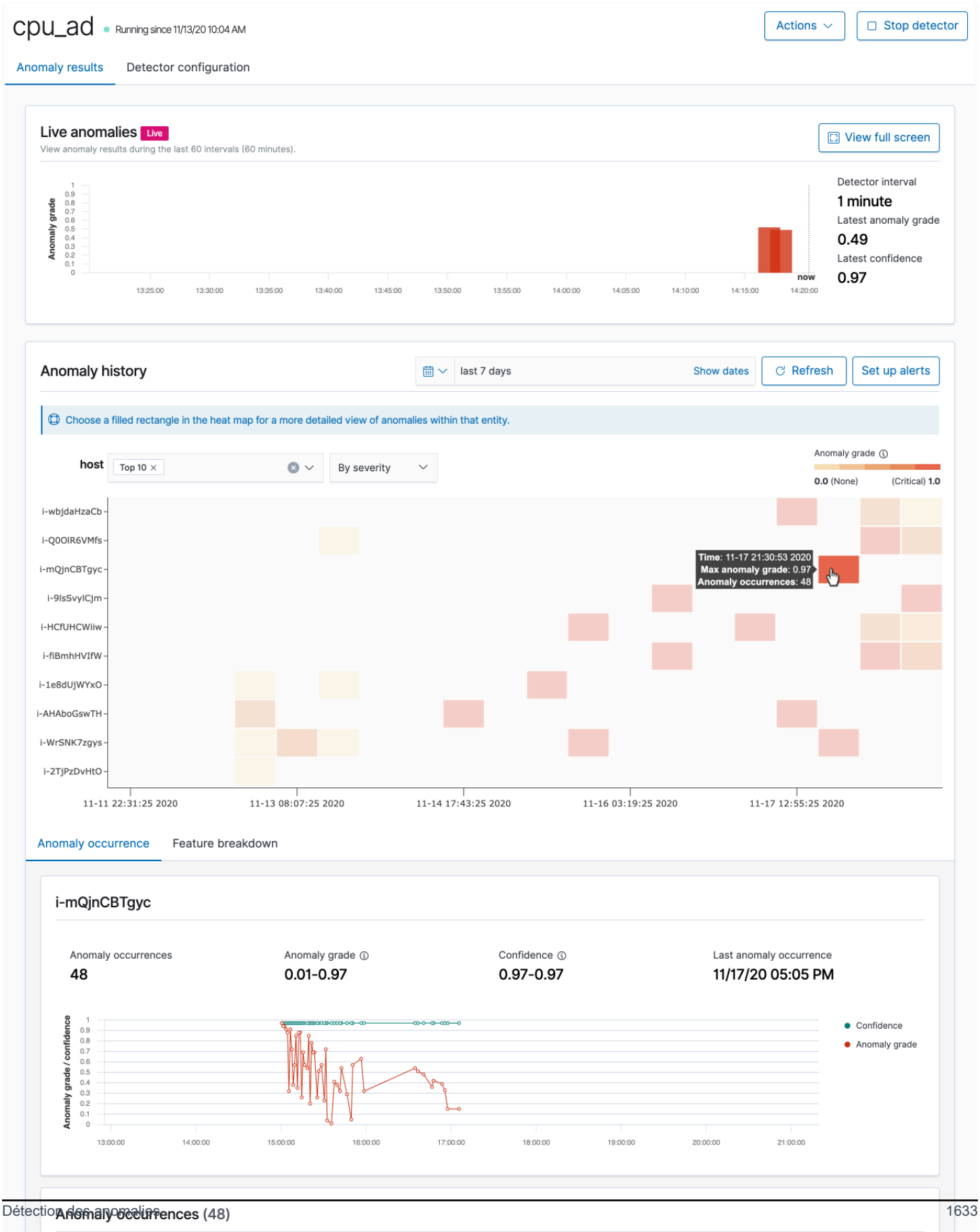
Vous pouvez configurer les paramètres facultatifs suivants (disponibles à partir de la version 7.7 d'Elasticsearch) :

- Champ Catégorie : classez ou découpez vos données à l'aide d'une dimension telle que l'adresse IP, l'ID du produit, le code du pays, etc.

- Taille de la fenêtre : définissez le nombre d'intervalles d'agrégation de votre flux de données à prendre en compte dans une fenêtre de détection.

Après avoir configuré vos fonctionnalités, prévisualisez des exemples d'anomalies et, si nécessaire, ajustez les paramètres des fonctionnalités.

Étape 3 : Observer les résultats



- Anomalies en direct : affiche les résultats des anomalies en direct pour les 60 derniers intervalles. Par exemple, si l'intervalle est défini sur 10, il affiche les résultats des 600 dernières minutes. Ce graphique est mis à jour toutes les 30 secondes.
- Historique des anomalies : représente le degré d'anomalie avec la métrique de confiance correspondante.
- Ventilation des fonctionnalités : représente les fonctionnalités en fonction de la méthode d'agrégation. Vous pouvez faire varier la plage date-heure du détecteur.
- Occurrences des anomalies : montre le `Start time`, `End time`, `Data confidence` et `Anomaly grade` pour chaque anomalie détectée.

  Si vous définissez le champ de catégorie, vous accédez à un graphique Carte thermique supplémentaire qui met en corrélation les résultats relatifs aux entités anormales. Choisissez un rectangle rempli pour obtenir une vue plus détaillée de l'anomalie.

Étape 4 : Configurer des alertes

Pour créer un système de suivi capable de vous envoyer des notifications lors de la détection d'anomalies, choisissez Set up alerts (Configurer les alertes). Le plug-in vous redirige vers la page Ajouter un système de suivi où vous pouvez configurer une alerte.

Didacticiel : Détection d'une utilisation élevée de l'UC avec un détecteur d'anomalies

Ce didacticiel explique comment créer un détecteur d'anomalies dans Amazon OpenSearch OpenSearch Service pour détecter une utilisation élevée de l'UC. Vous utiliserez OpenSearch OpenSearch Dashboards pour configurer un détecteur afin de surveiller l'utilisation de l'UC et de générer une alerte lorsque l'utilisation de l'UC dépasse un seuil spécifié.

> **Note**
>
> Ces étapes s'appliquent à la dernière version d' OpenSearch OpenSearch et peuvent différer pour les versions antérieures.

Prérequis

- Vous devez disposer d'un domaine OpenSearch OpenSearch Service exécutant Elasticsearch 7.4 ou une version ultérieure. OpenSearch
- Vous devez ingérer des fichiers journaux d'application dans votre cluster qui contiennent des données d'utilisation de l'UC.

Étape 1 : Créer un détecteur

Tout d'abord, créez un détecteur qui identifie les anomalies dans vos données d'utilisation de l'UC.

1. Ouvrez le menu du panneau gauche dans OpenSearch OpenSearch Dashboards et choisissez Anomaly Detection (Détection des anomalies), puis choisissez Create monitor (Créer un détecteur).
2. Nommez le détecteur **`high-cpu-usage`**.
3. Pour la source de données, choisissez l'index qui contient les fichiers journaux d'utilisation de l'UC dans lesquels vous voulez identifier les anomalies.
4. Choisissez le champ Timestamp (Horodatage) de vos données. Si nécessaire, vous pouvez ajouter un filtre de données. Ce filtre de données analyse uniquement un sous-ensemble de la source de données et réduit le bruit des données qui ne sont pas pertinentes.
5. Définissez le Detector interval (Intervalle du détecteur) sur 2 minutes. Cet intervalle définit le temps (par intervalle de minutes) pendant lequel le détecteur collecte les données.
6. Dans Window delay (Délai de la fenêtre), ajoutez un délai de 1 minute. Ce délai ajoute un temps de traitement supplémentaire pour garantir la présence de toutes les données dans la fenêtre.
7. Choisissez Next (Suivant). Sur le tableau de bord de la détection d'anomalies, sous le nom du détecteur, choisissez Configure model (Configurer le modèle).
8. Pour Feature name (Nom de la fonction), saisissez **`max_cpu_usage`**. Pour Feature state (État de la fonction), sélectionnez Enable feature (Activer la fonction).
9. Pour Find anomalies based on (Rechercher les anomalies en fonction de), choisissez Field value (Valeur du champ).
10. Pour Aggregation method (Méthode d'agrégation), choisissez **`max()`**.
11. Pour Field (Champ), sélectionnez le champ de vos données à vérifier pour les anomalies. Par exemple, il peut s'appeler `cpu_usage_percentage`.
12. Conservez tous les autres paramètres par défaut et choisissez Next (Suivant).

13. Ignorez la configuration des tâches du détecteur et choisissez Next (Suivant).
14. Dans la fenêtre contextuelle, choisissez quand démarrer le détecteur (automatiquement ou manuellement), puis choisissez Confirm (Confirmer).

Maintenant que le détecteur est configuré, après son initialisation, vous pourrez voir les résultats en temps réel de l'utilisation de l'UC dans la section Real-time results (Résultats en temps réel) de votre panneau de détecteur. La section Live anomalies (Anomalies en direct) affiche toutes les anomalies qui se produisent pendant l'ingestion des données en temps réel.

Étape 2 : Configurer une alerte

Maintenant que vous avez créé un détecteur, créez un moniteur qui déclenche une alerte pour envoyer un message à Slack lorsqu'il détecte une utilisation de l'UC qui répond aux conditions spécifiées dans les paramètres du détecteur. Vous recevrez des notifications Slack lorsque les données d'un ou plusieurs index répondent aux conditions qui déclenchent l'alerte.

1. Ouvrez le menu du panneau gauche dans OpenSearch OpenSearch Dashboards et choisissez Alerting (Alerte), puis choisissez Create monitor (Créer un moniteur).
2. Donnez un nom au moniteur.
3. Pour Monitor type (Type de moniteur), choisissez Per-query monitor (Moniteur par requête). Un moniteur par requête exécute une requête spécifiée et définit les déclencheurs.
4. Pour Monitor defining method (Méthode de définition du moniteur), choisissez Anomaly detector (Détecteur d'anomalies), puis sélectionnez le détecteur que vous avez créé dans la section précédente dans le menu déroulant Detector (Détecteur).
5. Pour Schedule (Planification), choisissez la fréquence à laquelle le moniteur collecte des données et la fréquence à laquelle vous recevez des alertes. Pour les besoins de ce didacticiel, définissez la planification pour une exécution toutes les 7 minutes.
6. Dans la section Triggers (Déclencheurs), choisissez Add trigger (Ajouter un déclencheur). Pour Trigger name (Nom du déclencheur), saisissez **`High CPU usage`**. Pour ce didacticiel, pour Severity level (Niveau de gravité), choisissez 1, qui est le niveau de gravité le plus élevé.
7. Pour Anomaly grade threshold (Seuil du niveau d'anomalie), choisissez IS ABOVE (EST AU-DESSUS). Dans le menu situé en dessous, choisissez le seuil de niveau à appliquer. Pour ce didacticiel, définissez le Anomaly grade (Niveau d'anomalie) à 0,7.
8. Pour Anomaly confidence threshold (Seuil de confiance de l'anomalie), choisissez IS ABOVE (EST AU-DESSUS). Dans le menu situé sous cette option, saisissez le même nombre que votre niveau

d'anomalie. Pour ce didacticiel, définissez Anomaly confidence threshold (Seuil de confiance de l'anomalie) sur 0,7.

9. Dans la section Actions, choisissez Destination. Dans le champ Name (Nom), choisissez le nom de la destination. Dans le menu Type, choisissez Slack. Dans le champ Webhook URL (URL du Webhook), saisissez l'URL du webhook vers lequel vous voulez recevoir les alertes. Pour plus d'informations, consultez Envoi de messages en utilisant des webhooks entrants (Langue Français non garanti).
10. Sélectionnez Create (Créer).

Ressources connexes

- the section called “Alerte”

- the section called “Détection des anomalies”
- API de détection d'anomalies

Développeur Amazon Q pour Amazon OpenSearch Service

Amazon Developer est un service Amazon Search Service (RCF). Amazon Q aide AWS les clients à coder, tester, déployer, dépanner et optimiser leurs applications en AWS cours d'exécution.

L'intégration d'Amazon Q à Amazon OpenSearch Service offre les fonctionnalités génératives suivantes :

- the section called “Génération de visualisations en langage naturel”
- the section called “Afficher les résumés et les informations sur les alertes”
- the section called “Consultez les résumés des résultats de requêtes générés par Amazon Q sur la page Discover”
- the section called “Décrire les détecteurs d'anomalies”
- the section called “Accédez au chat Amazon Q pour les questions OpenSearch de service”

Pour accéder aux fonctionnalités d'Amazon Q Developer dans OpenSearch Service qui sont pertinentes pour votre tâche, recherchez l'icône contextuelle suivante sur les pages de visualisation Alertes, Discover et Create. Cliquez sur l'icône pour demander de l'aide et utiliser les fonctionnalités décrites dans cette section.



Pour plus d'informations sur la création d'une représentation visuelle, cliquez sur l'icône Création d' OpenSearch un service. Amazon Q Chat permet de répondre aux questions concernant les fonctionnalités et fonctionnalités du OpenSearch Service.

> **Note**
>
> Le chat Amazon Q ne peut pas accéder à vos données. Pour cette raison, vous ne pouvez pas engager le chatbot dans une conversation concernant vos données.

Pour plus d'informations sur Amazon Q, consultez Qu'est-ce qu'Amazon Q Developer ? Pour plus d'informations sur le guide de l'utilisateur Amazon Q

Soutenu Régions AWS

Le support Amazon Q pour le OpenSearch service est disponible dans les pays suivants Régions AWS :

- USA Est (Virginie du Nord)
- USA Ouest (Oregon)
- Asie-Pacifique (Mumbai)
- Asie-Pacifique (Sydney)
- Asie-Pacifique (Tokyo)
- Canada (Centre)
- Europe (Francfort)
- Europe (Londres)
- Europe (Paris)
- Amérique du Sud (São Paulo)

Configuration d'Amazon Q OpenSearch

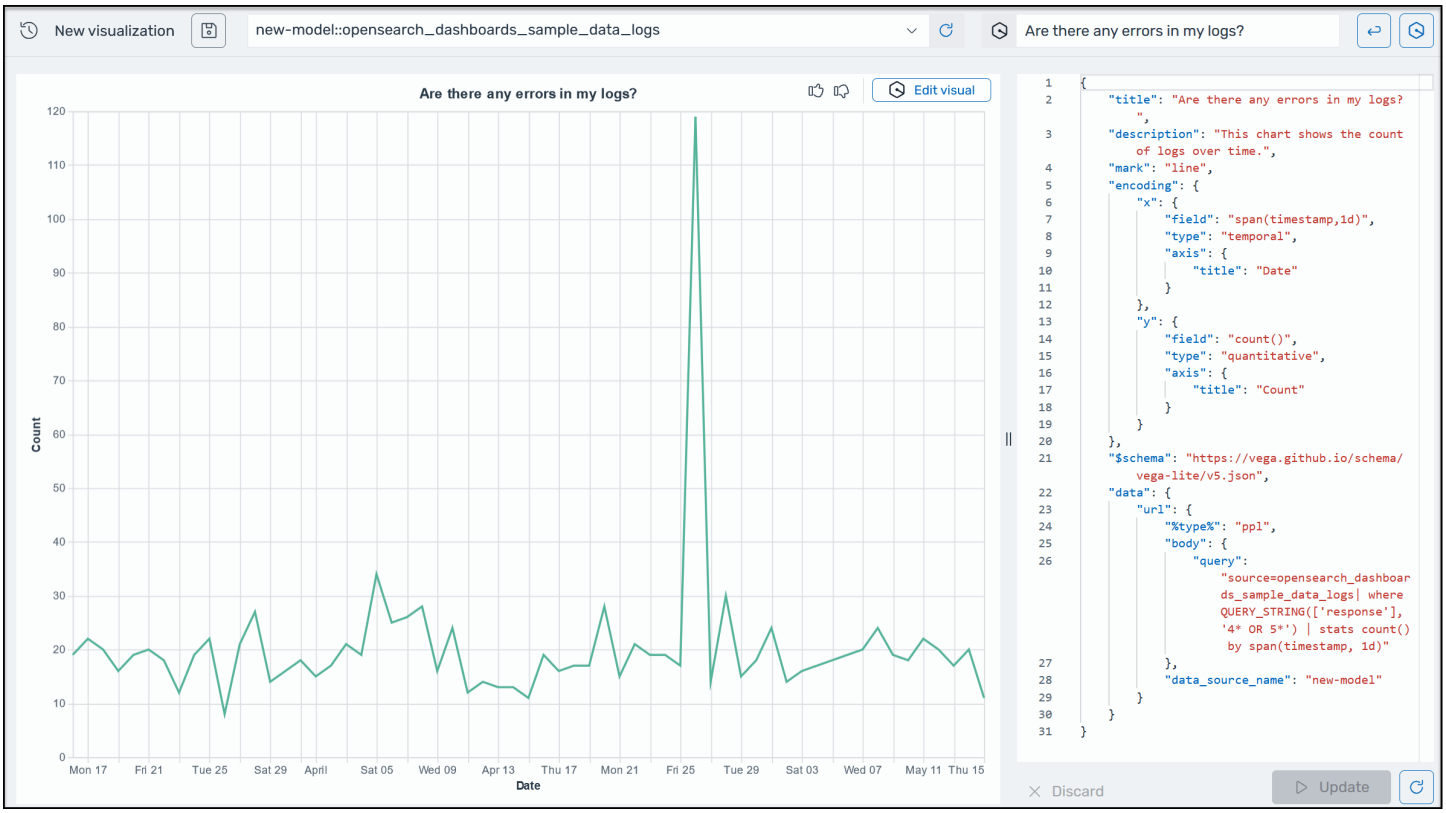
Procédez comme suit pour configurer un Amazon OpenSearch Elasticsearch.

1. Procédez comme suit pour configurer les conditions d'accès OpenSearch à l'aide de l'algorithme d'enregistrement. Pour de plus amples informations, veuillez consulter Contrôle d'accès précis dans Amazon Service OpenSearch .
2. Vérifiez que votre source de données utilise la OpenSearch version 2.17 ou ultérieure.
3. Vérifiez que vous avez coché la case Activer la génération de requêtes en langage naturel dans la section Intelligence artificielle (IA) et Machine Learning (ML) lors de la création du domaine ou en modifiant la configuration du cluster.

> **Note**
>
> Les OpenSearch fonctionnalités d'Amazon Q for Developer sont disponibles avec le niveau gratuit Q. Pour plus d'informations, consultez la section Comprendre les niveaux de service dans le guide de l'utilisateur Amazon Q.

Génération de visualisations en langage naturel

Pour vous aider à mieux comprendre vos données opérationnelles, Amazon Q Developer for OpenSearch Service prend en charge l'utilisation d'instructions en langage naturel pour créer des visualisations. Vous pouvez générer des visualisations comme dans l'exemple suivant en langage naturel à partir de la page Visualisations ou de la page Découvrir.



Les visualisations peuvent accélérer le dépannage en répartissant l'analyse des erreurs en fonction de différentes dimensions. Les visualisations peuvent également vous aider à identifier les modèles et les tendances, à accélérer la prise de décision, à révéler les relations et à simplifier les données complexes.

Pour générer une visualisation à l'aide du langage naturel

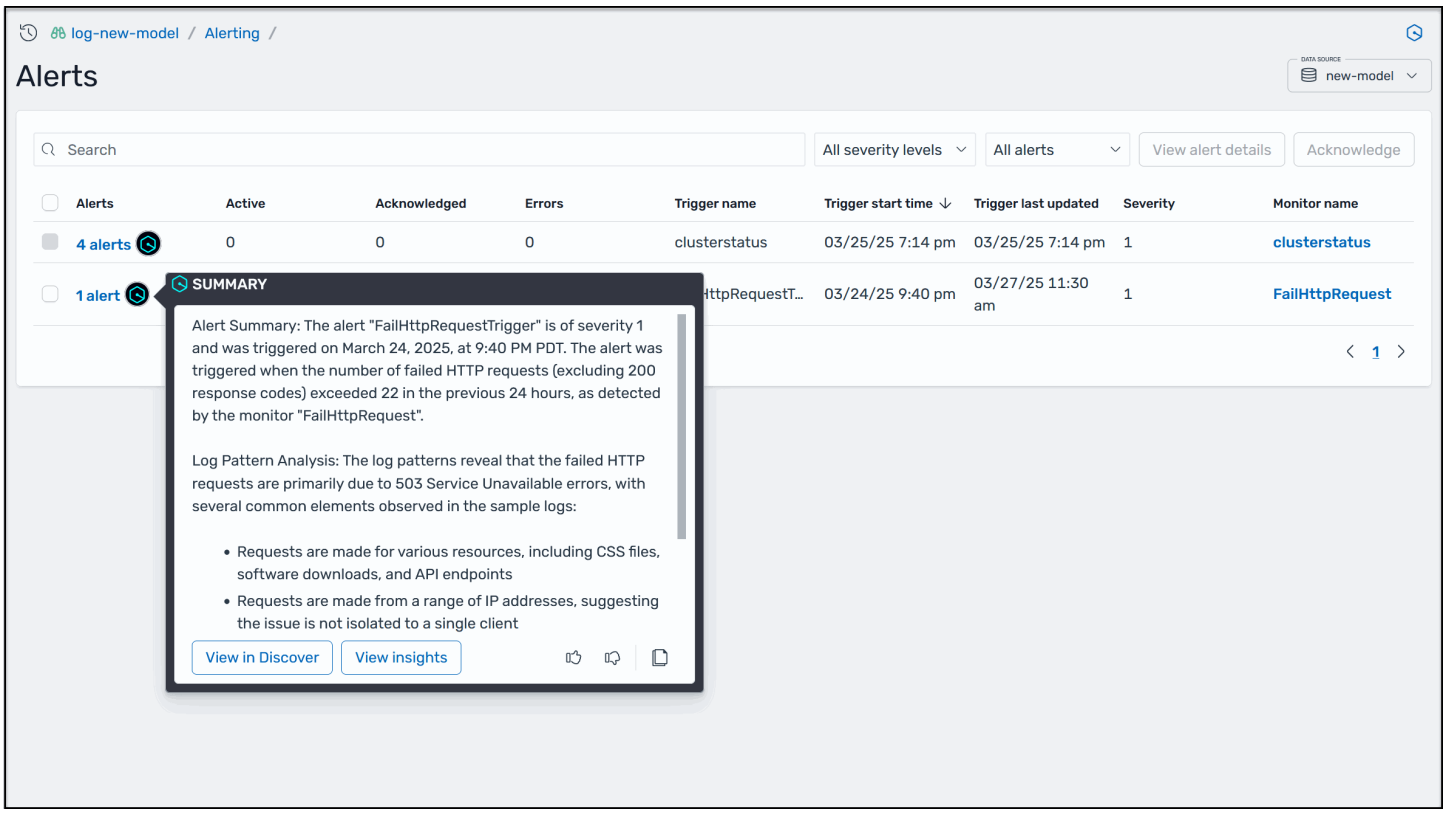
1. Vérifiez que vous avez configuré Amazon Q for OpenSearch Service.
2. Dans le menu principal OpenSearch des tableaux de bord, choisissez la page Découvrir, puis choisissez une source de données.
3. Dans le menu Amazon Q, choisissez Generate visualization, comme illustré dans la capture d'écran suivante.



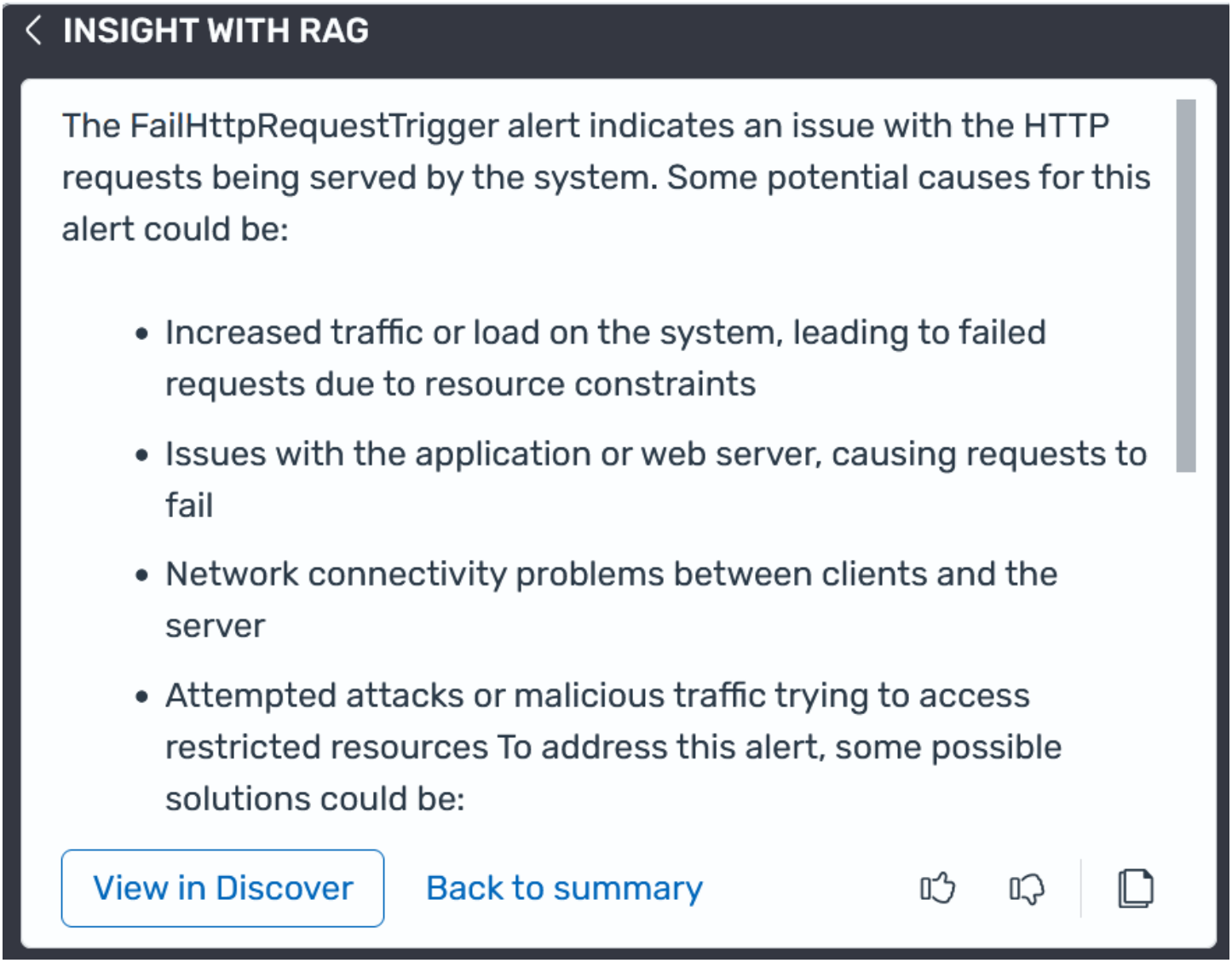
4. Si vous avez saisi une requête en langage naturel sur la page Découvrir, Amazon Q copie le contexte lors de la création d'une visualisation basée sur cette requête. Si vous n'avez pas encore saisi de requête en langage naturel, entrez une invite dans la zone de texte Amazon Q, puis cliquez sur le bouton à côté de la zone de texte pour exécuter la requête. L'ensemble du processus dure jusqu'à 10 minutes.
5. Pour plus d'informations sur le graphique situé dans le coin supérieur droit de la représentation visuelle, cliquez sur le bouton de téléchargement situé dans le coin supérieur droit de la représentation visuelle. Tapez une nouvelle invite, par exemple « Remplacer cela par un graphique linéaire », puis choisissez Appliquer.

Afficher les résumés et les informations sur les alertes

Pour plus d' OpenSearch informations sur les données d'un ou plusieurs index, consultez les données d'un ou plusieurs index. Pour vous aider à comprendre et à résoudre rapidement une alerte, vous pouvez consulter un résumé des alertes en cliquant sur l'icône Amazon Q Developer à côté de l'alerte. Un résumé fournit des détails sur le problème sous-jacent qui a déclenché l'alerte et, le cas échéant, une analyse supplémentaire pour vous aider à identifier la cause première du problème. La capture d'écran suivante illustre un exemple de détection d'anomalies.



Si vous connectez une base de connaissances pour fournir un contexte supplémentaire sur votre environnement, comme décrit plus loin dans cette rubrique, Amazon Q crée des informations sur une alerte. Insights fournit des informations détaillées et des options de dépannage pour vous aider à remédier à la cause première d'une alerte. Pour l'alerte précédemment affichée, Amazon Q a également produit les informations suivantes.



> **Note**
>
> En fonction de la nature de l'alerte et des informations disponibles, Amazon Q peut vous donner la possibilité de consulter les données d'alerte sur la page Découvrir OpenSearch des tableaux de bord. Si le bouton Afficher dans Discover apparaît au bas du résumé d'une alerte Amazon Q, cliquez sur le bouton pour ouvrir l'ensemble de données correspondant dans Discover avec un filtre actif pour les données d'alerte.

Rubriques

- Avant de commencer
- Affichage des résumés et des informations sur les alertes

Avant de commencer

Procédez comme suit pour configurer une base de connaissances Amazon Bedrock afin qu'Amazon Q puisse créer des informations pour les alertes OpenSearch de service.

Étape 1 : Créer le LambdaInvokeOpenSearch MLCommons rôle IAM IAM

Créez un nouveau rôle nommé `LambdaInvokeOpenSearchMLCommonsRole` dans AWS Identity and Access Management (IAM). OpenSearch Le service utilise ce rôle pour créer un connecteur AI OpenSearch qui permet de produire des informations basées sur des articles de base de connaissances configurés. Vous devez associer ce rôle au `ml_full_access` rôle OpenSearch Service, comme décrit à l'étape 2.

Lorsque vous créez le nouveau rôle, pour Type d'entité de confiance, choisissez le AWS compte. Vous n'avez pas besoin d'indiquer de profil. Sur la page Ajouter des autorisations, sélectionnez Suivant. Pour plus d'informations sur la création d'un rôle pour un service (console), consultez Création d'un rôle pour un AWS service (console).

Étape 2 : Associer le LambdaInvokeOpenSearch MLCommons rôle Role au rôle OpenSearch Service ml_full_access

Utilisez la procédure suivante pour associer le `LambdaInvokeOpenSearchMLCommonsRole` rôle au `ml_full_access` rôle OpenSearch Service. Ce mappage aide également le OpenSearch Service à créer le connecteur AI.

Pour mapper le rôle IAM requis au rôle OpenSearch Service ml_full_access

1. Ouvrez la page d'administration des données du tableau de bord des OpenSearch services.
2. Sous Accès aux données et utilisateur, sélectionnez Rôles.
3. Utilisez le champ de recherche pour afficher le `ml_full_access` guide.
4. Sur la page ml_full_access, choisissez l'onglet Utilisateurs mappés.
5. Choisissez Map users.
6. Dans le champ Rôles principaux, collez le nom de ressource Amazon (ARN) du `LambdaInvokeOpenSearchMLCommonsRole` rôle, puis choisissez Map.

Étape 3 : configurer une base de connaissances OpenSearch de service à l'aide de AWS CloudFormation

Utilisez la procédure suivante pour configurer une base OpenSearch de connaissances sur les services AWS CloudFormation afin qu'Amazon Q puisse générer des informations.

Pour configurer une base de connaissances pour obtenir des informations

1. Connectez-vous à la page d'https://console.aws.amazon.com/aos/accueil de la console Amazon OpenSearch Service dans un environnement compatible Région AWS. Pour de plus amples informations, veuillez consulter Soutenu Régions AWS.
2. Dans le volet de navigation, choisissez Intégrations.
3. Dans la section Modèles d'intégration, choisissez le modèle Intégrer à la base de connaissances via Amazon Bedrock. Par conséquent, vous n'avez pas besoin d'indiquer que vous avez saisi le bon code d'enregistrement.
4. Dans la vignette Intégrer à la base de connaissances via Amazon Bedrock, choisissez Configurer le domaine, puis choisissez l'une des options disponibles. OpenSearch Le service ouvre le modèle de AWS CloudFormation pile avec les champs obligatoires préremplis. La AWS CloudFormation pile prend en charge l'intégration pour les domaines publics et VPC.
5. Sélectionnez Créer la pile. Après avoir AWS CloudFormation créé les ressources, le service affiche l'agent Amazon Bedrock AgentIdConnectorId, et ModelId.

Le cas échéant, Amazon Q crée désormais des informations pour les alertes OpenSearch de service.

Affichage des résumés et des informations sur les alertes

Suivez la procédure suivante pour afficher les détecteurs d'anomalies OpenSearch .

Affichage des résumés et des informations sur les alertes

1. Vérifiez que vous avez configuré Amazon Q for OpenSearch Service.
2. Vérifiez que vous avez configuré des alertes pour le OpenSearch service.
3. Dans le menu principal OpenSearch des tableaux de bord, choisissez Alertes, puis Alertes.
4. Choisissez l'icône Amazon Q à côté d'une alerte. L'ensemble du processus dure jusqu'à 10 minutes.
5. Si cela figure dans le résumé de l'alerte, choisissez Afficher les informations pour afficher plus de détails sur l'alerte en fonction de votre base de connaissances configurée.

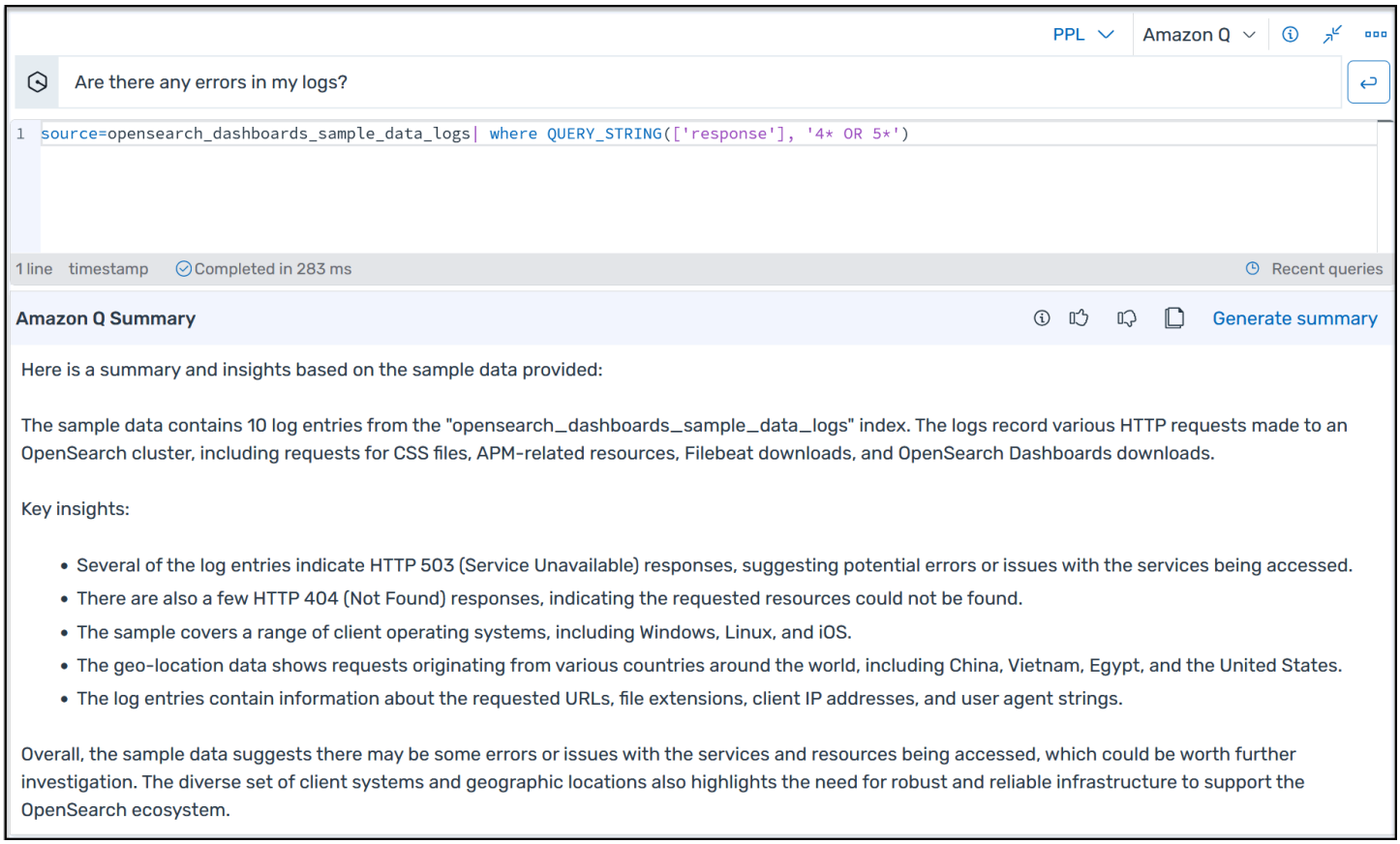
6. Si elles figurent dans le résumé des alertes, choisissez Afficher dans Discover pour afficher les données d'alerte sur la page Découvrir OpenSearch des tableaux de bord.

Consultez les résumés des résultats de requêtes générés par Amazon Q sur la page Discover

OpenSearch Le service vous permet d'interroger vos données à l'aide d'instructions en langage naturel à l'aide du langage de requête PPL (Piped Processing Language) sur la page Discover. Par exemple, vous pouvez ajouter ce paramètre comme le suivant :

- Y a-t-il des erreurs dans mes journaux d'erreurs ?
- Quelle est la taille moyenne des demandes par semaine ?
- Combien de demandes ont été regroupées par code de réponse la semaine dernière ?

En réponse, Amazon Q génère des résumés en langage naturel des résultats de vos requêtes sur la base des dix premiers enregistrements, comme suit :



La combinaison de la génération de requêtes en langage naturel et de résumés de requêtes peut ajouter un niveau d'interrogation supplémentaire lors de la résolution d'une alerte ou constituer un moyen convivial de comprendre vos données sans avoir à rédiger de requêtes complexes.

Pour consulter les résumés des résultats de requêtes générés par Amazon Q sur la page Discover

1. Vérifiez que vous avez configuré Amazon Q for OpenSearch Service.
2. Dans le menu principal OpenSearch des tableaux de bord, sélectionnez Découvrir.
3. Dans la liste déroulante, sélectionnez la liste déroulante.
4. Dans la zone de texte Amazon Q, entrez une invite, puis cliquez sur le bouton à côté de la zone de texte pour exécuter la requête. L'ensemble du processus dure jusqu'à 10 minutes. Après votre requête initiale, vous devez choisir Générer un résumé pour les résumés suivants.

> **Note**
>
> Vous pouvez désactiver la génération de résumés depuis la liste déroulante Amazon Q dans Discover.

Décrire les détecteurs d'anomalies

La fonctionnalité de détection des anomalies d'Amazon OpenSearch OpenSearch Service détecte automatiquement et en temps quasi réel les anomalies OpenSearch présentes dans vos données OpenSearch à l'aide de l'algorithme Random Cut Forest (RCF). RCF est un algorithme de machine learning non supervisé qui modélise un croquis de votre flux de données entrant. Cet algorithme calcule une valeur `anomaly grade` et `confidence score` pour chaque point de données entrant. La fonctionnalité de détection des anomalies utilise ces valeurs pour différencier une anomalie des variations normales de vos données.

Pour simplifier le processus de création de détecteurs d'anomalies, Amazon Q peut générer des suggestions de détecteurs en fonction de la source de données que vous avez sélectionnée sur la page Découvrir. Amazon Q prend en charge les détecteurs d'anomalies suggérés pour toutes les langues.

Pour consulter les détecteurs d'anomalies recommandés par Amazon Q

1. Vérifiez que vous avez configuré Amazon Q for OpenSearch Service.

2. Dans le menu principal OpenSearch des tableaux de bord, choisissez la page Découvrir, puis choisissez une source de données.
3. Dans le menu Amazon Q, choisissez Suggérer un détecteur d'anomalie, comme illustré dans la capture d'écran suivante.

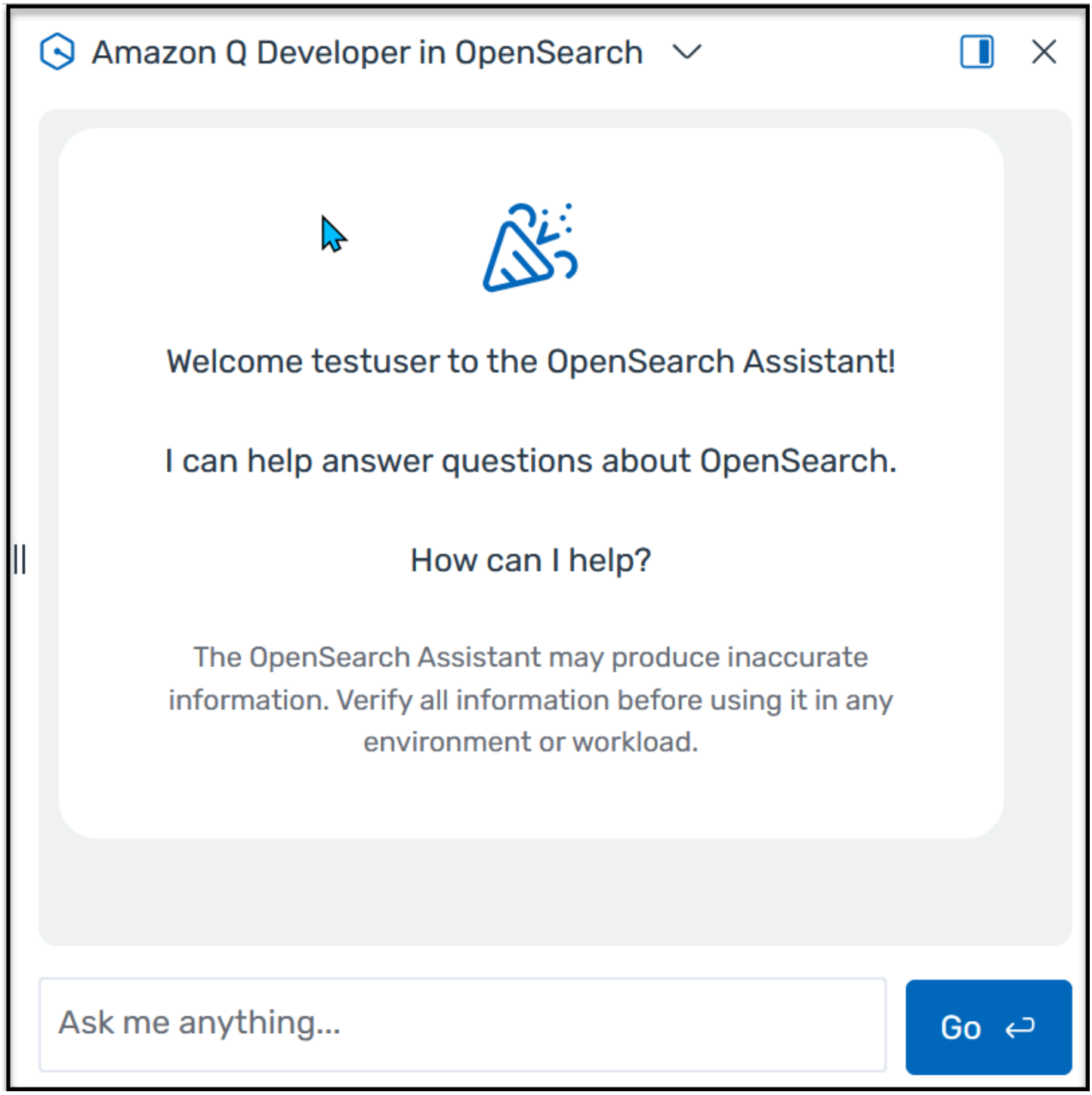


Amazon Q peut mettre quelques secondes à générer les fonctionnalités du détecteur.

4. Choisissez Créer un détecteur.

Accédez au chat Amazon Q pour les OpenSearch questions connexes

Si vous avez des questions concernant Amazon OpenSearch Service, notamment des questions conceptuelles ou procédurales liées aux fonctionnalités du OpenSearch Service, vous pouvez demander des informations à Amazon Q Developer. Le chatbot prend en charge un chat conversationnel qui préserve le contexte de votre discussion. Il enregistre également un historique de vos conversations pour référence ultérieure. Pour accéder au chatbot, cliquez sur l'icône Amazon Q en haut à droite de n'importe quelle page de OpenSearch service. Tapez une question dans la zone de texte et cliquez sur OK. L'ensemble du processus dure jusqu'à 10 minutes.



Voici un exemple de réponse.

Can you suggest some node management best practices?

Based on your context of being in the OpenSearch Dashboard in the us-west-2 region, here are some node management best practices for Amazon OpenSearch Service: [1]

1. Implement Dedicated Manager Nodes:
    - Use dedicated manager nodes to improve cluster stability. [2]
    - Configure three dedicated manager nodes for optimal domain stability across three Availability Zones.
    - This offloads cluster management tasks from data nodes, enhancing overall performance.
2. Multi-AZ Deployment:
    - Deploy your domain across multiple Availability Zones for increased availability and resilience.
    - Consider using the Multi-AZ with Standby option for 99.99% availability and consistent performance.
3. Proper Sizing:

> **Note**
>
> Le chat Amazon Q ne peut pas accéder à vos données. Pour cette raison, vous ne pouvez pas engager le chatbot dans une conversation concernant vos données.

Apprentissage automatique pour Amazon OpenSearch Service

ML Commons est un OpenSearch plugin qui fournit un ensemble d'algorithmes d'apprentissage automatique (ML) courants par le biais d'appels d'API de transport et d'API REST. Ces appels choisissent les nœuds et les ressources appropriés pour chaque demande de machine learning et surveillent les tâches de machine learning pour garantir la disponibilité. Cela vous permet de tirer parti des algorithmes de ML open source existants et de réduire les efforts nécessaires au développement de nouvelles fonctionnalités de ML. Pour en savoir plus sur le plugin, consultez la section Machine learning dans la OpenSearch documentation. Ce chapitre explique comment utiliser le plugin avec Amazon OpenSearch Service.

Rubriques

- Connecteurs Amazon OpenSearch Service ML pour Services AWS
- Connecteurs Amazon OpenSearch Service ML pour plateformes tierces
- Utilisation AWS CloudFormation pour configurer l'inférence à distance pour la recherche sémantique
- Paramètres ML Commons non pris en charge
- OpenSearch Modèles de framework de flux de services

Connecteurs Amazon OpenSearch Service ML pour Services AWS

Lorsque vous utilisez des connecteurs d'apprentissage automatique (ML) Amazon OpenSearch Service avec un autre Service AWS, vous devez configurer un rôle IAM pour connecter le OpenSearch Service à ce service en toute sécurité. Services AWS que vous pouvez configurer un connecteur pour inclure Amazon SageMaker AI et Amazon Bedrock. Dans ce didacticiel, nous expliquons comment créer un connecteur entre OpenSearch Service et SageMaker Runtime. Pour plus d'informations sur les connecteurs, voir Connecteurs pris en charge.

Rubriques

- Prérequis
- Création d'un connecteur OpenSearch de service

Prérequis

Pour créer un connecteur, vous devez disposer d'un point de terminaison Amazon SageMaker AI Domain et d'un rôle IAM qui accorde l'accès au OpenSearch service.

Configurer un domaine Amazon SageMaker AI

Consultez la section Déployer un modèle dans Amazon SageMaker AI dans le manuel Amazon SageMaker AI Developer Guide pour déployer votre modèle d'apprentissage automatique. Notez l'URL du point de terminaison de votre modèle, dont vous avez besoin pour créer un connecteur AI.

Créer un rôle IAM

Configurez un rôle IAM pour déléguer les autorisations SageMaker d'exécution au OpenSearch service. Pour créer un nouveau rôle, consultez la section Création d'un rôle IAM (console) dans le guide de l'utilisateur IAM. Vous pouvez éventuellement utiliser un rôle existant à condition qu'il dispose du même ensemble de privilèges. Si vous créez un nouveau rôle au lieu d'utiliser un rôle AWS géré, remplacez-le `opensearch-sagemaker-role` dans ce didacticiel par le nom de votre propre rôle.

1. Associez la politique IAM gérée suivante à votre nouveau rôle pour permettre au OpenSearch Service d'accéder à votre point de terminaison SageMaker AI. Pour associer une politique à un rôle, consultez la section Ajout d'autorisations d'identité IAM.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "sagemaker:InvokeEndpointAsync",
                "sagemaker:InvokeEndpoint"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

2. Suivez les instructions de la section Modification d'une politique de confiance de rôle pour modifier la relation de confiance du rôle. Vous devez spécifier OpenSearch Service dans la `Principal` déclaration :

   JSON

   ```
   {
       "Version": "2012-10-17",
       "Statement": [
           {
               "Action": [
                   "sts:AssumeRole"
               ],
               "Effect": "Allow",
               "Principal": {
                   "Service": [
                       "opensearchservice.amazonaws.com"
                   ]
               }
           }
       ]
   }
   ```

   Nous vous recommandons d'utiliser les clés de `aws:SourceArn` condition `aws:SourceAccount` et pour limiter l'accès à un domaine spécifique. `SourceAccount`Il s'agit de l' Compte AWS ID qui appartient au propriétaire du domaine et de l'ARN du domaine. `SourceArn` Par exemple, vous pouvez ajouter le bloc de condition suivant à la politique de confiance :

   ```
   "Condition": {
       "StringEquals": {
           "aws:SourceAccount": "account-id"
       },
       "ArnLike": {
           "aws:SourceArn": "arn:aws:es:region:account-id:domain/domain-name"
       }
   }
   ```

Configurer des autorisations

Pour créer le connecteur, vous devez être autorisé à transmettre le rôle IAM au OpenSearch Service. Vous avez également besoin de l'accès à l'action `es:ESHttpPost`. Pour accorder ces deux autorisations, attachez la politique suivante au rôle IAM dont les informations d'identification sont utilisées pour signer la demande :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::account-id:role/opensearch-sagemaker-role"
    },
    {
      "Effect": "Allow",
      "Action": "es:ESHttpPost",
      "Resource": "arn:aws:es:region:account-id:domain/domain-name/*"
    }
  ]
}
```

Si votre utilisateur ou votre rôle n'est pas `iam:PassRole` autorisé à transmettre votre rôle, il se peut que vous rencontriez une erreur d'autorisation lorsque vous tenterez d'enregistrer un référentiel à l'étape suivante.

Cartographier le rôle du machine learning dans OpenSearch les tableaux de bord (si vous utilisez un contrôle d'accès précis)

Le contrôle d'accès précis introduit une étape supplémentaire lors de la configuration d'un connecteur. Même si vous utilisez l'authentification de base HTTP à toutes les autres fins, vous devez mapper le rôle `ml_full_access` à votre rôle IAM qui a les autorisations `iam:PassRole` pour transmettre `opensearch-sagemaker-role`.

1. Accédez au plugin OpenSearch Dashboards correspondant à votre domaine OpenSearch de service. Vous pouvez trouver le point de terminaison Dashboards sur le tableau de bord de votre domaine sur la console OpenSearch de service.
2. Dans le menu principal, choisissez Sécurité, Rôles, puis sélectionnez le rôle ml_full_access.
3. Choisissez Mapped users (Utilisateurs mappés), Manage mapping (Gérer le mappage).
4. Sous Rôles principaux, ajoutez l'ARN du rôle autorisé à transmettreo`opensearch-sagemaker-role`.

   ```
   arn:aws:iam::account-id:role/role-name
   ```

5. Sélectionnez Mapper et vérifiez que l'utilisateur ou le rôle s'affiche sous Utilisateurs mappés.

Création d'un connecteur OpenSearch de service

Pour créer un connecteur, envoyez une POST demande au point de terminaison du domaine de OpenSearch service. Vous pouvez utiliser curl, le client Python d'exemple, Postman ou une autre méthode pour envoyer une demande signée. Notez que vous ne pouvez pas utiliser de POST requête dans la console Kibana. La demande se présente au format suivant :

```
POST domain-endpoint/_plugins/_ml/connectors/_create
{
   "name": "sagemaker: embedding",
   "description": "Test connector for Sagemaker embedding model",
   "version": 1,
   "protocol": "aws_sigv4",
   "credential": {
      "roleArn": "arn:aws:iam::account-id:role/opensearch-sagemaker-role"
   },
   "parameters": {
      "region": "region",
      "service_name": "sagemaker"
   },
   "actions": [
      {
         "action_type": "predict",
         "method": "POST",
         "headers": {
            "content-type": "application/json"
         },
```

```
        "url": "https://runtime.sagemaker.region.amazonaws.com/endpoints/endpoint-id/
invocations",
        "request_body": "{ \"inputs\": { \"question\": \"${parameters.question}\",
 \"context\": \"${parameters.context}\" } }"
      }
   ]
}
```

Si votre domaine réside dans un cloud privé virtuel (VPC), votre ordinateur doit être connecté au VPC pour que la demande puisse créer correctement le connecteur AI. L'accès à un VPC varie en fonction de la configuration réseau, mais implique généralement de se connecter à un VPN ou à un réseau d'entreprise. Pour vérifier que vous pouvez accéder à votre domaine de OpenSearch service, accédez `https://your-vpc-domain.region.es.amazonaws.com` à un navigateur Web et vérifiez que vous recevez la réponse JSON par défaut.

Exemple de client Python

Le client Python est plus simple à automatiser qu'une requête HTTP et offre une meilleure réutilisabilité. Pour créer le connecteur AI avec le client Python, enregistrez l'exemple de code suivant dans un fichier Python. Le client a besoin des `requests-aws4auth`packages AWS SDK pour Python (Boto3)`requests`,, et.

```
import boto3
import requests
from requests_aws4auth import AWS4Auth

host = 'domain-endpoint/'
region = 'region'
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

# Register repository
path = '_plugins/_ml/connectors/_create'
url = host + path

payload = {
   "name": "sagemaker: embedding",
   "description": "Test connector for Sagemaker embedding model",
   "version": 1,
   "protocol": "aws_sigv4",
```

```
   "credential": {
      "roleArn": "arn:aws:iam::account-id:role/opensearch-sagemaker-role"
   },
   "parameters": {
      "region": "region",
      "service_name": "sagemaker"
   },
   "actions": [
      {
         "action_type": "predict",
         "method": "POST",
         "headers": {
            "content-type": "application/json"
         },
         "url": "https://runtime.sagemaker.region.amazonaws.com/endpoints/endpoint-id/
invocations",
         "request_body": "{ \"inputs\": { \"question\": \"${parameters.question}\",
 \"context\": \"${parameters.context}\" } }"
      }
   ]
}
headers = {"Content-Type": "application/json"}

r = requests.post(url, auth=awsauth, json=payload, headers=headers)
print(r.status_code)
print(r.text)
```

Connecteurs Amazon OpenSearch Service ML pour plateformes tierces

Dans ce didacticiel, nous expliquons comment créer un connecteur de OpenSearch Service à Cohere. Pour plus d'informations sur les connecteurs, voir Connecteurs pris en charge.

Lorsque vous utilisez un connecteur d'apprentissage automatique (ML) Amazon OpenSearch Service avec un modèle de télécommande externe, vous devez y stocker vos informations d'autorisation spécifiques AWS Secrets Manager. Il peut s'agir d'une clé API ou d'une combinaison de nom d'utilisateur et de mot de passe. Cela signifie que vous devez également créer un rôle IAM qui autorise l'accès au OpenSearch service à lire depuis Secrets Manager.

Rubriques

- Prérequis

- Création d'un connecteur OpenSearch de service

Prérequis

Pour créer un connecteur pour Cohere ou tout autre fournisseur externe avec OpenSearch Service, vous devez disposer d'un rôle IAM qui accorde l'accès au OpenSearch Service AWS Secrets Manager, dans lequel vous stockez vos informations d'identification. Vous devez également enregistrer vos informations d'identification dans Secrets Manager.

Créer un rôle IAM

Configurez un rôle IAM pour déléguer les autorisations de Secrets Manager au OpenSearch Service. Vous pouvez également utiliser le `SecretManagerReadWrite` rôle existant. Pour créer un nouveau rôle, consultez la section Création d'un rôle IAM (console) dans le guide de l'utilisateur IAM. Si vous créez un nouveau rôle au lieu d'utiliser un rôle AWS géré, remplacez-le `opensearch-secretmanager-role` dans ce didacticiel par le nom de votre propre rôle.

1. Associez la politique IAM gérée suivante à votre nouveau rôle pour permettre au OpenSearch Service d'accéder aux valeurs de vos Secrets Manager. Pour associer une politique à un rôle, consultez la section Ajout d'autorisations d'identité IAM.

   JSON

   ```
   {
       "Version": "2012-10-17",
       "Statement": [
           {
               "Action": [
                   "secretsmanager:GetSecretValue"
               ],
               "Effect": "Allow",
               "Resource": "*"
           }
       ]
   }
   ```

2. Suivez les instructions de la section Modification d'une politique de confiance de rôle pour modifier la relation de confiance du rôle. Vous devez spécifier OpenSearch Service dans la `Principal` déclaration :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "sts:AssumeRole"
            ],
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "opensearchservice.amazonaws.com"
                ]
            }
        }
    ]
}
```

Nous vous recommandons d'utiliser les touches de `aws:SourceArn` condition `aws:SourceAccount` et pour limiter l'accès à un domaine spécifique. `SourceAccount`Il s'agit de l' Compte AWS ID qui appartient au propriétaire du domaine et de l'ARN du domaine. `SourceArn` Par exemple, vous pouvez ajouter le bloc de condition suivant à la politique de confiance :

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account-id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:aws:es:region:account-id:domain/domain-name"
    }
}
```

Configurer des autorisations

Pour créer le connecteur, vous devez être autorisé à transmettre le rôle IAM au OpenSearch Service. Vous avez également besoin de l'accès à l'action `es:ESHttpPost`. Pour accorder ces deux autorisations, attachez la politique suivante au rôle IAM dont les informations d'identification sont utilisées pour signer la demande :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::account-id:role/opensearch-secretmanager-role"
    },
    {
      "Effect": "Allow",
      "Action": "es:ESHttpPost",
      "Resource": "arn:aws:es:region:account-id:domain/domain-name/*"
    }
  ]
}
```

Si votre utilisateur ou votre rôle n'est pas `iam:PassRole` autorisé à transmettre votre rôle, il se peut que vous rencontriez une erreur d'autorisation lorsque vous tenterez d'enregistrer un référentiel à l'étape suivante.

Configurez AWS Secrets Manager

Pour enregistrer vos informations d'autorisation dans Secrets Manager, consultez la section Créer un AWS Secrets Manager secret dans le Guide de AWS Secrets Manager l'utilisateur.

Une fois que Secrets Manager a accepté votre paire clé-valeur en tant que secret, vous recevez un ARN au format :. `arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret-a1b2c3` Conservez un enregistrement de cet ARN, tel que vous l'utilisez, ainsi que de votre clé lorsque vous créerez un connecteur à l'étape suivante.

Cartographier le rôle du machine learning dans OpenSearch les tableaux de bord (si vous utilisez un contrôle d'accès précis)

Le contrôle d'accès précis introduit une étape supplémentaire lors de la configuration d'un connecteur. Même si vous utilisez l'authentification de base HTTP à toutes les autres fins, vous devez mapper le rôle `ml_full_access` à votre rôle IAM qui a les autorisations `iam:PassRole` pour transmettre `opensearch-sagemaker-role`.

1. Accédez au plugin OpenSearch Dashboards correspondant à votre domaine OpenSearch de service. Vous pouvez trouver le point de terminaison Dashboards sur le tableau de bord de votre domaine sur la console OpenSearch de service.
2. Dans le menu principal, choisissez Sécurité, Rôles, puis sélectionnez le rôle ml_full_access.
3. Choisissez Mapped users (Utilisateurs mappés), Manage mapping (Gérer le mappage).
4. Sous Rôles principaux, ajoutez l'ARN du rôle autorisé à transmettre`opensearch-sagemaker-role`.

   ```
   arn:aws:iam::account-id:role/role-name
   ```

5. Sélectionnez Mapper et vérifiez que l'utilisateur ou le rôle s'affiche sous Utilisateurs mappés.

Création d'un connecteur OpenSearch de service

Pour créer un connecteur, envoyez une POST demande au point de terminaison du domaine de OpenSearch service. Vous pouvez utiliser curl, le client Python d'exemple, Postman ou une autre méthode pour envoyer une demande signée. Notez que vous ne pouvez pas utiliser de POST requête dans la console Kibana. La demande se présente au format suivant :

```
POST domain-endpoint/_plugins/_ml/connectors/_create
{
    "name": "Cohere Connector: embedding",
    "description": "The connector to cohere embedding model",
    "version": 1,
    "protocol": "http",
    "credential": {
        "secretArn": "arn:aws:secretsmanager:region:account-id:secret:cohere-key-id",
        "roleArn": "arn:aws:iam::account-id:role/opensearch-secretmanager-role"
    },
    "actions": [
        {
            "action_type": "predict",
            "method": "POST",
            "url": "https://api.cohere.ai/v1/embed",
            "headers": {
                "Authorization": "Bearer ${credential.secretArn.cohere-key-used-in-
secrets-manager}"
            },
            "request_body": "{ \"texts\": ${parameters.texts}, \"truncate\": \"END\" }"
        }
```

```
    ]
}
```

Le corps de la demande pour cette demande est différent de celui d'une demande de connecteur open source de deux manières. Dans le `credential` champ, vous transmettez l'ARN du rôle IAM qui permet au OpenSearch Service de lire depuis Secrets Manager, ainsi que l'ARN du secret quel. Dans le `headers` champ, vous faites référence au secret à l'aide de la clé secrète et du fait qu'il provient d'un ARN.

Si votre domaine réside dans un cloud privé virtuel (VPC), votre ordinateur doit être connecté au VPC pour que la demande puisse créer correctement le connecteur AI. L'accès à un VPC varie en fonction de la configuration réseau, mais implique généralement de se connecter à un VPN ou à un réseau d'entreprise. Pour vérifier que vous pouvez accéder à votre domaine de OpenSearch service, accédez `https://`*`your-vpc-domain`*`.`*`region`*`.es.amazonaws.com` à un navigateur Web et vérifiez que vous recevez la réponse JSON par défaut.

Exemple de client Python

Le client Python est plus simple à automatiser qu'une requête HTTP et offre une meilleure réutilisabilité. Pour créer le connecteur AI avec le client Python, enregistrez l'exemple de code suivant dans un fichier Python. Le client a besoin des `requests-aws4auth`packages AWS SDK pour Python (Boto3)`requests`,, et.

```
import boto3
import requests
from requests_aws4auth import AWS4Auth

host = 'domain-endpoint/'
region = 'region'
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

path = '_plugins/_ml/connectors/_create'
url = host + path

payload = {
    "name": "Cohere Connector: embedding",
    "description": "The connector to cohere embedding model",
    "version": 1,
```

```
    "protocol": "http",
    "credential": {
        "secretArn": "arn:aws:secretsmanager:region:account-id:secret:cohere-key-id",
        "roleArn": "arn:aws:iam::account-id:role/opensearch-secretmanager-role"
    },
    "actions": [
        {
            "action_type": "predict",
            "method": "POST",
            "url": "https://api.cohere.ai/v1/embed",
            "headers": {
                "Authorization": "Bearer ${credential.secretArn.cohere-key-used-in-
secrets-manager}"
            },
            "request_body": "{ \"texts\": ${parameters.texts}, \"truncate\": \"END\" }"
        }
    ]
}

headers = {"Content-Type": "application/json"}

r = requests.post(url, auth=awsauth, json=payload, headers=headers)
print(r.status_code)
print(r.text)
```

Utilisation AWS CloudFormation pour configurer l'inférence à distance pour la recherche sémantique

À partir de OpenSearch la version 2.9, vous pouvez utiliser l'inférence à distance avec la recherche sémantique pour héberger vos propres modèles d'apprentissage automatique (ML). L'inférence à distance utilise le plugin ML Commons pour vous permettre d'héberger vos inférences de modèles à distance sur des services ML, tels qu' Amazon SageMaker AI Amazon BedRock, et de les connecter à Amazon OpenSearch Service à l'aide de connecteurs ML.

Pour faciliter la configuration de l'inférence à distance, Amazon OpenSearch Service fournit un AWS CloudFormationmodèle dans la console. CloudFormation est un outil Service AWS qui vous permet de modéliser, de provisionner et de gérer AWS des ressources tierces en traitant l'infrastructure comme du code.

Le OpenSearch CloudFormation modèle automatise le processus de mise en service du modèle pour vous, afin que vous puissiez facilement créer un modèle dans votre domaine de OpenSearch

service, puis utiliser l'ID du modèle pour ingérer des données et exécuter des requêtes de recherche neuronale.

Lorsque vous utilisez des encodeurs neuronaux épars avec les versions 2.12 et ultérieures du OpenSearch Service, nous vous recommandons d'utiliser le modèle tokenizer localement plutôt que de le déployer à distance. Pour plus d'informations, consultez la section Modèles de codage épars dans la OpenSearch documentation.

Rubriques

- Prérequis
- Amazon SageMaker AI modèles
- Modèles Amazon Bedrock

Prérequis

Pour utiliser un CloudFormation modèle avec OpenSearch Service, remplissez les conditions préalables suivantes.

Configuration d'un domaine OpenSearch de service

Avant de pouvoir utiliser un CloudFormation modèle, vous devez configurer un domaine Amazon OpenSearch Service avec la version 2.9 ou ultérieure et activer le contrôle d'accès détaillé. Créez un rôle OpenSearch de backend de service pour autoriser le plugin ML Commons à créer votre connecteur pour vous.

Le CloudFormation modèle crée pour vous un rôle Lambda IAM avec le nom par défaut`LambdaInvokeOpenSearchMLCommonsRole`, que vous pouvez remplacer si vous souhaitez en choisir un autre. Une fois que le modèle a créé ce rôle IAM, vous devez autoriser la fonction Lambda à appeler OpenSearch votre domaine de service. Pour ce faire, associez le rôle nommé `ml_full_access` à votre rôle OpenSearch principal de service en procédant comme suit :

1. Accédez au plugin OpenSearch Dashboards correspondant à votre domaine OpenSearch de service. Vous pouvez trouver le point de terminaison Dashboards sur le tableau de bord de votre domaine sur la console OpenSearch de service.
2. Dans le menu principal, choisissez Sécurité, Rôles, puis sélectionnez le rôle ml_full_access.
3. Choisissez Mapped users (Utilisateurs mappés), Manage mapping (Gérer le mappage).

4. Sous Rôles principaux, ajoutez l'ARN du rôle Lambda qui a besoin d'une autorisation pour appeler votre domaine.

```
arn:aws:iam::account-id:role/role-name
```

5. Sélectionnez Mapper et vérifiez que l'utilisateur ou le rôle s'affiche sous Utilisateurs mappés.

Après avoir mappé le rôle, accédez à la configuration de sécurité de votre domaine et ajoutez le rôle Lambda IAM à OpenSearch votre politique d'accès au service.

Activez les autorisations sur votre Compte AWS

Vous Compte AWS devez être autorisé à accéder CloudFormation à Lambda, ainsi qu'à celui que Service AWS vous choisissez pour votre modèle (SageMaker Runtime ou Amazon). BedRock

Si vous utilisez Amazon Bedrock, vous devez également enregistrer votre modèle. Consultez la section Accès aux modèles dans le guide de l'utilisateur d'Amazon Bedrock pour enregistrer votre modèle.

Si vous utilisez votre propre compartiment Amazon S3 pour fournir des artefacts de modèle, vous devez ajouter le rôle CloudFormation IAM à votre politique d'accès S3. Pour plus d'informations, consultez la rubrique Ajout et suppression d'autorisations basées sur l'identité IAM du Guide de l'utilisateur IAM.

Amazon SageMaker AI modèles

Les CloudFormation modèles Amazon SageMaker AI définissent plusieurs AWS ressources afin de configurer le plugin neuronal et la recherche sémantique pour vous.

Tout d'abord, utilisez le modèle Intégration aux modèles d'intégration de texte via Amazon pour déployer un SageMaker modèle d'intégration de texte dans SageMaker Runtime en tant que serveur. Si vous ne fournissez pas de point de terminaison modèle, CloudFormation crée un rôle IAM qui permet à SageMaker Runtime de télécharger les artefacts du modèle depuis Amazon S3 et de les déployer sur le serveur. Si vous fournissez un point de terminaison, CloudFormation créez un rôle IAM qui permet à la fonction Lambda d'accéder OpenSearch au domaine de service ou, si le rôle existe déjà, le met à jour et le réutilise. Le point de terminaison sert le modèle distant utilisé pour le connecteur ML avec le plugin ML Commons.

Ensuite, utilisez le modèle Integration with Sparse Encoders through Amazon Sagemaker pour créer une fonction Lambda qui permet à votre domaine de configurer des connecteurs d'inférence à

distance. Une fois le connecteur créé dans OpenSearch Service, l'inférence à distance peut exécuter une recherche sémantique à l'aide du modèle distant dans SageMaker Runtime. Le modèle vous renvoie l'ID du modèle de votre domaine afin que vous puissiez commencer la recherche.

Pour utiliser les CloudFormation modèles Amazon SageMaker AI

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, sélectionnez Intégrations.
3. Sous chacun des modèles Amazon SageMaker AI, choisissez Configurer le domaine, Configurer le domaine public.
4. Suivez les instructions de la CloudFormation console pour approvisionner votre stack et configurer un modèle.

> **Note**
>
> OpenSearch Le service fournit également un modèle distinct pour configurer le domaine VPC. Si vous utilisez ce modèle, vous devez fournir l'ID VPC de la fonction Lambda.

Modèles Amazon Bedrock

À l'instar des CloudFormation modèles Amazon SageMaker AI, le CloudFormation modèle Amazon Bedrock fournit les AWS ressources nécessaires pour créer des connecteurs entre OpenSearch Service et Amazon Bedrock.

Tout d'abord, le modèle crée un rôle IAM qui permet à la future fonction Lambda d'accéder à OpenSearch votre domaine de service. Le modèle crée ensuite la fonction Lambda, qui permet au domaine de créer un connecteur à l'aide du plugin ML Commons. Une fois que OpenSearch Service a créé le connecteur, la configuration de l'inférence à distance est terminée et vous pouvez exécuter des recherches sémantiques à l'aide des opérations de l'API Amazon Bedrock.

Notez qu'Amazon Bedrock héberge ses propres modèles de ML, il n'est pas nécessaire de déployer un modèle dans SageMaker Runtime. Au lieu de cela, le modèle utilise un point de terminaison prédéterminé pour Amazon Bedrock et ignore les étapes de fourniture du point de terminaison.

Pour utiliser le modèle Amazon Bedrock CloudFormation

1. Ouvrez la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Dans le volet de navigation de gauche, sélectionnez Intégrations.
3. Sous Intégrer au modèle Amazon Titan Text Embeddings via Amazon Bedrock, choisissez Configurer le domaine, Configurer le domaine public.
4. Suivez les instructions pour configurer votre modèle.

> **Note**
>
> OpenSearch Le service fournit également un modèle distinct pour configurer le domaine VPC. Si vous utilisez ce modèle, vous devez fournir l'ID VPC de la fonction Lambda.

En outre, OpenSearch Service fournit les modèles Amazon Bedrock suivants pour se connecter au modèle Cohere et au modèle d'intégration multimodale Amazon Titan :

- `Integration with Cohere Embed through Amazon Bedrock`
- `Integrate with Amazon Bedrock Titan Multi-modal`

Paramètres ML Commons non pris en charge

Amazon OpenSearch Service ne prend pas en charge l'utilisation des paramètres ML Commons suivants :

- `plugins.ml_commons.allow_registering_model_via_url`
- `plugins.ml_commons.allow_registering_model_via_local_file`

> **Important**
>
> Sur les clusters de production, ne désactivez pas le paramètre du cluster `plugins.ml_commons.only_run_on_ml_node` (ne le définissez pas sur`false`). L'option permettant de désactiver cette protection est destinée à faciliter le développement, mais les

clusters de production doivent utiliser les connecteurs. Pour de plus amples informations, veuillez consulter the section called “Connecteurs pour Services AWS”.

Pour plus d'informations sur les paramètres de ML Commons, consultez la section Paramètres du cluster ML Commons.

OpenSearch Modèles de framework de flux de services

Les modèles de framework Amazon OpenSearch Service Flow vous permettent d'automatiser les tâches complexes de configuration et de prétraitement des OpenSearch services en fournissant des modèles pour les cas d'utilisation courants. Par exemple, vous pouvez utiliser des modèles de structure de flux pour automatiser les tâches de configuration du machine learning. Les modèles de framework Amazon OpenSearch Service Flow fournissent une description compacte du processus de configuration dans un document JSON ou YAML. Ces modèles décrivent les configurations de flux de travail automatisés pour le chat conversationnel ou la génération de requêtes, les connecteurs AI, les outils, les agents et les autres composants qui préparent le OpenSearch service à une utilisation en backend pour les modèles génératifs.

Les modèles de framework Amazon OpenSearch Service Flow peuvent être personnalisés pour répondre à vos besoins spécifiques. Pour voir un exemple de modèle de structure de flux personnalisé, voir flow-framework. Pour les modèles fournis par le OpenSearch service, voir modèles de flux de travail. Pour une documentation complète, y compris les étapes détaillées, une référence d'API et une référence de tous les paramètres disponibles, consultez la section Automatisation de la configuration dans la OpenSearch documentation open source.

> **Note**
>
> Flow-framework ne prend pas en charge le filtrage des rôles principaux pour OpenSearch le Service 2.17.

Création de connecteurs ML dans OpenSearch Service

Les modèles de framework Amazon OpenSearch Service Flow vous permettent de configurer et d'installer des connecteurs ML en utilisant l'API de création de connecteurs proposée dans ml-commons. Vous pouvez utiliser les connecteurs ML pour connecter le OpenSearch service à d'autres AWS services ou à des plateformes tierces. Pour plus d'informations à ce sujet, consultez

Création de connecteurs pour des plateformes ML tierces. L'API Amazon OpenSearch Service Flow Framework vous permet d'automatiser les tâches de configuration et de prétraitement du OpenSearch service et peut être utilisée pour créer des connecteurs ML.

Avant de créer un connecteur dans OpenSearch Service, vous devez effectuer les opérations suivantes :

- Créez un domaine Amazon SageMaker AI.
- Créez un rôle IAM.
- Configurez l'autorisation du rôle de passe.
- Mappez les rôles flow-framework et ml-commons dans les tableaux de bord. OpenSearch

Pour plus d'informations sur la configuration des connecteurs ML pour les AWS services, consultez la section Connecteurs ML Amazon OpenSearch Service pour les AWS services. Pour en savoir plus sur l'utilisation des connecteurs OpenSearch Service ML avec des plateformes tierces, consultez la section Connecteurs Amazon OpenSearch Service ML pour plateformes tierces.

Création d'un connecteur via un service Flow-Framework

Pour créer un modèle de framework de flux avec connecteur, vous devez envoyer une `POST` demande au point de terminaison de votre domaine OpenSearch de service. Vous pouvez utiliser cURL, un exemple de client Python, Postman ou une autre méthode pour envoyer une demande signée. La `POST` demande prend le format suivant :

```
POST /_plugins/_flow_framework/workflow
{
  "name": "Deploy Claude Model",
  "description": "Deploy a model using a connector to Claude",
  "use_case": "PROVISION",
  "version": {
    "template": "1.0.0",
    "compatibility": [
      "2.12.0",
      "3.0.0"
    ]
  },
  "workflows": {
    "provision": {
      "nodes": [
        {
```

```
          "id": "create_claude_connector",
          "type": "create_connector",
          "user_inputs": {
            "name": "Claude Instant Runtime Connector",
            "version": "1",
            "protocol": "aws_sigv4",
            "description": "The connector to BedRock service for Claude model",
            "actions": [
              {
                "headers": {
                  "x-amz-content-sha256": "required",
                  "content-type": "application/json"
                },
                "method": "POST",
                "request_body": "{ \"prompt\":\"${parameters.prompt}\",
 \"max_tokens_to_sample\":${parameters.max_tokens_to_sample},
 \"temperature\":${parameters.temperature},  \"anthropic_version\":
\"${parameters.anthropic_version}\" }",
                "action_type": "predict",
                "url": "https://bedrock-runtime.us-west-2.amazonaws.com/model/
anthropic.claude-instant-v1/invoke"
              }
            ],
            "credential": {
                "roleArn": "arn:aws:iam::account-id:role/opensearch-secretmanager-
role"
             },
            "parameters": {
              "endpoint": "bedrock-runtime.us-west-2.amazonaws.com",
              "content_type": "application/json",
              "auth": "Sig_V4",
              "max_tokens_to_sample": "8000",
              "service_name": "bedrock",
              "temperature": "0.0001",
              "response_filter": "$.completion",
              "region": "us-west-2",
              "anthropic_version": "bedrock-2023-05-31"
            }
          }
        }
      ]
    }
  }
```

```
}
```

Si votre domaine réside dans un cloud privé virtuel (Amazon VPC), vous devez être connecté à Amazon VPC pour que la demande puisse créer correctement le connecteur AI. L'accès à un Amazon VPC varie en fonction de la configuration du réseau, mais implique généralement de se connecter à un VPN ou à un réseau d'entreprise. Pour vérifier que vous pouvez accéder à votre domaine de OpenSearch service, accédez `https://`*`your-vpc-domain`*`.`*`region`*`.es.amazonaws.com` à un navigateur Web et vérifiez que vous recevez la réponse JSON par défaut. (*`placeholder text`*Remplacez-les par vos propres valeurs.

Exemple de client Python

Le client Python est plus simple à automatiser qu'une HTTP requête et offre une meilleure réutilisabilité. Pour créer le connecteur AI avec le client Python, enregistrez l'exemple de code suivant dans un fichier Python. Le client a besoin des packages AWS SDK for Python (Boto3), requests:HTTP for Humanset requests-aws4auth 1.2.3.

```
import boto3
import requests
from requests_aws4auth import AWS4Auth

host = 'domain-endpoint/'
region = 'region'
service = 'es'
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

path = '_plugins/_flow_framework/workflow'
url = host + path

payload = {
  "name": "Deploy Claude Model",
  "description": "Deploy a model using a connector to Claude",
  "use_case": "PROVISION",
  "version": {
    "template": "1.0.0",
    "compatibility": [
      "2.12.0",
      "3.0.0"
    ]
  },
```

```
  "workflows": {
    "provision": {
      "nodes": [
        {
          "id": "create_claude_connector",
          "type": "create_connector",
          "user_inputs": {
            "name": "Claude Instant Runtime Connector",
            "version": "1",
            "protocol": "aws_sigv4",
            "description": "The connector to BedRock service for Claude model",
            "actions": [
              {
                "headers": {
                  "x-amz-content-sha256": "required",
                  "content-type": "application/json"
                },
                "method": "POST",
                "request_body": "{ \"prompt\":\"${parameters.prompt}\",
 \"max_tokens_to_sample\":${parameters.max_tokens_to_sample},
 \"temperature\":${parameters.temperature},  \"anthropic_version\":
\"${parameters.anthropic_version}\" }",
                "action_type": "predict",
                "url": "https://bedrock-runtime.us-west-2.amazonaws.com/model/
anthropic.claude-instant-v1/invoke"
              }
            ],
            "credential": {
                "roleArn": "arn:aws:iam::account-id:role/opensearch-secretmanager-
role"
             },
            "parameters": {
              "endpoint": "bedrock-runtime.us-west-2.amazonaws.com",
              "content_type": "application/json",
              "auth": "Sig_V4",
              "max_tokens_to_sample": "8000",
              "service_name": "bedrock",
              "temperature": "0.0001",
              "response_filter": "$.completion",
              "region": "us-west-2",
              "anthropic_version": "bedrock-2023-05-31"
            }
          }
        }
```

```
      ]
    }
  }
}

headers = {"Content-Type": "application/json"}

r = requests.post(url, auth=awsauth, json=payload, headers=headers)
print(r.status_code)
print(r.text)
```

Modèles de flux de travail prédéfinis

Amazon OpenSearch Service fournit plusieurs modèles de flux de travail pour certains cas d'utilisation courants du machine learning (ML). L'utilisation d'un modèle simplifie les configurations complexes et fournit de nombreuses valeurs par défaut pour des cas d'utilisation tels que la recherche sémantique ou conversationnelle. Vous pouvez spécifier un modèle de flux de travail lorsque vous appelez l'API Create Workflow.

- Pour utiliser un modèle de flux de travail fourni par le OpenSearch service, spécifiez le cas d'utilisation du modèle comme paramètre de `use_case` requête.
- Pour utiliser un modèle de flux de travail personnalisé, fournissez le modèle complet dans le corps de la demande. Pour un exemple de modèle personnalisé, consultez un exemple de modèle JSON ou un exemple de modèle YAML.

Cas d'utilisation des modèles

Ce tableau fournit une vue d'ensemble des différents modèles disponibles, une description des modèles et les paramètres requis.

Cas d'utilisation du modèle	Description	Paramètres requis
`bedrock_titan_embedding_model_deploy`	Crée et déploie un modèle d'intégration Amazon Bedrock (par défaut, `titan-embed-text-v1`	`create_connector.credential.roleArn`

Cas d'utilisation du modèle	Description	Paramètres requis
`bedrock_titan_embedding_model_deploy`	Crée et déploie un modèle d'intégration multimodal Amazon Bedrock (par défaut, `titan-embed-text-v1`	`create_connector.credential.roleArn`
`cohere_embedding_model_deploy`	Crée et déploie un modèle d'intégration Cohere (par défaut, embed-english-v 3.0).	`create_connector.credential.roleArn` , `create_connector.credential.secretArn`
`cohere_chat_model_deploy`	Crée et déploie un modèle de chat Cohere (par défaut, Cohere Command).	`create_connector.credential.roleArn` , `create_connector.credential.secretArn`
`open_ai_embedding_model_deploy`	Crée et déploie un modèle d'intégration OpenAI (par défaut text-embedding-ada, -002).	`create_connector.credential.roleArn` , `create_connector.credential.secretArn`
`openai_chat_model_deploy`	Crée et déploie un modèle de chat OpenAI (par défaut, gpt-3.5-turbo).	`create_connector.credential.roleArn` , `create_connector.credential.secretArn`
`semantic_search_with_cohere_embedding`	Configure la recherche sémantique et déploie un modèle d'intégration Cohere. Vous devez fournir la clé d'API pour le modèle Cohere.	`create_connector.credential.roleArn` , `create_connector.credential.secretArn`

Cas d'utilisation du modèle	Description	Paramètres requis
`semantic_search_with_cohere_embedding_query_enricher`	Configure la recherche sémantique et déploie un modèle d'intégration Cohere. Ajoute un processeur de recherche query_enricher qui définit un ID de modèle par défaut pour les requêtes neuronales. Vous devez fournir la clé d'API pour le modèle Cohere.	`create_connector.credential.roleArn` , `create_connector.credential.secretArn`
`multimodal_search_with_bedrock_titan`	Déploie un modèle multimodal Amazon Bedrock et configure un pipeline d'ingestion avec un processeur text_image_embedding et un index K-nn pour la recherche multimodale. Vous devez fournir vos AWS informations d'identification.	`create_connector.credential.roleArn`

> **Note**
>
> Pour tous les modèles qui nécessitent un ARN secret, le secret est stocké par défaut sous le nom de clé « key » dans AWS Secrets Manager.

Modèles par défaut avec modèles préentraînés

Amazon OpenSearch Service propose deux modèles de flux de travail par défaut supplémentaires qui ne sont pas disponibles dans le service open OpenSearch source.

Cas d'utilisation du modèle	Description
`semantic_search_with_local_model`	Configure la recherche sémantique et déploie un modèle préentraîné (). `msmarco-distilbert-base-tas-b` Ajoute un processeur `neural_query_enricher` de

Cas d'utilisation du modèle	Description
	recherche qui définit un ID de modèle par défaut pour les requêtes neuronales et crée un index K-nn lié appelé « my-nlp-index ».
`hybrid_search_with_local_model`	Configure la recherche hybride et déploie un modèle préentraîné (). `msmarco-d istilbert-base-tas-b` Ajoute un processeur `neural_query_enricher` de recherche qui définit un ID de modèle par défaut pour les requêtes neuronales et crée un index K-nn lié appelé « my-nlp-index ».

Configurer des autorisations

Si vous créez un nouveau domaine avec la version 2.13 ou ultérieure, les autorisations sont déjà en place. Si vous activez l'infrastructure de flux sur un domaine de OpenSearch service préexistant avec la version 2.11 ou une version antérieure que vous mettez ensuite à niveau vers la version 2.13 ou ultérieure, vous devez définir le rôle. `flow_framework_manager` Les utilisateurs non-administrateurs doivent être mappés à ce rôle pour gérer les index à chaud des domaines utilisant le contrôle précis des accès. Pour créer manuellement le rôle `flow_framework_manager`, procédez comme suit :

1. Dans les OpenSearch tableaux de bord, accédez à Sécurité, puis sélectionnez Autorisations.
2. Choisissez Create action group (Créer un groupe d'actions) et configurez les groupes suivants :

Nom du groupe	Autorisations
`flow_framework_full_access`	• `cluster:admin/opensearch/flow_framework/*` • `cluster_monitor`
`flow_framework_read_accesss`	• `cluster:admin/opensearch/flow_framework/workflow/get` • `cluster:admin/opensearch/flow_framework/workflow/search`

Nom du groupe	Autorisations
	• `cluster:admin/opensearch/flow_framework/workflow_state/get` • `cluster:admin/opensearch/flow_framework/workflow_state/search`

3. Choisissez Roles (Rôles), puis Create role (Créer un rôle).
4. Nommez le rôle flow_framework_manager.
5. Pour Cluster permissions (Autorisations de cluster), sélectionnez `flow_framework_full_access` et`flow_framework_read_access`.
6. Pour Index, saisissez *.
7. Pour Index permissions (Autorisations d'index), sélectionnez `indices:admin/aliases/get`, `indices:admin/mappings/get` et `indices_monitor`.
8. Choisissez Créer.
9. Après avoir créé le rôle, associez-le à n'importe quel rôle d'utilisateur ou de backend qui gérera les index de la structure de flux.

Analyses de sécurité pour Amazon OpenSearch Service

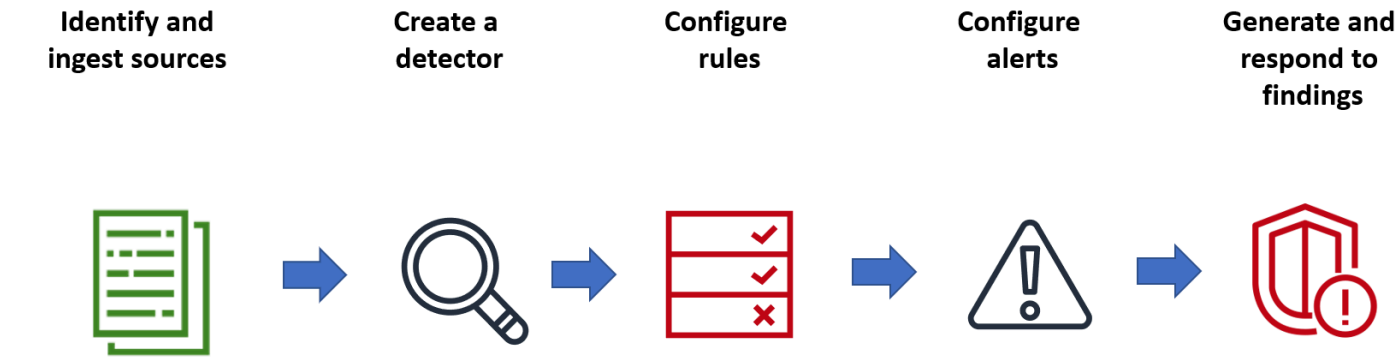
Security Analytics est une OpenSearch solution qui fournit une visibilité sur l'infrastructure de votre entreprise, surveille les activités anormales, détecte les menaces de sécurité potentielles en temps réel et déclenche des alertes vers des destinations préconfigurées. Vous pouvez surveiller les activités malveillantes à partir de vos journaux d'événements de sécurité en évaluant en permanence les règles de sécurité et en examinant les résultats de sécurité générés automatiquement. En outre, Security Analytics peut générer des alertes automatisées et les envoyer à un canal de notification spécifique, tel que Slack ou par e-mail.

Vous pouvez utiliser le plug-in Security Analytics pour détecter les menaces courantes out-of-the-box et générer des informations de sécurité critiques à partir de vos journaux d'événements de sécurité existants, tels que les journaux de pare-feu, les journaux Windows et les journaux d'audit d'authentification. Pour utiliser Security Analytics, votre domaine doit exécuter OpenSearch la version 2.5 ou ultérieure.

> **Note**
>
> Cette documentation fournit une brève présentation de Security Analytics pour Amazon OpenSearch Service. Il définit les concepts clés et fournit des étapes pour configurer les autorisations. Pour une documentation complète, y compris un guide de configuration, une référence d'API et une référence de tous les paramètres disponibles, consultez Security Analytics dans la OpenSearch documentation.

Composants et concepts d'analyse de sécurité

Un certain nombre d'outils et de fonctionnalités constituent la base du fonctionnement de Security Analytics. Les principaux composants du plugin incluent les détecteurs, les types de journaux, les règles, les résultats et les alertes.



Types de journal

OpenSearch prend en charge plusieurs types de journaux et fournit des out-of-the-box mappages pour chaque type. Vous spécifiez le type de journal et configurez un intervalle de temps lorsque vous créez un détecteur, puis Security Analytics active automatiquement un ensemble de règles pertinent qui s'exécutent à cet intervalle.

Détecteurs

Les détecteurs identifient un éventail de menaces de cybersécurité pour un type de journal dans vos index de données. Vous configurez votre détecteur pour utiliser à la fois des règles personnalisées et des règles Sigma prédéfinies qui évaluent les événements survenant dans le système. Le détecteur génère ensuite des résultats de sécurité à partir de ces événements. Pour plus d'informations sur les détecteurs, consultez la section Création de détecteurs dans la OpenSearch documentation.

Règles

Les règles de détection des menaces définissent les conditions que les détecteurs appliquent aux données de journal ingérées pour identifier un événement de sécurité. Security Analytics prend en charge l'importation, la création et la personnalisation de règles pour répondre à vos besoins, et fournit également des règles Sigma open source prédéfinies pour détecter les menaces courantes dans vos journaux. Security Analytics associe de nombreuses règles à une base de connaissances toujours croissante sur les tactiques et techniques de l'adversaire gérée par l'organisation MITRE ATT&CK. Vous pouvez utiliser les deux OpenSearch tableaux de bord ou le APIs pour créer et utiliser des règles. Pour plus d'informations sur les règles du, consultez Utilisation des règles de la OpenSearch documentation.

Conclusions

Lorsqu'un détecteur associe une règle à un événement de journal, il génère une constatation. Chaque résultat inclut une combinaison unique de règles sélectionnées, d'un type de journal et d'une sévérité de règle. Les résultats n'indiquent pas nécessairement des menaces imminentes au sein du système, mais ils isolent toujours un événement intéressant. Pour plus d'informations sur les résultats, consultez la section Utilisation des résultats dans la OpenSearch documentation.

Alerts (Alertes)

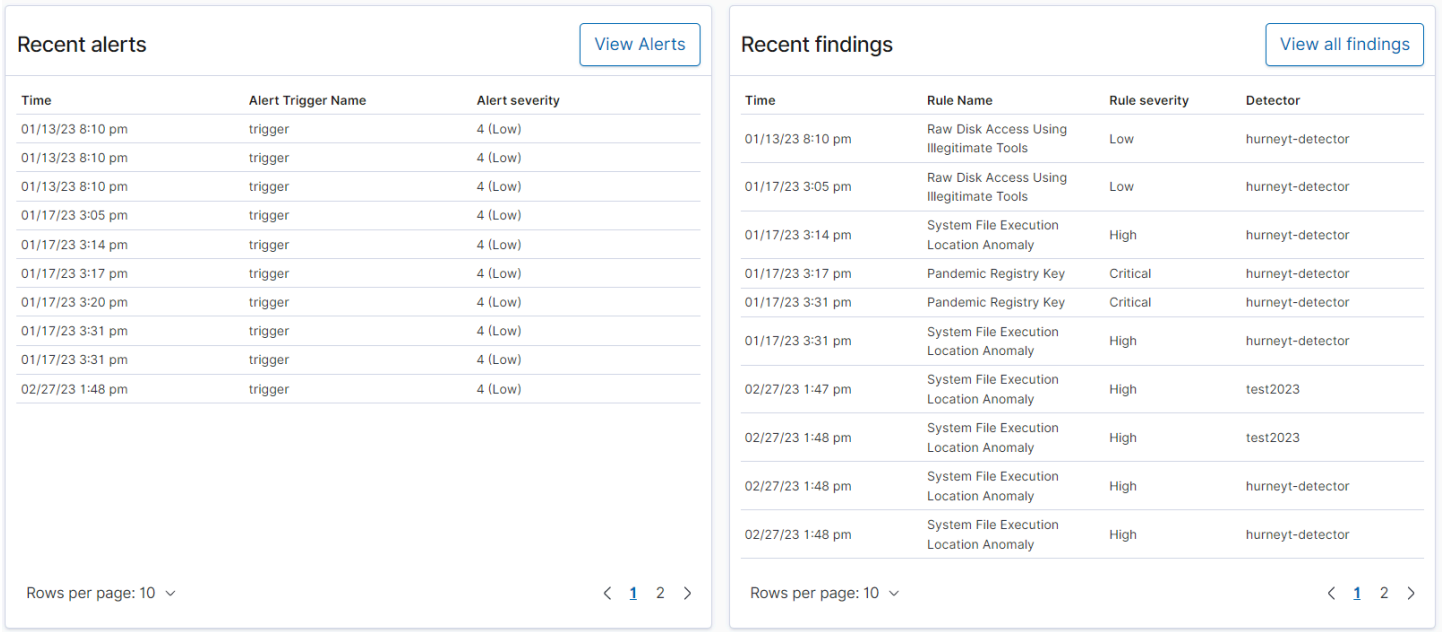
Lorsque vous créez un détecteur, vous pouvez spécifier une ou plusieurs conditions qui déclenchent une alerte. Une alerte est une notification envoyée à un canal préféré, tel que Slack ou par e-mail. Vous définissez l'alerte pour qu'elle soit déclenchée lorsque le détecteur répond à une ou plusieurs règles, et vous pouvez personnaliser le message de notification. Pour plus d'informations sur les alertes, consultez la section Utilisation des alertes dans la OpenSearch documentation.

Exploration des données de sécurité

Vous pouvez utiliser OpenSearch les tableaux de bord pour visualiser et mieux comprendre votre plugin Security Analytics. La vue d'ensemble fournit des informations telles que les résultats et le nombre d'alertes, les résultats et alertes récents, les règles de détection fréquente et la liste de vos détecteurs. Vous pouvez voir une vue récapitulative composée de plusieurs visualisations. Le graphique suivant, par exemple, montre les résultats et la tendance des alertes pour différents types de journaux sur une période donnée.



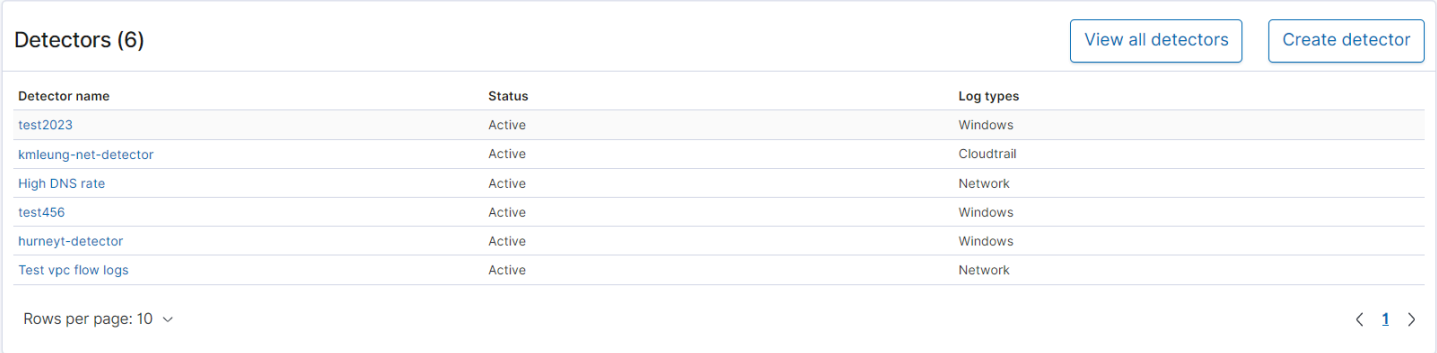
Plus bas sur la page, vous pouvez consulter vos dernières découvertes et alertes.



En outre, vous pouvez voir la répartition des règles les plus fréquemment déclenchées sur tous les détecteurs actifs. Cela peut vous aider à détecter et à étudier différents types d'activité malveillante.



Enfin, vous pouvez afficher le statut des détecteurs configurés. Depuis ce panneau, vous pouvez également accéder au flux de travail de création du détecteur.



Pour configurer votre configuration Security Analytics, créez des règles à l'aide de la page Règles et utilisez-les pour écrire des détecteurs sur la page Détecteurs. Pour une vue plus précise des résultats de vos analyses de sécurité, vous pouvez utiliser les pages Résultats et Alertes.

Configurer des autorisations

Si vous activez Security Analytics sur un domaine OpenSearch Service préexistant, le `security_analytics_manager` rôle peut ne pas y être défini. Les utilisateurs non-administrateurs doivent être mappés à ce rôle pour gérer les index à chaud des domaines utilisant le contrôle précis des accès. Pour créer manuellement le rôle `security_analytics_manager`, procédez comme suit :

1. Dans les OpenSearch tableaux de bord, accédez à Sécurité, puis sélectionnez Autorisations.
2. Choisissez Create action group (Créer un groupe d'actions) et configurez les groupes suivants :

Nom du groupe	Autorisations
security_analytics_full_access	• cluster:admin/opensearch/securityanalytics/alerts/* • cluster:admin/opensearch/securityanalytics/detector/* • cluster:admin/opensearch/securityanalytics/findings/* • cluster:admin/opensearch/securityanalytics/mapping/* • cluster:admin/opensearch/securityanalytics/rule/*
security_analytics_read_access	• cluster:admin/opensearch/securityanalytics/alerts/get • cluster:admin/opensearch/securityanalytics/detector/get • cluster:admin/opensearch/securityanalytics/detector/search • cluster:admin/opensearch/securityanalytics/findings/get • cluster:admin/opensearch/securityanalytics/mapping/get • cluster:admin/opensearch/securityanalytics/mapping/view/get • cluster:admin/opensearch/securityanalytics/rule/get • cluster:admin/opensearch/securityanalytics/rule/search

3. Choisissez Roles (Rôles), puis Create role (Créer un rôle).
4. Nommez le rôle security_analytics_manager.

5. Pour Cluster permissions (Autorisations de cluster), sélectionnez `security_analytics_full_access` et`security_analytics_read_access`.
6. Pour Index, saisissez `*`.
7. Pour les autorisations d'indexation, sélectionnez `indices:admin/mapping/put` et`indices:admin/mappings/get`.
8. Choisissez Créer.
9. Après avoir créé le rôle, mappez-le à n'importe quel rôle utilisateur ou backend qui gèrera les index Security Analytics.

Résolution des problèmes

Aucune erreur d'index de ce type

Si vous n'avez aucun détecteur et que vous ouvrez le tableau de bord de Security Analytics, vous verrez peut-être une notification en bas à droite indiquant `[index_not_found_exception] no such index [.opensearch-sap-detectors-config]` que Vous pouvez ignorer cette notification, qui disparaît en quelques secondes et ne réapparaîtra pas une fois que vous aurez créé un détecteur.

Observabilité dans Amazon Amazon Service OpenSearch

L'installation par défaut de OpenSearch Dashboards for Amazon OpenSearch Service OpenSearch Le plugin nécessite version OpenSearch 1.2 ou ultérieure.

Le plugin Observability offre une expérience unifiée pour la collecte et la surveillance des métriques, des journaux et des traces provenant de sources de données communes. La collecte et la surveillance des données en un seul point permettent une end-to-end observabilité complète de l'ensemble de votre infrastructure.

> **Note**
>
> Cette documentation fournit un bref aperçu de l'observabilité en OpenSearch service. Pour une documentation complète du plugin Observability, y compris les autorisations, consultez Observability.

Le processus d'exploration des données est différent pour chacun. Si vous débutez dans l'exploration de données et la création de visualisations, nous vous recommandons d'essayer un flux comme le suivant.

Explorez vos données grâce à l'analytique des événements

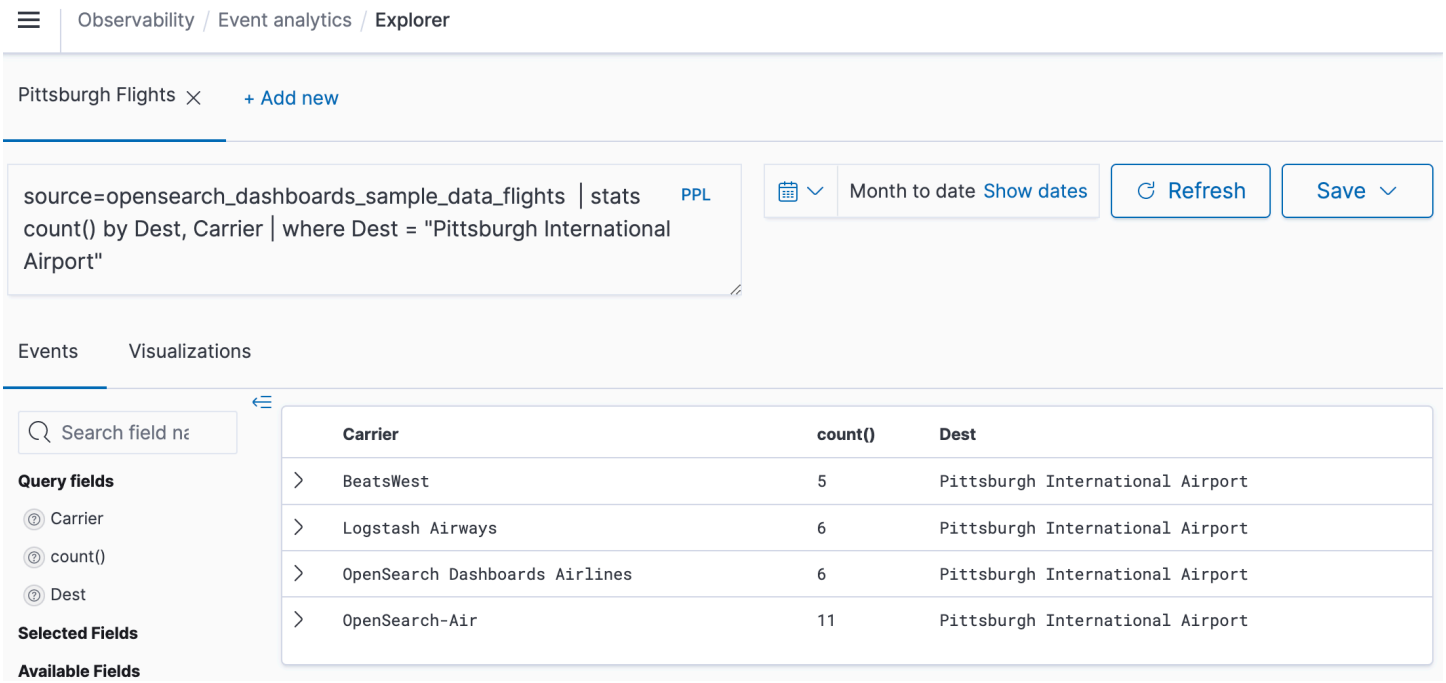
Pour démarrer, supposons que vous collectiez des données de vol dans votre domaine de OpenSearch service et que vous vouliez savoir quelle compagnie aérienne a eu le plus de vols arrivant à l'aéroport international de Pittsburgh le mois dernier. Vous écrivez la requête PPL suivante :

```
source=opensearch_dashboards_sample_data_flights |
    stats count() by Dest, Carrier |
    where Dest = "Pittsburgh International Airport"
```

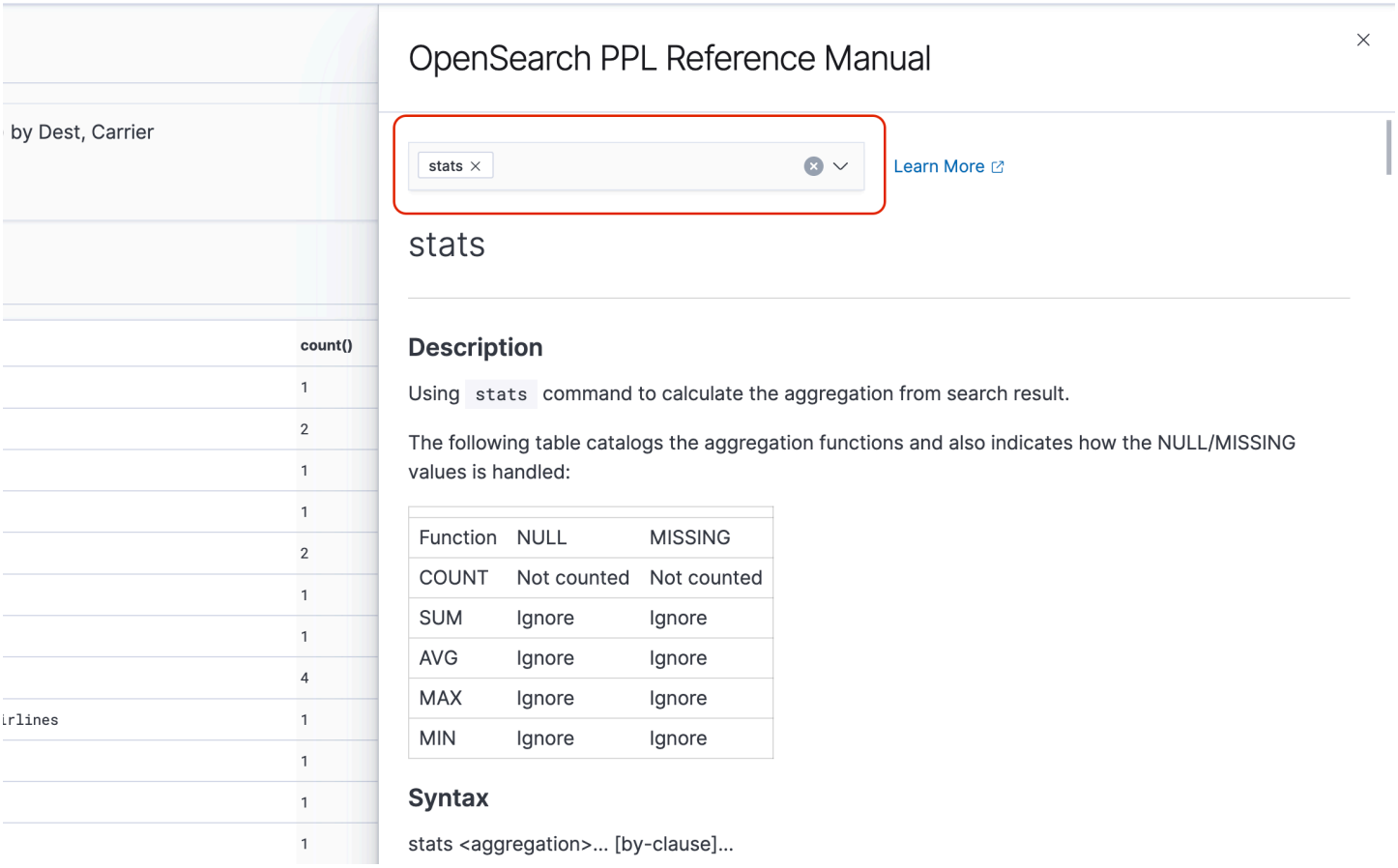
Cette requête extrait les données de l'index nommé `opensearch_dashboards_sample_data_flights`. Il utilise ensuite la commande `stats` pour

obtenir un nombre total de vols et les regrouper en fonction de l'aéroport de destination et de la compagnie aérienne. Enfin, il utilise la clause `where` pour filtrer les résultats sur les vols arrivant à l'aéroport international de Pittsburgh.

Voici à quoi ressemblent les données lorsqu'elles sont affichées sur le mois dernier :



Vous pouvez choisir le bouton PPL dans l'éditeur de requêtes pour obtenir des informations d'utilisation et des exemples pour chaque commande PPL :



Examinons un exemple plus complexe, qui demande des informations sur les retards de vol :

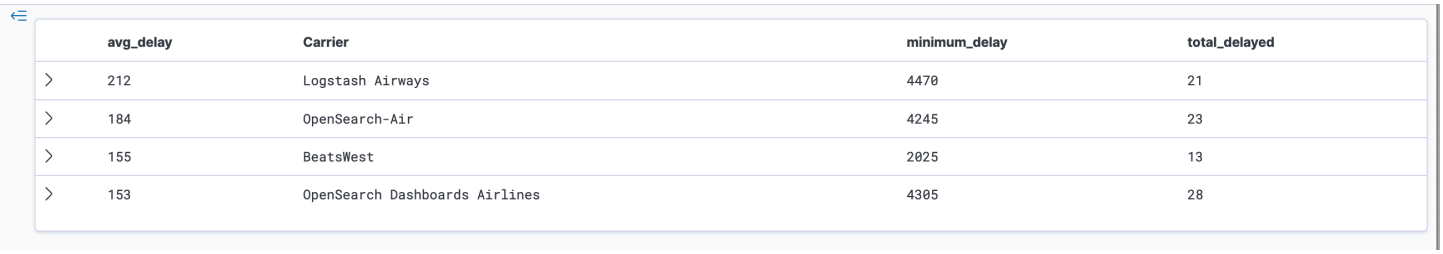
```
source=opensearch_dashboards_sample_data_flights |
    where FlightDelayMin > 0 |
    stats sum(FlightDelayMin) as minimum_delay, count() as total_delayed by Carrier,
 Dest |
    eval avg_delay=minimum_delay / total_delayed |
    sort - avg_delay
```

Chaque commande de la requête a un impact sur la sortie finale :

- `source=opensearch_dashboards_sample_data_flights` – extrait les données du même index que l'exemple précédent
- `where FlightDelayMin > 0` – filtre les données sur les vols retardés
- `stats sum(FlightDelayMin) as minimum_delay, count() as total_delayed by Carrier` – pour chaque compagnie aérienne, obtient le retard minimum total et le nombre total de vols retardés

- `eval avg_delay=minimum_delay / total_delayed` – calcule le temps de retard moyen pour chaque compagnie aérienne en divisant le retard minimum par le nombre total de vols retardés
- `sort - avg_delay` – trie les résultats en fonction du retard moyen dans l'ordre décroissant

Avec cette requête, vous pouvez déterminer que OpenSearch Dashboards Airlines a, en moyenne, moins de retards.



	avg_delay	Carrier	minimum_delay	total_delayed
>	212	Logstash Airways	4470	21
>	184	OpenSearch-Air	4245	23
>	155	BeatsWest	2025	13
>	153	OpenSearch Dashboards Airlines	4305	28

Plus d'exemples de requêtes PPL sous Queries and Visualizations (Requêtes et visualisations) sur la page Event analytics (Analytique des événements).

Créer des visualisations

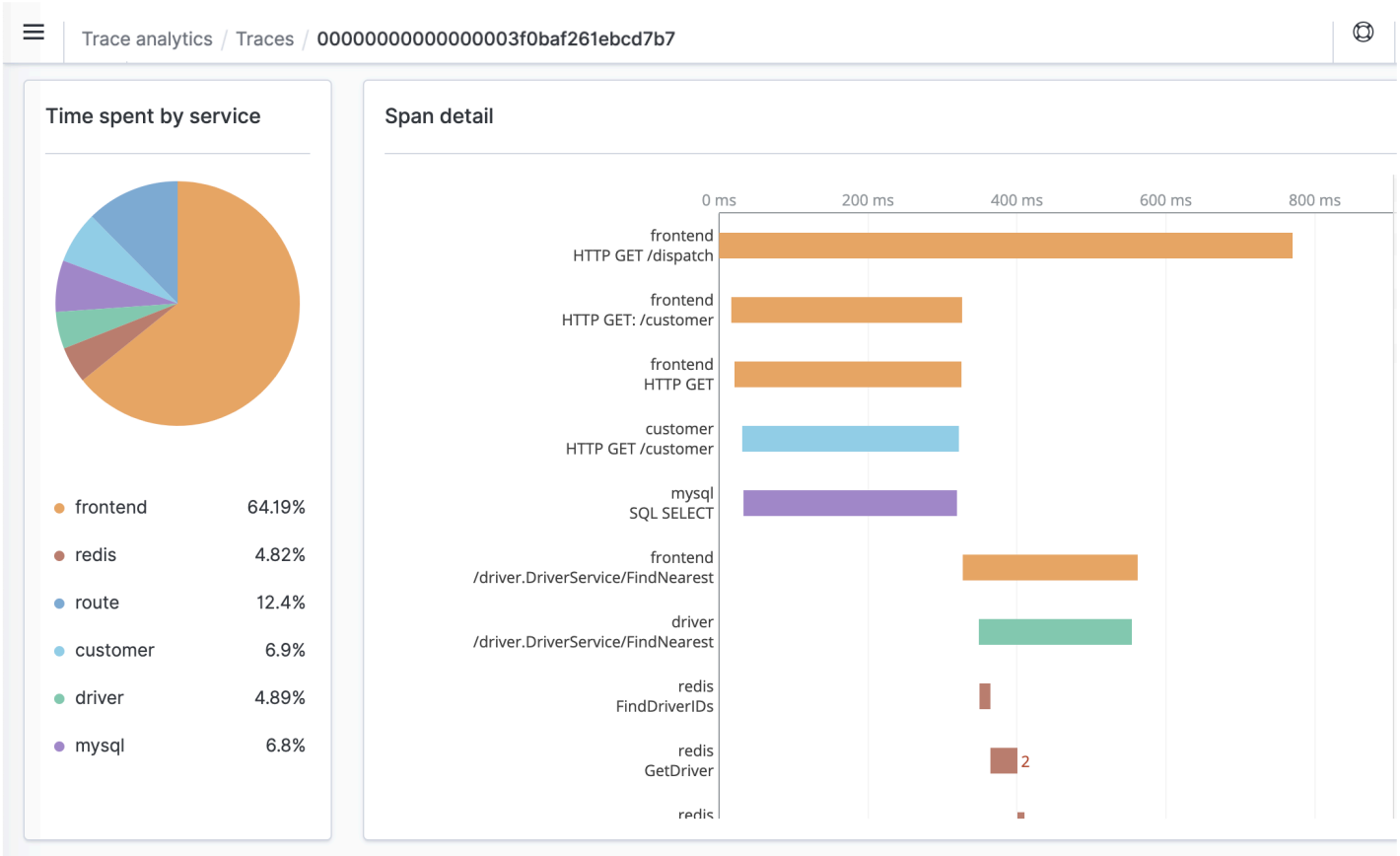
Une fois que vous avez correctement interrogé les données qui vous intéressent, vous pouvez enregistrer ces requêtes sous forme de visualisations :



Ajoutez ensuite ces visualisations aux panneaux opérationnels pour comparer différents éléments de données. Exploitez les blocs-notes pour combiner différentes visualisations et blocs de code que vous pouvez partager avec les membres de l'équipe.

Plongez plus profondément avec Trace Analytics

Trace Analytics permet de visualiser le flux d'événements dans vos OpenSearch données afin d'identifier et de résoudre les problèmes de performances dans les applications distribuées.



Trace Analytics pour Amazon OpenSearch Service

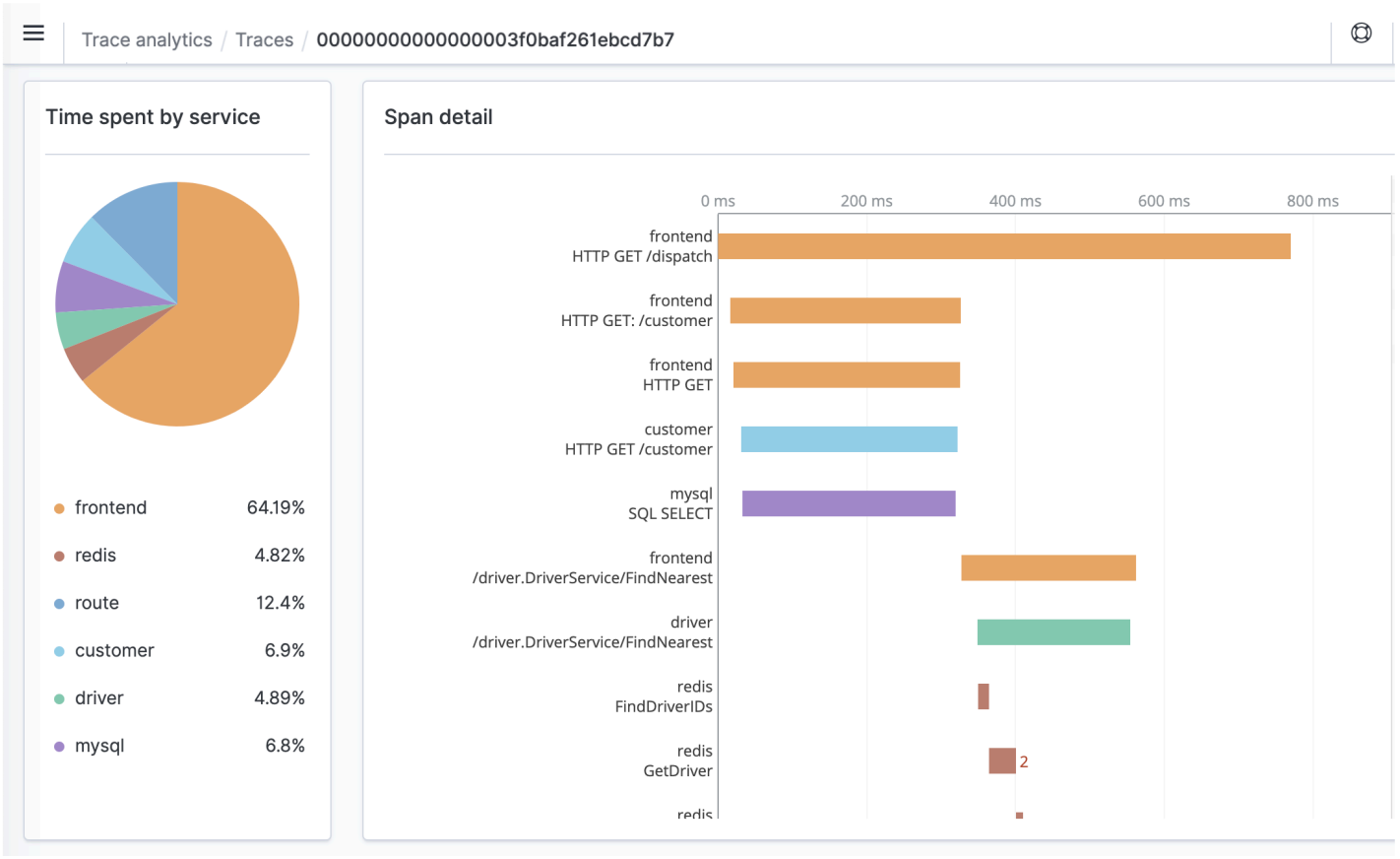
Vous pouvez utiliser Trace Analytics, qui fait partie du plugin OpenSearch Observability, pour analyser les données de trace des applications distribuées. Trace Analytics nécessite OpenSearch Elasticsearch version 7.9 ou ultérieure.

Dans une application distribuée, une seule opération, telle qu'un utilisateur cliquant sur un bouton, peut déclencher une série étendue d'événements. Par exemple, le frontend de l'application peut appeler un service backend, qui appelle un autre service, qui interroge une base de données, qui

traite la requête et renvoie un résultat. Ensuite, le premier service backend envoie une confirmation au frontend, qui met à jour l'interface utilisateur.

Vous pouvez utiliser Trace Analytics pour mieux visualiser ce flux d'événements et identifier les problèmes de performances.

> **Note**
>
> Cette documentation fournit un bref aperçu de Trace Analytics. Pour une documentation complète, consultez Trace Analytics dans la OpenSearch documentation open source.



Prérequis

Trace Analytics vous oblige à ajouter de l'instrumentation à votre application et à générer des données de trace à l'aide d'une bibliothèque OpenTelemetry compatible telle que Jaeger ou Zipkin. Cette étape intervient en dehors du OpenSearch Service. La OpenTelemetry documentation AWS

Distro contient des exemples d'applications pour de nombreux langages de programmation qui peuvent vous aider à démarrer, y compris Java, Python, Go et JavaScript.

Après avoir ajouté une instrumentation à votre application, le OpenTelemetryCollecteur reçoit les données de l'application et les formate en OpenTelemetry données. Consultez la liste des récepteurs sur GitHub. AWS Distro for OpenTelemetry inclut un récepteur pour AWS X-Ray.

Enfin, vous pouvez utiliser OpenSearch Ingestion d'Amazon pour formater ces OpenTelemetry données pour les utiliser avec OpenSearch.

OpenTelemetry Configuration des échantillons de collecteur

Pour utiliser le OpenTelemetry Collecteur avec OpenSearch Ingestion d'Amazon, essayez l'exemple de configuration suivant :

```
extensions:
  sigv4auth:
    region: "us-east-1"
    service: "osis"

receivers:
  jaeger:
    protocols:
      grpc:

exporters:
  otlphttp:
    traces_endpoint: "https://pipeline-endpoint.us-east-1.osis.amazonaws.com/
opentelemetry.proto.collector.trace.v1.TraceService/Export"
    auth:
      authenticator: sigv4auth
    compression: none

service:
  extensions: [sigv4auth]
  pipelines:
    traces:
      receivers: [jaeger]
      exporters: [otlphttp]
```

OpenSearch Configuration de l'échantillon d'ingestion

Pour envoyer des données de suivi à un domaine OpenSearch de service, essayez l'exemple de configuration de pipeline d' OpenSearch ingestion suivant. Pour obtenir des instructions sur la création d'un pipeline, consultezthe section called “Création de pipelines”.

```
version: "2"
otel-trace-pipeline:
  source:
    otel_trace_source:
      "/${pipelineName}/ingest"
  processor:
    - trace_peer_forwarder:
  sink:
    - pipeline:
        name: "trace_pipeline"
    - pipeline:
        name: "service_map_pipeline"
trace-pipeline:
  source:
    pipeline:
      name: "otel-trace-pipeline"
  processor:
    - otel_traces:
  sink:
    - opensearch:
        hosts: ["https://domain-endpoint"]
        index_type: trace-analytics-raw
        aws:
          # IAM role that OpenSearch Ingestion assumes to access the domain sink
          sts_role_arn: "arn:aws:iam::{account-id}:role/pipeline-role"
          region: "us-east-1"

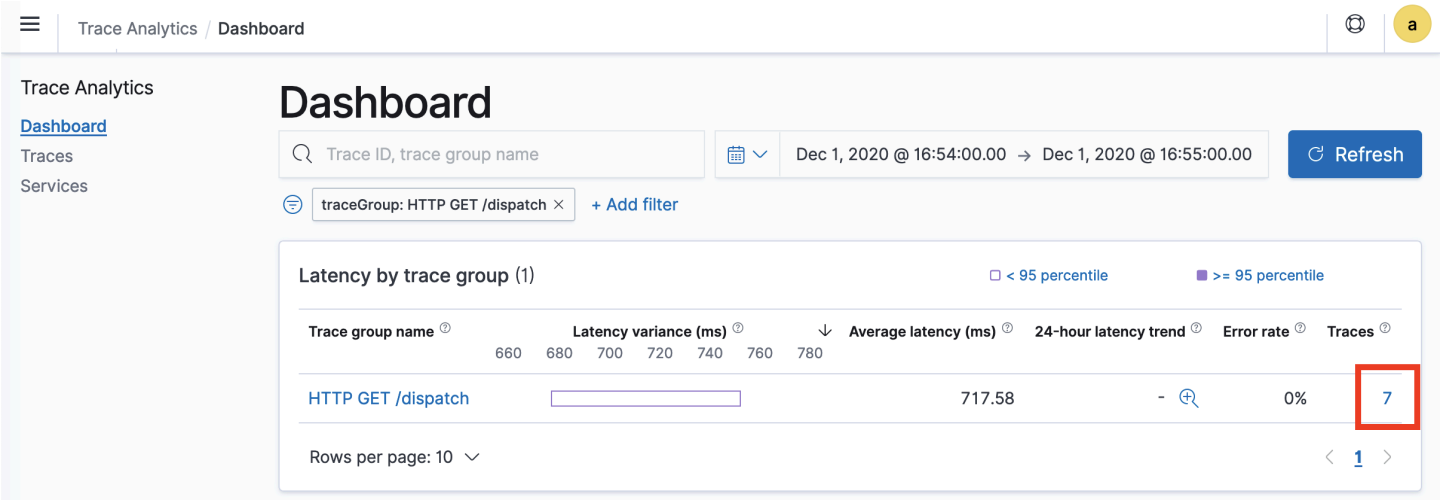
service-map-pipeline:
  source:
    pipeline:
      name: "otel-trace-pipeline"
  processor:
    - service_map:
  sink:
    - opensearch:
        hosts: ["https://domain-endpoint"]
        index_type: trace-analytics-service-map
```

```
    aws:
      # IAM role that the pipeline assumes to access the domain sink
      sts_role_arn: "arn:aws:iam::{account-id}:role/pipeline-role"
      region: "us-east-1"
```

Le rôle de pipeline que vous spécifiez dans l'`sts_role_arn`option doit disposer d'autorisations en écriture pour le récepteur. Pour obtenir des instructions sur la configuration des autorisations pour le rôle de pipeline, consultezthe section called “Configuration des rôles et des utilisateurs”.

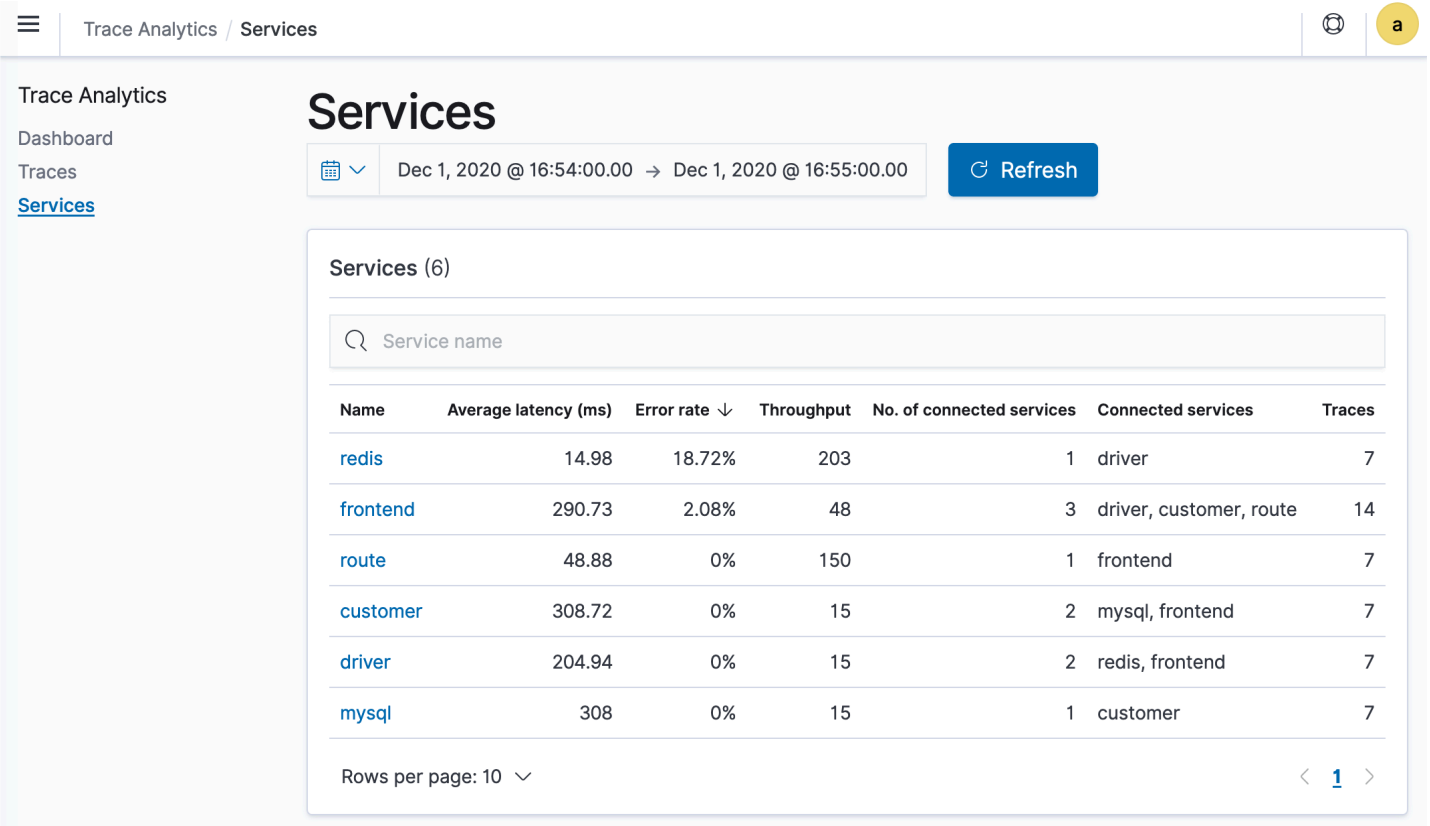
Exploration des données de suivi

La vue Tableau de bord regroupe les traces par méthode HTTP et chemin d'accès pour vous permettre de voir la latence moyenne, le taux d'erreurs et les tendances associées à une opération particulière. Pour une vue plus ciblée, essayez de filtrer par nom de groupe de trace.



Pour explorer les traces qui composent un groupe de traces, choisissez le nombre de traces dans la colonne de droite. Choisissez ensuite une trace individuelle pour obtenir un résumé détaillé.

La vue Services répertorie tous les services de l'application, ainsi qu'une carte interactive qui montre comment les différents services se connectent les uns aux autres. Contrairement au tableau de bord (qui permet d'identifier les problèmes par opération), la carte de service vous aide à identifier les problèmes par service. Essayez de trier par taux d'erreurs ou latence afin d'identifier les possibles zones problématiques de votre application.



Interrogation des données Amazon OpenSearch Service en utilisant le langage PPL (Piped Processing Language)

Le langage PPL (Piped Processing Language) est un langage de requête qui vous permet d'utiliser la syntaxe avec barre | verticale () pour interroger les données stockées dans Amazon OpenSearch Service. PPL nécessite Elasticsearch 7.9 OpenSearch ou version ultérieure.

> **Note**
>
> Cette documentation fournit une brève présentation de PPL pour Amazon OpenSearch Service. Pour les étapes détaillées et une référence complète des commandes, voir PPL dans la OpenSearch documentation open source.

La syntaxe PPL se compose de commandes délimitées par une barre verticale (|) où les données circulent de gauche à droite à travers chaque pipeline. Par exemple, la syntaxe PPL pour rechercher le nombre d'hôtes avec des erreurs HTTP 403 ou 503, les agréger par hôte, puis les trier par ordre d'impact est la suivante :

```
source = dashboards_sample_data_logs | where response='403' or response='503' | stats
 count(request) as request_count by host, response | sort -request_count
```

Pour commencer, choisissez Query Workbench (Workbench de requête) dans OpenSearch Dashboards, puis sélectionnez PPL. Utilisez l'opération `bulk` pour indexer quelques exemples de données :

```
PUT accounts/_bulk?refresh
{"index":{"_id":"1"}}
{"account_number":1,"balance":39225,"firstname":"Amber","lastname":"Duke","age":32,"gender":"M"
 Holmes
 Lane","employer":"Pyrami","email":"amberduke@pyrami.com","city":"Brogan","state":"IL"}
{"index":{"_id":"6"}}
{"account_number":6,"balance":5686,"firstname":"Hattie","lastname":"Bond","age":36,"gender":"M"
 Bristol
 Street","employer":"Netagy","email":"hattiebond@netagy.com","city":"Dante","state":"TN"}
{"index":{"_id":"13"}}
{"account_number":13,"balance":32838,"firstname":"Nanette","lastname":"Bates","age":28,"gender"
 Mady Street","employer":"Quility","city":"Nogal","state":"VA"}
{"index":{"_id":"18"}}
{"account_number":18,"balance":4180,"firstname":"Dale","lastname":"Adams","age":33,"gender":"M"
 Hutchinson Court","email":"daleadams@boink.com","city":"Orick","state":"MD"}
```

L'exemple suivant renvoie les champs `firstname` et `lastname` pour les documents d'un index de comptes dont l'`age` est supérieur à 18 :

```
search source=accounts | where age > 18 | fields firstname, lastname
```

Exemple de réponse

id	firstname	lastname
0	Amber	Duke
1	Hattie	Bond
2	Nanette	Bates
3	Dale	Adams

Vous pouvez utiliser un ensemble complet de commandes en lecture seule comme `search`, `where`, `fields`, `rename`, `dedup`, `stats`, `sort`, `eval`, `head`, `top` et `rare`. Le plug-in PPL prend en charge toutes les fonctions SQL, y compris les opérateurs et expressions mathématiques, trigonométriques, date-heure, chaîne, agrégat et avancés. Pour en savoir plus, veuillez consulter le manuel de référence sur le OpenSearch PPL.

Bonnes pratiques opérationnelles pour Amazon OpenSearch Service

Ce chapitre décrit les meilleures pratiques relatives à l'exploitation des domaines Amazon OpenSearch Service et inclut des directives générales qui s'appliquent à de nombreux cas d'utilisation. Chaque charge de travail est unique, avec des caractéristiques propres, de sorte qu'aucune recommandation générique ne convient exactement à chaque cas d'utilisation. La bonne pratique la plus importante consiste à déployer, tester et régler vos domaines dans un cycle continu pour trouver la configuration, la stabilité et le coût optimaux pour votre charge de travail.

Rubriques

- Surveillance et alertes
- Stratégie de partition
- Stabilité
- Performances
- Sécurité
- Optimisation des coûts
- Dimensionnement des domaines Amazon OpenSearch Service
- Mise à l'échelle d'une capacité de plusieurs péta-octets dans Amazon Service OpenSearch
- Nœuds de coordination dédiés dans Amazon OpenSearch Service
- Nœuds principaux dédiés dans Amazon OpenSearch Service

Surveillance et alertes

Les meilleures pratiques suivantes s'appliquent à la surveillance de vos domaines OpenSearch de service.

Configuration des CloudWatch alarmes

OpenSearch Le service transmet des indicateurs de performance à Amazon CloudWatch. Passez régulièrement en revue les métriques de votre cluster et de votre instance et configurez les CloudWatch alarmes recommandées en fonction des performances de votre charge de travail.

Activer la publication des journaux

OpenSearch Le service expose les journaux OpenSearch d'erreurs, les journaux lents de recherche, l'indexation des journaux lents et les journaux d'audit dans Amazon CloudWatch Logs. Les journaux lents de recherche, les journaux lents d'indexation et les journaux d'erreurs permettent de résoudre les problèmes de performances et de stabilité. Les journaux d'audit, qui ne sont disponibles que si vous activez le contrôle précis des accès, suivent l'activité des utilisateurs. Pour plus d'informations, consultez la section Logs de la OpenSearch documentation.

Les journaux lents de recherche et les journaux lents d'indexation sont des outils importants pour comprendre et résoudre les problèmes de performance de vos opérations de recherche et d'indexation. Activez la livraison de journaux lents de recherche et d'indexation pour tous les domaines de production. Vous devez également configurer des seuils de journalisation, sinon les journaux ne CloudWatch seront pas capturés.

Stratégie de partition

Les partitions répartissent votre charge de travail entre les nœuds de données de votre domaine OpenSearch de service. Des index correctement configurés peuvent contribuer à améliorer les performances globales du domaine.

Lorsque vous envoyez des données à OpenSearch Service, vous les envoyez vers un index. Un index est comparable à une table de base de données, les documents étant les lignes et les champs les colonnes. Lorsque vous créez l'index, vous indiquez le OpenSearch nombre de partitions principales que vous souhaitez créer. Les partitions principales sont des partitions indépendantes de l'ensemble de données complet. OpenSearch Le service distribue automatiquement vos données sur les partitions principales d'un index. Vous pouvez également configurer des réplicas de l'index. Chaque partition réplica comprend un jeu complet de copies des partitions principales pour cet index.

OpenSearch Le service mappe les partitions de chaque index sur les nœuds de données de votre cluster. Il garantit que les partitions principales et réplicas de l'index résident sur des nœuds de données différents. Le premier réplica garantit que vous disposez de deux copies des données dans l'index. Vous devez toujours utiliser au moins un réplica. Des réplicas supplémentaires fournissent une redondance et une capacité de lecture supplémentaires.

OpenSearch envoie des demandes d'indexation à tous les nœuds de données contenant des fragments appartenant à l'index. Il envoie les demandes d'indexation d'abord aux nœuds de données contenant des partitions principales, puis aux nœuds de données contenant des partitions de réplica.

Les requêtes de recherche sont acheminées par le nœud coordinateur vers une partition principale ou de réplica pour toutes les partitions appartenant à l'index.

Par exemple, pour un index avec cinq partitions principales et un réplica, chaque requête d'indexation implique 10 partitions. En revanche, les requêtes de recherche sont envoyées à n partitions, où n est le nombre de partitions principales. Pour un index avec cinq partitions principales et un réplica, chaque requête de recherche implique cinq partitions (principales ou de réplica) de cet index.

Déterminer le nombre de partitions et de nœuds de données

Utilisez les bonnes pratiques suivantes pour déterminer le nombre de partitions et de nœuds de données pour votre domaine.

Taille de la partition : la taille des données sur le disque est un résultat direct de la taille de vos données source, et elle change au fur et à mesure que vous indexez plus de données. Le source-to-index ratio peut varier énormément, de 1:10 à 10:1 ou plus, mais il se situe généralement autour de 1:1,10. Vous pouvez utiliser ce ratio pour prévoir la taille de l'index sur le disque. Vous pouvez également indexer certaines données et récupérer les tailles d'index réelles pour déterminer le ratio pour votre charge de travail. Une fois que vous avez prédit la taille de l'index, définissez un nombre de partitions de sorte que chaque partition soit comprise entre 10 et 30 Gio (pour les charges de travail de recherche) ou entre 30 et 50 Gio (pour les charges de travail de journaux). 50 Gio devrait être le maximum – assurez-vous de planifier la croissance.

Nombre de partitions : la distribution des partitions aux nœuds de données a un impact important sur les performances d'un domaine. Lorsque vous avez des index contenant plusieurs partitions, essayez de faire en sorte que le nombre de partitions soit un multiple du nombre de nœuds de données. Cela permet de garantir que les partitions sont réparties de manière uniforme entre les nœuds de données et d'éviter les nœuds chauds. Par exemple, si vous avez 12 partitions principales, votre nombre de nœuds de données devrait être de 2, 3, 4, 6 ou 12. Toutefois, le nombre de partitions est secondaire par rapport à la taille des partitions – si vous avez 5 Gio de données, vous devez toujours utiliser une seule partition.

Partitions par nœud de données : le nombre total de partitions qu'un nœud peut contenir est proportionnel à la mémoire de tas Java virtual machine (JVM) du nœud. Visez 25 partitions ou moins par Gio de mémoire de tas. Par exemple, un nœud avec 32 Gio de mémoire de tas ne doit pas contenir plus de 800 partitions. Bien que la distribution des partitions puisse varier en fonction de vos modèles de charge de travail, il existe une limite de 1 000 partitions par nœud pour Elasticsearch, de OpenSearch 1,1 à 2,15 et de 4 000 pour les versions 2.17 et supérieures. OpenSearch L'API cat/

allocation fournit une vue rapide du nombre de partitions et du stockage total des partitions sur les nœuds de données.

Ratio partition/CPU : lorsqu'une partition est impliquée dans une demande d'indexation ou de recherche, elle utilise un vCPU pour traiter la demande. Comme bonne pratique, utilisez un point d'échelle initial de 1,5 vCPU par partition. Si votre type d'instance est de 8 VCPUs, définissez le nombre de nœuds de données de manière à ce que chaque nœud ne contienne pas plus de six partitions. Notez qu'il s'agit d'une approximation. Assurez-vous de tester votre charge de travail et de mettre votre cluster à l'échelle en conséquence.

Pour des recommandations sur le volume de stockage, la taille des partitions et le type d'instance, consultez les ressources suivantes :

- the section called “Dimensionnement des domaines”
- the section called “Mise à l'échelle d'une capacité de plusieurs péta-octets”

Éviter l'asymétrie de stockage

L'asymétrie de stockage se produit lorsqu'un ou plusieurs nœuds au sein d'un cluster détient une proportion plus élevée de stockage pour un ou plusieurs index que les autres. Les indications d'une asymétrie de stockage comprennent une utilisation inégale de l'UC, une latence intermittente et inégale, et une mise en file d'attente inégale sur les nœuds de données. Pour déterminer si vous avez des problèmes d'asymétrie, consultez les sections de dépannage suivantes :

- the section called “Asymétrie des partitions et de stockage des nœuds”
- the section called “Asymétrie des partitions et du stockage des index”

Stabilité

Les meilleures pratiques suivantes s'appliquent au maintien d'un domaine de OpenSearch service stable et sain.

Tenez-vous au courant de OpenSearch

Mises à jour du logiciel de service

OpenSearch Le service publie régulièrement des mises à jour logicielles qui ajoutent des fonctionnalités ou améliorent vos domaines. Les mises à jour ne modifient pas la OpenSearch

version du moteur Elasticsearch. Nous vous recommandons de planifier une période récurrente pour exécuter l'opération d'DescribeDomainAPI et de lancer une mise à jour du logiciel de service si `UpdateStatus` c'est le cas`ELIGIBLE`. Si vous ne mettez pas à jour votre domaine dans un certain délai (généralement deux semaines), le OpenSearch Service effectue automatiquement la mise à jour.

OpenSearch mises à niveau de version

OpenSearch Le service ajoute régulièrement la prise en charge des versions gérées par la communauté de. OpenSearch Effectuez toujours une mise à niveau vers les dernières OpenSearch versions dès qu'elles sont disponibles.

OpenSearch Le service met à niveau simultanément OpenSearch les deux OpenSearch tableaux de bord (ou Elasticsearch et Kibana si votre domaine utilise un ancien moteur). Si le cluster dispose de nœuds maîtres dédiés, les mises à niveau sont exécutées sans temps d'arrêt. Dans le cas contraire, le cluster risque de ne pas répondre pendant plusieurs secondes après la mise à niveau pendant qu'il élit un nœud maître. OpenSearch Les tableaux de bord peuvent être indisponibles pendant une partie ou la totalité de la mise à niveau.

Il existe deux façons de mettre à niveau un domaine :

- Mise à niveau sur place : cette option est plus simple car vous conservez le même cluster.
- Mise à niveau d'instantané/restauration : cette option est bonne pour tester de nouvelles versions sur un nouveau cluster ou pour migrer entre des clusters.

Quel que soit le processus de mise à niveau que vous utilisez, nous vous recommandons de conserver un domaine destiné uniquement au développement et aux tests, et d'en installer la nouvelle version avant de mettre à niveau votre domaine de production. Choisissez Development and testing (Développement et test) pour le type de déploiement lors de la création du domaine de test. Assurez-vous de mettre à niveau tous les clients vers des versions compatibles immédiatement après la mise à niveau du domaine.

Améliorez les performances des instantanés

Pour éviter que le traitement de votre instantané ne soit bloqué, le type d'instance du nœud maître dédié doit correspondre au nombre de partitions. Pour de plus amples informations, veuillez consulter the section called “Choix des types d'instance pour les nœuds principaux dédiés”. En outre, chaque nœud ne doit pas contenir plus de 25 partitions recommandées par GiB de mémoire Java. Pour de plus amples informations, veuillez consulter the section called “Choix du nombre de partitions”.

Activer les nœuds principaux dédiés

Les nœuds principaux dédiés améliorent la stabilité du cluster. Un nœud principal dédié effectue les tâches de gestion du cluster, mais ne détient pas les données d'index et ne répond pas aux demandes des clients. Ce déchargement des tâches de gestion du cluster augmente la stabilité de votre domaine et permet d'effectuer certaines modifications de configuration sans temps d'arrêt.

Activez et utilisez trois nœuds principaux dédiés pour une stabilité de domaine optimale dans trois zones de disponibilité. Le déploiement avec Multi-AZ with Standby permet de configurer trois nœuds principaux dédiés pour vous. Pour des recommandations sur le type d'instance, consultez the section called “Choix des types d'instance pour les nœuds principaux dédiés”.

Déployer sur plusieurs zones de disponibilité

Pour éviter la perte de données et minimiser le temps d'arrêt du cluster en cas d'interruption de service, vous pouvez répartir les nœuds sur deux ou trois zones de disponibilité dans la même Région AWS. La meilleure pratique consiste à effectuer un déploiement à l'aide de la technologie Multi-AZ avec veille, qui configure trois zones de disponibilité, dont deux zones sont actives et une sert de veille, et avec deux répliques de fragments par index. Cette configuration permet au OpenSearch Service de distribuer des répliques à des partitions différentes AZs de leurs partitions principales correspondantes. Il n'y a aucun frais de transfert de données entre zones de disponibilité pour les communications entre clusters.

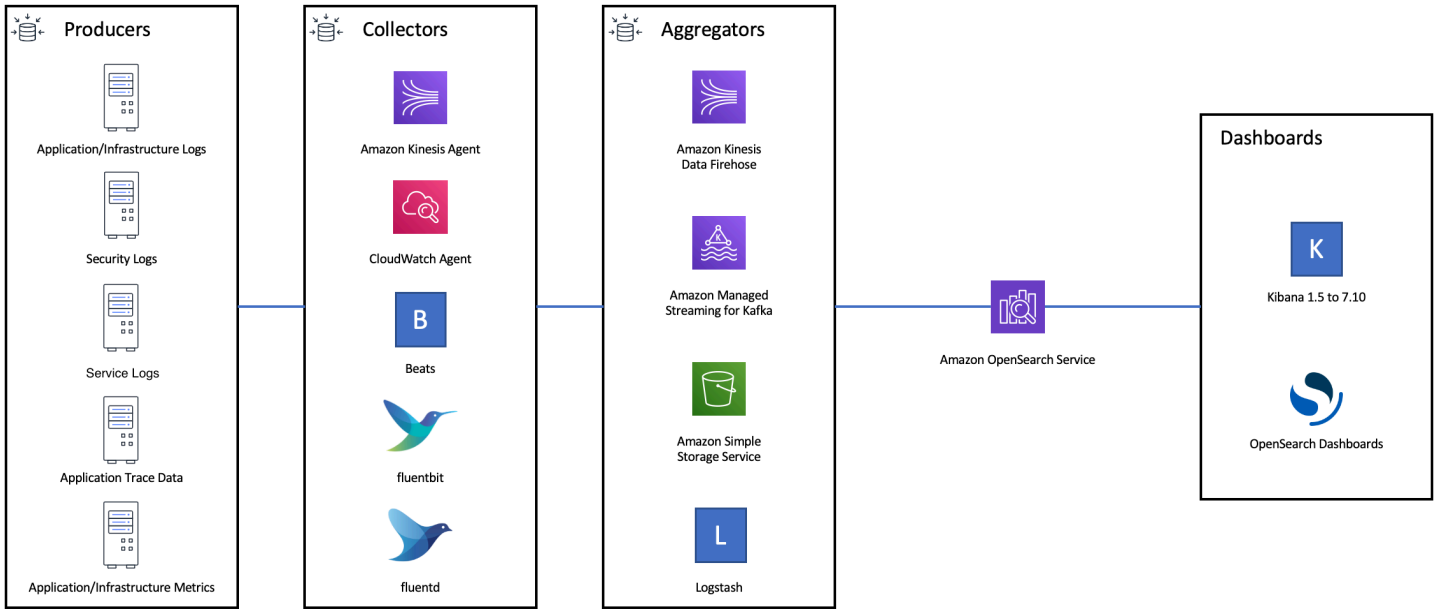
Les zones de disponibilité sont des emplacements isolés au sein de chaque région . Avec une configuration à deux zones de disponibilité, la perte d'une zone signifie que vous perdez la moitié de la capacité totale du domaine. Le passage à trois zones de disponibilité réduit davantage l'impact de la perte d'une seule zone.

Contrôler le flux d'ingestion et la mise en mémoire tampon

Nous vous recommandons de limiter le nombre total de demandes à l'aide de l'opération d'API _bulk. Il est plus efficace d'envoyer une demande `_bulk` contenant 5 000 documents que d'envoyer 5 000 demandes contenant un seul document.

Pour une stabilité opérationnelle optimale, il est parfois nécessaire de limiter ou même de mettre en pause le flux en amont des demandes d'indexation. La limitation du débit des demandes d'indexation est un mécanisme important pour gérer les pics de demandes inattendus ou occasionnels qui pourraient autrement inonder le cluster. Envisagez d'intégrer un mécanisme de contrôle de flux dans votre architecture en amont.

Le diagramme suivant montre plusieurs options de composants pour une architecture d'ingestion de journaux. Configurez la couche d'agrégation afin de disposer d'un espace suffisant pour mettre en mémoire tampon les données entrantes en cas de pic de trafic soudain et de brève maintenance du domaine.



Créer des mappages pour les charges de travail de recherche

Pour les charges de travail de recherche, créez des mappages qui définissent le mode de stockage et OpenSearch d'indexation des documents et de leurs champs. Définissez `dynamic` sur `strict` afin d'éviter l'ajout accidentel de nouveaux champs.

```
PUT my-index
{
  "mappings": {
    "dynamic": "strict",
    "properties": {
      "title": { "type" : "text" },
      "author": { "type" : "integer" },
      "year": { "type" : "text" }
    }
  }
}
```

Utiliser des modèles d'index

Vous pouvez utiliser un modèle d'index pour indiquer OpenSearch comment configurer un index lors de sa création. Configurez les modèles d'index avant de créer les index. Ensuite, lorsque vous créez un index, celui-ci hérite des paramètres et des mappages du modèle. Vous pouvez appliquer plus d'un modèle à un seul index, ce qui vous permet de spécifier les paramètres dans un modèle et les mappages dans un autre. Cette stratégie permet d'utiliser un modèle pour les paramètres communs à plusieurs index, et des modèles distincts pour les paramètres et mappages plus spécifiques.

Les paramètres suivants sont utiles à configurer dans les modèles :

- Nombre de partitions primaires et de réplica
- Intervalle d'actualisation (fréquence d'actualisation et de mise à disposition des modifications récentes de l'index pour la recherche)
- Contrôle de mappage dynamique
- Mappages de champs explicites

L'exemple de modèle suivant contient chacun de ces paramètres :

```
{
   "index_patterns":[
      "index-*"
   ],
   "order": 0,
   "settings": {
      "index": {
         "number_of_shards": 3,
         "number_of_replicas": 1,
         "refresh_interval": "60s"
      }
   },
   "mappings": {
      "dynamic": false,
      "properties": {
         "field_name1": {
            "type": "keyword"
         }
      }
   }
}
```

Même s'ils changent rarement, il OpenSearch est plus simple de gérer la définition centralisée des paramètres et des mappages que la mise à jour de plusieurs clients en amont.

Gérer les index avec Index State Management

Si vous gérez des journaux ou des données de séries temporelles, nous vous recommandons d'utiliser Index State Management (ISM). ISM vous permet d'automatiser les tâches régulières de gestion du cycle de vie des index. Avec ISM, vous pouvez créer des stratégies qui déclenchent les renouvellements d'alias d'index, la prise d'instantanés d'index, le déplacement d'index entre les niveaux de stockage et la suppression d'anciens index. Vous pouvez même utiliser l'opération de renouvellement d'ISM comme stratégie alternative de gestion du cycle de vie des données afin d'éviter l'asymétrie des partitions.

Tout d'abord, configurez une stratégie ISM. Pour obtenir un exemple, consultez the section called “Exemples de politiques”. Ensuite, attachez la stratégie à un ou plusieurs index. Si vous incluez un champ de modèle ISM dans la politique, OpenSearch Service applique automatiquement la politique à tout index correspondant au modèle spécifié.

Supprimez les index inutilisés

Examinez régulièrement les index de votre cluster et identifiez ceux qui ne sont pas utilisés. Prenez un instantané de ces index pour qu'ils soient stockés dans S3, puis supprimez-les. Lorsque vous supprimez des index inutilisés, vous réduisez le nombre de partitions et vous permettez une distribution du stockage et une utilisation des ressources plus équilibrées entre les nœuds. Même lorsqu'ils sont inutilisés, les index consomment certaines ressources pendant les activités internes de maintenance des index.

Plutôt que de supprimer manuellement les index inutilisés, vous pouvez utiliser ISM pour prendre automatiquement un instantané et supprimer les index après un certain temps.

Utiliser plusieurs domaines pour une haute disponibilité

Pour obtenir une haute disponibilité supérieure à 99,9 % sur plusieurs régions, envisagez d'utiliser deux domaines. Pour des jeux de données de petite taille ou qui ne sont pas modifiés fréquemment, vous pouvez configurer une réplication inter-clusters pour obtenir un modèle actif-passif. Dans ce modèle, seul le domaine principal fait l'objet d'une écriture, mais l'un ou l'autre des domaines peut être lu. Pour des jeux de données plus importants et des données qui sont modifiées fréquemment, configurez une double diffusion dans votre pipeline d'ingestion afin que les données soient écrites indépendamment dans les deux domaines dans un modèle actif-actif.

Concevez vos applications en amont et en aval en tenant compte du basculement. Assurez-vous de tester le processus de basculement en même temps que les autres processus de reprise après sinistre.

Performances

Les bonnes pratiques suivantes s'appliquent au réglage de vos domaines pour des performances optimales.

Optimiser la taille et la compression des demandes groupées

La taille des groupes dépend de vos données, des analyses et de la configuration du cluster, mais un bon point de départ est de 3-5 Mio par demande groupée.

Envoyez des demandes et recevez des réponses de la part de vos OpenSearch domaines en utilisant la compression gzip pour réduire la taille de la charge utile des demandes et des réponses. Vous pouvez utiliser la compression gzip avec le client OpenSearch Python ou en incluant les en-têtes suivants du côté client :

- `'Accept-Encoding': 'gzip'`
- `'Content-Encoding': 'gzip'`

Pour optimiser la taille de vos demandes groupées, commencez par une taille de 3 Mio. Augmentez ensuite lentement la taille de la demande jusqu'à ce que les performances d'indexation cessent de s'améliorer.

> **Note**
>
> Pour activer la compression gzip dans les domaines exécutant Elasticsearch version 6.x, vous devez définir `http_compression.enabled` au niveau du cluster. Ce paramètre est vrai par défaut dans les versions 7.x d'Elasticsearch et dans toutes les versions de. OpenSearch

Réduire la taille des réponses aux demandes groupées

Pour réduire la taille des OpenSearch réponses, excluez les champs inutiles à l'aide du `filter_path` paramètre. Assurez-vous de ne pas filtrer les champs qui sont nécessaires pour

identifier ou relancer les demandes ayant échoué. Pour plus d'informations et d'exemples, consultez the section called "Réduction de la taille des réponses".

Régler les intervalles d'actualisation

OpenSearch les index ont finalement une cohérence de lecture. Une opération d'actualisation rend toutes les mises à jour effectuées sur un index disponibles pour la recherche. L'intervalle d'actualisation par défaut est d'une seconde, ce qui OpenSearch signifie qu'une actualisation est effectuée toutes les secondes pendant l'écriture d'un index.

Moins vous actualisez un index (intervalle d'actualisation plus élevé), meilleures sont les performances globales de l'indexation. L'augmentation de l'intervalle d'actualisation entraîne un délai plus long entre la mise à jour de l'index et le moment où les nouvelles données sont disponibles pour la recherche. Définissez un intervalle d'actualisation aussi haut que possible pour améliorer les performances globales.

Nous vous recommandons de définir le paramètre `refresh_interval` pour tous vos index sur 30 secondes ou plus.

Activer Auto-Tune

Auto-Tune utilise les indicateurs de performance et d'utilisation de votre OpenSearch cluster pour suggérer des modifications de la taille des files d'attente, de la taille du cache et des paramètres de machine virtuelle Java (JVM) sur vos nœuds. Ces modifications facultatives améliorent la vitesse et la stabilité du cluster. Vous pouvez revenir aux paramètres de OpenSearch service par défaut à tout moment. Auto-Tune est activé par défaut sur les nouveaux domaines, sauf si vous le désactivez volontairement.

Nous vous recommandons d'activer Auto-Tune sur tous les domaines et de définir une fenêtre de maintenance récurrente ou de revoir périodiquement ses recommandations.

Sécurité

Les bonnes pratiques suivantes s'appliquent à la sécurisation de vos domaines.

Activer le contrôle précis des accès

Le contrôle d'accès détaillé vous permet de contrôler qui peut accéder à certaines données au sein d'un domaine de OpenSearch service. Par rapport au contrôle d'accès généralisé, le contrôle précis des accès attribue à chaque cluster, index, document et champ sa propre stratégie d'accès. Les

critères d'accès peuvent être basés sur un certain nombre de facteurs, notamment le rôle de la personne qui demande l'accès et l'action qu'elle compte effectuer sur les données. Par exemple, vous pouvez accorder à un utilisateur l'accès à l'écriture dans un index, et à un autre utilisateur l'accès uniquement pour lire les données de l'index sans y apporter de modifications.

Le contrôle précis des accès permet aux données ayant des exigences d'accès différentes d'exister dans le même espace de stockage sans rencontrer de problèmes de sécurité ou de conformité.

Nous vous recommandons d'activer le contrôle précis des accès sur vos domaines.

Déployer des domaines dans un VPC

Le fait de placer votre domaine de OpenSearch service dans un cloud privé virtuel (VPC) permet de sécuriser les communications entre le OpenSearch service et les autres services au sein du VPC, sans avoir besoin d'une passerelle Internet, d'un périphérique NAT ou d'une connexion VPN. Tout le trafic reste sécurisé dans le AWS cloud. En raison de leur isolement logique, les domaines résidant au sein d'un VPC possèdent une couche de sécurité supplémentaire par rapport aux domaines qui utilisent des points de terminaison publics.

Nous vous recommandons de créer vos domaines au sein d'un VPC.

Appliquer une stratégie d'accès restrictive

Même si votre domaine est déployé au sein d'un VPC, la meilleure pratique consiste à mettre en œuvre la sécurité par couches. Assurez-vous de vérifier la configuration de vos stratégies d'accès actuelles.

Appliquez une politique d'accès restrictive basée sur les ressources à vos domaines et suivez le principe du moindre privilège lorsque vous accordez l'accès à l'API de configuration et aux opérations de l' OpenSearch API. En règle générale, évitez d'utiliser le principal utilisateur anonyme `"Principal": {"AWS": "*" }` dans vos stratégies d'accès.

Cependant, dans certaines situations, il est acceptable d'utiliser une stratégie d'accès ouverte, par exemple lorsque vous activez le contrôle précis des accès. Une stratégie d'accès ouverte peut vous permettre d'accéder au domaine dans les cas où la signature de la demande est difficile ou impossible, par exemple à partir de certains clients et outils.

Activer le chiffrement au repos

OpenSearch Les domaines de service permettent de chiffrer les données au repos afin d'empêcher tout accès non autorisé à vos données. Le chiffrement au repos utilise AWS Key Management

Service (AWS KMS) pour stocker et gérer vos clés de chiffrement, et l'algorithme Advanced Encryption Standard avec des clés de 256 bits (AES-256) pour effectuer le chiffrement.

Si votre domaine stocke des données sensibles, activez le chiffrement des données au repos.

Activer node-to-node le chiffrement

Node-to-node le chiffrement fournit une couche de sécurité supplémentaire en plus des fonctionnalités de sécurité par défaut du OpenSearch Service. Il implémente le protocole TLS (Transport Layer Security) pour toutes les communications entre les nœuds qui y sont provisionnés. OpenSearch Node-to-nodechiffrement : toutes les données envoyées à votre domaine de OpenSearch service via HTTPS restent cryptées en transit pendant leur distribution et leur réplication entre les nœuds.

Si votre domaine stocke des données sensibles, activez node-to-node le chiffrement.

Moniteur avec AWS Security Hub

Surveillez votre utilisation du OpenSearch Service en ce qui concerne les meilleures pratiques de sécurité en utilisant AWS Security Hub. Security Hub utilise des contrôles de sécurité pour évaluer les configurations des ressources et les normes de sécurité afin de vous aider à respecter divers cadres de conformité. Pour plus d'informations sur l'utilisation de Security Hub pour évaluer les ressources OpenSearch du service, consultez la section Amazon OpenSearch Service Contrôles du Guide de AWS Security Hub l'utilisateur.

Optimisation des coûts

Les meilleures pratiques suivantes s'appliquent à l'optimisation et à la réduction de vos coûts OpenSearch de service.

Utiliser les types d'instance de dernière génération

OpenSearch Le service adopte constamment de nouveaux types d' EC2 instances Amazon qui offrent de meilleures performances à moindre coût. Nous vous recommandons de toujours utiliser les instances de dernière génération.

Évitez d'utiliser des instances T2 ou `t3.small` pour les domaines de production, car elles peuvent devenir instables sous une charge élevée soutenue. Les instances `r6g.large` constituent une bonne option pour les petites charges de travail de production (à la fois en tant que nœuds de données et en tant que nœuds principaux dédiés).

Utilisation des derniers volumes Amazon EBS gp3

OpenSearch les nœuds de données nécessitent une faible latence et un stockage à haut débit pour permettre une indexation et des requêtes rapides. Grâce aux volumes Amazon EBS gp3, vous obtenez des performances de base supérieures (IOPS et débit) à un coût inférieur de 9,6 % à celui du type de volume Amazon EBS gp2 proposé précédemment. Vous pouvez fournir des IOPS et des débits supplémentaires indépendamment de la taille du volume à l'aide de gp3. Ces volumes sont également plus stables que ceux de la génération précédente, car ils n'utilisent pas de crédits en rafale. Le type de volume gp3 double également les limites de taille de per-data-node volume du type de volume gp2. Grâce à ces volumes plus importants, vous pouvez réduire le coût des données passives en augmentant la quantité de stockage par nœud de données.

Utilisation UltraWarm et stockage à froid pour les données des journaux de séries chronologiques

Si vous OpenSearch les utilisez pour l'analyse des journaux, déplacez vos données vers un UltraWarm stockage à froid afin de réduire les coûts. Utilisez Index State Management (ISM) pour migrer les données entre les niveaux de stockage et gérer la conservation des données.

UltraWarmconstitue un moyen rentable de stocker de grandes quantités de données en lecture seule dans OpenSearch Service. UltraWarm utilise Amazon S3 pour le stockage, ce qui signifie que les données sont immuables et qu'une seule copie est nécessaire. Vous ne payez que pour un stockage équivalent à la taille des partitions principales de vos index. Les latences des UltraWarm requêtes augmentent en fonction de la quantité de données S3 nécessaires pour traiter la requête. Une fois les données mises en cache sur les nœuds, les requêtes vers les UltraWarm index fonctionnent de la même manière que les requêtes vers les index actifs.

Le stockage cold est également basé sur S3. Lorsque vous devez interroger des données confidentielles, vous pouvez les associer de manière sélective à des UltraWarm nœuds existants. Le coût du stockage géré pour les données froides est le même que pour le stockage à froid UltraWarm, mais les objets stockés à froid ne consomment pas les ressources des UltraWarm nœuds. Par conséquent, le stockage à froid fournit une capacité de stockage significative sans impact sur la taille ou le nombre de UltraWarm nœuds.

UltraWarm devient rentable lorsque vous avez environ 2,5 TiB de données à migrer depuis le stockage à chaud. Surveillez votre taux de remplissage et prévoyez de déplacer les index UltraWarm avant d'atteindre ce volume de données.

Examiner les recommandations pour les instances réservées

Envisagez d'acheter des instances réservées (RIs) une fois que vous aurez une bonne base de référence sur vos performances et votre consommation de calcul. Les remises commencent aux alentours de 30 % pour les réservations d'un an sans versement initial et peuvent augmenter jusqu'à 50 % pour les engagements initiaux de trois ans.

Après avoir observé un fonctionnement stable pendant au moins 14 jours, consultez la section Accès aux recommandations de réservation dans le guide de AWS Cost Management l'utilisateur. L'en-tête Amazon OpenSearch Service affiche des recommandations d'achat spécifiques au RI et des économies prévues.

Dimensionnement des domaines Amazon OpenSearch Service

Il n'existe pas de méthode parfaite pour dimensionner les domaines Amazon OpenSearch Service. Cependant, en commençant par comprendre vos besoins en stockage, le service et OpenSearch lui-même, vous pouvez faire une première estimation éclairée de vos besoins en matériel. Cette estimation peut servir de point de départ utile pour l'aspect le plus critique du dimensionnement des domaines : tester ceux-ci avec des charges de travail représentatives et surveiller leurs performances.

Rubriques

- Calcul des exigences de stockage
- Choix du nombre de partitions
- Choix des types d'instances et test

Calcul des exigences de stockage

La plupart des OpenSearch charges de travail se répartissent dans l'une des deux grandes catégories suivantes :

- Index à longue durée de vie : vous écrivez du code qui traite les données dans un ou plusieurs OpenSearch index, puis vous mettez à jour ces index périodiquement à mesure que les données source changent. Parmi les exemples courants, figurent les recherches de site web, de documents et de commerce en ligne.
- Index glissants : les données arrivent en continu dans un jeu d'index temporaires, avec une période d'indexation et une fenêtre de conservation (par exemple, un jeu d'index quotidiens qui sont

conservés pendant deux semaines). Parmi les exemples courants, figurent l'analyse des journaux, le traitement de séries chronologiques et l'analyse des flux de clics.

Pour les charges de travail à index de longue durée, vous pouvez examiner les données source sur le disque et facilement déterminer l'espace de stockage qu'elles consomment. Si les données proviennent de plusieurs sources, il vous suffit de rassembler ces sources.

Pour les index glissants, vous pouvez multiplier la quantité de données générées au cours d'une période de temps représentative par la période de conservation. Par exemple, si vous générez 200 Mio de données de journal par heure, cela représente 4,7 Gio par jour, soit 66 Gio de données à un instant donné si vous disposez d'une période de conservation de deux semaines.

Cependant, la taille de vos données source n'est qu'un aspect de vos exigences de stockage. Vous devez également tenir compte des éléments suivants :

- Nombre de répliques : chaque réplique est une copie complète de la partition principale, la taille du magasin de l'index indique la taille prise par la partition principale et la partition de réplique. Par défaut, chaque OpenSearch index possède une réplique. Nous vous recommandons d'en avoir au moins un pour empêcher toute perte de données. Les réplicas améliorent également les performances de recherche. Vous souhaiterez donc en avoir plus en cas de charge de travail à lecture intensive. Utilisez `PUT /my-index/_settings` pour mettre à jour le paramètre `number_of_replicas` de votre index.
- OpenSearch surcharge d'indexation : la taille sur disque d'un index varie. La taille totale des données source et de l'index correspond souvent à 110 % de la source, l'index pouvant atteindre 10 % des données source. Après avoir indexé vos données, vous pouvez utiliser l'API `_cat/indices?v` et la valeur `pri.store.size` pour calculer la surcharge exacte. `_cat/allocation?v` offre également un résumé utile.
- Espace réservé par le système d'exploitation : par défaut, Linux réserve 5 % du système de fichiers à l'utilisateur `root` pour les processus critiques, la récupération du système et la protection des données contre les problèmes de fragmentation de disque.
- OpenSearch Frais de service : le OpenSearch service réserve 20 % de l'espace de stockage de chaque instance (jusqu'à 20 GiB) aux fusions de segments, aux journaux et à d'autres opérations internes.

  En raison de ce plafond de 20 Gio, la quantité totale d'espace réservé peut varier considérablement en fonction du nombre d'instances dans votre domaine. Par exemple, un domaine peut comporter trois instances `m6g.xlarge.search`, chacune dotée de 500 Gio

d'espace de stockage, pour un total de 1,46 Tio. Dans ce cas, l'espace réservé total est seulement de 60 Gio. Un autre domaine peut comporter 10 instances `m3.medium.search`, chacune dotée de 100 Gio d'espace de stockage, pour un total de 0,98 Tio. Dans ce cas, l'espace réservé total est de 200 Gio, même si le premier domaine est 50 % plus grand.

Dans la formule suivante, nous appliquons une estimation du « pire scénario » pour les frais généraux. Cette estimation inclut de l'espace libre supplémentaire afin de minimiser l'impact des défaillances des nœuds et des pannes de zone de disponibilité.

En résumé, si vous disposez de 66 Gio de données à un instant donné et que vous voulez un réplica, votre espace de stockage minimal requis se rapproche de 66 * 2 * 1,1/0,95/0,8 = 191 Gio. Vous pouvez généraliser ce calcul comme suit :

Données source* (1 + nombre de répliques) * (1 + surcharge d'indexation)/(1 - espace réservé Linux)/(1 - surcharge de OpenSearch service) = espace de stockage minimal requis

Vous pouvez également utiliser cette version simplifiée :

Données source * (1 + nombre de réplicas) * 1,45 = Exigences minimales de stockage

L'insuffisance d'espace de stockage est l'une des causes les plus courantes d'instabilité des clusters. Vous devez donc vérifier les chiffres lorsque vous choisissez les types d'instances, le nombre d'instances et les volumes de stockage.

D'autres considérations en matière de stockage existent :

- Si vos exigences de stockage minimal dépassent 1 Po, consultez the section called “Mise à l'échelle d'une capacité de plusieurs péta-octets”.
- Si vous avez des index glissants et que vous voulez utiliser une architecture hot-warm, consultez the section called “UltraWarm rangement”.

Choix du nombre de partitions

Une fois que vous avez déterminé vos exigences de stockage, vous pouvez examiner votre stratégie d'indexation. Par défaut, dans OpenSearch Service, chaque index est divisé en cinq partitions principales et une réplique (10 partitions au total). Ce comportement est différent de celui de l'open source OpenSearch, qui utilise par défaut une partition principale et une partition de réplique. Comme vous ne pouvez pas modifier aisément le nombre de partitions principales pour un index existant, vous devez décider du nombre de partitions avant d'indexer votre premier document.

L'objectif général du choix d'un nombre de partitions est de répartir un index de manière uniforme sur tous les nœuds de données du cluster. Toutefois, ces partitions ne doivent pas être trop grandes, ni trop nombreuses. En règle général, la taille des partitions doit être comprise entre 10 et 30 Gio pour les charges de travail où la latence de recherche est un objectif de performance clé, et entre 30 et 50 Gio pour les charges de travail lourdes en écriture, telles que l'analyse des journaux.

Les partitions volumineuses peuvent compliquer le rétablissement après une panne, mais comme chaque partition utilise une certaine quantité de processeur et de mémoire, le fait d'avoir trop de petites partitions peut entraîner des problèmes de performances et des erreurs de mémoire insuffisante. OpenSearch En d'autres termes, les partitions doivent être suffisamment petites pour que l'instance de OpenSearch service sous-jacente puisse les gérer, mais pas au point de surcharger inutilement le matériel.

Par exemple, supposons que vous disposez de 66 Gio de données. Vous ne prévoyez pas que ce nombre augmente au fil du temps, et vous voulez maintenir vos partitions autour de 30 Gio chacune. Le nombre de partitions doit donc être d'environ 66 * 1,1/30 = 3. Vous pouvez généraliser ce calcul comme suit :

(Données source + marge de croissance) * (1 + surcharge d'indexation) / taille de partition souhaitée = Nombre approximatif de partitions principales

Cette équation permet de compenser la croissance du volume de données au fil du temps. Si vous vous attendez à ce que ces 66 Gio de données quadruplent au cours de l'année suivante, le nombre approximatif de partitions est de (66 + 198) * 1,1/30 = 10. Gardez toutefois à l'esprit que vous ne disposez pas encore de ces 198 Gio de données supplémentaires. Vérifiez que cette préparation pour l'avenir ne crée pas de partitions inutilement petites qui consomment actuellement d'énormes quantités d'UC et de mémoire. Dans ce cas, 66 * 1,1/10 partitions = 7,26 Gio par partition, ce qui consomme des ressources supplémentaires et se situe au-dessous de la plage de tailles recommandées. Vous pourriez envisager l' middle-of-the-roadapproche de six partitions, ce qui vous laisse avec des partitions de 12 Go aujourd'hui et des partitions de 48 Go dans le futur. Là encore, vous préférerez peut-être commencer avec trois partitions et réindexer vos données lorsque les partitions dépasseront 50 Gio.

Un problème beaucoup moins fréquent consiste à limiter le nombre de partitions par nœud. Si vous dimensionnez vos partitions de manière appropriée, vous manquez généralement d'espace disque longtemps avant d'atteindre cette limite. Par exemple, une instance `m6g.large.search` a une taille de disque maximale de 512 Go. Si vous restez en dessous de 80 % d'utilisation du disque et que vous dimensionnez vos partitions à 20 Go, il peut accueillir environ 20 partitions. Elasticsearch 7. x et versions ultérieures, ainsi que toutes les versions antérieures OpenSearch à 2.15, ont

une limite de 1 000 partitions par nœud. Pour régler le nombre maximal de partitions par nœud, configurez le paramètre `cluster.max_shards_per_node`. À partir de la version OpenSearch 2.17, OpenSearch Service prend en charge 1 000 partitions pour 16 Go de mémoire de segment de mémoire JVM, jusqu'à un maximum de 4 000 partitions par nœud. Pour obtenir un exemple, consultez Paramètres du cluster. Pour plus d'informations sur le nombre de partitions, consultezthe section called “Quotas de nombre d'unités”.

Le dimensionnement approprié des partitions vous permet de rester presque toujours en dessous de cette limite, mais vous pouvez également prendre en compte le nombre de partitions pour chaque Go de segments de mémoire Java. Sur un nœud donné, ne dépassez pas 25 partitions par Gio de segments de mémoire Java. Par exemple, une instance `m5.large.search` présente un segment de mémoire de 4 Gio, de sorte que chaque nœud ne devrait pas avoir plus de 100 partitions. Avec un tel nombre de partitions, chacune d'elles a une taille d'environ 5 Go, ce qui est bien inférieur à notre recommandation.

Choix des types d'instances et test

Une fois que vous avez calculé vos exigences de stockage et choisi le nombre de partitions dont vous avez besoin, vous pouvez commencer à prendre des décisions en termes de matériel. Les exigences matérielles varient considérablement selon la charge de travail, mais nous pouvons quand même vous fournir quelques recommandations de base.

En général, les limites de stockage pour chaque type d'instance sont mappées à la quantité d'UC et de mémoire dont vous pouvez avoir besoin pour des charges de travail légères. Par exemple, une instance `m6g.large.search` possède une taille de volume EBS maximale de 512 Gio, 2 cœurs vCPU et 8 Gio de mémoire. Si votre cluster comporte de nombreuses partitions, effectue des regroupements de taxe et met à jour des documents fréquemment, ou traite un grand nombre de requêtes, ces ressources peuvent être insuffisantes pour vos besoins. Si votre cluster se trouve dans l'une de ces catégories, essayez de commencer avec une configuration plus proche de 2 cœurs vCPU et de 8 Gio de mémoire tous les 100 Gio de votre espace de stockage requis.

> **Tip**
>
> Pour obtenir un résumé des ressources matérielles allouées à chaque type d'instance, consultez la tarification d'Amazon OpenSearch Service.

Cependant, même ces ressources peuvent être insuffisantes. Certains OpenSearch utilisateurs signalent qu'ils ont besoin de plusieurs fois ces ressources pour répondre à leurs besoins. Pour

trouver le matériel adéquat pour votre charge de travail, vous devez réaliser une estimation initiale informée, effectuer des tests avec des charges de travail représentatives, ajuster et tester à nouveau :

Étape 1 : Effectuer une estimation initiale

Pour commencer, nous recommandons un minimum de trois nœuds afin d'éviter des OpenSearch problèmes potentiels, tels qu'un état de division du cerveau (lorsqu'une interruption de communication entraîne la création d'un cluster de deux nœuds principaux). Si vous disposez de trois nœuds principaux dédiés, nous recommandons au moins deux nœuds de données pour la réplication.

Étape 2 : Calculer les besoins en stockage par nœud

Si votre espace de stockage requis est de 184 Gio et le nombre minimal de nœuds recommandé de trois, utilisez l'équation 184/3 = 61 Gio pour trouver la quantité de stockage dont chaque nœud a besoin. Dans cet exemple, vous pouvez sélectionner trois instances `m6g.large.search`, ou chacune utilise un volume de stockage EBS de 90 Gio pour vous permettre de disposer d'un filet de sécurité et d'une marge de croissance au fil du temps. Cette configuration fournit 6 cœurs vCPU et 24 Gio de mémoire. Elle est donc adaptée à des charges de travail plus légères.

Pour un exemple plus significatif, envisagez un espace de stockage requis de 14 Tio et une charge de travail importante. Dans ce cas, vous pouvez choisir de commencer le test avec 2 * 144 = 288 cœurs vCPU et 8 * 144 = 1152 Gio de mémoire. Ces numéros fonctionnent sur environ 18 instances `i3.4xlarge.search`. Si vous n'avez pas besoin d'un stockage rapide en local, vous pouvez également tester 18 instances `r6g.4xlarge.search`, chacune utilisant un volume de stockage EBS de 1 Tio.

Si votre cluster inclut des centaines de téraoctets de données, consultez the section called “Mise à l'échelle d'une capacité de plusieurs péta-octets”.

Étape 3 : Effectuer des tests représentatifs

Après avoir configuré le cluster, vous pouvez ajouter vos index en utilisant le nombre de partitions que vous avez calculé précédemment, effectuer des tests clients représentatifs à l'aide d'un ensemble de données réaliste et surveiller CloudWatch les métriques pour voir comment le cluster gère la charge de travail.

Étape 4 : Réussir ou itérer

Si les performances répondent à vos besoins, que les tests réussissent et que CloudWatch les indicateurs sont normaux, le cluster est prêt à être utilisé. N'oubliez pas de définir CloudWatch des alarmes pour détecter une mauvaise utilisation des ressources.

Si les performances ne sont pas acceptables, que les tests échouent ou que les valeurs de `CPUUtilization` ou `JVMMemoryPressure` sont élevées, vous devez choisir un autre type d'instance (ou ajouter des instances) et continuer les tests. Au fur et à mesure que vous ajoutez des instances, la distribution des partitions est OpenSearch automatiquement rééquilibrée dans le cluster.

Étant donné qu'il est plus facile de mesurer la capacité excédentaire d'un cluster suralimenté que le déficit d'un cluster sous-alimenté, nous vous recommandons de commencer par un cluster plus large que ce dont vous pensez avoir besoin. Ensuite, testez et passez à un cluster efficace qui dispose des ressources supplémentaires pour assurer la stabilité des opérations pendant les périodes d'activité accrue.

Les clusters de production ou les clusters avec des états complexes tirent profit des nœuds principaux dédiés, qui améliorent les performances et la fiabilité du cluster.

Mise à l'échelle d'une capacité de plusieurs péta-octets dans Amazon Service OpenSearch

Les domaines Amazon OpenSearch Service offrent un stockage attaché allant jusqu'à 10 Po. Vous pouvez configurer un domaine avec 1 000 types d'`OR1.16xlarge.search`instances, chacun avec 36 To de stockage. En raison de la différence de l'échelle, des recommandations pour les domaines de cette taille diffèrent de nos recommandations générales. Cette section présente les éléments à prendre en compte pour la création de domaines, les coûts, le stockage et la taille de la partition.

Bien que cette section fasse fréquemment référence aux types d'`i3.16xlarge.search`instances, vous pouvez utiliser plusieurs autres types d'instances pour atteindre 10 Po de stockage de domaine total.

Création de domaines

> Les domaines de cette taille dépassent la limite par défaut de 80 instances par domaine. Pour demander une augmentation de limite de service jusqu'à 1 000 instances par domaine, ouvrez une demande auprès du Centre AWS Support.

Tarification

Avant de créer un domaine de cette taille, consultez la page Tarification d'Amazon OpenSearch Service afin de vous assurer que les coûts associés correspondent à vos attentes. Examinez the section called “UltraWarm rangement” pour voir si une architecture chaude correspond à votre cas d'utilisation.

Stockage

Les types d'`i3`instances sont conçus pour fournir un stockage rapide et local non volatile express (NVMe) en mémoire non volatile. Dans la mesure où ce stockage local tend à offrir des avantages en termes de performances par rapport à Amazon Elastic Block Store, les volumes EBS ne sont pas une option lorsque vous sélectionnez ces types d'instances dans OpenSearch Service. Si vous préférez le stockage EBS, utilisez un autre type d'instance, par exemple `r6.12xlarge.search`.

Nombre et taille des partitions

Une OpenSearch recommandation courante consiste à ne pas dépasser 50 Go par partition. Étant donné le nombre de partitions nécessaires pour accueillir les grands domaines et les ressources disponibles pour les instances `i3.16xlarge.search`, nous vous recommandons d'utiliser une taille de partition de 100 Go.

En résumé, si vous disposez de 450 Go de données sources et que vous souhaitez avoir une copie, votre espace de stockage minimum est plus près de 450 To * 2 * 1,1/0,95 = 1,04 Go. Pour une explication de ce calcul, consultez the section called “Calcul des exigences de stockage”. Bien que 1.04 Po/15 To = 70 instances, vous pouvez sélectionner un minimum de 90 instances `i3.16xlarge.search` pour vous donner un filet de sécurité de stockage, gérer les défaillances de nœuds et tenir compte de la variation de la quantité de données au fil du temps. Chaque instance ajoute 20 Gio à votre espace de stockage minimal requis. Pour les disques de cette taille, ces 20 Gio sont presque négligeables.

Il est difficile de contrôler le nombre de partitions. OpenSearch les utilisateurs font souvent la rotation des index sur une base quotidienne et conservent les données pendant une semaine ou deux. Dans ce cas, il peut être utile de faire la distinction entre les partitions « actives » et « inactives ». Les partitions actives sont celles qui sont activement utilisées pour l'écriture ou la lecture. Les partitions inactives peuvent prendre en charge quelques demandes de lecture, mais sont principalement inactives. En général, le nombre de partitions actives doit rester inférieur à quelques milliers. À mesure que le nombre de partitions actives s'approche de 10 000, des risques peuvent peser sur les performances et la stabilité.

Pour calculer le nombre de partitions principales, utilisez cette formule : 450 000 Go * 1,1/100 Go par partition = 4 950 partitions. Si vous multipliez ce chiffre par deux pour prendre en compte le nombre de réplicas, vous obtenez 9 900 partitions, ce qui représente un problème majeur si toutes les partitions sont actives. Mais si vous procédez à la rotation des index et que seulement $1/7^{e}$ ou $1/14^{e}$ des partitions sont actives sur un jour donné (1 414 ou 707 partitions, respectivement), le cluster peut fonctionner correctement. Comme toujours, l'étape la plus importante du dimensionnement et de la configuration de votre domaine consiste à effectuer des tests clients représentatifs à l'aide d'un ensemble de données réalistes.

Nœuds de coordination dédiés dans Amazon OpenSearch Service

Les nœuds de coordination dédiés dans Amazon OpenSearch Service sont des nœuds spécialisés qui déchargent les tâches de coordination des nœuds de données. Ces tâches incluent la gestion des demandes de recherche et l'hébergement de OpenSearch tableaux de bord. En séparant ces fonctions, les nœuds coordinateurs dédiés réduisent la charge sur les nœuds de données, ce qui leur permet de se concentrer sur le stockage des données, l'indexation et les opérations de recherche. Cela améliore les performances globales du cluster et l'utilisation des ressources.

En outre, les nœuds de coordination dédiés contribuent à réduire le nombre d'adresses IP privées requises pour les configurations VPC, ce qui permet une gestion du réseau plus efficace. Cette configuration peut entraîner une amélioration allant jusqu'à 15 % du débit d'indexation et une amélioration des performances des requêtes de 20 %, en fonction des caractéristiques de la charge de travail.

Quand utiliser des nœuds de coordination dédiés

Les nœuds de coordination dédiés sont particulièrement utiles dans les scénarios suivants.

- Grands clusters : dans les environnements comportant un volume élevé de données ou des requêtes complexes, le transfert des tâches de coordination vers des nœuds dédiés peut améliorer les performances des clusters.
- Requêtes fréquentes — Les charges de travail impliquant des requêtes de recherche ou des agrégations fréquentes, en particulier celles comportant des histogrammes de dates complexes ou des agrégations multiples, bénéficient d'un traitement des requêtes plus rapide.
- Utilisation intensive des tableaux de bord — Les OpenSearch tableaux de bord peuvent être gourmands en ressources. Le fait de déléguer cette responsabilité à des nœuds de coordination dédiés réduit la pression sur les nœuds de données.

Architecture et comportement

Dans un OpenSearch cluster, les nœuds de coordination dédiés assument deux responsabilités principales.

- Gestion des demandes — Ces nœuds reçoivent les demandes de recherche entrantes et les transmettent aux nœuds de données appropriés, qui stockent les données pertinentes. Ils consolident ensuite les résultats de plusieurs nœuds de données en un seul ensemble de résultats global, qui est renvoyé au client.
- Hébergement de tableaux de bord — Les nœuds de coordination gèrent les OpenSearch tableaux de bord, ce qui soulage les nœuds de données de la charge supplémentaire liée à l'hébergement des OpenSearch tableaux de bord et à la gestion du trafic associé.

Dans les domaines VPC, les nœuds de coordination dédiés se voient attribuer des interfaces réseau élastiques (ENIs) plutôt que des nœuds de données. Cette disposition permet de réduire le nombre d'adresses IP privées requises VPCs, ce qui améliore l'efficacité du réseau. Généralement, les nœuds coordinateurs dédiés représentent environ 10 % du total des nœuds de données.

Exigences et limitations

Les nœuds de coordination dédiés sont soumis aux exigences et limites suivantes.

- Les nœuds de coordination dédiés sont pris en charge dans toutes les OpenSearch versions et dans les versions 6.8 à 7.10 d'Elasticsearch.
- Pour activer les nœuds coordinateurs dédiés, les nœuds maîtres dédiés doivent être activés dans votre domaine. Pour de plus amples informations, veuillez consulter the section called “Nœuds maîtres dédiés”.
- Le provisionnement de nœuds de coordination dédiés peut entraîner des coûts supplémentaires. Cependant, l'amélioration de l'efficacité des ressources et des performances justifient l'investissement, en particulier pour les clusters de grande taille ou complexes.

Provisionnement de nœuds de coordination dédiés

Procédez comme suit pour provisionner des nœuds de coordination dédiés dans un domaine existant. Assurez-vous que les nœuds maîtres dédiés sont activés sur votre domaine avant de configurer des nœuds coordinateurs.

console

Pour fournir des nœuds de coordination dédiés dans le AWS Management Console

1. Connectez-vous à la console Amazon OpenSearch Service à la https://console.aws.amazon.com/aos/maison.
2. Choisissez Domaines, puis sélectionnez le domaine que vous souhaitez modifier.
3. Dans la section Configuration du cluster, choisissez Modifier.
4. Choisissez Activer les nœuds de coordination dédiés.
5. Sélectionnez le type d'instance et le nombre de nœuds de coordination à approvisionner.
6. Sélectionnez Enregistrer les modifications. La mise à jour du domaine peut prendre plusieurs minutes.

AWS CLI

Pour approvisionner des nœuds de coordination dédiés à l'aide de AWS CLI, utilisez la update-domain-configcommande. L'exemple suivant fournit trois nœuds de `r6g.large.search` coordination dans un domaine.

```
aws opensearch update-domain-config \
  --domain-name my-opensearch-domain \
  --cluster-config
 InstanceCount=3,InstanceType=r6g.large.search,DedicatedCoordinatorCount=3,ZoneAwarenessEnabled
```

Cette commande active des nœuds de coordination dédiés, définit le type d'instance et le nombre de nœuds de coordination, et permet de connaître les zones pour une meilleure disponibilité.

Bonnes pratiques

Tenez compte des meilleures pratiques suivantes lorsque vous utilisez des nœuds de coordination dédiés.

- Utilisez des instances à usage général pour la plupart des cas d'utilisation. Ils proposent une approche équilibrée entre les coûts et les performances. Les instances optimisées pour la mémoire sont idéales pour les charges de travail qui nécessitent des ressources de mémoire importantes, telles que celles qui impliquent des agrégations complexes ou des recherches à grande échelle.
- Un bon point de départ consiste à fournir entre 5 % et 10 % de vos nœuds de données en tant que nœuds de coordination dédiés. Par exemple, si votre domaine possède 90 nœuds de données d'un

type d'instance particulier, envisagez de provisionner 5 à 9 nœuds de coordination du même type d'instance.

> **Note**
>
> La disponibilité des types d'instances varie selon les régions. Lorsque vous sélectionnez des types d'instance pour les nœuds de coordination, vérifiez que le type d'instance que vous avez choisi est disponible dans votre région cible. Vous pouvez vérifier la disponibilité du type d'instance dans la console de OpenSearch service lors de la création ou de la modification de votre domaine.

- Pour minimiser le risque d'un point de défaillance unique, configurez au moins deux nœuds de coordination dédiés. Cela garantit que votre cluster reste opérationnel même en cas de défaillance d'un nœud.
- Si vous utilisez la recherche interrégionale, fournissez des nœuds de coordination dédiés dans les domaines de destination. Les domaines source ne gèrent généralement pas les tâches de coordination.
- Pour les environnements à forte indexation, pensez à des instances optimisées pour le processeur qui correspondent à la taille d'instance de vos nœuds de données pour des performances optimales.
- Pour les charges de travail gourmandes en mémoire, utilisez un type d'instance légèrement plus grand pour vos nœuds de coordination dédiés afin de gérer les demandes de mémoire accrues.
- Suivez les statistiques `CoordinatorCPUUtilization` d' CloudWatch Amazon. S'il dépasse régulièrement 80 %, cela peut indiquer que vous avez besoin de nœuds de coordination plus grands ou supplémentaires pour gérer la charge.

Recommandations de nœuds par taille de cluster

Utilisez les directives suivantes comme point de départ pour le provisionnement de nœuds de coordination dédiés en fonction de la taille de votre cluster. Ajustez le nombre et le type de nœuds en fonction des caractéristiques de la charge de travail et des indicateurs de performance.

Taille du cluster	Nœuds de coordination recommandés	Type d’instance
Petit (jusqu'à 50 nœuds)	3 à 5 nœuds	Usage général
Moyen (50 à 100 nœuds)	5 à 9 nœuds	Optimisé pour la mémoire
Grand (plus de 100 nœuds)	10 à 15 nœuds	Optimisé pour la mémoire

Nœuds principaux dédiés dans Amazon OpenSearch Service

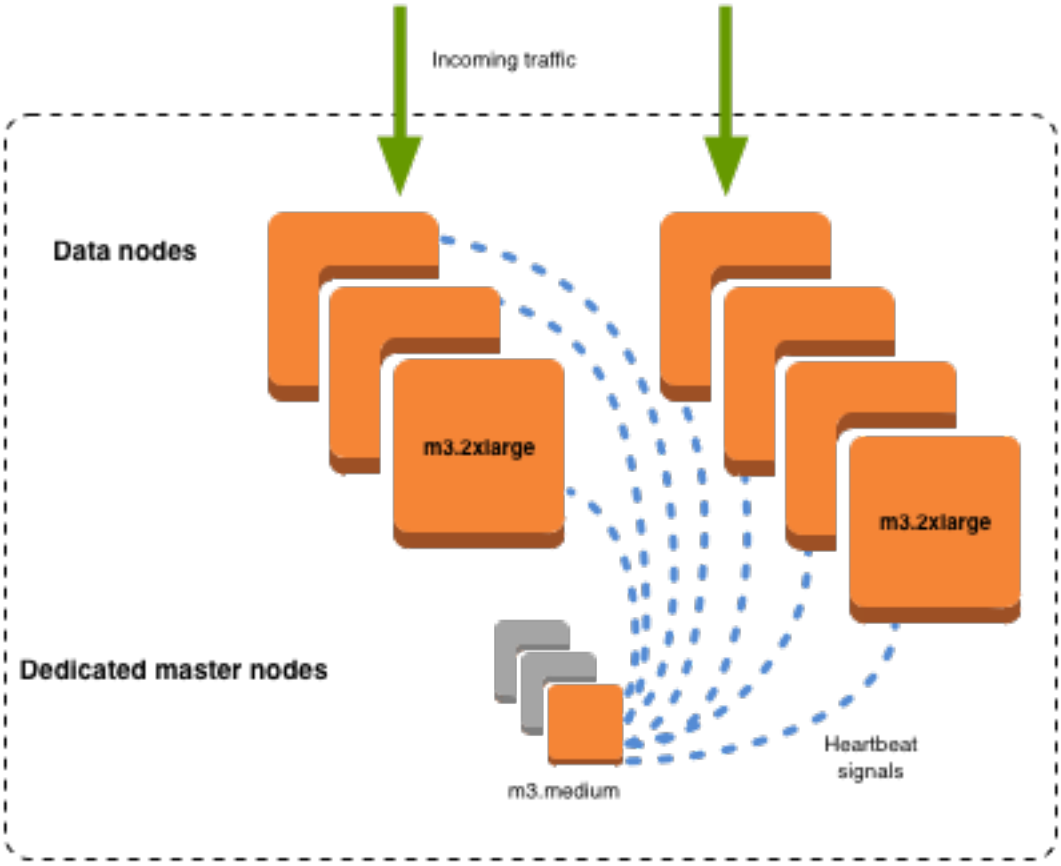
Amazon OpenSearch Service utilise des nœuds maîtres dédiés pour améliorer la stabilité du cluster. Un nœud principal dédié effectue des tâches de gestion du cluster, mais qui ne stocke pas de données ni ne répond aux demandes de chargement de données. Ce déchargement des tâches de gestion du cluster augmente la stabilité de votre domaine. Comme pour tous les autres types de nœuds, un tarif horaire s'applique à chaque nœud principal dédié.

Les nœuds principaux dédiés effectuent les tâches de gestion du cluster suivantes :

- Suivre tous les nœuds dans le cluster.
- Suivre le nombre d'index dans le cluster.
- Suivre le nombre de partitions appartenant à chaque index.
- Gérer les informations de routage pour les nœuds dans le cluster.
- Mettre à jour l'état du cluster après les changements d'état, tels que la création d'un index et l'ajout ou la suppression de nœuds dans le cluster.
- Répliquer les changements d'état du cluster sur tous les nœuds dans le cluster.
- Surveiller l'état de santé de tous les nœuds du cluster en envoyant des signaux de pulsation, des signaux périodiques qui contrôlent la disponibilité des nœuds de données dans le cluster.

L'illustration suivante montre un domaine OpenSearch de service avec 10 instances. Sept des instances sont des nœuds de données et trois sont des nœuds principaux dédiés. Un seul des nœuds principaux dédiés est actif. Les deux nœuds principaux dédiés gris patientent comme

sauvegarde en cas de défaillance du nœud principal dédié actif. Toutes les demandes de chargement de données sont correctement traitées par les sept nœuds de données et toutes les tâches de gestion du cluster sont déchargées sur le nœud principal dédié actif.



Choix du nombre de nœuds principaux dédiés

Nous vous recommandons d'utiliser Multi-AZ with Standby, qui ajoute trois nœuds maîtres dédiés à chaque domaine de OpenSearch service de production. Si vous déployez en mode multi-AZ sans mode veille ou mono-AZ, nous recommandons tout de même trois nœuds maîtres dédiés. Ne choisissez jamais un nombre pair de nœuds principaux dédiés. Tenez compte des éléments suivants lors du choix du nombre de nœuds principaux dédiés :

- Un nœud maître dédié est explicitement interdit par le OpenSearch Service car vous ne disposez d'aucune sauvegarde en cas de panne. Si vous essayez de créer un domaine avec un seul nœud principal dédié, vous recevrez une exception de validation.
- Si vous avez deux nœuds principaux dédiés, votre cluster ne dispose pas du quorum de nœuds nécessaire pour choisir un nouveau nœud principal en cas de défaillance.

Un quorum est le nombre de nœuds principaux dédiés / 2 + 1 (arrondi au nombre entier le plus proche). Dans ce cas, 2/2 + 1 = 2. Comme un nœud principal dédié a échoué et qu'il n'existe qu'une seule sauvegarde, le cluster n'a pas de quorum et ne peut pas choisir un nouveau nœud principal.

- Trois nœuds principaux dédiés, le nombre recommandé, fournissent deux nœuds de secours en cas de défaillance d'un nœud principal et le quorum nécessaire (2) pour choisir un nouveau nœud principal.
- Quatre nœuds principaux dédiés ne valent pas mieux que trois. Cela peut entraîner des problèmes si vous utilisez des zones de disponibilité multiples.
  - Si un nœud principal échoue, vous disposez du quorum (3) pour choisir un nouveau nœud principal. Si deux nœuds échouent, vous perdez ce quorum, comme vous le feriez avec trois nœuds principaux dédiés.
  - Dans une configuration à trois zones de disponibilité, deux d'entre elles AZs disposent d'un nœud maître dédié et une AZ en possède deux. Si cet AZ est perturbé, les deux autres AZs n'ont pas le quorum nécessaire (3) pour élire un nouveau maître.
- Cinq nœuds maîtres dédiés fonctionne aussi bien que trois et vous permet de perdre deux nœuds tout en conservant un quorum. Mais comme un seul nœud maître dédié est actif à un moment donné, cette configuration signifie que vous payez quatre nœuds inactifs. De nombreux utilisateurs jugent ce niveau de protection par basculement excessif.

Si un cluster possède un nombre pair de nœuds éligibles au master, OpenSearch et Elasticsearch version 7. x et plus tard, ignorez un nœud afin que la configuration de vote soit toujours un nombre impair. Dans ce cas, quatre nœuds principaux dédiés équivalent à trois (et deux à un).

> **Note**
>
> Si votre cluster ne dispose pas du quorum nécessaire pour choisir un nouveau nœud principal, les demandes en écriture et en lecture sur le cluster échouent. Ce comportement est différent de celui OpenSearch par défaut.

Choix des types d'instance pour les nœuds principaux dédiés

OpenSearch Quotas d'instance et de domaine de service

Bien que les nœuds maîtres dédiés ne traitent pas les demandes de recherche et de requête, leur taille est étroitement liée à la taille de l'instance et au nombre d'instances, d'index et de partitions qu'ils peuvent gérer. Pour les clusters de production, nous recommandons, au minimum, les types d'instances suivants pour les nœuds maîtres dédiés.

Ces recommandations reposent sur les charges de travail classiques et peuvent varier en fonction de vos besoins. Les clusters avec plusieurs partitions ou mappages de champ peuvent bénéficier de types d'instance plus large. Pour plus d'informations, consultez la section CloudWatch Alarmes recommandées pour Amazon OpenSearch Service afin de déterminer si vous devez utiliser un type d'instance plus important.

RAM	Support maximal de nœuds pour Elasticsearch et OpenSearch Service 1.x à 2.15	Support maximal pour Elasticsearch et OpenSearch Service 1.x à 2.15	Support maximal de nœuds pour le OpenSearch service 2.17 et versions ultérieures	Support de Max Shard pour OpenSearch Service 2.17 et versions ultérieures
2 Go	Ne s’applique pas	Ne s’applique pas	10	1 KM
4 Go	Ne s’applique pas	Ne s’applique pas	10	5 KM
8 Go	10	10 000	30	15 KM
16 Go	30	30 000	60	30 000
32 GO	75	40 000	120	60 000
64 Go	125	75 000	240	120 000
128 Go	200	75 000	480	240 000

RAM	Support maximal de nœuds pour Elasticsearch et OpenSearch Service 1.x à 2.15	Support maximal pour Elasticsearch et OpenSearch Service 1.x à 2.15	Support maximal de nœuds pour le OpenSearch service 2.17 et versions ultérieures	Support de Max Shard pour OpenSearch Service 2.17 et versions ultérieures
256 Go	Ne s'applique pas	Ne s'applique pas	1 002	500 000

CloudWatch Alarmes recommandées pour Amazon OpenSearch Service

CloudWatch les alarmes exécutent une action lorsqu'une CloudWatch métrique dépasse une valeur spécifiée pendant un certain temps. Par exemple, vous souhaiterez peut-être vous AWS envoyer un e-mail si l'état de santé de votre cluster `red` dure plus d'une minute. Cette section inclut certaines alarmes recommandées pour Amazon OpenSearch Service et explique comment y répondre.

Vous pouvez déployer automatiquement ces alarmes à l'aide de AWS CloudFormation. Pour un exemple de pile, consultez le GitHubréférentiel correspondant.

> **Note**
>
> Si vous déployez la CloudFormation pile, les `KMSKeyInaccessible` alarmes `KMSKeyError` et existeront dans un `Insufficient Data` état défini, car ces métriques n'apparaîtront que si un domaine rencontre un problème avec sa clé de chiffrement.

Pour plus d'informations sur la configuration des alarmes, consultez la section Création d' CloudWatchalarmes Amazon dans le guide de CloudWatch l'utilisateur Amazon.

alerte	Problème
La valeur maximale de `ClusterStatus.red` est >= 1 pendant 1 minute, 1 fois consécutive	Au moins une partition principale et ses réplicas ne sont pas alloués à un nœud. Consultez the section called “Statut de cluster rouge”.

alerte	Problème
La valeur maximale de `ClusterStatus.yellow` est >= 1 pendant 1 minute, 5 fois consécutives	Au moins une partition de réplica n'est pas allouée à un nœud. Consultez the section called “Statut de cluster jaune”.
La valeur minimale de `FreeStorageSpace` est <= 20480 pendant 1 minute, 1 fois consécutive	Un nœud de votre cluster est descendu à 20 Gio d'espace de stockage disponible. Consultez the section called “Manque d'espace de stockage disponible”. Cette valeur est en Mio. Par conséquent, au lieu de 20 480, nous vous recommandons de la définir sur 25 % de l'espace de stockage pour chaque nœud.
La valeur de `ClusterIndexWritesBlocked` est >= 1 pendant 5 minutes, 1 fois consécutive	Votre cluster bloque les demandes d'écriture. Consultez the section called “ClusterBlockException”.
La valeur minimale de `Nodes` est < x pendant 1 jour, 1 fois consécutive	x est le nombre de nœuds de votre cluster. Cette alarme indique qu'au moins un nœud de votre cluster a été inaccessible pendant 1 jour. Consultez the section called “Nœuds de cluster en échec”.

La valeur maximale de `AutomatedSnapshotFailure` est >= 1 pendant 1 minute, 1 fois consécutive	Un instantané automatique a échoué. Cette défaillance est souvent le résultat d'un état de santé de cluster rouge. Consultez the section called “Statut de cluster rouge”.
	
	Pour obtenir un récapitulatif de tous les instantanés automatiques et des informations sur les défaillances, essayez l'une des requêtes suivantes :
	
	`GET domain_endpoint /_snapshot/cs-automated/_all`
	`GET domain_endpoint /_snapshot/cs-automated-enc/_all`

alerte	Problème
`CPUUtilization` ou `WarmCPUUtilization` maximum est >= 80 % pendant 15 minutes, 3 fois consécutives	Une utilisation à 100 % de l'UC peut parfois se produire, mais une utilisation élevée et soutenue est problématique. Envisagez d'utiliser des types d'instances plus grands ou d'ajouter des instances.
La valeur maximale de `JVMMemoryPressure` est >= 95 % pendant 1 minutes, 3 fois consécutives	Le cluster peut rencontrer des erreurs de mémoire insuffisante si l'utilisation augmente. Envisagez de le dimensionner verticalement. OpenSearch Le service utilise la moitié de la RAM d'une instance pour le tas Java, jusqu'à une taille de segment de 32 GiB. Vous pouvez mettre à l'échelle des instances verticalement jusqu'à 64 Gio de RAM, après quoi vous pouvez effectuer une mise à l'échelle horizontale en ajoutant des instances.
La valeur maximale de `OldGenJVMMemoryPressure` est >= 80 % pendant 1 minutes, 3 fois consécutives	
La valeur maximale de `MasterCPUUtilization` est >= 50 % pendant 15 minutes, 3 fois consécutives	Envisagez d'utiliser des types d'instance plus grands pour vos nœuds principaux dédiés. En raison de leur rôle dans la stabilité du cluster et les déploiements bleu/vert, les nœuds principaux dédiés devraient avoir une utilisation de l'UC moyenne inférieure à celle des nœuds de données.
La valeur maximale de `MasterJVMMemoryPressure` est >= 95 % pendant 1 minutes, 3 fois consécutives	

alerte	Problème
La valeur maximale de `MasterOldGenJVMMemoryPressure` est >= 80 % pendant 1 minutes, 3 fois consécutives	
La valeur de `KMSKeyError` est >= 1 pendant 1 minute, 1 fois consécutive	La clé de AWS KMS chiffrement utilisée pour chiffrer les données au repos dans votre domaine est désactivée. Réactivez-la pour revenir à un fonctionnement normal. Pour de plus amples informations, veuillez consulter the section called “Chiffrement au repos”.
La valeur de `KMSKeyInaccessible` est >= 1 pendant 1 minute, 1 fois consécutive	La clé de AWS KMS chiffrement utilisée pour chiffrer les données au repos dans votre domaine a été supprimée ou a révoqué ses autorisations au OpenSearch Service. Vous ne pouvez pas récupérer des domaines qui sont à cet état. Cependant, si vous disposez d'un instantané manuel, vous pouvez l'utiliser pour migrer vers un nouveau domaine. Pour en savoir plus, veuillez consulter la section the section called “Chiffrement au repos”.
La valeur de `shards.active` est >= 30 000 pendant 1 minute, 1 fois consécutive	Le nombre total de partitions primaires et de partitions de réplica actives est supérieur à 30 000. La rotation des index est peut-être trop fréquente. Envisagez d'utiliser ISM pour supprimer les index une fois qu'ils atteignent un âge spécifique.
Alarmes *5xx* >= 10 % de `OpenSearchRequests`	Un ou plusieurs nœuds de données peuvent être surchargés ou les requêtes ne parviennent pas à être terminées pendant la période de délai d'inactivité. Pensez à passer à des types d'instances plus volumineuses ou à ajouter des nœuds supplémentaires au cluster. Confirmez que vous suivez les bonnes pratiques pour l'architecture de partitions et de clusters.

alerte	Problème
`MasterReachableFromNode` le maximum est inférieur à 1 pendant 5 minutes, 1 fois consécutive	Cette alarme indique que le nœud principal s'est arrêté ou est inaccessible. Ces défaillances sont généralement le résultat d'un problème de connectivité réseau ou d'un problème de AWS dépendance.
La valeur de `ThreadpoolWriteQueue` est >= 100 pendant 1 minute, 1 fois consécutive	Le cluster connaît une concurrence d'indexation élevée. Examinez et contrôlez les requêtes d'indexation ou augmentez les ressources du cluster.
La valeur de `ThreadpoolSearchQueue` est >= 500 pendant 1 minute, 1 fois consécutive	Le cluster connaît une concurrence d'indexation élevée. Pensez à dimensionner votre cluster. Vous pouvez également augmenter la taille de la file de recherche, mais son augmentation excessive peut entraîner des erreurs de mémoire insuffisante.
La valeur maximale de `ThreadpoolSearchQueue` est >= 5 000 pendant 1 minute, 1 fois consécutive	
L'augmentation de `ThreadpoolSearchRejected` SUM est >=1 {expression mathématique DIFF ()} pendant 1 minute, 1 fois consécutive	Ces alarmes vous informent des problèmes liés au domaine qui peuvent avoir un impact sur les performances et la stabilité.

alerte	Problème
L'augmentation de `Threadpool WriteRejected` SUM est >=1 {expression mathématique DIFF ()} pendant 1 minute, 1 fois consécutive	

> **Note**
>
> Si vous voulez simplement afficher les métriques, consultez the section called “Surveillance des métriques d'un cluster”.

Autres alarmes intéressantes

Pensez à configurer les alarmes suivantes en fonction des fonctionnalités OpenSearch du Service que vous utilisez régulièrement.

alerte	Problème
`WarmFreeStorageSpace` est >= 10 %	Vous avez atteint 10 % de votre espace de stockage chaud gratuit total. `WarmFreeStorageSpace` mesure la somme de votre espace de stockage chaud libre en MiB. UltraWarm utilise Amazon S3 plutôt que des disques attachés.
La valeur de `HotToWarmMigrationQueueSize` est >= 20 pendant 1 minute, 3 fois consécutives	Un grand nombre d'index passent simultanément du mode « hot » au UltraWarm « stockage ». Pensez à dimensionner votre cluster.

alerte	Problème
La valeur minimale de `HotToWarmMigrationSuccessLatency` est >= 1 jour, 1 fois consécutive	Configurez cette alarme pour être averti lorsque la valeur `HotToWarmMigrationSuccessCount` x latence dépasse 24 heures si vous essayez de déployer des index quotidiens.
La valeur maximale de `WarmJVMMemoryPressure` est >= 95 % pendant 1 minutes, 3 fois consécutives	Le cluster peut rencontrer des erreurs de mémoire insuffisante si l'utilisation augmente. Envisagez une mise à l'échelle verticale. OpenSearch Le service utilise la moitié de la RAM d'une instance pour le tas Java, jusqu'à une taille de segment de 32 GiB. Vous pouvez mettre à l'échelle des instances verticalement jusqu'à 64 Gio de RAM, après quoi vous pouvez effectuer une mise à l'échelle horizontale en ajoutant des instances.
La valeur maximale de `WarmOldGenJVMMemoryPressure` est >= 80 % pendant 1 minutes, 3 fois consécutives	
La valeur de `WarmToColdMigrationQueueSize` est >= 20 pendant 1 minute, 3 fois consécutives	Un grand nombre d'index passent simultanément de l'entrepôt UltraWarm frigorifique. Pensez à dimensionner votre cluster.

alerte	Problème
La valeur de `HotToWarm Migration FailureCount` est >= 1 pendant 1 minute, 1 fois consécutive	Les migrations peuvent échouer pendant les instantanés, les relocations de partition ou les fusions forcées. Les échecs lors des instantanés ou de la relocalisation de partitions sont généralement dus à des défaillan ces de nœud ou à des problèmes de connectivité S3. Le manque d'espace disque est généralement la cause sous-jacente des échecs de fusion forcée.
La valeur de `WarmToCol dMigratio nFailureCount` est >= 1 pendant 1 minute, 1 fois consécutive	Les migrations échouent généralement lorsque les tentatives de migration des métadonnées d'index vers un stockage frigorifique échouent. Des échecs peuvent également se produire lorsque l'état du cluster d'index à chaud est supprimé.
La valeur de `WarmToCol dMigratio nLatency` est >= 1 jour, 1 fois consécutive	Configurez cette alarme pour être averti lorsque la valeur `WarmToCol dMigrationSuccessCount` x latence dépasse 24 heures si vous essayez de déployer des index quotidiens.
La valeur de `AlertingDegraded` est >= 1 pendant 1 minute, 1 fois consécutive	L'index d'alerte est rouge, ou un ou plusieurs nœuds ne sont pas prévus.
La valeur de `ADPluginU nhealthy` est >= 1 pendant 1 minute, 1 fois consécutive	Le plug-in de détection d'anomalies ne fonctionne pas correctement, soit en raison de taux d'échecs élevés, soit parce que l'un des index utilisés est rouge.

alerte	Problème
La valeur de `AsynchronousSearchFailureRate` est >= 1 pendant 1 minute, 1 fois consécutive	Au moins une recherche asynchrone a échoué à la dernière minute, ce qui signifie probablement que le nœud du coordinateur a échoué. Le cycle de vie d'une requête de recherche asynchrone est géré uniquement sur le nœud du coordinateur. Par conséquent, si le coordinateur tombe en panne, la requête échoue.
La valeur de `AsynchronousSearchStoreHealth` est >= 1 pendant 1 minute, 1 fois consécutive	L'état du magasin de réponses de recherche asynchrone dans l'index persistant est rouge. Vous stockez peut-être des réponses asynchrones volumineuses, ce qui peut déstabiliser un cluster. Essayez de limiter vos réponses de recherche asynchrones à 10 Mo ou moins.
La valeur de `SQLUnhealthy` est >= 1 pendant 1 minute, 3 fois consécutives	Le plugin SQL renvoie 5 xx codes de réponse ou transmet une requête DSL non valide à OpenSearch. Résolvez les demandes que vos clients adressent au plugin.
La valeur de `LTRStatus.red` est >= 1 pendant 1 minute, 1 fois consécutive	Au moins un des index nécessaires à l'exécution du plug-in Learning to Rank contient des partitions principales manquantes et n'est pas fonctionnel.

Référence générale pour Amazon OpenSearch Service

Amazon OpenSearch Service prend en charge une variété d'instances, d'opérations, de plug-ins et d'autres ressources.

Rubriques

- Types d'instances pris en charge dans Amazon OpenSearch Service
- Fonctionnalités par version de moteur dans Amazon OpenSearch Service
- Plug-ins par version de moteur dans Amazon OpenSearch Service
- Opérations prises en charge dans Amazon OpenSearch Service
- Quotas Amazon OpenSearch Service
- Instances réservées dans Amazon OpenSearch Service
- Autres ressources prises en charge dans Amazon OpenSearch Service

Types d'instances pris en charge dans Amazon OpenSearch Service

Amazon OpenSearch Service prend en charge les types d'instances suivants. Certaines régions ne prennent pas en charge certains types d'instance. Pour en savoir plus sur la disponibilité, consultez les tarifs d'Amazon OpenSearch Service.

Pour plus d'informations sur le type d'instance qui convient à votre cas d'utilisation, consultez the section called “Dimensionnement des domaines”, the section called “Quotas de taille du volume EBS” et the section called “Quotas de réseau”.

Types d'instance de la génération actuelle

Pour de meilleures performances, nous vous recommandons d'utiliser les types d'instances suivants lorsque vous créez de nouveaux domaines OpenSearch de service.

Type d'instance	instances	Restrictions
i4i	i4i.large.search i4i.xlarge.search i4i.2xlarge.search i4i.4xlarge.search i4i.8xlarge.search i4i.12xlarge.search i4i.16xlarge.search i4i.24xlarge.search i4i.32xlarge.search	Les types d'instances i4i nécessitent Elasticsearch 5.1 ou version ultérieure ou toute autre version de OpenSearch, et ne prennent pas en charge le stockage de volume EBS.
i4g	i4g.large.search i4g.xlarge.search i4g.2xlarge.search	Les types d'instances i4g nécessitent Elasticsearch 7.9 ou version ultérieure ou toute autre version de OpenSearch, et ne prennent pas en charge les volumes de stockage EBS.

Type d’instance	instances	Restrictions
	i4g.4xlarge.search i4g.8xlarge.search i4g.16xlarge.search	

Type d'instance	instances	Restrictions
Graviton 3	C7G.Large.Rechercher C7G.xLarge.Rechercher C7G.2XLarge.Rechercher C7G.4XLarge.Rechercher C7G.8XLarge.Rechercher C7G. 12 x Large. Rechercher C7G.16X Large. Rechercher M7G.Large.Rechercher m7g.xlarge.Search M7G.2XLarge.Rechercher	Graviton3 ne prend en charge que. GP3

Type d'instance	instances	Restrictions
	M7G.4XLarge.Rechercher	
	M7G.8XLarge.Rechercher	
	M7G. 12 x Large. Rechercher	
	M7G.16XLarge.Rechercher	
	R7G.Medium.Search	
	R7G.Large.Rechercher	
	R7G.xLarge.Search	
	R7G.2XLarge.Rechercher	
	R7G.4XLarge.Rechercher	
	R7G.8XLarge.Rechercher	

Type d’instance	instances	Restrictions
	R7G. 12 x Large. Rechercher R7G.16XLarge.Rechercher R7GD.Large.Rechercher r7gd.xlarge.Rechercher R7GD.2XLarge.Rechercher R7GD.4XLarge.Rechercher R7GD.8XLarge.Rechercher R7GD.12XLarge.Rechercher R7GD.16XLarge.Rechercher	

Type d’instance	instances	Restrictions

OR1	`or1.medium.search`
	`or1.large.search`
	`or1.xlarge.search`
	`or1.2xlarge.search`
	`or1.4xlarge.search`
	`or1.8xlarge.search`
	`or1.12xlarge.search`
	`or1.16xlarge.search`
		• Les types d' OR1 instance nécessitent la version OpenSearch 2.11 ou ultérieure.
		• OR1 les instances ne sont compatibles qu'avec les nœuds maîtres des autres types d'instances Graviton (C6g, M6g, R6g).

Type d'instance	instances	Restrictions
OR2	or2.medium.search or2.large.search or2.xlarge.search or2.2xlarge.search or2.4xlarge.search or2.8xlarge.search or2.12xlarge.search or2.16xlarge.search	

Type d'instance	instances	Restrictions
OM2	om2.large.search om2.xlarge.search om2.2xlarge.search om2.4xlarge.search om2.8xlarge.search om2.12xlarge.search om2.16xlarge.search	

Type d'instance	instances	Restrictions

Im4gn	`im4gn.large.search`
	`im4gn.xlarge.search`
	`im4gn.2xlarge.search`
	`im4gn.4xlarge.search`
	`im4gn.8xlarge.search`
	`im4gn.16xlarge.search`
		• Les types d'instances Im4gn nécessitent Elasticsearch 7.9 ou version ultérieure ou toute autre version de OpenSearch, et ne prennent pas en charge les volumes de stockage EBS.
		• Les instances Im4gn ne sont compatibles qu'avec les autres types d'instances Graviton (C6g, m6g, R6g, R6gd). Vous ne pouvez pas combiner des instances Graviton et non-Graviton au sein du même cluster.

Type d’instance	instances	Restrictions

C5	`c5.large.search`
	`c5.xlarge.search`
	`c5.2xlarge.search`
	`c5.4xlarge.search`
	`c5.9xlarge.search`
	`c5.18xlarge.search`
		Les types d'instances C5 nécessitent Elasticsearch 5.1 ou version ultérieure ou toute version de. OpenSearch

Type d’instance	instances	Restrictions

C6g	`c6g.large.search`
	`c6g.xlarge.search`
	`c6g.2xlarge.search`
	`c6g.4xlarge.search`
	`c6g.8xlarge.search`
	`c6g.12xlarge.search`
		• Les types d'instances C6g nécessitent Elasticsearch 7.9 ou version ultérieure ou toute version de. OpenSearch
		• Les instances C6g ne sont compatibles qu'avec les autres types d'instances Graviton (Im4gn, m6g, R6g, R6gd). Vous ne pouvez pas combiner des instances Graviton et non-Graviton au sein du même cluster.

Type d'instance	instances	Restrictions
I3	i3.large.search i3.xlarge.search i3.2xlarge.search i3.4xlarge.search i3.8xlarge.search i3.16xlarge.search	
M5	m5.large.search m5.xlarge.search m5.2xlarge.search m5.4xlarge.search m5.12xlarge.search	Les types d'instances M5 nécessitent Elasticsearch 5.1 ou version ultérieure ou toute version de. OpenSearch

Type d’instance	instances	Restrictions

M6g	`m6g.large.search`
	`m6g.xlarge.search`
	`m6g.2xlarge.search`
	`m6g.4xlarge.search`
	`m6g.8xlarge.search`
	`m6g.12xlarge.search`
		• Les types d'instances m6G nécessitent Elasticsearch 7.9 ou version ultérieure ou toute version de. OpenSearch
		• Les instances m6G ne sont compatibles qu'avec les autres types d'instances Graviton (Im4gn, C6g, R6g, R6gd). Vous ne pouvez pas combiner des instances Graviton et non-Graviton au sein du même cluster.

Type d’instance	instances	Restrictions

R5	`r5.large.search`
	`r5.xlarge.search`
	`r5.2xlarge.search`
	`r5.4xlarge.search`
	`r5.12xlarge.search`
		Les types d'instances R5 nécessitent Elasticsearch 5.1 ou version ultérieure ou toute version de. OpenSearch

Type d’instance	instances	Restrictions

R6g	`r6g.large.search`
	`r6g.xlarge.search`
	`r6g.2xlarge.search`
	`r6g.4xlarge.search`
	`r6g.8xlarge.search`
	`r6g.12xlarge.search`
		• Les types d'instances R6g nécessitent Elasticsearch 7.9 ou version ultérieure ou toute version de. OpenSearch
		• Les instances R6g ne sont compatibles qu'avec les autres types d'instances Graviton (Im4gn, C6g, m6g, R6gd). Vous ne pouvez pas combiner des instances Graviton et non-Graviton au sein du même cluster.

Type d'instance	instances	Restrictions

R6gd	`r6gd.large.search`
	`r6gd.xlarge.search`
	`r6gd.2xlarge.search`
	`r6gd.4xlarge.search`
	`r6gd.8xlarge.search`
	`r6gd.12xlarge.search`
	`r6gd.16xlarge.search`
		• Les types d'instances R6gd nécessitent Elasticsearch 7.9 ou version ultérieure ou toute autre version de OpenSearch, et ne prennent pas en charge les volumes de stockage EBS.
		• Les instances R6gd ne sont compatibles qu'avec les autres types d'instances Graviton (Im4gn, C6g, M6g, R6g). Vous ne pouvez pas combiner des instances Graviton et non-Graviton au sein du même cluster.

Type d'instance	instances	Restrictions

T3	`t3.small.search`
	
	`t3.medium.search`
		• Les types d'instances T3 nécessitent Elasticsearch 5.6 ou version ultérieure ou toute version de. OpenSearch
		• Vous ne pouvez utiliser les types d'instances T3 que si votre domaine est provisionné sans mode veille. Pour de plus amples informations, veuillez consulter the section called "Multi-AZ sans mode veille".
		• Vous ne pouvez utiliser les types d'instance T3 que si le nombre d'instances pour votre domaine est inférieur ou égal à 10.
		• Les types d'instances T3 ne prennent pas en charge le UltraWarm stockage, le stockage à froid ou le réglage automatiq ue.

Type d’instance	instances	Restrictions

c7i	`c7i.large.search`
	`c7i.xlarge.search`
	`c7i.2xlarge.search`
	`c7i.4xlarge.search`
	`c7i.8xlarge.search`
	`c7i.12xlarge.search`
	`c7i.16xlarge.search`
		• L'instance c7i nécessite Elasticsearch 5.1 ou version ultérieure ou toute autre version de OpenSearch, et ne prend en charge que les volumes de stockage. GP3

Type d’instance	instances	Restrictions

m7i	`m7i.large.search`
	`m7i.xlarge.search`
	`m7i.2xlarge.search`
	`m7i.4xlarge.search`
	`m7i.8xlarge.search`
	`m7i.12xlarge.search`
	`m7i.16xlarge.search`
		• L'instance m7i nécessite Elasticsearch 5.1 ou version ultérieure ou toute autre version de OpenSearch, et ne prend en charge que les volumes de stockage. GP3

Type d’instance	instances	Restrictions

r7i	`r7i.large.search`
	`r7i.xlarge.search`
	`r7i.2xlarge.search`
	`r7i.4xlarge.search`
	`r7i.8xlarge.search`
	`r7i.12xlarge.search`
	`r7i.16xlarge.search`
		• L'instance r7i nécessite Elasticsearch 5.1 ou version ultérieure ou toute version de OpenSearch, et ne prend en charge que les volumes de stockage. GP3

Types d’instance d’ancienne génération

OpenSearch Le service propose des types d'instances de génération précédente aux utilisateurs qui ont optimisé leurs applications autour d'eux et qui n'ont pas encore effectué de mise à niveau. Nous vous encourageons à utiliser les types d’instance de la génération actuelle pour obtenir les meilleures performances, mais nous continuons à prendre en charge les types d’instance de la génération précédente suivants.

Type d'instance	instances	Restrictions

C4	`c4.large.search`
	`c4.xlarge.search`
	`c4.2xlarge.search`
	`c4.4xlarge.search`
	`c4.8xlarge.search`	

I2	`i2.xlarge.search` `i2.2xlarge.search`	

M3	`m3.medium.search`
	`m3.large.search`
	`m3.xlarge.search`
	`m3.2xlarge.search`
		• Les types d'instance M3 ne prennent pas en charge le chiffrement des données au repos, le contrôle précis des accès ou la recherche inter-clusters.
		• Les types d'instances M3 sont soumis à des restrictions supplémentaires par OpenSearch version. Pour en savoir plus, veuillez consulter la section the section called "Type d'instance M3 non valide".

M4	`m4.large.search`	

Type d'instance	instances	Restrictions

	`m4.xlarge.search`
	`m4.2xlarge.search`
	`m4.4xlarge.search`
	`m4.10xlarge.search`	

R3	`r3.large.search`
	`r3.xlarge.search`
	`r3.2xlarge.search`
	`r3.4xlarge.search`
	`r3.8xlarge.search`
		Les types d'instance R3 ne prennent pas en charge le chiffrement des données au repos ou le contrôle précis des accès.

Type d'instance	instances	Restrictions

R4	`r4.large.search`
	`r4.xlarge.search`
	`r4.2xlarge.search`
	`r4.4xlarge.search`
	`r4.8xlarge.search`
	`r4.16xlarge.search`	

T2	`t2.micro.search`
	`t2.small.search`
	`t2.medium.search`
		• Vous pouvez utiliser les types d'instance T2 uniquement si le nombre d'instances pour votre domaine est de 10 au maximum.
		• Le type d'instance `t2.micro.search` prend uniquement en charge les versions 1.5 et 2.3 d'Elasticsearch.
		• Les types d'instances T2 ne prennent pas en charge le chiffrement des données au repos, le contrôle d'accès précis, le UltraWarm stockage, le stockage à froid, la recherche entre clusters ou le réglage automatique.

Tip

Nous recommandons souvent d'utiliser différents types d'instances pour les nœuds principaux dédiés et les nœuds de données.

Fonctionnalités par version de moteur dans Amazon OpenSearch Service

De nombreuses fonctionnalités du OpenSearch Service nécessitent une OpenSearch version minimale ou une version existante d'Elasticsearch OSS. Si vous disposez de la version minimale requise pour une fonctionnalité, mais que cette dernière n'est pas disponible sur votre domaine, mettez à jour le logiciel de service de votre domaine.

Fonctionnalité	OpenSearch Version minimale requise	Version minimale d'Elasticsearch requise
Plugins personnalisés	2.15	Non incluse
Nœud de coordination dédié	1.0	6.8
Prise en charge de VPC	1.0	1.0
Exiger HTTPS pour tout le trafic vers le domaine		
Prise en charge multi-AZ		
Nœuds maîtres dédiés		

Fonctionnalité	OpenSearch Version minimale requise	Version minimale d'Elasticsearch requise
Packages personnalisés		
Points de terminaison personnalisés		
Publication des journaux lents		
Publication des journaux d'erreurs	1.0	5.1
Chiffrement de données au repos		
Authentification Cognito pour les tableaux de bord OpenSearch		
Mises à niveau sur place		

Fonctionnalité	OpenSearch Version minimale requise	Version minimale d'Elasticsearch requise
Prise en charge de Curator	Non incluse	5.1
Instantanés horaires automatisés	1.0	5.3
Node-to-node chiffrement	1.0	6.0
Prise en charge du client REST de haut niveau Java		
Compression des requêtes et réponses HTTP		
Alerte	1.0	6.2
SQL	1.0	6,5
Recherche croisée entre clusters	1.0	6.7
Contrôle précis des accès		

Fonctionnalité	OpenSearch Version minimale requise	Version minimale d'Elasticsearch requise
Authentification SAML pour les tableaux de bord OpenSearch		
Auto-Tune		
Réindexation à distance		
UltraWarm	1.0	6.8
Gestion d'états des index		
k-NN par distance euclidienne	1.0	7.1
Détection des anomalies	1.0	7.4
k-NN par similarité cosinus	1.0	7.7
Learning to Rank		

Fonctionnalité	OpenSearch Version minimale requise	Version minimale d'Elasticsearch requise
Piped Processing Language	1.0	7,9
OpenSearch Tableaux de bord et rapports		
OpenSearch Tableaux de bord Trace Analytics		
Instances Graviton basées sur ARM		
Stockage à froid		
Distance de Hamming, distance de norme L1 et scripting indolore pour k-NN	1.0	7,10
Recherche asynchrone		

Fonctionnalité	OpenSearch Version minimale requise	Version minimale d'Elasticsearch requise
Transformations d'index	1.0	Non incluse
Réplication inter-clusters (CCR)	1.1	7,10
ML Commons	1.3	Non incluse
Notifications	2.3	Non incluse
Recherche ponctuelle	2,5	Non incluse
Connecteurs d'apprentissage automatique	2.9	Non incluse
Recherche sémantique	2.9	Non incluse
Recherche sémantique multimodale	2.11	Non incluse
Sources de données à requête directe	2.11	Non incluse
Recherche par segment simultanée	2,13	Non incluse

Fonctionnalité	OpenSearch Version minimale requise	Version minimale d'Elasticsearch requise
Génération de requêtes en langage naturel	2,13	Non incluse
Prise en charge d'Amazon Q	2,17	Non incluse

Pour de plus amples informations sur les plugins, qui permettent certaines de ces fonctionnalités et fonctionnalités supplémentaires, veuillez consulter the section called "Plugins par version du moteur". Pour plus d'informations sur l'API OpenSearch pour chaque version, consultez the section called "Opérations prises en charge".

Plug-ins par version de moteur dans Amazon OpenSearch Service

Les domaines Amazon OpenSearch Service sont fournis avec des plugins fournis par la OpenSearch communauté. Le service déploie et gère automatiquement les plug-ins pour vous, mais il déploie différents plug-ins en fonction de la version OpenSearch ou de l'ancienne version d'Elasticsearch OSS que vous avez choisie pour votre domaine.

Le tableau suivant répertorie les plug-ins par OpenSearch version, ainsi que les versions compatibles de l'ancien logiciel Elasticsearch OSS. Il inclut uniquement les plugins avec lesquels vous pourriez interagir ; il n'est pas exhaustif. OpenSearch Le service utilise des plug-ins supplémentaires pour activer les fonctionnalités de base du service, tels que le plug-in S3 Repository pour les instantanés et le plug-in OpenSearchPerformance Analyzer pour l'optimisation et la surveillance. Pour obtenir la liste complète de tous les plugins exécutés sur votre domaine, effectuez la demande suivante :

```
GET _cat/plugins?v
```

Plugin	OpenSearch Version minimale requise	Version minimale d'Elasticsearch requise
HanLP	2.11	Non pris en charge
Analyse hébraïque	2.11	Non pris en charge
Amazon Personalize Search Ranking	2.9	Non pris en charge
Recherche neuronale	2.9	Non pris en charge
Analyses de sécurité	2,5	Non pris en charge
OpenSearch notifications	2.3	Non pris en charge
ML Commons	1.3	Non pris en charge
Analyse Sudachi (recommandée pour les japonais)	1.3	Non pris en charge
STConvert	1.3	Non pris en charge
Analyse du pinyin	1.3	Non pris en charge
Analyse Nori	1.3	Non pris en charge

Plugin	OpenSearch Version minimale requise	Version minimale d'Elasticsearch requise
OpenSearch observabilité	1.2	Non pris en charge
OpenSearch réplication entre clusters	1.1	7,10
OpenSearch recherche asynchrone	1.0	7,10
IK (Chinese) Analysis	1.0	7.7
Vietnamese Analysis		
Thai analysis		
Learning to Rank		
OpenSearch détection d'anomalies	1.0	7.4
OpenSearch k-NN	1.0	7.1
OpenSearch Gestion d'états des index	1.0	6.8

Plugin	OpenSearch Version minimale requise	Version minimale d'Elasticsearch requise
OpenSearch sécurité	1.0	6.7
OpenSearch SQL	1.0	6,5
OpenSearch alertant	1.0	6.2
Ukrainian Analysis	1.0	5.3
Mapper Size	1.0	5.3
Mapper Murmur3	1.0	5.1
Ingest User Agent Processor	1.0	5.1
Ingest Attachment Processor	1.0	5.1
Stempel Polish Analysis	1.0	5.1
Smart Chinese Analysis	1.0	5.1

Plugin	OpenSearch Version minimale requise	Version minimale d'Elasticsearch requise
Analyse coréenne Seunjeon	1.0	5.1
Phonetic Analysis	1.0	2.3
Japanese (kuromoji) Analysis	1.0	Inclus sur tous les domaines
ICU Analysis	1.0	Inclus sur tous les domaines

Plug-ins optionnels

Outre les plug-ins par défaut préinstallés, Amazon OpenSearch Service prend en charge plusieurs plug-ins d'analyse de langue facultatifs. Vous pouvez utiliser le AWS Management Console et AWS CLI pour associer un plugin à un domaine, dissocier un plugin d'un domaine et répertorier tous les plugins. Un package de plug-in optionnel est compatible avec une OpenSearch version spécifique et ne peut être associé qu'à des domaines dotés de cette version.

Notez que pour le plugin Sudachi, lorsque vous réassociez un fichier de dictionnaire, cela ne se répercute pas immédiatement sur le domaine. Le dictionnaire est actualisé lorsque le prochain blue/green déploiement s'exécute sur le domaine dans le cadre d'une modification de configuration ou d'une autre mise à jour. Vous pouvez également créer un nouveau package avec les données mises à jour, créer un nouvel index à l'aide de ce nouveau package, réindexer l'index existant dans le nouvel index, puis supprimer l'ancien index. Si vous préférez utiliser l'approche de réindexation, utilisez un alias d'index afin de ne pas perturber votre trafic.

Les plugins facultatifs utilisent le type de `ZIP-PLUGIN` package. Pour plus d'informations sur les plug-ins facultatifs, consultezthe section called “Packages”.

Opérations prises en charge dans Amazon OpenSearch Service

OpenSearch Le service prend en charge de nombreuses versions OpenSearch et des anciennes versions d'Elasticsearch OSS. Les sections suivantes présentent les opérations prises en charge par OpenSearch Service pour chaque version.

Rubriques

- Différences notables entre API

Différences notables entre API

Nouvelle liste APIs

Pour prendre en charge les grands clusters comportant un grand nombre d'index et de partitions, nous avons introduit une nouvelle liste APIs avec support de pagination, telle que _. list/indices and _list/shards L'API List récupère les statistiques relatives aux index et aux partitions dans un format paginé. Cela simplifie le traitement des réponses contenant de nombreux index.

- `_list/indices`: _liste/indices
- `_list/shards`: _listes/fragments

Modifications apportées à une version existante APIs

Pour prendre en charge les grands clusters, nous avons ajouté un support dans l'`_cluster/stats`API pour ajouter des filtres métriques supplémentaires afin de permettre de récupérer uniquement les réponses statistiques pertinentes, par exemple `_cluster/stats/<metric>/nodes/<node-filters>` et`_cluster/stats/<metric>/<index_metric>/nodes/<node-filters>`. Pour plus de détails, consultez _cluster/stats.

Nous avons ajouté la prise en charge de l'annulation des tâches dans l'`_cat/shards`API en spécifiant un paramètre de `cancel_after_time_interval` demande. Pour plus de détails, consultez _cat/shards.

Limiter la taille de réponse pour l'API _cat

Pour prendre en charge les clusters de grande taille avec un nombre total d'instances supérieur à 200 entre les nœuds de données et les nœuds chauds, nous avons une limite de 10 000 au nombre

d'index renvoyés par le. `_cat/segments API` Si le nombre d'index dans la réponse dépasse cette limite, l'API renvoie une erreur 429. Pour éviter cela, vous pouvez spécifier un filtre de modèle d'index dans votre requête, tel que`_cat/segments/<index-pattern>`.

Paramètres et statistiques

OpenSearch Le service accepte uniquement les demandes PUT adressées à l'`_cluster/settings`API qui utilisent le formulaire de paramètres « plat ». Il rejette celles qui utilisent le formulaire des paramètres étendus.

```
// Accepted
PUT _cluster/settings
{
  "persistent" : {
    "action.auto_create_index" : false
  }
}

// Rejected
PUT _cluster/settings
{
  "persistent": {
    "action": {
      "auto_create_index": false
    }
  }
}
```

Le client Java REST de haut niveau utilise le formulaire étendu, donc si vous devez envoyer des demandes de paramètres, utilisez le client de bas niveau.

Avant Elasticsearch 5.3, l'`_cluster/settings`API sur les domaines de OpenSearch service ne prenait en charge que la `PUT` méthode HTTP, et non la `GET` méthode. OpenSearch et les versions ultérieures d'Elasticsearch prennent en charge `GET` cette méthode, comme illustré dans l'exemple suivant :

```
GET https://domain-name.region.es.amazonaws.com/_cluster/settings?pretty
```

Voici un exemple de retour :

```
{
```

```
  "persistent": {
    "cluster": {
      "routing": {
        "allocation": {
          "cluster_concurrent_rebalance": "2",
          "node_concurrent_recoveries": "2",
          "disk": {
            "watermark": {
              "low": "1.35gb",
              "flood_stage": "0.45gb",
              "high": "0.9gb"
            }
          },
          "node_initial_primarirecoveries": "4"
        }
      }
    },
    "indices": {
      "recovery": {
        "max_bytper_sec": "40mb"
      }
    }
  }
}
```

Si vous comparez les réponses d'un OpenSearch cluster open source et d'un OpenSearch service pour certains paramètres et statistiques APIs, vous remarquerez peut-être des champs manquants. OpenSearch Le service expédie certaines informations qui exposent les éléments internes du service, tels que le chemin de données du système de fichiers depuis `_nodes/stats` ou le nom et la version du système d'exploitation depuis. `_nodes`

API Shrink

L'API `_shrink` peut entraîner l'échec des mises à jour, modifications de configuration et suppressions de domaine. Nous vous déconseillons de l'utiliser sur les domaines qui exécutent les versions 5.3 et 5.1 d'Elasticsearch. Ces versions ont un bogue qui peut entraîner l'échec de la restauration des instantanés des index réduits.

Si vous utilisez l'`_shrink`API sur d'autres OpenSearch versions ou versions d'Elasticsearch, effectuez la demande suivante avant de démarrer l'opération de réduction :

```
PUT https://domain-name.region.es.amazonaws.com/source-index/_settings
```

```
{
  "settings": {
    "index.routing.allocation.require._name": "name-of-the-node-to-shrink-to",
    "index.blocks.read_only": true
  }
}
```

Puis, effectuez les demandes suivantes après avoir exécuté l'opération de réduction :

```
PUT https://domain-name.region.es.amazonaws.com/source-index/_settings
{
  "settings": {
    "index.routing.allocation.require._name": null,
    "index.blocks.read_only": false
  }
}

PUT https://domain-name.region.es.amazonaws.com/shrunken-index/_settings
{
  "settings": {
    "index.routing.allocation.require._name": null,
    "index.blocks.read_only": false
  }
}
```

Nouvelle liste APIs

Pour prendre en charge les grands clusters contenant un grand nombre d'index et de partitions, nous avons introduit une nouvelle liste APIs avec support de pagination, c'est-à-dire et. `_list/indices` `_list/shards` L'API List récupère les statistiques relatives aux index et aux partitions dans un format paginé. Cela simplifie le traitement des réponses contenant de nombreux index. Pour plus d'informations sur`_list/indices`, consultez la section Liste des indices. Pour plus d'informations sur`_list/shards`, voir Répertorier les partitions.

Modifications apportées à une version existante APIs

Pour prendre en charge les grands clusters, nous avons ajouté un support dans `_cluster/stats/<metric>/nodes/<node-filters>` et`_cluster/stats/<metric>/<index_metric>/nodes/<node-filters>`. Pour plus d'informations`_cluster/stats`, consultez la section Statistiques du cluster.

Limiter la taille de réponse pour _cat APIs

Pour prendre en charge les clusters de grande taille dont le nombre total d'instances est supérieur à 200 entre les nœuds de données et les nœuds chauds, nous avons fixé une limite de 10 000 au nombre d'index renvoyés par l'API _cat/segments. Si le nombre d'index de la réponse dépasse cette limite, l'API renvoie une 429 erreur. Pour éviter cela, vous pouvez spécifier un filtre de modèle d'index dans votre requête (par exemple,`_cat/segments/<index-pattern>`).

De plus, la prise en charge de l'annulation des tâches est désormais disponible pour `_cat/shards` l'API pour l'annulation des tâches en spécifiant le paramètre de `cancel_after_time_interval` demande. Pour plus d'informations à ce sujet, consultez la section CAT shards.

Choix des types d'instances pour les nœuds maîtres dédiés

Le tableau suivant fournit des recommandations pour choisir les types d'instances appropriés pour les nœuds maîtres dédiés :

RAM	Nombre maximum de nœuds pris en charge	Nombre maximal de partitions prises en charge	
2 Go	10	1 000	
4 Go	10	5 000	
8 Go	30	15 000	
16 Go	60	30 000	
32 GO	120	60 000	
64 Go	240	120 000	
128 Go	480	240 000	
256 Go	1 002	500 000	

OpenSearch versions 2.18 et 2.19

Pour plus d'informations sur les opérations OpenSearch 2.18 et 2.19, consultez la référence d'OpenSearch API REST ou la référence d'API du plugin spécifique. Pour plus de détails sur les modifications apportées à ces versions, consultez les notes de version 2.18 et les notes de version 2.19.

OpenSearch version 2.17

Pour la OpenSearch version 2.17, le OpenSearch Service prend en charge les opérations suivantes. Pour plus d'informations sur la plupart des opérations, consultez la référence de l'OpenSearch API REST ou la référence de l'API du plugin en question.

> **Note**
>
> À partir de la OpenSearch version 2.17, le `cluster.max_shards_per_node` paramètre ne peut pas être modifié. À partir de la version OpenSearch 2.17, OpenSearch Service prend en charge 1 000 partitions pour 16 Go de mémoire de segment de mémoire JVM, jusqu'à un maximum de 4 000 partitions par nœud.

- Toutes les opérations dans le chemin d'index (telles que */index-name* `/_forcemerge` , */index-name* `/update/`*id* et */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (sauf `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_list`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/pipeline`
- `/_search/point_in_time`
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`

- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.search.request.slowlog.level
  - cluster.search.request.slowlog.threshold.warn
  - cluster.search.request.slowlog.threshold.info
  - cluster.search.request.slowlog.threshold.debug
  - cluster.search.request.slowlog.threshold.trace
  - search.phase_took_enabled
- /_cluster/state
- /_cluster/stats
- /_count
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

- `/_dashboards`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Reportez-vous à la `PUT` méthode. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux OpenSearch opérations génériques prises en charge par le OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

> **Note**
>
> À l'heure actuelle, la modification de la fonctionnalité de `cluster.max_shards_per_node` configuration n'est pas activée pour les clients utilisant le mode Multi-AZ (zone de disponibilité) avec mode veille.

OpenSearch version 2.15

Pour la OpenSearch version 2.15, le OpenSearch Service prend en charge les opérations suivantes. Pour plus d'informations sur la plupart des opérations, consultez la référence de l'OpenSearch API REST ou la référence de l'API du plugin en question.

- Toutes les opérations dans le chemin d'index (telles que `/index-name /_forcemerge` , `/index-name /`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`

update/*id* et /*index-name* /_close)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
  - cluster.search.request.slowlog.level
  - cluster.search.request.slowlog.threshold.warn
- /_flush
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search/pipeline
- /_search/point_in_time
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

    - `cluster.search.request.slowlog.threshold.info`
    - `cluster.search.request.slowlog.threshold.debug`
    - `cluster.search.request.slowlog.threshold.trace`
    - `search.phase_took_enabled`
  - `/_cluster/state`
  - `/_cluster/stats`
  - `/_count`
  - `/_dashboards`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux OpenSearch opérations génériques prises en charge par le OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

OpenSearch version 2.13

Pour la OpenSearch version 2.13, le OpenSearch Service prend en charge les opérations suivantes. Pour plus d'informations sur la plupart des opérations, consultez la référence de l'OpenSearch API REST ou la référence de l'API du plugin en question.

- Toutes les opérations dans le chemin d'index (telles que */index-name* /_forcemerge , */index-name* /update/*id* et */index-name* /_close)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_refresh
- /_reindex [1]
- /_render
- /_resolve/index
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search/pipeline
- /_search/point_in_time
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

  - indices.breaker.total.limit
  - cluster.max_shards_per_node
  - cluster.search.request.slowlog.level
  - cluster.search.request.slowlog.threshold.warn
  - cluster.search.request.slowlog.threshold.info
  - cluster.search.request.slowlog.threshold.debug
  - cluster.search.request.slowlog.threshold.trace
  - search.phase_took_enabled
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération /_tasks avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour /_search/scroll avec un corps de message doivent spécifier "Content-Length" dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans scroll_id les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre scroll_id des valeurs à OpenSearch Service.

3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode PUT. Pour obtenir des informations sur la méthode GET, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux OpenSearch opérations génériques prises en charge par le OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

OpenSearch version 2.11

Pour la OpenSearch version 2.11, le OpenSearch Service prend en charge les opérations suivantes. Pour plus d'informations sur la plupart des opérations, consultez la référence de l'OpenSearch API REST ou la référence de l'API du plugin en question.

- Toutes les opérations dans le chemin d'index (telles que `/index-name /_forcemerge`, `/index-name /update/id` et `/index-name /_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (sauf `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` pour plusieurs propriétés[4] :
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/pipeline`
- `/_search/point_in_time`
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`

- `action.auto_create_index`
- `action.search.shard_count.limit`
- `indices.breaker.fielddata.limit`
- `indices.breaker.request.limit`
- `indices.breaker.total.limit`
- `cluster.max_shards_per_node`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_dashboards`
- `/_plugins/_ml`
- `/_plugins/_notifications`
- `/_plugins/_ppl`
- `/_plugins/_security`
- `/_plugins/_security_analytics`
- `/_plugins/_sm`
- `/_plugins/_sql`
- `/_percolate`
- `/_rank_eval`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux OpenSearch opérations génériques prises en charge par le OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

OpenSearch version 2.9

Pour la OpenSearch version 2.9, le OpenSearch Service prend en charge les opérations suivantes. Pour plus d'informations sur la plupart des opérations, consultez la référence de l'OpenSearch API REST ou la référence de l'API du plugin en question.

- Toutes les opérations dans le chemin d'index (telles que */index-name* /_forcemerge , */index-name* /update/*id* et */index-name* /_close)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_refresh
- /_reindex [1]
- /_render
- /_resolve/index
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search/pipeline
- /_search/point_in_time
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

- `indices.breaker.total.limit`
- `cluster.max_shards_per_node`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_dashboards`
- `/_plugins/_security_analytics`
- `/_plugins/_sm`
- `/_plugins/_sql`
- `/_percolate`
- `/_rank_eval`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode PUT. Pour obtenir des informations sur la méthode GET, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux OpenSearch opérations génériques prises en charge par le OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

OpenSearch version 2.7

Pour la OpenSearch version 2.7, le OpenSearch Service prend en charge les opérations suivantes. Pour plus d'informations sur la plupart des opérations, consultez la référence de l'OpenSearch API REST ou la référence de l'API du plugin en question.

- Toutes les opérations dans le chemin d'index (telles que
- `/_delete_by_query` [1]
- `/_explain`
- `/_refresh`
- `/_reindex` [1]

- */index-name* /_forcemerge , */index-name* /update/*id* et */index-name* /_close)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_render
- /_resolve/index
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search/point_in_time
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

- `/_dashboards`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux OpenSearch opérations génériques prises en charge par le OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

OpenSearch version 2.5

Pour la OpenSearch version 2.5, le OpenSearch Service prend en charge les opérations suivantes. Pour plus d'informations sur la plupart des opérations, consultez la référence de l'OpenSearch API REST ou la référence de l'API du plugin en question.

- Toutes les opérations dans le chemin d'index (telles que */index-name* `/_forcemerge` , */index-name* `/update/`*id* et */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search/point_in_time`

- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ml
- /_plugins/_notifications
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_security_analytics
- /_plugins/_sm
- /_plugins/_sql
- /_percolate
- /_rank_eval
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération /_tasks avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour /_search/scroll avec un corps de message doivent spécifier "Content-Length" dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut.

Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.

3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode PUT. Pour obtenir des informations sur la méthode GET, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux OpenSearch opérations génériques prises en charge par le OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

OpenSearch version 2.3

Pour la OpenSearch version 2.3, le OpenSearch Service prend en charge les opérations suivantes. Pour plus d'informations sur la plupart des opérations, consultez la référence de l'OpenSearch API REST ou la référence de l'API du plugin en question.

- Toutes les opérations dans le chemin d'index (telles que */index-name* `/_forcemerge` , */index-name* `/update/`*id* et */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (sauf `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`

- `/_cluster/settings` pour plusieurs propriétés[4] :
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `cluster.max_shards_per_node`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_dashboards`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ml`
- `_plugins/_notifications`
- `/_plugins/_ppl`
- `/_plugins/_security`
- `/_plugins/_sql`
- `/_percolate`
- `/_rank_eval`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux OpenSearch opérations génériques prises en charge par le OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.

5. Consultez the section called “API Shrink”.

OpenSearch version 1.3

Pour la OpenSearch version 1.3, le OpenSearch Service prend en charge les opérations suivantes. Pour plus d'informations sur la plupart des opérations, consultez la référence de l'OpenSearch API REST ou la référence de l'API du plugin en question.

- Toutes les opérations dans le chemin d'index (telles que */index-name* `/_forcemerge` , */index-name* `/update/`*id* et */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (sauf `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` pour plusieurs propriétés[4] :
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ml`
- `/_plugins/_ppl`
- `/_plugins/_security`
- `/_plugins/_sql`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_percolate
- /_rank_eval

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux OpenSearch opérations génériques prises en charge par le OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

OpenSearch version 1.2

Pour la OpenSearch version 1.2, le OpenSearch Service prend en charge les opérations suivantes. Pour plus d'informations sur la plupart des opérations, consultez la référence de l'OpenSearch API REST ou la référence de l'API du plugin en question.

- Toutes les opérations dans le chemin d'index (telles que `/index-name /_forcemerge`, `/index-name /update/id` et `/index-name /_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (sauf `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` pour plusieurs propriétés[4] :
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `cluster.max_shards_per_node`
- `/_cluster/state`
- `/_delete_by_query`[1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_plugins/_alerting`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ppl`
- `/_plugins/_security`
- `/_plugins/_sql`
- `/_percolate`
- `/_rank_eval`
- `/_refresh`
- `/_reindex`[1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts`[3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query`[1]
- `/_validate`

- `/_cluster/stats`
- `/_count`
- `/_dashboards`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux OpenSearch opérations génériques prises en charge par le OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

OpenSearch version 1.1

Pour la OpenSearch version 1.1, le OpenSearch Service prend en charge les opérations suivantes. Pour plus d'informations sur la plupart des opérations, consultez la référence de l'OpenSearch API REST ou la référence de l'API du plugin en question.

- Toutes les opérations dans le chemin d'index (telles que */index-name* `/_forcemerge` , */index-name* `/update/`*id* et */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]

- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_dashboards
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_plugins/_asynchronous_search
- /_plugins/_alerting
- /_plugins/_anomaly_detection
- /_plugins/_ism
- /_plugins/_ppl
- /_plugins/_security
- /_plugins/_sql
- /_plugins/_transforms
- /_percolate
- /_rank_eval
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération /_tasks avec ces opérations pour vérifier que les demandes se sont correctement terminées.

2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux OpenSearch opérations génériques prises en charge par le OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

OpenSearch version 1.0

Pour la OpenSearch version 1.0, OpenSearch Service prend en charge les opérations suivantes. Pour plus d'informations sur la plupart des opérations, consultez la référence de l'OpenSearch API REST ou la référence de l'API du plugin en question.

- Toutes les opérations dans le chemin d'index (telles que `/index-name /_forcemerge` , `/index-name /update/id` et `/index-name /_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (sauf `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_ltr`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_plugins/_asynchronous_search`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_resolve/index`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`

- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` pour plusieurs propriétés[4] :
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `cluster.max_shards_per_node`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_dashboards`
- `/_plugins/_alerting`
- `/_plugins/_anomaly_detection`
- `/_plugins/_ism`
- `/_plugins/_ppl`
- `/_plugins/_security`
- `/_plugins/_sql`
- `/_plugins/_transforms`
- `/_percolate`
- `/_rank_eval`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux OpenSearch

opérations génériques prises en charge par le OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.

5. Consultez the section called “API Shrink”.

Elasticsearch version 7.10

Pour Elasticsearch 7.10, le OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que */index-name* /_forcemerge , */index-name* /update/*id* et */index-name* /_close)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_index_template [6]
- /_ingest/pipeline
- /_index_template
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_opendistro/_asynchronous_search
- /_opendistro/_anomaly_detection
- /_opendistro/_ism
- /_opendistro/_ppl
- /_opendistro/_security
- /_refresh
- /_reindex [1]
- /_render
- /_resolve/index
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template [6]
- /_update_by_query [1]
- /_validate

  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_cluster/state
- /_cluster/stats
- /_count
- /_opendistro/_sql
- /_percolate
- /_plugin/kibana
- /_plugins/_replication
- /_rank_eval

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode PUT. Pour obtenir des informations sur la méthode GET, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.
6. Les modèles d'index hérités (`_template`) ont été remplacés par des modèles composables (`_index_template`) à partir d'Elasticsearch 7.8. Les modèles composables ont la priorité sur les modèles hérités. Si aucun modèle composable ne correspond à un index donné, un modèle hérité peut toujours correspondre et être appliqué. L'`_template`opération fonctionne toujours sur OpenSearch les versions ultérieures d'Elasticsearch OSS, mais les appels GET aux deux types de modèles renvoient des résultats différents.

Elasticsearch version 7.9

Pour Elasticsearch 7.9, OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que */index-name* /_forcemerge , */index-name* /update/*id* et */index-name* /_close)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_index_template [6]
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_opendistro/_anomaly_detection
- /_opendistro/_ism
- /_opendistro/_ppl
- /_opendistro/_security
- /_opendistro/_sql
- /_percolate
- /_plugin/kibana
- /_rank_eval
- /_refresh
- /_reindex [1]
- /_render
- /_resolve/index
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template [6]
- /_update_by_query [1]
- /_validate

  - `cluster.max_shards_per_node`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux OpenSearch opérations génériques prises en charge par le OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.
6. Les modèles d'index hérités (`_template`) ont été remplacés par des modèles composables (`_index_template`) à partir d'Elasticsearch 7.8. Les modèles composables ont la priorité sur les modèles hérités. Si aucun modèle composable ne correspond à un index donné, un modèle hérité peut toujours correspondre et être appliqué. L'`_template`opération fonctionne toujours sur OpenSearch les versions ultérieures d'Elasticsearch OSS, mais les appels GET aux deux types de modèles renvoient des résultats différents.

Elasticsearch version 7.8

Pour Elasticsearch 7.8, le OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que
- `/_cluster/state`
- `/_refresh`

- */index-name* /_forceme rge , */index-name* / update/*id* et */index-name* / _close)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nod eattrs)
- /_cluster/allocation/ explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create _index
  - action.search.shar d_count.limit
  - indices.breaker.fi elddata.limit
  - indices.breaker.re quest.limit
  - indices.breaker.to tal.limit
  - cluster.max_shards _per_node
- /_cluster/stats
- /_count
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_index_template [6]
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_aler ting
- /_opendistro/_anom aly_detection
- /_opendistro/_ism
- /_opendistro/_secu rity
- /_opendistro/_sql
- /_percolate
- /_plugin/kibana
- /_rank_eval
- /_reindex [1]
- /_render
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template [6]
- /_update_by_query [1]
- /_validate

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode PUT. Pour obtenir des informations sur la méthode GET, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.
6. Les modèles d'index hérités (`_template`) ont été remplacés par des modèles composables (`_index_template`) à partir d'Elasticsearch 7.8. Les modèles composables ont la priorité sur les modèles hérités. Si aucun modèle composable ne correspond à un index donné, un modèle hérité peut toujours correspondre et être appliqué. L'`_template`opération fonctionne toujours sur OpenSearch les versions ultérieures d'Elasticsearch OSS, mais les appels GET aux deux types de modèles renvoient des résultats différents.

Elasticsearch version 7.7

Pour Elasticsearch 7.7, le OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que */index-name* `/_forcemerge` , */index-name* `/update/`*id* et */index-name* `/_close`)
- `/_alias`
- `/_aliases`
- `/_all`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`

- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - cluster.max_shards_per_node
- /_ingest/pipeline
- /_ltr
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_opendistro/_anomaly_detection
- /_opendistro/_ism
- /_opendistro/_security
- /_opendistro/_sql
- /_percolate
- /_plugin/kibana
- /_rank_eval
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération /_tasks avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour /_search/scroll avec un corps de message doivent spécifier "Content-Length" dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans scroll_id les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre scroll_id des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.

4. Fait référence à la méthode PUT. Pour obtenir des informations sur la méthode GET, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

Elasticsearch version 7.4

Pour Elasticsearch 7.4, le OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que */index-name* /_forcemerge , */index-name* /update/*id* et */index-name* /_close)
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
- /_cluster/state
- /_cluster/stats
- /_count
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_opendistro/_anomaly_detection
- /_opendistro/_ism
- /_opendistro/_security
- /_opendistro/_sql
- /_refresh
- /_reindex [1]
- /_render
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

- `indices.breaker.fielddata.limit`
- `indices.breaker.request.limit`
- `indices.breaker.total.limit`
- `cluster.max_shards_per_node`

- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode PUT. Pour obtenir des informations sur la méthode GET, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

Elasticsearch version 7.1

Pour Elasticsearch 7.1, le OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que */index-name* `/_forcemerge` et */index-name* `/update/`*id*) sauf */index-name* `/_close`

- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]

- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`

- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (sauf `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` pour plusieurs propriétés[4] :
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `cluster.max_shards_per_node`
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_opendistro/_ism`
- `/_opendistro/_security`
- `/_opendistro/_sql`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de

la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.

3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

Elasticsearch version 6.8

Pour Elasticsearch 6.8, OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que `/index-name /_forcemerge` et `/index-name /update/id`) sauf `/index-name /_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cat` (sauf `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` pour plusieurs propriétés[4] :
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_opendistro/_ism`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

- `action.auto_create_index`
- `action.search.shard_count.limit`
- `indices.breaker.fielddata.limit`
- `indices.breaker.request.limit`
- `indices.breaker.total.limit`
- `cluster.max_shards_per_node`
- `cluster.blocks.read_only`

- `/_opendistro/_security`
- `/_opendistro/_sql`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

Elasticsearch version 6.7

Pour Elasticsearch 6.7, OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que */index-name* /_forcemerge et */index-name* /update/*id*) sauf */index-name* /_close
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
- /_cluster/state
- /_cluster/stats
- /_count
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_opendistro/_security
- /_opendistro/_sql
- /_percolate
- /_plugin/kibana
- /_rank_eval
- /_refresh
- /_reindex [1]
- /_render
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

• `cluster.max_shards_per_node`		

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

Elasticsearch version 6.5

Pour Elasticsearch 6.5, OpenSearch Service prend en charge les opérations suivantes.

• Toutes les opérations dans le chemin d'index (telles que */index-name* `/_forcemerge` et */index-name* `/update/`*id*) sauf */index-name* `/_close` • `/_alias` • `/_aliases` • `/_all` • `/_analyze` • `/_bulk`	• `/_cluster/state` • `/_cluster/stats` • `/_count` • `/_delete_by_query` [1] • `/_explain` • `/_field_caps` • `/_field_stats` • `/_flush` • `/_ingest/pipeline` • `/_mapping`	• `/_refresh` • `/_reindex` [1] • `/_render` • `/_rollover` • `/_scripts` [3] • `/_search`[2] • `/_search profile` • `/_shard_stores` • `/_shrink`[5] • `/_snapshot`

- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_opendistro/_sql
- /_percolate
- /_plugin/kibana
- /_rank_eval
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération /_tasks avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour /_search/scroll avec un corps de message doivent spécifier "Content-Length" dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans scroll_id les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre scroll_id des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode PUT. Pour obtenir des informations sur la méthode GET, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.

5. Consultez the section called “API Shrink”.

Elasticsearch version 6.4

Pour Elasticsearch 6.4, OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que */index-name* /_forcemerge et */index-name* /update/*id*) sauf */index-name* /_close
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
- /_cluster/state
- /_cluster/stats
- /_count
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_percolate
- /_plugin/kibana
- /_rank_eval
- /_refresh
- /_reindex [1]
- /_render
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

- `indices.breaker.total.limit`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

Elasticsearch version 6.3

Pour Elasticsearch 6.3, OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que */index-name* `/_forcemerge` et */index-name* `/update/`*id*) sauf */index-name* `/_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cluster/state`
- `/_cluster/stats`
- `/_count`
- `/_delete_by_query` [1]
- `/_explain`
- `/_field_caps`
- `/_field_stats`
- `/_flush`
- `/_ingest/pipeline`
- `/_mapping`
- `/_refresh`
- `/_reindex` [1]
- `/_render`
- `/_rollover`
- `/_scripts` [3]
- `/_search`[2]
- `/_search profile`
- `/_shard_stores`
- `/_shrink`[5]
- `/_snapshot`

- `/_cat` (sauf `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` pour plusieurs propriétés[4] :
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_opendistro/_alerting`
- `/_percolate`
- `/_plugin/kibana`
- `/_rank_eval`
- `/_split`
- `/_stats`
- `/_status`
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.

5. Consultez the section called “API Shrink”.

Elasticsearch version 6.2

Pour Elasticsearch 6.2, OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que */index-name* /_forcemerge et */index-name* /update/*id*) sauf */index-name* /_close
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
- /_cluster/state
- /_cluster/stats
- /_count
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_opendistro/_alerting
- /_percolate
- /_plugin/kibana
- /_rank_eval
- /_refresh
- /_reindex [1]
- /_render
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_split
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

• indices.breaker.total.limit		

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération /_tasks avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour /_search/scroll avec un corps de message doivent spécifier "Content-Length" dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans scroll_id les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre scroll_id des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode PUT. Pour obtenir des informations sur la méthode GET, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

Elasticsearch version 6.0

Pour Elasticsearch 6.0, OpenSearch Service prend en charge les opérations suivantes.

• Toutes les opérations dans le chemin d'index (telles que */index-name* /_forcemerge et */index-name* /update/*id*) sauf */index-name* /_close • /_alias • /_aliases • /_all • /_analyze • /_bulk	• /_cluster/state • /_cluster/stats • /_count • /_delete_by_query [1] • /_explain • /_field_caps • /_field_stats • /_flush • /_ingest/pipeline • /_mapping	• /_render • /_rollover • /_scripts [3] • /_search[2] • /_search profile • /_shard_stores • /_shrink[5] • /_snapshot • /_stats • /_status

- `/_cat` (sauf `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` pour plusieurs propriétés[4] :
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_percolate`
- `/_plugin/kibana`
- `/_refresh`
- `/_reindex` [1]
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.

5. Consultez the section called “API Shrink”.

Elasticsearch version 5.6

Pour Elasticsearch 5.6, le OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que */index-name* /_forcemerge et */index-name* /update/*id*) sauf */index-name* /_close
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[4] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
- /_cluster/state
- /_cluster/stats
- /_count
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_percolate
- /_plugin/kibana
- /_refresh
- /_reindex [1]
- /_render
- /_rollover
- /_scripts [3]
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[5]
- /_snapshot
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

• indices.breaker.total.limit		

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération /_tasks avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour /_search/scroll avec un corps de message doivent spécifier "Content-Length" dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans scroll_id les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre scroll_id des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode PUT. Pour obtenir des informations sur la méthode GET, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
5. Consultez the section called “API Shrink”.

Elasticsearch version 5.5

Pour Elasticsearch 5.5, OpenSearch Service prend en charge les opérations suivantes.

• Toutes les opérations dans le chemin d'index (telles que */index-name* /_forcemerge et */index-name* /update/*id*) sauf */index-name* /_close • /_alias • /_aliases • /_all • /_analyze • /_bulk	• /_cluster/state • /_cluster/stats • /_count • /_delete_by_query [1] • /_explain • /_field_caps • /_field_stats • /_flush • /_ingest/pipeline • /_mapping	• /_render • /_rollover • /_scripts [3] • /_search[2] • /_search profile • /_shard_stores • /_shrink[5] • /_snapshot • /_stats • /_status

- `/_cat` (sauf `/_cat/nodeattrs`)
- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` pour plusieurs propriétés[4] :
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
- `/_mget`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_percolate`
- `/_plugin/kibana`
- `/_refresh`
- `/_reindex` [1]
- `/_tasks`
- `/_template`
- `/_update_by_query` [1]
- `/_validate`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Pour en savoir plus sur l'utilisation des scripts, consultez the section called “Autres ressource prises en charge”.
4. Fait référence à la méthode `PUT`. Pour obtenir des informations sur la méthode `GET`, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.

5. Consultez the section called “API Shrink”.

Elasticsearch version 5.3

Pour Elasticsearch 5.3, OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que */index-name* /_forcemerge et */index-name* /update/*id*) sauf */index-name* /_close
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/allocation/explain
- /_cluster/health
- /_cluster/pending_tasks
- /_cluster/settings pour plusieurs propriétés[3] :
  - action.auto_create_index
  - action.search.shard_count.limit
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
- /_cluster/state
- /_cluster/stats
- /_count
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_msearch
- /_mtermvectors
- /_nodes
- /_percolate
- /_plugin/kibana
- /_refresh
- /_reindex [1]
- /_render
- /_rollover
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[4]
- /_snapshot
- /_stats
- /_status
- /_tasks
- /_template
- /_update_by_query [1]
- /_validate

- indices.breaker.total.limit

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération /_tasks avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour /_search/scroll avec un corps de message doivent spécifier "Content-Length" dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans scroll_id les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre scroll_id des valeurs à OpenSearch Service.
3. Fait référence à la méthode PUT. Pour obtenir des informations sur la méthode GET, consultez the section called “Différences notables entre API”. Cette liste ne fait référence qu'aux opérations Elasticsearch génériques prises en charge par OpenSearch Service et n'inclut pas les opérations prises en charge par des plugins spécifiques pour la détection des anomalies, l'ISM, etc.
4. Consultez the section called “API Shrink”.

Elasticsearch version 5.1

Pour Elasticsearch 5.1, OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que */index-name* /_forcemerge et */index-name* /update/*id*) sauf */index-name* /_close
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat (sauf /_cat/nodeattrs)
- /_cluster/state
- /_cluster/stats
- /_count
- /_delete_by_query [1]
- /_explain
- /_field_caps
- /_field_stats
- /_flush
- /_ingest/pipeline
- /_mapping
- /_mget
- /_render
- /_rollover
- /_search[2]
- /_search profile
- /_shard_stores
- /_shrink[3]
- /_snapshot
- /_stats
- /_status
- /_tasks
- /_template

- `/_cluster/allocation/explain`
- `/_cluster/health`
- `/_cluster/pending_tasks`
- `/_cluster/settings` pour plusieurs propriétés (PUT uniquement) :
  - `action.auto_create_index`
  - `action.search.shard_count.limit`
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
- `/_msearch`
- `/_mtermvectors`
- `/_nodes`
- `/_percolate`
- `/_plugin/kibana`
- `/_refresh`
- `/_reindex` [1]
- `/_update_by_query` [1]
- `/_validate`

1. Les modifications apportées à la configuration de cluster peuvent interrompre ces opérations avant la fin. Nous vous recommandons d'utiliser l'opération `/_tasks` avec ces opérations pour vérifier que les demandes se sont correctement terminées.
2. Les demandes DELETE pour `/_search/scroll` avec un corps de message doivent spécifier `"Content-Length"` dans l'en-tête HTTP. La plupart des clients ajoutent cet en-tête par défaut. Pour éviter tout problème lié aux = caractères dans `scroll_id` les valeurs, utilisez le corps de la requête, et non la chaîne de requête, pour transmettre `scroll_id` des valeurs à OpenSearch Service.
3. Consultez the section called “API Shrink”.

Elasticsearch version 2.3

Pour Elasticsearch 2.3, OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations dans le chemin d'index (telles que `/index-name /_forcemerge` et `/index-name /_recovery`) sauf `/index-name /_close`
- `/_alias`
- `/_aliases`
- `/_all`
- `/_analyze`
- `/_bulk`
- `/_cache/clear` (index uniquement)
- `/_cat` (sauf `/_cat/nodeattrs`)
- `/_cluster/health`
- `/_cluster/settings` pour plusieurs propriétés (PUT uniquement) :
  - `indices.breaker.fielddata.limit`
  - `indices.breaker.request.limit`
  - `indices.breaker.total.limit`
  - `threadpool.get.queue_size`
  - `threadpool.bulk.queue_size`
  - `threadpool.index.queue_size`
  - `threadpool.percolate.queue_size`
  - `threadpool.search.queue_size`
  - `threadpool.suggest.queue_size`
- `/_cluster/stats`
- `/_count`
- `/_flush`
- `/_mapping`
- `/_mget`
- `/_msearch`
- `/_nodes`
- `/_percolate`
- `/_plugin/kibana`
- `/_refresh`
- `/_render`
- `/_search`
- `/_snapshot`
- `/_stats`
- `/_status`
- `/_template`

Elasticsearch version 1.5

Pour Elasticsearch 1.5, OpenSearch Service prend en charge les opérations suivantes.

- Toutes les opérations du chemin d'index, telles que `/index-name /_optimize` et
- `/_cluster/stats`

  /*index-name* /_warmer, sauf /*index-name* /_close
- /_alias
- /_aliases
- /_all
- /_analyze
- /_bulk
- /_cat
- /_cluster/health
- /_cluster/settings pour plusieurs propriétés (PUT uniquement) :
  - indices.breaker.fielddata.limit
  - indices.breaker.request.limit
  - indices.breaker.total.limit
  - threadpool.get.queue_size
  - threadpool.bulk.queue_size
  - threadpool.index.queue_size
  - threadpool.percolate.queue_size
  - threadpool.search.queue_size
  - threadpool.suggest.queue_size
- /_count
- /_flush
- /_mapping
- /_mget
- /_msearch
- /_nodes
- /_percolate
- /_plugin/kibana
- /_plugin/kibana3
- /_plugin/migration
- /_refresh
- /_search
- /_snapshot
- /_stats
- /_status
- /_template

Quotas Amazon OpenSearch Service

Votre AWS compte dispose de quotas par défaut, anciennement appelés limites, pour chaque AWS service. Sauf indication contraire, chaque quota est spécifique à la région.

Pour consulter les quotas pour les domaines et instances de OpenSearch service, Amazon OpenSearch Serverless et Amazon OpenSearch Ingestion, consultez les quotas Amazon OpenSearch Service dans le Références générales AWS.

Pour consulter les quotas de OpenSearch Service dans le AWS Management Console, ouvrez la console Service Quotas. Dans le volet de navigation, sélectionnez AWS services, puis Amazon OpenSearch Service. Pour demander une augmentation de quota, consultez Demande d'augmentation de quota dans le Guide de l'utilisateur Service Quotas.

UltraWarm quotas de stockage

Le tableau suivant répertorie les types d' UltraWarm instances et la quantité maximale de stockage que chaque type peut utiliser. Pour plus d'informations sur UltraWarm, voirthe section called “UltraWarm rangement”.

Type d’instance	Stockage maximum
ultrawarm1.medium.search	1,5 Tio
ultrawarm1.large.search	20 TiO

Nombre de nœuds de données par AZ

Le tableau suivant indique que le nombre total de nœuds de données pour le déploiement AZ est inférieur. La limite globale indique le nombre de nœuds de données par limite, y compris le nombre de nœuds chauds et chauds. Stockage que chaque type peut utiliser.

Configuration AZ	Limite du nombre de hot nodes	Limite du nombre de nœuds chauds	Limite globale (chaud et chaud)
1 - AZ	334	250	334
2 - AZ	668	500	668
3 - AZ	1 002	750	1 002

Limite totale de nœuds par famille d'instances

Le tableau suivant répertorie le nombre total de nœuds par famille d'instances.

Famille d’instances	ElasticSearch OpenSearch jusqu'à 2,15	OpenSearch 2.17 et versions ultérieures	Limite par défaut
T2	10	10	10
T3	10	10	10
M3, C4, M4, R4, C5, M5, R5, I2, I3	10	200	80
Graviton 2, Graviton 3	200	400	80
C7, R7i, M7i, i4i	200	400	80
OR1.medium.search OR1.large.search OR2.medium.search OR2.large.search OM2.large.search	200	400	80
OR1.xlarge.search et versions ultérieures OR2.xlarge.search et versions ultérieures OM2.xlarge.search et versions ultérieures	200	1 002	80
Ultrawarm 1	150	750	150

Quotas de taille du volume EBS

Le tableau suivant indique les tailles minimale et maximale des volumes EBS pour chaque type d'instance pris en charge par OpenSearch Service. Pour plus d'informations sur les types d'instances

qui incluent le stockage d'instance et des informations supplémentaires sur le matériel, consultez la tarification d'Amazon OpenSearch Service.

- Si vous choisissez le stockage magnétique sous le type de volume EBS lors de la création de votre domaine, la taille maximale du volume est de 100 GiB pour tous les types d'instance sauf t2.small et t2.medium, ainsi que pour toutes les instances Graviton (m6g, C6g, R6g et R6gd), qui ne prennent pas en charge le stockage magnétique. Pour connaître les tailles maximales indiquées dans le tableau suivant, sélectionnez l'une des options SSD.
- Certains types d'instances plus anciens incluent le stockage d'instance, mais prennent également en charge le stockage EBS. Si vous choisissez le stockage EBS pour l'un de ces types d'instances, les volumes de stockage ne s'additionnent pas. Vous pouvez utiliser un volume EBS ou le stockage d'instance, mais pas les deux.

Type d'instance	Taille EBS minimum	Taille EBS maximum (gp2)	Taille EBS maximum (gp3)
t2.micro.search	10 Gio	35 Gio	N/A
t2.small.search	10 Gio	35 Gio	N/A
t2.medium.search	10 Gio	35 Gio	N/A
t3.small.search	10 Gio	100 Gio	100 Gio
t3.medium.search	10 Gio	200 Gio	200 Gio
m3.medium.search	10 Gio	100 Gio	N/A
m3.large.search	10 Gio	512 Gio	N/A
m3.xlarge.search	10 Gio	512 Gio	N/A
m3.2xlarge.search	10 Gio	512 Gio	N/A
m4.large.search	10 Gio	512 Gio	N/A
m4.xlarge.search	10 Gio	1 Tio	N/A

Type d'instance	Taille EBS minimum	Taille EBS maximum (gp2)	Taille EBS maximum (gp3)
m4.2xlarge.search	10 Gio	1,5 Tio	N/A
m4.4xlarge.search	10 Gio	1,5 Tio	N/A
m4.10xlarge.search	10 Gio	1,5 Tio	N/A
m5.large.search	10 Gio	512 Gio	1 Tio
m5.xlarge.search	10 Gio	1 Tio	2 Tio
m5.2xlarge.search	10 Gio	1,5 Tio	3 Tio
m5.4xlarge.search	10 Gio	3 Tio	6 Tio
m5.12xlarge.search	10 Gio	9 Tio	18 Tio
m6g.large.search	10 Gio	512 Gio	1 Tio
m6g.xlarge.search	10 Gio	1 Tio	2 Tio
m6g.2xlarge.search	10 Gio	1,5 Tio	3 Tio
m6g.4xlarge.search	10 Gio	3 Tio	6 Tio
m6g.8xlarge.search	10 Gio	6 Tio	12 Tio
m6g.12xlarge.search	10 Gio	9 Tio	18 Tio
c4.large.search	10 Gio	100 Gio	N/A
c4.xlarge.search	10 Gio	512 Gio	N/A
c4.2xlarge.search	10 Gio	1 Tio	N/A
c4.4xlarge.search	10 Gio	1,5 Tio	N/A
c4.8xlarge.search	10 Gio	1,5 Tio	N/A

Type d'instance	Taille EBS minimum	Taille EBS maximum (gp2)	Taille EBS maximum (gp3)
c5.large.search	10 Gio	256 Gio	256 Gio
c5.xlarge.search	10 Gio	512 Gio	512 Gio
c5.2xlarge.search	10 Gio	1 Tio	1 Tio
c5.4xlarge.search	10 Gio	1,5 Tio	1,5 Tio
c5.9xlarge.search	10 Gio	3,5 Tio	3,5 Tio
c5.18xlarge.search	10 Gio	7 Tio	7 Tio
c6g.large.search	10 Gio	256 Gio	256 Gio
c6g.xlarge.search	10 Gio	512 Gio	512 Gio
c6g.2xlarge.search	10 Gio	1 Tio	1 Tio
c6g.4xlarge.search	10 Gio	1,5 Tio	1,5 Tio
c6g.8xlarge.search	10 Gio	3 Tio	3 Tio
c6g.12xlarge.search	10 Gio	4,5 Tio	4,5 Tio
r3.large.search	10 Gio	512 Gio	N/A
r3.xlarge.search	10 Gio	512 Gio	N/A
r3.2xlarge.search	10 Gio	512 Gio	N/A
r3.4xlarge.search	10 Gio	512 Gio	N/A
r3.8 x large.search	10 Gio	512 Gio	N/A
r4.large.search	10 Gio	1 Tio	N/A
r4.xlarge.search	10 Gio	1,5 Tio	N/A

Type d'instance	Taille EBS minimum	Taille EBS maximum (gp2)	Taille EBS maximum (gp3)
r4.2xlarge.search	10 Gio	1,5 Tio	N/A
r4.4xlarge.search	10 Gio	1,5 Tio	N/A
r4.8 x large.search	10 Gio	1,5 Tio	N/A
r4.16xlarge.search	10 Gio	1,5 Tio	N/A
r5.large.search	10 Gio	1 Tio	2 Tio
r5.xlarge.search	10 Gio	1,5 Tio	3 Tio
r5.2xlarge.search	10 Gio	3 Tio	6 Tio
r5.4xlarge.search	10 Gio	6 Tio	12 Tio
r5.12xlarge.search	10 Gio	12 Tio	24 Tio
r6g.large.search	10 Gio	1 Tio	2 Tio
r6g.xlarge.search	10 Gio	1,5 Tio	3 Tio
r6g.2xlarge.search	10 Gio	3 Tio	6 Tio
r6g.4xlarge.search	10 Gio	6 Tio	12 Tio
r6g.8xlarge.search	10 Gio	8 Tio	16 Tio
r6g.12xlarge.search	10 Gio	12 Tio	24 Tio
r6gd.large.search	N/A	N/A	N/A
r6gd.xlarge.search	N/A	N/A	N/A
r6gd.2xlarge.search	N/A	N/A	N/A
r6gd.4xlarge.search	N/A	N/A	N/A

Type d'instance	Taille EBS minimum	Taille EBS maximum (gp2)	Taille EBS maximum (gp3)
r6gd.8xlarge.search	N/A	N/A	N/A
r6gd.12xlarge.search	N/A	N/A	N/A
r6gd.16xlarge.search	N/A	N/A	N/A
i2.xlarge.search	10 Gio	512 Gio	N/A
i2.2xlarge.search	10 Gio	512 Gio	N/A
i3.large.search	N/A	N/A	N/A
i3.xlarge.search	N/A	N/A	N/A
i3.2xlarge.search	N/A	N/A	N/A
i3.4xlarge.search	N/A	N/A	N/A
i3.8 x large.search	N/A	N/A	N/A
i3.16xlarge.search	N/A	N/A	N/A
ou 1.medium.search	20 GiO	N/A	768 Gio
ou 1.large.search	20 GiO	N/A	1532 GiB
ou 1.xlarge.search	20 GiO	N/A	3 Tio
ou 1,2 x large.search	20 GiO	N/A	6 Tio
ou 1,4 x large.search	20 GiO	N/A	12 Tio
ou 1,8 x large.search	20 GiO	N/A	16 Tio
ou 1,12 x large.search	20 GiO	N/A	24 Tio
ou 1,16 x large.search	20 GiO	N/A	36 TiB

Type d'instance	Taille EBS minimum	Taille EBS maximum (gp2)	Taille EBS maximum (gp3)
ou 2.medium.search	20 GiO	N/A	768 Gio
ou 2.large.search	20 GiO	N/A	1532 GiB
ou 2.xlarge.search	20 GiO	N/A	3 Tio
ou 2,2 x large.search	20 GiO	N/A	6 Tio
ou 2,4 x large.search	20 GiO	N/A	12 Tio
ou 2,8 x large.search	20 GiO	N/A	16 Tio
ou 2,12 x large.search	20 GiO	N/A	24 Tio
ou 2,16 x large.search	20 GiO	N/A	36 TiB
om2.large.search	20 GiO	N/A	768 Gio
om2.xlarge.search	20 GiO	N/A	2 Go
om 2.2xlarge.search	20 GiO	N/A	3 Go
om 2.4xlarge.search	20 GiO	N/A	6 Go
om 2.8xlarge.search	20 GiO	N/A	12 Go
om 2.12xlarge.search	20 GiO	N/A	18 Go
om 2.16xlarge.search	20 GiO	N/A	24 Go
im4gn.large.search	N/A	N/A	N/A
im4gn.xlarge.search	N/A	N/A	N/A
im4gn.2xlarge.search	N/A	N/A	N/A
im4gn.4xlarge.search	N/A	N/A	N/A

Type d'instance	Taille EBS minimum	Taille EBS maximum (gp2)	Taille EBS maximum (gp3)
im4gn.8xlarge.search	N/A	N/A	N/A
im4gn.16xlarge.search	N/A	N/A	N/A
C7G.Large.Rechercher	10 Gio	N/A	256 Gio
C7G.xLarge.Rechercher	10 Gio	N/A	512 Gio
C7G.2XLarge.Rechercher	10 Gio	N/A	1 Tio
C7G.4XLarge.Rechercher	10 Gio	N/A	1,5 Tio
C7G.8XLarge.Rechercher	10 Gio	N/A	3 Tio
C7G. 12 x Large. Rechercher	10 Gio	N/A	4,5 Tio
C7G.16X Large. Rechercher	10 Gio	N/A	6 Tio
m7g.medium.Search	10 Gio	N/A	512 Gio
M7G.Large.Rechercher	10 Gio	N/A	768 Gio
m7g.xlarge.Search	10 Gio	N/A	2 Tio
M7G.2XLarge.Rechercher	10 Gio	N/A	3 Tio
M7G.4XLarge.Rechercher	10 Gio	N/A	6 Tio
M7G.8XLarge.Rechercher	10 Gio	N/A	12 Tio
M7G. 12 x Large. Rechercher	10 Gio	N/A	18 Tio
M7G.16XLarge.Rechercher	10 Gio	N/A	24 Tio
R7G.Medium.Search	10 Gio	N/A	768 Gio
R7G.Large.Rechercher	10 Gio	N/A	1,5 Tio

Type d'instance	Taille EBS minimum	Taille EBS maximum (gp2)	Taille EBS maximum (gp3)
R7G.xLarge.Search	10 Gio	N/A	3 Tio
R7G.2XLarge.Rechercher	10 Gio	N/A	6 Tio
R7G.4XLarge.Rechercher	10 Gio	N/A	12 Tio
R7G.8XLarge.Rechercher	10 Gio	N/A	16 Tio
R7G. 12 x Large. Rechercher	10 Gio	N/A	24 Tio
R7G.16XLarge.Rechercher	10 Gio	N/A	36 TiB
R7GD.Large.Rechercher	N/A	N/A	N/A
r7gd.xlarge.Rechercher	N/A	N/A	N/A
R7GD.2XLarge.Rechercher	N/A	N/A	N/A
R7GD.4XLarge.Rechercher	N/A	N/A	N/A
R7GD.8XLarge.Rechercher	N/A	N/A	N/A
R7GD.12XLarge.Rechercher	N/A	N/A	N/A
R7GD.16XLarge.Rechercher	N/A	N/A	N/A
i4i.large.search	10 Gio	N/A	N/A
i4i.xlarge.search	10 Gio	N/A	N/A
i4i.2xlarge.search	10 Gio	N/A	N/A
i4i.4xlarge.search	10 Gio	N/A	N/A
i4i.8xlarge.search	10 Gio	N/A	N/A
i4i.12xlarge.search	10 Gio	N/A	N/A

Type d'instance	Taille EBS minimum	Taille EBS maximum (gp2)	Taille EBS maximum (gp3)
i4i.16xlarge.search	10 Gio	N/A	N/A
i4i.24xlarge.search	10 Gio	N/A	N/A
i4i.32xlarge.search	10 Gio	N/A	N/A
i4g.large.search	10 Gio	N/A	N/A
i4g.xlarge.search	10 Gio	N/A	N/A
i4g.2xlarge.search	10 Gio	N/A	N/A
i4g.4xlarge.search	10 Gio	N/A	N/A
i4g.8xlarge.search	10 Gio	N/A	N/A
i4g.16xlarge.search	10 Gio	N/A	N/A
c7i.large.search	10 Gio	N/A	256 Gio
c7i.xlarge.search	10 Gio	N/A	512 Gio
c7i.2xlarge.search	10 Gio	N/A	1 Tio
c7i.4xlarge.search	10 Gio	N/A	1,5 Tio
c7i.8xlarge.search	10 Gio	N/A	3 Tio
c7i.12xlarge.search	10 Gio	N/A	4,5 Tio
c7i.16xlarge.search	10 Gio	N/A	6 Tio
m7i.large.search	10 Gio	N/A	768 Gio
m7i.xlarge.search	10 Gio	N/A	2 Tio
m7i.2xlarge.search	10 Gio	N/A	3 Tio

Type d'instance	Taille EBS minimum	Taille EBS maximum (gp2)	Taille EBS maximum (gp3)
m7i.4xlarge.search	10 Gio	N/A	6 Tio
m7i.8xlarge.search	10 Gio	N/A	12 Tio
m7i.12xlarge.search	10 Gio	N/A	18 Tio
m7i.16xlarge.search	10 Gio	N/A	24 Tio
r7i.large.search	10 Gio	N/A	1,5 Tio
r7i.xlarge.search	10 Gio	N/A	3 Tio
r7i.2xlarge.search	10 Gio	N/A	6 Tio
r7i.4xlarge.search	10 Gio	N/A	12 Tio
r7i.8xlarge.search	10 Gio	N/A	16 Tio
r7i.12xlarge.search	10 Gio	N/A	24 Tio
r7i.12xlarge.search	10 Gio	N/A	36 TiB

Quotas de réseau

Le tableau ci-après indique la taille maximum des charges utiles de requête HTTP.

Type d'instance	Taille maximale des charges utiles des requêtes HTTP
t2.micro.search	10 Mio
t2.small.search	10 Mio
t2.medium.search	10 Mio
t3.small.search	10 Mio

Type d'instance	Taille maximale des charges utiles des requêtes HTTP
t3.medium.search	10 Mio
m3.medium.search	10 Mio
m3.large.search	10 Mio
m3.xlarge.search	100 Mio
m3.2xlarge.search	100 Mio
m4.large.search	10 Mio
m4.xlarge.search	100 Mio
m4.2xlarge.search	100 Mio
m4.4xlarge.search	100 Mio
m4.10xlarge.search	100 Mio
m5.large.search	10 Mio
m5.xlarge.search	100 Mio
m5.2xlarge.search	100 Mio
m5.4xlarge.search	100 Mio
m5.12xlarge.search	100 Mio
m6g.large.search	10 Mio
m6g.xlarge.search	100 Mio
m6g.2xlarge.search	100 Mio
m6g.4xlarge.search	100 Mio
m6g.8xlarge.search	100 Mio

Type d'instance	Taille maximale des charges utiles des requêtes HTTP
m6g.12xlarge.search	100 Mio
c4.large.search	10 Mio
c4.xlarge.search	100 Mio
c4.2xlarge.search	100 Mio
c4.4xlarge.search	100 Mio
c4.8xlarge.search	100 Mio
c5.large.search	10 Mio
c5.xlarge.search	100 Mio
c5.2xlarge.search	100 Mio
c5.4xlarge.search	100 Mio
c5.9xlarge.search	100 Mio
c5.18xlarge.search	100 Mio
c6g.large.search	10 Mio
c6g.xlarge.search	100 Mio
c6g.2xlarge.search	100 Mio
c6g.4xlarge.search	100 Mio
c6g.8xlarge.search	100 Mio
c6g.12xlarge.search	100 Mio
r3.large.search	10 Mio
r3.xlarge.search	100 Mio

Type d'instance	Taille maximale des charges utiles des requêtes HTTP
r3.2xlarge.search	100 Mio
r3.4xlarge.search	100 Mio
r3.8 x large.search	100 Mio
r4.large.search	100 Mio
r4.xlarge.search	100 Mio
r4.2xlarge.search	100 Mio
r4.4xlarge.search	100 Mio
r4.8 x large.search	100 Mio
r4.16xlarge.search	100 Mio
r5.large.search	100 Mio
r5.xlarge.search	100 Mio
r5.2xlarge.search	100 Mio
r5.4xlarge.search	100 Mio
r5.12xlarge.search	100 Mio
r6g.large.search	100 Mio
r6g.xlarge.search	100 Mio
r6g.2xlarge.search	100 Mio
r6g.4xlarge.search	100 Mio
r6g.8xlarge.search	100 Mio
r6g.12xlarge.search	100 Mio

Type d'instance	Taille maximale des charges utiles des requêtes HTTP
r6gd.large.search	100 Mio
r6gd.xlarge.search	100 Mio
r6gd.2xlarge.search	100 Mio
r6gd.4xlarge.search	100 Mio
r6gd.8xlarge.search	100 Mio
r6gd.12xlarge.search	100 Mio
r6gd.16xlarge.search	100 Mio
i2.xlarge.search	100 Mio
i2.2xlarge.search	100 Mio
i3.large.search	100 Mio
i3.xlarge.search	100 Mio
i3.2xlarge.search	100 Mio
i3.4xlarge.search	100 Mio
i3.8 x large.search	100 Mio
i3.16xlarge.search	100 Mio
ou 1.medium.search	10 Mio
ou 1.large.search	100 Mio
ou 1.xlarge.search	100 Mio
ou 1,2 x large.search	100 Mio
ou 1,4 x large.search	100 Mio

Type d'instance	Taille maximale des charges utiles des requêtes HTTP
ou 1,8 x large.search	100 Mio
ou 1,12 x large.search	100 Mio
ou 1,16 x large.search	100 Mio
ou 2.medium.search	100 Mio
ou 2.large.search	100 Mio
ou 2.xlarge.search	100 Mio
ou 2,2 x large.search	100 Mio
ou 2,4 x large.search	100 Mio
ou 2,8 x large.search	100 Mio
ou 2,12 x large.search	100 Mio
ou 2,16 x large.search	100 Mio
om2.large.search	100 Mio
om2.xlarge.search	100 Mio
om 2.2xlarge.search	100 Mio
om 2.4xlarge.search	100 Mio
om 2.8xlarge.search	100 Mio
om 2.12xlarge.search	100 Mio
om 2.16xlarge.search	100 Mio
im4gn.large.search	100 Mio
im4gn.xlarge.search	100 Mio

Type d'instance	Taille maximale des charges utiles des requêtes HTTP
im4gn.2xlarge.search	100 Mio
im4gn.4xlarge.search	100 Mio
im4gn.8xlarge.search	100 Mio
im4gn.16xlarge.search	100 Mio
i4i.large.search	100 Mio
i4i.xlarge.search	100 Mio
i4i.2xlarge.search	100 Mio
i4i.4xlarge.search	100 Mio
i4i.8xlarge.search	100 Mio
i4i.12xlarge.search	100 Mio
i4i.16xlarge.search	100 Mio
i4i.24xlarge.search	100 Mio
i4i.32xlarge.search	100 Mio
i4g.large.search	100 Mio
i4g.xlarge.search	100 Mio
i4g.2xlarge.search	100 Mio
i4g.4xlarge.search	100 Mio
i4g.8xlarge.search	100 Mio
i4g.16xlarge.search	100 Mio
c7i.large.search	100 Mio

Type d'instance	Taille maximale des charges utiles des requêtes HTTP
c7i.xlarge.search	100 Mio
c7i.2xlarge.search	100 Mio
c7i.4xlarge.search	100 Mio
c7i.8xlarge.search	100 Mio
c7i.12xlarge.search	100 Mio
c7i.16xlarge.search	100 Mio
m7i.large.search	100 Mio
m7i.xlarge.search	100 Mio
m7i.2xlarge.search	100 Mio
m7i.4xlarge.search	100 Mio
m7i.8xlarge.search	100 Mio
m7i.12xlarge.search	100 Mio
m7i.16xlarge.search	100 Mio
r7i.large.search	100 Mio
r7i.xlarge.search	100 Mio
r7i.2xlarge.search	100 Mio
r7i.4xlarge.search	100 Mio
r7i.8xlarge.search	100 Mio
r7i.12xlarge.search	100 Mio
r7i.16xlarge.search	100 Mio

Quotas de taille de partition

La section suivante répertorie les tailles de partition maximales pour les différentes familles d'instances.

Type d’instance	Multi-AZ sans mode veille	Multi-AZ avec mode veille
R5, C5, M5	N/A	65 GiB
I3	N/A	65 GiB
R6g, C6g, M6g, R6Gd	N/A	65 GiB
OR1, OR2, OM2	100 Gio	65 GiB
Im4gn	N/A	65 GiB

Pour demander une augmentation de quota, contactez le AWS Support.

Quotas de nombre d'unités

La section suivante répertorie le nombre maximal de partitions pour les OpenSearch versions.

Version du moteur	Limite	Remarques
Elasticsearch 1.5 à 6.x	Aucune limite par défaut	
Elasticsearch 7.x	1 000	La limite par défaut peut être modifiée via le paramètre cluster.max_shards_per_node.
OpenSearch 1.x à 2.15	1 000	La limite par défaut peut être modifiée via le paramètre cluster.max_shards_per_node.
OpenSearch 2.17 et versions ultérieures	1 000 par tranche de 16 Go de mémoire, jusqu'à un maximum de 4 000	La limite par défaut ne peut pas être modifiée.

Quota de processus Java

OpenSearch Le service limite les processus Java à une taille de tas de 32 GiB. Les utilisateurs avancés peuvent spécifier le pourcentage de la pile utilisé pour le champ de données. Pour plus d'informations, consultez the section called “Paramètres avancés du cluster” et the section called “JVM OutOfMemoryError”.

Quota de stratégie de domaine

OpenSearch Le service limite les politiques d'accès aux domaines à 100 KiB.

Instances réservées dans Amazon OpenSearch Service

Dans Amazon OpenSearch Service, les instances réservées (RIs) vous permettent de bénéficier de remises importantes par rapport aux instances à la demande standard. Les instances sont identiques ; RIs il s'agit d'une remise de facturation appliquée à des instances à la demande dans votre compte. Pour les applications à longue durée de vie dont l'utilisation est prévisible, cela RIs peut favoriser des économies considérables au fil du temps.

OpenSearch Le service RIs nécessite des durées d'un an ou de trois ans, et propose trois options de paiement qui affectent le taux de remise :

- Sans frais initiaux : vous n'avez rien à payer au départ. Vous payez un taux horaire avec remise pour chaque heure incluse dans la durée définie.
- Frais initiaux partiels : vous payez initialement une partie des frais et vous payez un taux horaire avec remise pour chaque heure incluse dans la durée définie.
- Paiement total anticipé : vous payez l'intégralité des frais de manière anticipé. Vous ne payez pas de taux horaire pour la durée définie.

En règle générale, un plus grand paiement initial induit une plus grande remise. Vous ne pouvez pas annuler les instances réservées (lorsque vous les réservez, vous vous engagez à payer pour la durée complète) et les paiements initiaux ne sont pas remboursables.

RIs ne sont pas flexibles ; ils s'appliquent exclusivement au type d'instance précis que vous réservez. Par exemple, une réservation pour huit instances `c5.2xlarge.search` ne s'applique ni à seize instances `c5.xlarge.search` ni à quatre instance `c5.4xlarge.search`. Toutefois, les comptes associés faisant partie d'une organisation AWS Organizations peuvent bénéficier de toute demande

de discount non utilisée émanant du compte propriétaire du RI, à condition que le type d'instance, la région, la famille et la taille correspondent. Pour plus d'informations, consultez les tarifs et FAQ d'Amazon OpenSearch Service.

Achat d'instances réservées (console)

La console vous permet d'afficher vos instances réservées existantes et d'en acheter de nouvelles.

Pour acheter une réservation

1. Connectez-vous à https://aws.amazon.com puis choisissez Sign In to the Console (Connectez-vous à la console).
2. Sous Analytics, sélectionnez Amazon OpenSearch Service.
3. Choisissez Reserved Instance Leases (Baux d'instance réservée) dans le panneau de navigation.

   Dans cette page, vous pouvez consulter vos réservations existantes. Si vous avez de nombreuses réservations, vous pouvez les filtrer pour identifier et afficher plus facilement une réservation particulière.

   > **Tip**
   >
   > Si le lien Reserved Instance Leases (Baux d'instances réservées) ne s'affiche pas, créez un domaine dans la Région AWS.

4. Choisissez Order Reserved Instance (Commander une instance réservée).
5. Saisissez un nom unique et descriptif.
6. Choisissez un type d'instance et le nombre d'instances. Pour de plus amples informations, consultez the section called “Dimensionnement des domaines”.
7. Choisissez une durée et une option de paiement. Vérifiez attentivement les détails de paiement.
8. Choisissez Next (Suivant).
9. Vérifiez attentivement le récapitulatif d'achat. Les achats d'instances réservées ne sont pas remboursables.
10. Choisissez Order (Commander).

Achat d'instances réservées (AWS CLI)

L' AWS CLI comporte des commandes pour l'affichage des offres, l'achat d'une réservation et l'affichage de vos réservations. La commande suivante et l'exemple de réponse montrent les offres pour une donnée Région AWS :

```
aws opensearch describe-reserved-instance-offerings --region us-east-1
{
  "ReservedInstanceOfferings": [
    {
      "FixedPrice": x,
      "ReservedInstanceOfferingId": "1a2a3a4a5-1a2a-3a4a-5a6a-1a2a3a4a5a6a",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": y,
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "UsagePrice": 0.0,
      "PaymentOption": "PARTIAL_UPFRONT",
      "Duration": 31536000,
      "InstanceType": "m4.2xlarge.search",
      "CurrencyCode": "USD"
    }
  ]
}
```

Pour une explication de chaque valeur renvoyée, consultez le tableau suivant.

Champ	Description
`FixedPrice`	Coût initial de la réservation.
`ReservedInstanceOfferingId`	ID de l'offre. Notez cette valeur si vous souhaitez réserver l'offre.
`RecurringCharges`	Taux horaire de la réservation.
`UsagePrice`	Champ hérité. Pour OpenSearch Service, cette valeur est toujours 0.

Champ	Description
`PaymentOption`	Sans frais initiaux, frais initiaux partiels ou paiement initial complet.
`Duration`	Durée en secondes : • 31536000 secondes équivalent à un an. • 94608000 secondes équivalent à trois ans.
`InstanceType`	Type d'instance pour la réservation. Pour en savoir plus sur les ressources matérielles allouées à chaque type d'instance, consultez Tarification Amazon OpenSearch Service.
`CurrencyCode`	Devise utilisée pour `FixedPrice` et `RecurringChargeAmount` .

L'exemple suivant illustre l'achat d'une réservation :

```
aws opensearch purchase-reserved-instance-offering --reserved-instance-offering-
id 1a2a3a4a5-1a2a-3a4a-5a6a-1a2a3a4a5a6a --reservation-name my-reservation --instance-
count 3 --region us-east-1
{
  "ReservationName": "my-reservation",
  "ReservedInstanceId": "9a8a7a6a-5a4a-3a2a-1a0a-9a8a7a6a5a4a"
}
```

Enfin, vous pouvez répertorier vos réservations pour une région donnée à l'aide de l'exemple suivant :

```
aws opensearch describe-reserved-instances --region us-east-1
{
  "ReservedInstances": [
    {
      "FixedPrice": x,
      "ReservedInstanceOfferingId": "1a2a3a4a5-1a2a-3a4a-5a6a-1a2a3a4a5a6a",
      "ReservationName": "my-reservation",
      "PaymentOption": "PARTIAL_UPFRONT",
      "UsagePrice": 0.0,
```

```
      "ReservedInstanceId": "9a8a7a6a-5a4a-3a2a-1a0a-9a8a7a6a5a4a",
      "RecurringCharges": [
        {
          "RecurringChargeAmount": y,
          "RecurringChargeFrequency": "Hourly"
        }
      ],
      "State": "payment-pending",
      "StartTime": 1522872571.229,
      "InstanceCount": 3,
      "Duration": 31536000,
      "InstanceType": "m4.2xlarge.search",
      "CurrencyCode": "USD"
    }
  ]
}
```

> **Note**
>
> `StartTime` est l'heure Unix, qui correspond au nombre de secondes écoulées depuis le 1er janvier 1970 à minuit, heure UTC. Par exemple, 1522872571 correspond à 20:09:31, heure UTC, le 4 avril 2018. Vous pouvez utiliser des convertisseurs en ligne.

Pour en savoir plus sur les commandes utilisées dans les exemples précédents, consultez Références des commandes AWS CLI.

Achat d'instances réservées (AWS SDKs)

AWS SDKs (sauf Android et iOS SDKs) prennent en charge toutes les opérations qui sont définies dans la Référence de l'API Amazon OpenSearch Service, notamment les opérations suivantes :

- `DescribeReservedInstanceOfferings`
- `PurchaseReservedInstanceOffering`
- `DescribeReservedInstances`

Cet exemple de script utilise le client Python de OpenSearchServicebas niveau du AWS SDK pour Python (Boto3) pour acheter des instances réservées. Vous devez fournir une valeur pour `instance_type`.

```
import boto3
from botocore.config import Config

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default region.

my_config = Config(
    # Optionally lets you specify a region other than your default.
    region_name='us-east-1'
)

client = boto3.client('opensearch', config=my_config)

instance_type = '' # e.g. m4.2xlarge.search


def describe_RI_offerings(client):
    """Gets the Reserved Instance offerings for this account"""

    response = client.describe_reserved_instance_offerings()
    offerings = (response['ReservedInstanceOfferings'])
    return offerings


def check_instance(offering):
    """Returns True if instance type is the one you specified above"""

    if offering['InstanceType'] == instance_type:
        return True

    return False


def get_instance_id():
    """Iterates through the available offerings to find the ID of the one you
 specified"""

    instance_type_iterator = filter(
        check_instance, describe_RI_offerings(client))
    offering = list(instance_type_iterator)
    id = offering[0]['ReservedInstanceOfferingId']
    return id
```

```
def purchase_RI_offering(client):
    """Purchase Reserved Instances"""

    response = client.purchase_reserved_instance_offering(
        ReservedInstanceOfferingId = get_instance_id(),
        ReservationName = 'my-reservation',
        InstanceCount = 1
    )
    print('Purchased reserved instance offering of type ' + instance_type)
    print(response)


def main():
    """Purchase Reserved Instances"""
    purchase_RI_offering(client)
```

Pour plus d'informations sur l'installation et l'utilisation du AWS SDKs, consultez Kits de développement AWS logiciel.

Examen des coûts

Cost Explorer est un outil gratuit que vous pouvez utiliser pour afficher vos données de dépenses des 13 derniers mois. L'analyse de ces données vous aidera à identifier des tendances et à comprendre si cela RIs correspond à votre cas d'utilisation. Si vous l'avez déjà fait RIs, vous pouvez les regrouper par option d'achat et afficher les coûts amortis afin de comparer ces dépenses à celles des instances à la demande. Vous pouvez également définir des budgets d'utilisation pour garantir que vous tirez pleinement parti de vos réservations. Pour plus d'informations, consultez Analyse de vos coûts à l'aide de Cost Explorer dans le Guide de l'utilisateur AWS Billing .

Autres ressources prises en charge dans Amazon OpenSearch Service

Cette rubrique décrit les ressources supplémentaires prises en charge par Amazon OpenSearch Service.

bootstrap.memory_lock

OpenSearch Le service active `bootstrap.memory_lock` l'`opensearch.yml`entrée, qui verrouille la mémoire JVM et empêche le système d'exploitation de la remplacer par un disque. Cela s'applique à tous les types d'instances pris en charge, à l'exception des types suivants :

- `t2.micro.search`
- `t2.small.search`
- `t2.medium.search`
- `t3.small.search`
- `t3.medium.search`

Scripting du module

OpenSearch Le service prend en charge les scripts pour Elasticsearch 5. domaines x et versions ultérieures. Il ne prend pas en charge le scripting pour les versions 1.5 et 2.3.

Les options de scripting prises en charge sont les suivantes :

- Painless
- Expressions Lucene
- Mustache

Pour les domaines Elasticsearch 5.5 et versions ultérieures, ainsi que pour tous les OpenSearch domaines, OpenSearch Service prend en charge les scripts stockés à l'aide du `_scripts` point de terminaison. Les domaines Elasticsearch 5.3 et 5.1 prennent uniquement en charge les scripts en ligne.

Transport TLS

OpenSearch Le service prend en charge le protocole HTTP sur le port 80 et le protocole HTTPS sur le port 443, mais ne prend pas en charge le transport TLS.

Tutoriels Amazon OpenSearch Service

Ce chapitre inclut plusieurs start-to-finish didacticiels pour travailler avec Amazon OpenSearch Service, notamment comment migrer vers le service, créer une application de recherche simple et créer une visualisation dans les OpenSearch tableaux de bord.

Rubriques

- Didacticiel : Création et recherche de documents dans Amazon OpenSearch Service
- Tutoriel : Migration vers Amazon Service OpenSearch
- Tutoriel : Création d'une application de recherche avec Amazon OpenSearch Service
- Tutoriel : Visualisation des appels au service clientèle avec OpenSearch Service et OpenSearch Dashboards

Didacticiel : Création et recherche de documents dans Amazon OpenSearch Service

Dans ce didacticiel, vous apprenez à créer et à rechercher un document dans Amazon OpenSearch Service. Vous ajoutez des données à un index sous la forme d'un document JSON. OpenSearch Service crée un index autour du premier document que vous ajoutez.

Ce didacticiel explique comment effectuer des requêtes HTTP pour créer des documents, générer automatiquement un ID pour un document, et effectuer des recherches de base et avancées sur vos documents.

> **Note**
>
> Ce didacticiel utilise un domaine en accès libre. Pour le plus haut niveau de sécurité, nous vous recommandons de placer votre domaine dans un cloud privé virtuel (VPC).

Prérequis

Ce didacticiel nécessite la configuration suivante :

- Vous devez avoir un Compte AWS.
- Vous devez avoir un domaine OpenSearch Service actif.

Ajout d'un document à un index

Pour ajouter un document à un index, vous pouvez utiliser n'importe quel outil HTTP, comme Postman, cURL ou OpenSearch la console Dashboards. Ces exemples supposent que vous utilisez la console développeur dans OpenSearch Dashboards. Si vous utilisez un autre outil, adaptez-le en conséquence en fournissant l'URL complète et les informations d'identification, si nécessaire.

Pour ajouter un document à un index

1. Accédez à l'URL OpenSearch Dashboards correspondant à votre domaine. Vous pouvez trouver l'URL dans le tableau de bord du domaine dans la console OpenSearch de service. Le format de l'URL est le suivant :

   ```
   domain-endpoint/_dashboards/
   ```

2. Connectez-vous au moyen de votre nom d'utilisateur principal et de votre mot de passe.
3. Ouvrez le panneau de navigation de gauche et choisissez Dev Tools (Outils de développement).
4. Le verbe HTTP permettant de créer une nouvelle ressource est PUT. C'est ce que vous utilisez pour créer un nouveau document et un nouvel index. Saisissez la commande suivante dans la console :

   ```
   PUT fruit/_doc/1
   {
     "name":"strawberry",
     "color":"red"
   }
   ```

   La requête PUT crée un index nommé fruit et ajoute un seul document à l'index avec un ID de 1. Elle produit la réponse suivante :

   ```
   {
     "_index" : "fruit",
     "_type" : "_doc",
     "_id" : "1",
     "_version" : 1,
     "result" : "created",
     "_shards" : {
       "total" : 2,
       "successful" : 2,
       "failed" : 0
   ```

```
  },
  "_seq_no" : 0,
  "_primary_term" : 1
}
```

Création d'ID IDs

OpenSearch Le service peut générer automatiquement un ID pour vos documents. La commande à générer IDs utilise une requête POST au lieu d'une requête PUT, et elle ne requiert aucun ID de document (par rapport à la requête précédente).

Saisissez la requête suivante dans la console du développeur :

```
POST veggies/_doc
{
  "name":"beet",
  "color":"red",
  "classification":"root"
}
```

Cette requête crée un index nommé veggies et ajoute le document à l'index. Elle produit la réponse suivante :

```
{
  "_index" : "veggies",
  "_type" : "_doc",
  "_id" : "3WgyS4IB5DLqbRIvLxtF",
  "_version" : 1,
  "result" : "created",
  "_shards" : {
    "total" : 2,
    "successful" : 2,
    "failed" : 0
  },
  "_seq_no" : 0,
  "_primary_term" : 1
}
```

Notez `_id` le champ supplémentaire dans la réponse, qui indique qu'un ID a été automatiquement créé.

> **Note**
>
> Vous ne fournissez rien après `_doc` dans l'URL, où se trouve normalement l'ID. Comme vous créez un document avec un ID généré, vous n'en fournissez pas encore. C'est réservé aux mises à jour.

Mise à jour d'un document avec une commande POST

Pour mettre à jour un document, vous utilisez une commande HTTP `POST` avec le numéro d'ID.

Tout d'abord, créez un document avec un ID de `42` :

```
POST fruits/_doc/42
{
  "name":"banana",
  "color":"yellow"
}
```

Utilisez ensuite cet ID pour mettre à jour le document :

```
POST fruits/_doc/42
{
  "name":"banana",
  "color":"yellow",
  "classification":"berries"
}
```

Cette commande met à jour le document avec le nouveau champ `classification`. Elle produit la réponse suivante :

```
{
  "_index" : "fruits",
  "_type" : "_doc",
  "_id" : "42",
  "_version" : 2,
  "result" : "updated",
  "_shards" : {
    "total" : 2,
    "successful" : 2,
    "failed" : 0
```

```
  },
  "_seq_no" : 1,
  "_primary_term" : 1
}
```

> **Note**
>
> Si vous essayez de mettre à jour un document qui n'existe pas, OpenSearch Service crée le document.

Exécution d'actions en bloc

Vous pouvez utiliser l'opération API `POST _bulk` pour effectuer plusieurs actions sur un ou plusieurs index en une seule requête. Les commandes d'actions en bloc ont le format suivant :

```
POST /_bulk
<action_meta>\n
<action_data>\n
<action_meta>\n
<action_data>\n
```

Chaque action nécessite deux lignes de JSON. D'abord, vous fournissez la description de l'action ou les métadonnées. Sur la ligne suivante, vous fournissez les données. Chaque partie est séparée par un caractère de saut de ligne (\n). Une description d'action pour un insert pourrait ressembler à ceci :

```
{ "create" : { "_index" : "veggies", "_type" : "_doc", "_id" : "7" } }
```

Et la ligne suivante contenant les données pourrait ressembler à ceci :

```
{ "name":"kale", "color":"green", "classification":"leafy-green" }
```

Prises ensemble, les métadonnées et les données représentent une seule action dans une opération en bloc. Vous pouvez effectuer plusieurs opérations en une seule requête, comme ceci :

```
POST /_bulk
{ "create" : { "_index" : "veggies", "_id" : "35" } }
{ "name":"kale", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "36" } }
```

```
{ "name":"spinach", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "37" } }
{ "name":"arugula", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "38" } }
{ "name":"endive", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "39" } }
{ "name":"lettuce", "color":"green", "classification":"leafy-green" }
{ "delete" : { "_index" : "vegetables", "_id" : "1" } }
```

Notez que la dernière action est un `delete`. Il n'y a pas de données après l'action `delete`.

Recherche de documents

Maintenant que les données existent dans votre cluster, vous pouvez les rechercher. Par exemple, vous pouvez rechercher tous les légumes-racines, obtenir un décompte de tous les légumes-feuilles ou rechercher le nombre d'erreurs journalisées par heure.

Recherches de base

Une recherche de base ressemble à ceci :

```
GET veggies/_search?q=name:l*
```

La requête produit une réponse JSON qui contient le document de laitue.

Recherches avancées

Vous pouvez effectuer des recherches plus avancées en fournissant les options de la requête sous forme de JSON dans le corps de la requête :

```
GET veggies/_search
{
  "query": {
    "term": {
      "name": "lettuce"
    }
  }
}
```

Cet exemple produit également une réponse JSON contenant le document de laitue.

Tri

Vous pouvez effectuer d'autres recherches de ce type en utilisant le tri. Tout d'abord, vous devez recréer l'index, car le mappage automatique des champs a choisi des types qui ne peuvent pas être triés par défaut. Envoyez les requêtes suivantes pour supprimer et recréer l'index :

```
DELETE /veggies

PUT /veggies
{
   "mappings":{
      "properties":{
         "name":{
            "type":"keyword"
         },
         "color":{
            "type":"keyword"
         },
         "classification":{
            "type":"keyword"
         }
      }
   }
}
```

Puis repeuplez l'index avec des données :

```
POST /_bulk
{ "create" : { "_index" : "veggies", "_id" : "7"  } }
{ "name":"kale", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "8" } }
{ "name":"spinach", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "9" } }
{ "name":"arugula", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "10" } }
{ "name":"endive", "color":"green", "classification":"leafy-green" }
{ "create" : { "_index" : "veggies", "_id" : "11" } }
{ "name":"lettuce", "color":"green", "classification":"leafy-green" }
```

Vous pouvez maintenant effectuer une recherche avec un tri. Cette requête ajoute un tri ascendant selon la classification :

```
GET veggies/_search
{
```

```
  "query" : {
    "term": { "color": "green" }
  },
  "sort" : [
      "classification"
  ]
}
```

Ressources connexes

Pour plus d'informations, consultez les ressources suivantes :

- Premiers pas
- Indexation des données
- Recherche de données

Tutoriel : Migration vers Amazon Service OpenSearch

Les instantanés d'index constituent un moyen populaire de migrer d'un cluster Elasticsearch autogéré OpenSearch ou existant vers Amazon Service. OpenSearch Voici les grandes étapes de ce processus :

1. Créez un instantané du cluster existant et chargez-le dans un compartiment Amazon S3.
2. Créez un domaine OpenSearch de service.
3. Donnez au OpenSearch service les autorisations nécessaires pour accéder au compartiment et assurez-vous que vous disposez des autorisations nécessaires pour utiliser les instantanés.
4. Restaurez le snapshot sur le domaine OpenSearch de service.

Cette démonstration fournit des étapes plus détaillées et d'autres options, le cas échéant.

Création et chargement de l'instantané

Bien que vous puissiez utiliser le plugin repository-s3 pour créer des instantanés directement sur S3, vous devez l'installer sur chaque nœud, le modifier `opensearch.yml` (ou `elasticsearch.yml` si vous utilisez un cluster Elasticsearch), redémarrer chaque nœud, ajouter vos AWS informations d'identification et enfin prendre le snapshot. Le plug-in est une excellente option pour une utilisation continue ou pour la migration de clusters plus volumineux.

Pour les clusters plus petits, une approche unique consiste à prendre un instantané du système de fichiers partagé, puis AWS CLI à l'utiliser pour le télécharger sur S3. Si vous avez déjà créé l'instantané, passez directement à l'étape 4.

Pour créer un instantané et le charger sur Amazon S3

1. Ajoutez le paramètre `path.repo` à `opensearch.yml` (ou `Elasticsearch.yml`) sur tous les nœuds, puis redémarrez chaque nœud.

   ```
   path.repo: ["/my/shared/directory/snapshots"]
   ```

2. Enregistrez un référentiel d'instantanés, ce qui est nécessaire avant de prendre un instantané. Un référentiel n'est qu'un emplacement de stockage : un système de fichiers partagé, Amazon S3, un système de fichiers distribué Hadoop (HDFS), etc. Dans ce cas, nous utiliserons un système de fichiers partagé (« fs ») :

   ```
   PUT _snapshot/my-snapshot-repo-name
   {
     "type": "fs",
     "settings": {
       "location": "/my/shared/directory/snapshots"
     }
   }
   ```

3. Créez l'instantané :

   ```
   PUT _snapshot/my-snapshot-repo-name/my-snapshot-name
   {
     "indices": "migration-index1,migration-index2,other-indices-*",
     "include_global_state": false
   }
   ```

4. Installez l'AWS CLI, puis exécutez `aws configure` pour ajouter vos informations d'identification.

5. Accédez au répertoire de l'instantané. Exécutez ensuite les commandes suivantes pour créer un compartiment S3 et charger le contenu du répertoire de l'instantané dans ce compartiment :

   ```
   aws s3 mb s3://amzn-s3-demo-bucket --region us-west-2
   aws s3 sync . s3://amzn-s3-demo-bucket --sse AES256
   ```

Selon la taille de l'instantané et la vitesse de votre connexion Internet, cette opération peut prendre un certain temps.

Création d'un domaine

Bien que la console soit le moyen le plus simple de créer un domaine, dans ce cas, le terminal est déjà ouvert et AWS CLI installé. Modifiez la commande suivante pour créer un domaine qui correspond à vos besoins :

```
aws opensearch create-domain \
  --domain-name migration-domain \
  --engine-version OpenSearch_1.0 \
  --cluster-config InstanceType=c5.large.search,InstanceCount=2 \
  --ebs-options EBSEnabled=true,VolumeType=gp2,VolumeSize=100 \
  --node-to-node-encryption-options Enabled=true \
  --encryption-at-rest-options Enabled=true \
  --domain-endpoint-options EnforceHTTPS=true,TLSSecurityPolicy=Policy-Min-
TLS-1-2-2019-07 \
  --advanced-security-options
 Enabled=true,InternalUserDatabaseEnabled=true,MasterUserOptions='{MasterUserName=master-
user,MasterUserPassword=master-user-password}' \
  --access-policies '{"Version":"2012-10-17","Statement":
[{"Effect":"Allow","Principal":{"AWS":["*"]},"Action":
["es:ESHttp*"],"Resource":"arn:aws:es:us-west-2:123456789012:domain/migration-domain/
*"}]}' \
  --region us-west-2
```

En l'état, la commande crée un domaine accessible à Internet avec deux nœuds de données, chacun avec 100 Gio de stockage. Il permet également un contrôle d'accès précis avec l'authentification de base HTTP et tous les paramètres de chiffrement. Utilisez la console OpenSearch de service si vous avez besoin d'une configuration de sécurité plus avancée, telle qu'un VPC.

Avant d'exécuter la commande, modifiez le nom de domaine, les informations d'identification de l'utilisateur principal et le numéro de compte. Spécifiez la même version Région AWS que celle que vous avez utilisée pour le compartiment S3 et une OpenSearch/Elasticsearch version compatible avec votre instantané.

> **Important**
>
> Les instantanés ne sont compatibles qu'avec la version actuellement installée et les versions supérieures majeures. Par exemple, vous ne pouvez pas restaurer un instantané à partir d'un OpenSearch 1. cluster x sur un Elasticsearch 7. x cluster, seulement un OpenSearch 1. x ou 2. grappe x. La version mineure compte aussi. Vous ne pouvez pas restaurer un instantané à partir d'un cluster 5.3.3 autogéré sur un domaine de service 5.3.2 OpenSearch . Nous vous recommandons de choisir la version la plus récente OpenSearch d'Elasticsearch prise en charge par votre instantané. Pour obtenir un tableau des versions compatibles, consultez the section called “Utilisation d'un instantané pour migrer des données”.

Accordez des autorisations d'accès au compartiment S3.

Dans la console AWS Identity and Access Management (IAM), créez un rôle avec les autorisations et la relation de confiance suivantes. Lors de la création du rôle, choisissez S3 en tant que Service AWS . Nommez le rôle `OpenSearchSnapshotRole` pour qu'il soit facile à identifier.

Autorisations

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
      "Action": [
        "s3:ListBucket"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket"
      ]
    },
    {
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:DeleteObject"
      ],
      "Effect": "Allow",
```

```
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket/*"
      ]
    }
  ]
}
```

Relation d'approbation

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
      "Effect": "Allow",
      "Principal": {
        "Service": "es.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Ensuite, accordez à votre rôle IAM personnel les autorisations de prendre en charge `OpenSearchSnapshotRole`. Créez la stratégie suivante et attachez-la à votre identité :

Autorisations

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::123456789012:role/OpenSearchSnapshotRole"
    }
  ]
}
```

Cartographier le rôle des instantanés dans OpenSearch les tableaux de bord (si vous utilisez un contrôle d'accès précis)

Si vous avez activé le contrôle d'accès précis, même si vous utilisez l'authentification de base HTTP à toutes les autres fins, vous devez mapper le rôle `manage_snapshots` à votre rôle IAM afin d'utiliser des instantanés.

Pour accorder à votre identité les autorisations nécessaires pour utiliser des instantanés

1. Connectez-vous aux tableaux de bord à l'aide des informations d'identification de l'utilisateur principal que vous avez spécifiées lors de la création du domaine OpenSearch de service. Vous trouverez l'URL des tableaux de bord dans la console de OpenSearch service. Elle prend la forme suivante : `https://domain-endpoint/_dashboards/`.
2. Dans le menu principal, choisissez Sécurité, Rôles, puis sélectionnez le rôle manage_snapshots.
3. Choisissez Mapped users (Utilisateurs mappés), Manage mapping (Gérer le mappage).
4. Ajoutez l'ARN de domaine de votre rôle IAM personnel dans le champ approprié. L'ARN se présente dans l'un des formats suivants :

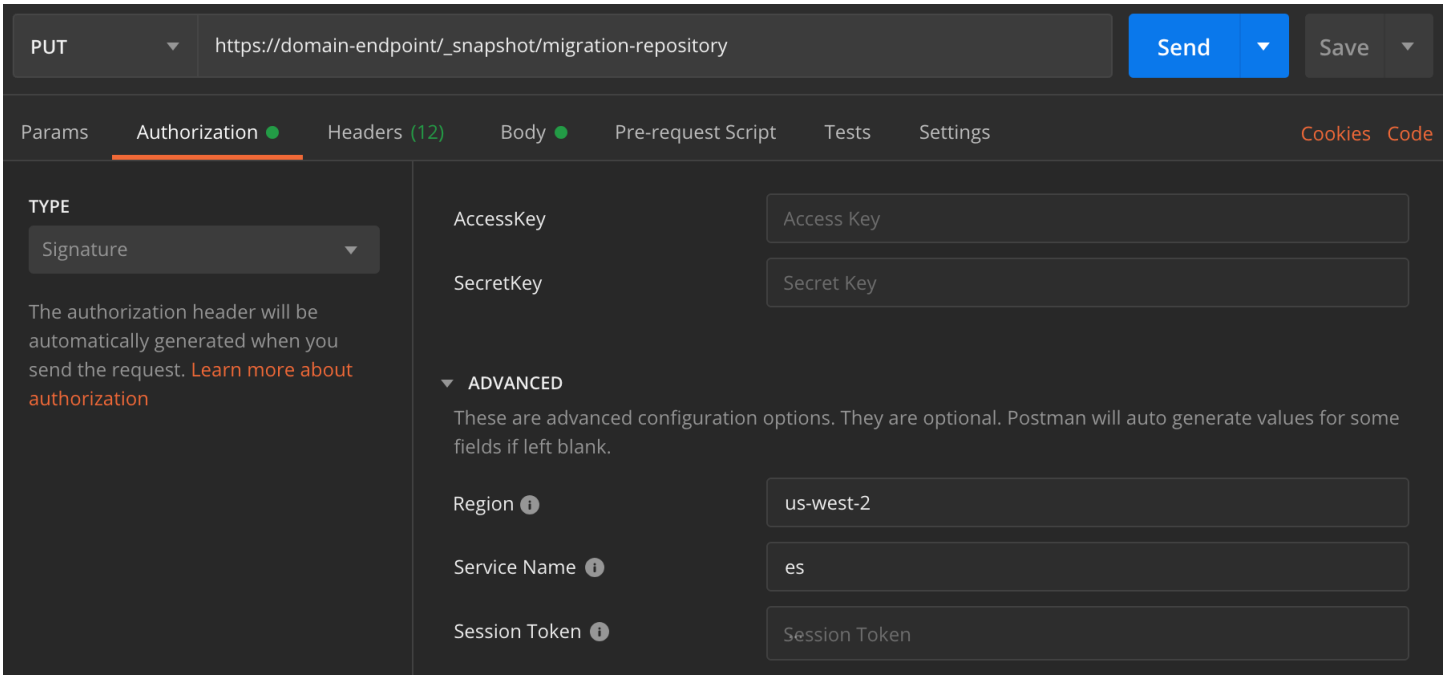
   ```
   arn:aws:iam::123456789123:user/user-name
   ```

   ```
   arn:aws:iam::123456789123:role/role-name
   ```
5. Sélectionnez Map (Mapper) et vérifiez que le rôle s'affiche sous Mapped users (Utilisateurs mappés).

Restaurer l'instantané.

À ce stade, vous pouvez accéder à votre domaine de OpenSearch service de deux manières : l'authentification HTTP de base avec vos informations d'identification d'utilisateur principal ou l' AWS authentification à l'aide de vos informations d'identification IAM. Étant donné que les instantanés utilisent Amazon S3, qui n'a aucune notion d'utilisateur principal, vous devez utiliser vos informations d'identification IAM pour enregistrer le référentiel de clichés auprès de votre domaine de OpenSearch service.

La plupart des langages de programmation disposent de bibliothèques pour faciliter la signature des demandes, mais l'approche la plus simple consiste à utiliser un outil tel que Postman et à saisir vos informations d'identification IAM dans la section Autorisation.



Pour restaurer l'instantané

1. Quelle que soit la manière dont vous choisissez de signer vos demandes, la première étape consiste à enregistrer le référentiel :

```
PUT _snapshot/my-snapshot-repo-name
{
  "type": "s3",
  "settings": {
    "bucket": "amzn-s3-demo-bucket",
    "region": "us-west-2",
    "role_arn": "arn:aws:iam::123456789012:role/OpenSearchSnapshotRole"
  }
}
```

2. Ensuite, répertoriez les instantanés dans le référentiel et identifiez celui que vous souhaitez restaurer. À ce stade, vous pouvez continuer à utiliser Postman ou opter pour un outil comme curl.

   Shorthand

   ```
   GET _snapshot/my-snapshot-repo-name/_all
   ```

   curl

```
curl -XGET -u 'master-user:master-user-password' https://domain-endpoint/
_snapshot/my-snapshot-repo-name/_all
```

3. Restaurez l'instantané.

   Shorthand

   ```
   POST _snapshot/my-snapshot-repo-name/my-snapshot-name/_restore
   {
     "indices": "migration-index1,migration-index2,other-indices-*",
     "include_global_state": false
   }
   ```

   curl

   ```
   curl -XPOST -u 'master-user:master-user-password' https://domain-endpoint/
   _snapshot/my-snapshot-repo-name/my-snapshot-name/_restore \
     -H 'Content-Type: application/json' \
     -d '{"indices":"migration-index1,migration-index2,other-indices-
   *","include_global_state":false}'
   ```

4. Enfin, vérifiez que vos index sont restaurés comme prévu.

   Shorthand

   ```
   GET _cat/indices?v
   ```

   curl

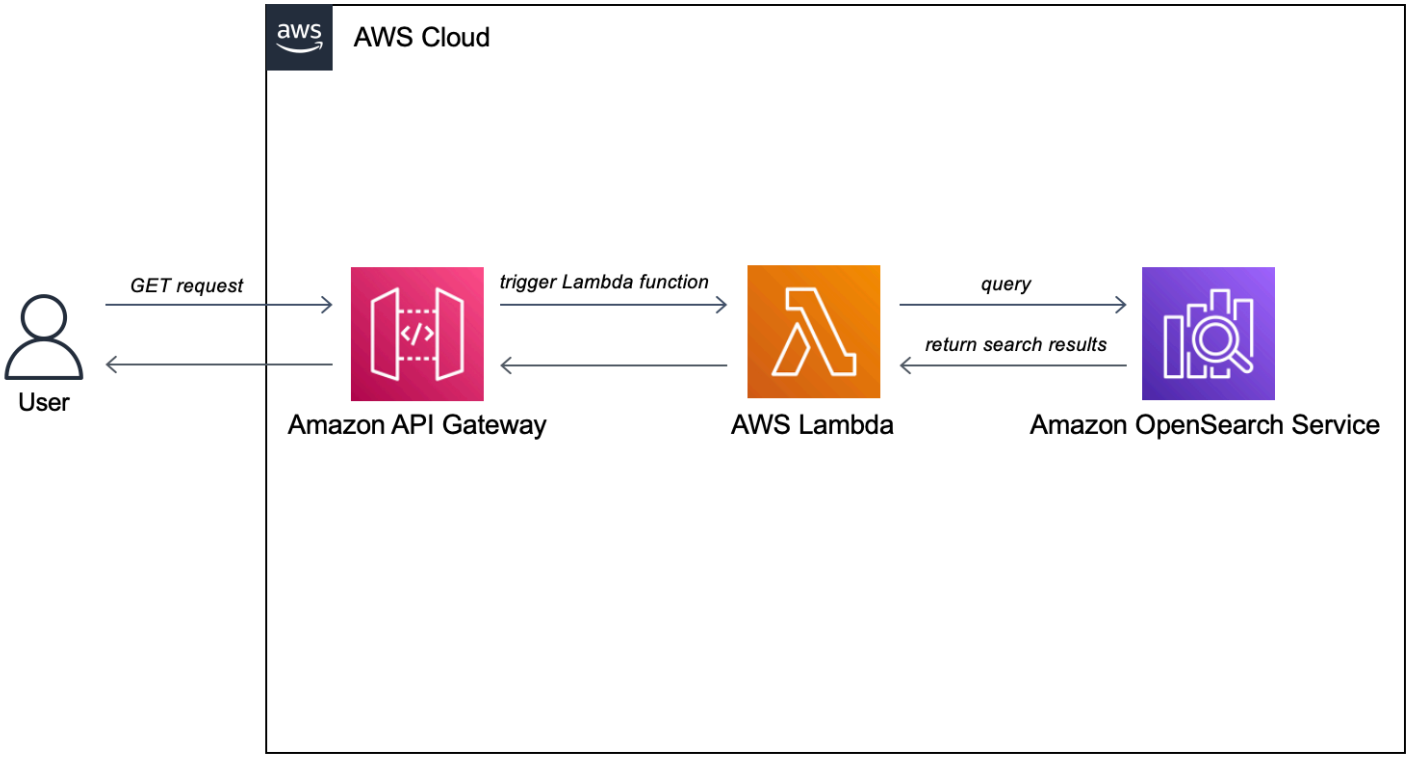
   ```
   curl -XGET -u 'master-user:master-user-password' https://domain-endpoint/_cat/
   indices?v
   ```

À ce stade, la migration est terminée. Vous pouvez configurer vos clients pour qu'ils utilisent le nouveau point de terminaison du OpenSearch service, redimensionner le domaine en fonction de votre charge de travail, vérifier le nombre de partitions pour vos index, passer à un utilisateur principal IAM ou commencer à créer des visualisations dans les tableaux de bord. OpenSearch

Tutoriel : Création d'une application de recherche avec Amazon OpenSearch Service

Une méthode courante pour créer une application de recherche avec Amazon OpenSearch Service consiste à utiliser des formulaires Web pour envoyer les requêtes des utilisateurs à un serveur. Vous pouvez ensuite autoriser le serveur à les appeler OpenSearch APIs directement et demander au serveur d'envoyer des demandes au OpenSearch Service. Toutefois, si vous souhaitez écrire du code côté client qui ne repose sur aucun serveur, vous devez compenser les risques de sécurité et de performances. OpenSearch APIs Il est déconseillé d'autoriser l'accès public non signé au. Les utilisateurs risquent d'accéder à des points de terminaison non sécurisés, ou d'impacter les performances du cluster via des requêtes trop étendues (ou trop nombreuses).

Ce chapitre présente une solution : utilisez Amazon API Gateway pour limiter les utilisateurs à un sous-ensemble de l'API Gateway AWS Lambda to Service OpenSearch APIs et pour signer les demandes émanant d'API Gateway to OpenSearch Service.



> **Note**
>
> La tarification API Gateway et Lambda standard s'applique, mais les coûts devraient être négligeables dans le cadre de l'utilisation limitée de ce tutoriel.

Prérequis

La condition préalable à ce didacticiel est un domaine OpenSearch de service. Si vous n'en avez pas déjà un, suivez les étapes décrites dans Créer un domaine OpenSearch de service pour en créer un.

Étape 1 : Indexer des exemples de données

Téléchargez sample-movies.zip, décompressez-le et utilisez l'opération d'API _bulk pour ajouter les 5 000 documents à l'index `movies` :

```
POST https://search-my-domain.us-west-1.es.amazonaws.com/_bulk
{ "index": { "_index": "movies", "_id": "tt1979320" } }
{"directors":["Ron
 Howard"],"release_date":"2013-09-02T00:00:00Z","rating":8.3,"genres":
["Action","Biography","Drama","Sport"],"image_url":"http://ia.media-imdb.com/images/
M/MV5BMTQyMDE0MTY0OV5BMl5BanBnXkFtZTcwMjI2OTI0OQ@@._V1_SX400_.jpg","plot":"A re-
creation of the merciless 1970s rivalry between Formula One rivals James Hunt and
 Niki Lauda.","title":"Rush","rank":2,"running_time_secs":7380,"actors":["Daniel
 Brühl","Chris Hemsworth","Olivia Wilde"],"year":2013,"id":"tt1979320","type":"add"}
{ "index": { "_index": "movies", "_id": "tt1951264" } }
{"directors":["Francis Lawrence"],"release_date":"2013-11-11T00:00:00Z","genres":
["Action","Adventure","Sci-Fi","Thriller"],"image_url":"http://ia.media-imdb.com/
images/M/
MV5BMTAyMjQ3OTAxMzNeQTJeQWpwZ15BbWU4MDU0NzA1MzAx._V1_SX400_.jpg","plot":"Katniss
 Everdeen and Peeta Mellark become targets of the Capitol after
 their victory in the 74th Hunger Games sparks a rebellion in
 the Districts of Panem.","title":"The Hunger Games: Catching
 Fire","rank":4,"running_time_secs":8760,"actors":["Jennifer Lawrence","Josh
 Hutcherson","Liam Hemsworth"],"year":2013,"id":"tt1951264","type":"add"}
...
```

Notez que ce qui précède est un exemple de commande avec un petit sous-ensemble des données disponibles. Pour effectuer l'`_bulk`opération, vous devez copier et coller l'intégralité du contenu du `sample-movies` fichier. Pour de plus amples instructions, voirthe section called “Option 2 : Charger plusieurs documents”.

Vous pouvez également utiliser la commande curl suivante pour obtenir le même résultat :

```
curl -XPOST -u 'master-user:master-user-password' 'domain-endpoint/_bulk' --data-binary
 @bulk_movies.json -H 'Content-Type: application/json'
```

Étape 2 : Création et déploiement de la fonction Lambda

Avant de créer votre API dans API Gateway, créez la fonction Lambda à laquelle elle transmet les demandes.

Créer la fonction Lambda

Dans cette solution, API Gateway transmet les requêtes à une fonction Lambda, qui interroge le OpenSearch service et renvoie les résultats. Comme cet exemple de fonction utilise des bibliothèques externes, vous devez créer un package de déploiement et le télécharger sur Lambda.

Pour créer le package de déploiement

1. Ouvrez une invite de commandes et créez un répertoire de projet `my-opensearch-function`. Par exemple, sur macOS :

   ```
   mkdir my-opensearch-function
   ```

2. Accédez au répertoire du projet `my-sourcecode-function`.

   ```
   cd my-opensearch-function
   ```

3. Copiez le contenu de l'exemple de code Python suivant et enregistrez-le dans un nouveau fichier nomméopensearch-lambda.py. Ajoutez votre région et votre point de terminaison hôte au fichier.

   ```
   import boto3
   import json
   import requests
   from requests_aws4auth import AWS4Auth

   region = '' # For example, us-west-1
   service = 'es'
   credentials = boto3.Session().get_credentials()
   awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
    session_token=credentials.token)
   ```

```
host = '' # The OpenSearch domain endpoint with https:// and without a trailing
 slash
index = 'movies'
url = host + '/' + index + '/_search'

# Lambda execution starts here
def lambda_handler(event, context):

    # Put the user query into the query DSL for more accurate search results.
    # Note that certain fields are boosted (^).
    query = {
        "size": 25,
        "query": {
            "multi_match": {
                "query": event['queryStringParameters']['q'],
                "fields": ["title^4", "plot^2", "actors", "directors"]
            }
        }
    }

    # Elasticsearch 6.x requires an explicit Content-Type header
    headers = { "Content-Type": "application/json" }

    # Make the signed HTTP request
    r = requests.get(url, auth=awsauth, headers=headers, data=json.dumps(query))

    # Create the response and add some extra content to support CORS
    response = {
        "statusCode": 200,
        "headers": {
            "Access-Control-Allow-Origin": '*'
        },
        "isBase64Encoded": False
    }

    # Add the search results to the response
    response['body'] = r.text
    return response
```

4. Installez les bibliothèques externes dans un nouveau `package` répertoire.

```
pip3 install --target ./package boto3
pip3 install --target ./package requests
```

```
pip3 install --target ./package requests_aws4auth
```

5. Créez un package de déploiement avec la bibliothèque installée à la racine. La commande suivante génère un `my-deployment-package.zip` fichier dans le répertoire de votre projet.

```
cd package
zip -r ../my-deployment-package.zip .
```

6. Ajoutez le fichier `opensearch-lambda.py` à la racine du fichier .zip.

```
cd ..
zip my-deployment-package.zip opensearch-lambda.py
```

Pour plus d'informations sur la création de fonctions Lambda et des packages de déploiement, consultez Déployer des fonctions Lambda Python avec des archives de fichiers .zip dans le Guide du développeur AWS Lambda et the section called “Créer le package de déploiement Lambda” dans ce guide.

Pour créer votre fonction à l'aide de la console Lambda

1. Accédez à la console Lambda à https://console.aws.amazon.com/lambda/ la maison. Dans le volet de navigation de gauche, sélectionnez Functions.
2. Sélectionnez Create function (Créer une fonction).
3. Configurez les champs suivants :

   - Nom de la fonction : opensearch-function
   - Temps d'exécution : Python 3.9
   - Architecture : x86_64

   Conservez toutes les autres options par défaut et choisissez Créer une fonction.
4. Dans la section Source du code de la page de résumé des fonctions, choisissez le menu déroulant Télécharger depuis et sélectionnez le fichier .zip. Localisez le `my-deployment-package.zip` fichier que vous avez créé et choisissez Enregistrer.
5. Le gestionnaire est la méthode de votre code de fonction qui traite les événements. Sous Paramètres d'exécution, choisissez Modifier et modifiez le nom du gestionnaire en fonction du nom du fichier dans votre package de déploiement où se trouve la fonction Lambda. Puisque votre fichier est nomméopensearch-lambda.py, renommez le gestionnaire en. *opensearch-*

lambda.lambda_handler Pour plus d'informations, consultez Gestionnaire de fonctions Lambda dans Python.

Étape 3 : Création de l'API dans API Gateway

L'utilisation d'API Gateway vous permet de créer une API plus limitée et de simplifier le processus d'interaction avec l' OpenSearch _searchAPI. API Gateway vous permet également d'activer des fonctions de sécurité, telles que l'authentification Amazon Cognito et la limitation des demandes. Effectuez les étapes suivantes pour créer et déployer une API :

Créer et configurer l'API

Pour créer votre API à l'aide de la console API Gateway

1. Accédez à la console API Gateway à l'https://console.aws.amazon.com/apigateway/accueil. Dans le volet de navigation de gauche, choisissez APIs.
2. Recherchez API REST (non privée) et choisissez Créer.
3. Sur la page suivante, recherchez la section Créer une nouvelle API et assurez-vous que l'option Nouvelle API est sélectionnée.
4. Configurez les champs suivants :
   - Nom d'API :opensearch-api
   - Description : API publique pour rechercher un domaine Amazon OpenSearch Service
   - Type de point de terminaison : régional
5. Sélectionnez Create API (Créer une API).
6. Choisissez Actions (Actions) et Create Method (Créer une méthode).
7. Choisissez GET dans la liste déroulante et cliquez sur la coche pour confirmer.
8. Configurez les paramètres suivants, puis cliquez sur Save (Enregistrer) :

Paramètre	Value
Type d’intégration	fonction Lambda
Utiliser une intégration proxy Lambda	Oui

Paramètre	Value
Région Lambda	*us-west-1*
fonction Lambda	opensearch-lambda
Utiliser le délai d'attente par défaut	Oui

Configurer la demande de méthode

Choisissez Demande de méthode et configurez les paramètres suivants :

Paramètre	Value
Autorisation	NONE
Validateur de demande	Valider les paramètres et en-têtes des chaînes de requête
Clé d'API requise	false

Sous Paramètres de chaîne de requête URL, choisissez Ajouter une chaîne de requête et configurez le paramètre suivant :

Paramètre	Value
Nom	q
Obligatoire	Oui

Déployer l'API et configurer une étape

La console API Gateway vous permet de déployer une API en créant un déploiement et en l'associant à une étape nouvelle ou existante.

1. Choisissez Actions (Actions) et Deploy API (Déployer l'API).

2. Pour Étape de déploiement, choisissez Nouvelle étape et nommez l'étape `opensearch-api-test`.
3. Choisissez Déployer.
4. Configurez les paramètres suivants dans l'éditeur d'étape, puis choisissez Enregistrer les modifications :

Paramètre	Value
Activer les limitations	Oui
Vitesse	1 000
Mode rafale	500

Ces paramètres permettent de configurer une API qui n'a qu'une méthode : une demande `GET` au point de terminaison racine (`https://some-id.execute-api.us-west-1.amazonaws.com/search-es-api-test`). La demande nécessite un paramètre unique (q), la chaîne de requête à rechercher. Lorsqu'elle est appelée, la méthode transmet la demande à Lambda, qui exécute la fonction `opensearch-lambda`. Pour plus d'informations, consultez Création d'une API dans Amazon API Gateway et Déploiement d'une API REST dans Amazon API Gateway.

Étape 4 : (Facultatif) modifier la stratégie d'accès au domaine

Votre domaine OpenSearch de service doit autoriser la fonction Lambda à envoyer des `GET` requêtes à l'`movies`index. Si votre domaine dispose d'une stratégie d'accès libre où le contrôle détaillé de l'accès est activé, vous pouvez le laisser tel quel :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
```

```
      "Action": "es:*",
      "Resource": "arn:aws:es:us-west-1:123456789012:domain/domain-name/*"
    }
  ]
}
```

Vous pouvez également choisir de rendre votre stratégie d'accès au domaine plus détaillée. Par exemple, la politique minimale suivante fournit un accès en lecture `opensearch-lambda-role` (créé par Lambda) à l'index `movies`. Pour obtenir le nom exact du rôle créé automatiquement par Lambda, accédez à la console AWS Identity and Access Management (IAM), choisissez Rôles et recherchez « lambda ».

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:role/service-role/opensearch-lambda-role-1abcdefg"
      },
      "Action": "es:ESHttpGet",
      "Resource": "arn:aws:es:us-west-1:123456789012:domain/domain-name/movies/_search"
    }
  ]
}
```

Important

Si le contrôle d'accès détaillé est activé pour le domaine, vous devez également associer le rôle à un utilisateur dans les OpenSearch tableaux de bord, sinon vous verrez des erreurs d'autorisation.

Configurer les autorisations du rôle d'exécution Lambda

Outre la configuration de la politique d'accès au domaine, vous devez également vous assurer que le rôle d'exécution Lambda dispose des autorisations IAM nécessaires pour accéder à votre OpenSearch domaine de service. La fonction Lambda nécessite des autorisations spécifiques selon que vous utilisez un domaine géré ou une collection OpenSearch Service Serverless.

Pour les domaines OpenSearch de services gérés :

Associez la politique IAM suivante à votre rôle d'exécution Lambda pour lui permettre d'envoyer des demandes à OpenSearch votre domaine de service :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "es:ESHttpGet",
        "es:ESHttpPost"
      ],
      "Resource": "arn:aws:es:us-west-1:123456789012:domain/domain-name/*"
    }
  ]
}
```

Pour les collections OpenSearch Service Serverless :

Si vous utilisez OpenSearch Service Serverless, associez la politique IAM suivante à votre rôle d'exécution Lambda :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

```
      "Action": "aoss:*",
      "Resource": "arn:aws:aoss:us-west-1:123456789012:collection/collection-id"
    }
  ]
}
```

Pour associer ces politiques à votre rôle d'exécution Lambda :

1. Accédez à la console IAM à https://console.aws.amazon.com/iam/l'adresse.
2. Choisissez Rôles et recherchez votre rôle d'exécution Lambda (généralement nomméopensearch-lambda-role-*xxxxxxxx*).
3. Choisissez Ajouter des autorisations, puis Créer une politique intégrée.
4. Choisissez l'onglet JSON et collez la politique appropriée ci-dessus, en remplaçant les valeurs d'espace réservé par votre ressource ARNs réelle.
5. Choisissez Réviser la politique, saisissez un nom tel que`OpenSearchAccess`, puis choisissez Créer une politique.

> **Note**
>
> Sans ces autorisations IAM, votre fonction Lambda recevra des erreurs « Accès refusé » lorsqu'elle tentera d'interroger OpenSearch votre domaine de service, même si la politique d'accès au domaine autorise les demandes.

Pour plus d'informations sur les stratégies d'accès, consultez the section called “Configuration des politiques d'accès”.

Mapper le rôle Lambda (si vous utilisez le contrôle d'accès précis)

Le contrôle d'accès précis introduit une étape supplémentaire avant de pouvoir tester l'application. Même si vous utilisez l'authentification de base HTTP à d'autres fins, vous devez mapper le rôle Lambda à l'utilisateur, sinon vous verrez des erreurs d'autorisation.

1. Accédez à l'URL OpenSearch des tableaux de bord du domaine.
2. Dans le menu principal, choisissez Sécurité, Rôles, puis sélectionnez le lien vers `all_access` le rôle auquel vous devez associer le rôle Lambda.

3. Choisissez Mapped users (Utilisateurs mappés), Manage mapping (Gérer le mappage).
4. Sous Backend roles (Rôles backend), ajoutez l'Amazon Resource Name (ARN) du rôle Lambda. L'ARN doit prendre la forme `dearn:aws:iam::123456789123:role/service-role/opensearch-lambda-role-1abcdefg`.
5. Sélectionnez Mapper et vérifiez que l'utilisateur ou le rôle s'affiche sous Utilisateurs mappés.

Étape 5 : Tester l'application web

Pour tester l'application web

1. Téléchargez sample-site.zip, décompressez-le et ouvrez `scripts/search.js` dans votre éditeur de texte favori.
2. Mettez à jour la `apigatewayendpoint` variable pour qu'elle pointe vers votre point de terminaison API Gateway et ajoutez une barre oblique inverse à la fin du chemin indiqué. Vous pouvez rapidement trouver le point de terminaison dans API Gateway en choisissant Stages (Étapes) et en sélectionnant le nom de l'API. La `apigatewayendpoint` variable doit prendre la forme `https://some-id.execute-api.us-west-1.amazonaws.com/opensearch-api-test` /.
3. Ouvrez `index.html` et essayez d'effectuer des recherches pour thor, house et d'autres termes.

Movie Search

thor

Found 7 results.



Thor

2011 — The powerful but arrogant god Thor is cast out of Asgard to live amongst humans in Midgard (Earth), where he soon becomes one of their finest defenders.



Thor: The Dark World

2013 — Faced with an enemy that even Odin and Asgard cannot withstand, Thor must embark on his most perilous and personal journey yet, one that will reunite him with Jane Foster and force him to sacrifice everything to save us all.



Vikingdom

2013 — A forgotten king, Eirick, is tasked with the impossible odds to defeat Thor, the God of Thunder.

Dépannage des erreurs CORS

Même si la fonction Lambda inclut du contenu dans la réponse à la prise en charge de CORS, l'erreur suivante peut toujours s'afficher :

```
Access to XMLHttpRequest at '<api-gateway-endpoint>' from origin 'null' has been
 blocked by CORS policy: No 'Access-Control-Allow-Origin' header is present in the
 requested resource.
```

Si c'est le cas, essayez ce qui suit :

1. Activez CORS sur la ressource GET. Sous Advanced (Avancé), configurez Access-Control-Allow-Credentials comme `'true'`.
2. Redéployez votre API dans API Gateway (Actions, Deploy API) (Actions, Déploiement de l'API).
3. Supprimez et ajoutez à nouveau votre déclencheur de fonction Lambda. Ajoutez-le à nouveau, choisissez Ajouter un déclencheur et créez le point de terminaison HTTP qui invoque votre fonction. Le déclencheur doit présenter la configuration suivante :

Déclencheur	« Hello, World! »	Étape de déploiement	Sécurité
API Gateway	opensearch-api	opensearch-api-test	Ouvrir

Étapes suivantes

Ce chapitre n'est qu'un point de départ pour illustrer un concept. Vous pouvez envisager les modifications suivantes :

- Ajoutez vos propres données au domaine OpenSearch de service.
- Ajouter des méthodes à votre API.
- Dans la fonction Lambda, modifier la requête de recherche ou optimiser différents champs.
- Appliquer des styles différents aux résultats ou modifier `search.js` afin d'afficher différents champs pour l'utilisateur.

Tutoriel : Visualisation des appels au service clientèle avec OpenSearch Service et OpenSearch Dashboards

Ce chapitre est une procédure détaillée de la situation suivante : une entreprise reçoit un certain nombre d'appels au support client et les analyse. Quel est l'objet de chaque appel ? Combien étaient positifs ? Combien étaient négatifs ? Comment les gestionnaires peuvent-ils chercher ou revoir les transcriptions de ces appels ?

Un flux de travail manuel peut impliquer des employés qui écoutent des enregistrements, notant l'objet de chaque appel et décidant si l'interaction avec le client était positive ou non.

Ce processus serait extrêmement fastidieux. En supposant un temps moyen de 10 minutes par appel, chaque employé pourrait écouter seulement 48 appels par jour. À moins d'un biais humain, les données générées seraient très précises, mais la quantité de données serait minime : simplement l'objet de l'appel et une valeur booléenne pour savoir si le client a été satisfait. Tout ce qui est plus complexe, par exemple une transcription complète, prendrait beaucoup de temps.

À l'aide d'Amazon S3, Amazon Transcribe, Amazon Comprehend et OpenSearch Amazon Service, vous pouvez automatiser un processus similaire avec très peu de code et obtenir ainsi beaucoup plus de données. Par exemple, vous pouvez obtenir une transcription complète de l'appel, des mots-clés de la transcription et un « sentiment » général de l'appel (positif, négatif, neutre ou mixte). Ensuite, vous pouvez utiliser OpenSearch et OpenSearch Dashboards pour rechercher et visualiser les données.

Bien que vous puissiez utiliser cette procédure en l'état, l'objectif est de susciter des idées sur la façon d'enrichir vos documents JSON avant de les indexer dans OpenSearch Service.

Coûts estimés

En général, l'exécution des étapes de cette procédure devrait coûter moins de 2 USD. La procédure utilise les ressources suivantes :

- Compartiment S3 avec moins de 100 Mo transférés et stockés

  Pour en savoir plus, consultez la page relative à la Tarification Amazon S3.
- OpenSearch Domaine de service avec une `t2.medium` instance et 10 Gio de stockage EBS pendant plusieurs heures

  Pour en savoir plus, consultez Tarification Amazon OpenSearch Service.

- Plusieurs appels à Amazon Transcribe

  Pour en savoir plus, consultez Tarification Amazon Transcribe.
- Plusieurs appels de traitement du langage naturel à Amazon Comprehend

  Pour en savoir plus, consultez Tarification Amazon Comprehend.

Rubriques

- Étape 1 : Configurer les prérequis
- Étape 2 : Copier un exemple de code
- (Facultatif) Étape 3 : Indexer des exemples de données
- Étape 4 : Analyser et visualiser vos données
- Étape 5 : Nettoyage des ressources et étapes suivantes

Étape 1 : Configurer les prérequis

Avant de poursuivre, vous devez disposer des ressources suivantes.

Prérequis	Description
Compartiment Amazon S3	Pour de plus amples informations, veuillez consulter Création d'un compartiment dans le Guide de l'utilisateur d'Amazon Simple Storage Service.
OpenSearch Domaine de service	La destination des données. Pour plus d'informations, consultez la section Création OpenSearch de domaines de service.

Si vous ne disposez pas déjà de ces ressources, vous pouvez les créer à l'aide des commandes AWS CLI suivantes :

```
aws s3 mb s3://my-transcribe-test --region us-west-2
```

```
aws opensearch create-domain --domain-name my-transcribe-test --engine-version
 OpenSearch_1.0 --cluster-config  InstanceType=t2.medium.search,InstanceCount=1
 --ebs-options EBSEnabled=true,VolumeType=standard,VolumeSize=10 --access-
```

```
policies '{"Version":"2012-10-17","Statement":[{"Effect":"Allow","Principal":
{"AWS":"arn:aws:iam::123456789012:root"},"Action":"es:*","Resource":"arn:aws:es:us-
west-2:123456789012:domain/my-transcribe-test/*"}]}' --region us-west-2
```

> **Note**
>
> Ces commandes utilisent la région `us-west-2`, mais vous pouvez utiliser n'importe quelle région prise en charge par Amazon Comprehend. Pour en savoir plus, consultez Références générales AWS.

Étape 2 : Copier un exemple de code

1. Copiez et collez l'exemple de code Python 3 suivant dans un nouveau fichier nommé `call-center.py` :

```
import boto3
import datetime
import json
import requests
from requests_aws4auth import AWS4Auth
import time
import urllib.request

# Variables to update
audio_file_name = '' # For example, 000001.mp3
bucket_name = '' # For example, my-transcribe-test
domain = '' # For example, https://search-my-transcribe-test-12345.us-
west-2.es.amazonaws.com
index = 'support-calls'
type = '_doc'
region = 'us-west-2'

# Upload audio file to S3.
s3_client = boto3.client('s3')

audio_file = open(audio_file_name, 'rb')

print('Uploading ' + audio_file_name + '...')
response = s3_client.put_object(
    Body=audio_file,
```

```
    Bucket=bucket_name,
    Key=audio_file_name
)

# # Build the URL to the audio file on S3.
# # Only for the us-east-1 region.
# mp3_uri = 'https://' + bucket_name + '.s3.amazonaws.com/' + audio_file_name

# Get the necessary details and build the URL to the audio file on S3.
# For all other regions.
response = s3_client.get_bucket_location(
    Bucket=bucket_name
)
bucket_region = response['LocationConstraint']
mp3_uri = 'https://' + bucket_name + '.s3-' + bucket_region + '.amazonaws.com/' +
 audio_file_name

# Start transcription job.
transcribe_client = boto3.client('transcribe')

print('Starting transcription job...')
response = transcribe_client.start_transcription_job(
    TranscriptionJobName=audio_file_name,
    LanguageCode='en-US',
    MediaFormat='mp3',
    Media={
        'MediaFileUri': mp3_uri
    },
    Settings={
        'ShowSpeakerLabels': True,
        'MaxSpeakerLabels': 2 # assumes two people on a phone call
    }
)

# Wait for the transcription job to finish.
print('Waiting for job to complete...')
while True:
    response =
 transcribe_client.get_transcription_job(TranscriptionJobName=audio_file_name)
    if response['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
 'FAILED']:
        break
    else:
        print('Still waiting...')
```

```
    time.sleep(10)

transcript_uri = response['TranscriptionJob']['Transcript']['TranscriptFileUri']

# Open the JSON file, read it, and get the transcript.
response = urllib.request.urlopen(transcript_uri)
raw_json = response.read()
loaded_json = json.loads(raw_json)
transcript = loaded_json['results']['transcripts'][0]['transcript']

# Send transcript to Comprehend for key phrases and sentiment.
comprehend_client = boto3.client('comprehend')

# If necessary, trim the transcript.
# If the transcript is more than 5 KB, the Comprehend calls fail.
if len(transcript) > 5000:
    trimmed_transcript = transcript[:5000]
else:
    trimmed_transcript = transcript

print('Detecting key phrases...')
response = comprehend_client.detect_key_phrases(
    Text=trimmed_transcript,
    LanguageCode='en'
)

keywords = []
for keyword in response['KeyPhrases']:
    keywords.append(keyword['Text'])

print('Detecting sentiment...')
response = comprehend_client.detect_sentiment(
    Text=trimmed_transcript,
    LanguageCode='en'
)

sentiment = response['Sentiment']

# Build the Amazon OpenSearch Service URL.
id = audio_file_name.strip('.mp3')
url = domain + '/' + index + '/' + type + '/' + id

# Create the JSON document.
```

```
json_document = {'transcript': transcript, 'keywords': keywords, 'sentiment':
 sentiment, 'timestamp': datetime.datetime.now().isoformat()}

# Provide all details necessary to sign the indexing request.
credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region,
 'opensearchservice', session_token=credentials.token)

# Index the document.
print('Indexing document...')
response = requests.put(url, auth=awsauth, json=json_document, headers=headers)

print(response)
print(response.json())
```

2. Mettez à jour les six variables initiales.
3. Installez les packages requis à l'aide des commandes suivantes :

```
pip install boto3
pip install requests
pip install requests_aws4auth
```

4. MP3 Placez-le dans le même répertoire que le script `call-center.py` et exécutez-le. Un exemple de sortie suit :

```
$ python call-center.py
Uploading 000001.mp3...
Starting transcription job...
Waiting for job to complete...
Still waiting...
Still waiting...
Still waiting...
Still waiting...
Still waiting...
Still waiting...
Still waiting...
Detecting key phrases...
Detecting sentiment...
Indexing document...
<Response [201]>
```

```
{u'_type': u'call', u'_seq_no': 0, u'_shards': {u'successful': 1, u'failed': 0,
 u'total': 2}, u'_index': u'support-calls4', u'_version': 1, u'_primary_term': 1,
 u'result': u'created', u'_id': u'000001'}
```

`call-center.py` effectue un certain nombre d'opérations :

1. Le script télécharge un fichier audio (dans ce cas, un MP3, mais Amazon Transcribe prend en charge plusieurs formats) dans votre compartiment S3.
2. Il envoie l'URL du fichier audio à Amazon Transcribe et attend que la tâche de transcription se termine.

   Le temps nécessaire pour terminer la tâche de transcription dépend de la longueur du fichier audio. Supposons des minutes, pas des secondes.

   > **Tip**
   >
   > Pour améliorer la qualité de la transcription, vous pouvez configurer un vocabulaire personnalisé pour Amazon Transcribe.

3. Une fois la tâche de transcription terminée, le script extrait la transcription, la tronque à 5 000 caractères et l'envoie à Amazon Comprehend pour l'analyse des mots-clés et des sentiments.
4. Enfin, le script ajoute la transcription complète, les mots-clés, le sentiment et l'horodatage actuel à un document JSON et l'indexe dans OpenSearch Service.

> **Tip**
>
> LibriVoxcontient des livres audio du domaine public que vous pouvez utiliser à des fins de test.

(Facultatif) Étape 3 : Indexer des exemples de données

Si vous n'avez pas beaucoup d'enregistrements d'appels (ce qui est souvent le cas) à portée de main, vous pouvez indexer les exemples de document dans sample-calls.zip, qui sont comparables à ce que `call-center.py` produit.

1. Créez un fichier nommé `bulk-helper.py`:

```
import boto3
from opensearchpy import OpenSearch, RequestsHttpConnection
import json
from requests_aws4auth import AWS4Auth

host = '' # For example, my-test-domain.us-west-2.es.amazonaws.com
region = '' # For example, us-west-2
service = 'es'

bulk_file = open('sample-calls.bulk', 'r').read()

credentials = boto3.Session().get_credentials()
awsauth = AWS4Auth(credentials.access_key, credentials.secret_key, region, service,
 session_token=credentials.token)

search = OpenSearch(
    hosts = [{'host': host, 'port': 443}],
    http_auth = awsauth,
    use_ssl = True,
    verify_certs = True,
    connection_class = RequestsHttpConnection
)

response = search.bulk(bulk_file)
print(json.dumps(response, indent=2, sort_keys=True))
```

2. Mettez à jour les deux variables initiales pour `host` et `region`.
3. Installez le package requis à l'aide de la commande suivante :

```
pip install opensearch-py
```

4. Téléchargez et décompressez sample-calls.zip.
5. Placez `sample-calls.bulk` dans le même répertoire que `bulk-helper.py` et lancez l'assistant. Un exemple de sortie suit :

```
$ python bulk-helper.py
{
  "errors": false,
  "items": [
    {
      "index": {
```

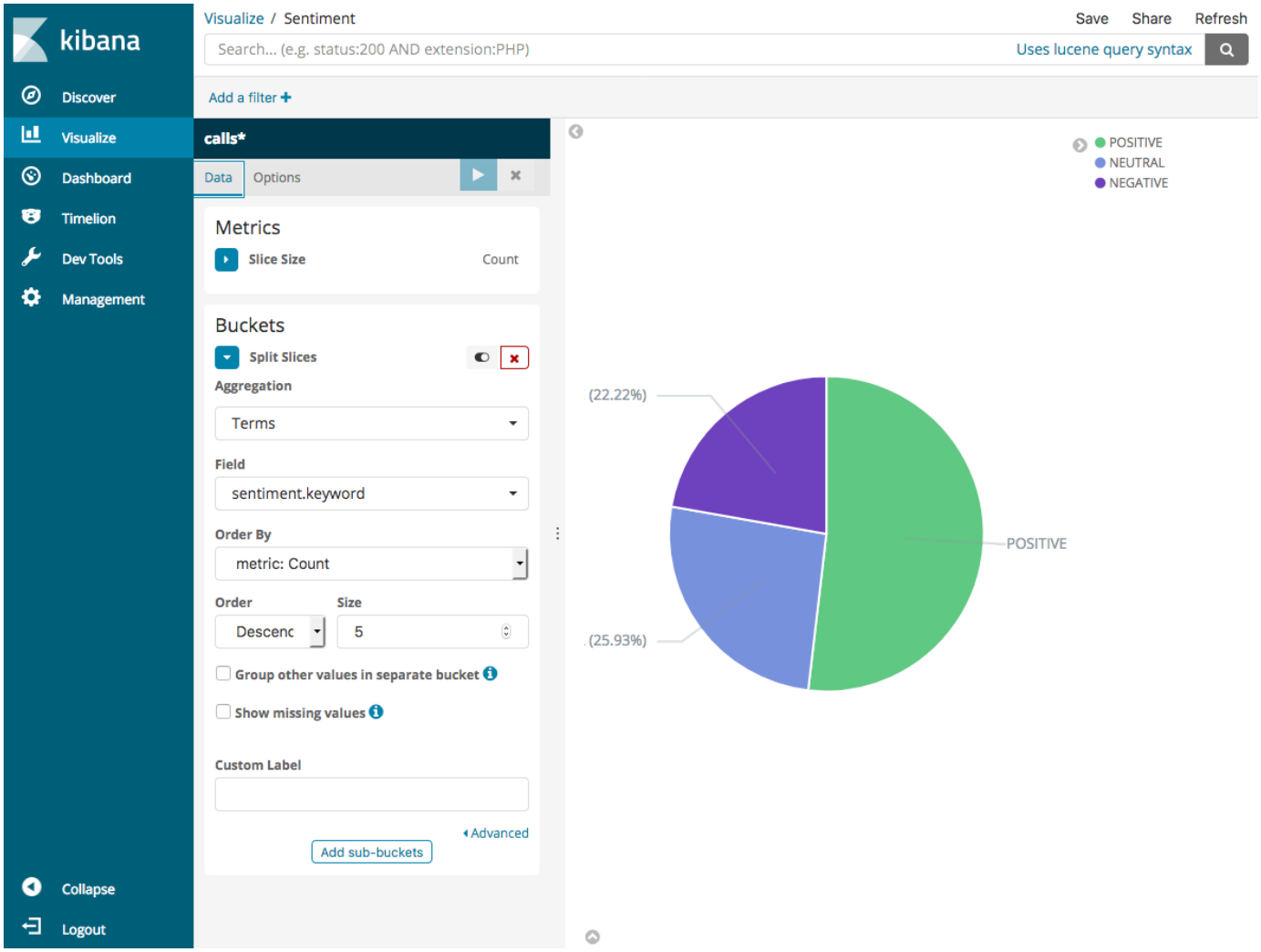
```
        "_id": "1",
        "_index": "support-calls",
        "_primary_term": 1,
        "_seq_no": 42,
        "_shards": {
          "failed": 0,
          "successful": 1,
          "total": 2
        },
        "_type": "_doc",
        "_version": 9,
        "result": "updated",
        "status": 200
      }
    },
    ...
  ],
  "took": 27
}
```

Étape 4 : Analyser et visualiser vos données

Maintenant que vous disposez de données dans OpenSearch Service, vous pouvez les visualiser à l'aide de OpenSearch Dashboards.

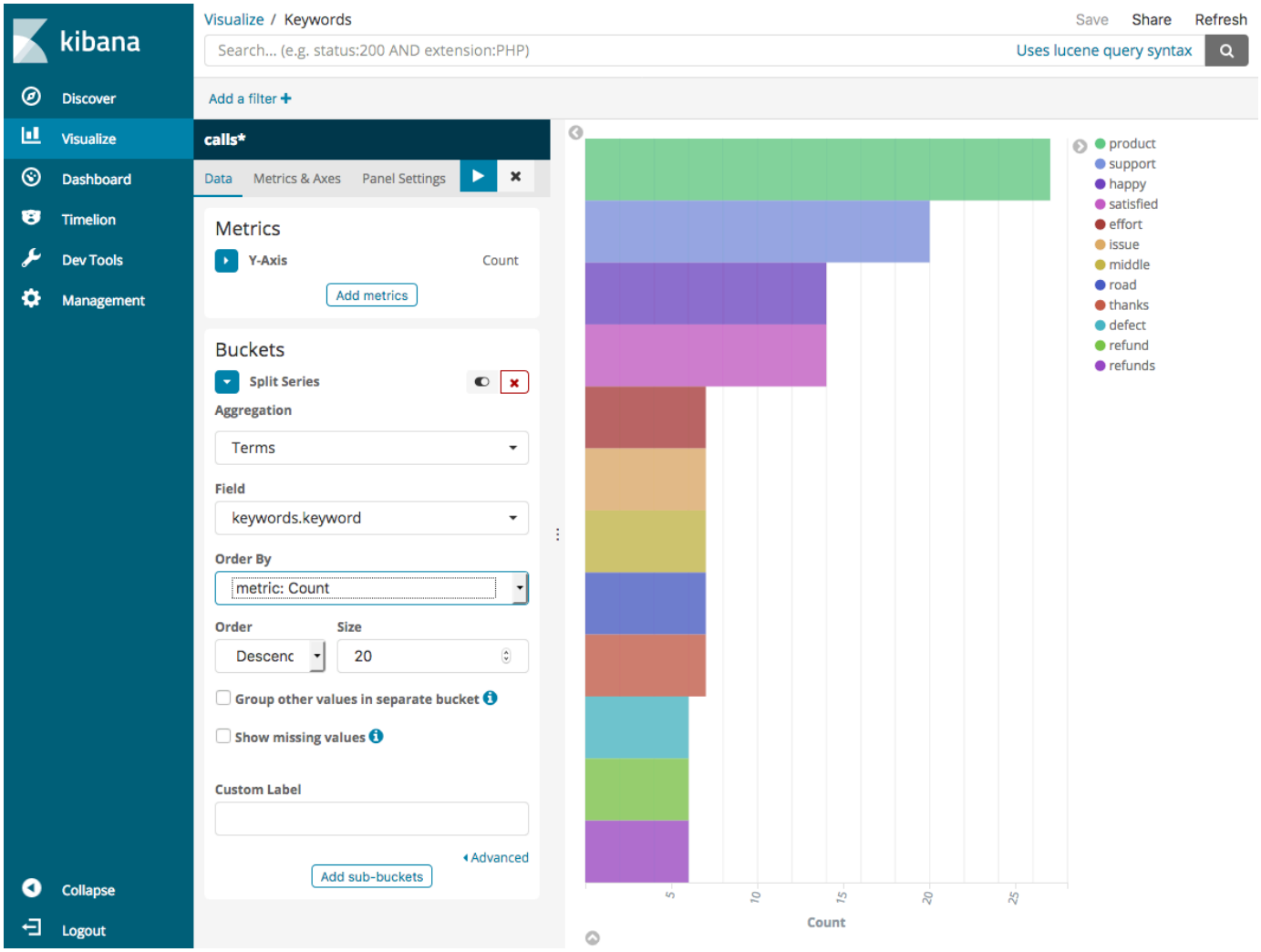
1. Accédez à `https://search-`*`domain`*`.`*`region`*`.es.amazonaws.com/_dashboards`.
2. Avant de pouvoir utiliser OpenSearch Dashboards, vous devez définir un modèle d'index. Dashboards utilise des modèles d'index pour affiner votre analyse à un ou plusieurs index. Pour faire correspondre l'index `support-calls` créé par `call-center.py`, accédez à Stack Management (Gestion de pile), Index Patterns (Modèles d'index), et définissez un modèle d'index de `support*`, puis choisissez Next step (Étape suivante).
3. Pour Time Filter field name (Nom du champ de filtre de temps), choisissez timestamp (horodatage).
4. Maintenant, vous pouvez commencer à créer des visualisations. Choisissez Visualize (Visualiser), puis ajoutez une nouvelle visualisation.
5. Choisissez le diagramme à secteurs et le modèle d'index `support*`.
6. La valeur par défaut est la visualisation de base. De ce fait, choisissez Split Slices (Fractionner des tranches) pour créer une visualisation plus intéressante.

Pour Aggregation (Regroupement), choisissez Terms (Termes). Pour Field (Champ), choisissez sentiment.keyword. Puis, choisissez Apply changes (Appliquer les modifications) et Save (Enregistrer).

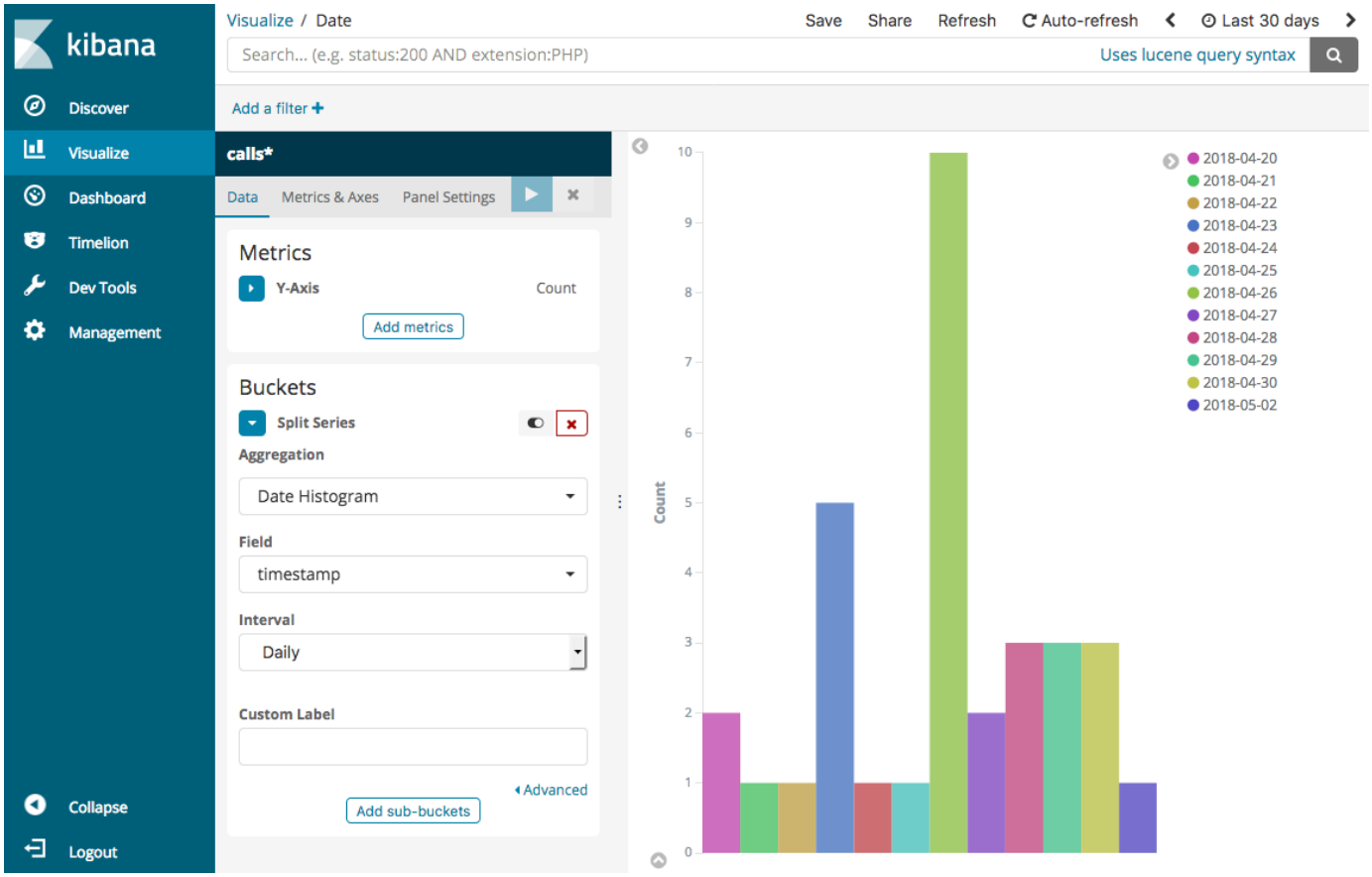


7. Revenez à la page Visualize (Visualiser) et ajoutez une autre visualisation. Cette fois-ci, sélectionnez le diagramme à barres horizontales.
8. Sélectionnez Split Series (Fractionner les séries).

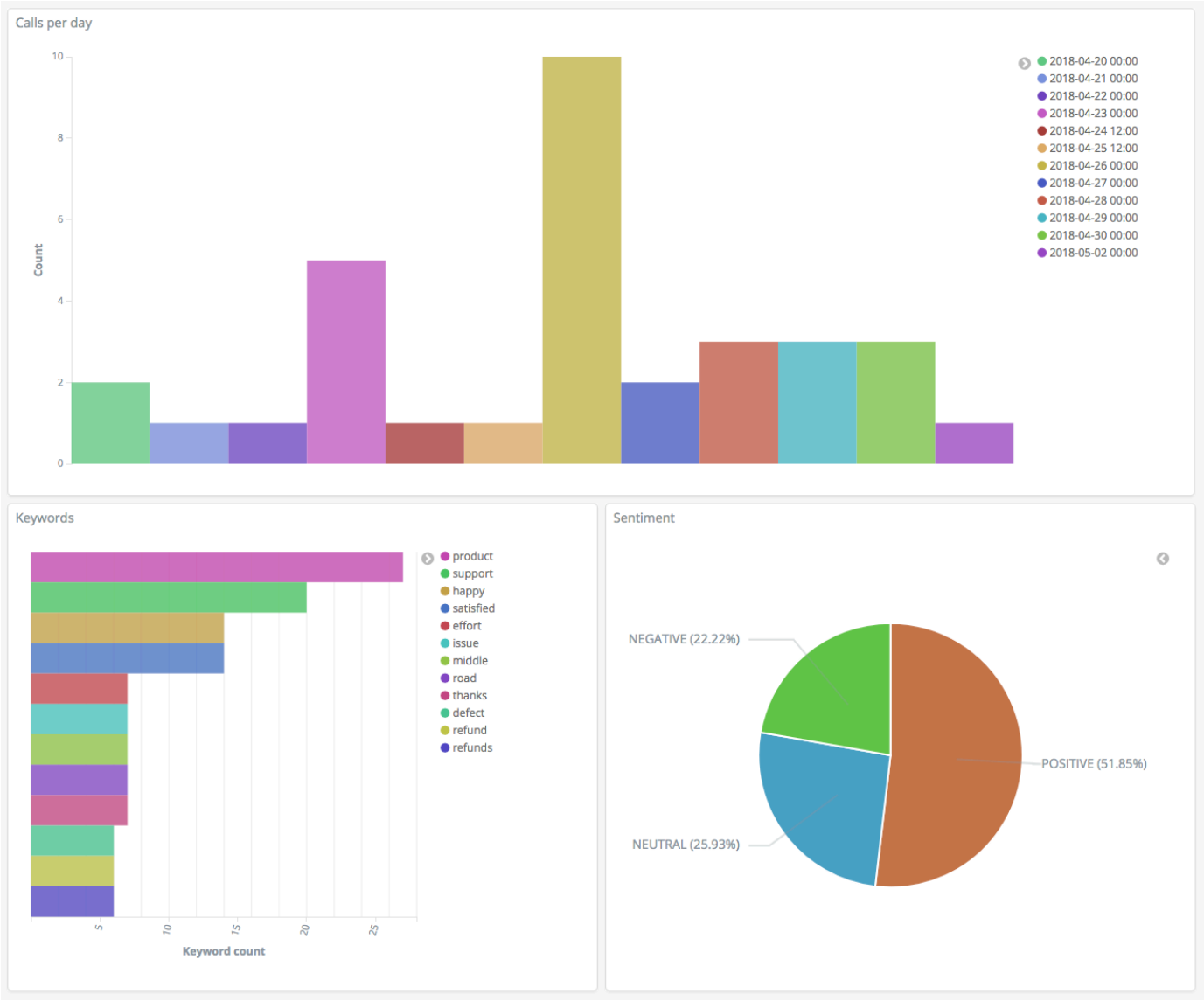
   Pour Aggregation (Regroupement), choisissez Terms (Termes). Pour Field (Champ), choisissez keywords.keyword et définissez Size (Taille) sur 20. Puis, choisissez Apply Changes (Appliquer les modifications) et Save (Enregistrer).



9. Revenez à la page Visualize (Visualiser) et ajoutez une visualisation finale, un diagramme à barres verticales.
10. Sélectionnez Split Series (Fractionner les séries). Pour Aggregation (Regroupement), choisissez Date Histogram (Histogramme par date). Pour Field (Champ), choisissez timestamp (horodatage) et définissez Interval (Intervalle) sur Daily (Quotidien).
11. Choisissez Metrics & Axes (Métriques et axes) et définissez Mode sur normal.
12. Choisissez Apply Changes (Appliquer les modifications) et Save (Enregistrer).



13. Maintenant que vous avez trois visualisations, vous pouvez les ajouter à un tableau de bord Dashboards. Choisissez Dashboard (Tableau de bord), créez un tableau de bord, puis ajoutez vos visualisations.



Étape 5 : Nettoyage des ressources et étapes suivantes

Pour éviter des frais inutiles, supprimez le compartiment S3 et le domaine OpenSearch Service. Pour en savoir plus, consultez Supprimer un compartiment dans le guide de l'utilisateur d'Amazon Simple Storage Service et Supprimer un domaine de OpenSearch service dans ce guide.

Les transcriptions nécessitent beaucoup moins d'espace disque que MP3 les fichiers. Vous pouvez raccourcir votre fenêtre de MP3 conservation, par exemple, de trois mois d'enregistrements d'appels à un mois. Vous pouvez conserver des années de transcription tout en continuant d'économiser sur les coûts de stockage.

Vous pouvez également automatiser le processus de transcription à l'aide de AWS Step Functions et Lambda, ajouter des métadonnées supplémentaires avant l'indexation, ou élaborer des visualisations plus complexes pour s'adapter à votre cas d'utilisation.

Changement de nom d'Amazon OpenSearch Service : résumé des modifications

Le 8 septembre 2021, notre suite de recherche et d'analyse a été renommé Amazon OpenSearch Service. OpenSearch Service pris en charge OpenSearch ainsi que l'ancien système d'exploitation hérité Elasticsearch. Les sections suivantes décrivent les différentes parties du service qui ont changé avec le nouveau nom et les actions que vous devez effectuer afin de garantir le bon fonctionnement de vos domaines.

Certaines de ces modifications s'appliquent uniquement lorsque vous mettez à niveau vos domaines d'Elasticsearch vers. OpenSearch Dans d'autres cas, par exemple dans la console Billing and Cost Management, l'expérience change immédiatement.

Veuillez noter que cette liste n'est pas exhaustive. Bien que d'autres parties du produit aient également changé, ces mises à jour sont les plus pertinentes.

Nouvelle version d'API

La nouvelle version de l'API OpenSearch de configuration (2021-01-01) fonctionne avec OpenSearch le système d'exploitation hérité Elasticsearch. 21 opérations d'API ont été remplacées par des noms plus concis et indépendants du moteur (par exemple, `CreateElasticsearchDomain` remplacé par`CreateDomain`), mais OpenSearch Service continue de prendre en charge les deux versions de l'API.

Nous vous recommandons d'utiliser les nouvelles opérations d'API pour créer et gérer des domaines à l'avenir. Notez que lorsque vous utilisez les nouvelles opérations d'API pour créer un domaine, vous devez spécifier le paramètre `EngineVersion` au format `Elasticsearch_X.Y` ou `OpenSearch_X.Y`, plutôt que simplement le numéro de version. Si vous ne spécifiez aucune version, elles utiliseront par défaut la plus récente version de OpenSearch.

Mettez à niveau AWS CLI vers la version 1.20.40 ou version ultérieure afin de pouvoir l'utiliser `aws opensearch ...` pour créer et gérer vos domaines. Pour connaître le nouveau format CLI, consultez la Référence de la OpenSearch CLI.

Types d'instances renommés

Les types d'instances dans Amazon OpenSearch Service sont désormais au format`<type>.<size>.search`, par exemple, `m6g.large.search` plutôt que`m6g.large.elasticsearch`. Aucune action de votre part n'est requise. Les domaines existants commenceront automatiquement à faire référence aux nouveaux types d'instances dans l'API et dans la console Billing and Cost Management.

Si vous possédez des instances réservées (RIs), votre contrat ne sera pas affecté par la modification. L'ancienne version de l'API de configuration est toujours compatible avec l'ancien format de dénomination, mais si vous souhaitez utiliser la nouvelle version de l'API, vous devrez utiliser le nouveau format.

Modifications des stratégies d'accès

Les sections suivantes décrivent les actions que vous devez effectuer pour mettre à jour vos stratégies d'accès.

Politiques IAM

Nous vous recommandons de mettre à jour les politiques IAM afin de pouvoir utiliser les opérations API renommées. Cependant, OpenSearch Service continuera de respecter les politiques existantes en répliquant en interne les anciennes autorisations d'API. Par exemple, si vous disposez actuellement de l'autorisation d'exécuter l'opération `CreateElasticsearchDomain`, vous pourrez maintenant appeler à la fois `CreateElasticsearchDomain` (ancienne opération d'API) et `CreateDomain` (nouvelle opération d'API). Il en va de même pour les refus explicites. Pour obtenir la liste des opérations d'API mises à jour, consultez la référence des éléments de politique.

Politiques SCP

Les politiques de contrôle des services (SCPs) introduisent une couche de complexité supplémentaire par rapport à la politique IAM standard. Pour éviter que vos politiques SCP ne soient rompues, vous devez ajouter l'ancienne et la nouvelle opération d'API à chacune de vos politiques SCP. Par exemple, si un utilisateur dispose actuellement d'autorisations pour `CreateElasticsearchDomain`, vous devez également lui octroyer des autorisations pour `CreateDomain` afin qu'il puisse conserver la possibilité de créer des domaines. Il en va de même pour les refus explicites.

Par exemple :

```
"Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "es:CreateElasticsearchDomain",
        "es:CreateDomain"
         ...
      ],
    },
      "Effect": "Deny",
      "Action:" [
        "es:DeleteElasticsearchDomain",
        "es:DeleteDomain"
         ...
```

Nouveaux types de ressources

OpenSearch Service introduit les nouveaux types de ressources suivants :

Ressource	Description
`AWS::OpenSearchService::Domain`	Représente un domaine Amazon OpenSearch Service. Cette ressource existe au niveau du service et n'est pas propre au logiciel exécuté sur le domaine. Elle s'applique à des services comme AWS CloudFormation et Resource Groups AWS, dans lesquels vous créez et gérez des ressources pour le service dans son ensemble. Pour obtenir des instructions sur la mise à niveau CloudFormation des domaines définis dans Elasticsearch vers OpenSearch, consultez Remarques du Guide de l' CloudFormation utilisateur.
`AWS::OpenSearch::Domain`	Représente le logiciel OpenSearch /Elasticsearch exécuté sur un domaine. Cette ressource s'applique à des services tels que AWS

Ressource	Description
	CloudTrailet AWS Config, qui font référence au logiciel exécuté sur le domaine plutôt qu'au OpenSearch Service dans son ensemble. Ces services contiennent désormais des types de ressources distincts pour les domaines exécutant Elasticsearch (`AWS::Elasticsearch::Domain`) par rapport aux domaines exécutant OpenSearch (`AWS::OpenSearch::Domain`).

> **Note**
>
> Dans AWS Config, vous continuerez à voir vos données sous le type de `AWS::Elasticsearch::Domain` ressource pendant plusieurs semaines, même si vous mettez à niveau un ou plusieurs domaines vers OpenSearch.

Kibana renommé Tableaux de bord OpenSearch

OpenSearch Tableaux de bord, l' AWS alternative à Kibana, est un outil de visualisation open source conçu pour fonctionner. OpenSearch Après la mise à niveau d'un domaine depuis Elasticsearch vers OpenSearch, le `/_plugin/kibana` point de terminaison devient. `/_dashboards` OpenSearch Le service redirigera toutes les demandes vers le nouveau point de terminaison, mais si vous utilisez le point de terminaison Kibana dans l'une des politiques IAM, mettez à jour ces politiques pour inclure également le nouveau `/_dashboards` point de terminaison.

Si vous utilisezthe section called “Authentification SAML pour les tableaux de bord OpenSearch ”, avant de mettre à niveau votre domaine OpenSearch, vous devez remplacer tous les Kibana URLs configurés dans votre fournisseur d'identité (IdP) `/_plugin/kibana` par. `/_dashboards` Les plus courants URLs sont Assertion Consumer Service (ACS) URLs et le destinataire URLs.

Le `kibana_read_only` rôle par défaut pour les OpenSearch tableaux de bord a été renommé en`opensearch_dashboards_read_only`, et le `kibana_user` rôle a été renommé en`opensearch_dashboards_user`. La modification s'applique à tous les 1 nouvellement créés OpenSearch . domaines x exécutant le logiciel de service R20211203 ou version ultérieure. Si vous

mettez à niveau un domaine existant vers le logiciel de service R20211203, les noms des rôles restent les mêmes.

CloudWatch Métriques renommées

Plusieurs CloudWatch métriques changent pour des domaines exécutés OpenSearch. Lorsque vous mettez à niveau un domaine vers OpenSearch, les métriques changent automatiquement et vos CloudWatch alarmes actuelles seront désactivées. Avant de mettre à niveau votre cluster d'une version Elasticsearch vers une autre OpenSearch version, veillez à mettre à jour vos CloudWatch alarmes afin de pouvoir utiliser les nouvelles métriques.

Les métriques suivantes ont été modifiées :

Nom initial de la métrique	Nouveau nom
`KibanaHealthyNodes`	`OpenSearchDashboardsHealthyNodes`
`KibanaConcurrentConnections`	`OpenSearchDashboardsConcurrentConnections`
`KibanaHeapTotal`	`OpenSearchDashboardsHeapTotal`
`KibanaHeapUsed`	`OpenSearchDashboardsHeapUsed`
`KibanaHeapUtilization`	`OpenSearchDashboardsHeapUtilization`
`KibanaOS1MinuteLoad`	`OpenSearchDashboardsOS1MinuteLoad`
`KibanaRequestTotal`	`OpenSearchDashboardsRequestTotal`
`KibanaResponseTimesMaxInMillis`	`OpenSearchDashboardsResponseTimesMaxInMillis`
`ESReportingFailedRequestSysErrCount`	`KibanaReportingFailedRequestSysErrCount`
`ESReportingRequestCount`	`KibanaReportingRequestCount`

Nom initial de la métrique	Nouveau nom
`ESReportingFailedRequestUserErrCount`	`KibanaReportingFailedRequestUserErrCount`
`ESReportingSuccessCount`	`KibanaReportingSuccessCount`
`ElasticsearchRequests`	`OpenSearchRequests`

Pour obtenir la liste complète des statistiques que OpenSearch Service envoie à Amazon CloudWatch, consultezthe section called “Surveillance des métriques d'un cluster”.

Modifications apportées à la console Billing and Cost Management

Les données historiques contenues dans la console Billing and Cost Management et dans les Rapports d'utilisation et de coût continueront d'utiliser l'ancien nom de service, vous devrez donc commencer à utiliser des filtres pour Amazon OpenSearch Service et pour l'ancien nom Elasticsearch lorsque vous rechercherez des données. Si vous avez déjà enregistré des rapports, mettez les filtres à jour afin de vous assurer qu'ils incluent également OpenSearch Service. Il se peut que vous receviez initialement une alerte lorsque votre utilisation d'Elasticsearch diminuera au profit d'Elasticsearch OpenSearch, mais celle-ci disparaîtra après quelques jours.

En plus du nom du service, les champs suivants seront modifiés pour toutes les opérations d'API de rapports, de factures et de tarification :

Champ	Ancien format	Nouveau format
Type d'instance	`m5.large.elasticsearch`	`m5.large.search`
Famille de produits	Instance Elasticsearch Volume Elasticsearch	Instance Amazon OpenSearch Service Volume Amazon OpenSearch Service

Champ	Ancien format	Nouveau format
Description de la tarification	5,098 USD par heure d'instanc e c5.18xlarge.elasticsearch (ou heure partielle) – UE	5,098 USD par heure d'instanc e c5.18xlarge.search (ou heure partielle) – UE
Famille d'instances	`ultrawarm.elastics earch`	`ultrawarm.search`

Nouveau format d'événement

Le format des événements envoyés par OpenSearch Service à Amazon EventBridge et Amazon CloudWatch a changé, en particulier le `detail-type` champ. Le champ source (`aws.es`) reste le même. Pour connaître le format complet de chaque type d'événement, consultez the section called "Surveillance des événements". Si vous disposez de règles d'événements qui dépendent de l'ancien format, veillez à les mettre à jour pour qu'elles soient conformes au nouveau format.

Qu'est-ce qui demeure identique ?

Les fonctions et fonctionnalités suivantes, ainsi que d'autres non répertoriées ici, demeureront identiques :

- Principal du service (`es.amazonaws.com`)
- Code fournisseur
- Domaine ARNs
- Points de terminaison de domaine

Mise en route : mettez à niveau vos domaines vers OpenSearch 1.x

OpenSearch 1. x prend en charge les mises à niveau à partir d'Elasticsearch versions 6.8 et 7. x. Pour obtenir des instructions sur la mise à niveau de votre domaine, consultez the section called "Mise à niveau d'un domaine (console)". Si vous utilisez la AWS CLI ou l'API de configuration pour mettre à niveau votre domaine, vous devez indiquer la « `TargetVersion` as `OpenSearch_1.x` ».

OpenSearch 1. x introduit un paramètre de domaine supplémentaire appelé Activer le mode de compatibilité. Étant donné que certains clients et plugins Elasticsearch OSS vérifient la version du cluster avant de se connecter, le mode OpenSearch de compatibilité définit la version 7.10 afin que ces clients continuent de fonctionner.

Vous pouvez activer le mode de compatibilité lorsque vous créez des OpenSearch domaines pour la première fois ou lorsque vous effectuez une mise à niveau OpenSearch vers une version Elasticsearch. S'il n'est pas défini, le paramètre sera défini par défaut sur `false` lorsque vous créerez un domaine, et sur `true` lorsque vous mettrez un domaine à niveau.

Pour activer le mode de compatibilité à l'aide de l'API de configuration, définissez `override_main_response_version` sur `true` :

```
POST https://es.us-east-1.amazonaws.com/2021-01-01/opensearch/upgradeDomain
{
  "DomainName": "domain-name",
  "TargetVersion": "OpenSearch_1.0",
  "AdvancedOptions": {
    "override_main_response_version": "true"
   }
}
```

Pour activer ou désactiver le mode de compatibilité sur des OpenSearch domaines existants, vous devez utiliser l'opération d'API OpenSearch _cluster/settings :

```
PUT /_cluster/settings
{
  "persistent" : {
    "compatibility.override_main_response_version" : true
  }
}
```

Résolution des problèmes liés à Amazon OpenSearch Service

Cette rubrique explique comment identifier et résoudre les problèmes courants d'Amazon OpenSearch Service. Consultez les informations de cette section avant de contacter Support AWS.

Impossible d'accéder aux OpenSearch tableaux de bord

Le point de terminaison OpenSearch Dashboards ne prend pas en charge les demandes signées. Si la stratégie de contrôle d'accès de votre domaine n'accorde l'accès qu'à certains rôles IAM et que vous n'avez pas configuré l'authentification Amazon Cognito, vous pouvez recevoir l'erreur suivante lorsque vous tentez d'accéder à Dashboards :

```
"User: anonymous is not authorized to perform: es:ESHttpGet"
```

Si votre domaine OpenSearch de service utilise l'accès VPC, il se peut que vous ne receviez pas cette erreur, mais le délai de la demande peut expirer. Pour en savoir plus sur la correction ce problème et les différentes options de configuration disponibles the section called “Contrôle de l'accès aux tableaux de bord ”, consultez the section called “À propos des stratégies d'accès pour les domaines de VPC” et the section called “Gestion de l’identité et des accès”.

Impossible d'accéder au domaine VPC

Consultez the section called “À propos des stratégies d'accès pour les domaines de VPC” et the section called “Test des domaines de VPC”.

Cluster en lecture seule

Par rapport aux versions antérieures d'Elasticsearch OpenSearch et à Elasticsearch 7. x utiliser un système différent pour la coordination des clusters. Dans ce nouveau système, lorsque le cluster perd le quorum, le cluster est indisponible jusqu'à ce que vous preniez des mesures. La perte de quorum peut prendre deux formes :

- Si votre cluster utilise des nœuds principaux dédiés, une perte de quorum se produit lorsque la moitié ou plus n'est pas disponible.

- Si votre cluster n'utilise pas de nœuds principaux dédiés, une perte de quorum se produit lorsque la moitié ou plus de vos nœuds de données sont indisponibles.

En cas de perte de quorum et que votre cluster possède plusieurs nœuds, le OpenSearch service rétablit le quorum et place le cluster en mode lecture seule. Vous avez deux options :

- Supprimez l'état en lecture seule et utilisez le cluster en l'état.
- Restaurez le cluster ou des index individuels à partir d'un instantané.

Si vous préférez utiliser le cluster tel quel, vérifiez que l'état du cluster est vert à l'aide de la demande suivante :

```
GET _cat/health?v
```

Si l'état du cluster est rouge, nous vous recommandons de restaurer le cluster à partir d'un instantané. Vous pouvez également consulter the section called “Statut de cluster rouge” pour connaître les étapes de dépannage. Si l'état du cluster est vert, vérifiez que tous les index attendus sont présents à l'aide de la demande suivante :

```
GET _cat/indices?v
```

Ensuite, exécutez quelques recherches pour vérifier que les données attendues sont présentes. Si c'est le cas, vous pouvez supprimer l'état en lecture seule à l'aide de la demande suivante :

```
PUT _cluster/settings
{
  "persistent": {
    "cluster.blocks.read_only": false
  }
}
```

En cas de perte du quorum et que votre cluster ne possède qu'un seul nœud, le OpenSearch service remplace le nœud et ne place pas le cluster en mode lecture seule. Sinon, vos options sont les mêmes : utilisez le cluster en l'état ou restaurez-le à partir d'un instantané.

Dans les deux cas, le OpenSearch service envoie deux événements à votre AWS Health Dashboard. La premier événement vous informe de la perte du quorum. La seconde se produit une fois que le

OpenSearch service a rétabli le quorum avec succès. Pour plus d'informations sur l'utilisation du AWS Health Dashboard, consultez le guide de AWS Health l'utilisateur.

Statut de cluster rouge

Un statut de cluster rouge signifie qu'au moins une partition principale et ses répliques ne sont pas allouées à un nœud. OpenSearch Le service continue d'essayer de prendre des instantanés automatisés de tous les index, quel que soit leur statut, mais les instantanés échouent tant que l'état du cluster rouge persiste.

Les causes les plus courantes d'un statut de cluster rouge sont des nœuds de cluster défaillants et l'échec du processus OpenSearch en raison d'une importante charge de traitement continue.

> **Note**
>
> OpenSearch Le service stocke les instantanés automatisés pendant 14 jours, quel que soit l'état du cluster. Si le statut de cluster rouge persiste au-delà de deux semaines, le dernier instantané automatique sain est supprimé et vous risquez de perdre définitivement les données de votre cluster. Si votre domaine de OpenSearch service passe au statut de cluster rouge, Support vous pouvez vous contacter pour vous demander si vous souhaitez résoudre le problème vous-même ou si vous souhaitez que l'équipe d'assistance vous aide. Vous pouvez définir une CloudWatch alarme pour vous avertir lorsqu'un statut de cluster rouge apparaît.

Finalement, des partitions rouges entraînent des clusters rouges et des index rouges entraînent des partitions rouges. Pour identifier les index à l'origine de l'état du cluster rouge, OpenSearch cela est utile APIs.

- `GET /_cluster/allocation/explain` choisit la première partition non attribuée trouvée et explique pourquoi celle-ci ne peut pas être allouée à un nœud :

```
{
    "index": "test4",
    "shard": 0,
    "primary": true,
    "current_state": "unassigned",
    "can_allocate": "no",
```

```
    "allocate_explanation": "cannot allocate because allocation is not permitted to
 any of the nodes"
}
```

- `GET /_cat/indices?v` affiche l'état de santé, le nombre de documents et l'utilisation du disque pour chaque index :

```
health status index           uuid                   pri rep docs.count docs.deleted
 store.size pri.store.size
green  open   test1           30h1EiMvS5uAFr2t5CEVoQ   5   0        820            0
       14mb           14mb
green  open   test2           sdIxs_WDT56afFGu5KPbFQ   1   0          0            0
       233b           233b
green  open   test3           GGRZp_TBRZuSaZpAGk2pmw   1   1          2            0
     14.7kb          7.3kb
red    open   test4           BJxfAErbTtu5HBjIXJV_7A   1   0
green  open   test5           _8C6MIXOSxCqVYicH3jsEA   1   0          7            0
     24.3kb         24.3kb
```

La suppression des index rouges constitue le moyen le plus rapide de résoudre un statut de cluster rouge. En fonction de la raison de l'état du cluster rouge, vous pouvez ensuite redimensionner votre domaine de OpenSearch service pour utiliser des types d'instances plus grands, davantage d'instances ou davantage de stockage basé sur EBS et essayer de recréer les index problématiques.

Si la suppression d'un index problématique n'est pas possible, vous pouvez restaurer un instantané, supprimer des documents de l'index, modifier les paramètres d'index, réduire le nombre de réplicas ou supprimer d'autres index pour libérer de l'espace sur le disque. L'étape importante consiste à résoudre l'état du cluster rouge avant de reconfigurer votre domaine OpenSearch de service. La reconfiguration d'un domaine avec un statut de cluster rouge peut aggraver le problème et entraîner le blocage du domaine dans un état de configuration En cours de traitement tant que vous n'aurez pas résolu le statut.

Correction automatique des clusters rouges

Si l'état de votre cluster reste rouge pendant plus d'une heure, le OpenSearch service tente de le corriger automatiquement en redirigeant les partitions non allouées ou en effectuant une restauration à partir d'anciens instantanés.

S'il ne parvient pas à corriger un ou plusieurs index rouges et que l'état du cluster reste rouge pendant 14 jours au total, le OpenSearch service ne prend d'autres mesures que si le cluster répond à au moins l'un des critères suivants :

- Il n'a qu'une seule zone de disponibilité
- Il n'a pas de nœuds principaux dédiés
- Il contient des types d'instances burstables (T2 ou T3)

À l'heure actuelle, si votre cluster répond à l'un de ces critères, OpenSearch Service vous envoie des notifications quotidiennes au cours des 7 prochains jours expliquant que si vous ne corrigez pas ces index, toutes les partitions non attribuées seront supprimées. Si l'état de votre cluster est toujours rouge au bout de 21 jours, OpenSearch Service supprime les partitions non attribuées (stockage et calcul) sur tous les index rouges. Vous recevez des notifications dans le panneau Notifications de la console de OpenSearch service pour chacun de ces événements. Pour de plus amples informations, veuillez consulter the section called “Événements relatifs à l'état du cluster”.

Récupération après une importante charge de traitement continue

Pour déterminer si un statut de cluster rouge est dû à une importante charge de traitement continue sur un nœud de données, surveillez les métriques de cluster suivantes.

Métrique pertinente	Description	Récupération
JVMMemoryPression	Spécifie le pourcentage du tas Java utilisé pour tous les nœuds de données d'un cluster. Affichez la statistique Maximum pour cette métrique et recherchez les variations de sollicitation de la mémoire alors que le processus de nettoyage Java ne parvient pas à récupérer suffisamment de mémoire. Ce modèle est vraisemblablement dû à des requêtes complexes ou des champs de données volumineux.	Définissez des disjoncteurs de circuit de mémoire pour la JVM. Pour plus d'informations, consultez the section called “JVM OutOfMemoryError”. Si le problème persiste, supprimez les index inutiles, réduisez le nombre ou la complexité des demandes envoyées au domaine, ajoutez des instances ou

Métrique pertinente	Description	Récupération
	Les types d'instance x86 utilisent le récupérateur de mémoire Concurrent Mark Sweep (CMS), lequel s'exécute avec les threads d'application visant à limiter les arrêts. Si CMS n'est pas en mesure de récupérer suffisamment de mémoire lors de ses récupérations habituelles, il déclenchera une récupération complète de la mémoire, laquelle peut entraîner de longues suspensions d'application et avoir un impact sur la stabilité du cluster. Les types d'instance Graviton basés sur ARM utilisent le récupérateur de mémoire Garbage-First (G1), qui est similaire au CMS, mais utilise des pauses courtes et une défragmentation de tas supplémentaires pour réduire davantage le besoin de récupération de mémoire complète. Dans les deux cas, si l'utilisation de la mémoire continue d'augmenter au-delà de ce que le ramasse-miettes peut récupérer pendant une collecte complète des déchets, cela OpenSearch se bloque en raison d'une erreur de mémoire insuffisante. Sur tous les types d'instance, une règle consiste à garder l'utilisation en-dessous de 80 %. L'API `_nodes/stats/jvm` récapitule de manière pratique les	utilisez de plus grands types d'instance.

Métrique pertinente	Description	Récupération
	statistiques JVM, l'utilisation du pool de mémoire et les informations relatives au nettoyage de la mémoire : `GET domain-endpoint /_nodes/stats/jvm?pretty`	
CPUUtilization	Spécifie le pourcentage de ressources d'UC utilisées pour les nœuds de données d'un cluster. Affichez la statistique Maximum pour cette métrique et recherchez un modèle continu d'utilisation élevée.	Ajoutez des nœuds de données ou augmentez la taille des types d'instances des nœuds de données existants.
Nœuds	Spécifie le nombre de nœuds dans un cluster. Affichez la statistique Minimum pour cette métrique. Cette valeur fluctue lorsque le service déploie une nouvelle flotte d'instances pour un cluster.	Ajoutez des nœuds de données.

Statut de cluster jaune

Un statut de cluster jaune signifie que les partitions principales de tous les index sont attribuées aux nœuds d'un cluster, sauf pour les partitions de réplica d'au moins un index. Les clusters à nœud unique s'initialisent toujours avec un statut de cluster jaune car il n'existe aucun autre nœud auquel le OpenSearch Service peut attribuer une réplique. Pour obtenir un statut de cluster vert, augmentez votre nombre de nœuds. Pour en savoir plus, consultez the section called “Dimensionnement des domaines”.

Les clusters à plusieurs nœuds peuvent brièvement présenter un statut de cluster jaune après la création d'un nouvel index ou après une défaillance de nœud. Cet état se résout automatiquement au fur et à mesure que OpenSearch les données sont répliquées dans le cluster. Un manque d'espace

disque peut également provoquer un statut de cluster jaune ; le cluster peut uniquement distribuer des partitions de réplica si les nœuds disposent de l'espace disque nécessaire pour les accueillir.

ClusterBlockException

Vous pouvez recevoir une erreur `ClusterBlockException` pour les raisons suivantes.

Manque d'espace de stockage disponible

Si un ou plusieurs nœuds de votre cluster disposent d'un espace de stockage inférieur à la valeur minimale de 1) 20 % de l'espace de stockage disponible ou 2) 20 GiB d'espace de stockage, les opérations d'écriture de base telles que l'ajout de documents et la création d'index peuvent commencer à échouer. the section called “Calcul des exigences de stockage”fournit un résumé de la façon dont le OpenSearch Service utilise l'espace disque.

Pour éviter tout problème, surveillez la `FreeStorageSpace` métrique dans la console de OpenSearch service et créez des CloudWatch alarmes qui se déclenchent en cas de `FreeStorageSpace` chute en dessous d'un certain seuil. `GET /_cat/allocation?v`fournit également un résumé utile de l'allocation des partitions et de l'utilisation du disque. Pour résoudre les problèmes liés au manque d'espace de stockage, adaptez votre domaine de OpenSearch service pour utiliser des types d'instances plus importants, davantage d'instances ou davantage de stockage basé sur EBS.

Pression mémoire élevée de la JVM

Lorsque la métrique de JVMMemorypression dépasse 92 % pendant 30 minutes, le OpenSearch service déclenche un mécanisme de protection et bloque toutes les opérations d'écriture pour empêcher le cluster d'atteindre le statut rouge. Lorsque la protection est activée, les opérations d'écriture échouent avec une erreur `ClusterBlockException`, aucun nouvel index ne peut être créé et l'erreur `IndexCreateBlockException` est générée.

Lorsque la métrique de JVMMemorypression revient à 88 % ou moins pendant cinq minutes, la protection est désactivée et les opérations d'écriture sur le cluster sont débloquées.

Une pression élevée sur la mémoire de la JVM peut être due à des pics de demandes adressées au cluster, à des allocations de partitions déséquilibrées entre les nœuds, à un trop grand nombre de partitions dans un cluster, à des explosions de données de champ ou de mappage d'index, ou à des types d'instances incapables de gérer les charges entrantes. Cela peut également être dû

à l'utilisation d'agrégations, de caractères génériques ou de longues plages de temps dans les requêtes.

Pour réduire le trafic vers le cluster et résoudre les problèmes de forte pression sur la mémoire de la JVM, essayez l'une ou plusieurs des solutions suivantes :

- Mettez le domaine à l'échelle de sorte que la taille maximale du tas par nœud soit de 32 Go.
- Réduisez le nombre de partitions en supprimant les index anciens ou inutilisés.
- Videz le cache de données à l'aide de l'opération d'API `POST index-name/_cache/clear?fielddata=true`. Notez que vider le cache peut perturber les requêtes en cours.

En général, pour éviter une forte pression sur la mémoire de la JVM à l'avenir, suivez ces meilleures pratiques :

- Évitez les agrégations sur les champs de texte ou modifiez le type de mappage de vos index en `keyword`.
- Optimisez les demandes de recherche et d'indexation en choisissant le bon nombre de shards.
- Configurez des stratégies de gestion de l'état des index (ISM) pour supprimer régulièrement les index inutilisés.

Erreur lors de la migration vers le mode Multi-AZ avec mode veille

Les problèmes suivants peuvent se produire lorsque vous migrez un domaine existant vers le mode Multi-AZ avec mode veille.

Création d'un index, d'un modèle d'index ou d'une politique ISM lors de la migration de domaines sans mode veille vers des domaines en mode veille

Si vous créez un index lors de la migration d'un domaine de Multi-AZ sans mode veille vers mode veille et que le modèle d'index ou la politique ISM ne respectent pas les directives de copie de données recommandées, cela peut entraîner une incohérence des données et la migration peut échouer. Pour éviter cette situation, créez le nouvel index avec un nombre de copies de données (y compris les nœuds principaux et les répliques) multiple de trois. Vous pouvez vérifier la progression de la migration à l'aide de l'`DescribeDomainChangeProgress`API. Si vous rencontrez une erreur liée au nombre de répliques, corrigez-la, puis contactez le AWS Support pour réessayer la migration.

Nombre de copies de données incorrect

Si vous n'avez pas le bon nombre de copies de données dans votre domaine, la migration vers Multi-AZ with Standby échouera.

JVM OutOfMemoryError

Une erreur JVM `OutOfMemoryError` signifie généralement que l'un des disjoncteurs suivants du circuit JVM a été atteint.

Disjoncteur de circuit	Description	Propriété de configuration du cluster
Disjoncteur parent	Pourcentage total de mémoire de tas JVM autorisé pour tous les disjoncteurs de circuit. La valeur par défaut est 95 %.	`indices.breaker.total.limit`
Disjoncteur de données de champ	Pourcentage de mémoire du tas JVM autorisé à charger un seul champ de données en mémoire. La valeur par défaut est 40%. Si vous chargez des données avec des champs volumineux, vous devrez peut-être augmenter cette limite.	`indices.breaker.fielddata.limit`
Disjoncteur de requête	Pourcentage de mémoire du tas JVM autorisé pour les structures de données utilisées pour répondre à une demande de service. La valeur par défaut est 60%. Si vos demandes	`indices.breaker.request.limit`

Disjoncteur de circuit	Description	Propriété de configuration du cluster
	de service impliquent de calculer des agrégatio ns, vous devrez peut-être augmenter cette limite.	

Nœuds de cluster en échec

EC2 Les instances Amazon peuvent subir des interruptions et des redémarrages inattendus. Généralement, le OpenSearch service redémarre les nœuds pour vous. Cependant, il est possible qu'un ou plusieurs nœuds dans un cluster OpenSearch reste en échec.

Pour vérifier cette condition, ouvrez le tableau de bord de votre domaine sur la console OpenSearch de service. Accédez à l'onglet Santé du cluster et recherchez la métrique Total des nœuds. Vérifiez si le nombre de nœuds indiqué est inférieur au nombre que vous avez configuré pour votre cluster. Si la métrique indique qu'un ou plusieurs nœuds sont en panne pendant plus d'une journée, contactez AWS Support.

Vous pouvez également définir une CloudWatch alarme pour vous avertir lorsque ce problème survient.

> **Note**
>
> La métrique Total des nœuds n'est pas précise lors de modifications dans la configuration de votre cluster et lors d'opérations de maintenance du service. Ce comportement est normal. Cette métrique rapportera bientôt le nombre correct de nœuds du cluster. Pour en savoir plus, veuillez consulter la section the section called “Configuration changes”.

Pour protéger vos clusters contre les fermetures et les redémarrages inattendus de nœuds, créez au moins une réplique pour chaque index de votre domaine de OpenSearch service.

Limite maximale de partitions dépassée

OpenSearch ainsi que 7. Les versions x d'Elasticsearch ont un paramètre par défaut de 1 000 partitions maximum par nœud. OpenSearch/Elasticsearch génère une erreur si une demande, telle

que la création d'un nouvel index, vous fait dépasser cette limite. Si vous rencontrez cette erreur, vous disposez de plusieurs options :

- Ajoutez d'autres nœuds de données au cluster.
- Augmentez la valeur du paramètre `_cluster/settings/cluster.max_shards_per_node`.
- Utilisez l'API _shrink pour réduire le nombre de partitions sur le nœud.

Domaine bloqué dans l'état de traitement

Votre domaine OpenSearch de service passe à l'état « En cours de traitement » lorsqu'il est en cours de modification de configuration. Lorsque vous initiez une modification de configuration, le statut du domaine passe à « Traitement » tandis que le OpenSearch service crée un nouvel environnement. Dans le nouvel environnement, OpenSearch Service lance un nouvel ensemble de nœuds applicables (tels que data, master ou UltraWarm). Une fois la migration terminée, les nœuds plus anciens sont résiliés.

Le cluster peut rester bloqué dans l'état « Processing » (Traitement en cours) si l'une de ces situations se produit :

- Un nouvel ensemble de nœuds de données ne parvient pas à se lancer.

- La migration des partitions vers le nouvel ensemble de nœuds de données échoue.
- Le contrôle de validation a échoué avec des erreurs.

Pour connaître les étapes de résolution détaillées dans chacune de ces situations, consultez Pourquoi mon domaine Amazon OpenSearch Service est-il bloqué dans l'état « En cours de traitement » ? .

Solde de débordement EBS faible

OpenSearch Le service vous envoie une notification sur console lorsque le solde de rupture EBS sur l'un de vos volumes à usage général (SSD) est inférieur à 70 %, et une notification de suivi si le solde tombe en dessous de 20 %. Pour résoudre ce problème, vous pouvez soit augmenter la capacité de votre cluster, soit réduire les IOPS de lecture et d'écriture afin que le solde de débordement puisse être crédité. Le solde de rafale reste à 0 pour les domaines avec des types de volumes gp3 et les domaines avec des volumes gp2 dont la taille de volume est supérieure à 1 000 Gio. Pour plus

d'informations, consultez Volumes SSD à usage général (gp2). Vous pouvez surveiller l'équilibre des rafales EBS à l'aide de la `BurstBalance` CloudWatch métrique.

La métrique EBS augmente lors du redimensionnement du volume

Lorsque vous modifiez la taille des volumes d'Amazon Elastic Block Store, vous pouvez observer des augmentations temporaires de diverses métriques EBS telles que `Write ThroughputWrite Throughput Micro bursting`,`Disk Queue Depth`, et`IOPS`. Ce comportement est attendu lors de l'opération de redimensionnement et dure généralement quelques secondes. La durée peut toutefois varier en fonction de la taille du volume modifiée.

Pour éviter les problèmes de latence et les rejets de demandes, effectuez des redimensionnements de volume EBS uniquement lorsque le cluster est sain et que le trafic réseau est faible.

Impossible d'activer les journaux d'audit

L'erreur suivante peut s'afficher lorsque vous essayez d'activer la publication du journal d'audit à l'aide de la console OpenSearch de service :

La politique d'accès aux ressources spécifiée pour le groupe de CloudWatch journaux Logs n'accorde pas les autorisations suffisantes à Amazon OpenSearch Service pour créer un flux de journaux. Vérifiez la stratégie d'accès aux ressources.

Si vous rencontrez cette erreur, vérifiez que l'élément `resource` de votre politique inclut l'ARN du groupe de journaux qui convient. Si tel est le cas, procédez comme suit :

1. Attendez quelques minutes.
2. Actualisez la page dans votre navigateur web.
3. Choisissez Select existing group (Sélectionner un groupe existant).
4. Pour Existing log group (Groupe de journaux existant), choisissez le groupe de journaux que vous avez créé avant de recevoir le message d'erreur.
5. Dans la section Stratégie d'accès, choisissez Select existing policy (Sélectionner une stratégie existante).
6. Pour Existing policy (Politique existante), choisissez la politique que vous avez créée avant de recevoir le message d'erreur.
7. Choisissez Enable (Activer).

Si l'erreur persiste après avoir répété le processus plusieurs fois, contactez Support AWS.

Impossible de fermer l'index

OpenSearch Le service prend en charge l'`_close`API uniquement pour les versions 7.4 OpenSearch et ultérieures d'Elasticsearch. Si vous utiliser une version plus ancienne et restaurez un index à partir d'un instantané, vous pouvez supprimer l'index existant (avant ou après l'avoir réindexé).

Vérifications des licences des clients

Les distributions par défaut de Logstash et Beats incluent une vérification de licence propriétaire et ne permettent pas de se connecter à la version open source de. OpenSearch Assurez-vous d'utiliser les distributions Apache 2.0 (OSS) de ces clients avec OpenSearch Service.

Limitation des demandes

Si vous recevez constamment des erreurs `403 Request throttled due to too many requests` ou `429 Too Many Requests`, envisagez une mise à l'échelle verticale. Amazon OpenSearch Service limite les demandes si la charge utile est susceptible d'entraîner une utilisation de la mémoire supérieure à la taille maximale du segment Java.

Impossible d'accéder au nœud via SSH

Vous ne pouvez pas utiliser SSH pour accéder aux nœuds de votre OpenSearch cluster, et vous ne pouvez pas les modifier `opensearch.yml` directement. Utilisez plutôt la console ou SDKs configurez votre domaine. AWS CLI Vous pouvez également spécifier quelques paramètres au niveau du cluster à l'aide du OpenSearch REST APIs. Pour en savoir plus, consultez le manuel Amazon OpenSearch Service API Reference etthe section called “Opérations prises en charge”.

Si vous avez besoin de plus d'informations sur les performances du cluster, vous pouvez publier des journaux d'erreurs et des journaux lents sur CloudWatch.

Erreur d'instantané « Non valide pour la classe de stockage de l'objet »

OpenSearch Les instantanés de service ne prennent pas en charge la classe de stockage S3 Glacier. Vous pouvez rencontrer cette erreur lorsque vous essayez de répertorier les instantanés si votre

compartiment S3 comprend une règle de cycle de vie qui transfère des objets vers la classe de stockage S3 Glacier.

Si vous devez restaurer un instantané à partir du compartiment, restaurez les objets à partir de S3 Glacier, copiez les objets dans un nouveau compartiment et enregistrez le nouveau compartiment en tant que référentiel d'instantanés.

En-tête d'hôte non valide

OpenSearch Le service nécessite que les clients le spécifient `Host` dans les en-têtes de demande. Une valeur `Host` valide est le point de terminaison du domaine sans `https://`, comme :

```
Host: search-my-sample-domain-ih2lhn2ew2scurji.us-west-2.es.amazonaws.com
```

Si vous recevez un `Invalid Host Header` message d'erreur lorsque vous faites une demande, vérifiez que votre client ou proxy inclut le point de terminaison du domaine de OpenSearch service (et non, par exemple, son adresse IP) dans l'`Host`en-tête.

Type d'instance M3 non valide

OpenSearch Le service ne prend pas en charge l'ajout ou la modification d'instances M3 à des domaines existants exécutant OpenSearch ou à des versions 6.7 ou ultérieures d'Elasticsearch. Vous pouvez continuer à utiliser des instances M3 avec des versions 6.5 et antérieures d'Elasticsearch.

Nous vous recommandons de choisir un type d'instance plus récent. Pour les domaines exécutant OpenSearch Elasticsearch 6.7 ou version ultérieure, les restrictions suivantes s'appliquent :

- Si votre domaine existant n'utilise pas d'instances M3, vous ne pouvez plus les modifier.
- Si vous faites passer un domaine existant d'un type d'instance M3 vers un autre type d'instance, vous ne pouvez pas revenir en arrière.

Les hot queries cessent de fonctionner après l'activation UltraWarm

Lorsque vous l'activez UltraWarm sur un domaine, s'il n'existe aucune dérogation préexistante au `search.max_buckets` paramètre, OpenSearch Service définit automatiquement la valeur sur pour empêcher les requêtes gourmandes en mémoire de `10000` saturer les nœuds chauds. Si vos requêtes actives utilisent plus de 10 000 compartiments, elles risquent de ne plus fonctionner lorsque vous les activez UltraWarm.

Comme vous ne pouvez pas modifier ce paramètre en raison de la nature gérée d'Amazon OpenSearch Service, vous devez ouvrir un dossier d'assistance pour augmenter la limite. L'augmentation des limites ne nécessite pas d'abonnement Premium Support.

Impossible de revenir à une version plus ancienne après la mise à niveau

Les mises à niveau sur place sont irréversibles, mais si vous contactez l'Assistance AWS, ils peuvent vous aider à restaurer l'instantané automatique d'avant la mise à niveau sur un nouveau domaine. Par exemple, si vous mettez à niveau un domaine d'Elasticsearch 5.6 vers la version 6.4, le AWS Support peut vous aider à restaurer le snapshot antérieur à la mise à niveau sur un nouveau domaine Elasticsearch 5.6. Si vous prenez un instantané manuel du domaine d'origine, vous pouvez effectuer cette étape vous-même.

Résumé nécessaire des domaines pour toutes les Régions AWS

Le script suivant utilise la AWS CLI commande Amazon EC2 describe-regions pour créer une liste de toutes les régions dans lesquelles le OpenSearch service pourrait être disponible. Ensuite, il demande list-domain-namesà chaque région de :

```
for region in `aws ec2 describe-regions --output text | cut -f4`
do
    echo "\nListing domains in region '$region':"
    aws opensearch list-domain-names --region $region --query 'DomainNames'
done
```

Vous recevez la sortie suivante pour chaque région :

```
Listing domains in region:'us-west-2'...
[
  {
    "DomainName": "sample-domain"
  }
]
```

Les régions dans lesquelles le OpenSearch service n'est pas disponible renvoient « Impossible de se connecter à l'URL du point de terminaison ».

Erreur du navigateur lors de l'utilisation des OpenSearch tableaux de bord

Votre navigateur intègre les messages d'erreur de service dans des objets de réponse HTTP lorsque vous utilisez des tableaux de bord pour afficher les données de votre domaine de OpenSearch service. Vous pouvez utiliser les outils de développement généralement disponibles dans les navigateurs Web, tels que Developer Mode dans Chrome, pour afficher les erreurs de service sous-jacentes et aider vos efforts de débogage.

Pour afficher les erreurs de service dans Chrome

1. Dans la barre de menu Chrome, choisissez Afficher, Développeur, Outils pour développeur.
2. Choisissez l'onglet Network (Réseau).
3. Dans la colonne Status (État), choisissez une session HTTP ayant 500 comme état.

Pour afficher les erreurs de service dans Firefox

1. Dans le menu, choisissez Tools (Outils), Web Developer (Développeur Web), Network (Réseau).
2. Choisissez n'importe quelle session HTTP ayant un état 500.
3. Choisissez l'onglet Response (Réponse) pour afficher la réponse du service.

Asymétrie des partitions et de stockage des nœuds

L'asymétrie des partitions d'un nœud se produit lorsqu'un ou plusieurs nœuds d'un cluster possèdent beaucoup plus de partitions que les autres nœuds. L'asymétrie de stockage d'un nœud se produit lorsqu'un ou plusieurs nœuds d'un cluster possèdent beaucoup plus de stockage (`disk.indices`) que les autres nœuds. Bien que ces deux conditions puissent se produire temporairement, comme lorsqu'un domaine a remplacé un nœud et lui alloue toujours des partitions, vous devez y remédier si elles persistent.

Pour identifier les deux types d'asymétrie, exécutez l'opération d'API _cat/allocation et comparez les entrées `shards` et `disk.indices` dans la réponse :

```
shards    | disk.indices  | disk.used  | disk.avail   | disk.total   | disk.percent |
host     | ip       | node
   264    |      465.3mb  |   229.9mb  |      1.4tb   |      1.5tb   |            0 |
 x.x.x.x  | x.x.x.x  | node1
```

```
   115    |        7.9mb  |    83.7mb  |     49.1gb   |     49.2gb   |           0 |
x.x.x.x  | x.x.x.x  | node2
   264    |      465.3mb  |   235.3mb  |      1.4tb   |      1.5tb   |           0 |
 x.x.x.x  | x.x.x.x  | node3
   116    |        7.9mb  |    82.8mb  |     49.1gb   |     49.2gb   |           0 |
x.x.x.x  | x.x.x.x  | node4
   115    |        8.4mb  |      85mb  |     49.1gb   |     49.2gb   |           0 |
x.x.x.x  | x.x.x.x  | node5
```

Bien qu'une certaine asymétrie de stockage soit normale, tout écart de plus de 10 % par rapport à la moyenne est significatif. Lorsque la distribution des partitions est asymétrique, l'utilisation du CPU, du réseau et de la bande passante du disque peut également être asymétrique. Dans la mesure où plus de données signifient généralement plus d'opérations d'indexation et de recherche, les nœuds les plus chargés ont également tendance à être les plus sollicités en termes de ressources, tandis que les moins chargés représentent une capacité sous-utilisée.

Correction : utilisez des partitions dont le nombre est un multiple du nombre de nœuds de données pour garantir que chaque index est réparti uniformément sur les nœuds de données.

Asymétrie des partitions et du stockage des index

L'asymétrie des partitions d'un index se produit lorsqu'un ou plusieurs nœuds détiennent plus de partitions d'un index que les autres nœuds. L'asymétrie de stockage d'un index se produit lorsqu'un ou plusieurs nœuds détiennent une quantité disproportionnée du stockage total d'un index.

L'asymétrie d'index est plus difficile à identifier que l'asymétrie de nœuds car elle nécessite une certaine manipulation de la sortie de l'API _cat/shards. Examinez l'asymétrie de l'index s'il y a des indications d'asymétrie dans les métriques du cluster ou du nœud. Voici quelques indications courantes d'asymétrie d'index :

- Erreurs HTTP 429 se produisant sur un sous-ensemble de nœuds de données
- Mise en file d'attente inégale des index ou des opérations de recherche sur les nœuds de données
- Utilisation inégale du and/or processeur JVM Heap Heap sur les nœuds de données

Correction : utilisez des partitions dont le nombre est un multiple du nombre de nœuds de données pour garantir que chaque index est réparti uniformément sur les nœuds de données. Si vous constatez toujours un stockage d'index ou un biais de partition, vous devrez peut-être forcer une réallocation de partition, ce qui se produit à chaque déploiement bleu/vert de votre domaine de service. OpenSearch

Opération non autorisée après la sélection de l'accès VPC

Lorsque vous créez un nouveau domaine à l'aide de la console de OpenSearch service, vous avez la possibilité de sélectionner un accès VPC ou public. Si vous sélectionnez l'accès au VPC, le OpenSearch service demande des informations sur le VPC et échoue si vous ne disposez pas des autorisations appropriées :

```
You are not authorized to perform this operation. (Service: AmazonEC2; Status Code:
 403; Error Code: UnauthorizedOperation
```

Pour activer cette requête, vous devez avoir accès aux opérations `ec2:DescribeVpcs`, `ec2:DescribeSubnets` et `ec2:DescribeSecurityGroups`. Cette exigence concerne uniquement la console. Si vous utilisez la AWS CLI pour créer et configurer un domaine avec un point de terminaison VPC, vous n'avez pas besoin d'accéder à ces opérations.

Blocage du chargement suite à la création d'un domaine VPC

Après avoir créé un nouveau domaine qui utilise un accès VPC, l'état de configuration du domaine ne dépasse pas le stade du chargement. Si ce problème se produit, vous avez probablement désactivé AWS Security Token Service (AWS STS) pour votre région.

Pour ajouter des points de terminaison VPC à votre VPC, le OpenSearch service doit assumer le rôle. `AWSServiceRoleForAmazonOpenSearchService` AWS STS Il doit donc être activé pour créer de nouveaux domaines utilisant l'accès VPC dans une région donnée. Pour en savoir plus sur l'activation et la désactivation AWS STS, consultez le guide de l'utilisateur IAM.

Demandes refusées à l' OpenSearch API

Avec l'introduction du contrôle d'accès basé sur des balises pour l' OpenSearch API, vous pourriez commencer à voir des erreurs de refus d'accès là où vous ne le faisiez pas auparavant. Cela peut être dû au fait qu'une ou plusieurs des stratégies d'accès contiennent le `Deny` utilisant la condition `ResourceTag` et ces conditions sont maintenant respectées.

Par exemple, la stratégie suivante est utilisée uniquement pour refuser l'accès à l'action `CreateDomain` de l'API de configuration, si le domaine comportait la balise `environment=production`. Bien que la liste d'actions inclue également `ESHttpPut`, la déclaration de refus ne s'appliquait pas à cette action ou à toute autre action `ESHttp*`.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Action": [
      "es:CreateDomain",
      "es:ESHttpPut"
    ],
    "Effect": "Deny",
    "Resource": "*",
    "Condition": {
      "ForAnyValue:StringEquals": {
        "aws:ResourceTag/environment": [
          "production"
        ]
      }
    }
  }]
}
```

Avec la prise en charge supplémentaire des balises pour les méthodes OpenSearch HTTP, une politique basée sur l'identité IAM comme celle décrite ci-dessus empêchera l'utilisateur attaché d'accéder à l'action. `ESHttpPut` Auparavant, en l'absence de la validation à l'aide de balises, l'utilisateur attaché pouvait toujours envoyer des requêtes PUT.

Si vous commencez à voir des erreurs d'accès refusé après la mise à jour des domaines au logiciel de service R20220323 ou version ultérieure, vérifiez les stratégies d'accès basées sur l'identité pour voir si tel est le cas et mettez-les à jour si nécessaire pour autoriser l'accès.

Impossible de se connecter à partir d'Alpine Linux

Alpine Linux limite la taille de réponse DNS à 512 octets. Si vous essayez de vous connecter à votre domaine de OpenSearch service depuis Alpine Linux version 3.18.0 ou inférieure, la résolution DNS peut échouer si le domaine se trouve dans un VPC et compte plus de 20 nœuds. Si vous utilisez une version d'Alpine Linux supérieure à 3.18.0, vous devriez être en mesure de résoudre plus de 20 hôtes. Pour plus d'informations, consultez les notes de mise à jour d'Alpine Linux 3.18.0.

Si votre domaine est dans un VPC, nous vous recommandons d'utiliser d'autres distributions de Linux, telles que Debian, Ubuntu, CentOS, Red Hat Enterprise Linux ou Amazon Linux 2, pour vous y connecter.

Trop de demandes pour Search Backpressure

Le contrôle d'admission basé sur le processeur est un mécanisme de contrôle d'accès qui limite de manière proactive le nombre de demandes adressées à un nœud en fonction de sa capacité actuelle, à la fois en cas d'augmentation organique et de pic de trafic. Les demandes excessives renvoient un code d'état HTTP 429 « Trop de demandes » en cas de rejet. Cette erreur indique soit des ressources de cluster insuffisantes, soit des demandes de recherche gourmandes en ressources, soit une augmentation involontaire de la charge de travail.

La contre-pression de recherche est à l'origine du rejet, ce qui peut aider à affiner les demandes de recherche gourmandes en ressources. En cas de pics de trafic, nous recommandons de réessayer côté client avec un recul et une instabilité exponentiels.

Erreur de certificat lors de l'utilisation du kit SDK

Dans la AWS SDKs mesure où vous utilisez les certificats CA de votre ordinateur, les modifications apportées aux certificats sur les AWS serveurs peuvent provoquer des échecs de connexion lorsque vous tentez d'utiliser un SDK. Les messages d'erreur varient, mais ils contiennent en général le texte suivant :

```
Failed to query OpenSearch
...
SSL3_GET_SERVER_CERTIFICATE:certificate verify failed
```

Vous pouvez éviter ces défaillances en conservant les certificats CA et le système d'exploitation de votre ordinateur up-to-date. Si vous rencontrez ce problème dans un environnement d'entreprise et que vous ne gérez pas votre propre ordinateur, vous pourrez être amené à demander à un administrateur de vous aider pour effectuer la mise à jour.

La liste suivante présente les versions minimales requises pour le système d'exploitation et Java :

- Les versions de Microsoft Windows sur lesquelles des mises à jour sont installées depuis janvier 2005 contiennent au moins l'une des versions requises CAs dans leur liste de confiance.

- Mac OS X 10.4 avec Java pour Mac OS X 10.4 version 5 (février 2007), Mac OS X 10.5 (octobre 2007) et les versions ultérieures contiennent au moins l'un des éléments requis CAs dans leur liste de confiance.
- Red Hat Enterprise Linux 5 (mars 2007), 6 et 7 et CentOS 5, 6 et 7 contiennent tous au moins l'un des éléments requis CAs dans leur liste d'autorités de certification fiables par défaut.
- Java 1.4.2_12 (mai 2006), 5 Update 2 (mars 2005) et toutes les versions ultérieures, y compris Java 6 (décembre 2006), 7 et 8, contiennent au moins l'un des éléments requis CAs dans leur liste d'autorités de certification fiables par défaut.

Les trois autorités de certification sont :

- Amazon Root CA 1
- Starfield Services Root Certificate Authority – G2
- Starfield Class 2 Certification Authority

Les certificats racine des deux premières autorités sont disponibles auprès d'Amazon Trust Services, mais la solution la plus simple consiste à conserver votre ordinateur up-to-date. Pour en savoir plus sur les certificats fournis par ACM, consultez. AWS Certificate Manager FAQs

> **Note**
>
> Actuellement, les domaines OpenSearch de service de la région us-east-1 utilisent des certificats provenant d'une autre autorité. Nous prévoyons de mettre à jour la région afin d'utiliser ces nouvelles CA dans un avenir proche.

L'installation du plugin personnalisé échoue en raison de la compatibilité des versions

Problème : L'installation du plugin a échoué en raison d'une incompatibilité de version entre le plugin et l' OpenSearch instance en cours d'exécution. Le système renvoie le message d'erreur suivant :

```
PluginValidationFailureReason : The provided plugin could not be loaded.
```

Cause : Le plugin a été compilé pour OpenSearch ${*MAJOR*}. ${*MINOR*}.{*PATCH*}, mais votre environnement exécute OpenSearch ${*MAJOR*}. {*MINOR*}0,00$. OpenSearch nécessite une

correspondance de version exacte entre les plugins et l' OpenSearchinstallation principale pour des raisons de stabilité et de sécurité.

Solution possible : créez le plugin avec OpenSearch la version ${*MAJOR*}. $ *{MINOR}* .0 pour correspondre à la version de votre cluster.

Pour vérifier et mettre à jour la version de OpenSearch

1. Utilisez l'API ou le tableau de bord de votre cluster pour exécuter la commande suivante. Remplacez *default placeholder values* par vos propres informations.

   Demande d'API :

   ```
   curl -X GET your-opensearch-endpoint/
   ```

   Console Dev Tools dans le tableau de bord :

   ```
   GET /
   ```

   La commande renvoie des informations au format suivant.

   ```
   {
     "name": "node-id",
     "cluster_name": "account-id:domain-name",
     "cluster_uuid": "cluster-uuid",
     "version": {
       "distribution": "opensearch",
       "number": "2.17.0",
       "build_type": "tar",
       "build_hash": "unknown",
       "build_date": "2024-12-17T11:00:09.799828091Z",
       "build_snapshot": false,
       "lucene_version": "9.11.1",
       "minimum_wire_compatibility_version": "7.10.0",
       "minimum_index_compatibility_version": "7.0.0"
     },
     "tagline": "The OpenSearch Project: https://opensearch.org/"
   }
   ```

2. Si le numéro de version n'est pas ${*MAJOR*}. $ *{MINOR}* .0, reconstruisez le plugin en procédant comme suit :

a. Mettez à jour les plugins `descriptor.properties` pour spécifier la version $*{MAJOR}*.*{MINOR}*0,00$.

b. Reconstruisez le plugin à l'aide de la commande correspondant à votre type de projet.

c. Exécutez la commande update-package à l'aide du nouveau fichier. `.zip`

d. Exécutez la commande associate-package pour associer la dernière version du plugin créée lors de l'exécution de la `update-package` commande à l'étape précédente.

Historique du document pour Amazon OpenSearch Service

Cette rubrique décrit les modifications importantes apportées à Amazon OpenSearch Service. Les mises à jour de logiciel de service ajoutent la prise en charge des nouvelles fonctionnalités, des correctifs de sécurité, des corrections de bogues et d'autres améliorations. Pour utiliser de nouvelles fonctionnalités, vous devrez peut-être mettre à jour le logiciel de service sur votre domaine. Pour de plus amples informations, veuillez consulter the section called “Mises à jour du logiciel de service”.

Les fonctionnalités du service sont déployées progressivement en fonction de l' Régions AWS endroit où un service est disponible. Nous mettons à jour cette documentation pour la première version uniquement. Nous ne fournissons pas d'informations sur la disponibilité des régions et n'annonçons pas les déploiements régionaux ultérieurs. Pour plus d'informations sur la disponibilité des fonctionnalités du service par région et pour vous abonner aux notifications concernant les mises à jour, voir Quelles sont les nouveautés AWS ?

Pour recevoir des notifications sur les mises à jour, vous pouvez vous abonner au flux RSS.

> **Note**
>
> Versions de correctifs : les versions du logiciel de service qui se terminent par « -P » et un numéro, comme R20211203-P4, sont des versions de correctifs. Les correctifs sont susceptibles d'inclure des améliorations de performance, des corrections de bogues mineurs et des corrections de sécurité ou des améliorations de posture. Étant donné que les correctifs n'incluent pas de nouvelles fonctions ou de modifications importantes, ils n'ont généralement pas d'impact direct sur l'utilisateur ou la documentation, c'est pourquoi les détails de chaque correctif ne sont pas inclus dans l'historique de ce document.

Modification	Description	Date
Nouveau sujet : intégration d'Atlassian à Ingestion OpenSearch	Vous pouvez désormais utiliser les pipelines Amazon OpenSearch Ingestion pour traiter les données issues des services Atlassian Jira et Confluence. Cette intégration vous permet de créer	5 juin 2025

une base de connaissa
nces consultable unifiée en
synchronisant des projets
Jira complets et des espaces
Confluence dans. OpenSearc
h L'intégration préserve la
pertinence en temps réel
grâce à une surveillance
continue et à une synchroni
sation automatique des mises
à jour, tout en offrant des
options de filtrage flexibles
pour des projets, des types de
problèmes et des contenus
spécifiques. Pour plus
d'informations, consultez la
section Utilisation d'un pipeline
d' OpenSearch ingestion avec
Atlassian Services.

Gestion personnalisée des plugins avec AWS CLI	Vous pouvez désormais gérer les plugins personnalisés pour Amazon OpenSearch Service à l'aide du AWS CLI. Cette fonctionnalité vous permet d'installer, de mettre à jour, de sécuriser et de désinstaller des plugins personnalisés via un flux de travail en ligne de commande rationalisé. Vous pouvez également sécuriser vos plugins à l'aide de AWS KMS clés et gérer les mises à niveau des plugins lors de la mise à jour vers des versions plus récentes de OpenSearch. Pour plus d'informations, consultez Gestion des plugins personnalisés à l'aide AWS CLI de l' AWS KMSintégration des packages personnalisés Amazon OpenSearch Service	27 mai 2025
Support pour OpenSearch 2.18 et 2.19	Amazon OpenSearch Service prend en charge les OpenSearch versions 2.18 et 2.19. Pour plus d'informations, consultez les notes de mise à jour OpenSearch 2.19 et 2.18. Pour plus d'informations sur les opérations OpenSearch 2.18 et 2.19, consultez la référence de l'OpenSearch API REST ou la référence de l'API du plugin spécifique.	30 avril 2025

Nouveau support pour la spécification des rôles du pipeline dans OpenSearch Ingestion	Lorsque vous créez ou mettez à jour un pipeline dans Amazon OpenSearch Ingestion, vous ne spécifiez plus de rôle de pipeline dans une configuration de pipeline au format YAML. Au lieu de cela, dans la console, vous spécifiez le rôle dans un nouveau champ de rôle dans le pipeline. Si vous utilisez le AWS CLI, vous utilisez un nouveau `--pipeline-role-arn` paramètre. Pour plus d'informations, consultez les rubriques suivantes : • Configuration des rôles et des utilisateurs dans Amazon OpenSearch Ingestion • Création de pipelines OpenSearch Amazon Ingestion • Tutoriel : Ingestion de données dans un domaine à l'aide d'Amazon OpenSearch Ingestion • Tutoriel : Ingestion de données dans une collection à l'aide d'Amazon OpenSearch Ingestion • CreatePipeline • UpdatePipeline	22 avril 2025

Nouvelle rubrique	La rubrique de résolution des problèmes L'installation du plugin personnalisé échoue en raison de la compatibilité des versions. Indiquez les étapes à suivre lorsque vous rencontrez l'erreur`PluginValidationFailureReason : The provided plugin could not be loaded` . Cette erreur est généralement le résultat d'une incompatibilité de version entre le plugin et l' OpenSearch instance en cours d'exécution.	21 avril 2025
Nouvelle fonctionnalité : compatibilité de l' OpenSearch interface utilisateur avec la recherche entre clusters	OpenSearch L'interface utilisateur est désormais compatible avec la recherche entre clusters. Cela vous permet d'utiliser l' OpenSearch interface utilisateur d'une seule Région AWS pour accéder aux clusters d'une autre région. Cela se fait en le configurant en tant que cluster distant connecté à un cluster de la même région. Pour plus d'informations, voir Accès aux données entre régions et entre comptes avec recherche entre clusters.	16 avril 2025

Contenu révisé : « Utilisation de l' OpenSearch interface utilisateur dans Amazon OpenSearch Service »	Nous avons révisé et développé le chapitre Utilisation de l'OpenSearch interface utilisateur dans Amazon OpenSearch Service, y compris des procédures étendues, des détails de configuration supplémentaires et un nouvel historique des versions du support Amazon OpenSearch Service pour OpenSearch l'interface utilisateur.	15 avril 2025
Développeur Amazon Q pour Amazon OpenSearch Service	L'intégration d'Amazon Q à Amazon OpenSearch Service offre les fonctionnalités génératives suivantes : • Génération de visualisations à l'aide du langage naturel • Afficher les résumés et les informations sur les alertes • Consultez les résumés des résultats de requêtes générés par Amazon Q sur la page Discover • Afficher les détecteurs d'anomalies recommandés • Accédez au chat Amazon Q pour les OpenSearch questions connexes	31 mars 2025

La politique AmazonOpenSearchServiceRolePolicy gérée a été mise à jour	La politique permet OpenSearch de mettre à jour l'étendue d'accès de toute AWS IAM Identity Center application uniquement gérée par OpenSearch.	28 mars 2025
OR2 et OM2 instances pour Amazon OpenSearch Service	Amazon OpenSearch Service prend désormais en charge OR2 et instaure OM2 des instances. Selon les benchmarks internes, le débit d'indexation OR2 était jusqu'à 26 % supérieur à celui du R7 OR1 et 70 % supérieur à celui de R7. OM2 ont montré jusqu'à 15 % de mieux que le m7G OR1 et 66 % de plus.	25 mars 2025
Requête directe pour CloudWatch Logs and Security Lake	Amazon OpenSearch Service prend désormais en charge les requêtes directes pour interroger les données dans CloudWatch Logs et Security Lake.	1er décembre 2024
Indices K-nn	À partir de OpenSearch la version 2.17, vous pouvez déplacer les index K-nn vers les niveaux UltraWarm et cold storage.	13 novembre 2024

OpenSearch 2.17 assistance	Amazon OpenSearch Service prend désormais en charge OpenSearch la version 2.17. Cette version inclut toutes les fonctionnalités qui faisaient partie des versions 2.16 et 2.17. Pour plus d'informations, consultez les notes de mise à jour 2.16 et 2.17.	13 novembre 2024
OpenSearch 2.15 assistance	Amazon OpenSearch Service prend désormais en charge OpenSearch la version 2.15. Cette version inclut toutes les fonctionnalités qui faisaient partie des versions 2.14 et 2.15. Pour plus d'informations, consultez les notes de mise à jour 2.14 et 2.15.	11 octobre 2024
Nouveau rôle lié à un service	Amazon OpenSearch Service ajoute un rôle lié à un service appelé`AWSServiceRoleForOpensearchIngestionSelfManagedVpce` , qui permet à Amazon OpenSearch Ingestion d'envoyer des données métriques à des pipelines dotés de points de Amazon CloudWatch terminaison VPC autogérés.	12 juin 2024

Requêtes directes	Amazon OpenSearch Service prend désormais en charge l'exécution de requêtes directement sur les données stockées dans Amazon S3, sans qu'il soit nécessaire d'intégrer les données dans un OpenSearch index.	22 mai 2024
OpenSearch 2.13 assistance	Amazon OpenSearch Service prend désormais en charge OpenSearch la version 2.13. Cette version inclut toutes les fonctionnalités qui faisaient partie des versions 2.12 et 2.13. Pour plus d'informations, consultez les notes de mise à jour 2.12 et 2.13.	21 mai 2024
Support OpenSearch d'Amazon Ingestion pour Data Prepper version 2.7	Amazon OpenSearch Ingestion ajoute la prise en charge de la version 2.7 de Data Prepper. Pour plus d'informations, consultez les notes de mise à jour de la version 2.7.	4 avril 2024
Service AWS accès privé pour les OpenSearch collections sans serveur	Vous pouvez désormais accorder un accès spécifique Services AWS, tel qu'Amazon Bedrock, à vos collections OpenSearch Serverless dans le cadre d'une politique d'accès au réseau.	28 mars 2024

Mises à jour EBS sur place	Vous pouvez désormais apporter certaines modifications EBS à vos domaines sans déploiement bleu/vert dans Amazon Service. OpenSearch	14 février 2024
Visibilité des modifications de configuration	Vous pouvez désormais suivre les modifications de configuration du domaine dans la console Amazon OpenSearch Service et à l'aide de l'API de configuration.	6 février 2024
Disponibilité générale des collections de recherche vectorielle	Les collections de recherche vectorielle Amazon OpenSearch Serverless sont désormais généralement disponibles. Les améliorations notables suivantes ont été apportées au cours de la phase de prévisualisation : • Les collections de recherche vectorielle prennent désormais en charge des charges de travail contenant des milliards de vecteurs, chacun comportant jusqu'à 128 dimensions. • OpenSearch Les tableaux de bord prennent désormais en charge les collections de recherche vectorielle.	29 novembre 2023

OR1 instances	Amazon OpenSearch Service prend désormais en charge les types d' OR1 instances.	29 novembre 2023
Requêtes directes avec Amazon S3 (version préliminaire)	Les requêtes directes fournissent une solution entièrement gérée permettant de rendre les données transactionnelles disponibles dans Amazon OpenSearch Service quelques secondes après leur écriture dans un compartiment Amazon S3.	29 novembre 2023
Capacité de 10 TiB pour les collections de séries chronologiques	Amazon OpenSearch Serverless prend en charge jusqu'à 10 TiB de données d'index pour les collections de séries chronologiques. Cette version prend également en charge une capacité maximale autorisée de 200 OCUs pour tous les types de collections et la possibilité de désactiver les répliques de secours lorsque vous créez une collection.	29 novembre 2023
OpenSearch 2.11 assistance	Amazon OpenSearch Service prend désormais en charge OpenSearch la version 2.11. Cette version inclut toutes les fonctionnalités qui faisaient partie des versions 2.10 et 2.11. Pour plus d'informations, consultez les notes de mise à jour 2.10 et 2.11.	17 novembre 2023

Support OpenSearch d'Amazon Ingestion pour Data Prepper version 2.6	Amazon OpenSearch Ingestion ajoute la prise en charge de la version 2.6 de Data Prepper. Pour plus d'informations, consultez les notes de mise à jour 2.6. En outre, vous pouvez spécifier Amazon DynamoDB comme source de pipeline. Pour plus d'informations, consultez la section Utilisation d'un pipeline d' OpenSearch ingestion avec Amazon DynamoDB.	17 novembre 2023
Support OpenSearch d'Amazon Ingestion pour Data Prepper version 2.5	Amazon OpenSearch Ingestion ajoute la prise en charge de la version 2.5 de Data Prepper. Pour plus d'informations, consultez les notes de mise à jour de la version 2.5. En outre, vous pouvez désormais spécifier un domaine OpenSearch de service ou une collection OpenSearch sans serveur comme source de pipeline. Pour plus d'informations, consultez le plugin OpenSearch source dans la documentation de Data Prepper.	17 novembre 2023

CloudFormation modèle d'inférence à distance	Pour faciliter la configuration de l'inférence à distance pour la recherche sémantique, Amazon OpenSearch Service fournit un AWS CloudFormation modèle dans la console qui automatise le processus de mise en service des modèles pour vous.	7 novembre 2023
Mise à jour de la politique des rôles liés aux services	Ajoute les autorisations nécessaires à la politique de rôle liée au service pour attribuer et `AmazonOpenSearchServiceRolePolicy` annuler IPv6 l'attribution d'adresses. La politique obsolète d'Elasticsearch `AmazonElasticsearchServiceRolePolicy` a également été mise à jour pour garantir la rétrocompatibilité.	26 octobre 2023

Politiques relatives au cycle de vie d'Amazon OpenSearch Serverless	Amazon OpenSearch Serverless introduit des politiques relatives au cycle de vie des index afin de rationaliser la gestion de la conservation et de la suppression des données. Vous pouvez désormais utiliser APIs une interface de configuration dans la console pour définir des politiques de conservation des données pour les collections de séries chronologiques, éliminant ainsi le besoin de créer des index quotidiens ou des scripts pour supprimer les anciennes données.	25 octobre 2023
Prise en charge des instances Im4gn	Amazon OpenSearch Service prend désormais en charge les types d'instances IM4gn. Les instances Im4gn sont optimisées pour les charges de travail qui gèrent de grands ensembles de données et nécessitent une densité de stockage élevée par vCPU.	20 octobre 2023

Options administratives	Amazon OpenSearch Service propose désormais plusieurs options administratives qui fournissent un contrôle granulaire si vous devez résoudre des problèmes liés à votre domaine. Ces options incluent la possibilité de redémarrer le OpenSearch processus sur un nœud de données et la possibilité de redémarrer un nœud de données.	17 octobre 2023
Plug-ins optionnels	Amazon OpenSearch Service prend désormais en charge quatre nouveaux plug-ins d'analyse de langue : Nori (coréen), Sudachi (japonais), Pinyin (chinois) et STConvert Analysis (chinois), ainsi que le plug-in Amazon Personalize Search Ranking.	16 octobre 2023
OpenSearch 2.9 assistance	Amazon OpenSearch Service prend désormais en charge OpenSearch la version 2.9. Cette version inclut toutes les fonctionnalités qui faisaient partie des versions 2.8 et 2.9. Pour plus d'informations, consultez les notes de publication des versions 2.8 et 2.9.	2 octobre 2023

Connecteurs ML	Amazon OpenSearch Service ajoute la prise en charge des connecteurs d'apprent issage automatique (ML). Les connecteurs facilitent l'accès aux modèles de machine learning hébergés sur d'autres Services AWS plateformes d'apprentissage automatique (ML) ou sur des plateformes tierces.	6 septembre 2023
Amazon OpenSearch Ingestion ajoute la prise en charge de la version 2.4 de Data Prepper	Amazon OpenSearch Ingestion ajoute la prise en charge de la version 2.4 de Data Prepper. Pour plus d'informations, consultez les notes de mise à jour de la version 2.4. En outre, vous pouvez désormais spécifier Amazon Managed Streaming for Apache Kafka (Amazon MSK) comme source de pipeline.	31 août 2023
Capacité de 6 TiB pour les collections de séries chronolog iques	Amazon OpenSearch Serverless prend en charge jusqu'à 6 TiB de données d'index pour les collections de séries chronologiques. Cette version prend également en charge une capacité maximale autorisée de 100 OCUs pour les recherches et les collectio ns de séries chronologiques.	15 août 2023

Collections de recherche vectorielle	Amazon OpenSearch Serverless ajoute la possibilité de créer une collection de recherche vectorielle, que vous pouvez utiliser pour stocker des intégrations vectorielles afin de favoriser la similiarité et les recherches sémantiques.	26 juillet 2023
OpenSearch 2.7 assistance	Amazon OpenSearch Service prend désormais en charge OpenSearch la version 2.7. Cette version inclut toutes les fonctionnalités qui faisaient partie des versions 2.6 et 2.7. Pour plus d'informations, consultez les notes de mise à jour 2.6 et 2.7.	10 juillet 2023
Prise en charge de Data Prepper 2.3	Amazon OpenSearch Ingestion ajoute le support de Data Prepper version 2.3. Pour plus d'informations, consultez les notes de publication de la version 2.3. En outre, vous pouvez désormais spécifier Amazon Security Lake comme source de pipeline.	26 juin 2023

Multi-AZ avec mode veille	Amazon OpenSearch Service ajoute la possibilité de déployer un domaine dans trois zones de disponibilité (AZs), chaque zone de disponibilité contenant une copie complète des données et les nœuds de l'une de ces zones AZs faisant office de réserve. L'option de déploiement Multi-AZ avec veille assure une disponibilité de 99,99 % et des performances constantes en cas de défaillance de l'infrastructure.	3 mai 2023
Nouveau rôle lié à un service	Amazon OpenSearch Service ajoute un rôle lié à un service appelé`AWSServiceRoleForAmazonOpenSearchIngestionService` , qui permet à Amazon OpenSearch Ingestion d'envoyer des données métriques à. Amazon CloudWatch	26 avril 2023

OpenSearch Ingestion d'Amazon	Amazon OpenSearch Ingestion est un collecteur de données entièrement géré qui fournit des données de journalisation et de suivi en temps réel aux domaines de OpenSearch service et aux collections OpenSearch sans serveur. OpenSearch L'ingestion vous évite d'avoir à utiliser des solutions tierces telles que Logstash ou Jaeger pour ingérer des données dans vos domaines et collections.	26 avril 2023
OpenSearch 2.5 assistance	Amazon OpenSearch Service prend désormais en charge OpenSearch la version 2.5. Cette version inclut toutes les fonctionnalités qui faisaient partie des versions 2.4 et 2.5. Pour plus d'informations, consultez les notes de mise à jour 2.4 et 2.5.	13 mars 2023

Fenêtres de maintenance hors pointe	Amazon OpenSearch Service ajoute des périodes creuses, c'est-à-dire des plages horaires quotidiennes de 10 heures à faible trafic pendant lesquelles il peut planifier des mises à jour du logiciel de service et des optimisat ions Auto-Tune nécessitant un déploiement. blue/green Les mises à jour hors pointe permettent de minimiser la pression sur les nœuds principaux dédiés d'un cluster pendant les périodes de trafic élevé. Pour les nouveaux domaines créés après le 16 février, la période creuse est automatiq uement configurée entre 22 h 00 et 8 h 00, heure locale. Pour les domaines existants , vous devez activer explicite ment la fenêtre.	16 février 2023
Configuration de l'authent ification SAML lors de la création du domaine	Amazon OpenSearch Service prend désormais en charge la configuration de l'authent ification SAML lors de la création du domaine. Auparavant, vous deviez configurer les options SAML après la création du domaine.	1er février 2023

Réindexation à distance pour les domaines VPC	Amazon OpenSearch Service ajoute l'option pour une connexion de point de terminaison VPC entre deux domaines. Vous pouvez désormais utiliser la réindexation à distance pour copier des index d'un domaine VPC à un autre sans proxy inverse. Vos domaines VPC doivent exécuter le logiciel de service R20221114 ou une version ultérieure pour utiliser cette fonction.	31 janvier 2023

Disponibilité générale d'Amazon OpenSearch Serverless	Amazon OpenSearch Serverless est désormais disponible pour tous. Les améliorations notables suivantes ont été apportées au cours de la phase de prévisualisation : • La capacité peut désormais être réduite au minimum configuré en OCUs cas de diminution du trafic sur le point de terminaison de collecte. • Le maximum autorisé OCUs pour l'indexation et la recherche a été augmenté de 20 à 50. Chaque OCU comprend suffisamment de stockage éphémère à chaud pour 120 Gio de données d'index. • Vous pouvez désormais configurer les paramètres d'accès aux données tout en créant des collections, plutôt que de devoir les configurer dans un flux distinct.	25 janvier 2023

Test à blanc asynchrone	Amazon OpenSearch Service prend désormais en charge l'exécution à sec asynchrone, ce qui vous permet d'effectuer un contrôle de validation avant de modifier la configuration et de vous informer si vos modifications entraîneront un blue/green déploiement.	19 janvier 2023
Nouveau rôle lié à un service	Amazon OpenSearch Service ajoute un rôle lié à un service appelé`AWSServiceRoleForAmazonOpenSearchServerless` , qui permet à OpenSearch Serverless d'envoyer des données métriques à. Amazon CloudWatch	29 novembre 2022
Aperçu d'Amazon OpenSearch Serverless	Amazon OpenSearch Serverless est une configuration sans serveur à la demande, à mise à l'échelle automatique, pour Amazon OpenSearch Service. Le mode Serverless élimine les complexités opérationnelles liées au provisionnement, à la configuration et au réglage de vos clusters. OpenSearch	29 novembre 2022

OpenSearch 2.3 assistance	Amazon OpenSearch Service prend désormais en charge OpenSearch la version 2.3. Cette version inclut toutes les fonctionnalités qui faisaient partie des versions 2.0, 2.1 et 2.2. Pour plus d'informations, veuillez consulter les notes de mise à jour 2.0, 2.1, 2.2 et 2.3. La version 2.3 contient une modification majeure. Pour plus d'informations, veuillez consulter la rubrique Chemins de mise à niveau pris en charge.	15 novembre 2022
Prise en charge du plugin Notifications	Amazon OpenSearch Service prend désormais en charge le plug-in Notifications, qui offre un emplacement central pour toutes vos notifications provenant de OpenSearch plugins. À partir de la version 2.0, les destinations d'alerte sont devenues obsolètes et ont été remplacées par des canaux de notification.	15 novembre 2022

Prise en charge de Kibana 7.1.1	Les domaines Amazon OpenSearch Service exécutant Elasticsearch 7.1 prennent désormais en charge le dernier correctif pour Kibana 7.1.1, qui corrige des bogues et améliore la sécurité. Lorsque vous mettez à jour vos domaines 7.1 vers le logiciel de service R20221114, le OpenSearch service les met automatiquement à niveau vers cette version de correctif.	15 novembre 2022
Prise en charge de Kibana 6.8.13	Les domaines Amazon OpenSearch Service exécutant Elasticsearch 6.8 prennent désormais en charge le dernier correctif pour Kibana 6.8.13, qui corrige des bogues et améliore la sécurité. Lorsque vous mettez à jour vos domaines 6.8 vers le logiciel de service R20221114, le OpenSearch service les met automatiquement à niveau vers cette version de correctif.	15 novembre 2022

Prise en charge de Kibana 6.3.2	Les domaines Amazon OpenSearch Service exécutant Elasticsearch 6.3 prennent désormais en charge le dernier correctif pour Kibana 6.3.2, qui corrige des bogues et améliore la sécurité. Lorsque vous mettez à jour vos domaines 6.3 vers le logiciel de service R20221114, le OpenSearch service les met automatiquement à niveau vers cette version de correctif.	15 novembre 2022

AWS PrivateLink	Avec les points de OpenSearch terminaison VPC gérés par Amazon Service, vous pouvez vous connecter directement aux domaines Service OpenSearch VPC en utilisant un point de terminaison VPC d'interface au lieu de vous connecter via Internet. Un point de OpenSearch terminaison VPC géré par un service n'est accessible que dans le VPC où le point de terminaison est provisionné, ou depuis tout point apparenté au VPC où le point de terminaison est VPCs provisionné, comme le permettent les tables de routage et les groupes de sécurité. Votre domaine VPC doit exécuter le logiciel de service R20220928 ou version ultérieure pour se connecter à un point de terminaison d'un VPC d'interface.	7 novembre 2022

Correctifs de bogues et améliorations de performances	Le logiciel de service R20220928 inclut des corrections de bogues et des améliorations de performances, y compris une journalisation SAML améliorée. La mise à jour remplace également le locataire par défaut en `Global` plutôt que `Private`.	3 octobre 2022
Référence d'API améliorée	Amazon OpenSearch Service propose une référence d'API de configuration améliorée et complète. Les nouvelles références contiennent toutes les actions et tous les types de données disponibles, des exemples de syntaxe de demande et de réponse, ainsi que des liens vers les références de SDK correspondantes pour tous les langages pris en charge.	13 septembre 2022
Validation bleu/vert	Amazon OpenSearch Service effectue désormais un contrôle de validation avant les blue/green déploiements et détecte des erreurs de validation si votre domaine n'est pas éligible à une mise à jour.	16 août 2022

OpenSearch 1.3 assistance	Amazon OpenSearch Service prend désormais en charge OpenSearch la version 1.3. Pour plus d'informations, consultez Notes de mise à jour de la version 1.3.	27 juillet 2022
Prise en charge du plugin ML Commons	Amazon OpenSearch Service ajoute la prise en charge du plugin ML Commons, qui fournit un ensemble d'algorithmes d'apprentissage automatique courants par le biais d'appels d'API de transport et d'API REST. Vous pouvez également interagir avec le plugin ML Commons via des commandes PPL.	27 juillet 2022
Prise en charge des volumes gp3	Amazon OpenSearch Service ajoute la prise en charge du type de volume SSD à usage général gp3 EBS. Vous pouvez spécifier des IOPS provisionnés et des débits supplémentaires lorsque vous créez ou modifiez le domaine.	26 juillet 2022
Documentation améliorée sur les bonnes pratiques	La documentation Amazon OpenSearch Service fournit des meilleures pratiques opérationnelles améliorée s et des recommandations générales pour la création et l'exploitation de domaines OpenSearch de service.	6 juillet 2022

Intégration à Service Quotas	Vous pouvez désormais consulter les quotas d'Amazon OpenSearch Service et demander des augmentations de quotas depuis la console Service Quotas.	29 juin 2022
Contrôle d'accès basé sur des balises pour l'API OpenSearch	Vous pouvez désormais utiliser des balises pour contrôler l'accès au OpenSearch APIs. Auparavant, vous ne pouviez utiliser des balises que pour contrôler l'accès à l'API de configuration.	16 juin 2022
Recherche entre clusters entre les régions	La recherche entre clusters est désormais prise en charge Régions AWS tant que les deux domaines exécutent Elasticsearch version 7.10 ou ultérieure, ou une version quelconque de. OpenSearch	14 juin 2022
Prise en charge de Single Kibana 5.6	Amazon OpenSearch Service ajoute la prise en charge de la version 5.6.16 unique de Kibana. Avec Single Kibana 5.6.16, vous pouvez utiliser Kibana 5.6 comme frontend tout en vous connectant aux versions Elasticsearch 5.1, 5.3, 5.5 et 5.6. Vous devez être sur le logiciel de service R20220323 ou plus récent pour utiliser Single Kibana 5.6.	4 avril 2022

R20220323-P1	Amazon OpenSearch Service a récemment publié la mise à jour logicielle de service R20220323, mais la mise à jour a ensuite été annulée en raison d'un problème. Nous vous recommandons de mettre à jour vos domaines vers la version corrective R20220323-P1 ou ultérieure, qui résout ce problème.	4 avril 2022
OpenSearch 1.2 assistance	Amazon OpenSearch Service prend désormais en charge OpenSearch la version 1.2. Pour plus d'informations, consultez les notes de mise à jour de la version 1.2.	4 avril 2022
Observabilité	L'installation par défaut de OpenSearch Dashboards for Amazon OpenSearch Service inclut le plugin Observabi lity, que vous pouvez utiliser pour visualiser les événement s pilotés par les données à l'aide du langage PPL (Piped Processing Language) afin d'explorer et d'interroger vos données. Le plug-in nécessite la version OpenSearch 1.2 ou ultérieure et le logiciel de service R20220323 ou version ultérieure.	4 avril 2022

Prise en charge de Kibana 7.7.1	Les domaines Amazon OpenSearch Service exécutant Elasticsearch 7.7 prennent désormais en charge le dernier correctif pour Kibana 7.7, qui corrige des bogues et améliore la sécurité. Lorsque vous mettez à jour vos domaines 7.7 vers le logiciel de service R20220323 ou version ultérieure, le OpenSearch service les met automatiquement à niveau vers cette version de correctif.	4 avril 2022

Changements de métrique de la charge mémoire de la JVM	Amazon OpenSearch Service a modifié la logique des `JVMMemoryPressure` CloudWatch métriques afin de refléter plus préciséme nt l'utilisation de la mémoire. Auparavant, les métriques ne prenaient en compte que l'ancienne génération du groupe de mémoire du tas JVM. Avec ce changement, la métrique prend également en compte la nouvelle générati on du groupe de mémoire. Après avoir mis à jour votre domaine vers le logiciel de service R20220323, il se peut que vous constatiez une augmentation des métriques `JVMMemoryPressure` , `MasterJVMMemoryPre ssure` et/ou `WarmJVMMe moryPressure` .	4 avril 2022
Dictionnaires personnalisés avec le plugin IK (Chinese) Analysis	Amazon OpenSearch Service prend désormais en charge l'utilisation de dictionnaires personnalisés avec le plugin IK (Chinese) Analysis.	4 avril 2022

Réplication croisée entre clusters sur les domaines existants	Amazon OpenSearch Service a supprimé la limitation selon laquelle vous ne pouvez implémenter la recherche et la réplication entre clusters que sur des domaines créés le 3 juin 2020 ou après cette date. Vous pouvez désormais activer ces fonctions sur tous les domaines, quel que soit le moment où ils ont été créés. Les deux domaines doivent être sur le logiciel de service R20220323 ou version ultérieure.	4 avril 2022
Visibilité des déploiements bleu/vert	Amazon OpenSearch Service offre désormais une meilleure visibilité sur la progression des déploiements bleu/vert. Vous pouvez surveiller ces détails dans la console ou à l'aide de l'API de configuration.	27 janvier 2022

Contrôle précis des accès sur des domaines existants	Vous pouvez désormais activer le contrôle précis des accès sur les domaines existants. Vous pouvez activer une période de migration temporaire pour les stratégies d'accès Open/basées sur l'IP afin de vous assurer que les utilisateurs peuvent continuer à accéder à votre domaine pendant que vous créez et mappez les rôles. L'activation du contrôle précis des accès sur les domaines existants nécessite le logiciel de service R20211203 ou une version ultérieure.	6 janvier 2022
Rôles de OpenSearch tableau de bord renommés	Avec le logiciel de service R20211203, le rôle `kibana_user` a été renommé en `opensearch_dashboards_user`, et `kibana_read_only` a été renommé en `opensearch_dashboards_read_only`. Cette modification s'applique à tous les 1 nouvellement créés OpenSearch . domaines x. Pour les OpenSearch domaines existants que vous mettez à niveau vers le logiciel de service R20211203, les rôles restent les mêmes.	4 janvier 2022

OpenSearch 1.1 assistance	Amazon OpenSearch Service prend désormais en charge OpenSearch la version 1.1. Pour plus d'informations, consultez les notes de mise à jour de la version 1.1.	4 janvier 2022
Éditeur visuel ISM	L'installation par défaut de OpenSearch Dashboards for Amazon OpenSearch Service prend désormais en charge l'éditeur visuel pour les politiques ISM. Cette fonctionn alité nécessite la OpenSearch version 1.1 ou ultérieure.	4 janvier 2022
Mise à jour de la prévention du problème de l'adjoint confus entre services	Amazon OpenSearch Service prend en charge l'utilisa tion des clés de contexte de condition `aws:Sourc eAccount` globale `aws:SourceArn` et des clés de contexte dans les politiques de ressources IAM afin d'éviter le problème de confusion lié aux adjoints. Vous devez être sur le logiciel de service R20211203 ou plus récent pour utiliser ces clés de condition.	4 janvier 2022

Correctif Log4j	Le logiciel de service R20211203-P2 met à jour la version de Log4j utilisée dans OpenSearch Service conformément aux recommandations des CVE-2021-44228 et CVE-2021-45046. Le correctif s'applique aux domaines exécutant toutes les versions d'Elasticsearch OpenSearch et d'Elasticsearch. OpenSearch Le service continuera à mettre à jour les différentes versions de Log4j en interne, et il ne sera pas nécessairement limité à la dernière version de Log4j. La version de Log4j sur votre domaine dépend de la version du logiciel que le domaine exécute. Toutefois, quelle que soit la version de Log4j, tant que vous exécutez la version R20211203-P2 ou une version ultérieure, vos domaines contiennent la mise à jour de Log4j requise pour résoudre les problèmes présentés par CVE-2021-44228 et CVE-2021-45046.	15 décembre 2021

Réplication inter-clusters (CCR)	La réplication entre clusters vous permet de répliquer des index, des mappages et des métadonnées d'un domaine de OpenSearch service à un autre. La réplication entre clusters nécessite un domaine exécutant Elasticsearch 7.10 ou 1.1 ou OpenSearch version ultérieure.	5 octobre 2021
Nouvelles AWS politiques gérées	Le lancement d'Amazon OpenSearch Service inclut de nouvelles politiques AWS gérées et la dépréciation des anciennes politiques.	8 septembre 2021
Prise en charge de Kibana 6.4.3	Les domaines Amazon OpenSearch Service exécutant l'ancienne version 6.4 d'Elasticsearch prennent désormais en charge le dernier correctif pour Kibana 6.4, qui corrige des bogues et améliore la sécurité. OpenSearch Le service mettra automatiquement à niveau les domaines vers cette version de correctif.	8 septembre 2021

Flux de données	Amazon OpenSearch Service ajoute la prise en charge des flux de données, ce qui simplifie le processus de gestion des données chronologiques. Votre domaine doit exécuter la OpenSearch version 1.0 ou une version ultérieure pour utiliser les flux de données.	8 septembre 2021
Amazon OpenSearch Service	AWS renomme Amazon OpenSearch Service pour supprimer l'ancienne marque « Elasticsearch ». Amazon OpenSearch Service prend en charge les systèmes OpenSearch d'exploitation Elasticsearch OSS existants . Lorsque vous créez un cluster, vous pouvez choisir le moteur de recherche à utiliser. OpenSearch Le service offre une large compatibilité avec Elasticsearch OSS 7.10, la dernière version open source du logiciel.	8 septembre 2021

Stockage à froid	Le stockage à froid est un nouveau niveau de stockage pour les données historiques ou à accès peu fréquent. Les index à froid occupent uniquement le stockage S3 et aucun calcul ne leur est attaché. Le stockage à froid nécessite un domaine exécutant Elasticsearch 7.9 ou version ultérieure et un logiciel de service R20210426 ou version ultérieure.	13 mai 2021
Instances Graviton basées sur ARM	Amazon OpenSearch Service prend désormais en charge les types d'instances Graviton basés sur ARM (M6G, C6G, R6G et R6GD). Les types d'instances Graviton sont disponibles dans les domaines nouveaux et existants exécutant Elasticsearch 7.9 ou version ultérieure et sur le logiciel de service R20210331 ou version ultérieure.	4 mai 2021

Modèles ISM	Amazon OpenSearch Service prend également en charge les modèles ISM, qui vous permettent d'associer automatiquement une politique ISM à un index si celui-ci correspond à un modèle défini dans la politique. Les modèles ISM nécessitent un logiciel de service R20210426 ou version ultérieure. Cette mise à jour rend également obsolète le paramètre `policy_id` , ce qui signifie que vous ne pouvez plus utiliser de modèles d'index pour appliquer des politique s ISM aux index nouvellem ent créés. La mise à jour introduit un changement radical pour les CloudForm ation modèles existants utilisant ce paramètre.	27 avril 2021
Prise en charge d'Elastic search 7.10	Amazon OpenSearch Service prend désormais en charge la version 7.10 d'Elasticsearch. Pour plus d'informations, consultez Notes de mise à jour de la version 7.10.	21 avril 2021

Recherche asynchrone	Amazon OpenSearch Service prend désormais en charge la recherche asynchrone, ce qui vous permet d'exécuter des demandes de recherche en arrière-plan. La recherche asynchrone nécessite un domaine exécutant Elasticsearch 7.10 ou version ultérieure et un logiciel de service R20210331 ou version ultérieure.	21 avril 2021
Contrôle d'accès basé sur l'identification pour l'API de configuration	Vous pouvez désormais utiliser des AWS balises pour contrôler l'accès à l'API de configuration Amazon ES.	2 mars 2021
Auto-Tune	Amazon OpenSearch Service ajoute Auto-Tune, qui utilise les indicateurs de performance et d'utilisation de votre cluster pour suggérer des modifications des paramètres JVM sur vos nœuds. Auto-Tune nécessite un domaine exécutant Elasticsearch 6.7 ou version ultérieure et un logiciel de service R20201117 ou version ultérieure.	24 février 2021

Trace Analytics	L'installation par défaut de Kibana pour Amazon OpenSearch Service inclut désormais le plug-in d'analyse des traces, qui vous permet de surveiller les données de suivi de vos applications distribué es. Le plugin nécessite un domaine exécutant Elasticse arch 7.9 ou version ultérieur e et un logiciel de service R20210201 ou version ultérieure.	17 février 2021
Métriques de partition	Amazon OpenSearch Service ajoute les CloudWatc h mesures suivantes pour suivre l'état des partitions :`Shards.ac tive` ,,`Shards.un assigned` , `Shards.de layedUnassigned` `Shards.activePrima ry` `Shards.initializin g` ,`Shards.relocating` . Les métriques sont disponibles sur les domaines exécutant le logiciel de service R20210201 ou version ultérieure.	17 février 2021

Rapports Kibana	L'installation par défaut de Kibana pour Amazon OpenSearch Service prend désormais en charge les rapports à la demande pour les pages Discover, Visualize et Dashboard. Cette fonction nécessite un domaine exécutant Elasticsearch 7.9 ou version ultérieure et un logiciel de service R20210201 ou version ultérieure.	17 février 2021
Prise en charge de Kibana 5.6.16	Les domaines Amazon OpenSearch Service exécutant Elasticsearch 5.6 prennent désormais en charge le dernier correctif pour Kibana 5.6, qui corrige des bogues et améliore la sécurité. Amazon ES mettra automatiquement à niveau les domaines vers cette version de correctif.	17 février 2021
Chiffrement pour les domaines existants	Amazon OpenSearch Service prend désormais en charge l'activation du chiffrement des données au repos et du node-to-node chiffrement sur les domaines existants exécutant Elasticsearch 6.7 ou version ultérieure. Une fois ces paramètres activés, vous ne pouvez pas les désactiver.	27 janvier 2021

Réindexation à distance	Amazon OpenSearch Service prend désormais en charge la réindexation à distance, qui vous permet de migrer des index depuis des domaines distants. Cette fonction nécessite le logiciel de service R20201117 ou version ultérieure.	24 novembre 2020
Langage PPL (Piped Processing Language)	Amazon OpenSearch Service prend désormais en charge le langage PPL (Piped Processin g Language), un langage de requête qui vous permet d'utiliser la syntaxe pipe (|) pour interroger les données stockées dans Elasticsearch. Cette fonction nécessite le logiciel de service R20201117 ou version ultérieure. Pour en savoir plus, veuillez consulter la section .	24 novembre 2020
Blocs-notes Kibana	Amazon OpenSearch Service prend désormais en charge les blocs-notes Kibana, ce qui vous permet de combiner des visualisations en direct et du texte narratif dans une seule interface. Cette fonction nécessite le logiciel de service R20201117 ou version ultérieure.	24 novembre 2020

Diagrammes de Gantt	L'installation par défaut de Kibana pour Amazon OpenSearch Service prend désormais en charge un nouveau type de visualisation, les diagrammes de Gantt. Cette fonction nécessite le logiciel de service R20201117 ou version ultérieure.	24 novembre 2020
Prise en charge d'Elasticsearch 7.9	Amazon OpenSearch Service prend désormais en charge la version 7.9 d'Elasticsearch. Pour plus d'informations, consultez Notes de mise à jour de la version 7.9.	24 novembre 2020
Mises à jour de la détection d'anomalies	La détection des anomalies pour Amazon OpenSearch Service ajoute la prise en charge de la cardinalité élevée, ce qui vous permet de classer les anomalies selon une dimension telle que l'adresse IP, l'identifiant du produit, le code du pays, etc. Cette fonction nécessite le logiciel de service R20201117 ou version ultérieure.	24 novembre 2020

Mises à jour dynamiques du dictionnaire	Amazon OpenSearch Service vous permet désormais de mettre à jour vos analyseurs de recherche sans avoir à les réindexer. Vous pouvez mettre à jour les fichiers du dictionnaire sur tout ou partie de vos domaines. Amazon ES suit les versions de package au fil du temps afin d'obtenir un historique des modifications. Cette fonction nécessite le logiciel de service R20201019 ou une version ultérieure.	17 novembre 2020
Points de terminaison personnalisés	Amazon OpenSearch Service prend désormais en charge les points de terminaison personnalisés, qui vous permettent d'attribuer une nouvelle URL à votre domaine Amazon ES. Si vous permutez des domaines, vous pouvez conserver la même URL. Cette fonction nécessite le logiciel de service R20201019 ou une version ultérieure.	5 novembre 2020

Nouveaux plugins de langues	Amazon OpenSearch Service prend désormais en charge les plug-ins IK (chinois) Analysis, Vietnamese Analysis et Thai Analysis sur les domaines exécutant Elasticsearch 7.7 ou version ultérieure avec le logiciel de service R20201019 ou version ultérieure.	28 octobre 2020
Prise en charge d'Elastic search 7.8	Amazon OpenSearch Service prend désormais en charge la version 7.8 d'Elasticsearch. Pour plus d'informations, consultez Notes de mise à jour de la version 7.8.	28 octobre 2020
Authentification SAML pour Kibana	Amazon OpenSearch Service prend désormais en charge l'authentification SAML pour Kibana, qui vous permet de faire appel à des fournisse urs d'identité tiers pour vous connecter à Kibana, gérer un contrôle d'accès précis, effectuer des recherches dans vos données et créer des visualisations. Cette fonction nécessite le logiciel de service R20201019 ou une version ultérieure.	27 octobre 2020
Instances T3	Amazon OpenSearch Service prend désormais en charge les types d'`t3.medium` instance `t3.small` et.	23 septembre 2020

Journaux d'audit	Amazon OpenSearch Service prend désormais en charge les journaux d'audit de vos données, ce qui vous permet de suivre les tentatives de connexion infructueuses, l'accès des utilisateurs aux index, aux documents et aux champs, et bien plus encore. Cette fonction nécessite le logiciel de service R20200910 ou version ultérieure.	16 septembre 2020
UltraWarm mises à jour	UltraWarm for Amazon OpenSearch Service ajoute de nouvelles métriques, de nouveaux paramètres, une file d'attente de migration plus importante et une API d'annulation. Ces mises à jour nécessitent le logiciel de service R20200910 ou version ultérieure. Pour en savoir plus, consultez .	14 septembre 2020
Learning to Rank	Amazon OpenSearch Service prend désormais en charge le plugin open source Learning to Rank, qui vous permet d'utiliser les technologies d'apprentissage automatique pour améliorer la pertinenc e des recherches. Cette fonction nécessite le logiciel de service R20200721 ou version ultérieure.	27 juillet 2020

Similarité cosinus k-NN	k-Nearest Neighbor (k-NN) vous permet désormais de rechercher les « voisins les plus proches » par similarit é cosinus en plus de la distance euclidienne. Cette fonction nécessite le logiciel de service R20200721 ou version ultérieure.	23 juillet 2020
Compression gzip	Amazon OpenSearch Service prend désormais en charge la compression gzip pour la plupart des requêtes et réponses HTTP, ce qui permet de réduire la latence et de préserver la bande passante. Cette fonction nécessite le logiciel de service R20200721 ou version ultérieure.	23 juillet 2020
Prise en charge d'Elastic search 7.7	Amazon OpenSearch Service prend désormais en charge la version 7.7 d'Elasticsearch. Pour plus d'informations, consultez Notes de mise à jour de la version 7.7.	23 juillet 2020
Service de cartographie Kibana	L'installation par défaut de Kibana pour Amazon OpenSearch Service inclut désormais un serveur de carte WMS, à l'exception des domaines des régions de l'Inde et de la Chine.	18 juin 2020

Améliorations SQL	Le support SQL pour Amazon OpenSearch Service prend désormais en charge de nombreuses nouvelles opérations, une interface utilisateur Kibana dédiée à l'exploration des données et une CLI interactive.	3 juin 2020
Recherche inter-clusters	Amazon OpenSearch Service vous permet d'effectuer des requêtes et des agrégations entre clusters sur plusieurs domaines connectés.	3 juin 2020
Détection des anomalies	Amazon OpenSearch Service vous permet de détecter automatiquement les anomalies en temps quasi réel.	3 juin 2020
UltraWarm	UltraWarm le stockage pour Amazon OpenSearch Service a quitté la version préliminaire publique et est désormais disponible pour tous. La fonctionnalité prend désormais en charge un plus large éventail de versions et Régions AWS. Pour de plus amples informations, veuillez consulter .	5 mai 2020

Dictionnaires personnalisés	Amazon OpenSearch Service vous permet de télécharger des fichiers de dictionnaire personnalisés à utiliser avec votre cluster. Ces fichiers améliorent les résultats de votre recherche en demandant à Elasticsearch d'ignorer certains mots les plus fréquents ou de traiter des termes comme des équivalents.	21 avril 2020
Prise en charge d'Elasticsearch 7.4	Amazon OpenSearch Service prend désormais en charge la version 7.4 d'Elasticsearch. Pour plus d'informations, consultez Versions prises en charge.	12 mars 2020
k-NN	Amazon OpenSearch Service ajoute la prise en charge de la recherche K-Nearest Neighbor (K-nn). K-nn nécessite le logiciel de service R20200302 ou version ultérieure.	3 mars 2020
Gestion d'états des index	Amazon OpenSearch Service ajoute la gestion de l'état des index (ISM), qui vous permet d'automatiser les tâches de routine, telles que la suppression d'index lorsqu'ils atteignent un certain âge. Cette fonctionnalité nécessite le logiciel de service R20200302 ou une version ultérieure.	3 mars 2020

Prise en charge d'Elastic search 5.6.16	Amazon OpenSearch Service prend désormais en charge le dernier correctif pour la version 5.6, qui ajoute des corrections de bogues et améliore la sécurité. Amazon ES mettra automatiquement à niveau les domaines 5.6 existants vers cette version. Notez que cette version d'Elasticsearch indique par erreur sa version comme 5.6.17.	2 mars 2020
Contrôle précis des accès	Amazon OpenSearch Service prend désormais en charge le contrôle d'accès détaillé, qui assure la sécurité au niveau de l'index, du document et du champ, la mutualisation de Kibana et l'authentification HTTP de base optionnelle pour votre cluster.	11 février 2020

UltraWarm stockage (aperçu)	Amazon OpenSearch Service ajoute UltraWarm un nouveau niveau de stockage à chaud qui utilise Amazon S3 et une solution de mise en cache sophistiquée pour améliorer les performances. Pour les indices sur lesquels vous n'écrivez pas activement et que vous interrogez moins fréquemment, le UltraWarm stockage permet de réduire considérablement les coûts par GiB.	3 décembre 2019
Fonctions de chiffrement pour les régions Chine	Le chiffrement des données au repos et le node-to-node chiffrement sont désormais disponibles dans les régions de `cn-north-1` Chine (Pékin) et de `cn-northwest-1` Chine (Ningxia).	20 novembre 2019
Require HTTPS (Exiger HTTPS)	Vous pouvez désormais exiger que tout le trafic vers vos domaines Amazon ES arrive via HTTPS. Lorsque vous configurez votre domaine, cochez la case Require HTTPS (Exiger HTTPS). Cette fonctionnalité nécessite le logiciel de service R20190808 ou version ultérieure.	3 octobre 2019

Prise en charge d'Elastic search 7.1 et 6.8	Amazon OpenSearch Service prend désormais en charge les versions 7.1 et 6.8 d'Elastic search. Pour plus d'informa tions, consultez Versions prises en charge.	13 août 2019
Instantanés toutes les heures	Plutôt que des instantan és quotidiens, Amazon OpenSearch Service prend désormais des instantan és toutes les heures des domaines exécutant Elasticse arch 5.3 et versions ultérieur es afin que vous puissiez effectuer des sauvegard es plus fréquentes à partir desquelles restaurer vos données.	8 juillet 2019
Prise en charge d'Elastic search 6.7	Amazon OpenSearch Service prend désormais en charge la version 6.7 d'Elasticsearch. Pour plus d'informations, consultez Versions prises en charge.	29 mai 2019
Prise en charge de SQL	Amazon OpenSearch Service vous permet désormais d'interroger vos données à l'aide de SQL. La prise en charge de SQL nécessite le logiciel de service R20190418 ou version ultérieure.	15 mai 2019

Types d'instances de la série 5	Amazon OpenSearch Service prend désormais en charge les types d'instances M5, C5 et R5. Par rapport aux types d'instance de la génération précédente, ces nouveaux types offrent de meilleure s performances à des prix inférieurs. Pour plus d'informa tions, consultez Limites .	24 avril 2019
Prise en charge d'Elastic search 6.5	Amazon OpenSearch Service prend désormais en charge la version 6.5 d'Elasticsearch.	8 avril 2019
Alerte	Les alertes pour Amazon OpenSearch Service vous avertissent lorsque les données d'un ou de plusieurs indices Amazon ES répondent à certaines conditions. Les alertes nécessitent le logiciel de service R20190221 ou version ultérieure.	25 mars 2019
Prise en charge de trois zones de disponibilité	Amazon OpenSearch Service prend désormais en charge trois zones de disponibilité dans de nombreuses régions. Cette version inclut également une expérience de console rationalisée. Cette fonctionn alité multi-AZ nécessite un logiciel de service R20181023 ou une version ultérieure.	7 février 2019

Prise en charge d'Elastic search 6.4	Amazon OpenSearch Service prend désormais en charge la version 6.4 d'Elasticsearch.	23 janvier 2019
Clusters à 200 nœuds	Amazon ES vous permet désormais de créer des clusters avec jusqu'à 200 nœuds de données pour un stockage total de 3 Po.	22 janvier 2019
Mises à jour du logiciel de service	Amazon ES vous permet désormais de mettre à jour manuellement le logiciel de service pour votre domaine afin de bénéficier plus rapidement des nouvelles fonctionnalités ou d'effectuer la mise à jour à un moment où le trafic est faible. Pour en savoir plus, veuillez consulter la section .	20 novembre 2018
Nouveaux CloudWatch indicateurs	Amazon ES propose désormais des métriques au niveau des nœuds et les nouveaux onglets Cluster Health (État du cluster) et Instance Health (État de l'instance) dans la console Amazon ES.	20 novembre 2018
Prise en charge dans la région Chine (Beijing)	Amazon OpenSearch Service est désormais disponible dans la région cn-north-1, où il prend en charge les types d'instances M4, C4 et R4.	17 octobre 2018

Node-to-node chiffrement	Amazon OpenSearch Service prend désormais en charge node-to-node le chiffrement, ce qui permet de chiffrer vos données lorsqu'Amazon ES les distribue dans l'ensemble de votre cluster.	18 septembre 2018
Mises à niveau des versions sur place	Amazon OpenSearch Service prend désormais en charge les mises à niveau de version sur place.	14 août 2018
Prise en charge d'Elastic search 6.3 et 5.6	Amazon OpenSearch Service prend désormais en charge les versions 6.3 et 5.6 d'Elastic search.	14 août 2018
Journaux des erreurs	Amazon ES vous permet désormais de publier des journaux d'erreurs Elasticse arch sur Amazon. CloudWatch	31 juillet 2018
Instances réservées pour la Chine (Ningxia)	Amazon ES propose désormais des instances réservées en Chine (Ningxia).	29 mai 2018
Instances réservées	Amazon ES propose désormais une assistance pour les instances réservées.	7 mai 2018

Mises à jour antérieures

Le tableau suivant décrit les modifications importantes apportées à Amazon ES antérieures au mois de mai 2018.

Modification	Description	Date
Consultez Authentification Amazon Cognito pour Kibana	Amazon ES propose désormais la protection de la page de connexion pour Kibana. Pour en savoir plus, veuillez consulter la section the section called “Authentification Amazon Cognito pour les tableaux de bord OpenSearch ”.	2 avril 2018
Prise en charge d'Elasticsearch 6.2	Amazon OpenSearch Service prend désormais en charge la version 6.2 d'Elasticsearch.	14 mars 2018
Plugin d'analyse coréen	Amazon ES prend désormais en charge une version optimisée pour la mémoire du plugin d'analyse coréen Seunjeon.	13 mars 2018
Mises à jour instantanées du contrôle d'accès	Les modifications apportées aux politiques de contrôle d'accès sur les domaines Amazon ES prennent désormais effet instantanément.	7 mars 2018
Mise à l'échelle d'une capacité de plusieurs péta-octets	Amazon ES prend désormais en charge les types d'instance I3 et un stockage de domaine total jusqu'à 1,5 Po. Pour en savoir plus, consultez la section the section called “Mise à l'échelle d'une capacité de plusieurs péta-octets”.	19 décembre 2017
Chiffrement de données au repos	Amazon ES prend désormais en charge le chiffrement des données au repos. Pour en savoir plus, consultez la section the section called “Chiffrement au repos”.	7 décembre 2017
Prise en charge d'Elasticsearch 6.0	Amazon ES prend désormais en charge Elasticsearch version 6.0. Pour des considérations et des instructions sur la migration, consultez the section called “Mise à niveau de domaines”.	6 décembre 2017
VPC Support	Amazon ES vous permet désormais de lancer des domaines dans un Amazon Virtual Private Cloud. Le support VPC fournit une couche supplémentaire de sécurité et simplifie les communications entre Amazon ES et les autres services au sein d'un VPC. Pour en savoir	17 octobre 2017

Modification	Description	Date
	plus, veuillez consulter la section the section called “Prise en charge de VPC”.	
Publication des journaux lents	Amazon ES prend désormais en charge la publication des journaux lents dans CloudWatch Logs. Pour en savoir plus, veuillez consulter la section the section called “Surveillance des journaux”.	16 octobre 2017
Prise en charge d'Elasticsearch 5.5	Amazon ES prend désormais en charge Elasticsearch version 5.5. Vous pouvez désormais restaurer les instantanés automatiques sans contacter Support et stocker des scripts à l'aide de l'API `_scripts`.	7 septembre 2017
Prise en charge d'Elasticsearch 5.3	Amazon ES prend désormais en charge Elasticsearch version 5.3.	1 juin 2017
Plus d'instances et de capacité EBS par cluster	Amazon ES prend désormais en charge jusqu'à 100 nœuds et une capacité EBS de 150 To par cluster.	5 avril 2017
Support pour le Canada (Centre) et l'UE (Londres)	Amazon ES a ajouté la prise en charge des régions suivantes : Canada (Centre), ca-central-1, et UE (Londres) , eu-west-2.	20 mars 2017
Plus d'instances et des volumes EBS plus importants	Amazon ES prend désormais en charge un plus grand nombre d'instances et des volumes EBS plus importants.	21 février 2017
Prise en charge d'Elasticsearch 5.1	Amazon ES prend désormais en charge Elasticsearch version 5.1.	30 janvier 2017
Prise en charge du plugin d'analyse phonétique	Amazon ES offre désormais une intégration embarquée avec le plugin Phonetic Analysis, qui vous permet d'exécuter des requêtes liées aux sons sur vos données.	22 décembre 2016

Modification	Description	Date
Prise en charge de la région USA Est (Ohio)	Amazon ES prend désormais en charge la région USA Est (Ohio), us-east-2.	17 octobre 2016
Nouvelle métrique de performance	Amazon ES a ajouté une métrique de performance, `ClusterUsedSpace` .	29 juillet 2016
Prise en charge d'Elasticsearch 2.3	Amazon ES prend désormais en charge Elasticsearch version 2.3.	27 juillet 2016
Prise en charge de la région Asie-Pacifique (Mumbai)	Amazon ES a ajouté la prise en charge de la région suivante : Asie-Pacifique (Mumbai), ap-south-1.	27 juin 2016
Plus d'instances par cluster	Amazon ES a augmenté le nombre maximal d'instances (nombre d'instances) par cluster de 10 à 20.	18 mai 2016
Prise en charge de la région Asie-Pacifique (Séoul)	Amazon ES a ajouté la prise en charge de la région suivante : Asie-Pacifique (Séoul), ap-northeast-2.	28 janvier 2016
Amazon ES	Première version.	1 octobre 2015

AWS Glossaire

Pour la AWS terminologie la plus récente, consultez le AWS glossaire dans la Glossaire AWS référence.

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.