ユーザーガイド

AWS Resource Explorer

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Resource Explorer: ユーザーガイド

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon の商標およびトレードドレスは、Amazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していないその他のすべての商標は、Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

Table of Contents

Resource Explorer 1
初めてご使用になる場合 1
Resource Explorer の特長 2
サポートされるリージョン 2
関連サービス 6
料金 7
使用開始 8
Resource Explorer へのアクセス 8
用語と概念 9
Resource Explorer 管理者 12
Resource Explorer ユーザー 13
[Index] (インデックス) 14
ビュー 15
[リソース] 17
AWS Management Console での統合検索 17
マルチアカウント検索 18
前提条件 18
にサインアップする AWS アカウント 19
管理アクセスを持つユーザーを作成する 19
Resource Explorer のセットアップ 21
Quick Setup 22
詳細設定 23
で Resource Explorer のステータスを特定する AWS リージョン 29
特定のリージョンでの Resource Explorer ステータスを確認する 29
特定のリージョンをオンにする 31
特定のリージョンに Resource Explorer インデックスを作成する 32
オプトインリージョンについて 34
オプトアウト挙動 35
クロスリージョン検索を有効にする 36
アグリゲーターインデックスについて 36
アグリゲーターインデックスの作成 38
アグリゲーターインデックスの降格 40
マルチアカウント検索を有効にする 43
前提条件 43

マルチアカウント検索を有効にする 43
マルチアカウントの Quick Setup 44
アカウントアクションがマルチアカウント検索に及ぼす影響 45
Resource Explorer を無効にする 45
メンバーアカウントが組織から削除されている 45
アカウントの停止 45
アカウントの閉鎖 46
アカウントのオプトアウト 46
コンソール統合検索のサポート 47
組織へのデプロイ 48
前提条件 48
Resource Explorer 用スタックセットの作成 49
サンプル AWS CloudFormation テンプレート 50
Resource Explorer をオフにする 54
1 つの で Resource Explorer をオフにする AWS リージョン 54
すべての をオフにする AWS リージョン 56
ビューの管理 59
デフォルトビュー 61
ビューの作成 62
ビューへのアクセス許可の付与 66
タグベースの認証を使用してビューへのアクセスを制御します。 68
デフォルトビューの設定 69
ビューのタグ付け 71
ビューにタグを追加する 71
タグによるアクセス許可の制御 72
ABAC ポリシー内のタグを参照する 72
ビューの共有 73
AWS アカウントとビューを共有するための権限ポリシー 74
ビューの削除 76
リソースの検索 78
検索結果を CSV ファイルにエクスポートする 81
サポートされているリソースタイプ 83
サポートされているサービスとリソースタイプ 84
Amazon API Gateway 87
AWS App Runner 87
Amazon AppStream 2.0 87

AWS AppSync 87
Amazon Athena 87
AWS Backup 87
AWS Batch 88
AWS CloudFormation 88
Amazon CloudFront 88
AWS CloudTrail 88
Amazon CloudWatch 88
Amazon CloudWatch Evidently 89
Amazon CloudWatch ログ 89
AWS CodeArtifact 89
AWS CodeBuild 89
AWS CodeCommit 89
Amazon CodeGuru Profiler 89
AWS CodePipeline 90
AWS CodeConnections 90
Amazon Cognito 90
Amazon Connect 90
Amazon Connect Wisdom 90
Amazon Detective 90
Amazon DynamoDB 90
EC2 Image Builder 91
Amazon ECR Public 91
AWS Elastic Beanstalk 91
Amazon ElastiCache 91
Amazon Elastic Compute Cloud (Amazon EC2) 92
Amazon Elastic Container Registry 94
Amazon Elastic Container Service 94
Amazon Elastic File System 94
Elastic Load Balancing 94
AWS Elemental MediaPackage 94
AWS Elemental MediaTailor 95
Amazon EMR Serverless 95
Amazon EventBridge 95
AWS Fault Injection Service 95
Amazon Forecast 95

Amazon Fraud Detector 95
Amazon GameLift 96
AWS Global Accelerator 96
AWS Glue 96
AWS Glue DataBrew 96
AWS Identity and Access Management 96
Amazon Interactive Video Service 97
AWS IoT 97
AWS IoT Analytics 97
AWS IoT Events 97
AWS IoT Greengrass Version 1 98
AWS IoT SiteWise 98
AWS IoT TwinMaker 98
AWS Key Management Service 98
Amazon Kinesis 98
Amazon Data Firehose 98
Amazon Kinesis Video Streams 99
AWS Lambda 99
Amazon Lex 99
Amazon Location Service 99
Amazon Lookout for Metrics 99
Amazon Lookout for Vision 99
Amazon Managed Service for Apache Flink 99
Amazon Managed Service for Prometheus 100
Amazon Managed Service for Prometheus 100
Amazon Managed Streaming for Apache Kafka 100
AWS Migration Hub Refactor Spaces 100
AWS Network Firewall 100
AWS Network Manager 100
Amazon OpenSearch サービス 101
AWS Panorama 101
Amazon Personalize 101
AWS Private Certificate Authority 101
Amazon QLDB 101
Amazon Redshift 101
Amazon Rekognition 102

Amazon Relational Database Service (Amazon RDS) 102
AWS Resilience Hub 102
AWS Resource Groups 102
AWS Resource Explorer 103
Amazon Route 53 103
Amazon Route 53 Recovery 準備状況 103
Amazon Route 53 Resolver 103
Amazon SageMaker 103
AWS Secrets Manager 103
AWS Service Catalog 104
Amazon Simple Notification Service 104
Amazon Simple Queue Service 104
Amazon Simple Storage Service (Amazon S3) 104
AWS Step Functions 104
AWS Systems Manager 104
AWS Verified Access 105
AWS Wavelength 105
サポートされているリソースタイプのリストにプログラムからアクセスする 105
他のリソースタイプとして表示されるリソースタイプ 106
検索クエリ構文 108
Resource Explorer でのクエリの仕組み 108
クエリ文字列の構文 108
基本 108
フィルター 109
フィルター演算子 113
クエリの例 117
タグ付けされていないリソース 117
リソースのタグ付け 118
欠落しているタグ 118
無効なタグ 118
リージョンのサブセット 119
グローバルリソース 119
複数のフィルタ 119
複数ワードの用語には引用符を使用する 120
AWS CloudFormation スタックメンバー 120
統合検索 121

統合検索が有効になっているか確認する 122
統合検索を有効にする 122
CloudFormation の使用 123
Resource Explorer と CloudFormation テンプレート 123
AWS CloudFormation の詳細情報 126
チャットアプリケーションでの Amazon Q Developer を使用する 127
AWS リソースに関する質問 127
前提条件 127
リソースに関するよくある質問 128
セキュリティ 129
IAM ポリシーを にアップグレードする IPv6 130
から IPv4 へのアップグレードの影響を受けるお客様 IPv6 130
IPv6 とは? 130
の IAMポリシーの更新 IPv6 131
クライアントが をサポートできることを確認する IPv6 132
ID およびアクセス管理 133
対象者 134
アイデンティティを使用した認証 135
ポリシーを使用したアクセスの管理 138
Resource Explorer と IAM 141
アイデンティティベースポリシーの例 147
SCP の例 153
AWS マネージドポリシー 154
サービスリンクロールの使用 172
アクセス許可のトラブルシューティング 174
データ保護 176
保管中の暗号化 177
転送中の暗号化 177
コンプライアンス検証 177
耐障害性 178
インフラストラクチャセキュリティ 179
モニタリング 180
CloudTrail ログ 180
CloudTrail 上の Resource Explorer 情報 180
Resource Explorer のログファイルエントリについて理解する 182
トラブルシューティング 192

一般的な問題 192
Resource Explorer へのリンクに AWS リージョン がない 192
統合検索 CloudTrail エラー 193
セットアップの問題 194
Resource Explorer にリクエストを送信すると、「アクセスが拒否されました」というメッセージが表示される 195
一時的なセキュリティ認証情報を使用してリクエストを送信すると「アクセスが拒否されました」というメッセージが表示される 196
検索に関する問題 196
Resource Explorer の検索結果に一部のリソースが表示されない 196
コンソールの統合検索結果に自分のリソースが表示されない 199
コンソールと Resource Explorer の統合検索の結果が異なることがある 199
リソースを検索するのに必要なアクセス許可 199
クォータ 201
の使用 AWS SDKs 202
ドキュメント履歴 204
..... ccx

とは AWS Resource Explorer

AWS Resource Explorer はリソース検索および検出サービスです。Resource Explorer を使用すると、インターネット検索エンジンのようなエクスペリエンスを使用して、Amazon Elastic Compute Cloud インスタンス、Amazon Kinesis ストリーム、Amazon DynamoDB テーブルなどのリソースを探索できます。名前、タグ、 などのリソースメタデータを使用してリソースを検索できます IDs。Resource Explorer はアカウント AWS リージョン 内で動作し、リージョン間のワークロードを簡素化します。

Resource Explorer は、 AWS Resource Explorer サービスによって作成および維持されるインデックスを使用して、検索クエリへの迅速な応答を提供します。Resource Explorer は、さまざまなデータソースを使用して、 AWS アカウント内のリソースに関する情報を収集します。Resource Explorer は、その情報を Resource Explorer が検索できるように各インデックスに保存します。

> このドキュメントに関するフィードバックをお待ちしています。
>
> 私たちの目標は、Resource Explorer をユーザーの皆様に最大限活用していただくことです。このガイドが皆様のお役に立てたら、ぜひお知らせください。またガイドにご満足いただけない場合には、問題に対処できるよう、ご意見をお聞かせください。各ページの右上の [フィードバック] リンクを使用してコメントを送信できます。送信されたコメントは、本ガイドの作成チームに直接転送されます。私たちはすべての提出物を精査し、ドキュメントの継続的な改善に努めています。皆さまのご協力をよろしくお願いします。

トピック

- Resource Explorer を初めてご使用になる方へ
- Resource Explorer の特長
- Resource Explorer でサポートされているリージョン
- 関連 AWS のサービス
- 料金

Resource Explorer を初めてご使用になる方へ

Resource Explorer を初めてご使用になる場合には、まず [使用の開始] セクションからお読みいただくことをお勧めします。

- Resource Explorer の用語と概念
- Quick Setup を使用して Resource Explorer をセットアップする

Resource Explorer の特長

Resource Explorer には次の特長があります。

- ユーザーは、 内の AWS リージョン または リージョン間でリソースを検索できます AWS アカウント。
- ユーザーは、キーワード、検索演算子、およびタグなどの属性を使用して、条件と一致するリソースのみに検索結果を絞り込むことができます。
- ユーザーは検索結果で必要なリソースを見つけたら、すぐにそのリソースのネイティブコンソールに移動してそのリソースを操作できます。
- 管理者は、どのリソースを検索結果に含めるかを定義するビューを作成できます。管理者は、タスクに基づいてユーザーグループごとに異なるビューを作成し、必要なユーザーのみにビューへのアクセス権限を付与できます。
- Resource Explorer は、他の多くの と同様に AWS のサービス、最終的には整合性のある になります。Resource Explorer は、世界中の Amazon データセンター内の複数のサーバーにデータをリプリケートすることにより、高可用性を実現します。何らかのデータの変更リクエストが正常に受け付けられると、当該変更はコミットされ、安全に保管されます。ただし、変更を Resource Explorer 全体にリプリケートするには多少時間がかかることがあります。これには例として、Resource Explorer が 1 つのリージョン内でリソースを発見した後、そのアカウントのアグリゲーターインデックスを含むリージョンにそのリソースをリプリケートするプロセスが含まれます。

Resource Explorer でサポートされているリージョン

リージョン名	リージョン	エンドポイント	プロトコル	
米国東部 (オハイオ)	us-east-2	resource-explorer-2.us-east-2.amazonaws.com	HTTPS	
		resource-explorer-2-fips.us-east-2.amazonaws.com	HTTPS	
			HTTPS	

リージョン名	リージョン	エンドポイント	プロトコル
		resource-explorer-2-fips.us-east-2.api.aws	
米国東部 (バージニア北部)	us-east-1	resource-explorer-2.us-east-1.amazonaws.com resource-explorer-2-fips.us-east-1.amazonaws.com resource-explorer-2-fips.us-east-1.api.aws	HTTPS HTTPS HTTPS
米国西部 (北カリフォルニア)	us-west-1	resource-explorer-2.us-west-1.amazonaws.com resource-explorer-2-fips.us-west-1.amazonaws.com resource-explorer-2-fips.us-west-1.api.aws	HTTPS HTTPS HTTPS
米国西部 (オレゴン)	us-west-2	resource-explorer-2.us-west-2.amazonaws.com resource-explorer-2-fips.us-west-2.amazonaws.com resource-explorer-2-fips.us-west-2.api.aws	HTTPS HTTPS HTTPS
アフリカ (ケープタウン)	af-south-1	resource-explorer-2.af-south-1.amazonaws.com	HTTPS
アジアパシフィック (香港)	ap-east-1	resource-explorer-2.ap-east-1.amazonaws.com	HTTPS
アジアパシフィック (ハイデラバード)	ap-south-2	resource-explorer-2.ap-south-2.amazonaws.com	HTTPS

リージョン名	リージョン	エンドポイント	プロトコル
アジアパシフィック (ジャカルタ)	ap-southeast-3	resource-explorer-2.ap-southeast-3.amazonaws.com	HTTPS
アジアパシフィック (メルボルン)	ap-southeast-4	resource-explorer-2.ap-southeast-4.amazonaws.com	HTTPS
アジアパシフィック (ムンバイ)	ap-south-1	resource-explorer-2.ap-south-1.amazonaws.com	HTTPS
アジアパシフィック (大阪)	ap-northeast-3	resource-explorer-2.ap-northeast-3.amazonaws.com	HTTPS
アジアパシフィック (ソウル)	ap-northeast-2	resource-explorer-2.ap-northeast-2.amazonaws.com	HTTPS
アジアパシフィック (シンガポール)	ap-southeast-1	resource-explorer-2.ap-southeast-1.amazonaws.com	HTTPS
アジアパシフィック (シドニー)	ap-southeast-2	resource-explorer-2.ap-southeast-2.amazonaws.com	HTTPS

リージョン名	リージョン	エンドポイント	プロトコル
アジアパシフィック (東京)	ap-northeast-1	resource-explorer-2.ap-northeast-1.amazonaws.com	HTTPS
カナダ (中部)	ca-central-1	resource-explorer-2.ca-central-1.amazonaws.com resource-explorer-2-fips.ca-central-1.amazonaws.com resource-explorer-2-fips.ca-central-1.api.aws	HTTPS HTTPS HTTPS
カナダ西部 (カルガリー)	ca-west-1	resource-explorer-2.ca-west-1.amazonaws.com resource-explorer-2-fips.ca-west-1.amazonaws.com resource-explorer-2-fips.ca-west-1.api.aws	HTTPS HTTPS HTTPS
欧州 (フランクフルト)	eu-central-1	resource-explorer-2.eu-central-1.amazonaws.com	HTTPS
欧州 (アイルランド)	eu-west-1	resource-explorer-2.eu-west-1.amazonaws.com	HTTPS
欧州 (ロンドン)	eu-west-2	resource-explorer-2.eu-west-2.amazonaws.com	HTTPS
ヨーロッパ (ミラノ)	eu-south-1	resource-explorer-2.eu-south-1.amazonaws.com	HTTPS
欧州 (パリ)	eu-west-3	resource-explorer-2.eu-west-3.amazonaws.com	HTTPS

リージョン名	リージョン	エンドポイント	プロトコル
欧州 (スペイン)	eu-south-2	resource-explorer-2.eu-south-2.amazonaws.com	HTTPS
欧州 (ストックホルム)	eu-north-1	resource-explorer-2.eu-north-1.amazonaws.com	HTTPS
欧州 (チューリッヒ)	eu-central-2	resource-explorer-2.eu-central-2.amazonaws.com	HTTPS
イスラエル (テルアビブ)	il-central-1	resource-explorer-2.il-central-1.amazonaws.com	HTTPS
中東 (バーレーン)	me-south-1	resource-explorer-2.me-south-1.amazonaws.com	HTTPS
中東 (UAE)	me-central-1	resource-explorer-2.me-central-1.amazonaws.com	HTTPS
南米 (サンパウロ)	sa-east-1	resource-explorer-2.sa-east-1.amazonaws.com	HTTPS

関連 AWS のサービス

もう 1 AWS のサービス つは、 AWS リソースの管理を支援することを主な目的としています。

AWS Resource Access Manager (AWS RAM)

1 つの のリソースを他の AWS アカウント と共有します AWS アカウント。アカウントが によって管理されている場合は AWS Organizations、 AWS RAM を使用して、組織単位のアカウント、または組織内のすべてのアカウントとリソースを共有できます。共有リソースは、ローカルアカウントで作成された場合と同様に、それらのアカウントのユーザーに対しても機能します。

AWS Resource Groups

AWS リソースのグループを作成します。そうすれば、すべてのリソースを個別に参照しなくても、各グループを 1 つの単位として使用、管理できます。リソースグループは、同じ AWS CloudFormation スタックに属するリソースのグループでも良いし、同じタグでタグ付けされたリソースのグループでも良いです。リソースタイプによっては、リソースグループに構成設定を適用して、そのグループ内のすべての関連リソースに影響を与えることもできます。

タグエディタと AWS Resource Groups Tagging API

タグはリソースにアタッチされるユーザー定義のメタデータです。コスト配分や属性ベースのアクセス制御などの目的でリソースを分類できます。

料金

ビューの作成 AWS Resource Explorer、リージョンの有効化、リソースの検索など、 を使用してリソースを検索する料金は発生しません。リソースインベントリを構築する過程で、Resource Explorer はAPIsユーザーに代わって を呼び出し、料金が発生する可能性があります。検索結果で見つかったリソースを操作すると、リソースタイプとその によって使用料が異なる場合があります AWS のサービス。特定のリソースタイプの通常の使用に対する AWS 請求の詳細については、そのリソースタイプの所有サービスのドキュメントを参照してください。

Resource Explorer の使用を開始する

このセクションのトピックを使用して、 で使用される概念と用語の基本的な理解を深めます AWS Resource Explorer。Resource Explorer を正しく使用するために必要となる前提条件と、 AWS アカウントで Resource Explorer を有効にする方法について説明します。

Resource Explorer へのアクセス

Resource Explorer には次の方法でアクセスできます。

Resource Explorer コンソール

Resource Explorer には、Resource Explorer コンソールというウェブベースのユーザーインターフェイスがあります。にサインアップした場合は AWS アカウント、 にサインインAWS Management Consoleし、コンソールのホームページから Resource Explorer を選択することで、Resource Explorer コンソールにアクセスできます。

また、ブラウザ操作により、[Resource Explorer ダッシュボード]ページや[リソース検索]ページに直接移動することもできます。まだサインインしていない場合、コンソールが表示される前にログインするように求められます。

> **Note**
>
> Resource Explorer コンソールはグローバルコンソールです。つまり、 AWS リージョン作業する を選択する必要はありません。ただし、Resource Explorer を使用してインデックスまたはビューを作成する場合は、どのリージョンにそのインデックスまたはビューを格納するかを指定する必要があります。Resource Explorer を使用して検索を実行する場合、アクセス権限のある任意のビューを選択できます。検索結果は選択したビューに関連付けられているリージョンから自動的に取得されます。アグリゲーターインデックスを含むリージョンのビューの場合、Resource Explorer インデックスを作成しているすべてのリージョンのリソースが検索結果に含まれます。

AWS Management Console 統合検索

の各ページの上部には AWS Management Console、検索バーがあります。Resource Explorer を、統合検索に参加するように設定することができます。統合検索テキストボックスで

Resource Explorer 検索クエリ構文を使用して検索を実行すれば、検索条件に一致するリソースが検索結果に表示されます。この機能を有効にすると、ユーザーは最初に Resource Explorer コンソールに切り替える AWS のサービス ことなく、 のコンソールからリソースを検索できます。

> **Important**
>
> 統合検索では、常にアグリゲータインデックス AWS リージョン を含む のデフォルトビューを使用して検索されます。

AWS CLI および Tools for Windows の Resource Explorer コマンド PowerShell

AWS CLI および のツール PowerShell では、Resource Explorer のパブリックAPIオペレーションに直接アクセスできます。これらのツールは、Windows、macOS、Linux で動作します。使用開始方法の詳細については、「AWS Command Line Interface ユーザーガイド」または「AWS Tools for Windows PowerShell ユーザーガイド」を参照してください。Resource Explorer コマンドの詳細については、「AWS CLI コマンドリファレンス」または「AWS Tools for Windows PowerShell コマンドレットリファレンス」を参照してください。

の Resource Explorer オペレーション AWS SDKs

AWS には、さまざまなプログラミング言語のAPIコマンドが用意されています。使用開始の詳細については、「AWS Resource Explorer で を使用する AWS SDK」を参照してください。

クエリ API

サポートされているプログラミング言語のいずれかを使用しない場合、Resource Explorer HTTPS クエリAPIを使用すると、Resource Explorer にプログラムでアクセスできます。Resource Explorer を使用するとAPI、サービスに直接HTTPSリクエストを発行できます。Resource Explorer を使用する場合はAPI、 AWS 認証情報を使用してリクエストにデジタル署名できるコードを含める必要があります。詳細については、「 AWS Resource Explorer APIリファレンス」を参照してください。

Resource Explorer の用語と概念

AWS Resource Explorer はリソース検索および発見サービスです。Resource Explorer では、インターネット検索エンジンのようなエクスペリエンスを使用してリソースを検索できます。名前、タグ、ID などのリソースのメタデータを使用して、Amazon Elastic Compute Cloud インスタンス、Amazon Kinesis ストリーム、Amazon DynamoDB テーブルなどのリソースを検索できま

す。Resource Explorer はアカウント全体の AWS リージョン で機能し、クロスリージョンのワークロードをシンプルにします。

Resource Explorer は、AWS Resource Explorer サービスによって作成および管理されるインデックスを使用して、検索クエリに迅速に応答します。Resource Explorer は、さまざまなデータソースを使用して、AWS アカウント 内のリソースに関する情報を収集します。Resource Explorer は、その情報を Resource Explorer が検索できるように各インデックスに保存します。

ユーザーが適切に AWS Resource Explorer を管理および設定できるようにするには、管理者が次の概念を理解する必要があります。

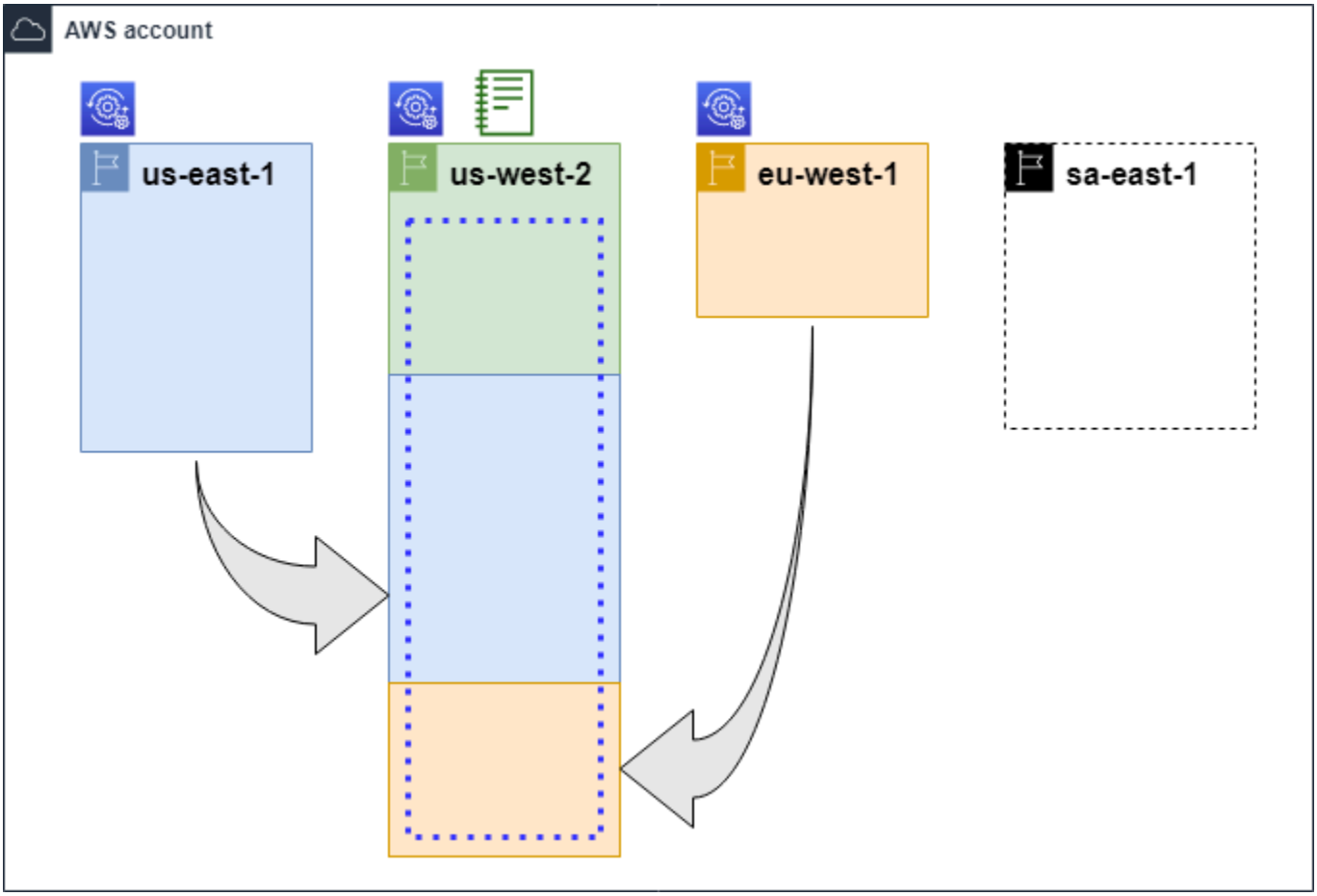
概念

- Resource Explorer 管理者
- Resource Explorer ユーザー
- [Index] (インデックス)
- ビュー
- [リソース]
- AWS Management Console での統合検索
- マルチアカウント検索

次の図は、管理者が Resource Explorer を有効にした 3 つの AWS リージョン と、管理者が Resource Explorer を有効にしないことを選択した 1 つのリージョンを示します。Resource Explorer が有効になっていないリージョンにはインデックスがありません。そのため、Resource Explorer のクエリではそのリージョンのリソースを検索できません。

このシナリオ例では、管理者は米国西部 (オレゴン) リージョン (`us-west-2`) にそのアカウントのアグリゲーターインデックスを格納するよう選択しました。有効にしたすべてのリージョンのローカルインデックスが、アグリゲーターインデックスのあるリージョンにリプリケートされます。

Resource Explorer によって作成されるデフォルトビューにはフィルターがありません。したがって、このビューで検索する結果には、そのアカウントで Resource Explorer がオンになっているすべてのリージョンのあらゆる種類のリソースが含まれます。



凡例	
	Resource Explorer はこの AWS リージョン でオンになっており、そのリージョンのリソースに関する情報はそのリージョンのローカルインデックスに保存されます。各リージョンのローカルインデックスは、アグリゲーターインデックスを含むリージョンにもリプリケート (矢印で示す)されます。
	この AWS リージョン 内のインデックスが、アカウントのアグリゲーターインデックスになるように設定されています。Resource Explorer は、Resource Explorer がオンになっている他のすべてのリージョンのローカルインデックスで収集されるリソース情報を、このリージョンのアグリゲーターインデックスにリプリケートします。このリージョンで行われる検索には、アカウント内のすべてのリージョンからの結果が含まれます。
	[Quick Setup] で作成されるデフォルトビューには、AWS リージョン 内のすべてのリソースが含まれます。

Resource Explorer 管理者

Resource Explorer の管理者は、組織内または AWS アカウント 。Resource Explorer 管理者は以下の機能を設定できます。

- AWS アカウント 内の各 AWS リージョン について、それぞれのリージョンのインデックスを作成することで Resource Explorer を有効にします。これにより、Resource Explorer はリソースを検出し、そのリソースに関する情報をインデックスに入力して、ユーザーがそのリージョンのリソースを検索できるようにします。
- 1 つの AWS リージョン のインデックスタイプが、その AWS アカウント のアグリゲーターインデックスになるように更新します。このリージョンのアグリゲーターインデックスは、アカウント内で Resource Explorer がオンになっている他のすべてのリージョンからのリソース情報のリプリケートコピーを受け取ります。
- ユーザーが Resource Explorer で検索し発見できるインデックス付き情報のサブセットを定義するビューを作成します。
- Resource Explorer のアクションには含まれていませんが、Resource Explorer 管理者はアカウント内の各プリンシパルに検索権限を付与する権限も持っている必要もあります。管理者は、必要なアクセス許可を既存の IAM アクセス許可ポリシーに追加するか、Resource Explorer の読み取り専用 AWS マネージドポリシーを使用することで、これらのアクセス許可をプリンシパルに付与できます。

  アクセス権限を付与するには、ユーザー、グループ、またはロールにアクセス許可を追加します。

  - AWS IAM Identity Center のユーザーとグループ:

    アクセス許可セットを作成します。「AWS IAM Identity Center ユーザーガイド」の「アクセス許可一式を作成」の手順を実行します。
  - IAM 内で、ID プロバイダーによって管理されているユーザー:

    ID フェデレーションのロールを作成します。詳細については、「IAM ユーザーガイド」の「サードパーティー ID プロバイダー (フェデレーション) 用のロールの作成」を参照してください。
  - IAM ユーザー:
    - ユーザーに設定できるロールを作成します。手順については、「IAM ユーザーガイド」の「IAM ユーザー用ロールの作成」を参照してください。

- (お奨めできない方法) ポリシーをユーザーに直接アタッチするか、ユーザーをユーザーグループに追加する。「IAM ユーザーガイド」の「ユーザー (コンソール) へのアクセス許可の追加」の指示に従います。

管理者は通常、インデックスやビューを含むすべての Resource Explorer リソースに対するすべての Resource Explorer 権限 (`resource-explorer-2:*`) を持っています。これらの権限は、Resource Explorer のフルアクセス AWS マネージドポリシーを使用して付与することができます。

Resource Explorer ユーザー

Resource Explorer のユーザーは、次の 1 つ以上のタスクを実行する権限を持つ IAM プリンシパルです。

- ビューを使用してリソースを検索することにより Resource Explorer にクエリを実行します。Resource Explorer ユーザーは、AWS リソースの検索と発見を行うため、通常は Resource Explorer コンソール、または AWS SDK または AWS CLI が提供する Resource Explorer `Search` 操作を使用します。

  ロールまたはユーザーは、次の 2 つの手段のいずれかにより検索に必要な IAM get 権限を使用できます。

  - その IAM ロール、グループまたはユーザーに対する Resource Explorer の読み取り専用 AWS マネージドポリシー。
  - その IAM ロール、グループ、またはユーザーに対する以下の最小限の権限を含むステートメントを含む IAM アクセス許可ポリシー。

  ```
  {
      "Effect": "Allow",
      "Action": [
          "resource-explorer-2:Search",
          "resource-explorer-2:GetView",
      "Resource": "<ARN of the view>"
  }
  ```

- 一般的には管理者タスクと見なされますが、ビューの作成を定義する権限を信頼できるユーザーに委任することができます。そのために、管理者は関連するロール、グループ、またはユーザーにアタッチされた IAM アクセス許可ポリシーで `resource-explorer-2:CreateView` 操作を呼び出す権限を付与できます。ビューに特定の権限が必要な場合は、関連するユーザーの IAM ポリシーを追加または変更するためのプロビジョニングを行う必要があります。

Resource Explorer を使用してリソースを検索する方法については、AWS Resource Explorer を用いたリソースの検索 を参照してください。

[Index] (インデックス)

インデックスとは、AWS アカウント 内の 1 つの AWS リージョン について Resource Explorer が管理するすべての AWS リソースに関する情報をまとめたものです。Resource Explorer は、Resource Explorer をオンにした各リージョンについてインデックスを保持します。Resource Explorer は、AWS アカウント でリソースを作成したり削除したりすると、インデックスを自動的に更新します。前の図では、AWS リージョン 名の下にある各ボックスは、それぞれの AWS リージョン について管理されている Resource Explorer のインデックスを表しています。リージョン内のインデックスは、そのリージョンで作成されたすべてのビューの情報源です。インデックスを直接クエリすることはできません。常にビューを使用してクエリを実行する必要があります。

インデックスには次の 2 種類があります。

ローカルインデックス

Resource Explorer をオンにしている各 AWS リージョン についてそれぞれ 1 つのローカルインデックスがあります。ローカルインデックスには、そのリージョンのリソースに関する情報のみが含まれます。

アグリゲーターインデックス

Resource Explorer 管理者は、一つの AWS リージョン 内のインデックスを AWS アカウント のアグリゲーターインデックスとして指定することもできます。アグリゲーターインデックスは、そのアカウントで Resource Explorer がオンになっている他のすべてのリージョンのインデックスのコピーを受け取って保存します。アグリゲーターインデックスは、自リージョンのリソースに関する情報も受け取って保存します。前の図では、リージョン `us-west-2` にはそのアカウントのアグリゲーターインデックスが含まれています。アカウントにアグリゲーターインデックスを指定する主な理由は、アカウント内のすべてのリージョンのリソースを含むビューを作成できるようにするためです。一つの AWS アカウント にはアグリゲーターインデックスは 1 つしか作成できません。

Resource Explorer をオンにすると、アグリゲーターインデックスをどの AWS リージョン に格納するかを指定できるようになります。アグリゲーターインデックスに使用する AWS リージョン は後で変更することもできます。ローカルインデックスをその AWS アカウント のアグリゲーターインデックスに昇格させる方法については、アグリゲーターインデックスを作成してクロスリージョン検索を有効にする を参照してください。

インデックスとは、Amazon リソースネーム (ARN) を持つリソースです。この ARN は、アクセス許可ポリシー内でそのインデックスと直接やり取りする各種操作へのアクセス許可を許可する目的にのみ使用できます。それらの操作により、ビューを作成してそのリージョンのデフォルトとして設定したり、特定のリージョンについて Resource Explorer を有効または無効にしたり、アカウントのアグリゲーターインデックスを作成したりすることができます。インデックス ARN は以下の例のようになります。

```
arn:aws:resource-explorer-2:us-east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-
abcd11111111
```

ビュー

ビューとは、インデックスにリストされているリソースをクエリするために使用されるメカニズムです。ビューは、インデックス内のどの情報を表示して検索や発見に利用できるかを定義します。ユーザーが Resource Explorer のインデックスに直接クエリを実行することはありません。クエリは常にビューを経由する必要があります。これにより、ビューの作成者は、ユーザーの検索結果に表示されるリソースを制限できます。

ビューを作成するときは、検索結果に含まれるリソースを制限するフィルターを指定します。例えば、このビューへのアクセスを付与するユーザーが使用する、指定された少数のリソースタイプのリソースのみを含めるように選択できます。ビューを使用してユーザーが行ったクエリの結果は、ビューに一致するリソースのみを含むよう自動的にフィルタリングされます。

ビューを使用するためのアクセス許可を付与するには、次のいずれかの手段で権限の割り当てを行います。

アクセス権限を付与するには、ユーザー、グループ、またはロールにアクセス許可を追加します。

- AWS IAM Identity Center のユーザーとグループ:

  アクセス許可セットを作成します。「AWS IAM Identity Center ユーザーガイド」の「アクセス許可一式を作成」の手順を実行します。
- IAM 内で、ID プロバイダーによって管理されているユーザー:

  ID フェデレーションのロールを作成します。詳細については、「IAM ユーザーガイド」の「サードパーティー ID プロバイダー (フェデレーション) 用のロールの作成」を参照してください。
- IAM ユーザー:
  - ユーザーに設定できるロールを作成します。手順については、「IAM ユーザーガイド」の「IAM ユーザー用ロールの作成」を参照してください。

- (お奨めできない方法) ポリシーをユーザーに直接アタッチするか、ユーザーをユーザーグループに追加する。「IAM ユーザーガイド」の「ユーザー (コンソール) へのアクセス許可の追加」の指示に従います。

ロール、グループ、またはユーザーに対して、Amazon リソースネーム (ARN) で識別されるビューで `resource-explorer-2:GetView` および `resource-explorer-2:Search` オペレーションを呼び出すことを許可するアクセス許可を付与します。または、そのビューを使用して検索する必要のあるすべてのプリンシパルに対して、「Resource Explorer の読み取り専用 AWS マネージドポリシー」を使用することもできます。フィルターや範囲が異なる複数のビューを作成して、リソース情報のさまざまなサブセットを検索結果として返すことができます。これにより、それぞれのビューの結果に含まれる情報を確認する必要があるユーザーにそのビューの権限を付与できます。

Resource Explorer で検索を行うには、各ユーザーが少なくとも 1 つのビューを使用する権限を持っている必要があります。ビューを使わずに Resource Explorer で検索を実行することはできません。

ビューはリージョンごとに保存されます。ビューはその AWS リージョン の Resource Explorer インデックスにのみアクセスできます。アカウント全体の検索結果にアクセスするには、そのアカウントのアグリゲーターインデックスを含むリージョンのビューを使用する必要があります。[Quick Setup] オプションでは、そのアカウントで使用するすべての AWS リージョン のリソースを含むアグリゲーターインデックスとフィルターを含む AWS リージョン のデフォルトビューが作成されます。

ビューを作成する方法については、「検索アクセス許可を提供するための Resource Explorer ビューの管理」を参照してください。ビューをクエリに使用する方法については、「AWS Resource Explorer を用いたリソースの検索」を参照してください。

すべてのビューには Amazon リソースネーム (ARN) があり、これをアクセス許可ポリシー内で引用することによりそれぞれのビューへのアクセス許可を付与できます。ビューの ARN は、ビューとやり取りする任意の API または AWS CLI オペレーションにパラメータとして渡すこともできます。ビュー ARN は以下の例のようになります。

```
arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View-
Name/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111
```

> **Note**
>
> すべてのビュー ARN には、AWS で生成される UUID が末尾に付されます。これにより、削除された特定の名前のビューにアクセスできたユーザーが、同じ名前で作成された新しいビューに自動的にアクセスできないようにします。

[リソース]

リソースとは、ユーザーが操作できる AWS 内のエンティティです。リソースは、サービスの機能を使用する際に AWS のサービス により作成されます。例として、Amazon EC2 インスタンス、Amazon S3 バケット、AWS CloudFormation スタックなどがあります。一部のリソースタイプには顧客データが含まれる場合があります。すべてのリソースタイプには、名前、記述、およびリソースを一意に参照する Amazon リソースネーム (ARN) など、リソースを記述する属性またはメタデータが備わっています。ほとんどのリソースタイプはタグもサポートしています。タグは、請求時のコスト配分、属性ベースのアクセス制御によるセキュリティ認証、およびその他の分類ニーズへの対応など、さまざまな目的でリソースに添付できるカスタムメタデータです。

Resource Explorer の主な目的は、AWS アカウント に存在するリソースを検索しやすくすることです。Resource Explorer は、さまざまな手法を使用してすべてのリソースを検出し、その情報をインデックスに格納します。その後、管理者が提供している任意のビューを使用してインデックスをクエリできます。

> **Important**
>
> Resource Explorer は、含めると顧客データが公開されてしまうようリソースタイプを意図的に除外します。以下のリソースタイプは Resource Explorer ではインデックスされないため、検索結果として返されません。
>
> - バケット内に含まれる Amazon S3 オブジェクト
> - Amazon DynamoDB テーブルアイテム
> - DynamoDB 属性値

AWS Management Console での統合検索

それぞれの AWS のサービス の AWS Management Console の上部には検索バーがあり、AWS に関連するさまざまなものの検索に使用できます。サービスや機能を検索して、そのサービスのコンソー

ルの関連ページへのリンクを直接表示できます。検索語に関連するドキュメントやブログ記事を検索することもできます。

Resource Explorer をオンにしてアグリゲーターインデックスとデフォルトビューを作成すると、統合検索の検索結果にアカウントのリソースを含めることもできます。統合検索では、アカウントのアグリゲーターインデックスを含む AWS リージョン のデフォルトビューが自動的に使用されます。これにより、予め Resource Explorer を開かなくても、AWS Management Console のどのページからでもリソースを検索できます。ローカルインデックスをそのアカウントのアグリゲーターインデックスに昇格させない場合、またはアグリゲーターインデックスリージョンにデフォルトビューを作成しない場合、統合検索の検索結果にリソースは含まれません。また、検索を実行するプリンシパルが、アグリゲーターインデックスを含むリージョンのデフォルトビューを使用する権限を持っている必要があります。そうしないと、統合検索の検索結果にリソースが含まれません。

> **Important**
>
> 統合検索では、文字列の最初のキーワードの末尾にワイルドカード文字 (*) 演算子が自動的に挿入されます。つまり、統合検索結果には、指定されたキーワードで始まる任意の文字列と一致するリソースが含まれます。
> これに対し、Resource Explorer コンソールの [リソース検索] ページの [クエリ] テキストボックスから実行する検索では、ワイルドカード文字は自動的には追加されません。検索文字列の任意の用語の後に、* を手動で挿入できます。

統合検索と Resource Explorer との統合の詳細については、「AWS Management Console での統合検索の使用」を参照してください。

マルチアカウント検索

マルチアカウント検索では、一つのキーワード検索で AWS Organizations および AWS リージョン全体の リソースを検索して発見することができます。

マルチアカウント検索と Resource Explorer でマルチアカウント検索を有効にする方法の詳細については、「マルチアカウント検索を有効にする」を参照してください。

Resource Explorer を使用するための前提条件

AWS Resource Explorer を初めて使用する場合は、必要に応じて以下のタスクを完了してください。

タスク

- にサインアップする AWS アカウント
- 管理アクセスを持つユーザーを作成する

にサインアップする AWS アカウント

がない場合は AWS アカウント、次の手順を実行して作成します。

にサインアップするには AWS アカウント

1. https://portal.aws.amazon.com/billing/サインアップ を開きます。
2. オンラインの手順に従います。

   サインアップ手順の一環として、通話呼び出しを受け取り、電話キーパッドで検証コードを入力するように求められます。

   にサインアップすると AWS アカウント、 AWS アカウントのルートユーザー が作成されます。ルートユーザーには、アカウントのすべての AWS のサービス とリソースへのアクセス権があります。セキュリティのベストプラクティスとして、ユーザーに管理アクセスを割り当て、ルートユーザーのみを使用してルートユーザーアクセスが必要なタスクを実行してください。

AWS は、サインアッププロセスが完了した後に確認 E メールを送信します。/ に移動し、マイアカウント を選択して、いつでも現在のアカウントアクティビティを表示https://aws.amazon.comし、アカウントを管理できます。

管理アクセスを持つユーザーを作成する

にサインアップしたら AWS アカウント、 のセキュリティを確保し AWS アカウントのルートユーザー、 を有効にして管理ユーザーを作成し AWS IAM Identity Center、日常的なタスクにルートユーザーを使用しないようにします。

のセキュリティ保護 AWS アカウントのルートユーザー

1. ルートユーザーを選択し、 AWS アカウント E メールアドレスを入力して、アカウント所有者AWS Management Consoleとして にサインインします。次のページでパスワードを入力します。

ルートユーザーを使用してサインインする方法については、AWS サインイン ユーザーガイドのルートユーザーとしてサインインするを参照してください。

2. ルートユーザーの多要素認証 (MFA) を有効にします。

   手順については、「 ユーザーガイド」の AWS アカウント 「ルートユーザー (コンソール) の仮想MFAデバイスの有効化」を参照してください。 IAM

管理アクセスを持つユーザーを作成する

1. IAM Identity Center を有効にします。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「AWS IAM Identity Centerの有効化」を参照してください。

2. IAM Identity Center で、ユーザーに管理アクセスを許可します。

   ID ソース IAM アイデンティティセンターディレクトリ として を使用するためのチュートリアルについては、 AWS IAM Identity Center ユーザーガイドの「デフォルトを使用してユーザーアクセスを設定する IAM アイデンティティセンターディレクトリ」を参照してください。

管理アクセス権を持つユーザーとしてサインインする

- IAM Identity Center ユーザーでサインインするには、IAMIdentity Center ユーザーの作成時に E メールアドレスにURL送信されたサインインを使用します。

  IAM Identity Center ユーザーを使用してサインインする方法については、AWS サインイン「 ユーザーガイド」のAWS 「 アクセスポータルへのサインイン」を参照してください。

追加のユーザーにアクセス権を割り当てる

1. IAM Identity Center で、最小権限のアクセス許可を適用するベストプラクティスに従うアクセス許可セットを作成します。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「権限設定を作成する」を参照してください。

2. グループにユーザーを割り当て、そのグループにシングルサインオンアクセス権を割り当てます。

手順については、「AWS IAM Identity Center ユーザーガイド」の「グループの参加」を参照してください。

Resource Explorer のセットアップと設定

をセットアップして設定する前に AWS Resource Explorer、まず前提条件 を満たしていることを確認してください。その後、以下の手順で Resource Explorer オペレーションを実行するために必要なアクセス許可を持つ IAMロールまたはユーザーとしてサインインします。

この設定および設定手順を使用して、既存のアカウント、および組織に追加された新しいアカウントで Resource Explorer を設定できます。

Resource Explorer のセットアップには以下の 2 つの方法があります。

- Quick Setup
- 詳細設定

> **Important**
>
> 「すべて」というオプションを使用して Resource Explorer をセットアップすることを選択した場合 AWS リージョン、プロシージャ の実行時に で有効 AWS アカウント AWS リージョン になっているもののみがアクティブ化されます。 Resource Explorer は、今後 が AWS 追加 AWS リージョン する では自動的にオンになりません。が新しいリージョンを導入するときは、Resource Explorer AWS コンソールの設定ページに表示されるときにリージョンで Resource Explorer を手動で有効にするか、 CreateIndexオペレーションを呼び出すかを選択できます。

> **Note**
>
> Resource Explorer をセットアップすると、 AWS Management Consoleの統合検索バーを使用してリソースを検索する機能も有効にすることができるようになります。ユーザーが統合検索結果のリソースを見ることができるようにするには、クロスリージョンアグリゲーターインデックスとデフォルトビューを Resource Explorer 設定に含める必要があります。詳細については、以下に記載される手順を参照してください。また、検索するユーザーに、アグリゲータインデックスを含む のデフォルトビューを使用するアクセス許可 AWS リージョン

があることを確認する必要があります。詳細については、「AWS Management Console での統合検索の使用」を参照してください。

Quick Setup を使用して Resource Explorer をセットアップする

Quick Setup オプションを選択すると、Resource Explorer は次の処理を行います。

- のすべての にインデックス AWS リージョン を作成します AWS アカウント。
- 指定したリージョンのインデックスをアカウントのアグリゲーターインデックスとして更新します。
- アグリゲーターインデックスのリージョンにデフォルトビューを作成します。このビューにはフィルターがないため、インデックスで見つかったすべてのリソースを検索結果として返します。

最小限必要なアクセス権限

以下の手順のステップを実行するには、次のアクセス許可が必要です。

- アクション：`resource-explorer-2:*` — リソース：特定のリソースなし (*)
- アクション：`iam:CreateServiceLinkedRole` — リソース：特定のリソースなし (*)

AWS Management Console

Quick Setup を使用して Resource Explorer をセットアップする

1. https://console.aws.amazon.com/resource-explorer で AWS Resource Explorer コンソールを開きます。
2. [Resource Explorer を有効にする] を選択します。
3. [Resource Explorer を有効にする] ページで、[Quick Setup] を選択します。
4. アグリゲーターインデックスを含めるもの AWS リージョン を選択します。ユーザーの地理的位置に適したリージョンを選択する必要があります。
5. ページの下部で、[Resource Explorer を有効にする] を選択します。
6. [進捗] ページでは、Resource Explorer がインデックスを作成する間、それぞれの AWS リージョン を監視できます。このページには、アグリゲーターインデックスの作成状況とデフォルトビューの作成状況が表示されます。

すべてのステップが正常に完了したことを示した後、管理者もユーザーも[リソース検索] ページに移動して、リソースの検索を開始できます。

> **Note**
>
> インデックスのローカルにあるタグ付きリソースは、数分以内に検索結果に表示されます。タグ付けされていないリソースは、通常 2 時間以内に表示されますが、需要が高い場合はそれより時間がかかることがあります。また、既存のすべてのローカルインデックスから新しいアグリゲーターインデックスへの最初のレプリケーションが完了するまでに最大 1 時間かかることがあります。

次のステップ：作成されたデフォルトビューでユーザーが検索できるようにするには、そのビューで検索する権限をユーザーに付与する必要があります。詳細については、「検索用の Resource Explorer ビューへのアクセス許可の付与」を参照してください。

AWS CLI

AWS アカウント を使用して で Resource Explorer を設定する AWS CLI ことは、定義上、詳細設定オプションに相当します。これは、Resource Explorer コンソールのように、Resource Explorer CLIオペレーションが自動的にステップを実行しないためです。コンソールの使用と同等のコマンドについては、「」の AWS CLI タブ詳細設定を使用して Resource Explorer をセットアップする を参照してください。

詳細設定を使用して Resource Explorer をセットアップする

詳細設定オプションを選択すると、以下ができるようになります。

- Resource Explorer AWS リージョン を有効にする を選択します。
- 一つのリージョンをアグリゲーターインデックス付きで設定するかどうかを選択できます。その場合は、配置 AWS リージョン する を指定します。アグリゲーターインデックスにより、アカウント内のすべてのリージョンのリソースを含むビューを作成できます。詳細については、「アグリゲーターインデックスを作成してクロスリージョン検索を有効にする」を参照してください。
- デフォルトビューを作成するかどうかを選択できます。このビューでは、 AWS Resource Explorer をオンにしたリージョン内の任意のリソースを自動的に検索できます。Resource Explorer での検索にそのデフォルトビューを使用する必要があるすべてのプリンシパルが、デフォルトビューへのアクセス許可を備えているか確認してください。詳細については、「検索用の Resource Explorer ビューへのアクセス許可の付与」を参照してください。

> **Note**
>
> Resource Explorer を設定して、 AWS Management Consoleの統合検索機能によって提供される検索結果に自分のリソースを含めることができます。この機能を有効にするには、すべてのロールおよびユーザーが検索できるアグリゲーターインデックスとデフォルトビューを Resource Explorer の設定に含める必要があります。[Quick Setup] オプションではアグリゲーターインデックスとデフォルトビューの両方が作成されるため、[Quick Setup] オプションで Resource Explorer を有効にすることをお勧めします。

最小限必要なアクセス権限

以下の手順のステップを実行するには、次のアクセス許可が必要です。

- アクション：`resource-explorer-2:*` — リソース：特定のリソースなし (*)
- アクション：`iam:CreateServiceLinkedRole` — リソース：特定のリソースなし (*)

AWS Management Console

詳細設定を使用して Resource Explorer をオンにする

1. https://console.aws.amazon.com/resource-explorer で AWS Resource Explorer コンソールを開きます。
2. [Resource Explorer を有効にする] を選択します。
3. [Resource Explorer を有効にする] ページで、[詳細設定] を選択します。
4. ボックスのリージョン AWS リージョンで、すべての で Resource Explorer を有効にするか AWS リージョン、特定のリージョンのみで有効にするかを選択します。

   [このアカウントでは指定された AWS リージョン でのみ Resource Explorer を有効にする] を選択した場合は、検索結果に含めるリソースを持つ各リージョンを選択します。
5. [アグリゲーターインデックス] については、アグリゲーターインデックスを作成するかどうかを選択します。アグリゲーターインデックスを作成することを選択した場合、他のすべての はインデックスをこのリージョンに AWS リージョン レプリケートします。これにより、ユーザーは で選択したすべてのリージョンのリソースを検索できます AWS アカウント。アグリゲーターインデックス AWS リージョン を含む を選択します。ユーザーが最も時間を費やすリージョン、または少なくともユーザーがリソース検索の大部分を実行すると予想されるリージョンを指定することをお勧めします。

6. [デフォルトビュー] ボックスの[ビュー作成] で、デフォルトビューを作成するかどうかを選択します。このオプションは、アグリゲーターインデックスの作成を選択した場合にのみ使用できます。デフォルトビューを作成することを選択した場合、Resource Explorer はこのビューをアグリゲータインデックス AWS リージョン と同じ に配置します。これにより、デフォルトのビュー AWS リージョン に Resource Explorer を登録したすべての の結果を含めることができます。ユーザーがデフォルトビューのあるリージョンで検索を実行し、かつ特定のビューを指定しない場合、検索にはそのリージョンのデフォルトビューが使用されます。

> **Note**
>
> ユーザーがビューを使用して検索できるようにするには、そのビューを使用する権限をユーザーに付与する必要があります。詳細については、「検索用の Resource Explorer ビューへのアクセス許可の付与」を参照してください。

7. [Resource Explorer を有効にする] を選択します。

> **Note**
>
> インデックスのローカルにあるタグ付きリソースは、数分以内に検索結果に表示されます。タグ付けされていないリソースは、通常 2 時間以内に表示されますが、需要が高い場合はそれより時間がかかることがあります。また、既存のすべてのローカルインデックスから新しいアグリゲーターインデックスへの最初のレプリケーションが完了するまでに最大 1 時間かかることがあります。

AWS CLI

詳細設定を使用して Resource Explorer をセットアップする

Resource Explorer コンソールは、選択した内容に基づいて、ユーザーに代わって多くのAPIオペレーション呼び出しを実行します。次の AWS CLI コマンド例は、 を使用してコンソールの外部で同じ基本的な手順を実行する方法を示しています AWS CLI。

Example ステップ 1：目的の AWS リージョンにインデックスを作成して、Resource Explorer を有効にする

Resource Explorer をアクティブ化する各 AWS リージョン で、次のコマンドを実行します。以下のコマンド例により、 AWS CLIのデフォルトである AWS リージョン について Resource Explorer が有効化されます。

```
$ aws resource-explorer-2 create-index
{
    "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
    "CreatedAt": "2022-07-27T16:17:12.130000+00:00",
    "State": "CREATING"
}
```

Example ステップ 2: 1 つの のインデックスをアカウントのアグリゲーターインデックス AWS リージョン に更新する

Resource Explorer AWS リージョン でローカルインデックスをアカウントのアグリゲータインデックスに更新する で、次のコマンドを実行します。以下は、米国東部 (バージニア北部) のアグリゲーターインデックス (`us-east-1`) を更新するコマンドの例です。

```
$ aws resource-explorer-2 update-index-type \
    --arn arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111 \
    --type AGGREGATOR
{
    "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
    "LastUpdatedAt": "2022-07-27T16:29:49.231000+00:00",
    "State": "UPDATING",
    "Type": "AGGREGATOR"
}
```

Example ステップ 3: アグリゲータインデックス AWS リージョン を含むビューを に作成する

アグリゲータインデックスを作成した AWS リージョン で次のコマンドを実行します。以下のコマンド例では、Resource Explorer コンソールのセットアッププロセスで作成したのと同じビューが作成されます。この新しいビューには、インデックス情報の一部としてリソースに添付されたタグが含まれ、またタグキーまたは値によるリソース検索をサポートします。

```
$ aws resource-explorer-2 create-view \
    --view-name My-New-View \
    --included-properties Name=tags
{
    "View": {
        "Filters": {
            "FilterString": ""
        },
        "IncludedProperties": [
            {
                "Name": "tags"
            }
        ],
        "LastUpdatedAt": "2022-07-27T16:34:14.960000+00:00",
        "Owner": "123456789012",
        "Scope": "arn:aws:iam::123456789012:root",
        "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-New-
View/1a2b3c4d-5d6e-7f8a-9b0c-abcd22222222"
    }
}
```

Example ステップ 4: 新しいビューを のデフォルトとして設定する AWS リージョン

次は、前のステップで作成したビューをそのリージョンのデフォルトとして設定する例です。次のコマンドは、デフォルトビューを作成したのと同じ AWS リージョン で実行する必要があります。

```
$ aws resource-explorer-2 associate-default-view \
    --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-New-
View/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111
{
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-New-
View/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111"
}
```

ユーザーがビューを使用して検索できるようにするには、そのビューを使用する権限をユーザーに付与する必要があります。詳細については、「検索用の Resource Explorer ビューへのアクセス許可の付与」を参照してください。

これらのコマンドを実行すると、 AWS アカウント内の指定されたリージョンで Resource Explorer が実行されます。Resource Explorer は、リソースの詳細を含むそれぞれのリージョン

についてのインデックスを構築し維持します。Resource Explorer は、それぞれのリージョンインデックスを指定されたリージョンのアグリゲーターインデックスにリプリケートします。そのリージョンには、アカウント内のすべてのIAMロールまたはユーザーが、インデックスが作成されたすべてのリージョンのリソースを検索できるようにするビューも含まれています。

> **Note**
>
> インデックスのローカルにあるタグ付きリソースは、数分以内に検索結果に表示されます。タグ付けされていないリソースは、通常 2 時間以内に表示されますが、需要が高い場合はそれより時間がかかることがあります。また、既存のすべてのローカルインデックスから新しいアグリゲーターインデックスへの最初のレプリケーションが完了するまでに最大 1 時間かかることがあります。

AWS リージョン Resource Explorer が有効になっている を特定する

リージョン AWS リージョン に Resource Explorer のインデックスが含まれているかどうかを確認することで、どの AWS Resource Explorer がオンになっているかを特定できます。どのリージョンにインデックスがあるかを確認するには、このページの手順に従ってください。

> **Important**
>
> ユーザーは、Resource Explorer が有効になっているリージョンのみでリソースを検索できます。また、1 つのリージョンにアグリゲーターインデックスを作成して、すべてのリージョンのリソースを検索できるようにすることもできます。Resource Explorer は、Resource Explorer インデックスを含む他のすべてのリージョンのリソース情報をアグリゲーターインデックスのあるリージョンにリプリケートします。ユーザーは、Resource Explorer を使用して、インデックスのないリージョンのリソースを検索することはできません。

特定のリージョンでの Resource Explorer ステータスを確認する

Resource Explorer のインデックスがあるリージョンを確認するには、 を使用するか AWS Management Console、 AWS Command Line Interface （ AWS CLI) の コマンドを使用するか、 の APIオペレーションを使用します AWS SDK。

AWS Management Console

どのリージョンに Resource Explorer のインデックスがあるかを確認する

1. Resource Explorer コンソールの [設定] ページを開きます。
2. [インデックス] セクションのリストには、Resource Explorer インデックスを含むリージョンのみが含まれます。[タイプ] 列の値は、そのインデックスがリージョンの [ローカル] インデックスなのか、または AWS アカウントの [アグリゲーターインデックス] なのかを示します。
3. どのリージョンに Resource Explorer が含まれていないかを確認するには、[インデックスの作成] を選択します。リージョンが表示されていない場合、そのリージョンには Resource Explorer は含まれません。

AWS CLI

どのリージョンに Resource Explorer のインデックスがあるかを確認する

次のコマンドを実行して、Resource Explorer のインデックス AWS リージョン がある を確認します。

```
$ aws resource-explorer-2 list-indexes
{
    "Indexes": [
        {
            "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
            "Region": "us-east-1",
            "Type": "AGGREGATOR"
        },
        {
            "Arn": "arn:aws:resource-explorer-2:us-
west-2:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd22222222",
            "Region":"us-west-2",
            "Type":"LOCAL"
        }
    ]
}
```

で Resource Explorer を有効に AWS リージョン してリソースのインデックスを作成する

AWS Resource Explorer で最初に をオンにすると AWS アカウント、1 つ以上の でサービスのインデックスが作成されました AWS リージョン。この時 [Quick Setup] オプションを使用すると、Resource Explorer は AWS アカウントで有効化されているすべてのAWS リージョン についてインデックスを自動的に作成します。また、Resource Explorer サービスは、指定されたリージョンのインデックスをアカウントのアグリゲーターインデックスに昇格させます。[詳細設定] オプションを使用した場合は、ユーザーがインデックスを作成するリージョンを指定します。

トピック

- 特定のリージョンに Resource Explorer インデックスを作成する
- AWS オプトインリージョンに関する考慮事項

で Resource Explorer を有効にすると AWS リージョン、サービスは次のアクションを実行します。

- の最初のリージョンで Resource Explorer を起動すると AWS アカウント、Resource Explorer は という名前のアカウントにサービスにリンクされたロール`AWSServiceRoleForResourceExplorer`を作成します。このロールは、 AWS CloudTrail やタグ付けサービスなどのサービスを使用してアカウント内のリソースを検出してインデックスを作成する権限を Resource Explorer に付与します。サービスにリンクされたロールの作成は、アカウントで最初に AWS リージョン を登録した場合にのみ行われます。Resource Explorer は、後で追加するすべてのリージョンには同じサービスリンクロールを使用します。
- Resource Explorer は、指定されたリージョンにインデックスを作成し、そのリージョンのリソースに関する詳細を保存します。
- Resource Explorer は、指定されたリージョンのリソースの検出を開始し、見つかったリソースに関する情報をそのリージョンのインデックスに追加します。
- アカウントに別のリージョンのアグリゲーターインデックスがすでに存在する場合、Resource Explorer は新しいリージョンのインデックスからアグリゲーターインデックスへの情報のリプリケーションを開始し、クロスリージョン検索をサポートします。

これらのステップが完了すると、ユーザーはリソースに関する情報を検索し発見できるようになります。ユーザーは、同じリージョンまたはアグリゲーターインデックスを含むリージョンに定義されているビューのいずれかを使用して検索できます。

特定のリージョンに Resource Explorer インデックスを作成する

追加の で Resource Explorer インデックスを作成するには、 を使用する AWS リージョン か AWS Management Console、 AWS Command Line Interface （ AWS CLI) のコマンドを使用するか、 の APIオペレーションを使用します AWS SDK。一つのリージョン内に作成できるインデックスは 1 つだけです。

最小限必要なアクセス権限

以下の手順のステップを実行するには、次のアクセス許可が必要です。

- アクション：`resource-explorer-2:*` — リソース：特定のリソースなし (*)
- アクション：`iam:CreateServiceLinkedRole` — リソース：特定のリソースなし (*)

AWS Management Console

で Resource Explorer インデックスを作成するには AWS リージョン

1. Resource Explorer の [設定] ページに移動します。
2. [インデックス] セクションで、[インデックスの作成] を選択します。
3. インデックスの作成ページで、そのリージョン AWS リージョン のリソースの検索をサポートするインデックスを作成する の横にあるチェックボックスをオンにします。選択できないチェックボックスは、すでに Resource Explorer インデックスが格納されているリージョンを示します。
4. (オプション) [タグ] セクションで、当該インデックス向けタグキーと値のペアを指定します。
5. [インデックスの作成] を選択します。

   成功すると、Resource Explorer はページの上部に緑色のバナーを表示します。選択した 1 つ以上のリージョンでインデックスを作成する際にエラーが発生した場合は赤色のバナーが表示されます。

   > **Note**
   >
   > インデックスのローカルにあるタグ付きリソースは、数分以内に検索結果に表示されます。タグ付けされていないリソースは、通常 2 時間以内に表示されますが、需要が高い場合はそれより時間がかかることがあります。また、既存のすべてのローカル

インデックスから新しいアグリゲーターインデックスへの最初のレプリケーションが完了するまでに最大 1 時間かかることがあります。

次のステップ — アグリゲーターインデックスがすでに作成されている場合、新しいリージョンは自動的にインデックス情報をアグリゲーターインデックスにリプリケートし始めます。それがユーザーがすべての検索を行う場所である場合、新しいリージョンのリソースが検索結果に表示されるようになれば完了です。

ただし、新しくインデックスされたリージョンのみでユーザーがリソースを検索できるようにする場合は、そのリージョンのユーザー用のビューも作成し、そのビューに対する権限をユーザーに付与する必要があります。ビューを作成する手順については、「検索アクセス許可を提供するための Resource Explorer ビューの管理」を参照してください。

AWS CLI

で Resource Explorer インデックスを作成するには AWS リージョン

インデックスを作成する各 AWS リージョン に対して次のコマンドを実行して、そのリージョンのリソースの検索をサポートします。以下のコマンド例は、米国東部 (バージニア北部) (`us-east-1`) に Resource Explorer を登録します。

```
$ aws resource-explorer-2 create-index \
    --region us-east-1
{
    "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
    "CreatedAt": "2022-11-01T20:00:59.149Z",
    "State": "CREATING"
}
```

Resource Explorer をオンにするリージョンごとにこのコマンドを繰り返し、`--region` パラメーター内の該当するリージョンコードを置き換えます。

Resource Explorer はインデックス作成作業の一部をバックグラウンドで非同期タスクとして実行するため、応答が `CREATING` になることがあります。これは、バックグラウンドプロセスがまだ完了していないことを示しています。

> **Note**
>
> インデックスのローカルにあるタグ付きリソースは、数分以内に検索結果に表示されます。タグ付けされていないリソースは、通常 2 時間以内に表示されますが、需要が高い場合はそれより時間がかかることがあります。また、既存のすべてのローカルインデックスから新しいアグリゲーターインデックスへの最初のレプリケーションが完了するまでに最大 1 時間かかることがあります。

次のコマンドを実行して ACTIVE の状態を確認することで、最終的な完了を確認できます。

```
$ aws resource-explorer-2 get-index \
    --region us-east-1
{
    "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
    "CreatedAt": "2022-07-12T18:59:10.503000+00:00",
    "LastUpdatedAt": "2022-07-13T18:41:58.799000+00:00",
    "ReplicatingFrom": [],
    "State": "ACTIVE",
    "Tags": {},
    "Type": "LOCAL"
}
```

次のステップ — アグリゲーターインデックスがすでに作成されている場合、新しいリージョンは自動的にインデックス情報をアグリゲーターインデックスにリプリケートし始めます。それがユーザーがすべての検索を行う場所である場合、新しいリージョンのリソースが検索結果に表示されるようになれば完了です。

ただし、新しくインデックスされたリージョンのみでユーザーがリソースを検索できるようにする場合は、そのリージョンのユーザー用のビューも作成し、そのビューに対する権限をユーザーに付与する必要があります。ビューを作成する手順については、「検索アクセス許可を提供するための Resource Explorer ビューの管理」を参照してください。

AWS オプトインリージョンに関する考慮事項

オプトインリージョンは、オプトインリージョンのアカウントを介したIAMデータの共有に関連するため、商用リージョンよりもセキュリティ要件が高くなります。IAM サービスを通じて管理されるすべてのデータは、アイデンティティデータと見なされます。

オプトインリージョンはAWS Resource Explorer コンソールを使用して有効にできます。詳細については、「 で Resource Explorer を有効にする AWS リージョン 」を参照して、リソースのインデックスを作成します。

オプトアウト挙動

オプトインリージョンをオプトアウトする前に、以下の挙動を考慮してください。

> **Important**
>
> アグリゲーターインデックスのあるリージョンをオプトアウトする前に、アグリゲーターインデックスを削除するか、ローカルインデックスに降格することをお勧めします。Resource Explorer は、パーティション内のすべてのリージョンで 1 つのアグリゲーターインデックスをサポートします。

- インデックスは削除されず、無効化されるだけです。後でもう一度オプトインすると、設定は元に戻ります。
- IAM は、リージョン内のリソースIAMへのアクセスを無効にします。
- Resource Explorer は、オプトアウトしたリージョンのインデックスを無効にし、データの取り込みを停止します。`ListIndexes` API はリージョンインデックスを表示しなくなります。
- アグリゲーターインデックスが別のリージョンにある場合、Resource Explorer はオプトアウトしたリージョンからのデータレプリケーションを停止し、24 時間以内にデータをクリーンアップします。
- アグリゲーターインデックスリージョンからオプトアウトした場合、インデックスを削除または降格するには再度オプトインする必要があります。
- リージョンに再度オプトインすると、Resource Explorer はインデックスを再び有効にし、データの取り込みを開始します。
- オプトインリージョンのステータスを変更すると、変更が有効になるまでに約 24 時間かかります。

アグリゲーターインデックスを作成してクロスリージョン検索を有効にする

クロスリージョン検索を有効にすると、 内のすべてのリージョンのリソースを検索できます AWS アカウント。

トピック

- アグリゲーターインデックスについて
- ローカルインデックスをアカウントのアグリゲーターインデックスに昇格させる
- アグリゲーターインデックスをローカルインデックスに降格させる

アグリゲーターインデックスについて

AWS Resource Explorer は、リソースについて収集した情報を、Resource Explorer AWS リージョン がそのリージョンで作成および維持するローカルインデックスに保存します。例えば、米国西部 (オレゴン) リージョンに Amazon EC2インスタンスがあるとします。Resource Explorer は、そのリソースの詳細を、米国西部 (オレゴン) リージョンにあるローカルインデックスに保存します。

アカウント AWS リージョン 内のすべてのリソースの検索をサポートするには、1 つのリージョンのローカルインデックスをアカウントのアグリゲータインデックスに変換します。

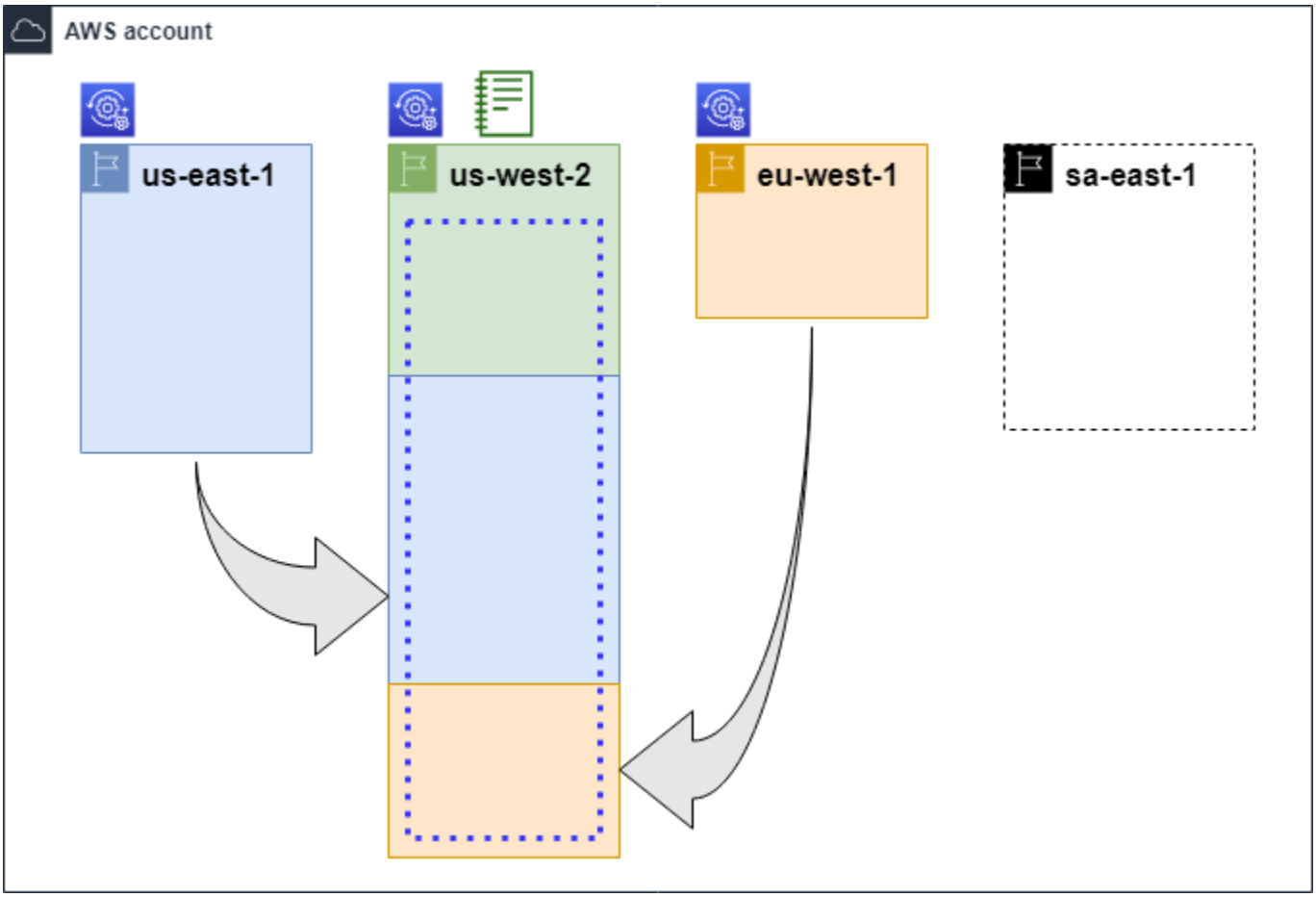
アグリゲーターインデックスには、Resource Explorer を有効にした他のすべてのリージョンにあるローカルインデックスのリプリケートコピーが含まれます。これにより、アカウント内のすべてのリソースを結果に含めることができるアグリゲータインデックスを含む リージョン AWS リージョンにビューを作成できます。

次の図は、アグリゲーターインデックスの動作例を示しています。この例 AWS アカウントでは、管理者は次のことを行います。

- これらのリージョンにインデックスを作成して、3 つの AWS リージョン （`us-east-1`、`us-west-2`、および `eu-west-1`) で Resource Explorer を有効にします。各リージョンには独自のローカルインデックスが含まれます。
- `sa-east-1` リージョンにインデックスを作成しないことを選択すると、ユーザーは `sa-east-1` についての検索を実行できず、そのリージョンからのリソースはどの検索結果にも表示されなくなります。

- アカウントのアグリゲーターインデックスを us-west-2 リージョンに作成します。これにより、Resource Explorer は、Resource Explorer がオンになっている他のすべてのリージョンのローカルインデックスからの情報をアグリゲーターインデックスにリプリケートします。これにより、Resource Explorer がオンになっている 3 つのリージョンすべてのリソースを検索対象にすることができます。us-west-2

この設定では、ユーザーはアグリゲーターインデックスを含む us-west-2 のみでクロスリージョン検索を実行することができます。そのリージョンからのビューのみが、アカウント内のすべてのリージョンからの検索結果を返すことができます。



凡例	
	Resource Explorer はこの で有効になっており AWS リージョン、そのリソースはそのリージョンのインデックスにカタログ化されます。このリージョンのインデックスは、アグリゲータインデックス AWS リージョン を含む にもレプリケートされます (矢印で示されます) 。

	これには、アグリゲータインデックス AWS リージョン が含まれます。Resource Explorer は、他のすべての で収集されたリソース情報をこのリージョン AWS リージョン にレプリケートします。
	[Quick Setup] で作成されるデフォルトビューには、 AWS リージョン内のすべてのリソースが含まれます。

ローカルインデックスをアカウントのアグリゲーターインデックスに昇格させる

AWS Resource Explorer を初めて設定するときに、一つの AWS リージョン についてアグリゲーターインデックスを作成することができます。詳細については、「Resource Explorer のセットアップと設定」を参照してください。ここに記載する手順は、初期設定時にローカルインデックスのいずれかをアカウントのアグリゲーターインデックスに昇格させなかった場合に、後でいずれかのローカルインデックスをアカウントのアグリゲーターインデックスに昇格させるためのものです。

> **Important**
>
> - 一つの AWS アカウント に設定できるアグリゲーターインデックスは一つのみです。アカウントにすでにアグリゲーターインデックスがある場合は、まずそのアグリゲーターインデックスをローカルインデックスに降格させるか、削除する必要があります。
> - アグリゲーターインデックスが含まれるリージョンを削除または変更した場合は、別のインデックスをアグリゲーターインデックスに昇格できるようになるまでに 24 時間待つ必要があります。

AWS Management Console

ローカルインデックスをアカウントのアグリゲーターインデックスに昇格させるには

1. Resource Explorer の [設定] ページを開きます。
2. [インデックス] セクションで、昇格するインデックスの横にあるチェックボックスを選択し、[インデックスタイプを変更する] を選択します。
3. [<リージョン名> のインデックスタイプを変更する] ダイアログで、[アグリゲーターインデックス] を選択し、[変更を保存] を選択します。

AWS CLI

ローカルインデックスをアカウントのアグリゲーターインデックスに昇格させるには

次のコマンド例では、指定された AWS リージョン のインデックスを LOCAL タイプから AGGREGATOR タイプへと更新します。アグリゲーターインデックスを含める AWS リージョン からオペレーションを呼び出す必要があります。

```
$ aws resource-explorer-2 update-index-type \
    --arn arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111 \
    --type AGGREGATOR \
    --region us-east-1
{
    "Arn":"arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
    "LastUpdatedAt":"2022-07-13T18:41:58.799Z",
    "State":"UPDATING",
    "Type":"AGGREGATOR"
}
```

このオペレーションは非同期的に実行され、State を UPDATING に設定して開始します。オペレーションが完了したかどうかを確認するには、次のコマンドを実行して、State 応答フィールドに ACTIVE 値が表示されるかを確認します。このコマンドは、チェックするインデックスを含むリージョンで実行する必要があります。

```
$ aws resource-explorer-2 get-index --region us-east-1
{
    "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
    "CreatedAt": "2022-10-12T21:31:37.277000+00:00",
    "LastUpdatedAt": "2022-10-12T21:31:37.677000+00:00",
    "ReplicatingFrom": [
        "us-west-2",
        "us-east-2",
        "us-west-1"
    ],
    "State": "ACTIVE",
    "Tags": {},
    "Type": "AGGREGATOR"
}
```

アグリゲーターインデックスをローカルインデックスに降格させる

アグリゲーターインデックスを別の AWS リージョン に移動する場合などに、アグリゲーターインデックスをローカルインデックスに降格することができます。

アグリゲーターインデックスをローカルインデックスに降格させると、Resource Explorer は他の AWS リージョン からのインデックスのレプリケーションを停止します。また、他のリージョンからリプリケートされた情報を削除する非同期のバックグラウンドタスクも開始されます。その非同期タスクが完了するまでは、一部のクロスリージョンの結果が検索結果に表示され続けることがあります。

> **注意**
>
> - アグリゲーターインデックスを降格させた後、同じインデックスまたは別のリージョンのインデックスをそのアカウントの新しいアグリゲーターインデックスに昇格できるようになるまで、24 時間待つ必要があります。
> - アグリゲータインデックスを降格させた後、バックグラウンド処理が完了し、他のリージョンからのすべてのリソース情報がそのリージョンで実行される検索結果から消えるまでに最大 36 時間かかることがあります。
> - 組織全体のビュー内でメンバーアカウントを降格させると、そのメンバーはマルチアカウント検索から削除されることがあります。

バックグラウンドタスクのステータスを確認するには、設定ページでインデックスのリストを表示するか、GetIndexオペレーションを使用します。非同期タスクが完了すると、そのインデックスの `Status` フィールドは `UPDATING` から `ACTIVE` に変わります。その状態では、ローカルリージョンの結果のみがクエリ結果に表示されます。

AWS Management Console

アグリゲーターインデックスをローカルインデックスに降格させるには

1. Resource Explorer の [設定] ページを開きます。
2. [インデックス] セクションで、ローカルインデックスに降格させるアグリゲーターインデックスを含むリージョンの横にあるチェックボックスを選択し、[インデックスタイプを変更する] を選択します。

3. [<リージョン名> のインデックスタイプを変更する] ダイアログで、[ローカルインデックス] を選択し、[変更を保存] を選択します。

AWS CLI

アグリゲーターインデックスをローカルインデックスに降格させるには

次の例では、指定されたアグリゲーターインデックスをローカルインデックスに降格します。現在アグリゲーターインデックスが含まれている AWS リージョン でオペレーションを呼び出す必要があります。

```
$ aws resource-explorer-2 update-index-type \
    --arn arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111 \
    --type LOCAL \
    --region us-east-1
{
    "Arn":"arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
    "LastUpdatedAt":"2022-07-13T18:41:58.799Z",
    "State":"UPDATING",
    "Type":"LOCAL"
}
```

このオペレーションは非同期的に実行され、State を UPDATING に設定して開始します。オペレーションが完了したかどうかを確認するには、次のコマンドを実行して、State 応答フィールドに ACTIVE 値が表示されるかを確認します。このコマンドは、チェックするインデックスを含むリージョンで実行する必要があります。

```
$ aws resource-explorer-2 get-index --region us-east-1
{
    "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
    "CreatedAt": "2022-10-12T21:31:37.277000+00:00",
    "LastUpdatedAt": "2022-10-12T21:31:37.677000+00:00",
    "ReplicatingFrom": [
        "us-west-2",
        "us-east-2",
        "us-west-1"
    ],
    "State": "ACTIVE",
```

```
    "Tags": {},
    "Type": "LOCAL"
}
```

マルチアカウント検索を有効にする

マルチアカウント検索を使用すると、 AWS Organizations または組織単位 (OU) でアクティブなインデックスを持つアカウント間でリソースを検索できます。

トピック

- 前提条件
- マルチアカウント検索を有効にする
- マルチアカウントの Quick Setup
- アカウントアクションが Resource Explorer のマルチアカウント検索に及ぼす影響

前提条件

組織のマルチアカウント検索を有効にするには、次の手順を実行します。

- オプトインリージョン の場合、マルチアカウント検索を有効にする管理アカウントもオプトインされていることを確認します。
- 管理者ユーザーを作成します。
- `aws iam create-service-linked-role --aws-service-name resource-explorer-2.amazonaws.com` を使用して、管理者アカウント内にサービスにリンクされたロールを作成します。
- で信頼されたアクセスを有効にします AWS Organizations。これにより、Resource Explorer との完全な統合が可能になり、組織のすべてのアカウントにわたるリソースを一覧表示できます。
- 委任管理者を割り当てます (推奨)。詳細については、「 AWS Organizations ユーザーガイド」の「Organizations と連携する AWS サービスの委任管理者」を参照してください。
  - Resource Explorer は、管理アカウントと同様のアクションを実行する委任管理者を 1 人だけサポートします。
  - 組織の委任管理者を削除または変更すると、そのアカウントで作成されたすべてのマルチアカウントビューが削除されます。

マルチアカウント検索を有効にする

組織のアカウント全体でリソースを検索して見つけるには、以下のステップを完了する必要があります。

1. AWS Resource Explorer の 1 つ以上のアカウントで をアクティブ化します AWS Organizations。
2. アグリゲーターインデックスを格納する 1 つのリージョンを登録します。
3. アグリゲーターインデックスを作成するリージョンを選択してください。このリージョンは、 全体で一貫している必要があります AWS Organizations。
4. AWS Organizations または組織単位を対象とする Resource Explorer ビューを作成します。このビューは、前のステップで登録したアグリゲーターリージョンに作成してください。
5. このビューを組織全体のアカウントと共有します。

マルチアカウントの Quick Setup

Quick Setup を使用して、組織内の複数のアカウント間で Resource Explorer を有効にできます。

> **Note**
>
> このプロセスでは、管理アカウントにリソースはデプロイされません。管理アカウントを使用していて、アカウントにインデックスが必要な場合は、Resource Explorer のオンボーディングフローを使用して手動で追加する必要があります。

1. Systems Manager コンソール内の、Resource Explorer の [Quick Setup] に移動します。
2. [アグリゲーターインデックスリージョン] を選択します。これにより、選択したターゲットアカウントのすべてのリージョンにあるリソースを検索できます。選択したターゲットアカウントのいずれかに別のリージョンですでにアグリゲーターインデックスが設定されている場合、既存のアグリゲーターインデックスは自動的にこの新しいリージョンに置き換えられます。
3. アカウントの [ターゲット] を選択します。Resource Explorer は、組織全体または特定の組織単位 () に対して有効にできますOUs。

   > **Note**
   >
   > 一度に最大 50,000 AWS CloudFormation スタックまでデプロイできます。複数のリージョンにまたがる大規模な組織の場合は、OU レベルでより小さなバッチ単位でデプロイしてください。

4. [作成] を選択する前に、確認事項のサマリーを確認してください。

アカウントアクションが Resource Explorer のマルチアカウント検索に及ぼす影響

> **Note**
> マルチアカウント検索結果からアカウントやリソースを削除するには、最大 24 時間かかります。

アカウントアクションは、 AWS Resource Explorer マルチアカウント検索に次の影響を与えます。

Resource Explorer を無効にする

アカウントの Resource Explorer を無効にすると、無効にする際に選択した AWS リージョン でのみアカウントの Resource Explorer が無効になります。

Resource Explorer を有効にしている全リージョンで、個別に Resource Explorer を無効化する必要があります。

24 時間が経過すると、このアカウントのリソースは検索結果に表示されなくなります。

Resource Explorer の他のデータや設定は削除されません。

メンバーアカウントが組織から削除されている

メンバーアカウントが組織から削除されると、Resource Explorer 管理者アカウントはそのメンバーアカウント内のリソースを閲覧する権限を失います。

削除されたアカウントが管理者アカウントまたは委任管理者アカウントであった場合、これらのアカウントによってそれまでに作成されたマルチアカウントビューもすべて削除されます。

Resource Explorer は引き続き両方のアカウント内で実行されます。

リソース検索結果には、このアカウントからのリソースは含まれなくなります。

アカウントの停止

アカウントが で停止されると AWS、そのアカウントは Resource Explorer でリソースを表示するアクセス許可を失います。停止されているアカウントの管理者アカウントは、既存のリソースを閲覧できます。

組織アカウントの場合、メンバーアカウントのステータスが [Account Suspended] (アカウントの停止) に変更されることもあります。これは、管理者アカウントがアカウントを有効にしようとしたときにアカウントが停止されている場合に発生します。[アカウントの停止] になっているアカウントの管理者アカウントは、そのアカウントのリソースを閲覧することはできません。

それ以外の場合、停止ステータスによってメンバーアカウントのステータスに影響が生じることはありません。

90 日後、アカウントは削除または再有効化されます。アカウントが再度有効になると、その Resource Explorer 権限が復元されます。メンバーアカウントのステータスが [Account Suspended] (アカウントの停止) の場合、管理者アカウントでそのアカウントを手動で有効にする必要があります。

アカウントの閉鎖

AWS アカウントが閉鎖されると、Resource Explorer は閉鎖に次のように応答します。

- Resource Explorer では、アカウントの閉鎖の発効日から 90 日間にわたり、そのアカウントのリソースをします。90 日経過した時点で、そのアカウントのすべてのリソースを恒久的に削除します。
- リソースを 90 日以上保持するには、 EventBridge ルールでカスタムアクションを使用して、リソースを Amazon S3 バケットに保存できます。Resource Explorer でリソースが保持されている限り、閉鎖されたアカウントを再度開いた際に、Resource Explorer でそのアカウントのリソースを復元することができます。
- そのアカウントが Resource Explorer 管理者アカウントである場合、アカウントは管理者としても削除され、かつすべてのメンバーアカウントも削除されます。アカウントがメンバーアカウントである場合、Resource Explorer 管理者アカウントとの関連付けが解除され、メンバーとして削除されます。
- 詳細については、「アカウントの解約」を参照してください。

アカウントのオプトアウト

アカウントが特定のリージョンをオプトアウトしても、検索結果には最大 24 時間そのリソースが表示されます。

24 時間が経過すると、このアカウントのリソースは検索結果に表示されなくなります。詳細については、「オプトアウト挙動」を参照してください。

AWS Management Consoleでの統合検索のサポート

には、すべてのコンソールページの上部に検索バー AWS Management Console があります。これにより、すべての で統一された検索エクスペリエンスが提供されます AWS のサービス。統合検索結果には次のような内容が含まれます。

- AWS のサービス および 機能コンソールページ。
- AWS ドキュメントページ。
- AWS ブログとナレッジベースの記事
- アカウント内のリソース — 以下の手順を実行すると含まれるようになります。

統合検索結果にお使いのアカウントのリソースを表示するには、次の手順を実行する必要があります。これは、 の初期設定時に実行できます AWS Resource Explorer。[Quick Setup] オプションを使用すれば、これらはすべて自動的に行われます。

- のアグリゲーターインデックスは、 の 1 つの に作成する必要があります AWS アカウント。AWS リージョン
- アグリゲーターインデックスを含む AWS リージョン にデフォルトビューを作成する必要があります。
- 統合検索バーでリソースを検索する必要があるすべてのプリンシパルに、そのデフォルトビューを使用して検索する権限を付与しておく必要があります。

統合検索では、常にアグリゲータインデックス AWS リージョン を含む のデフォルトビューを使用してすべての検索を実行します。

組織内のアカウントへの Resource Explorer のデプロイ

を使用すると AWS CloudFormation StackSets、 によって組織で管理されているすべてのアカウントを定義してデプロイできます AWS Organizations。スタックセットを定義するときは、 全体 AWS リージョン および指定したすべてのターゲットアカウントで作成する AWS リソースを指定します。すべてのアカウントが同じ組織に属している場合、Organizations と AWS CloudFormation の統合を活用して、それらのサービスにクロスアカウントロールの作成を処理させることができます。組織内の自動デプロイを有効にすると、将来ターゲット組織または組織単位 (OU) に追加する新しいアカウントにスタックインスタンスが自動的にデプロイされます。組織からアカウントを削除すると、AWS CloudFormation は組織のスタックインスタンスの一部としてデプロイされたリソースもすべて自動的に削除します。の詳細については StackSets、「 ユーザーガイド」の AWS CloudFormation StackSets「 の使用AWS CloudFormation 」を参照してください。

を使用して AWS CloudFormation StackSets 、組織内のすべてのアカウント AWS Resource Explorer で を有効にして設定し、有効な各リージョンにインデックスを作成し、必要な場所にビューを作成できます。

> **Important**
>
> あるリージョンにアグリゲーターインデックスをセットアップする場合は、そのアカウントの他のリージョンに既存のアグリゲーターインデックスがないことを確認する必要があります。アグリゲーターインデックスをローカルインデックスに降格したら、別のインデックスをそのアカウントの新しいアグリゲーターインデックスに昇格できるようになるまで 24 時間待つ必要があります。

前提条件

AWS CloudFormation StackSets を使用して Resource Explorer を組織内のアカウントにデプロイするには、ユーザーまたは組織の管理者が、まず次の手順を実行して、サービス管理アクセス許可を持つスタックを有効にする必要があります。

1. その組織で、すべての機能が有効になっている必要があります。一括請求 (コンソリデーティッドビリング) 機能のみが有効になっている場合、サービス管理権限付きスタックセットを作成することはできません。
2. AWS CloudFormation と Organizations 間の信頼されたアクセスを有効にします。これにより、組織の管理アカウントに必要なロールを作成するアクセス許可が付与 AWS CloudFormation され、

メンバーアカウント AWS CloudFormation は Resource Explorer インデックスとビューをデプロイします。

これで、サービスマネージド権限付きスタックセットを作成できます。

> **Important**
>
> スタックセットは、組織の管理アカウントで作成する必要があります。 AWS CloudFormation はリージョンサービスであるため、最初に作成したリージョンからのみ、作成したスタックセットを表示および管理できます。

Resource Explorer 用スタックセットの作成

Resource Explorer を完全にデプロイするには、2 つのスタックセットをデプロイする必要があります。

- 1 つ目のスタックセットは、ユーザーがアカウント内のすべてのリージョンのリソースを検索できるアグリゲーターインデックスとデフォルトビューを作成します。

  このスタックセットを、アグリゲーターインデックスを作成する 1 つのリージョンのみにデプロイします。
- 2 つ目のスタックセットは、ローカルインデックスとデフォルトビューを作成します。ローカルインデックスは、自コンテンツをアグリゲーターインデックスにリプリケートします。

  このスタックセットを、アグリゲーターインデックスを含むリージョンを除くアカウント内の有効なすべてのリージョンにデプロイします。スタックをデプロイするアカウントで有効になっていないリージョンは選択しないでください。そのようなリージョンを選択すると、デプロイは失敗します。

それぞれのサンプルテンプレートは、以下のセクションにあります。これらのテンプレートを使用してスタックセットを作成する方法については、 step-by-stepユーザーガイドの「サービスマネージド型のアクセス許可を持つスタックセットを作成するAWS CloudFormation 」を参照してください。

これらのスタックセットを組織にデプロイすると、選択した範囲、つまり組織または組織単位のすべてのアカウントに、指定されたリージョン内ではアグリゲーターインデックスが、他のすべてのリージョンではローカルインデックスが割り当てられます。

サンプル AWS CloudFormation テンプレート

次のサンプルテンプレートは、アカウントのアグリゲーターインデックスと、インデックスをデプロイするアカウントのすべてのリージョンのリソースを検索できるデフォルトビューを作成します。

YAML

```
Description: >-
  CFN Stack setting up ResourceExplorer with an Aggregator Index, and a new Default
 View.
Resources:
  Index:
    Type: 'AWS::ResourceExplorer2::Index'
    Properties:
      Type: AGGREGATOR
      Tags:
        Purpose: ResourceExplorer CFN Stack
  View:
    Type: 'AWS::ResourceExplorer2::View'
    Properties:
      ViewName: DefaultView
      IncludedProperties:
        - Name: tags
      Tags:
        Purpose: ResourceExplorer CFN Stack
    DependsOn: Index
  DefaultViewAssociation:
    Type: 'AWS::ResourceExplorer2::DefaultViewAssociation'
    Properties:
      ViewArn: !Ref View
```

JSON

```
{
    "Description": "CFN Stack setting up ResourceExplorer with an Aggregator Index,
 and a new Default View.",
    "Resources": {
        "Index": {
            "Type": "AWS::ResourceExplorer2::Index",
            "Properties": {
                "Type": "AGGREGATOR",
                "Tags": {
                    "Purpose": "ResourceExplorer CFN Stack"
```

```
                }
            }
        },
        "View": {
            "Type": "AWS::ResourceExplorer2::View",
            "Properties": {
                "ViewName": "DefaultView",
                "IncludedProperties": [{
                    "Name": "tags"
                }],
                "Tags": {
                    "Purpose": "ResourceExplorer CFN Stack"
                }
            },
            "DependsOn": "Index"
        },
        "DefaultViewAssociation": {
            "Type": "AWS::ResourceExplorer2::DefaultViewAssociation",
            "Properties": {
                "ViewArn": {
                    "Ref": "View"
                }
            }
        }
    }
}
```

次のサンプルテンプレートは、アグリゲーターインデックスを持つリージョン以外で、すべてのアカウントで有効化されている各リージョンにローカルインデックスを作成します。また、ユーザーがそのリージョンのみのリソースを検索できるデフォルトビューも作成されます。すべてのリージョンのリソースを検索するには、ユーザーはアグリゲーターリージョン付きのビューで検索する必要があります。

YAML

```
Description: >-
  CFN Stack setting up ResourceExplorer with a Local Index, and a new Default View.
Resources:
  Index:
    Type: 'AWS::ResourceExplorer2::Index'
    Properties:
      Type: LOCAL
```

```
      Tags:
        Purpose: ResourceExplorer CFN Stack
  View:
    Type: 'AWS::ResourceExplorer2::View'
    Properties:
      ViewName: DefaultView
      IncludedProperties:
        - Name: tags
      Tags:
        Purpose: ResourceExplorer CFN Stack
    DependsOn: Index
  DefaultViewAssociation:
    Type: 'AWS::ResourceExplorer2::DefaultViewAssociation'
    Properties:
      ViewArn: !Ref View
```

JSON

```
{
    "Description": "CFN Stack setting up ResourceExplorer with a Local Index, and a
 new Default View.",
    "Resources": {
        "Index": {
            "Type": "AWS::ResourceExplorer2::Index",
            "Properties": {
                "Type": "LOCAL",
                "Tags": {
                    "Purpose": "ResourceExplorer CFN Stack"
                }
            }
        },
        "View": {
            "Type": "AWS::ResourceExplorer2::View",
            "Properties": {
                "ViewName": "DefaultView",
                "IncludedProperties": [{
                    "Name": "tags"
                }],
                "Tags": {
                    "Purpose": "ResourceExplorer CFN Stack"
                }
            },
            "DependsOn": "Index"
```

```
        },
        "DefaultViewAssociation": {
            "Type": "AWS::ResourceExplorer2::DefaultViewAssociation",
            "Properties": {
                "ViewArn": {
                    "Ref": "View"
                }
            }
        }
    }
}
```

Resource Explorer をオフにする

特定の でリソースを検索する必要がなくなった場合は AWS リージョン、インデックスを削除してそのリージョン AWS Resource Explorer でのみオフにすることも、すべての で Resource Explorer を削除することもできます AWS リージョン。これを行うと、Resource Explorer はそのリージョン内の新規または更新されたリソースのスキャンを停止します。アカウントにアグリゲーターインデックスが含まれている場合、削除されたインデックスからのレプリケーションは停止し、削除されたインデックスからの情報はアグリゲーターインデックスから削除され、検索結果に表示されなくなります。アグリゲーターインデックスのあるリージョンの検索結果から削除されたインデックスのすべてのリソースが消えるまでに、最大 24 時間かかることがあります。

> **Note**
>
> 最初の を登録すると AWS リージョン、Resource Explorer は に という名前のサービスリンクロール (SLR) `AWSServiceRoleForResourceExplorer` を作成します AWS アカウント。Resource Explorer はこれを自動的に削除しません。 SLRアカウント内のすべてのリージョンで Resource Explorer インデックスを削除した後、今後 Resource Explorer を使用しないSLR場合は、IAMコンソールを使用して を削除できます。ロールを削除し、少なくとも 1 つの で Resource Explorer を再度有効にすることを選択した場合 AWS リージョン、Resource Explorer はサービスにリンクされたロールを自動的に再作成します。

1 つの で Resource Explorer をオフにする AWS リージョン

で Resource Explorer をオフにするには、 を使用する AWS リージョン か AWS Management Console、 AWS Command Line Interface （ AWS CLI) のコマンドを使用するか、 で APIオペレーションを使用します AWS SDK。

メンバーアカウントの Resource Explorer をオフにした時、そのメンバーが組織全体のビューに府含まれていた場合には、そのメンバーはマルチアカウント検索結果から削除されます。

アカウント内の 1 つまたは複数の AWS リージョン でリソース検索をサポートする必要がなくなった場合は、次に説明する手順を実行します。

> **Note**
>
> 削除するインデックスが のアグリゲーターインデックスである場合は AWS アカウント、別のローカルインデックスをアカウントのアグリゲーターインデックスに昇格させるには、24

時間待つ必要があります。別のアグリゲーターインデックスが設定されるまで、ユーザーは Resource Explorer を使用してアカウント全体の検索を実行することはできません。

AWS Management Console

で Resource Explorer インデックスを削除するには AWS リージョン

1. Resource Explorer の [設定] ページを開きます。
2. インデックス セクションで、削除するインデックス AWS リージョン がある の横にあるチェックボックスを選択し、「削除」を選択します。
3. [インデックスの削除] ページで、削除するインデックスのみが選択されていることを確認します。[確認] テキストボックスに **delete** と入力して、[インデックスの削除] をクリックします。

   成功した場合、Resource Explorer は緑色のバナーをページ上部に表示します。選択した 1 つ以上のリージョンでエラーが発生した場合は赤色のバナーが表示されます。

AWS CLI

で Resource Explorer インデックスを削除するには AWS リージョン

AWS リージョン アカウントの 1 つ以上の でリソースの検索をサポートしなくなった場合は、次のコマンドを実行します。

削除するインデックスのあるリージョンごとに以下のコマンドを実行します。このコマンドは、削除するインデックスのあるリージョンで実行する必要があります。次のコマンド例では、米国西部 (オレゴン) (`us-west-2`) の Resource Explorer インデックスを削除します。

```
$ aws resource-explorer-2 delete-index \
    --arn arn:aws:resource-explorer-2:us-
west-2:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd22222222 \
    --region us-west-2
{
    "Arn": "arn:aws:resource-explorer-2:us-
west-2:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd22222222",
    "State": "DELETING"
}
```

Resource Explorer は削除クリーンアップ作業の一部をバックグラウンドで非同期タスクとして実行するため、オペレーションが DELETING であることがレスポンスに示される場合があります。このステータスは、バックグラウンドプロセスがまだ完了していないことを示しています。次のコマンドを実行し、State が DELETED に変わっているかどうかを確認することで、最終的に完了したかどうかを確認できます。

```
$ aws resource-explorer-2 get-index \
    --region us-west-2
{
    "Arn": "arn:aws:resource-explorer-2:us-
west-2:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
    "CreatedAt": "2022-07-12T18:59:10.503000+00:00",
    "LastUpdatedAt": "2022-07-13T18:41:58.799000+00:00",
    "ReplicatingFrom": [],
    "State": "DELETED",
    "Tags": {},
    "Type": "LOCAL"
}
```

すべての で Resource Explorer をオフにする AWS リージョン

AWS Resource Explorer 完全にオフにする場合は、次の手順を実行します。

> **Note**
>
> Resource Explorer は、アカウント AWS リージョン の最初の AWSServiceRoleForResourceExplorer にインデックスを作成するときに、アカウントにという名前のサービスにリンクされたロールを作成します。Resource Explorer は、このサービスリンクロールを自動的に削除しません。すべてのリージョンで Resource Explorer インデックスを削除した後、今後 Resource Explorer を再度使用しないことが確実であれば、IAMコンソールを使用してロールを削除できます。ロールを削除し、少なくとも 1 つので Resource Explorer を起動することを選択した場合 AWS リージョン、Resource Explorer はサービスにリンクされたロールを再作成します。

Resource Explorer をオフにするには、 を使用するか AWS Management Console、 (AWS CLI) の AWS Command Line Interface コマンドを使用するか、 のAPIオペレーションを使用します AWS SDK。

AWS Management Console

AWS リージョン で リソースの検索をサポートしなくなった場合は AWS アカウント、次の手順を実行します。

すべての で Resource Explorer を無効にするには AWS リージョン

1. Resource Explorer の [設定] ページを開きます。
2. インデックス セクションで、登録されているすべての の横にあるチェックボックスを選択し AWS リージョン、削除 を選択します。

> **Tip**
> [インデックス] の横にあるテーブルヘッダー行のチェックボックスをオンにすると、すべてのリージョンのチェックボックスを 1 回の操作でオンにできます。

3. [インデックスの削除] ページで、すべてのインデックスを削除することを確認します。[確認] テキストボックスに **delete** と入力して、[インデックスの削除] をクリックします。

   成功した場合、Resource Explorer は緑色のバナーをページ上部に表示します。選択した 1 つ以上のリージョンでエラーが発生した場合は赤色のバナーが表示されます。

AWS CLI

すべての で Resource Explorer を無効にするには AWS リージョン

アカウント内の AWS リージョン でリソースの検索をサポートしなくなった場合は、次のコマンドを実行して、以前に Resource Explorer を有効にした各 AWS リージョン ですべてのインデックスARNの を見つけます。

```
$ aws resource-explorer-2 list-indexes --query Indexes[*].Arn[
"arn:aws:resource-explorer-2:us-east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
"arn:aws:resource-explorer-2:us-west-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd22222222",
"arn:aws:resource-explorer-2:us-west-2:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd33333333"
]
```

各レスポンスごとに以下のコマンドを実行して、そのリージョンの Resource Explorer インデックスを削除します。

```
$ aws resource-explorer-2 delete-index \
    --arn arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111 \
    --region us-east-1
{
    "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
    "State": "DELETING"
}
```

追加のリージョンごとに前のコマンドを繰り返します。

Resource Explorer はクリーンアップの一部をバックグラウンドで非同期タスクとして実行するため、オペレーションが DELETING であることがレスポンスに示される場合があります。このステータスは、バックグラウンドプロセスがまだ完了していないことを示しています。次のコマンドを実行し、ステータスが DELETED に変わったかどうかを確認することで、最終完了を確認できます。

```
$ aws resource-explorer-2 get-index \
    --region us-east-1
{
    "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
    "CreatedAt": "2022-07-12T18:59:10.503000+00:00",
    "LastUpdatedAt": "2022-07-13T18:41:58.799000+00:00",
    "ReplicatingFrom": [],
    "State": "DELETED",
    "Tags": {},
    "Type": "LOCAL"
}
```

検索アクセス許可を提供するための Resource Explorer ビューの管理

ビューはリソース検索のカギとなる要素です。すべての AWS Resource Explorer 検索オペレーションでビューを使用する必要があります。ビューは、管理者が AWS アカウント内のリソースに関する情報へのアクセスを制御するために使用する手段です。

ビューには、そのビューを使用するアクセス許可を持つプリンシパル (IAM ロールまたはユーザー) のみがアクセスできます。Resource Explorer で正常に検索するには、プリンシパルがビューの に対する オペレーション`resource-explorer-2:GetView`と `resource-explorer-2:Search`オペレーションの両方`Allow`にアクセスできる必要がありますARN。

ビューには組み込みのフィルタが含まれており、管理者はこれを使用して表示される結果が目的の項目のみになるよう制限できます。例えば、特定のプロジェクトに関連するリソースのみを含むビューを作成できます。他のプロジェクトに関する情報を閲覧する必要がないユーザーは、このビューを使用して目的のリソースのみを閲覧できます。

ビューはリージョンベースのリソースです。ビューは特定の AWS リージョン 内で作成および保存され、そのリージョンのインデックスからの情報のみを検索結果として返します。アカウント内のすべてのリージョンの結果を含めるには、そのビューがアグリゲーターインデックスを格納したリージョンにある必要があります。そのリージョンには、アカウント内の他のすべてのリージョンのインデックスの複製が含まれています。

各ビューには以下のようないくつかの重要な要素があります。

検索権限

標準の AWS アクセス許可ポリシーを使用して、各ビューを使用できるユーザーを制御できます。これは、各プリンシパルにアタッチされている ID ベースのアクセス許可ポリシーによって実現されます。これにより、各ビューで提供される情報を誰が見ることができるかをきめ細かく制御できます。例えば、`Production-resources` ビューへのアクセス権を付与して、生産サービスを運営するエンジニアだけがそのビューから検索できるようにすることができます。さらに、`Pre-production-resources` ビューに異なる権限を付与することで、開発者が量産前リソースを検索できるようにすることもできます。

プリンシパル`AWSResourceExplorerReadOnlyAccess`で という名前の管理 AWS ポリシーを使用すると、アカウント内の任意のビューを使用して検索できるようになります。

または、独自のアクセス許可ポリシーを作成して、指定したビューのみに以下のアクセス許可を付与することもできます。

- `resource-explorer-2:GetView`
- `resource-explorer-2:Search`

アクセス権限を付与するには、ユーザー、グループ、またはロールにアクセス許可を追加します。

- のユーザーとグループ AWS IAM Identity Center：

  アクセス許可セットを作成します。「AWS IAM Identity Center ユーザーガイド」の「権限設定を作成する」の手順に従ってください。

- ID プロバイダーIAMを介して で管理されるユーザー：

  ID フェデレーションのロールを作成します。「 IAMユーザーガイド」の「サードパーティー ID プロバイダーのロールの作成 (フェデレーション) 」の手順に従います。

- IAM ユーザー：
  - ユーザーが担当できるロールを作成します。「 IAMユーザーガイド」のIAM「 ユーザーのロールの作成」の手順に従います。
  - (お奨めできない方法) ポリシーをユーザーに直接アタッチするか、ユーザーをユーザーグループに追加する。「 ユーザーガイド」の「ユーザーへのアクセス許可の追加 (コンソール) IAM」の指示に従います。

ビュー関連アクセス許可の詳細については、「検索用の Resource Explorer ビューへのアクセス許可の付与」を参照してください。

検索のフィルタ処理

ビューは、ユーザーがアカウント内のリソースを確認できる仮想ウィンドウとして機能します。複数のビューを作成して、それぞれに異なる全体像を表現させることができます。例えば、リソースに付けられたタグで識別される、量産前環境に関連するリソースのみを検索できるビューを作成することができます。あるいは、タグ内のさまざまな値に基づいて、本番環境内のリソースのみを検索できる別のビューを作成することもできます。複数のビューに異なる `FilterString` 値を設定することで、検索するたびにそれらのクエリパラメータを再入力する必要がなくなります。

ビューでは、リソースに関するどのオプション情報を結果に含めるかを指定することもできます。デフォルトのフィールドリストは常に結果に含まれます。デフォルトのリストに加えて、リ

ソースに添付されているタグ、および () AWS Organizations の情報もビューに含めるようにリクエストできます。

検索範囲

- リージョンの範囲 – Resource Explorer で を検索する AWS リージョン と、結果には、そのリージョンでインデックスが作成されたリソースのみを含めることができます。ほとんどのリージョンのインデックスには、そのリージョン内のリソースに関する情報しか含まれていないため、`LOCAL` のラベルが付けられています。これらのリージョンを検索すると、それらのリソースのみが検索結果として返されます。
- アカウント範囲 — 1 つのローカルインデックスをアカウントのアグリゲーターインデックスに昇格できます。これを行うと、Resource Explorer がオンになっている他のすべてのリージョンは、アグリゲーターインデックスのあるリージョンに自リージョンのインデックス情報をリプリケートします。そのリージョンを検索すると、その結果にはアカウント内のすべてのリージョンのリソースが含まれます。[Quick Setup] オプションでサーバーを設定すると、Resource Explorer は指定したリージョンにアグリゲーターインデックスを自動的に作成します。また、[Quick Setup] オプションを使用すると、そのリージョンにデフォルトビューが作成され、すべてのリージョンのアカウント内のすべてのリソースを検索できるようになります。

デフォルトビュー

ユーザーが特定のビューを指定せずに検索を試みると、Resource Explorer はその AWS リージョンについて定義されているデフォルトビューを使用します。

そのリージョンのデフォルトビューが存在せず、ユーザーが使用するビューを指定しなかった場合、検索は失敗し、例外が生成されます。

Resource Explorer は、以下のプロセスでデフォルトビューを自動的に作成します。

- を使用して Resource Explorer を有効に AWS Management Console し、クイックセットアップオプションを選択する場合は、アカウントのアグリゲータインデックスを含むリージョンを指定する必要があります。Resource Explorer は、指定されたアグリゲーターインデックスリージョンにデフォルトビューを自動的に作成します。
- を使用して Resource Explorer を登録 AWS Management Console し、詳細設定オプションを選択した場合、オプションで、指定したリージョンのアカウントのアグリゲータインデックスを作成できます。これを行うと、Resource Explorer はアグリゲーターインデックスリージョンにデフォルトビューを自動的に作成します。

- コンソールを使用して Resource Explorer を登録し、かつアグリゲーターインデックスリージョンを登録しないことを選択した場合、Resource Explorer は各リージョンにローカルインデックスのデフォルトビューを作成します。
- AWS CLI または APIオペレーションを使用して Resource Explorer を登録した場合、Resource Explorer は自動的にデフォルトビューを作成しません。その場合、ユーザー検索が予想される各リージョンのデフォルトビューを手動で設定する必要があります。

検索に使用する Resource Explorer ビューの作成

すべての検索にはビューを使用する必要があります。ビューは、そのビューを使用するクエリによって返されるリソースを決定するフィルターを定義します。また、ビューはどのユーザーがリソースを検索できるかも制御します。

ビューは に保存され AWS リージョン、そのリージョンのインデックスからのみ検索結果を返します。そのリージョンにアグリゲーターインデックスが格納されている場合、ビューはアカウント内のすべてのリージョンのインデックスからの検索結果を返します。

マルチアカウントビューでは、組織全体の複数のアカウントのリソースを検索できます。これには、検索するそれぞれのアカウントがインデックス化されている必要があります。組織の管理アカウントまたは委任管理者アカウントのみが、マルチアカウントビューを作成できます。

AWS Resource Explorer Systems Manager コンソールの Resource Explorer の高速セットアップ または詳細セットアップ で関連するオプションを選択した場合、 は初期設定時にデフォルトビューを作成できます。後からいつでも、異なるユーザーセット向けに異なるフィルタを適用したビューを追加で作成できます。

ビューを作成するには、 を使用する AWS Management Console か、 で AWS CLI コマンドまたは同等のAPIオペレーションを実行します AWS SDK。

最小アクセス許可

この手順を実行するには、次のアクセス許可が必要です。

- アクション:`resource-explorer-2:CreateView`

  リソース： これは*、アカウント内の任意の でビューを作成できるようにするため AWS リージョン です。

AWS Management Console

ビューを作成するには

1. Resource Explorer コンソールの [ビュー] ページを開き、[ビューの作成] を選択します。
2. [ビューの作成] ページの [名前] に、ビューの名前を入力します。

   名前は 64 文字以下で、文字、数字、ハイフン (-) を使用できます。名前は 内で一意である必要があります AWS リージョン。
3. ビュー AWS リージョン を作成する を選択します。アカウント内のすべてのリージョンからリソースを返すビューを作成するには、アグリゲータインデックス AWS リージョン を含む を選択します。
4. (オプション) [範囲] で、検索に対してマルチアカウントのリソースを返すか、自アカウントのリソースのみを返すかを選択します。アカウントレベルの範囲がデフォルトです。

   マルチアカウントビューを作成するオプションが表示されるのは、管理アカウントまたは委任管理者のみです。
5. 検索結果をフィルタリングするかどうかを選択します。

   - [すべてのリソースを含める]

     クエリフィルターは含まれません。そのビューに関連するインデックス内のすべてのリソースが検索結果として返されます。
   - [指定したフィルターに一致するリソースのみを含める]

     フィルターの名称と演算子を選択できる [リソースフィルター] チェックボックスをオンにします。使用可能なフィルター名と演算子の説明については、「フィルター」を参照してください。
   - このビューの結果に含めるオプションリソース属性を選択します。[タグ] の横にあるチェックボックスを選択すると、ユーザーはタグキーの名前と値に基づいてリソースを検索することができます。ビューにタグを含めないと、ユーザーはタグキーと値を使用して結果を絞り込む検索リクエストを行うことができません。
   - 必要に応じて、タグをビューにアタッチできます。[タグ] ボックスを展開して、最大 50 組のタグキーと値のペアを入力できます。タグを使用してリソースを分類したり、属性ベースのアクセスコントロール (ABAC) セキュリティ許可戦略の一部として分類したりできます。詳細については、「ビューへのタグの追加」を参照してください。
   - [ビューの作成] を選択します。

コンソールは [検索] ページに戻り、新しいビューを使用して検索を実行できます。

次のステップ: アカウントのプリンシパルに、新しいビューで検索する権限を付与します。詳細については、「検索用の Resource Explorer ビューへのアクセス許可の付与」を参照してください

AWS CLI

ビューを作成するには

以下のコマンドを実行して、指定した AWS リージョンにビューを作成できます。次の例では、`Stage`キー と値 でタグ付けされた Amazon EC2サービスに関連するリソースのみを返すビューを作成します`prod`。

```
$ aws resource-explorer-2 create-view \
    --region us-west-2 \
    --view-name "My-EC2-Prod-Resources" \
    --filters FilterString="service:ec2 tag:stage=prod" \
    --included-properties Name=tags
{
    "View": {
        "Filters": {
            "FilterString": "service:ec2 tag:stage=prod"
        },
        "IncludedProperties": [
            {
                "Name": "tags"
            }
        ],
        "LastUpdatedAt": "2022-08-03T16:13:37.625000+00:00",
        "Owner": "123456789012",
        "Scope": "arn:aws:iam::123456789012:root",
        "ViewArn": "arn:aws:resource-explorer-2:us-west-2:123456789012:view/My-EC2-
Prod-Resources/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111"
    }
}
```

組織レベルのビューを作成するには

次の例では、組織全体のリソースを返すビューを作成します。これを行うには、組織の管理アカウントまたは委任管理者アカウントにサインインする必要があります。

1. `aws organizations describe-organization` コマンドを実行して、組織 を取得しますARN。
2. 以下のコマンドを実行して、指定された組織レベルのビューを作成します。

```
$ aws resource-explorer-2 create-view \
    --region us-west-2 \
    --view-name entire-org-view \
    --scope "arn:aws:organizations::111111111111:organization/o-exampleorgid"
{
    "View": {
        "Filters": {
            "FilterString": ""
        },
        "IncludedProperties": [],
        "LastUpdatedAt": "2022-08-03T16:13:37.625000+00:00",
        "Owner": "111111111111",
        "Scope": "arn:aws:organizations::111111111111:organization/o-
exampleorgid",
        "ViewArn": "arn:aws:resource-explorer-2:us-west-2:111111111111:view/
entire-org-view/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111"
    }
}
```

組織単位レベルのビュー作成するには

次の例では、この組織単位のすべてのメンバーからのリソースを返すビューを作成します。このビューは組織レベルのビューと同様に動作します。これを行うには、組織の管理アカウントまたは委任管理者アカウントにサインインする必要があります。

1. `aws organizations describe-organizational-unit` コマンドを実行して、組織 を取得しますARN。
2. 以下のコマンドを実行して、指定された組織単位レベルのビューを作成します。

```
$ aws resource-explorer-2 create-view \
    --region us-west-2 \
    --view-name entire-ou-view \
```

```
    --scope "arn:aws:organizations::222222222222:ou/o-exampleorgid/ou-
exampleouid"
{
    "View": {
        "Filters": {
            "FilterString": ""
        },
        "IncludedProperties": [],
        "LastUpdatedAt": "2022-08-03T16:13:37.625000+00:00",
        "Owner": "222222222222",
        "Scope": "arn:aws:organizations::222222222222:ou/o-exampleorgid/ou-
exampleouid",
        "ViewArn": "arn:aws:resource-explorer-2:us-west-2:222222222222:view/
entire-ou-view/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111"
    }
}
```

次のステップ: アカウントのプリンシパルに、新しいビューで検索する権限を付与します。詳細については、「検索用の Resource Explorer ビューへのアクセス許可の付与」を参照してください

検索用の Resource Explorer ビューへのアクセス許可の付与

ユーザーが新しいビューで検索するには、そのユーザーに AWS Resource Explorer ビューへのアクセス許可を付与する必要があります。そのためには、そのビューで検索を実行する必要がある AWS Identity and Access Management (IAM) プリンシパルに ID ベースのアクセス許可ポリシーを適用します。

アクセスを提供するには、ユーザー、グループ、またはロールにアクセス許可を追加します。

- AWS IAM Identity Center のユーザーとグループ:

  アクセス許可セットを作成します。「AWS IAM Identity Center ユーザーガイド」の「シークレットの作成と管理」の手順に従ってください。

- ID プロバイダーを通じて IAM で管理されているユーザー:

  ID フェデレーションのロールを作成する。詳細については、「IAM ユーザーガイド」の「サードパーティー ID プロバイダー (フェデレーション) 用のロールの作成」を参照してください。

- IAM ユーザー:

- ユーザーが実行できるロールを作成します。手順については、「IAM ユーザーガイド」の「IAM ユーザー用ロールの作成」を参照してください。
- (非推奨) ポリシーをユーザーに直接アタッチするか、ユーザーをユーザーグループに追加します。「IAM ユーザーガイド」の「ユーザー (コンソール) へのアクセス許可の追加」の指示に従います。

次のいずれかの方法を使用します。

- 既存の AWS マネージドポリシーを使用します。Resource Explorer には、あらかじめ定義された AWS マネージドポリシーがいくつか用意されています。使用可能なすべての AWS マネージドポリシーの詳細については、AWS の マネージドポリシー AWS Resource Explorer を参照してください。

  たとえば、`AWSResourceExplorerReadOnlyAccess` ポリシーを使用して、アカウント内のすべてのビューでの検索権限を付与できます。

- 独自のアクセス許可ポリシーを作成し、プリンシパルに割り当てます。独自のポリシーを作成する場合、ポリシーステートメントの `Resource` の項目で各ビューの Amazon リソースネーム (ARN) を指定することにより、アクセスを単一のビューまたは複数のビューの特定のサブセットに制限することができます。たとえば、次のサンプルポリシーを使用して、そのプリンシパルに 1 つのビューのみを使用して検索する権限を付与できます。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "resource-explorer-2:Search",
                "resource-explorer-2:GetView"
            ],
            "Resource": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/
MyTestView/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111
        }
    ]
}
```

IAM コンソールを使用してアクセス許可ポリシーを作成し、そのアクセス許可を必要とするプリンシパルに適用します。IAM アクセス許可ポリシーの詳細については、次のトピックを参照してください。

- IAM でのポリシーとアクセス許可
- IAM ID のアクセス許可の追加および削除
- ポリシーによって付与されるアクセス許可について

タグベースの認証を使用してビューへのアクセスを制御します。

特定のリソースのみを含む結果を返すフィルター付きのビューを複数作成する場合、それらのビューへのアクセスを、それらのリソースを見る必要のあるプリンシパルのみに制限したい場合があると思います。属性ベースのアクセス制御 (ABAC) 戦略を使用することで、アカウント内のビューについてこのようなセキュリティを提供できます。ABAC が使用する属性は、AWS で操作を実行しようとするプリンシパルと、プリンシパルがアクセスを試みるリソースの両方に付けられるタグです。

ABAC はプリンシパルにアタッチされた標準の IAM 権限ポリシーを使用します。ポリシーは、ポリシーステートメントの `Condition` の項目を使用して、リクエスト元のプリンシパルに添付されたタグと対象のリソースに添付されたタグの両方がポリシーの要件と一致する場合にのみアクセスを許可します。

たとえば、会社の生産アプリケーションをサポートするすべての AWS リソースに `"Environment" = "Production"` タグを付けることができます。本番環境へのアクセスを許可されたプリンシパルのみがリソースを参照できるようにするには、そのタグをフィルターとして使用する Resource Explorer ビューを作成します。次に、ビューへのアクセスを適切なプリンシパルのみに制限するには、以下の項目例のような条件を持つポリシーを使用してアクセス許可を付与します。

```
{
    "Effect": "Allow",
    "Action": [ "service:Action1", "service:Action2" ],
    "Resource": "arn:aws:arn-of-a-resource",
    "Condition": { "StringEquals": {"aws:ResourceTag/Environment":
 "${aws:PrincipalTag/Environment}"} }
}
```

前の例の `Condition` では、リクエストを行うプリンシパルにアタッチされた `Environment` タグが、リクエストで指定されたリソースに添付された `Environment` タグと一致する場合にのみリク

エストを許可するよう指定しています。この 2 つのタグが完全に一致しない場合、またはどちらかのタグが欠落している場合、Resource Explorer はリクエストを拒否します。

> **Important**
>
> ABAC を正しく使用してリソースへのアクセスを保護するには、まずプリンシパルとリソースに添付されているタグを追加または変更する機能へのアクセスを制限する必要があります。ユーザーが AWS プリンシパルまたはリソースに添付されているタグを追加または変更できる場合、そのユーザーはそれらのタグによって制御される権限に影響を与えることができます。安全な ABAC 環境では、承認されたセキュリティ管理者のみがプリンシパルに添付されたタグを追加または変更する権限を持ち、リソースに添付されたタグを追加または変更できるのはセキュリティ管理者とリソース所有者だけです。

ABAC 戦略の正しい実装方法の詳細については、「IAM ユーザーガイド」の以下のトピックを参照してください。

- IAM チュートリアル: タグに基づいて AWS リソースへのアクセス許可を定義する
- タグを使用した AWS リソースへのアクセスの制御

必要な ABAC インフラストラクチャが整ったら、[タグの使用開始] を使用して、どのユーザーにアカウント内の Resource Explorer ビューを使用した検索を許可するかを制御できます。この原則を説明するポリシー例については、以下のアクセス許可ポリシー例を参照してください。

- タグに基づいてビューへのアクセスを許可する
- タグベースのビュー作成のためのアクセス許可を付与する

AWS リージョン のデフォルトビューを設定する

AWS Resource Explorer では、一つの AWS リージョン に多数のビューを定義できます。各ビューは異なる検索要件に対応します。各リージョンにつき、1 つのビューをそのリージョンのデフォルトビューとして設定することをお勧めします。

Resource Explorer は、ユーザーが検索を実行するたびにデフォルトビューを使用し、どのビューを使用するかは明示的に指定しません。各 AWS Management Console ページの上部にある統合検索バーも、アグリゲーターインデックスを含むリージョンのデフォルトビューを自動的に使用して、ユーザーの検索クエリに一致するリソースを検索します。

そのリージョンのデフォルトビューとして選択できるのは、そのリージョンに存在するビューだけです。使用したいビューが別のリージョンにある場合は、まず、そのビューをデフォルトビューにしたいリージョンにそのビューのコピーを作成する必要があります。

> **Tip**
> 一度にビューをコピーするオペレーションは存在しません。まずターゲットリージョンで新しいビューを作成し、既存のビューから新しいビューに設定をコピーする必要があります。

AWS Management Console または AWS CLI のコマンドを使用するか、AWS SDK 内の同等の API オペレーションを実行することで、特定のビューをそのリージョンのデフォルトビューに指定できます。

AWS Management Console

デフォルトビューを設定するには

1. Resource Explorer の [ビュー]ページ で、リージョンのデフォルトにするビューの横にあるオプションボタンを選択します。
2. [アクション] を選択し、次に [デフォルトに設定] を選択します。

AWS CLI

デフォルトビューを設定するには

次のコマンドを実行して、指定されたビューをリージョンのデフォルトビューに設定します。次の例では、us-east-1リージョンで実行されるすべての検索について指定されたビューがデフォルトになるように設定します。そのビューは、コマンドを実行するリージョン内に存在している必要があります。

```
$ aws resource-explorer-2 associate-default-view \
    --region us-east-1 \
    --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/
MyViewName/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111
{
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/
MyViewName/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111"
}
```

ビューへのタグの追加

ビューにタグを追加することにより、ビューを分類できます。タグは、キー名の文字列とそれに関連するオプションの値文字列の形式をとる、顧客提供のメタデータです。AWS リソースへのタグ付けの詳細については、「Amazon Web Services 全般のリファレンス」の「AWS リソースのタグ付け」を参照してください。

ビューにタグを追加する

AWS Management Console または AWS CLI のコマンドを使用するか、AWS SDK 内の同等の API オペレーションを実行すると、Resource Explorer ビューにタグを追加できます。

AWS Management Console

ビューにタグを追加するには

1. Resource Explorer の [ビュー] ページを開き、タグ付けするビューの名前を選択して [詳細] ページを表示します。
2. [タグ] の項目で、[タグの管理] を選択します。
3. タグを追加するには、[タグの追加] を選択してタグキー名およびオプション値を入力します。

   > **Note**
   >
   > タグの横にある X を選択して、タグを削除することもできます。

   一つのリソースに最大 50 個のユーザー定義タグをアタッチできます。AWS によって自動的に作成および管理されるタグは、このタグ数の限度内にはカウントされません。
4. タグの変更が完了したら、[変更を保存] を選択します。

AWS CLI

ビューにタグを追加するには

ビューにタグを追加するには、次のコマンドを実行します。次の例では、キー名 `environment` と値 `production` を含むタグを指定したビューに追加します。

```
$ aws resource-explorer-2 tag-resource \
```

```
    --resource-id arn:aws:resource-explorer-2:us-east-1:123456789012:view/
MyViewName/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111 \
    --tags environment=production
```

上記のコマンドは成功時には何も出力を生成しません。

> **Note**
>
> ビューから既存のタグを削除するには、`untag-resource` のコマンドを使用します。

タグによるアクセス許可の制御

タグ付けの主な目的の 1 つは、属性ベースのアクセス制御 (ABAC) 戦略をサポートすることです。ABAC は、リソースにタグを付けることにより権限管理をシンプルにします。また、特定の方法でタグ付けされたリソースに対する権限をユーザーに付与することができます。

例えば、次のシナリオが考えられます。`ViewA` という名前のビューには、タグ `environment=prod` (キー名 = 値) を添付します。別の `ViewB` は `environment=beta` とタグ付けされているかもしれません。それぞれのロールやユーザーがアクセスできる環境に基づいて、各ロールとユーザーに同じタグと値をタグ付けします。

また、AWS Identity and Access Management (IAM) アクセス許可ポリシーを IAM ロール、グループ、ユーザーに割り当てることができます。このポリシーは、検索リクエストを行うロールまたはユーザーに、ビューに添付された `environment` タグと同じ値の `environment` タグがある場合にのみ、ビューにアクセスして検索する権限を付与します。

この方法の利点はダイナミックな管理が可能な点であり、誰がどのリソースにアクセスできるかをリスト管理する必要がない点です。ただし、すべてのリソース (ビュー) とプリンシパル (IAM ロールおよびユーザー) に正しくタグ付けすることが重要です。そうすれば、ポリシーを変更しなくても権限が自動的に更新されます。

ABAC ポリシー内のタグを参照する

ビューにタグを付けたら、それらのタグを使用してそのビューへのアクセスをダイナミックに制御できます。以下のポリシー例では、IAM プリンシパルとビューの両方にタグキー `environment` と何らかの値がタグ付けされていることを前提としています。それが完了したら、以下のポリシー例をプリンシパルにアタッチできます。これで、各ロールとユーザーは、プリンシパルに添付された `environment` タグと完全に一致する `environment` タグ値がタグ付けされた任意のビューを使用して `Search` を実行できます。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "resource-explorer-2:GetView",
                "resource-explorer-2:Search"
            ],
            "Resource": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aws:ResourceTag/environment": "${aws:PrincipalTag/environment}"
                }
            }
        }
    ]
}
```

プリンシパルとビューの両方に `environment` タグがあるが値が一致しない場合、またはどちらかに `environment` タグがない場合、Resource Explorer は検索リクエストを拒否します。

ABAC を使用してリソースへのアクセスを安全に許可する方法について詳しくは、「AWS の ABAC とは」を参照してください。

Resource Explorer ビューの共有

のビューは、 AWS Resource Explorer 主にリソースベースのポリシーを使用してアクセスを許可します。Amazon S3 バケットポリシーと同様に、これらのポリシーはビューにアタッチされて、ビューを使用できるユーザーを指定します。これは (IAM) アイデンティティベースのポリシーとは AWS Identity and Access Management 対照的です。IAM ID ベースのポリシーは、ロール、グループ、またはユーザーに割り当てられ、そのロール、グループ、またはユーザーがアクセスできるアクションとリソースを指定します。以下のように、Resource Explorer ビューではどちらのタイプのポリシーも使用できます。

- リソースを所有する管理アカウントまたは委任管理者アカウント内では、いずれかのポリシータイプを使用してアクセスを許可します。ただし、そのプリンシパルのビューへのアクセスを明示的に拒否するポリシーが他にない場合に限ります。
- どのアカウントでも、両方のポリシータイプを使用する必要があります。共有アカウントのビューにアタッチされたリソースベースのポリシーは、別の消費アカウントとの共有を有効にします。た

だし、そのポリシーでは、消費アカウント内の個々のユーザーやロールにはアクセス許可が付与されません。消費側アカウントの管理者が、消費側アカウント内の必要なロールとユーザーに ID ベースのポリシーを割り当てる必要があります。このポリシーは、ビューの Amazon リソース名 (ARN) へのアクセスを許可します。

ビューを他のアカウントと共有するには、 AWS Resource Access Manager （ AWS RAM) を使用する必要があります。 はリソースベースのポリシーの複雑さ AWS RAM に対処します。共有する前に、次のタスクを実行する必要があります。

- マルチアカウント検索 を有効にします。
- リソースベースのポリシー、またはビューの共有と共有解除に使用するIAMアイデンティティベースのポリシーに`resource-explorer-2:GetResourcePolicy`、、`resource-explorer-2:PutResourcePolicy`および アクセス`resource-explorer-2:DeleteResourcePolicy`許可が含まれていることを確認します。

ビューを共有するには、組織の管理アカウントまたは委任管理者アカウントにサインインする必要があります。リソースを共有するアカウントまたは ID を指定します。 は、共有するプリンシパルのタイプに基づいて、以下のセクションで説明されているようなポリシーを Resource Explorer views. AWS RAM uses で AWS RAM 完全にサポートします。リソースを共有する方法については、「AWS Resource Access Manager ユーザーガイド」の「AWS リソースの共有」を参照してください。

管理者および委任管理者は、組織範囲のビュー、組織単位 (OU) 範囲のビュー、アカウントレベル範囲のビューの 3 種類のビューを作成して共有できます。組織、OUs、または アカウントと共有できます。アカウントが組織に参加または組織を離れると、 は共有ビュー AWS RAM を自動的に付与または取り消します。

AWS アカウントとビューを共有するための権限ポリシー

次のポリシー例は、2 つの異なる のプリンシパルがビューを使用できるようにする方法を示しています AWS アカウント。

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

```
                "AWS": [ "111122223333", "444455556666" ]
            },
            "Action": [
                "resource-explorer-2:Search",
                "resource-explorer-2:GetView",
            ],
            "Resource": "arn:aws:resource-explorer-2:us-
east-1:123456789012:view/policy-name/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
            "Condition": {"StringEquals": {"aws:PrincipalOrgID": "o-123456789012"},
                "StringNotEquals": {"aws:PrincipalAccount": "123456789012"}
            }
        }
    ]
    }"
}
```

指定した各アカウントの管理者は、ID ベースのアクセス許可ポリシーをロール、グループ、およびユーザーにアタッチすることにより、どのロールやユーザーがビューにアクセスできるかを指定する必要があります。111122223333 または 444455556666 アカウントの管理者は、以下のサンプルポリシーを作成できます。次に、元のアカウントから共有されているビューを使用して検索できるアカウントのロール、グループ、ユーザーにポリシーを割り当てることができます。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "resource-explorer-2:Search",
                "resource-explorer-2:GetView",
            "Resource": "arn:aws:resource-explorer-2:us-
east-1:123456789012:view/policy-name/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111"
        }
    ]
}
```

これらのIAMアイデンティティベースのポリシーは、属性ベースのアクセスコントロール (ABAC) セキュリティ戦略の一部として使用できます。このパラダイムでは、すべてのリソースとすべての ID にタグが付けられていることを確認してください。次に、アクセスを許可するのに ID とリソース間でどのタグキーと値が一致する必要があるかをポリシー内で指定します。アカウント内のビューにタグを付ける方法については、「ビューへのタグの追加」を参照してください。属性ベースのアクセス

コントロールの詳細については、IAM「 ユーザーガイド」の「 ABACとは AWS」および「 タグ を使用した AWS リソースへのアクセスの制御」を参照してください。

Resource Explorer でのビューの削除

不要になった AWS Resource Explorer ビューは削除できます。AWS Management Console または AWS CLI コマンドを使用するか、AWS SDK 内の同等の API オペレーションを実行することでビューを削除できます。

> **Note**
>
> 現在 AWS リージョン のデフォルトに指定されているビューは削除できません。ビューを削除するには、そのビューのデフォルト設定を解除する必要があります。そのためには、そのリージョンで DisassociateDefaultView の API オペレーションを実行します。

最小アクセス許可

この手順を実行するには、次のアクセス許可が必要です。

- アクション: `resource-explorer-2:DeleteView`

  リソース: 削除するビューの ARN

AWS Management Console

ビューを削除するには

1. Resource Explorer コンソールの[ビュー] ページで、削除するビューの横にあるオプションボタンを選択します。
2. [アクション] を選択してから、[削除] を選択します。
3. 確認ダイアログボックスで、ビュー名を入力し、[削除] を選択します。

AWS CLI

ビューを削除するには

次のコマンドを実行して、指定した Amazon リソースネーム (ARN) のビューを削除します。

```
$ aws resource-explorer-2 delete-view \
    --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/
MyViewName/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111
{
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/
MyViewName/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111"
}
```

AWS Resource Explorer を用いたリソースの検索

AWS アカウント で AWS Resource Explorer を有効にする主な目的は、ユーザーがアカウント内のリソースを検索できるようにすることです。AWS Management Console または AWS Command Line Interface (AWS CLI) を使用して、Resource Explorer でリソースを検索することができます。

Resource Explorer による検索の主な特長は以下のとおりです。

- 検索には必ず特定のビューを使用する必要があります。

  ビューは、Resource Explorer 側でどのユーザーにどのリソースの閲覧を許可するかを管理するのに用いる手段です。Resource Explorer の検索操作でビューを使用する場合、ユーザーは指定されたビューに対する `resource-explorer-2:Search` 操作の `Allow` を持っている必要があります。この権限は、リクエストを行うプリンシパルにアタッチされている ID ベースのアクセス権限ポリシーにより付与されます。

  ビューには、検索結果にどのリソースを含めることができるかを制限するフィルターが含まれます。フィルターを使用するさまざまなビューを作成し、さまざまなプリンシパルにさまざまなビューへのアクセス権限を付与することで、各ユーザーグループが自分に関連するリソースのみを閲覧できる環境を構築できます。

  ビューの詳細については、検索アクセス許可を提供するための Resource Explorer ビューの管理を参照してください。

- Resource Explorer は、非同期のバックグラウンドプロセスを実行してインデックスを維持管理しています。

  Resource Explorer のインデックス処理プロセスが、新しく作成または変更されたリソースを検出してローカルインデックスに追加するまでに、しばらく時間がかかることがあります。Resource Explorer がローカルインデックスの変更をアグリゲーターインデックスにリプリケートするには、さらに時間がかかる場合があります。

  削除したリソースについても同様です。リソースを削除してから、その削除がインデックス処理プロセスによって検出され、そのリソースの情報がローカルインデックスから削除されるまでには、しばらく時間がかかることがあります。Resource Explorer がその削除をローカルインデックスからアカウントのアグリゲーターインデックスにリプリケートするには、さらに時間がかかります。

  リソースへの追加、変更、削除を行うと、リソースエクスプローラーを有効にしたすべてのリージョンの検索結果にその変更が表示されるまでに最大 36 時間かかることがあります。

- Resource Explorer での検索は、特定の AWS リージョン 内で行われます。

  Resource Explorer がオンになっている各リージョンには、そのリージョンに格納されているリソースのみのインデックスが含まれます。各ビューはそれぞれのリージョンに関連付けられており、そのリージョンのインデックスにあるリソースのみを返すことができます。ただし、アグリゲーターインデックスは例外です。アグリゲーターインデックスは、アカウント内のすべてのリージョンをまたぐ検索をサポートするために、すべてのローカルインデックスのリプリケートされたコピーを受け取ります。

- クロスリージョン検索には、アカウントのアグリゲーターインデックスが必要です。

  ユーザーがすべての AWS リージョン のリソースを検索できるようにするには、管理者はアカウントのアグリゲーターインデックスを格納するリージョンを 1 つ指定する必要があります。すべてのローカルインデックスのコピーは、自動的にアグリゲーターインデックスにリプリケートされます。

  そのため、アグリゲーターインデックスリージョンのビューのみが、アカウント内のすべての AWS リージョン のリソースを含む結果を返すことができます。

- クエリは、任意の数の自由形式のテキストキーワードとフィルターで構成されます。

  自由形式のキーワードは、論理演算子 **OR** を使用してクエリ内で組み合わされます。Resource Explorer で定義されたフィルター名を使用するフィルターは、論理演算子**AND**を使用してクエリ内で結合されます。次の例を考えます。

  ```
  test instance service:EC2 region:us-west-2
  ```

  これは Resource Explorer によって次のように評価されます。

  ```
  test OR instance AND service:EC2 AND region:us-west-2
  ```

  このクエリでは、一致するリソースは米国西部 (オレゴン州) リージョン内に存在する Amazon EC2 リソースである必要があり、指定されたキーワード (test、instance) の少なくとも 1 つが、名前、記述、タグなど、リソースにアタッチされている何らかの要素に含まれている必要があります。

> **Note**
>
> 暗示的 AND であるため、リソースに関連付けられる値が 1 つしかない属性には 1 つのフィルタしか使用できません。たとえば、1 つのリソースは 1 つの AWS リージョン にのみ属することができます。そのため、以下のクエリは結果を返しません。
>
> ```
> region:us-east-1 region:us-west-1
> ```
>
> この制限は、`tag:`、`tag.key:`、`tag.value:`など、同時に複数の値を持つことができる属性のフィルターには適用されません。

- 1 回の検索で返すことのできる結果は最初の 1,000 件のみです。

  この要件は、すべてのリソースに一致する空のクエリ文字列による検索にも適用されます。空のクエリ文字列によって返される 1,000 件を超えるリソースを表示するには、追加クエリを使用して一致する結果を確認したいリソースに限定し、一致件数を 1,000 件未満に制限する必要があります。

- 実行できる検索操作件数をアカウントごとに制限するクォータが設定されています。

  クォータは、1 秒あたりに実行できるクエリの件数と、1 か月あたりに実行できるクエリの件数の両方を制限します。具体的なクォータ限度については、Resource Explorer のクォータ を参照してください。

AWS Management Console

Resource Explorer を使用してリソースを検索するには

1. [リソース検索]ページで、まず使用したいビューを選択します。自分がアクセス権限を持っているビューからのみ選択できます。
2. [クエリ] に、表示したいリソースを識別する検索用語とフィルターを入力します。利用できるすべての構文オプションについては、Resource Explorer の検索クエリ構文リファレンスを参照してください。
3. [Enter] を押して選択内容を送信します。

   Resource Explorer には、ビューで定義されている `Filter` と指定した [クエリ] の両方に一致するすべての結果が表示されます。結果は関連度順に並べられ、クエリ用語に一致する

ワードが多いリソースはリストの上位に表示され、一致する用語が少ないリソースはリストの下位の方に表示されます。

4. 特定のリソースの識別子を選択するとそのリソースタイプのネイティブコンソールに移動するので、そこでそのサービスがサポートするあらゆる方法でリソースを操作できます。

AWS CLI

Resource Explorer を使用してリソースを検索するには

以下のコマンドを実行して、指定したビューを使用してリソースを検索します。そのビューは、操作を実行しているリージョン内に存在している必要があります。次の例では、米国東部 (オハイオ州) (us-east-2) リージョン内に存在し、`env=production` とタグ付けされている Amazon EC2 インスタンスを検索します。`query-string` パラメータに使用できるすべての構文オプションについては、Resource Explorer の検索クエリ構文リファレンス を参照してください。

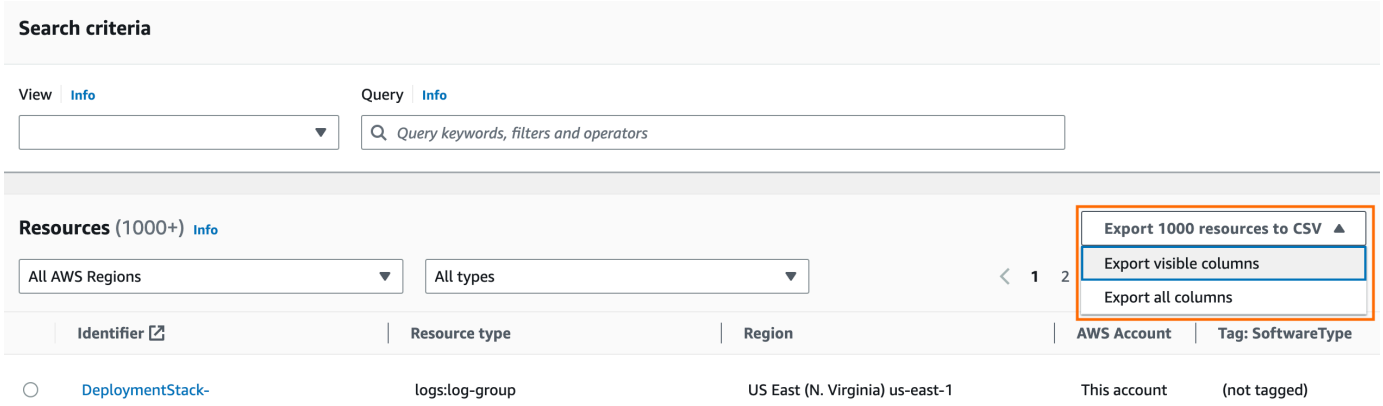
```
$ aws resource-explorer-2 search \
    --region us-east-1 \
    --query-string "resourcetype:AWS::EC2::Instance tag:env=production"
    --view-arn arn:aws:resource-explorer-2:us-east-2:123456789012:view/My-Resources-View/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111
```

検索結果を CSV ファイルにエクスポートする

[リソース検索] クエリの結果をカンマ区切り値 (CSV) ファイルにエクスポートすることができます。CSV ファイルには、そのリソースの識別子、リソースタイプ、リージョン、AWS アカウント、タグの合計数、および収集された各一意のタグキーごとの列が含まれます。CSV ファイルは、組織内での AWS リソースの構成設定、またはリソース間でのタグ付けの重複または不整合が存在する箇所の特定に役立ちます。

1. [リソース検索] クエリの結果画面で、[リソースを CSV にエクスポートする] を選択します。

   現在表示されている列のみの結果をエクスポートするか、あるいは利用可能なすべての列をエクスポートするかを選択できます。



2. ブラウザでプロンプトが表示されたら、CSV ファイルを 開くか、あるいは便利な場所に保存するかを選択します。

Resource Explorer で検索できるリソースタイプ

Resource Explorer は、多数の AWS サービスでリソースタイプをサポートしています。

トピック

- サポートされているサービスとリソースタイプ
- サポートされているリソースタイプのリストにプログラムからアクセスする
- 他のリソースタイプとして表示されるリソースタイプ

一部のリソースタイプは、別のリソースタイプと共通の形式を共有する Amazon リソース名 (ARN) 文字列によって識別されます。このような場合、Resource Explorer はそれらのリソースを他のリソースタイプとして報告することがあります。この問題の影響を受けるリソースタイプの一覧については、「他のリソースタイプとして表示されるリソースタイプ」を参照してください。

現時点では、ロールやユーザーなどの AWS Identity and Access Management （IAM) リソースにアタッチされたタグを検索に使用することはできません。

一部のリソースへのアクセスが暗号化されている場合は、Resource Explorer でそれらのリソースが検出されず、検索結果にも表示されません。

AWS Resource Explorerでの検索がサポートされているリソースタイプのリストを以下の表に示します。

> **Note**
>
> 2024 年 7 月 9 日をもって、Resource Explorer は次のリソースタイプをサポートしなくなりました。
>
> - Amazon Elastic Container Service — `ecs:task`
> - AWS Systems Manager — `ssm:automation-execution`
> - AWS Systems Manager — `ssm:patchbaseline`
>
> これらのリソースタイプは独自のサービスで引き続き使用できますが、Resource Explorer ではインデックス作成または検索できなくなりました。

サポートされているサービスとリソースタイプ

サポートされる AWS のサービス

- Amazon API Gateway
- AWS App Runner
- Amazon AppStream 2.0
- AWS AppSync
- Amazon Athena
- AWS Backup
- AWS Batch
- AWS CloudFormation
- Amazon CloudFront
- AWS CloudTrail
- Amazon CloudWatch
- Amazon CloudWatch Evidently
- Amazon CloudWatch ログ
- AWS CodeArtifact
- AWS CodeBuild
- AWS CodeCommit
- Amazon CodeGuru Profiler
- AWS CodePipeline
- AWS CodeConnections
- Amazon Cognito
- Amazon Connect
- Amazon Connect Wisdom
- Amazon Detective
- Amazon DynamoDB
- EC2 Image Builder
- Amazon ECR Public
- AWS Elastic Beanstalk

- Amazon ElastiCache
- Amazon Elastic Compute Cloud (Amazon EC2)
- Amazon Elastic Container Registry
- Amazon Elastic Container Service
- Amazon Elastic File System
- Elastic Load Balancing
- AWS Elemental MediaPackage
- AWS Elemental MediaTailor
- Amazon EMR Serverless
- Amazon EventBridge
- AWS Fault Injection Service
- Amazon Forecast
- Amazon Fraud Detector
- Amazon GameLift
- AWS Global Accelerator
- AWS Glue
- AWS Glue DataBrew
- AWS Identity and Access Management
- Amazon Interactive Video Service
- AWS IoT
- AWS IoT Analytics
- AWS IoT Events
- AWS IoT Greengrass Version 1
- AWS IoT SiteWise
- AWS IoT TwinMaker
- AWS Key Management Service
- Amazon Kinesis
- Amazon Data Firehose
- Amazon Kinesis Video Streams
- AWS Lambda

- Amazon Lex
- Amazon Location Service
- Amazon Lookout for Metrics
- Amazon Lookout for Vision
- Amazon Managed Service for Apache Flink
- Amazon Managed Service for Prometheus
- Amazon Managed Service for Prometheus
- Amazon Managed Streaming for Apache Kafka
- AWS Migration Hub Refactor Spaces
- AWS Network Firewall
- AWS Network Manager
- Amazon OpenSearch サービス
- AWS Panorama
- Amazon Personalize
- AWS Private Certificate Authority
- Amazon QLDB
- Amazon Redshift
- Amazon Rekognition
- Amazon Relational Database Service (Amazon RDS)
- AWS Resilience Hub
- AWS Resource Groups
- AWS Resource Explorer
- Amazon Route 53
- Amazon Route 53 Recovery 準備状況
- Amazon Route 53 Resolver
- Amazon SageMaker
- AWS Secrets Manager
- AWS Service Catalog
- Amazon Simple Notification Service
- Amazon Simple Queue Service

- Amazon Simple Storage Service (Amazon S3)
- AWS Step Functions
- AWS Systems Manager
- AWS Verified Access
- AWS Wavelength

Amazon API Gateway

- `apigateway:restapis`

AWS App Runner

- `apprunner:vpcconnector`

Amazon AppStream 2.0

- `appstream:appblock`
- `appstream:application`
- `appstream:fleet`
- `appstream:stack`

AWS AppSync

- `appsync:apis`

Amazon Athena

- `athena:datacatalog`
- `athena:workgroup`

AWS Backup

- `backup:backupplan`

AWS Batch

- `batch:computeenvironment`
- `batch:jobqueue`
- `batch:schedulingpolicy`

AWS CloudFormation

- `cloudformation:stack`
- `cloudformation:stackset`

Amazon CloudFront

- `cloudfront:cache-policy`
- `cloudfront:distribution`
- `cloudfront:function`
- `cloudfront:fieldlevelencryptionconfig`
- `cloudfront:fieldlevelencryptionprofile`
- `cloudfront:origin-access-identity`
- `cloudfront:originaccesscontrol`
- `cloudfront:origin-request-policy`
- `cloudfront:realtime-log-config`
- `cloudfront:response-headers-policy`

AWS CloudTrail

- `cloudtrail:trail`

Amazon CloudWatch

- `cloudwatch:alarm`
- `cloudwatch:dashboard`

- `cloudwatch:insight-rule`
- `cloudwatch:metric-stream`
- `evidently:project`

Amazon CloudWatch Evidently

- `evidently:project/experiment`
- `evidently:project/feature`
- `evidently:project/launch`

Amazon CloudWatch ログ

- `logs:destination`
- `logs:log-group`

AWS CodeArtifact

- `codeartifact:domain`
- `codeartifact:repository`

AWS CodeBuild

- `codebuild:project`

AWS CodeCommit

- `codecommit:repository`

Amazon CodeGuru Profiler

- `codeguru-profiler:profilingGroup`

AWS CodePipeline

- `codepipeline:pipeline`

AWS CodeConnections

- `codestarconnections:connect`

Amazon Cognito

- `cognito:identitypool`
- `cognito:userpool`

Amazon Connect

- `appintegrations:eventintegration`

Amazon Connect Wisdom

- `wisdom:assistant`
- `wisdom:association`
- `wisdom:knowledge-base`

Amazon Detective

- `detective:graph`

Amazon DynamoDB

- `dynamodb:table`

EC2 Image Builder

- `imagebuilder:component`
- `imagebuilder:containerrecipe`
- `imagebuilder:distributionconfiguration`
- `imagebuilder:image`
- `imagebuilder:imagepipeline`
- `imagebuilder:imagerecipe`
- `imagebuilder:infrastructureconfiguration`

Amazon ECR Public

- `ecrpublic:repository`

AWS Elastic Beanstalk

- `elasticbeanstalk:application`
- `elasticbeanstalk:applicationversion`
- `elasticbeanstalk:configurationtemplate`
- `elasticbeanstalk:environment`

Amazon ElastiCache

- `elasticache:cluster`
- `elasticache:globalreplicationgroup`
- `elasticache:parametergroup`
- `elasticache:replicationgroup`
- `elasticache:reserved-instance`
- `elasticache:snapshot`
- `elasticache:subnetgroup`
- `elasticache:user`
- `elasticache:usergroup`

Amazon Elastic Compute Cloud (Amazon EC2)

- `ec2:capacity-reservation`
- `ec2:capacity-reservation-fleet`
- `ec2:client-vpn-endpoint`
- `ec2:customer-gateway`
- `ec2:dedicated-host`
- `ec2:dhcp-options`
- `ec2:egress-only-internet-gateway`
- `ec2:elastic-gpu`
- `ec2:elastic-ip`
- `ec2:fleet`
- `ec2:fpga-image`
- `ec2:host-reservation`
- `ec2:image`
- `ec2:instance`
- `ec2:instance-event-window`
- `ec2:internet-gateway`
- `ec2:ipam`
- `ec2:ipam-pool`
- `ec2:ipam-scope`
- `ec2:ipv4pool-ec2`
- `ec2:key-pair`
- `ec2:launch-template`
- `ec2:natgateway`
- `ec2:network-acl`
- `ec2:network-insights-access-scope`
- `ec2:network-insights-access-scope-analysis`
- `ec2:network-insights-analysis`
- `ec2:network-insights-path`

- ec2:network-interface
- ec2:placement-group
- ec2:prefix-list
- ec2:reserved-instances
- ec2:route-table
- ec2:security-group
- ec2:security-group-rule
- ec2:snapshot
- ec2:spot-fleet-request
- ec2:spot-instances-request
- ec2:subnet
- ec2:subnet-cidr-reservation
- ec2:traffic-mirror-filter
- ec2:traffic-mirror-filter-rule
- ec2:traffic-mirror-session
- ec2:traffic-mirror-target
- ec2:transit-gateway
- ec2:transit-gateway-attachment
- ec2:transit-gateway-connect-peer
- ec2:transit-gateway-multicast-domain
- ec2:transit-gateway-policy-table
- ec2:transit-gateway-route-table
- ec2:transitgatewayroutetableannouncement
- ec2:volume
- ec2:vpc
- ec2:vpc-endpoint
- ec2:vpc-flow-log
- ec2:vpc-peering-connection
- ec2:vpn-connection
- ec2:vpn-gateway

Amazon Elastic Container Registry

- `ecr:repository`

Amazon Elastic Container Service

- `ecs:cluster`
- `ecs:container-instance`
- `ecs:service`
- `ecs:task-definition`
- `ecs:task-set`

Amazon Elastic File System

- `efs:filesystem`
- `efs:accesspoint`

Elastic Load Balancing

- `elasticloadbalancing:listener`
- `elasticloadbalancing:listener-rule`
- `elasticloadbalancing:listener-rule/app`
- `elasticloadbalancing:listener/app`
- `elasticloadbalancing:listener/net`
- `elasticloadbalancing:loadbalancer`
- `elasticloadbalancing:loadbalancer/app`
- `elasticloadbalancing:loadbalancer/net`
- `elasticloadbalancing:targetgroup`

AWS Elemental MediaPackage

- `mediapackage:channel`

- `mediapackage:originendpoint`
- `mediapackage-vod:packaging-configurations`
- `mediapackage-vod:packaging-groups`

AWS Elemental MediaTailor

- `mediatailor:playbackConfiguration`

Amazon EMR Serverless

- `emr-serverless:applications`

Amazon EventBridge

- `events:event-bus`
- `events:rule`

AWS Fault Injection Service

- `fis:experimenttemplate`

Amazon Forecast

- `forcast:dataset`
- `forcast:dataset-group`

Amazon Fraud Detector

- `frauddetector:detector`
- `frauddetector:entity-type`
- `frauddetector:event-type`
- `frauddetector:label`
- `frauddetector:outcome`

- `frauddetector:variable`

Amazon GameLift

- `gamelift:alias`

AWS Global Accelerator

- `globalaccelerator:accelerator`
- `globalaccelerator:accelerator/listener`
- `globalaccelerator:accelerator/listener/endpoint-group`

AWS Glue

- `glue:database`
- `glue:job`
- `glue:table`
- `glue:trigger`

AWS Glue DataBrew

- `databrew:dataset`
- `databrew:recipe`
- `databrew:ruleset`

AWS Identity and Access Management

- `iam:group`
- `iam:instance-profile`
- `iam:oidc-provider`
- `iam:policy`
- `iam:role`

- `iam:saml-provider`
- `iam:server-certificate`
- `iam:user`
- `iam:virtualmfadevice`

Amazon Interactive Video Service

- `ivs:channel`
- `ivs:streamkey`

AWS IoT

- `iot:authorizer`
- `iot:jobtemplate`
- `iot:mitigationaction`
- `iot:policy`
- `iot:provisioningtemplate`
- `iot:rolealias`
- `iot:securityprofile`
- `iot:thing`
- `iot:topicrule`

AWS IoT Analytics

- `iotanalytics:channel`
- `iotanalytics:dataset`
- `iotanalytics:datastore`
- `iotanalytics:pipeline`

AWS IoT Events

- `iotevents:alarmModel`

- `iotevents:detectorModel`
- `iotevents:input`

AWS IoT Greengrass Version 1

- `greengrass:components`
- `greengrass:groups`

AWS IoT SiteWise

- `iotsitewise:asset`
- `iotsitewise:assetmodel`
- `iotsitewise:gateway`

AWS IoT TwinMaker

- `iottwinmaker:workspace`
- `iottwinmaker:workspace/component-type`
- `iottwinmaker:workspace/entity`

AWS Key Management Service

- `kms:key`

Amazon Kinesis

- `kinesis:stream`

Amazon Data Firehose

- `kinesisfirehose:deliverystream`

Amazon Kinesis Video Streams

- `kinesisvideo:stream`

AWS Lambda

- `lambda:code-signing-config`
- `lambda:event-source-mapping`
- `lambda:function`

Amazon Lex

- `lex:bot`

Amazon Location Service

- `geo:place-index`
- `geo:tracker`

Amazon Lookout for Metrics

- `lookoutmetrics:Alert`

Amazon Lookout for Vision

- `lookoutvision:project`

Amazon Managed Service for Apache Flink

- `kinesisanalytics:application`

Amazon Managed Service for Prometheus

- `aps:rulegroupsnamespace`
- `aps:workspace`

Amazon Managed Service for Prometheus

- `memorydb:cluster`
- `memorydb:parametergroup`
- `memorydb:user`

Amazon Managed Streaming for Apache Kafka

- `kafka:cluster`
- `kafka:configuration`

AWS Migration Hub Refactor Spaces

- `refactor-spaces:environment`
- `refactor-spaces:environment/application`
- `refactor-spaces:environment/application/route`
- `refactor-spaces:environment/application/service`

AWS Network Firewall

- `network-firewall:firewall-policy`

AWS Network Manager

- `networkmanager:core-network`
- `networkmanager:device`
- `networkmanager:global-network`
- `networkmanager:link`

Amazon OpenSearch サービス

- `es:domain`

AWS Panorama

- `panorama:package`

Amazon Personalize

- `personalize:dataset`
- `personalize:dataset-group`
- `personalize:schema`

AWS Private Certificate Authority

- `acmpca:certificateauthority`

Amazon QLDB

- `qldb:ledger`
- `qldb:stream`

Amazon Redshift

- `redshift:cluster`
- `redshift:eventsubscription`
- `redshift:parametergroup`
- `redshift:snapshot`
- `redshift:snapshotcopygrant`
- `redshift:snapshotschedule`
- `redshift:subnetgroup`
- `redshift:usagelimit`

Amazon Rekognition

- `rekognition:project`

Amazon Relational Database Service (Amazon RDS)

- `rds:auto-backup`
- `rds:cev`
- `rds:cluster`
- `rds:cluster-endpoint`
- `rds:cluster-pg`
- `rds:cluster-snapshot`
- `rds:db`
- `rds:db-proxy`
- `rds:db-proxy-endpoint`
- `rds:deployment`
- `rds:es`
- `rds:global-cluster`
- `rds:og`
- `rds:pg`
- `rds:ri`
- `rds:secgrp`
- `rds:snapshot`
- `rds:subgrp`

AWS Resilience Hub

- `resiliencehub:resiliencypolicy`

AWS Resource Groups

- `resourcegroups:group`

AWS Resource Explorer

- `resource-explorer-2:index`
- `resource-explorer-2:view`

Amazon Route 53

- `route53:healthcheck`
- `route53:hostedzone`

Amazon Route 53 Recovery 準備状況

- `route53-recover-readiness:recovery-group`
- `route53-recover-readiness:resource-set`

Amazon Route 53 Resolver

- `route53resolver:firewalldomainlist`
- `route53resolver:firewallrulegroup`
- `route53resolver:resolverendpoint`
- `route53resolver:resolverrule`

Amazon SageMaker

- `sagemaker:model`
- `sagemaker:notebookinstance`

AWS Secrets Manager

- `secretsmanager:secret`

AWS Service Catalog

- `servicecatalog:applications`
- `servicecatalog:attribute-groups`

Amazon Simple Notification Service

- `sns:topic`

Amazon Simple Queue Service

- `sqs:queue`

Amazon Simple Storage Service (Amazon S3)

- `s3:accesspoint`
- `s3:bucket`
- `s3:storage-lens`

AWS Step Functions

- `states:statemachine`
- `stepfunctions:activity`

AWS Systems Manager

- `ssm:association`
- `ssm:document`
- `ssm:maintenancewindow`
- `ssm:managed-instance`
- `ssm:parameter`
- `ssm:resourcedatasync`
- `ssm:windowtarget`

- `ssm:windowtask`

AWS Verified Access

- `ec2:verifiedaccessendpoint`
- `ec2:verifiedaccessgroup`
- `ec2:verifiedaccessinstance`
- `ec2:verifiedaccesstrustprovider`

AWS Wavelength

- `ec2:carriergateway`

サポートされているリソースタイプのリストにプログラムからアクセスする

コードからサポートされているリソースタイプのリストにアクセスするには、任意の から ListSupportedResourceTypesオペレーションを呼び出します AWS SDK。

例えば、次の例に示すように (list-supported-resource-types AWS Command Line Interface AWS CLI) コマンドを実行できます。

```
$ aws resource-explorer-2 list-supported-resource-types
{
    "ResourceTypes": [
        {
            "ResourceType": "acm-pca:certificate-authority",
            "Service": "acm-pca"
        },
        {
            "ResourceType": "airflow:environment",
            "Service": "airflow"
        },
        {
            "ResourceType": "amplify:branches",
            "Service": "amplify"
        },
```

```
... truncated for brevity ...
```

他のリソースタイプとして表示されるリソースタイプ

一部のリソースタイプは、別のリソースタイプと共通の形式を共有する Amazon リソース名 (ARN) 文字列によって識別されます。このような場合、Resource Explorer はそのようなリソースを他のリソースタイプとして報告することがあります。これは以下の表のリソースタイプに影響します。

実際のリソースタイプ	報告されるリソースタイプ

`ec2:securitygroupegress`
 	
`ec2:securitygroupingress`
	`ec2:security-group-rule`

`elasticloadbalancingv2:load balancer`	`elasticloadbalancing:loadba lancer`

`docdb:dbcluster`
 	
`neptune:dbcluster`
 	
`rds:dbcluster`
	`rds:cluster`

`docdb:dbclusterparametergroup`
 	
`neptune:dbclusterparametergroup`
 	
`rds:dbclusterparametergroup`
	`rds:cluster-pg`

`docdb:clustersnapshot`
 	
`neptune:dbclustersnapshot`
 	
`rds:clustersnapshot`
	`rds:cluster-snapshot`

`docdb:dbinstance`
 	
`neptune:dbinstance`
 	
`rds:dbinstance`
	`rds:db`

実際のリソースタイプ	報告されるリソースタイプ
docdb:eventsubscription neptune:eventsubscription rds:eventsubscription	rds:es
docdb:globalcluster rds:globalcluster	rds:global-cluster
neptune:dbparametergroup rds:dbparametergroup	rds:pg
docdb:dbsubnetgroup neptune:dbsubnetgroup rds:dbsubnetgroup	rds:subgrp

Resource Explorer の検索クエリ構文リファレンス

AWS Resource Explorer は、 内の個々の AWS リソースを見つけるのに役立ちます AWS アカウント。探しているリソースを正確に見つけられるように、Resource Explorer では、このトピックで説明している構文をサポートする検索クエリ文字列を使用できます。ここで説明した機能の使用例を示すサンプルクエリについては、「Resource Explorer による検索クエリの例」を参照してください。

> **Note**
>
> 現時点では、ロールやユーザーなどの AWS Identity and Access Management（IAM) リソースにアタッチされたタグはインデックス化されません。

Resource Explorer でのクエリの仕組み

検索クエリは常に特定のビューを使用します。明示的に指定しない場合、Resource Explorer は作業 AWS リージョン している のデフォルトとして指定されたビューを使用します。

ビューによって、どんなリソースをクエリできるかが決まります。それぞれに異なるリソースセットを返す、さまざまなビューを作成できます。

例えば、キー `Environment` と値 `Production` のタグが付いたリソースのみを含むビューを作成することができます。あるいは、業務上そのリソースを閲覧する必要があるユーザーのみにそのビューへのアクセスを許可するように選択できます。`Alpha` または `Beta` 環境リソースを含む一つのビューには、それらのリソースを閲覧する必要がある複数の異なるユーザーがアクセスできます。どのビューにどのユーザーがアクセスできるかを制御する方法については、「検索用の Resource Explorer ビューへのアクセス許可の付与」を参照してください。

クエリ文字列の構文

このセクションでは、クエリ構文、フィルター、フィルター演算子の基本について説明します。

基本

最も基本的な `QueryString` は、論理演算子 **`OR`** によって暗示的に結合された自由形式のテキストキーワードのセットです。次の例に示すように、スペースを使用して各キーワードを区切ります。

```
ec2 billing test gamma
```

Resource Explorer は、このキーワードのリストを次の意味で評価します。

```
ec2 OR billing OR test OR gamma
```

Resource Explorer は検索結果を関連度の高い順に並べ替え、一致する検索語の数が多いリソースを優先して表示します。1 つ以上の用語に一致しないリソースも結果からは除外されません。ただし、Resource Explorer はそれらのリソースを関連性が低いと見なし、検索結果の下位の方に表示します。

`QueryString` パラメータに空の文字列を指定すると、クエリは操作に使用されたビューで使用できる最初の 1,000 リソースを返します。クエリによって返すことのできるリソースの最大数は 1,000 です。

> **Note**
>
> AWS は、フリーフォームテキストキーワードを評価するためのマッチングロジックと関連性アルゴリズムを更新して、お客様に最も関連性の高い結果を提供できるようにします。そのため、自由形式のテキストキーワードを使用した同じクエリで返される結果であっても、時間の経過とともにその内容が変化する可能性があります。より確定的な結果が必要な場合は、フィルタを使用することをお勧めします。フィルタの照合ロジックは、時間の経過とともに変化することはありません。

フィルター

フィルターを含めることで、クエリの結果をより厳密に絞り込むことができます。テキストキーワードとは異なり、フィルターは AND演算子を使用してクエリで評価されます。例えば、2 つの自由形式のキーワードと 2 つのフィルターで構成される次のクエリを考えてみます。

```
test instance service:EC2 region:us-west-2
```

このクエリは、次のように評価されます。

```
( test OR instance ) AND service:EC2 AND region:us-west-2
```

フィルターは常にAND論理演算子を使用して評価されます。リソースがフィルターと一致しない場合、そのリソースは結果に含まれません。クエリの結果の例には、Amazon に関連付けられEC2、米国西部 (オレゴン) にあり AWS リージョン 、何らかの方法で少なくとも 1 つのキーワードがアタッチされているリソースが含まれます。

> **Note**
>
> 暗示的 AND であるため、リソースに関連付けられる値が 1 つしかない属性には 1 つのフィルタしか使用できません。例えば、1 つのリソースは 1 つの AWS リージョンにのみ属することができます。そのため、以下のクエリは結果を返しません。
>
> ```
> region:us-east-1 region:us-west-1
> ```
>
> この制限は、`tag:`、`tag.key:`、`tag.value:`など、同時に複数の値を持つことができる属性のフィルターには適用されません。

次の表に、Resource Explorer 検索クエリに使用できるフィルター名の一覧を示します。

フィルター名	説明と例
`accountid:`	リソースを所有 AWS アカウント する 。Resource Explorer の検索結果には、指定したアカウントが所有するリソースのみが含まれます。 `accountid:123456789012`

`application:`	このフィルターでは、`awsApplication` タグキーとリソースグループ値を使用してリソースを検索します。アプリケーション名またはアプリケーションリソースグループ で検索できますARN。
	
	`application:MyApplicationName`
	
	`application:arn:aws:resource-groups: us-east-1 :123456789012:group/MyApplicationName/123456789abced`
	
	`arn:aws:resource-groups: us-east-1 :123456789012:group/MyApplicationName/123456789abced`
	
	Note
	このフィルターを使用するには、そのビューにタグ付けデータへのアクセスが設定されている必要があります。

フィルター名	説明と例
`id:`	Amazon リソース名 (ARN) で表される個々のリソースの識別子。 `id:arn:aws:license-manager: us-east-1 :123456789012:license-configuration:lic-ecbd5574fd92cb0d312baea26EXAMPLE`
`region:`	リソース AWS リージョン がある場所。Resource Explorer には、指定されたに存在するリソースのみが結果に含まれます AWS リージョン。 `region:us-east-1` **Note** リージョンコードだけを (`us-east-1` などのフィルタなしで) タイプ入力しても、`region:us-east-1` と同じ結果は返されません。これは、フィルターではない自由形式のテキストキーワードであるため、リージョンコードが個々の要素に分割して解釈されるためです。例えば、`us-east-1` は、`us`、`east`、および `1` として検索されます。`region:` プレフィックスを使用した場合、このような構成要素の分割は行われません。
`region:global`	`region:` フィルターの特別なケースで、個人に関連付けられていない AWS リージョン が、対象範囲内のグローバルと見なされるリソースを検索するために使用できます。 `region:global` **Note** キーワード `global` だけを入力しても、「global」という文字がグローバルリソースに関連付けられていないため、`region:global` と同じ結果は返されません。キーワードとして `global` を入力すると、その文字列がそのままリソースに関連付けられているリソースだけが返されます。

フィルター名	説明と例
resourcetype:	*service*:*type* 表記のリソースタイプです。Resource Explorer の検索結果には、指定されたタイプのリソースのみが含まれます。 resourcetype:ec2:instance
resourcetype.supports:	このフィルターを使用すると、タグをサポートするリソースを検索できます。 tagsはサポートされている唯一の値です。Resource Explorer には、タグ付け可能なリソースのみが結果に含まれます。 resourcetype.supports:tags
service:	リソースのタイプ AWS のサービス に関連付けられている 。Resource Explorer の検索結果には、指定されたサービスによって作成および管理されるリソースのみが含まれます。 service:ec2
tag:	タグキーと値のペアは <key>=<value> で表されます。Resource Explorer の検索結果には、一致するキーと指定された値の両方を持つタグを付したリソースのみが含まれます。 tag:environment=production
tag:all	Resource Explorer でリソースタイプがサポートされていない場合でも、ユーザーが作成したタグがアタッチされたリソースを検索できるtag:フィルターの特殊なケース。 **Note** サービス作成によるAWS タグが付いたリソースは、このフィルターの検索結果からは除外されませんす。

フィルター名	説明と例
`tag:none`	アタッチされたユーザー作成タグをまったく含まないリソースを検索できる、特殊な `tag:` フィルターです。 **Note** サービス作成によるAWS タグが付いたリソースは、このフィルターの検索結果からは除外されませんす。
`tag.key:`	タグキー。Resource Explorer の検索結果には、値に関係なく、一致するキーを持つタグを持つリソースのみが含まれます。 `tag.key:environment`
`tag.value:`	タグ値。Resource Explorer の検索結果には、キー名に関係なく、値が一致するタグを持つリソースのみが含まれます。 `tag.value:production`

フィルター演算子

次の表に示す演算子のいずれかを文字列の一部に含めることで、キーワードとフィルターを変更できます。

演算子	説明と例
"*multiple word phrase*" または "*hyphenated-phrase* "	1 つのキーワードとして扱うべき複数ワードの語句を、二重引用符 (" ") で囲みます。Resource Explorer の検索結果には、フレーズ全体のすべての単語が指定された順序で一致するリソースのみが含まれます。 二重引用符を使用しない場合、Resource Explorer はフレーズをスペースまたはハイフン区切り単位で個々の構成要素に分割し、それぞれの構成要素と一致するリソースを、それらが一緒になっていなかったり、順序が異なっていてもすべて含めます。引用符は、演算子の後にすべて配置する必要があります。 `"This matches only resources with the whole sentence."`

演算子	説明と例

	`This matches resources with any of the words.`
	
	`"us-east-1"` — 指定したリージョンそのものに関連付けられているリソースのみを検索します。
	
	`us-east-1` —「us」、「east」、「1」を含むすべてのリソースを照合します。
	
	`-tag:"environment=production"`

`keyword``*`	プレフィックスワイルドカード照合。ワイルドカード文字 (アスタリスク *) は文字列の末尾にのみ配置できます。Resource Explorer の検索結果には、* の前のプレフィックステキストで始まる値を持つリソースのみが含まれます。次の例は、 で始まるすべての に一致し AWS リージョン ますus-east。 `region:us-east*` **Important** 統合検索では、文字列の最初のキーワードの末尾にワイルドカード文字 (*) 演算子が自動的に挿入されます。つまり、統合検索結果には、指定されたキーワードで始まる任意の文字列と一致するリソースが含まれます。 これに対し、Resource Explorer コンソールの [リソース検索] ページの [クエリ] テキストボックスから実行する検索では、ワイルドカード文字は自動的には追加されません。検索文字列の任意の用語の後に、* を手動で挿入できます。

演算子	説明と例

-*keyword*	Not 演算子。キーワードまたはフィルターの先頭にハイフン (-) を付けると、検索結果が逆転します。Resource Explorer の検索結果は、この演算子の後に続くキーワードまたはフィルターに一致するリソースすべてを除外します。次の例では、Amazon EC2サービスに関連付けられたすべてのリソースが結果から除外されます。
	
	`-service:ec2`
	
	Important
	コマンドを使用し AWS CLI `search`、`--query-string` パラメータ値に -演算子を最初の文字として使用する場合は、パラメータ名を通常のスペース文字ではなく等しい符号文字 (=) で区切る必要があります。スペース文字を使用すると、 は文字列をCLI誤って解釈します。例えば、以下のクエリは失敗します。
	
	`aws resource-explorer-2 search --query-string "-tag:none region:us-east-1"`
	
	次の修正されたクエリ文字列は、スペースを = に置き換えたもので、期待どおりに機能します。
	
	`aws resource-explorer-2 search --query-string ="-tag:none region:us-east-1"`
	
	- がパラメータ値の最初の文字にならないようにクエリ文字列内のフィルターの順序を変更すれば、標準のスペース文字を使用することができます。次のクエリ文字列は正しく機能します。
	
	`aws resource-explorer-2 search --query-string "region:us-east-1 -tag:none"`

演算子	説明と例
\<special character>	解釈ではなく表示されているとおりに含める必要がある特殊文字にはエスケープ処理ができます。テキストにいずれかの特殊文字 (* " - : = \) が含まれている場合、その文字が表記どおりに解釈されるよう、文字の前にバックスラッシュ (\) を付ける必要があります。次の例は、ハイフン (-) 文字 ("my-key-word") を含む自由形式のテキストキーワードの使用方法を示しています。 また、Resource Explorer がハイフンでつながれた表現を 3 つのキーワードに分割しないように、フレーズ全体を二重引用符で囲むことができます。 "my\-key\-word" リテラルバックスラッシュを挿入するには、2 つのバックスラッシュ文字を連続して挿入します。最初のバックスラッシュはエスケープと解釈され、2 番目のバックスラッシュが挿入するリテラル文字です。 "some_text\\some_more_text"

> **Note**
>
> ビューにリソースに添付されたタグが含まれている場合、Search オペレーションは検索文字列の検証エラーを返しません。有効でないフィルターは、自由形式のテキスト検索としても解釈できるためです。例えば、cat:blue はフィルターのように見えますが、Resource Explorer はそれをフィルターとして解析しません。cat: は定義済みの有効フィルターに含まれていないからです。代わりに、Resource Explorer は文字列全体を自由形式の検索文字列として解釈し、タグキー名や の一部などと一致させますARN。
> 次のいずれかに該当する場合には、オペレーションが検証エラーを返します。
>
> - ビューがタグに関する情報を含んでいない。
> - 検索クエリがタグフィルター (tag.key:、tag.value:、tag:のいずれか) を明示的に使用している

Resource Explorer による検索クエリの例

以下の例は、AWS Resource Explorer で使用できる一般的な種類のクエリの構文を示しています。

> **Important**
>
> AWS CLI `search` のコマンド、および最初の文字として - 演算子を含む `--query-string` パラメーター値を使用する場合は、通常のスペース文字の代わりにイコール記号 (=) を使用してパラメーター名とパラメーター値を区切る必要があります。スペース文字を使用すると、CLI は文字列を誤って解釈します。例えば、以下のクエリは失敗します。
>
> ```
> aws resource-explorer-2 search --query-string "-tag:none region:us-east-1"
> ```
>
> スペースを = で置換した次の修正済みクエリは、期待どおりに機能します。
>
> ```
> aws resource-explorer-2 search --query-string="-tag:none region:us-east-1"
> ```
>
> - がパラメータ値の最初の文字にならないようにクエリ文字列内のフィルターの順序を変更すれば、標準のスペース文字を使用することができます。次のクエリは機能します。
>
> ```
> aws resource-explorer-2 search --query-string "region:us-east-1 -tag:none"
> ```

タグ付けされていないリソースの検索

アカウント内で属性ベースのアクセス制御 (ABAC) を使用するか、コストベースの割り当てを使用するか、リソースに対してタグベースの自動化を実行する場合は、アカウント内のどのリソースにタグがないかを把握しておく必要があります。次のクエリ例では、特殊なフィルタータグ：none を使用して、ユーザー生成タグのないリソースをすべて返します。

この `tag:none` フィルターは、ユーザーが作成したタグにのみ適用されます。AWS によって生成、管理されるタグはこのフィルター処理の例外となり、結果には引き続き表示されます。

```
tag:none
```

AWS で作成されたシステムタグもすべて除外するには、次の例に示すように 2 つ目のフィルターを追加します。クエリ文字列の最初の要素は、すべてのユーザー作成タグを除外するという点で前の

例と重複しています。AWS で作成されたシステムタグは常に aws の文字で始まります。したがって、tag.key フィルターで論理演算子 NOT (-) を使用することにより、キー名が aws で始まるタグを持つリソースをすべて除外することもできます。

```
tag:none -tag.key:aws*
```

タグ付けされているリソースの検索

任意のタイプのタグを持つリソースをすべて検索するには、以下のように 論理演算子 NOT (-) と特殊ケースの tag: none フィルターを組み合わせて使用します。

```
-tag:none
```

特定のタグが欠落しているリソースの検索

また、ABAC に関連して、指定されたキーのタグがないリソースをすべて検索したい場合があると思います。次の例では、論理演算子 NOT - を使用して、キー名 Department のタグがないすべてのリソースを返します。

```
-tag.key:Department
```

無効なタグ値を持つリソースの検索

コンプライアンス上の理由から、重要なタグのタグ値が欠落していたり、スペルが間違っていたりするリソースをすべて検索することをお勧めします。次の例では、キー名 environment のタグを持つすべてのリソースを返します。ただし、このクエリでは、prod、integ、devのいずれかの有効な値を持つリソースはすべて除外されます。このクエリで表示される結果には、調査して修正する必要のある他の何らかの値が含まれています。

> **Important**
>
> Resource Explorer の検索では大文字と小文字は区別されないので、大文字・小文字の使用だけが異なるキー名や値は判別することができません。たとえば、次の例の値は、PROD、prod、PrOd、または任意のバリエーションと一致します。ただし、アプリケーションによっては、大文字と小文字を区別してタグを使用する場合があります。小文字のみのタグキー名と値を使用するなど、組織における大文字・小文字使用戦略を標準化すること

をお勧めします。一貫したアプローチをとることで、大文字・小文字の使用方法だけが異なるタグの使用に伴う混乱を避けることができます。

```
tag.key:environment -tag:environment=prod -tag:environment=integ -tag:environment=dev
```

AWS リージョン のサブセット内のリソースの検索

'*' ワイルドカード演算子の使用により、世界の特定の地域内のすべてのリージョンを照合できます。次の例では、ヨーロッパ (EU) 域内の各リージョンにあるすべてのリソースを返します。

```
region:eu-*
```

グローバルリソースの検索

個々のリージョンとは関係がないと思われるグローバルリソースを検索するには、`region:` フィルターに特殊ケース `global` 値を使用してください。

```
region:global
```

特定のリージョン内の特定のタイプのリソースの検索

複数のフィルターを使用した場合、Resource Explorer はプレフィックスと暗示の AND 論理演算子の組み合わせにより式を評価します。次の例では、アジア太平洋 (香港) リージョン AND にあるリソースのすべてが Amazon EC2 インスタンスであることを返します。

```
region:ap-east-1 resourcetype:ec2:instance
```

> **Note**
>
> 暗示的 AND であるため、リソースに関連付けられる値が 1 つしかない属性には 1 つのフィルタしか使用できません。たとえば、1 つのリソースは 1 つの AWS リージョン にのみ属することができます。そのため、以下のクエリは結果を返しません。
>
> ```
> region:us-east-1 region:us-west-1
> ```

この制限は、tag:、tag.key:、tag.value:など、同時に複数の値を持つことができる属性のフィルターには適用されません。

複数のワードを含むリソースの検索

複数ワードの用語を二重引用符 (") で囲むと、指定した順序で用語全体が一致する結果だけが返されます。二重引用符を使用しない場合、Resource Explorer は用語を構成する個々の単語と一致するすべてのリソースを返します。たとえば、次のクエリは二重引用符を使用しているので、用語 "west wing" 全体と一致するリソースのみを返します。このクエリは、us-west-2 AWS リージョン (またはコードに west を含む他のリージョン) 内のリソースや、「west」を伴わなず「wing」のみと一致する文字列のリソースの照合は行いません。

```
"west wing"
```

指定した CloudFormation スタックの一部であるリソースの検索

特定の AWS CloudFormation スタックの一部としてリソースを作成すると、すべてのリソースにスタックの名前が自動的にタグ付けされます。次の例では、指定したスタックの一部として作成されたすべてのリソースを返します。

```
tag:aws:cloudformation:stack-name=my-stack-name
```

AWS Management Console での統合検索の使用

AWS Management Console では、各 AWS コンソールページの上部に検索バーが表示されます。この検索バーから、AWS のサービス ドキュメントやブログトピックを検索したり、各 AWS サービス コンソールのページに直接移動したりできます。また、必要な Resource Explorer 機能をオンにして統合検索機能を有効にすると、ユーザー AWS アカウント 内の必要なリソースも検索結果に含めることができます。

統合検索を使用すると、ユーザーは最初に AWS Resource Explorer コンソールに移動しなくても、どの AWS のサービス コンソールからでもリソースを検索できます。

> **Tip**
>
> 統合検索バーを使用してリソースだけを検索したい場合は、まず **`/Resources`** を入力してから検索クエリを開始します。これにより、検索結果では、リソースではない結果よりも AWS リソースのほうが検索結果の上位に表示されます。

トピック

- 統合検索が有効になっているか確認する
- 統合検索を有効にする

> **Important**
>
> 統合検索では、文字列の最初のキーワードの末尾にワイルドカード文字 (*) 演算子が自動的に挿入されます。つまり、統合検索結果には、指定されたキーワードで始まる任意の文字列と一致するリソースが含まれます。
> これに対し、Resource Explorer コンソールの [リソース検索] ページの [クエリ] テキストボックスから実行する検索では、ワイルドカード文字は自動的には追加されません。検索文字列の任意の用語の後に、* を手動で挿入できます。

統合検索が有効になっているか確認する

AWS アカウント で統合検索が有効になっているかどうかを確認するには、[設定] ページの上部を見てください。Resource Explorer は、各要件の現在のステータスをそこに表示します。統合検索に対する要件は次のとおりです。

- 少なくとも 1 つの AWS リージョン 内で Resource Explorer を有効にする必要があります。Resource Explorer インデックスのあるリージョンのリソースのみが、統合検索結果に表示されます。
- いずれか選択したリージョンにアグリゲーターインデックスを作成する必要があります。このリージョンで実行される検索では、アカウントに登録されているすべてのリージョンの結果が返されます。
- アグリゲーターインデックスを含むデフォルトビューをリージョンに作成する必要があります。リソースの統合検索を使用する必要があるすべてのユーザーには、このデフォルトビューを使用する権限が必要です。
- `resource-explorer-2:Get*`、`resource-explorer-2:List*`、`resource-explorer-2:Describe*`、`resource-explorer-2:Search`の各アクションを実行するアクセス権限を付与する AWS Identity and Access Management (IAM) アクセス権限ポリシーが、ユーザーが使用する IAM プリンシパルに割り当てられている必要があります。独自のカスタム IAM ポリシーを使用して、これらの権限を付与することもできます。これらの権限は、すでに以下の利用可能な AWS マネージドポリシーに組み込まれています。
  - AWSResourceExplorerReadOnlyAccess
  - AWSResourceExplorerFullAccess

統合検索を有効にする

どの AWS コンソールから統合検索を行ってもアカウントのリソースが検索結果に含まれるようにするには、次の手順を完了する必要があります。

1. アカウントの 1 つ以上の AWS リージョン でAWS Resource Explorer を有効化します。
2. アグリゲーターインデックスを格納する 1 つのリージョンを登録します。
3. アグリゲーターインデックスを含むリージョンにデフォルトビューを作成します。

CloudFormation を使用した Resource Explorer リソースの作成

AWS Resource Explorer は、AWS リソースのモデル化およびセットアップに役立つサービスである AWS CloudFormation に統合されています。これにより、リソースとインフラストラクチャの作成、管理に費やす時間を短縮できます。必要なすべての AWS リソースを説明するテンプレートを作成すれば、CloudFormation がお客様に代わってこれらのリソースのプロビジョニングや設定を処理します。リソースの例としては、インデックス、ビュー、または AWS リージョン へのデフォルトビューの割り当てなどがあります。

CloudFormation を使用すると、テンプレートを再利用して Resource Explorer リソースをいつでも繰り返しセットアップできます。リソースを一度記述するだけで、同じリソースを複数の AWS アカウント やリージョンで何度でもプロビジョニングすることができます。

AWS CloudFormation を使用して Resource Explorer を AWS Organizations にデプロイする

AWS CloudFormation StackSets を使用して、組織のすべてのアカウントを対象に Resource Explorer をデプロイできます。組織でメンバーアカウントを追加または作成すると、StackSets で、新しいメンバーアカウント向けに、指定したアグリゲーターインデックスを含めたそれぞれの AWS リージョン のインデックスを自動的に設定することができます。手順については、「組織内のアカウントへの Resource Explorer のデプロイ」を参照してください。

Resource Explorer と CloudFormation テンプレート

Resource Explorer および関連サービスのリソースをプロビジョニングして設定するには、AWS CloudFormation テンプレートについて理解しておく必要があります。テンプレートは、JSONまたはYAMLでフォーマットされたテキストファイルです。これらのテンプレートは、CloudFormation スタックでプロビジョニングするリソースについて記述します。JSON や YAML に不慣れな方は、AWS CloudFormation Designer を使えば CloudFormation テンプレートを使いこなすことができます。詳細については、「AWS CloudFormation ユーザーガイド」の「AWS CloudFormation Designer とは」を参照してください。

Resource Explorer は、CloudFormation での次のリソースタイプの作成をサポートします。

- インデックス — リージョンにインデックスを作成し、そのリージョンの Resource Explorer を有効にします。インデックスはローカルインデックスまたはアグリゲーターインデックスのいずれかを指定できます。AWS アカウント詳細については、で Resource Explorer を有効に AWS リー

ジョン してリソースのインデックスを作成する および アグリゲーターインデックスを作成してクロスリージョン検索を有効にする を参照してください。

- ビュー — ユーザが検索を実行したときにどのような結果を表示できるかを決定するビューを作成します。すべての検索操作についてビューを指定する必要があります。アクセスさせるビューを使用する権限をユーザーに付与する必要があります。詳細については、「検索アクセス許可を提供するための Resource Explorer ビューの管理」を参照してください。

> **Note**
>
> 同じリージョンでビューを作成する前に、そのリージョンにインデックスを作成する必要があります。インデックスとビューを同じスタックの一部として作成する場合は、次のテンプレート例のようにビューの DependsOn 属性を使用して、インデックスが最初に作成されるようにします。

- DefaultViewAssociation — 指定されたビューをそのリージョンのデフォルトとして割り当てます。ユーザーが検索操作に使用するビューを明示的に指定しない場合、Resource Explorer はユーザーが検索を実行しているリージョンに関連付けられたデフォルトビューを使用しようとします。詳細については、「AWS リージョン のデフォルトビューを設定する」を参照してください。

次の例は、同じリージョンに 1 つのインデックスと 1 つのビューを作成し、そのビューをリージョンのデフォルトに設定する方法を示しています。

YAML

```
Description: >-
  Sample CFN Stack setting up Resource Explorer with an aggregator index and a default
 view
Resources:
  SampleIndex:
    Type: 'AWS::ResourceExplorer2::Index'
    Properties:
      Type: AGGREGATOR
      Tags:
        Purpose: ResourceExplorer Sample CFN Stack
  SampleView:
    Type: 'AWS::ResourceExplorer2::View'
    Properties:
      ViewName: mySampleView
      IncludedProperties:
        - Name: tags
```

```
      Tags:
        Purpose: ResourceExplorer Sample CFN Stack
    DependsOn: SampleIndex
  SampleDefaultViewAssociation:
    Type: 'AWS::ResourceExplorer2::DefaultViewAssociation'
    Properties:
      ViewArn: !Ref SampleView
```

JSON

```
{
    "Description": "Sample CFN Stack setting up Resource Explorer with an aggregator
 index and a default view ",
    "Resources": {
        "SampleIndex": {
            "Type": "AWS::ResourceExplorer2::Index",
            "Properties": {
                "Type": "AGGREGATOR",
                "Tags": {
                    "Purpose": "ResourceExplorer Sample Stack"
                }
            }
        },
        "SampleView": {
            "Type": "AWS::ResourceExplorer2::View",
            "Properties": {
                "ViewName": "mySampleView",
                "IncludedProperties": [
                    {
                        "Name": "tags"
                    }
                ],
                "Tags": {
                    "Purpose": "ResourceExplorer Sample CFN Stack"
                }
            },
            "DependsOn": "SampleIndex"
        },
        "SampleDefaultViewAssociation": {
            "Type": "AWS::ResourceExplorer2::DefaultViewAssociation",
            "Properties": {
                "ViewArn": {
                    "Ref": "SampleView"
```

```
                    }
                }
            }
        }
    }
```

Resource Explorer のインデックスおよびビュー向け JSON テンプレートと YAML テンプレートの例を含む詳細情報については、「AWS CloudFormation ユーザーガイド」の「ResourceExplorer2 のリソースタイプリファレンス」を参照してください。

AWS CloudFormation の詳細情報

CloudFormation の詳細については、以下のリソースを参照してください。

- AWS CloudFormation
- AWS CloudFormation ユーザーガイド
- AWS CloudFormation コマンドラインインターフェイスユーザーガイド

チャットアプリケーションでの Amazon Q Developer を用いたリソースの検索

チャットアプリケーションでの Amazon Q Developer 自然言語での質問により AWS のサービス および AWS リソースに関する情報を検索、発見できます。チャットアプリケーションでの Amazon Q Developer は、サービス関連の質問に対し、関連する AWS ドキュメントやサポート記事の抜粋を用いてチャットチャネルで直接回答します。チャットアプリケーションでの Amazon Q Developer は、Resource Explorer を使用してリソース関連の質問に対する回答を検索して発見します。

詳細については、「チャットアプリケーションでの Amazon Q Developer 管理ガイド」の「チャットアプリケーションでの Amazon Q Developer とは」を参照してください。

AWS リソースに関する質問

チャットアプリケーションでの Amazon Q Developer は、Resource Explorer を使用してリソースを検索し発見します。チャットアプリケーションでの Amazon Q Developer はこれらの検索結果をリストに表示します。このリストには一致するリソースの上位 5 つまでが表示され、さらにリソースタイプ、AWS リージョン、タグで結果を絞り込むことができます。

前提条件

チャットアプリケーションでの Amazon Q Developer リソース関連の質問をするには、次のことを行う必要があります。

- AWS リージョン にアクティブなインデックスとビューがあること、またその中に少なくとも 1 つのデフォルトビューが存在することを確認してください。インデックスとビューを使用することにより、Resource Explorer でリソースをカタログ化してクエリできます。詳細については、「Resource Explorer の用語と概念」を参照してください。
- チャネルのアクセス許可スキームに応じて、 AWSResourceExplorerReadOnlyAccess ポリシーをチャネルロールまたは適切な各ユーザーロールに追加します。
- チャンネルガードレールポリシーでアクセス AWSResourceExplorerReadOnlyAccess 許可が付与されていることを確認します。

リソースに関するよくある質問

これらの質問はチャットチャネルから直接投稿できます。赤い文字列の文言を自分の情報に置き換えて問い合わせしてください。

```
@aws What services am I using in Region?
```

```
@aws What are the resources in my account with tags?
```

```
@aws What lambda functions do I have?
```

のセキュリティ AWS Resource Explorer

のクラウドセキュリティが最優先事項 AWS です。 AWS のお客様は、セキュリティを最も重視する組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャからメリットを得られます。

セキュリティは、 AWS とユーザーの間で共有される責任です。責任共有モデルでは、これをクラウドのセキュリティおよびクラウド内のセキュリティと説明しています。

- クラウドのセキュリティ — AWS は、 AWS のサービス で実行されるインフラストラクチャを保護する責任を担います AWS クラウド。また、 は、お客様が安全に使用できるサービス AWS も提供します。コンプライアンスAWS プログラムコンプライアンスプログラム の一環として、サードパーティーの監査者は定期的にセキュリティの有効性をテストおよび検証。Resource Explorer に適用されるコンプライアンスプログラムの詳細については、「コンプライアンスプログラムAWS のサービス による対象範囲内の」、「コンプライアンスプログラム」を参照してください。
- クラウドのセキュリティ — お客様の責任は AWS のサービス 、使用する によって決まります。また、お客様は、データの機密性、会社の要件、適用される法律や規制など、その他の要因についても責任を負います。

このドキュメントは、 を使用する際の責任共有モデルの適用方法を理解するのに役立ちます AWS Resource Explorer。ここでは、セキュリティとコンプライアンスの目標を満たすように Resource Explorer を設定する方法を説明します。また、Resource Explorer リソースのモニタリングや保護 AWS のサービス に役立つ他の の使用方法についても説明します。

内容

- IAM ポリシーを にアップグレードする IPv6
- のアイデンティティとアクセスの管理 AWS Resource Explorer
- でのデータ保護 AWS Resource Explorer
- AWS Resource Explorer のコンプライアンス検証
- AWS Resource Explorer での耐障害性
- のインフラストラクチャセキュリティ AWS Resource Explorer

IAM ポリシーを にアップグレードする IPv6

AWS Resource Explorer のお客様は、IAMポリシーを使用して IP アドレスの許容範囲を設定し、設定された範囲外の IP アドレスが Resource Explorer にアクセスできないようにしますAPIs。

resource-explorer-2。*region*Resource Explorer がホストされている .api.aws ドメインAPIsは、IPv6に加えて をサポートするようにアップグレードされていますIPv4。

アドレスを処理するように更新されていない IP IPv6 アドレスフィルタリングポリシーは、クライアントが Resource Explorer APIドメインのリソースにアクセスできなくなる可能性があります。

から IPv4 へのアップグレードの影響を受けるお客様 IPv6

aws:sourceIp を含むポリシーでデュアルアドレス指定を使用しているお客様は、このアップグレードの影響を受けます。デュアルアドレス指定は、ネットワークが IPv4と の両方をサポートしていることを意味しますIPv6。

デュアルアドレス指定を使用している場合は、現在IPv4フォーマットアドレスで設定されているIAMポリシーを更新して、IPv6フォーマットアドレスを含める必要があります。

アクセスに関する問題については、サポート にお問い合わせください。

> **Note**
>
> 次のお客様は、アップグレードの影響は受けません。
>
> - IPv4 ネットワークのみを利用しているお客様。
> - IPv6 ネットワークのみを利用しているお客様。

IPv6 とは?

IPv6 は、最終的に を置き換えることを意図した次世代 IP 標準ですIPv4。以前のバージョン ではIPv4、32 ビットのアドレス指定スキームを使用して 43 億台のデバイスをサポートしています。IPv6 代わりに、 は 128 ビットアドレス指定を使用して、約 340 兆兆 (または 128 番目の電力に 2) デバイスをサポートします。

```
2001:cdba:0000:0000:0000:0000:3257:9652
2001:cdba:0:0:0:0:3257:9652
2001:cdba::3257:965
```

の IAMポリシーの更新 IPv6

IAM ポリシーは現在、 `aws:SourceIp` フィルターを使用して IP アドレスの許容範囲を設定するために使用されます。

デュアルアドレス指定は、 IPv4および IPV6トラフィックの両方をサポートします。ネットワークでデュアルアドレス指定を使用している場合は、IP アドレスフィルタリングに使用されるIAMポリシーがIPv6アドレス範囲を含むように更新されていることを確認する必要があります。

例えば、この Amazon S3 バケットポリシーは、 `Condition`要素`203.0.113.0.*`で許可された IPv4アドレス範囲 `192.0.2.0.*` と を識別します。

```
# https://docs.aws.amazon.com/IAM/latest/UserGuide/
reference_policies_examples_aws_deny-ip.html
{
    "Version": "2012-10-17",
    "Statement": {
        "Effect": "Deny",
        "Action": "*",
        "Resource": "*",
        "Condition": {
            "NotIpAddress": {
                "*aws:SourceIp*": [
                    "*192.0.2.0/24*",
                    "*203.0.113.0/24*"
                ]
            },
            "Bool": {
                "aws:ViaAWSService": "false"
            }
        }
    }
}
```

このポリシーを更新するには、ポリシーの `Condition`要素が更新され、IPv6アドレス範囲 `2001:DB8:1234:5678::/64`と が含まれます`2001:cdba:3257:8593::/64`。

> **Note**
>
> 下位互換性のために必要になるため、NOTREMOVE既存のIPv4アドレスを実行します。

```
"Condition": {
            "NotIpAddress": {
                "*aws:SourceIp*": [
                    "*192.0.2.0/24*", <<DO NOT REMOVE existing IPv4 address>>
                    "*203.0.113.0/24*", <<DO NOT REMOVE existing IPv4 address>>
                    "*2001:DB8:1234:5678::/64*", <<New IPv6 IP address>>
                    "*2001:cdba:3257:8593::/64*" <<New IPv6 IP address>>
                ]
            },
            "Bool": {
                "aws:ViaAWSService": "false"
            }
        }
```

によるアクセス許可の管理の詳細についてはIAM、「 ユーザーガイド」の「 管理ポリシーとインラインポリシーAWS Identity and Access Management 」を参照してください。

クライアントが をサポートできることを確認する IPv6

resource-explorer-2.{region}.api.aws エンドポイントを使用しているお客様は、クライアントが既にIPv6有効になっている他の AWS のサービス エンドポイントにアクセスできるかどうかを確認することをお勧めします。次の手順では、これらのエンドポイントを検証する方法について説明します。

この例では、Linux および curl バージョン 8.6.0 を使用し、api.aws ドメイン にあるエンドポイントを有効にした Amazon Athena サービスエンドポイントを使用します。 IPv6

> **Note**
>
> AWS リージョン を、クライアントが配置されているのと同じリージョンに切り替えます。この例では、米国東部 (バージニア北部) – `us-east-1`エンドポイントを使用します。

1. 次の curl コマンドを使用して、エンドポイントがIPv6アドレスで解決されるかどうかを確認します。

   ```
   dig +short AAAA athena.us-east-1.api.aws
   2600:1f18:e2f:4e05:1a8a:948e:7c08:d2d6
   2600:1f18:e2f:4e03:4a1e:83b0:8823:4ce5
   2600:1f18:e2f:4e04:34c3:6e9a:2b0d:dc79
   ```

2. 次の curl コマンドIPv6を使用して、クライアントネットワークが接続を行えるかどうかを確認します。

```
curl --ipv6 -o /dev/null --silent -w "\nremote ip: %{remote_ip}\nresponse code:
 %{response_code}\n" https://athena.us-east-1.api.aws

remote ip: 2600:1f18:e2f:4e05:1a8a:948e:7c08:d2d6
response code: 404
```

リモート IP が特定され、レスポンスコードが でない場合`0`、 を使用してエンドポイントへのネットワーク接続が正常に行われましたIPv6。

リモート IP が空白の場合、またはレスポンスコードが の場合`0`、クライアントネットワークまたはエンドポイントへのネットワークパスは IPv4のみになります。この設定は、次の curl コマンドで確認できます。

```
curl -o /dev/null --silent -w "\nremote ip: %{remote_ip}\nresponse code:
 %{response_code}\n" https://athena.us-east-1.api.aws

remote ip: 3.210.103.49
response code: 404
```

リモート IP が特定され、レスポンスコードが でない場合`0`、 を使用してエンドポイントへのネットワーク接続が正常に行われましたIPv4。オペレーティングシステムはクライアントに有効なプロトコルを選択する必要があるため、リモート IP は IPv4 アドレスである必要があります。リモート IP が IPv4アドレスでない場合は、次のコマンドを使用して curl に の使用を強制しますIPv4。

```
curl --ipv4 -o /dev/null --silent -w "\nremote ip: %{remote_ip}\nresponse code:
 %{response_code}\n" https://athena.us-east-1.api.aws

remote ip: 35.170.237.34
response code: 404
```

のアイデンティティとアクセスの管理 AWS Resource Explorer

AWS Identity and Access Management （IAM) は、管理者が AWS リソースへのアクセスを安全に制御 AWS のサービス するのに役立つ です。IAM 管理者は、Resource Explorer リソースの使用を認証

(サインイン) および承認 (アクセス許可を持つ) できるユーザーを制御します。IAM は、追加料金なしで AWS のサービス 使用できる です。

トピック

- 対象者
- アイデンティティを使用した認証
- ポリシーを使用したアクセスの管理
- Resource Explorer と の連携方法 IAM
- AWS Resource Explorer アイデンティティベースポリシーの例
- AWS Organizations および Resource Explorer のサービスコントロールポリシーの例
- AWS の マネージドポリシー AWS Resource Explorer
- Resource Explorer でのサービスリンクロールの使用
- アクセス AWS Resource Explorer 許可のトラブルシューティング

対象者

AWS Identity and Access Management （ IAM) の使用方法は、Resource Explorer で行う作業によって異なります。

サービスユーザー – Resource Explorer サービスを使用してジョブを実行する場合は、必要なアクセス許可と認証情報を管理者が用意します。作業を実行するためにさらに多くの Resource Explorer 機能の使用を必要とする場合は、追加のアクセス許可が必要になる場合があります。アクセスの管理方法を理解しておくと、管理者に適切な許可をリクエストするうえで役立ちます。Resource Explorer のいずれかの機能にアクセスできない場合は、アクセス AWS Resource Explorer 許可のトラブルシューティングを参照してください。

サービス管理者 – 社内の Resource Explorer リソースの管理を担当している管理者は、通常、Resource Explorer へのフルアクセスを持っています。各サービスユーザーがどの Resource Explorer 機能やリソースにアクセスできるかを決めるのは管理者の仕事です。その後、サービスユーザーのアクセス許可を変更するリクエストをIAM管理者に送信する必要があります。このページの情報を確認して、 の基本概念を理解しますIAM。Resource Explorer IAMで を使用する方法の詳細については、「」を参照してくださいResource Explorer と の連携方法 IAM。

IAM 管理者 – IAM管理者の場合は、Resource Explorer へのアクセスを管理するポリシーの作成方法の詳細を知りたい場合があります。で使用できる Resource Explorer アイデンティティベースのポリ

シーの例を表示するにはIAM、「」を参照してくださいAWS Resource Explorer アイデンティティベースポリシーの例。

アイデンティティを使用した認証

認証は、アイデンティティ認証情報 AWS を使用して にサインインする方法です。として、IAMユーザーとして AWS アカウントのルートユーザー、またはIAMロールを引き受けて認証 (にサインイン AWS) される必要があります。

ID ソースを介して提供された認証情報を使用して、フェデレーティッド ID AWS として にサインインできます。 AWS IAM Identity Center （ IAM Identity Center) ユーザー、会社のシングルサインオン認証、Google または Facebook の認証情報は、フェデレーティッド ID の例です。フェデレーティッド ID としてサインインすると、管理者は以前にIAMロールを使用して ID フェデレーションをセットアップしていました。フェデレーション AWS を使用して にアクセスすると、間接的にロールを引き受けることになります。

ユーザーのタイプに応じて、 AWS Management Console または AWS アクセスポータルにサインインできます。へのサインインの詳細については AWS、 AWS サインイン ユーザーガイドの「 へのサインイン方法 AWS アカウント」を参照してください。

AWS プログラムで にアクセスする場合、 はソフトウェア開発キット (SDK) とコマンドラインインターフェイス (CLI) AWS を提供し、認証情報を使用してリクエストに暗号化して署名します。 AWS ツールを使用しない場合は、自分でリクエストに署名する必要があります。推奨される方法を使用してリクエストを自分で署名する方法の詳細については、IAM「 ユーザーガイド」の「リクエストの署名 AWS API」を参照してください。

使用する認証方法を問わず、追加セキュリティ情報の提供をリクエストされる場合もあります。例えば、 では、アカウントのセキュリティを高めるために多要素認証 (MFA) を使用する AWS ことをお勧めします。詳細については、AWS IAM Identity Center 「 ユーザーガイド」の「多要素認証の使用」および「 ユーザーガイド」の「多要素認証の使用 (MFA ） AWS」を参照してください。 IAM

AWS アカウント ルートユーザー

を作成するときは AWS アカウント、アカウント内のすべての および リソースへの AWS のサービス 完全なアクセス権を持つ 1 つのサインイン ID から始めます。この ID は AWS アカウント ルートユーザーと呼ばれ、アカウントの作成に使用した E メールアドレスとパスワードでサインインしてアクセスします。日常的なタスクには、ルートユーザーを使用しないことを強くお勧めします。ルートユーザーの認証情報は保護し、ルートユーザーでしか実行できないタスクを実行するときに使用します。ルートユーザーとしてサインインする必要があるタスクの完全なリストについて

は、IAM「 ユーザーガイド」の「ルートユーザーの認証情報を必要とするタスク」を参照してください。

ユーザーとグループ

IAM ユーザーとは、1 人のユーザーまたはアプリケーションに対して特定のアクセス許可 AWS アカウント を持つ 内の ID です。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を持つIAMユーザーを作成する代わりに、一時的な認証情報に依存することをお勧めします。ただし、IAMユーザーとの長期的な認証情報を必要とする特定のユースケースがある場合は、アクセスキーをローテーションすることをお勧めします。詳細については、IAM「 ユーザーガイド」の「長期的な認証情報を必要とするユースケースのアクセスキーを定期的にローテーションする」を参照してください。

IAM グループは、IAMユーザーのコレクションを指定する ID です。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、という名前のグループがありIAMAdmins、そのグループにIAMリソースを管理するアクセス許可を付与できます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユーザーには永続的な長期の認証情報がありますが、ロールでは一時認証情報が提供されます。詳細については、 IAM ユーザーガイドの (ロールではなく) IAM ユーザーを作成するタイミングを参照してください。

ロール

IAM ロールは、特定のアクセス許可 AWS アカウント を持つ 内の ID です。ユーザーと似ていますがIAM、特定の人物には関連付けられていません。IAM ロール を切り替える AWS Management Console ことで、 でロールを一時的に引き受けることができます。または AWS API オペレーションを AWS CLI 呼び出すか、カスタム を使用してロールを引き受けることができますURL。ロールを使用する方法の詳細については、 IAM ユーザーガイドの「ロールを引き受ける方法」を参照してください。

IAM 一時的な認証情報を持つ ロールは、以下の状況で役立ちます。

- フェデレーションユーザーアクセス – フェデレーティッド ID に許可を割り当てるには、ロールを作成してそのロールの許可を定義します。フェデレーティッド ID が認証されると、その ID はロールに関連付けられ、ロールで定義されている許可が付与されます。フェデレーションのロールの詳細については、 IAM ユーザーガイドの「サードパーティー ID プロバイダーのロールの作

成」を参照してください。IAM Identity Center を使用する場合は、アクセス許可セットを設定します。ID が認証された後にアクセスできる内容を制御するために、IAMIdentity Center はアクセス許可セットを のロールに関連付けますIAM。アクセス許可セットの詳細については、「AWS IAM Identity Center ユーザーガイド」の「アクセス許可セット」を参照してください。

- 一時的なIAMユーザーアクセス許可 – IAM ユーザーまたはロールは、特定のタスクに対して異なるアクセス許可を一時的に引き受けるIAMロールを引き受けることができます。
- クロスアカウントアクセス — IAMロールを使用して、別のアカウントの誰か (信頼できるプリンシパル) が自分のアカウントのリソースにアクセスすることを許可できます。クロスアカウントアクセスを許可する主な方法は、ロールを使用することです。ただし、一部の では AWS のサービス、 (プロキシとしてロールを使用する代わりに) リソースに直接ポリシーをアタッチできます。クロスアカウントアクセスのロールとリソースベースのポリシーの違いについては、IAM「 ユーザーガイド」の「 のクロスアカウントリソースアクセスIAM」を参照してください。
- クロスサービスアクセス — 他の の機能 AWS のサービス を使用するものもあります AWS のサービス。例えば、 サービスで呼び出しを行う場合、そのサービスが Amazon でアプリケーションを実行EC2したりAmazon S3にオブジェクトを保存したりするのが一般的です。サービスでは、呼び出し元プリンシパルの許可、サービスロール、またはサービスリンクロールを使用してこれを行う場合があります。
  - 転送アクセスセッション (FAS) – IAM ユーザーまたはロールを使用して でアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、を呼び出すプリンシパルのアクセス許可と AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストのリクエストを使用します。FAS リクエストは、サービスが他の AWS のサービス または リソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するための権限が必要です。FAS リクエストを行う際のポリシーの詳細については、「アクセスセッションの転送」を参照してください。
  - サービスロール – サービスロールは、ユーザーに代わってアクションを実行するためにサービスが引き受けるIAMロールです。IAM 管理者は、 内からサービスロールを作成、変更、削除できますIAM。詳細については、IAM「 ユーザーガイド」の「 にアクセス許可を委任するロールの作成 AWS のサービス」を参照してください。
  - サービスにリンクされたロール – サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント 、 サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示することはできますが、編集することはできません。

- Amazon で実行されているアプリケーション EC2 – IAMロールを使用して、EC2インスタンスで実行され、 AWS CLI または AWS API リクエストを行うアプリケーションの一時的な認証情報を管理できます。これは、EC2インスタンス内にアクセスキーを保存するよりも望ましいです。 AWS ロールをEC2インスタンスに割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスにアタッチされたインスタンスプロファイルを作成します。インスタンスプロファイルには ロールが含まれており、EC2インスタンスで実行されているプログラムが一時的な認証情報を取得できるようにします。詳細については、IAM「 ユーザーガイド」のIAM「ロールを使用して Amazon EC2インスタンスで実行されているアプリケーションにアクセス許可を付与する」を参照してください。

IAM ロールとIAMユーザーのどちらを使用するかについては、IAM「 ユーザーガイド」の「 (ユーザーではなく) IAMロールを作成するタイミング」を参照してください。

ポリシーを使用したアクセスの管理

でアクセスを制御するには、ポリシー AWS を作成し、 AWS アイデンティティまたはリソースにアタッチします。ポリシーは AWS 、アイデンティティまたはリソースに関連付けられているときにアクセス許可を定義するオブジェクトです。 は、プリンシパル (ユーザー、ルートユーザー、または ロールセッション) がリクエストを行うときに、これらのポリシー AWS を評価します。ポリシーでの権限により、リクエストが許可されるか拒否されるかが決まります。ほとんどのポリシーはJSON ドキュメント AWS として に保存されます。JSON ポリシードキュメントの構造と内容の詳細については、「 ユーザーガイド」のJSON「ポリシーの概要」を参照してください。 IAM

管理者はポリシーを使用して AWS JSON、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

デフォルトでは、ユーザーやロールに権限はありません。必要なリソースに対してアクションを実行するアクセス許可をユーザーに付与するには、IAM管理者はIAMポリシーを作成できます。その後、管理者はIAMポリシーをロールに追加し、ユーザーはロールを引き受けることができます。

IAM ポリシーは、オペレーションの実行に使用する方法に関係なく、アクションのアクセス許可を定義します。例えば、`iam:GetRole`アクションを許可するポリシーがあるとします。そのポリシーを持つユーザーは、 AWS Management Console、、 AWS CLIまたは AWS からロール情報を取得できますAPI。

アイデンティティベースのポリシー

ID ベースのポリシーは、IAMユーザー、ユーザーのグループ、ロールなどの ID にアタッチできるJSONアクセス許可ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。ID ベースのポリシーを作成する方法については、IAM「 ユーザーガイド」のIAM「ポリシーの作成」を参照してください。

アイデンティティベースのポリシーは、さらにインラインポリシーまたはマネージドポリシーに分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれています。マネージドポリシーは、 内の複数のユーザー、グループ、ロールにアタッチできるスタンドアロンポリシーです AWS アカウント。管理ポリシーには AWS 、管理ポリシーとカスタマー管理ポリシーが含まれます。マネージドポリシーまたはインラインポリシーを選択する方法については、IAM ユーザーガイドの「マネージドポリシーとインラインポリシーの選択」を参照してください。

リソースベースのポリシー

リソースベースのポリシーは、リソースにアタッチするJSONポリシードキュメントです。リソースベースのポリシーの例としては、IAMロール信頼ポリシー と Amazon S3 バケットポリシー があります。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスをコントロールできます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、プリンシパルを指定する必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーティッドユーザー、または を含めることができます AWS のサービス。

リソースベースのポリシーは、そのサービス内にあるインラインポリシーです。リソースベースのポリシーIAMでは、 から AWS 管理ポリシーを使用することはできません。

AWS Resource Explorer はリソースベースのポリシーをサポートしていません。

アクセスコントロールリスト (ACLs)

アクセスコントロールリスト (ACLs) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするアクセス許可を持っているかを制御します。ACLs はリソースベースのポリシーに似ていますが、JSONポリシードキュメント形式は使用されません。

Amazon S3、および Amazon VPCは AWS WAF、 をサポートするサービスの例ですACLs。の詳細についてはACLs、「Amazon Simple Storage Service デベロッパーガイド」の「アクセスコントロールリスト (ACL) 概要」を参照してください。

AWS Resource Explorer は をサポートしていませんACLs。

その他のポリシータイプ

AWS は、追加の低頻度のポリシータイプをサポートします。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- アクセス許可の境界 – アクセス許可の境界は、アイデンティティベースのポリシーがIAMエンティティ (IAMユーザーまたはロール) に付与できる最大アクセス許可を設定する高度な機能です。エンティティにアクセス許可の境界を設定できます。結果として得られる権限は、エンティティのアイデンティティベースポリシーとそのアクセス許可の境界の共通部分になります。`Principal` フィールドでユーザーまたはロールを指定するリソースベースのポリシーでは、アクセス許可の境界は制限されません。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。アクセス許可の境界の詳細については、IAM「 ユーザーガイド」のIAM「エンティティのアクセス許可の境界」を参照してください。
- サービスコントロールポリシー (SCPs) – SCPs は、 の組織または組織単位 (OU) の最大アクセス許可を指定するJSONポリシーです AWS Organizations。 AWS Organizations は、ビジネスが所有する複数の をグループ化して一元管理するためのサービス AWS アカウント です。組織内のすべての機能を有効にすると、サービスコントロールポリシー (SCPs) をアカウントのいずれかまたはすべてに適用できます。は、各 を含むメンバーアカウントのエンティティのアクセス許可SCPを制限します AWS アカウントのルートユーザー。Organizations と の詳細についてはSCPs、AWS Organizations 「 ユーザーガイド」の「サービスコントロールポリシー」を参照してください。
- セッションポリシー - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果としてセッションの権限は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合もあります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。詳細については、「 ユーザーガイド」の「セッションポリシー」を参照してください。 IAM

複数のポリシータイプ

1 つのリクエストに複数のタイプのポリシーが適用されると、結果として作成される権限を理解するのがさらに難しくなります。が複数のポリシータイプが関与する場合にリクエストを許可するかどうか AWS を決定する方法については、「 ユーザーガイド」の「ポリシー評価ロジック」を参照してください。 IAM

Resource Explorer と の連携方法 IAM

IAM を使用して へのアクセスを管理する前に AWS Resource Explorer、Resource Explorer で使用できるIAM機能を理解しておく必要があります。Resource Explorer およびその他の が と AWS のサービス 連携する方法の概要を把握するにはIAM、「 IAMユーザーガイド」の「 AWS のサービス と連携する IAM 」を参照してください。

トピック

- Resource Explorer アイデンティティベースのポリシー
- Resource Explorer タグに基づいた承認
- Resource Explorer IAMロール

他の と同様に AWS のサービス、Resource Explorer には、そのオペレーションを使用してリソースとやり取りするためのアクセス許可が必要です。検索するには、ユーザーはビューの詳細を取得する権限と、そのビューを使用して検索する権限を持っている必要があります。さらに、インデックスやビューを作成したり、それらおよびその他の Resource Explorer 設定を変更するには、追加の権限が必要です。

適切なIAMプリンシパルにこれらのアクセス許可を付与するIAMアイデンティティベースのポリシーを割り当てます。Resource Explorer には、共通のアクセス許可セットを事前定義したいくつかのマネージドポリシーが用意されています。これらをIAMプリンシパルに割り当てることができます。

Resource Explorer アイデンティティベースのポリシー

IAM アイデンティティベースのポリシーでは、特定のリソースに対する許可または拒否されたアクションと、それらのアクションが許可または拒否される条件を指定できます。Resource Explorer は、特定のアクション、リソース、および条件キーをサポートします。JSON ポリシーで使用するすべての要素については、「 ユーザーガイド」の「 IAMJSONポリシー要素のリファレンスIAM」を参照してください。

アクション

管理者はポリシーを使用して AWS JSON、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

JSON ポリシーの `Action`要素は、ポリシーでアクセスを許可または拒否するために使用できるアクションを記述します。ポリシーアクションの名前は通常、関連する AWS APIオペレーションと同じ

です。一致するAPIオペレーションがないアクセス許可のみのアクションなど、いくつかの例外があります。また、ポリシーに複数のアクションが必要なオペレーションもあります。これらの追加アクションは、依存アクションと呼ばれます。

このアクションは、関連付けられたオペレーションを実行するための権限を付与するポリシーで使用されます。

Resource Explorer のポリシーアクションは、アクションの前に `resource-explorer-2` サービスプレフィックスを使用します。例えば、Resource Explorer `Search`APIオペレーションを使用してビューを使用して検索するアクセス許可を付与するには、そのプリンシパルに割り当てられたポリシーに `resource-explorer-2:Search`アクションを含めます。ポリシーステートメントには、`Action` または `NotAction` 要素を含める必要があります。Resource Explorer は、このサービスで実行できるタスクを記述する独自のアクションセットを定義します。これらは Resource Explorer APIオペレーションと一致します。

1 つのステートメントで複数のアクションを指定するには、次の例のようにカンマで区切ります。

```
"Action": [
      "resource-explorer-2:action1",
      "resource-explorer-2:action2"
]
```

ワイルドカード文字 (*) を使用すると、複数のアクションを指定することができます。例えば、`Describe` という単語で始まるすべてのアクションを指定するには、次のアクションを含めます。

```
"Action": "resource-explorer-2:Describe*"
```

Resource Explorer アクションのリストを確認するには、「AWS サービス認証リファレンス」の「AWS Resource Explorerで定義されるアクション」を参照してください。

リソース

管理者はポリシーを使用して AWS JSON、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

Policy `Resource`JSON要素は、アクションが適用されるオブジェクトを指定します。ステートメントには、`Resource` または `NotResource` 要素を含める必要があります。ベストプラクティスとし

て、Amazon リソースネーム (ARN) を使用してリソースを指定します。これは、リソースレベルの許可と呼ばれる特定のリソースタイプをサポートするアクションに対して実行できます。

オペレーションのリスト化など、リソースレベルの権限をサポートしないアクションの場合は、ステートメントがすべてのリソースに適用されることを示すために、ワイルドカード (*) を使用します。

```
"Resource": "*"
```

ビュー

Resource Explorer の主要なリソースタイプはビューです。

Resource Explorer ビューリソースのARN形式は次のとおりです。

```
arn:${Partition}:resource-explorer-2:${Region}:${Account}:view/${ViewName}/${unique-id}
```

Resource Explorer ARNの形式を次の例に示します。

```
arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Search-
View/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111
```

> **Note**
>
> ビューARNの には、すべてのビューが一意であることを確認するために、末尾に一意の識別子が含まれています。これにより、削除された古いビューへのアクセス権を付与した IAMポリシーを使用して、古いビューと同じ名前の新しいビューへのアクセス権を誤って付与することがなくなります。すべての新しいビューは、 が再利用されないように、最後に新しい一意の ID ARNs を受け取ります。

の形式の詳細についてはARNs、「Amazon リソースネーム (ARNs)」を参照してください。

IAM プリンシパルに割り当てられたIAMアイデンティティベースのポリシーを使用し、ビューを として指定しますResource。これにより、あるビューからは 1 つのプリンシパルセットへの検索アクセスを許可し、同時にまったく異なるビューから別のプリンシパルセットへの検索アクセスを許可できます。

例えば、IAMポリシーステートメントProductionResourcesViewで という名前の単一のビューにアクセス許可を付与するには、まずビューの Amazon リソース名 (ARN) を取得します。コンソー

ルのビューページを使用してビューの詳細を表示したり、 ListViewsオペレーションを呼び出して必要なビューARN全体を取得したりできます。次に、1 つのビューの定義のみを変更する権限を付与する次の例のように、それをポリシーステートメントに含めます。

```
"Effect": "Allow",
"Action": "UpdateView",
"Resource": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/
ProductionResourcesView/<unique-id>"
```

特定のアカウントに属するすべてのビューでアクションを許可するには、 の関連部分でワイルドカード文字 (*) を使用しますARN。次の例では、指定した AWS リージョン およびアカウントのすべてのビューに検索権限を付与しています。

```
"Effect": "Allow",
"Action": "Search",
"Resource": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/*"
```

`CreateView` などの一部の Resource Explorer アクションは、次の例のようにリソースがまだ存在しない場合には特定のリソースに対しては実行されません。このような場合は、リソース 全体にワイルドカード文字 (*) を使用する必要がありますARN。

```
"Effect": "Allow",
"Action": "resource-explorer-2:CreateView"
"Resource": "*"
```

ワイルドカード文字で終わるパスを指定すると、承認されたパスのみを使用してビューを作成するように `CreateView` 操作を制限できます。以下のポリシー例は、プリンシパルがパス `view/ProductionViews/` 内のみでビューを作成できるようにする方法を示しています。

```
"Effect": "Allow",
"Action": "resource-explorer-2:CreateView"
"Resource": "arn:aws:resource-explorer-2:us-
east-1:123456789012:view/ProductionViews/*""
```

[Index] (インデックス)

Resource Explorer 機能へのアクセスをコントロールするために使用できるもう 1 つのリソースタイプは、インデックスです。

インデックスを操作する主な方法は、そのリージョンにインデックスを作成することにより AWS リージョン で Resource Explorer をオンにすることです。その後は、ビューを操作して他のほとんどすべてを行います。

インデックスでできることの 1 つは、各リージョンでどのユーザーがビューを作成できるかを制御することです。

> **Note**
>
> ビューを作成すると、 は、インデックスではなく、ビューARNの のみに対して他のすべてのビューアクションIAMを承認します。

インデックスには、アクセス許可ポリシーで参照ARNできる があります。Resource Explorer インデックスARNの形式は次のとおりです。

```
arn:${Partition}:resource-explorer-2:${Region}:${Account}:index/${unique-id}
```

Resource Explorer インデックス の次の例を参照してくださいARN。

```
arn:aws:resource-explorer-2:us-east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-
abcd22222222
```

Resource Explorer アクションの中には、複数のリソースタイプに対して認証をチェックするものがあります。例えば、 CreateViewオペレーションは、Resource Explorer がインデックスを作成した後と同様に、インデックスARNの とビューARNの の両方に対して を許可します。Resource Explorer サービスを管理する権限を管理者に付与するには、`"Resource": "*"` を使用して任意のリソース、インデックス、またはビューに対するアクションを承認します。

あるいは、プリンシパルが特定の Resource Explorer リソースのみを操作できるように制限することもできます。例えば、アクションを指定されたリージョンの Resource Explorer リソースのみに制限するには、インデックスとビューの両方に一致するARNテンプレートを含めることができますが、呼び出すリージョンは 1 つだけです。次の例では、 は、指定されたアカウントの`us-west-2`リージョンのみのインデックスまたはビューの両方ARNに一致します。の 3 番目のフィールドにリージョンを指定しますがARN、最後のフィールドにワイルドカード文字 (*) を使用して、任意のリソースタイプを照合します。

```
"Resource": "arn:aws:resource-explorer-2:us-west-2:123456789012:*
```

詳細については、「AWS サービス認証リファレンス」の「AWS Resource Explorerで定義されるリソース」を参照してください。各リソースARNの を指定できるアクションについては、「 で定義されるアクション AWS Resource Explorer」を参照してください。

条件キー

Resource Explorer にはサービス固有条件キーがありませんが、いくつかのグローバル条件キーの使用をサポートしています。すべての AWS グローバル条件キーを確認するには、「 ユーザーガイド」のAWS 「 グローバル条件コンテキストキーIAM」を参照してください。

管理者はポリシーを使用して AWS JSON、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルが、どのリソースに対してどのような条件下でアクションを実行できるかということです。

`Condition` 要素 (または `Condition` ブロック) を使用すると、ステートメントが有効な条件を指定できます。`Condition` 要素はオプションです。イコールや未満などの 条件演算子 を使用して条件式を作成することで、ポリシーの条件とリクエスト内の値を一致させることができます。

1 つのステートメントに複数の `Condition` 要素を指定する場合、または 1 つの `Condition` 要素に複数のキーを指定する場合、 AWS では `AND` 論理演算子を使用してそれらを評価します。1 つの条件キーに複数の値を指定すると、 は論理`OR`オペレーションを使用して条件 AWS を評価します。ステートメントの権限が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば、IAMユーザー名でタグ付けされている場合にのみ、リソースにアクセスするアクセス許可をIAMユーザーに付与できます。詳細については、「 ユーザーガイド」のIAM「ポリシー要素: 変数とタグIAM」を参照してください。

AWS は、グローバル条件キーとサービス固有の条件キーをサポートします。すべての AWS グローバル条件キーを確認するには、「 ユーザーガイド」のAWS 「 グローバル条件コンテキストキーIAM」を参照してください。

Resource Explorer で使用できる条件キーのリストについては、「AWS サービス認証リファレンス」の「AWS Resource Explorerの条件キー」を参照してください。どのアクションおよびリソースで条件キーを使用できるかについては、「AWS Resource Explorerで定義されるアクション」を参照してください。

例

Resource Explorer のアイデンティティベースポリシーの例を確認するには、「AWS Resource Explorer アイデンティティベースポリシーの例」を参照してください。

Resource Explorer タグに基づいた承認

タグを Resource Explorer ビューにアタッチすることも、Resource Explorer へのリクエストでタグを渡すこともできます。タグに基づいてアクセスを管理するには、`resource-explorer-2:ResourceTag/key-name`、`aws:RequestTag/key-name`、または `aws:TagKeys` の条件キーを使用して、ポリシーの 条件要素でタグ情報を提供します。Resource Explorer リソースへのタグ付けの詳細については、「ビューへのタグの追加」を参照してください。Resource Explorer でタグベースの認証を使用する方法については、「タグベースの認証を使用してビューへのアクセスを制御します。」を参照してください。

Resource Explorer IAMロール

IAM ロールは、特定のアクセス許可 AWS アカウント を持つ 内のプリンシパルです。

Resource Explorer を使用した一時認証情報の使用

一時的な認証情報を使用して、フェデレーションでサインインしたり、 IAMロールを引き受けたり、クロスアカウントロールを引き受けたりすることができます。一時的なセキュリティ認証情報を取得するには、 AssumeRoleや などの AWS Security Token Service （ AWS STS) APIオペレーションを呼び出しますGetFederationToken。

Resource Explorer では、一時認証情報の使用をサポートしています。

サービスリンクロール

サービスにリンクされたロールを使用すると AWS のサービス 、 は他の サービスのリソースにアクセスして、ユーザーに代わってアクションを実行できます。サービスにリンクされたロールはIAMアカウントに表示され、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示できますが、編集することはできません。

Resource Explorer は、サービスリンクロールを使用して作業を実行します。サービスリンクロールの詳細については、「Resource Explorer でのサービスリンクロールの使用」を参照してください。

AWS Resource Explorer アイデンティティベースポリシーの例

デフォルトでは、ロール、グループ、ユーザーなどのAWS Identity and Access Management (IAM) プリンシパルには、Resource Explorer リソースを作成または変更するアクセス許可はありません。また、AWS Management Console や AWS Command Line Interface (AWS CLI)、AWS API を使用してタスクを実行することもできません。IAM 管理者は、各プリンシパルが指定されたリソースで特

定の API オペレーションを実行するのに必要とするアクセス許可をプリンシパルに付与する IAM ポリシーを作成する必要があります。そのうえで、管理者はアクセス許可を必要とする各 IAM プリンシパルにそれらのポリシーをアタッチします。

アクセスを提供するには、ユーザー、グループ、またはロールにアクセス許可を追加します。

- AWS IAM Identity Center のユーザーとグループ:

  アクセス許可セットを作成します。「AWS IAM Identity Center ユーザーガイド」の「シークレットの作成と管理」の手順に従ってください。
- ID プロバイダーを通じて IAM で管理されているユーザー:

  ID フェデレーションのロールを作成する。詳細については、「IAM ユーザーガイド」の「サードパーティー ID プロバイダー (フェデレーション) 用のロールの作成」を参照してください。
- IAM ユーザー:
  - ユーザーが実行できるロールを作成します。手順については、「IAM ユーザーガイド」の「IAM ユーザー用ロールの作成」を参照してください。
  - (非推奨) ポリシーをユーザーに直接アタッチするか、ユーザーをユーザーグループに追加します。「IAM ユーザーガイド」の「ユーザー (コンソール) へのアクセス許可の追加」の指示に従います。

JSON ポリシードキュメントのこれらの例を使用して、IAM アイデンティティベースのポリシーを作成する方法については、「IAM ユーザーガイド」の「JSON タブでのポリシーの作成」を参照してください。

トピック

- ポリシーのベストプラクティス
- Resource Explorer コンソールの使用
- タグに基づいてビューへのアクセスを許可する
- タグベースのビュー作成のためのアクセス許可を付与する
- ユーザーが自分の権限を確認できるようにする

ポリシーのベストプラクティス

ID ベースのポリシーは、お使いのアカウントでそのユーザーが Resource Explorer リソースを作成、アクセス、または削除できるかどうかを決定します。これらのアクションを実行すると、AWS

アカウント に追加料金が発生する可能性があります。アイデンティティベースポリシーを作成したり編集したりする際には、以下のガイドラインと推奨事項に従ってください。

- AWS マネージドポリシーを使用して開始し、最小特権の許可に移行する – ユーザーとワークロードへの許可の付与を開始するには、多くの一般的なユースケースのために許可を付与する AWS マネージドポリシーを使用します。これらは AWS アカウント で使用できます。ユースケースに応じた AWS カスタマーマネージドポリシーを定義することで、許可をさらに減らすことをお勧めします。詳細については、「IAM ユーザーガイド」の「AWS マネージドポリシー」または「AWS ジョブ機能の管理ポリシー」を参照してください。
- 最小特権を適用する – IAM ポリシーで許可を設定するときは、タスクの実行に必要な許可のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアクションを定義します。これは、最小特権アクセス許可とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、「IAM ユーザーガイド」の「IAM でのポリシーとアクセス許可」を参照してください。
- IAM ポリシーで条件を使用してアクセスをさらに制限する – ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。例えば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定することができます。また、AWS のサービス などの特定の AWS CloudFormation を介して使用する場合、条件を使用してサービスアクションへのアクセスを許可することもできます。詳細については、「IAM ユーザーガイド」の [IAM JSON policy elements: Condition] (IAM JSON ポリシー要素：条件) を参照してください。
- IAM Access Analyzer を使用して IAM ポリシーを検証し、安全で機能的な許可を確保する - IAM Access Analyzer は、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM Access Analyzer は 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーを作成できるようサポートします。詳細については、「IAM ユーザーガイド」の「IAM Access Analyzer ポリシーの検証」を参照してください。
- 多要素認証 (MFA) を要求する – AWS アカウント で IAM ユーザーまたはルートユーザーを要求するシナリオがある場合は、セキュリティを強化するために MFA をオンにします。API オペレーションが呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、「IAM ユーザーガイド」の「MFA 保護 API アクセスの設定」を参照してください。

IAM でのベストプラクティスの詳細については、「IAM ユーザーガイド」の「IAM でのセキュリティのベストプラクティス」を参照してください。

Resource Explorer コンソールの使用

プリンシパルが AWS Resource Explorer コンソール内で検索を行うには、一連の最小限のアクセス許可が必要です。必要最小限のアクセス許可を持つ ID ベースのポリシーを作成しないと、Resource Explorer コンソールはアカウント内のプリンシパルに対して意図したとおりに機能しません。

`AWSResourceExplorerReadOnlyAccess` という名前の AWS マネージドポリシーを使用して、アカウント内の任意のビューを使用した Resource Explorer コンソールでの検索を可能にすることができます。1 つのビューのみで検索する権限を付与するには、検索用の Resource Explorer ビューへのアクセス許可の付与 および次の 2 つのセクションの例を参照してください。

AWS CLI または AWS API のみを呼び出すプリンシパルには、最小限のコンソール許可を付与する必要はありません。その場合、そのプリンシパルが実行する必要のある API 操作に一致するアクションのみへのアクセス許可を付与することができます。

タグに基づいてビューへのアクセスを許可する

この例では、お使いの AWS アカウント 内の Resource Explorer ビューへのアクセス許可をアカウント内のプリンシパルに付与します。そのためには、Resource Explorer で検索できるようにするプリンシパルに IAM ID ベースのポリシーを割り当てます。次の IAM ポリシーの例では、呼び出し元のプリンシパルに添付されている `Search-Group` タグが、リクエストで使用されているビューに添付されている同じタグの値と完全に一致する場合にそのリクエストへのアクセスを許可します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "resource-explorer-2:GetView",
                "resource-explorer-2:Search"
            ],
            "Resource": "arn:aws:resource-explorer-2:*:*:view/*",
            "Condition": {
                "StringEquals": {"aws:ResourceTag/Search-Group": "${aws:PrincipalTag/
Search-Group}"}
            }
        }
    ]
}
```

このポリシーをアカウント内の IAM プリンシパルに割り当てることができます。タグ `Search-Group=A` を持つプリンシパルが Resource Explorer ビューを使用して検索を行うには、ビュー側にも `Search-Group=A` タグが付されている必要があります。そうでない場合、そのプリンシパルはアクセスを拒否されます。条件キー名では大文字と小文字が区別されないため、条件タグキー `Search-Group` は `Search-group` と `search-group` の両方に一致します。詳細については、「IAM ユーザーガイド」の「IAM JSON ポリシー要素: 条件」を参照してください。

> **Important**
>
> AWS Management Console の統合検索結果にリソースを表示するには、そのプリンシパルがアグリゲーターインデックスを含む AWS リージョン のデフォルトビューに対する `GetView` 権限と `Search` 権限の両方を持っている必要があります。これらの権限を付与する最も簡単な方法は、高速セットアップまたは詳細設定を使用して Resource Explorer をオンにする時にビューに添付されるデフォルトのリソースベースの権限をそのまま使用することです。
>
> このシナリオでは、まず機密性の高いリソースを除外するようにデフォルトビューを設定してから前の例で説明したようにタグベースのアクセスを許可する追加ビューの設定を検討してください。

タグベースのビュー作成のためのアクセス許可を付与する

この例では、インデックスと同じタグが付けられたプリンシパルのみが、インデックスを含む AWS リージョン のビューを作成できるようにします。そのためには、ID ベースの権限を作成して、プリンシパルがビューを検索できるようにします。

これで、ビュー作成のためのアクセス許可を付与する準備ができました。この例のステートメントは、適切なプリンシパルに `Search` アクセス許可を付与するのに用いるのと同じアクセス許可ポリシーに追加できます。アクションは、ビューが関連付けられるオペレーションとインデックスを呼び出すプリンシパルに付けられたタグに基づいて許可または拒否されます。次の IAM ポリシーの例では、呼び出し元のプリンシパルにアタッチされた `Allow-Create-View` タグの値が、ビューが作成されたリージョンのインデックスにアタッチされた同じタグの値と完全に一致しない場合、ビュー作成リクエストを拒否します。

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

```
            "Effect": "Deny",
            "Action": "resource-explorer-2:CreateView",
            "Resource": "*",
            "Condition": {
                "StringNotEquals": {"aws:ResourceTag/Allow-Create-View":
 "${aws:PrincipalTag/Allow-Create-View}"}
            }
        }
    ]
}
```

ユーザーが自分の権限を確認できるようにする

この例では、ユーザーアイデンティティに添付されたインラインおよびマネージドポリシーの表示を IAM ユーザーに許可するポリシーを作成する方法を示します。このポリシーには、コンソールで、または AWS CLI か AWS API を使用してプログラム的に、このアクションを完了するアクセス許可が含まれています。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
```

```
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS Organizations および Resource Explorer のサービスコントロールポリシーの例

AWS Resource Explorer は、サービスコントロールポリシー (SCPsをサポートします。SCP は、組織内のアクセス許可を管理する目的で組織内の要素にアタッチされるポリシーです。SCP は、SCP をアタッチする 要素の下にある組織 AWS アカウント 内のすべての に適用されます。SCP では、組織のすべてのアカウントで使用可能な最大アクセス許可を一元的に制御できます。これらは、組織のアクセスコントロールガイドラインを確実に AWS アカウント 満たすのに役立ちます。詳細については、AWS Organizations ユーザーガイドの「サービスコントロールポリシー」を参照してください。

前提条件

SCP を使用するには、まず以下のことをする必要があります。

- 組織内のすべての機能の有効化。詳細については、「AWS Organizations ユーザーガイド」の「組織内のすべての機能の有効化」を参照してください。
- SCP を有効にして組織内で使用できるようにするには 詳細については、「AWS Organizations ユーザーガイド」の「ポリシータイプの有効化と無効化」を参照してください。
- 必要な SCP を作成します。SCP の作成の詳細については、AWS Organizations ユーザーガイドの「SCP の作成および更新」を参照してください。

サービスコントロールポリシーの例

次の例は、属性ベースのアクセスコントロール (ABAC) を使用して、Resource Explorer 管理操作へのアクセスを制御する方法を示します。このサンプルポリシーでは、リクエストを行う IAM プリンシパルに `ResourceExplorerAdmin=TRUE` のタグが付されていない限り、検索に必要な 2 つの権限である `resource-explorer-2:Search` および `resource-explorer-2:GetView` を除くすべての Resource Explorer 操作へのアクセスを拒否します。Resource Explorer で ABAC を使用する方

法の詳細については、タグベースの認証を使用してビューへのアクセスを制御します。 を参照してください。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "resource-explorer-2:AssociateDefaultView",
        "resource-explorer-2:BatchGetView",
        "resource-explorer-2:CreateIndex",
        "resource-explorer-2:CreateView",
        "resource-explorer-2:DeleteIndex",
        "resource-explorer-2:DeleteView",
        "resource-explorer-2:DisassociateDefaultView",
        "resource-explorer-2:GetDefaultView",
        "resource-explorer-2:GetIndex",
        "resource-explorer-2:ListIndexes",
        "resource-explorer-2:ListSupportedResourceTypes",
        "resource-explorer-2:ListTagsForResource",
        "resource-explorer-2:ListViews",
        "resource-explorer-2:TagResource",
        "resource-explorer-2:UntagResource",
        "resource-explorer-2:UpdateIndexType",
        "resource-explorer-2:UpdateView""
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
          "StringNotEqualsIgnoreCase": {"aws:PrincipalTag/ResourceExplorerAdmin":
 "TRUE"}
      }
   ]
}
```

AWS の マネージドポリシー AWS Resource Explorer

AWS 管理ポリシーは、 によって作成および管理されるスタンドアロンポリシーです AWS。 AWS 管理ポリシーは、多くの一般的なユースケースに対するアクセス許可を付与するように設計されているため、ユーザー、グループ、ロールへのアクセス許可の割り当てを開始できます。

AWS 管理ポリシーは、すべての AWS お客様が使用できるため、特定のユースケースに対して最小特権のアクセス許可を付与しない場合があることに注意してください。ユースケース別にカスタマーマネージドポリシーを定義して、マネージドポリシーを絞り込むことをお勧めします。

AWS 管理ポリシーで定義されているアクセス許可は変更できません。が AWS 管理ポリシーで定義されたアクセス許可 AWS を更新すると、ポリシーがアタッチされているすべてのプリンシパル ID (ユーザー、グループ、ロール) が更新されます。 は、新しい AWS のサービス が起動されたとき、または既存のサービスで新しい API AWS オペレーションが使用可能になったときに、 AWS 管理ポリシーを更新する可能性が最も高くなります。

詳細については、「IAM ユーザーガイド」の「AWS マネージドポリシー」を参照してください。

Resource Explorer のアクセス許可を含む一般的な AWS マネージドポリシー

- AdministratorAccess – AWS のサービス および リソースへのフルアクセスを許可します。
- ReadOnlyアクセス — AWS のサービス および リソースへの読み取り専用アクセスを許可します。
- ViewOnlyアクセス — のリソースと基本メタデータを表示するアクセス許可を付与します AWS のサービス。

> **Note**
>
> `ViewOnlyAccess` ポリシーに含まれる Resource Explorer `Get*` 権限は、`List` 権限のように動作しますが返される値は 1 つだけです。これは、一つのリージョンには 1 つのインデックスと 1 つのデフォルトビューしか含めることができないためです。

AWS Resource Explorer の マネージドポリシー

- AWSResourceExplorerFullAccess
- AWSResourceExplorerReadOnlyAccess
- AWSResourceExplorerServiceRolePolicy

AWS マネージドポリシー: AWSResourceExplorerFullAccess

`AWSResourceExplorerFullAccess` ポリシーは IAM ID に割り当てることができます。

このポリシーは、Resource Explorer サービスの完全な管理制御を可能にする許可を付与します。Resource Explorer の有効化と管理に関連するすべてのタスクを、お使いのアカウントの AWS リージョン で実行できます。

許可の詳細

このポリシーには、 で Resource Explorer を有効または無効にする、アカウントのアグリゲータインデックスを作成または削除する AWS リージョン、ビューを作成、更新、削除する、検索するなど、Resource Explorer のすべてのアクションを許可するアクセス許可が含まれます。またこのポリシーには、Resource Explorer の一部ではない権限も含まれています。

- `ec2:DescribeRegions` — Resource Explorer が、アカウントのリージョンに関する詳細にアクセスできるようにします。
- `ram:ListResources` — Resource Explorer で、そのリソースが属するリソース共有を一覧表示できるようにします。
- `ram:GetResourceShares` — Resource Explorer 上で、自分が所有している、または共有しているリソース共有に関する詳細を特定できるようにします。
- `iam:CreateServiceLinkedRole` — 最初のインデックス作成により Resource Explorer を有効化する時に、Resource Explorer 側で必要なサービスリンクロールを作成できるようにします。
- `organizations:DescribeOrganization` — Resource Explorer が、組織に関する情報にアクセスできるようにします。

この管理ポリシーの最新バージョンを確認するには、 AWS 「 管理ポリシーリファレンスガイドAWSResourceExplorerFullAccess」の「」を参照してください。 AWS

AWS 管理ポリシー: AWSResourceExplorerReadOnlyAccess

`AWSResourceExplorerReadOnlyAccess` ポリシーは IAM ID に割り当てることができます。

このポリシーは、リソースを発見するためのベーシックな検索を行う読み取り専用アクセス許可をユーザーに付与します。

許可の詳細

このポリシーには、Resource Explorer コンポーネント情報や設定情報を閲覧するための Resource Explorer `Get*`、`List*`、`Search` の各オペレーションを実行する権限をユーザーに付与しますが、ユーザーがそれらの情報を変更することは許可されていません。ユーザーは検索も実行できます。このポリシーには、Resource Explorer にはない 2 つの権限も含まれています。

- `ec2:DescribeRegions` — Resource Explorer が、アカウントのリージョンに関する詳細にアクセスできるようにします。
- `ram:ListResources` — Resource Explorer で、そのリソースが属するリソース共有を一覧表示できるようにします。
- `ram:GetResourceShares` — Resource Explorer 上で、自分が所有している、または共有しているリソース共有に関する詳細を特定できるようにします。
- `organizations:DescribeOrganization` — Resource Explorer が、組織に関する情報にアクセスできるようにします。

この管理ポリシーの最新バージョンを確認するには、 AWS 「 管理ポリシーリファレンスガイド`AWSResourceExplorerReadOnlyAccess`」の「」を参照してください。 AWS

AWS 管理ポリシー: AWSResourceExplorerServiceRolePolicy

IAM エンティティに自分で `AWSResourceExplorerServiceRolePolicy` をアタッチすることはできません。このポリシーは、ユーザーに代わって Resource Explorer がアクションを実行することを許可するサービスリンクロールにアタッチされます。詳細については、「Resource Explorer でのサービスリンクロールの使用」を参照してください。

このポリシーは、Resource Explorer がお持ちのリソースに関する情報の取得に必要とすアクセス許可を付与します。Resource Explorer は、登録する各 に保持 AWS リージョン するインデックスを入力します。

この AWS 管理ポリシーの最新バージョンを確認するには、IAM コンソール`AWSResourceExplorerServiceRolePolicy`の「」を参照してください。

AWS マネージドポリシー: AWSResourceExplorerOrganizationsAccess

IAM ID に `AWSResourceExplorerOrganizationsAccess` を割り当てすることができます。

このポリシーは、Resource Explorer に管理アクセス許可を付与し、このアクセスをサポートする読み取り専用アクセス許可 AWS のサービス を他の に付与します。 AWS Organizations 管理者は、コンソールでマルチアカウント検索を設定および管理するために、これらのアクセス許可が必要です。

許可の詳細

このポリシーには、管理者が組織のマルチアカウント検索を設定できる権限が含まれています。

- `ec2:DescribeRegions` — Resource Explorer が、アカウントのリージョンに関する詳細にアクセスできるようにします。

- `ram:ListResources` — Resource Explorer で、そのリソースが属するリソース共有を一覧表示できるようにします。
- `ram:GetResourceShares` — Resource Explorer 上で、自分が所有している、または共有しているリソース共有に関する詳細を特定できるようにします。
- `organizations:ListAccounts` — Resource Explorer が、組織内のアカウントを識別できるようにします。
- `organizations:ListRoots` — Resource Explorer が、組織内のルートアカウントを識別できるようにします。
- `organizations:ListOrganizationalUnitsForParent` — Resource Explorer が、親組織単位またはルート内の組織単位 (OU) を識別できるようにします。
- `organizations:ListAccountsForParent` — Resource Explorer が、指定したターゲットルートまたは OU に含まれる組織内のアカウントを識別できるようにします。
- `organizations:ListDelegatedAdministrators` — Resource Explorer が、この組織の委任管理者として指定されている AWS アカウントを識別できるようにします。
- `organizations:ListAWSServiceAccessForOrganization` — Resource Explorer が組織との統合が有効になってい AWS のサービス る のリストを識別できるようにします。
- `organizations:DescribeOrganization` — Resource Explorer が、ユーザーのアカウントが属する組織に関する情報を取得できるようにします。
- `organizations:EnableAWSServiceAccess` — Resource Explorer が AWS のサービス (で指定されたサービス`ServicePrincipal`) と の統合を有効にすることを許可します AWS Organizations。
- `organizations:DisableAWSServiceAccess` — Resource Explorer が (で指定された AWS のサービス サービス ServicePrincipal) と の統合を無効にすることを許可します AWS Organizations。
- `organizations:RegisterDelegatedAdministrator` — Resource Explorer が、指定されたメンバーアカウントを有効にして、指定された AWS サービスの組織の機能を管理できるようにします。
- `organizations:DeregisterDelegatedAdministrator` — Resource Explorer が、指定された の委任管理者 AWS アカウント として指定されたメンバーを削除できるようにします AWS のサービス。
- `iam:GetRole` - 指定されたロールに関して、ロールのパス、GUID、ARN、およびそのロールを引き受けるための許可を付与するロールの信頼ポリシーなどの情報を Resource Explorer で取得できるようにします。

- `iam:CreateServiceLinkedRole` — 最初のインデックス作成により Resource Explorer を有効化する時に、Resource Explorer 側で必要なサービスリンクロールを作成できるようにします。

この AWS 管理ポリシーの最新バージョンを確認するには、IAM コンソールAWSResourceExplorerOrganizationsAccessの「」を参照してください。

Resource Explorer の AWS マネージドポリシーの更新

Resource Explorer の AWS マネージドポリシーの更新に関する詳細を、このサービスがこれらの変更の追跡を開始した以降の分について表示します。このページの変更に関する自動通知を受信するには、「Resource Explorer ドキュメント履歴」ページの RSS フィードを購読してください。

変更	説明	日付
AWSResourceExplorerServiceRolePolicy - ポリシーのアクセス許可を更新して、追加のリソースタイプを表示	Resource Explorer は、Resource Explorer AWSResourceExplorerServiceRolePolicy が追加のリソースタイプを表示できるようにするアクセス許可をサービスにリンクされたロールポリシーに追加しました。 • apprunner:ListVpcConnectors • backup:ListReportPlans • emr-serverless:ListApplications • events:ListEventBuses • geo:ListPlaceIndexes • geo:ListTrackers	2023 年 12 月 12 日

変更	説明	日付
	• greengrass:ListComponents • greengrass:ListComponentVersions • iot:ListRoleAliases • iottwinmaker:ListComponentTypes • iottwinmaker:ListEntities • iottwinmaker:ListScenes • kafka:ListConfigurations • kms:ListKeys • kinesisanalytics:ListApplications • lex:ListBots • lex:ListBotAliases • mediapackage-vod:ListPackagingConfigurations • mediapackage-vod:ListPackagingGroups • mq:ListBrokers • personalize:ListDatasetGroups • personalize:ListDatasets • personalize:ListSchemas	

変更	説明	日付
	• `route53:ListHealthChecks` • `route53:ListHostedZones` • `secretsmanager:ListSecrets`	
新しい マネージドポリシー	Resource Explorer に次の AWS マネージドポリシーが追加されました。 • AWSResourceExplorerOrganizationsAccess	2023 年 11 月 14 日
更新されたマネージドポリシー	Resource Explorer は、マルチアカウント検索をサポートするように以下の AWS マネージドポリシーを更新しました。 • AWSResourceExplorerFullAccess • AWSResourceExplorerReadOnlyAccess	2023 年 11 月 14 日

変更	説明	日付
AWSResourceExplorerServiceRolePolicy – Organizations でのマルチアカウント検索をサポートするようにポリシーを更新	Resource Explorer では、Resource Explorer で Organizations でのマルチアカウント検索をサポートするためのアクセス許可をサービスリンクロールポリシー AWSResourceExplorerServiceRolePolicy に追加しました。 • organizations:ListAWSServiceAccessForOrganization • organizations:DescribeAccount • organizations:DescribeOrganization • organizations:ListAccounts • organizations:ListDelegatedAdministrators	2023 年 11 月 14 日

変更	説明	日付
AWSResourceExplorerServiceRolePolicy – 追加のリソースタイプをサポートするようにポリシーを更新しました	Resource Explorer では、サービスで以下のリソースタイプをインデックス化するためのアクセス許可をサービスリンクロールポリシー AWSResourceExplorerServiceRolePolicy に追加しました。 • accessanalyzer:analyzer • acmpca:certificateauthority • amplify:app • amplify:backendenvironment • amplify:branch • amplify:domainassociation • amplifyuibuilder:component • amplifyuibuilder:theme • appintegrations:eventintegration • apprunner:service • appstream:appblock • appstream:application • appstream:fleet • appstream:imagebuilder • appstream:stack • appsync:graphqlapi • aps:rulegroupsnamespace • aps:workspace • apigateway:restapi • apigateway:deployment	2023 年 10 月 17 日

変更	説明	日付
	• athena:datacatalog • athena:workgroup • autoscaling:autoscalinggroup • backup:backupplan • batch:computeenvironment • batch:jobqueue • batch:schedulingpolicy • cloudformation:stack • cloudformation:stackset • cloudfront:fieldlevelencryptionconfig • cloudfront:fieldlevelencryptionprofile • cloudfront:originaccesscontrol • cloudtrail:trail • codeartifact:domain • codeartifact:repository • codecommit:repository • codeguruprofiler:profilinggroup • codestarconnections:connection • databrew:dataset • databrew:recipe • databrew:ruleset • detective:graph • directoryservices:directory • ec2:carriergateway	

変更	説明	日付
	• ec2:verifiedaccessendpoint • ec2:verifiedaccessgroup • ec2:verifiedaccessinstance • ec2:verifiedaccesstrustprovider • ecr:repository • elasticache:cachesecuritygroup • elasticfilesystem:accesspoint • events:rule • evidently:experiment • evidently:feature • evidently:launch • evidently:project • finspace:environment • firehose:deliverystream • faultinjectionsimulator:experimenttemplate • forecast:datasetgroup • forecast:dataset • frauddetector:detector • frauddetector:entitytype • frauddetector:eventtype • frauddetector:label • frauddetector:outcome • frauddetector:variable • gamelift:alias • globalaccelerator:accelerator	

変更	説明	日付
	• globalaccelerator:endpointgroup • globalaccelerator:listener • glue:database • glue:job • glue:table • glue:trigger • greengrass:group • healthlake:fhirdatastore • iam:virtualmfadevice • imagebuilder:componentbuildversion • imagebuilder:component • imagebuilder:containerrecipe • imagebuilder:distributionconfiguration • imagebuilder:imagebuildversion • imagebuilder:imagepipeline • imagebuilder:imagerecipe • imagebuilder:image • imagebuilder:infrastructureconfiguration • iot:authorizer • iot:jobtemplate • iot:mitigationaction • iot:provisioningtemplate • iot:securityprofile • iot:thing	

変更	説明	日付
	• iot:topicruledestination • iotanalytics:channel • iotanalytics:dataset • iotanalytics:datastore • iotanalytics:pipeline • iotevents:alarmmodel • iotevents:detectormodel • iotevents:input • iotsitewise:assetmodel • iotsitewise:asset • iotsitewise:gateway • iottwinmaker:workspace • ivs:channel • ivs:streamkey • kafka:cluster • kinesisvideo:stream • lambda:alias • lambda:layerversion • lambda:layer • lookoutmetrics:alert • lookoutvision:project • mediapackage:channel • mediapackage:origi nendpoint • mediatailor:playbackconfigu ration • memorydb:acl • memorydb:cluster • memorydb:parametergroup	

変更	説明	日付
	• memorydb:user • mobiletargeting:app • mobiletargeting:segment • mobiletargeting:template • networkfirewall:firewallpolicy • networkfirewall:firewall • networkmanager:globalnetwork • networkmanager:device • networkmanager:link • networkmanager:attachment • networkmanager:corenetwork • panorama:package • qldb:journalkinesisstreamsforledger • qldb:ledger • rds:bluegreendeployment • refactorspaces:application • refactorspaces:environment • refactorspaces:route • refactorspaces:service • rekognition:project • resiliencehub:app • resiliencehub:resiliencypolicy • resourcegroups:group • route53:recoverygroup • route53:resourceset • route53:firewalldomain	

変更	説明	日付
	• route53:firewallrulegroup • route53:resolverendpoint • route53:resolverrule • sagemaker:model • sagemaker:notebook instance • signer:signingprofile • ssmincidents:responseplan • ssm:inventoryentry • ssm:resourcedatasync • states:activity • timestream:database • wisdom:assistant • wisdom:assistantassociation • wisdom:knowledgebase	

変更	説明	日付
AWSResourceExplorerServiceRolePolicy – 追加のリソースタイプをサポートするようにポリシーを更新しました	Resource Explorer では、サービスで以下のリソースタイプをインデックス化するためのアクセス許可をサービスリンクロールポリシー AWSResourceExplorerServiceRolePolicy に追加しました。 • codebuild:project • codepipeline:pipeline • cognito:identitypool • cognito:userpool • ecr:repository • efs:filesystem • elasticbeanstalk:application • elasticbeanstalk:applicationversion • elasticbeanstalk:environment • iot:policy • iot:topicrule • stepfunctions:statemachine • s3:bucket	2023 年 8 月 1 日

変更	説明	日付
AWSResourceExplorerServiceRolePolicy – 追加のリソースタイプをサポートするようにポリシーを更新しました	Resource Explorer では、サービスで以下のリソースタイプをインデックス化するためのアクセス許可をサービスリンクロールポリシー `AWSResourceExplorerServiceRolePolicy` に追加しました。 • elasticache:cluster • elasticache:globalreplicationgroup • elasticache:parametergroup • elasticache:replicationgroup • elasticache:reserved-instance • elasticache:snapshot • elasticache:subnetgroup • elasticache:user • elasticache:usergroup • lambda:code-signing-config • lambda:event-source-mapping • sqs:queue	2023 年 3 月 7 日

変更	説明	日付
新しいマネージドポリシー	Resource Explorer に次の AWS マネージドポリシーが追加されました。 • AWSResourceExplorerFullAccess • AWSResourceExplorerReadOnlyAccess • AWSResourceExplorerServiceRolePolicy	2022 年 11 月 7 日
Resource Explorer で変更の追跡を開始	Resource Explorer が AWS マネージドポリシーの変更の追跡を開始しました。	2022 年 11 月 7 日

Resource Explorer でのサービスリンクロールの使用

AWS Resource Explorer は AWS Identity and Access Management 、（IAM ）サービスにリンクされたロール を使用します。サービスにリンクされたロールは、Resource Explorer に直接リンクされた一意のタイプのIAMロールです。サービスにリンクされたロールは、Resource Explorer によって事前定義されており、 AWS のサービス ユーザーに代わってサービスが他の を呼び出すために必要なすべてのアクセス許可が含まれています。

サービスリンクロールを使用することで、必要なアクセス許可を手動で追加する必要がなくなるため、Resource Explorer の設定が簡単になります。サービスリンクロールの権限は Resource Explorer により定義されており、別段定義されない限り、Resource Explorer のみがそのロールを引き受けることができます。定義されたアクセス許可には、信頼ポリシーとアクセス許可ポリシーの両方が含まれており、そのアクセス許可ポリシーを他のIAMエンティティに割り当てることはできません。

サービスにリンクされたロールをサポートする他のサービスの詳細については、 ユーザーガイドのAWS 「 と連携する のサービスIAMIAM」を参照してください。[サービスリンクロール] 列が はいになっているサービスを見つけます。サービスにリンクされたロールに関するドキュメントをサービスで表示するには、[Yes] (はい) リンクを選択します。

Resource Explorer のサービスリンクロールにおけるアクセス許可

Resource Explorer は、`AWSServiceRoleForResourceExplorer` という名前のサービスリンクロールを使用します。このロールは、Resource Explorer サービスに、 AWS アカウント ユーザーに代わって のリソースと AWS CloudTrail イベントを表示し、検索をサポートするようにそれらのリソースのインデックスを作成するアクセス許可を付与します。

`AWSServiceRoleForResourceExplorer` サービスリンクロールは、次のサービスプリンシパルがロールを引き受けるサービスのみを信頼します。

- `resource-explorer-2.amazonaws.com`

という名前のロール許可ポリシーAWSResourceExplorerServiceRolePolicyは、サポートされているリソースのリソース名とプロパティを取得するための読み取り専用アクセスを Resource Explorer に許可します AWS 。Resource Explorer がサポートするサービスとリソースを確認するには、「Resource Explorer で検索可能なリソースタイプ」を参照してください。このロールが実行できるすべてのアクションの完全なリストについては、 IAMコンソールで`AWSResourceExplorerServiceRolePolicy`ポリシーを表示できます。

プリンシパルは、ユーザー、グループ、ロールなどのIAMエンティティです。アカウントの最初のリージョンでインデックスを作成するときに Resource Explorer 側でサービスリンクロールを自動作成させる場合、タスクを実行するプリンシパルが必要とするのは Resource Explorer インデックスの作成に必要な権限のみです。を使用してサービスにリンクされたロールを手動で作成するには IAM、タスクを実行するプリンシパルに、サービスにリンクされたロールを作成するアクセス許可が必要です。詳細については、「 ユーザーガイド」の「サービスにリンクされたロールのアクセス許可IAM」を参照してください。

Resource Explorer のサービスリンクロールの作成

サービスリンクロールを手動で作成する必要はありません。で Resource Explorer を有効にするか AWS Management Console、 AWS CLI または を使用してアカウント AWS リージョン の最初の CreateIndexで を実行すると AWS API、Resource Explorer によってサービスにリンクされたロールが作成されます。

このサービスリンクロールを削除した後に再作成する必要がある場合は、同じプロセスで、アカウントにロールを再作成することができます。アカウントの最初のリージョンRegisterResourceExplorerにいる場合、Resource Explorer によってサービスにリンクされたロールが再度作成されます。

Resource Explorer のサービスリンクロールの編集

Resource Explorer では、`AWSServiceRoleForResourceExplorer` サービスリンクロールの編集を許可していません。サービスリンクロールを作成すると、多くのエンティティによってロールが参照される可能性があるため、ロール名を変更することはできません。ただし、 を使用してロールの説明を編集することはできますIAM。詳細については、「 IAMユーザーガイド」の「サービスにリンクされたロールの編集」を参照してください。

Resource Explorer のサービスリンクロールの削除

IAM コンソール、、または AWS API を使用して AWS CLI、サービスにリンクされたロールを手動で削除できます。これを行うには、まず AWS リージョン アカウントのすべての から Resource Explorer インデックスを削除してから、サービスにリンクされたロールを手動で削除する必要があります。

> **Note**
>
> リソース削除時に Resource Explorer サービスでそのロールが使用されている場合、削除は失敗することがあります。失敗した場合は、すべてのリージョンのすべてのインデックスが削除されていることを確認し、数分待ってからもう一度オペレーションを実行してみてください。

を使用してサービスにリンクされたロールを手動で削除するには IAM

IAM コンソール、、または AWS API を使用して AWS CLI、`AWSServiceRoleForResourceExplorer`サービスにリンクされたロールを削除します。詳細については、「 ユーザーガイド」の「サービスにリンクされたロールの削除IAM」を参照してください。

Resource Explorer サービスリンクロールでサポートされるリージョン

Resource Explorer は、サービスが利用可能なすべてのリージョンでサービスリンクロールの使用をサポートします。詳細については、「Amazon Web Services 全般のリファレンス」の「AWS のサービス エンドポイント」を参照してください。

アクセス AWS Resource Explorer 許可のトラブルシューティング

以下の情報は、Resource Explorer と AWS Identity and Access Management (IAM) の使用時に発生する可能性がある一般的な問題の診断と修正に役立ちます。

トピック

- Resource Explorer でアクションを実行する権限がない
- 自分の 以外のユーザーに Resource Explorer リソース AWS アカウント へのアクセスを許可したい

Resource Explorer でアクションを実行する権限がない

がアクションを実行する権限がないと AWS Management Console 通知した場合は、管理者に連絡してサポートを依頼する必要があります。この操作に使用する認証情報を提供した担当者が管理者です。

以下の例のエラーは、IAM ロール `MyExampleRole` を引き受けたユーザーがコンソールを使用してビューの詳細を確認しようとしているが、`resource-explorer-2:GetView` の許可がない場合に発生します。

```
User: arn:aws:iam::123456789012:role/MyExampleRole is not authorized to perform:
 resource-explorer-2:GetView on resource: arn:aws:resource-explorer-2:us-
east-1:123456789012:view/EC2-Only-View/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111
```

この場合、そのロールを使用するユーザーは、`resource-explorer-2:GetView` アクションを使用したビューへのアクセスを許可するようにロール権限ポリシーの更新を管理者に依頼する必要があります。

自分の 以外のユーザーに Resource Explorer リソース AWS アカウント へのアクセスを許可したい

他のアカウントのユーザーや組織外の人が、リソースにアクセスするために使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定できます。リソースベースのポリシーまたはアクセスコントロールリスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、以下を参照してください:

- Resource Explorer がこれらの機能をサポートしているかどうかを確認するには、「Resource Explorer と の連携方法 IAM」を参照してください。
- 所有 AWS アカウント している のリソースへのアクセスを提供する方法については、IAM ユーザーガイドの「所有 AWS アカウント している別の の IAM ユーザーへのアクセスを提供する」を参照してください。

- リソースへのアクセスをサードパーティー に提供する方法については AWS アカウント、IAM ユーザーガイドの「サードパーティー AWS アカウント が所有する へのアクセスを提供する」を参照してください。
- ID フェデレーションを介してアクセスを提供する方法については、IAM ユーザーガイドの外部で認証されたユーザー (ID フェデレーション) へのアクセスの許可を参照してください。
- クロスアカウントアクセスにおけるロールとリソースベースのポリシーの使用方法の違いについては、「IAM ユーザーガイド」の「IAM でのクロスアカウントのリソースへのアクセス」を参照してください。

でのデータ保護 AWS Resource Explorer

責任 AWS 共有モデル 、 のデータ保護に適用されます AWS Resource Explorer。このモデルで説明されているように、 AWS はすべての を実行するグローバルインフラストラクチャを保護する責任があります AWS クラウド。お客様は、このインフラストラクチャでホストされているコンテンツに対する管理を維持する責任があります。また、使用する AWS のサービス のセキュリティ設定と管理タスクもユーザーの責任となります。データプライバシーの詳細については、「データプライバシーFAQ」を参照してください。欧州でのデータ保護の詳細については、AWS 「 責任共有モデル」とGDPRAWS 「セキュリティブログ」のブログ記事を参照してください。

データ保護の目的で、認証情報を保護し AWS アカウント 、 AWS IAM Identity Center または AWS Identity and Access Management () を使用して個々のユーザーを設定することをお勧めしますIAM。この方法により、それぞれのジョブを遂行するために必要な権限のみが各ユーザーに付与されます。また、次の方法でデータを保護することもお勧めします:

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。1.2 が必要でTLS、1.3 TLS をお勧めします。
- で APIとユーザーアクティビティのログ記録を設定します AWS CloudTrail。証 CloudTrail 跡を使用して AWS アクティビティをキャプチャする方法については、AWS CloudTrail 「 ユーザーガイド」の CloudTrail 「証跡の操作」を参照してください。
- AWS 暗号化ソリューションと、 内のすべてのデフォルトのセキュリティコントロールを使用します AWS のサービス。
- Amazon Macie などの高度なマネージドセキュリティサービスを使用します。これらは、Amazon S3 に保存されている機密データの検出と保護を支援します。
- コマンドラインインターフェイスまたは AWS を介して にアクセスするときに FIPS 140-3 検証済みの暗号化モジュールが必要な場合はAPI、FIPSエンドポイントを使用します。利用可能なFIPSエンドポイントの詳細については、「連邦情報処理標準 (FIPS) 140-3」を参照してください。

お客様の E メールアドレスなどの極秘または機密情報は、タグ、または名前フィールドなどの自由形式のテキストフィールドに配置しないことを強くお勧めします。これには、コンソール、、または API AWS CLIを使用して Resource Explorer または他の AWS のサービス を操作する場合も含まれます AWS SDKs。名前に使用する自由記述のテキストフィールドやタグに入力したデータは、課金や診断ログに使用される場合があります。URL を外部サーバーに提供する場合は、そのサーバーへのリクエストを検証URLするために認証情報を に含めないことを強くお勧めします。

保管中の暗号化

Resource Explorer によって保存されるデータには、リソースのインデックス付きリストと、カスタマーARNsが使用するそれに関連するリスト、およびそれらにアクセスするためのビューが含まれます。

このデータは、256 AWS Key Management Service ビットキー (-256-AWS KMS) で Galois カウンターモード () で Advanced Encryption Standard () を実装する () 対称暗号化キーを使用して保管時に暗号化されますGCM。 AES GCM AES

転送中の暗号化

顧客のリクエストとすべての関連データは、Transport Later Security (TLS) 1.2 以降を使用して転送中に暗号化されます。すべての Resource Explorer エンドポイントHTTPSは、転送中のデータの暗号化をサポートしています。Resource Explorer サービスエンドポイントのリストについては、「AWS 全般のリファレンス」の「AWS Resource Explorer エンドポイントとクォータ」を参照してください。

AWS Resource Explorer のコンプライアンス検証

任意の AWS のサービス が特定のコンプライアンスプログラムの対象範囲内に含まれるかについては、「コンプライアンスプログラムの対象範囲内の AWS のサービス」を参照してください。一般的な情報については、「AWS コンプライアンスプログラム」を参照してください。

AWS Artifact を使用して、サードパーティーの監査レポートをダウンロードできます。詳細については、「AWS Artifact ユーザーガイド」の「」「AWS Artifact でレポートをダウンロードする」を参照してください。

Resource Explorer を使用する際のユーザーのコンプライアンス責任は、ユーザーのデータの機密性や会社のコンプライアンス目標、適用される法律および規制によって決まります。AWS では、コンプライアンスに役立つ以下のリソースを提供しています。

- 「セキュリティ＆コンプライアンスクイックリファレンスガイド」 – これらのデプロイガイドには、アーキテクチャ上の考慮事項の説明と、AWS でセキュリティとコンプライアンスに重点を置いたベースライン環境をデプロイするための手順が記載されています。
- 「Amazon Web Services での HIPAA のセキュリティとコンプライアンスのためのアーキテクチャ」 – このホワイトペーパーは、企業が AWS を使用して HIPAA 対応アプリケーションを作成する方法を説明しています。

> **Note**
>
> すべての AWS のサービス が HIPAA 適格なわけではありません。詳細については、「HIPAA 対応サービスのリファレンス」を参照してください。

- AWS コンプライアンスのリソース - このワークブックおよびガイドのコレクションは、顧客の業界と拠点に適用されるものである場合があります。
- AWS Config デベロッパーガイドの「ルールでのリソースの評価」– AWS Config は、リソース設定が、社内のプラクティス、業界のガイドラインそして規制にどの程度適合しているのかを評価します。
- AWS Security Hub – この AWS のサービスは、AWS 内でのユーザーのセキュリティ状態に関する包括的な見解を提供し、業界のセキュリティ標準、およびベストプラクティスに対するコンプライアンスを確認するために役立ちます。

AWS Resource Explorer での耐障害性

AWS のグローバルインフラストラクチャは AWS リージョン とアベイラビリティーゾーンを中心として構築されます。リージョンには、低レイテンシー、高いスループット、そして高度の冗長ネットワークで接続されている複数の物理的に独立および隔離されたアベイラビリティーゾーンがあります。アベイラビリティーゾーンでは、ゾーン間で中断することなく自動的にフェイルオーバーするアプリケーションとデータベースを設計および運用することができます。アベイラビリティーゾーンは、従来の単一または複数のデータセンターインフラストラクチャよりも可用性が高く、フォールトトレラントで、スケーラブルです。

AWS リージョン とアベイラビリティーゾーンの詳細については、「AWS グローバルインフラストラクチャ」を参照してください。

のインフラストラクチャセキュリティ AWS Resource Explorer

マネージドサービスである AWS Resource Explorer は、 AWS グローバルネットワークセキュリティで保護されています。 AWS セキュリティサービスと がインフラストラクチャ AWS を保護する方法については、AWS 「 クラウドセキュリティ」を参照してください。インフラストラクチャセキュリティのベストプラクティスを使用して AWS 環境を設計するには、「 Security Pillar AWS Well-Architected Framework」の「Infrastructure Protection」を参照してください。

が AWS 公開したAPI呼び出しを使用して、ネットワーク経由で Resource Explorer にアクセスします。クライアントは以下をサポートする必要があります:

- Transport Layer Security (TLS) 。1TLS.2 が必要で、1.3 TLS をお勧めします。
- (Ephemeral Diffie-HellmanPFS) や DHE (Elliptic Curve Ephemeral Diffie-Hellman) などの完全前方秘匿性 ECDHE () を備えた暗号スイート。これらのモードは、Java 7 以降など、ほとんどの最新システムでサポートされています。

さらに、リクエストは、 IAMプリンシパルに関連付けられたアクセスキー ID とシークレットアクセスキーを使用して署名する必要があります。または、AWS Security Token Service (AWS STS) を使用して、一時セキュリティ認証情報を生成し、リクエストに署名することもできます。

AWS グローバルネットワークセキュリティ手順の詳細については、ホワイトペーパー「Amazon Web Services: セキュリティプロセスの概要」を参照してください。

AWS Resource Explorer のモニタリング

モニタリングは、AWS Resource Explorer およびその他の AWS ソリューションの信頼性、可用性、およびパフォーマンスの維持における重要な要素です。AWS は、Resource Explorer をモニタリングし、問題が発生した場合には報告を行い、必要に応じて自動アクションを実行するために以下のモニタリングツールを提供しています。

- AWS CloudTrail は、AWS アカウント により、またはそのアカウントに代わって行われた API コールや関連イベントを取得し、指定した Amazon S3 バケットにログファイルを配信します。AWS を呼び出したユーザーとアカウント、呼び出し元の IP アドレス、および呼び出し日時を特定できます。詳細については、AWS Resource Explorerを使用したAWS CloudTrailAPI コールのログ記録 および AWS CloudTrail ユーザーガイドを参照してください。

AWS Resource Explorerを使用したAWS CloudTrailAPI コールのログ記録

AWS Resource Explorer は、ユーザーやロール、または Resource Explorer 内の AWS のサービス により実行されるアクションをレコードするサービスである AWS CloudTrail と統合されています。CloudTrail は、R esource Explorer 内のすべての API コールをイベントとしてキャプチャします。キャプチャされる呼び出しには、Resource Explorer コンソールからの呼び出しと、Resource Explorer API オペレーションへのコード呼び出しが含まれます。

トレイルを作成することで、Resource Groups のイベントなど、Amazon S3 バケットへの CloudTrail イベントの継続的な配信を有効にすることができます。「トレイル」は、指定した Simple Storage Service (Amazon S3) バケットにイベントをログファイルとして配信するように設定できます。追跡を設定しない場合でも、CloudTrail コンソールの [イベント履歴] で最新のイベントを確認できます。CloudTrail で収集された情報を使用して、Resource Explorer に対して行われたリクエスト、リクエスト元の IP アドレス、リクエスト者、リクエスト日時などの詳細を確認できます。

CloudTrail の詳細については、AWS CloudTrail ユーザーガイドを参照してください。

CloudTrail 上の Resource Explorer 情報

CloudTrail は、アカウント作成時に AWS アカウント で有効になります。Resource Explorer でアクティビティが発生すると、そのアクティビティは[イベント履歴] 内の他の AWS のサービス イベントと共に CloudTrail イベントに記録されます。最近のイベントは、AWS アカウント で表示、検索、ダ

ウンロードできます。詳細については、「CloudTrail Event 履歴でのイベントの表示」を参照してください。

> **Important**
>
> すべての Resource Explorer イベントは、[イベントソース] = [resource-explorer-2.amazonaws.com] を検索することで見つけることができます。

Resource Explorer イベントなど、AWS アカウント 内でのイベントの継続的な記録については、トレイルを作成します。追跡により、CloudTrail はログファイルを Amazon S3 バケットに配信できます。デフォルトでは、コンソールで証跡を作成するときに、証跡がすべての AWS リージョン に適用されます 証跡は、AWS パーティションのすべてのリージョンからのイベントをログに記録し、指定した Amazon S3 バケットにログファイルを配信します。さらに、CloudTrail ログで収集されたイベントデータをより詳細に分析し、それを基にアクションを取るために他の AWS のサービス を設定できます。次のトピックの詳細については、「AWS CloudTrail ユーザーガイド」を参照してください。

- AWS アカウント の追跡の作成
- AWS サービスと CloudTrail ログの統合
- CloudTrail の Amazon SNS 通知の設定
- 複数のリージョンからの CloudTrail ログファイルの受信
- 複数のアカウントからの CloudTrail ログファイルの受信

すべての Resource Explorer アクションは CloudTrail によりログに記録されます。これらのアクションについては、AWS Resource Explorer API リファレンスで説明しています。例えば `CreateIndex`、`DeleteIndex`、`UpdateIndex` の各アクションに対する呼び出しにより、CloudTrail ログファイルにエントリが生成されます。

各イベントまたはログエントリには、リクエストを行ったユーザーに関する情報が含まれます。

- AWS アカウント ルート認証情報
- AWS Identity and Access Management (IAM) ロールまたはフェデレーションユーザーの一時的なセキュリティ認証情報
- IAM ユーザーからの長期的なセキュリティ認証情報
- 別の AWS のサービス

> **Important**
>
> セキュリティ上の理由から、`Tags`、`Filters`、`QueryString`の値はすべて CloudTrail トレイルエントリから墨消しされます。

詳細については、「CloudTrail userIdentity 要素」を参照してください。

Resource Explorer のログファイルエントリについて理解する

「トレイル」は、指定した Simple Storage Service (Amazon S3) バケットにイベントをログファイルとして配信するように設定できます。CloudTrail のログファイルには、単一か複数のログエントリがあります。イベントはあらゆるソースからの単一のリクエストを表し、リクエストされたアクション、アクションの日時、リクエストのパラメータなどの情報が含まれます。CloudTrail ログファイルは、パブリック API コールの順序付けられたスタックトレースではないため、特定の順序では表示されません。

トピック

- CreateIndex
- DeleteIndex
- UpdateIndexType
- 検索
- CreateView
- DeleteView
- DisassociateDefaultView

CreateIndex

次の例は、`CreateIndex` アクションを示す CloudTrail ログエントリです。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAEXAMPLEEXAMPLE:botocore-session-166EXAMPLE",
        "arn": "arn:aws:sts::123456789012:assumed-role/cli-role/botocore-
session-166EXAMPLE",
        "accountId": "123456789012",
```

```
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAEXAMPLEEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/cli-role",
                "accountId": "123456789012",
                "userName": "cli-role"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-08-23T19:13:59Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2022-08-23T19:13:59Z",
    "eventSource": "resource-explorer-2.amazonaws.com",
    "eventName": "CreateIndex",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "10.24.34.15",
    "userAgent": "aws-cli/2.7.14 Python/3.9.11 Windows/10 exe/AMD64 prompt/off command/
resource-explorer-2.create-index",
    "requestParameters": {
        "ClientToken": "792ee665-58af-423c-bfdb-d7c9aEXAMPLE"
    },
    "responseElements": {
        "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
        "State": "CREATING",
        "CreatedAt": "2022-08-23T19:13:59.775Z"
    },
    "requestID": "a193afe9-17ff-4f30-ae0a-73bb0EXAMPLE",
    "eventID": "2ec50598-4de6-474d-bd0e-f5c00EXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

DeleteIndex

次の例は、DeleteIndex アクションを示す CloudTrail ログエントリです。

> **Note**
>
> このアクションでは、そのリージョンのアカウントのすべてのビューも非同期的に削除されるため、削除されたビューごとに一つの DeleteView イベントが発生します。


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAEXAMPLEEXAMPLE:My-Role-Name",
        "arn": "arn:aws:sts::123456789012:assumed-role/My-Admin-Role/My-Delegated-
Role",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAEXAMPLEEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/My-Admin-Role",
                "accountId": "123456789012",
                "userName": "My-Admin-Role"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-08-23T18:33:06Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2022-08-23T19:04:06Z",
    "eventSource": "resource-explorer-2.amazonaws.com",
    "eventName": "DeleteIndex",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "10.24.34.15",
    "userAgent": "aws-cli/2.7.14 Python/3.9.11 Windows/10 exe/AMD64 prompt/off command/
resource-explorer-2.delete-index",
    "requestParameters": {
```

```
        "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111"
    },
    "responseElements": {
        "Access-Control-Expose-Headers": "x-amzn-RequestId,x-amzn-ErrorType,x-amzn-
ErrorMessage,Date",
        "State": "DELETING",
        "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111"
    },
    "requestID": "d7d80bd2-cd2d-47fb-88d6-5133aEXAMPLE",
    "eventID": "675eab39-c514-4d32-989d-0ea98EXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

UpdateIndexType

以下の例は、インデックスをタイプ LOCAL から AGGREGATOR に昇格する UpdateIndexType アクションを示す CloudTrail ログエントリを示しています。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAEXAMPLEEXAMPLE:botocore-session-1661282039",
        "arn": "arn:aws:sts::123456789012:assumed-role/cli-role/botocore-
session-1661282039",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAEXAMPLEEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/cli-role",
                "accountId": "123456789012",
                "userName": "cli-role"
            },
            "webIdFederationData": {},
            "attributes": {
```

```
                "creationDate": "2022-08-23T19:13:59Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2022-08-23T19:21:18Z",
    "eventSource": "resource-explorer-2.amazonaws.com",
    "eventName": "UpdateIndexType",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "10.24.34.15",
    "userAgent": "aws-cli/2.7.14 Python/3.9.11 Windows/10 exe/AMD64 prompt/off command/
resource-explorer-2.update-index-type",
    "requestParameters": {
        "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
        "Type": "AGGREGATOR"
    },
    "responseElements": {
        "Type": "AGGREGATOR",
        "Arn": "arn:aws:resource-explorer-2:us-
east-1:123456789012:index/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111",
        "LastUpdatedAt": "2022-08-23T19:21:17.924Z",
        "State": "UPDATING"
    },
    "requestID": "a145309d-df14-4c2e-a9f6-8ed45EXAMPLE",
    "eventID": "ed33ab96-f5c6-4a77-a69a-8585aEXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```


検索

次の例は、`Search` アクションを示す CloudTrail ログエントリです。

> **Note**
>
> セキュリティ上の理由から、`Tag`、`Filters`、および `QueryString` パラメータへの参照はすべて CloudTrail トレイルエントリ内で墨消しされます。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAEXAMPLEEXAMPLE:botocore-session-1661282039",
        "arn": "arn:aws:sts::123456789012:assumed-role/cli-role/botocore-
session-1661282039",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAEXAMPLEEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/cli-role",
                "accountId": "123456789012",
                "userName": "cli-role"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-08-23T19:13:59Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2022-08-03T16:50:11Z",
    "eventSource": "resource-explorer-2.amazonaws.com",
    "eventName": "Search",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "10.24.34.15",
    "userAgent": "aws-cli/2.7.14 Python/3.9.11 Windows/10 exe/AMD64 prompt/off command/
resource-explorer-2.search",
    "requestParameters": {
        "QueryString": "***"
    },
    "responseElements": null,
    "requestID": "22320db5-b194-446f-b9f4-e603bEXAMPLE",
    "eventID": "addb3bca-0c41-46bf-a5e6-42299EXAMPLE",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

CreateView

次の例は、`CreateView` アクションを示す CloudTrail ログエントリです。


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAEXAMPLEEXAMPLE:botocore-session-1661282039",
        "arn": "arn:aws:sts::123456789012:assumed-role/cli-role/botocore-
session-1661282039",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAEXAMPLEEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/cli-role",
                "accountId": "123456789012",
                "userName": "cli-role"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-08-23T19:13:59Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-01-20T21:54:48Z",
    "eventSource": "resource-explorer-2.amazonaws.com",
    "eventName": "CreateView",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "10.24.34.15",
    "userAgent": "aws-cli/2.7.14 Python/3.9.11 Windows/10 exe/AMD64 prompt/off command/
resource-explorer-2.create-view",
    "requestParameters": {
        "ViewName": "CTTagsTest",
        "Tags": "***"
    },
    "responseElements": {
        "View": {
            "Filters": "***",
            "IncludedProperties": [],
            "LastUpdatedAt": "2023-01-20T21:54:48.079Z",
```

```
            "Owner": "123456789012",
            "Scope": "arn:aws:iam::123456789012:root",
            "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/
CTTest/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111"
        }
    },
    "requestID": "b22d8ced-4905-42c4-b1aa-ef713EXAMPLE",
    "eventID": "f62e339f-1070-41a8-a6ec-12491EXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

DeleteView

次の例は、DeleteIndex オペレーションによって同じ AWS リージョン 内で DeleteView アクションが自動的に開始されたときのイベントを示す CloudTrail ログエントリです。

> **Note**
>
> 削除されたビューがそのリージョンのデフォルトビューであった場合は、このアクションによってビューのデフォルト設定も非同期的に解除されます。これも一つの DisassociateDefaultView イベントとなります。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAEXAMPLEEXAMPLE:botocore-session-1661282039",
        "arn": "arn:aws:sts::123456789012:assumed-role/cli-role/botocore-
session-1661282039",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAEXAMPLEEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/cli-role",
                "accountId": "123456789012",
```

```
                "userName": "cli-role"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2022-08-23T19:13:59Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2022-09-16T19:33:27Z",
    "eventSource": "resource-explorer-2.amazonaws.com",
    "eventName": "DeleteView",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "10.24.34.15",
    "userAgent": "aws-cli/2.7.14 Python/3.9.11 Windows/10 exe/AMD64 prompt/off command/
resource-explorer-2.delete-view",
    "requestParameters": null,
    "responseElements": null,
    "eventID": "cd174d1e-0a24-4b47-8b67-d024aEXAMPLE",
    "readOnly": false,
    "resources": [{
        "accountId": "334026708824",
        "type": "AWS::ResourceExplorer2::View",
        "ARN": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/
CTTest/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111"
    }],
    "eventType": "AwsServiceEvent",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

DisassociateDefaultView

次の例は、現在のデフォルトビュー上で `DeleteView` オペレーションにより `DisassociateDefaultView` アクションが自動的に開始されたときのイベントを示す CloudTrail ログエントリです。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "accountId": "123456789012",
        "invokedBy": "resource-explorer-2.amazonaws.com"
```

```
    },
    "eventTime": "2022-09-16T19:33:26Z",
    "eventSource": "resource-explorer-2.amazonaws.com",
    "eventName": "DisassociateDefaultView",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "10.24.34.15",
    "userAgent": "aws-cli/2.7.14 Python/3.9.11 Windows/10 exe/AMD64 prompt/off command/
resource-explorer-2.disassociate-default-view",
    "requestParameters": null,
    "responseElements": null,
    "eventID": "d8016cb1-5c23-4ea4-bda2-70b03EXAMPLE",
    "readOnly": false,
    "eventType": "AwsServiceEvent",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

Resource Explorer のトラブルシューティング

Resource Explorer の操作中に問題が発生した場合は、このセクションのトピックを参照してください。本ガイドの [セキュリティ] セクションの「アクセス AWS Resource Explorer 許可のトラブルシューティング」も参照してください。

トピック

- 一般的な問題 (このページ)
- Resource Explorer のセットアップと設定に関する問題のトラブルシューティング
- Resource Explorer での検索に関する問題のトラブルシューティング

一般的な問題

トピック

- Resource Explorer へのリンクを受け取ったが、開くとコンソールにエラーのみが表示されます。
- コンソールの統合検索で CloudTrail ログに「アクセスが拒否されました」エラーが発生する理由は何ですか?

Resource Explorer へのリンクを受け取ったが、開くとコンソールにエラーのみが表示されます。

一部のサードパーティツールでは、Resource Explorer ページへのリンク URL を生成します。ただしこれらの URL には、コンソールを特定の AWS リージョン ページに誘導するパラメーターが含まれていない場合があります。このようなリンクを開くと、Resource Explorer コンソールには使用するリージョンが通知されず、ユーザーが最後にサインインしたリージョンがデフォルトで使用されます。ユーザーがそのリージョンの Resource Explorer にアクセスする権限を持っていない場合、コンソールは米国東部 (バージニア北部) (`us-east-1`) リージョンを使用するか、もしくはコンソールが `us-east-1` にアクセスできない場合は米国西部 (オレゴン州) (`us-west-2`) リージョンを使用しようとします。

ユーザーがこれらのリージョンのインデックスへのアクセス許可を持っていないと、Resource Explorer コンソールはエラーを返します。

この問題を防ぐには、すべてのユーザーが以下の権限を持っていることを確認する必要があります。

- `ListIndexes` — 特定のリソースはありません。* を使用してください。
- アカウントで作成された各インデックスの ARN 用 `GetIndex`。インデックスを削除後に再作成する場合にアクセス許可ポリシーを再実施する必要がないように、* を使用することをお勧めします。

これを実現するための最小限のポリシーは、例えば次の例のようになります。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "resource-explorer-2:GetIndex",
                "resource-explorer-2:ListIndexes",
            ],
            "Resource": "*"
        }
    ]
}
```

あるいは、Resource Explorer を使用する必要のあるすべてのユーザーにAWS マネージド権限 `AWSResourceExplorerReadOnlyAccess`を付与することを検討してもよいでしょう。これにより、これらの必要な権限に加えて、そのリージョンで利用可能なビューを表示し、それらのビューを使用して検索するのに必要な権限が付与されます。

コンソールの統合検索で CloudTrail ログに「アクセスが拒否されました」エラーが発生する理由は何ですか?

AWS Management Console での統合検索により、プリンシパルは AWS Management Console 内のどのページからでも検索を実行できます。Resource Explorer がオンになっていて、統合検索をサポートするように設定されている場合、検索結果にはそのプリンシパルのアカウントのリソースが含まれる可能性があります。統合検索バーに入力し始めると、統合検索は `resource-explorer-2:ListIndexes` 操作を呼び出して、ユーザーのアカウントのリソースを結果に含めてもよいかどうかを確認しようとします。

統合検索は、現在サインインしているユーザーの権限を使用してこのチェックを実行します。そのユーザーに、添付された AWS Identity and Access Management (IAM) アクセス許可ポリシーで付

与される resource-explorer-2:ListIndexes 呼び出し権限がない場合、チェックは失敗します。その失敗は、CloudTrail ログの Access denied エントリとして追加されます。

この CloudTrail ログエントリには以下の特徴があります。

- イベントソース：resource-explorer-2.amazonaws.com
- イベント名：ListIndexes
- エラーコード：403 (アクセスが拒否されました)

以下の AWS マネージドポリシーには、resource-explorer-2:ListIndexes を呼び出すためのアクセス許可が含まれます。これらのいずれかをプリンシパルに割り当てるか、またはその権限を含むその他のポリシーを割り当てれば、このエラーは発生しません。

- AWSResourceExplorerReadOnlyAccess
- AWSResourceExplorerFullAccess
- ReadOnlyAccess
- ViewOnlyAccess

Resource Explorer のセットアップと設定に関する問題のトラブルシューティング

このセクションの情報を参考にして、最初に AWS Resource Explorer をセットアップまたは設定するときに発生する問題を診断して修復してください。

トピック

- Resource Explorer にリクエストを送信すると、「アクセスが拒否されました」というメッセージが表示される
- 一時的なセキュリティ認証情報を使用してリクエストを送信すると「アクセスが拒否されました」というメッセージが表示される

Resource Explorer にリクエストを送信すると、「アクセスが拒否されました」というメッセージが表示される

- 要求したアクションとリソースを呼び出す権限を持っているかを確認します。管理者は、ロール、グループ、ユーザーなどの IAM プリンシパルに AWS Identity and Access Management (IAM) アクセス許可ポリシーを割り当てることによってアクセス許可を付与します。

  アクセスを提供するには、ユーザー、グループ、またはロールにアクセス許可を追加します。

  - AWS IAM Identity Center のユーザーとグループ:

    アクセス許可セットを作成します。「AWS IAM Identity Center ユーザーガイド」の「シークレットの作成と管理」の手順に従ってください。

  - ID プロバイダーを通じて IAM で管理されているユーザー:

    ID フェデレーションのロールを作成する。詳細については、「IAM ユーザーガイド」の「サードパーティー ID プロバイダー (フェデレーション) 用のロールの作成」を参照してください。

  - IAM ユーザー:

    - ユーザーが実行できるロールを作成します。手順については、「IAM ユーザーガイド」の「IAM ユーザー用ロールの作成」を参照してください。
    - (非推奨) ポリシーをユーザーに直接アタッチするか、ユーザーをユーザーグループに追加します。「IAM ユーザーガイド」の「ユーザー (コンソール) へのアクセス許可の追加」の指示に従います。

  アクセスする `Resource` に対してリクエストされる `Action` が、ポリシーで許可されている必要があります。

  ポリシーが時間帯または IP アドレス制限などの条件を含む権限を付与する記述をしている場合は、リクエストを送信する際にそれらの条件を満たす必要もあります。IAM プリンシパル向けポリシーを確認または変更する方法については、「IAM ユーザーガイド」の「IAM ポリシーの管理」を参照してください。

- 手動で API リクエストに署名する (AWS SDK を使用しない) 場合は、正しくリクエストに署名していることを確認してください。

一時的なセキュリティ認証情報を使用してリクエストを送信すると「アクセスが拒否されました」というメッセージが表示される

- リクエストの作成に使用している IAM プリンシパルに適切なアクセス許可があるかどうかを確認してください。一時的なセキュリティ認証情報を使用するアクセス許可は IAM に定義されたプリンシパルから生じるものであり、そのプリンシパルに付与されたアクセス許可に制限されます。一時的なセキュリティ認証情報のアクセス許可がどのように決定されるかについては、「IAM ユーザーガイド」の「一次的セキュリティ認証情報のアクセス許可管理」を参照してください。
- リクエストが正しく署名されており、そのリクエストの形式が正しいことを確認します。詳細については、選択した SDK の ツールキット ドキュメント、または「IAM ユーザーガイド」の「AWS リソースに対する一時的セキュリティ認証情報の使用」を参照してください。
- 一時的な認証情報が失効していないことを確認します。詳細については、「IAM ユーザーガイド」の「一次的セキュリティ認証情報のリクエスト」を参照してください。

Resource Explorer での検索に関する問題のトラブルシューティング

このセクションの情報を参考にして、Resource Explorer を使用してリソースを検索するときに発生する一般的なエラーの診断と修正を行ってください。

トピック

- Resource Explorer の検索結果に一部のリソースが表示されない
- コンソールの統合検索結果に自分のリソースが表示されない
- コンソールと Resource Explorer の統合検索の結果が異なることがある
- リソースを検索するのに必要なアクセス許可

Resource Explorer の検索結果に一部のリソースが表示されない

以下のリストは、一部のリソースが検索結果に想定どおりに表示されない理由を示しています。

最初のインデックス作成が完了していない

で Resource Explorer を最初に有効にした後 AWS リージョン、インデックス作成とアグリゲータインデックスへのレプリケーションが完了するまでに最大 36 時間かかることがあります。後ほどもう一度検索をお試しください。

まだ新しいリソースである

新しいリソースが Resource Explorer によって発見されローカルインデックスに追加されるまでに、数分かかる場合があります。数分後にもう一度お試しください。

あるリージョンの新しいリソースに関する情報が、まだアグリゲーターインデックスに伝達されていない

1 つのリージョンで検出された新しいリソースの詳細が独自のリージョンでインデックス作成され、アカウントのアグリゲータインデックスにレプリケートされるまでに時間がかかる場合があります。新しいリソースは、レプリケーションが完了した後にのみクロスリージョン検索結果に表示されます。後ほどもう一度検索をお試しください。

そのリソースのあるリージョンで Resource Explorer が有効化されていない

管理者は、Resource Explorer AWS リージョン を操作できる を決定します。[設定] ページには、Resource Explorer が有効化され、インデックスが含まれているリージョンが一覧表示されます。リソースのあるリージョンがオンになっていない場合は、そのリージョンで Resource Explorer を有効にするよう管理者に依頼してください。

リソースが別のリージョンに存在しており、検索を実行したリージョンにはアグリゲーターインデックスが含まれていない

アグリゲーターインデックスを含むリージョンのビューを使用する場合のみ、アカウント内のすべてのリージョンのリソースを検索できます。他のリージョンで検索を実行すると、検索を実行したリージョンのリソースのみが返されます。

ビューのフィルターによりそのリソースが除外されている

各ビューには、検索結果に表示される内容を制限するフィルターが設定されている場合があります。探しているリソースが、検索に使用しているビューのフィルターと一致していることを確認してください。フィルターの詳細については、「」を参照してください[フィルター]。

リソースタイプは Resource Explorer ではサポートされていません

一部のリソースタイプは Resource Explorer ではサポートされていません。詳細については、「[Resource Explorer で検索できるリソースタイプ]」を参照してください。

インデックスまたはビューがコンソールリージョンで設定されていない

インデックスまたはビューが、ウィジェットを使用するコンソールで予想されるリージョンで設定されていない場合、期待される結果は表示されません。詳細については、「[アグリゲーターインデックスを作成してクロスリージョン検索を有効にする]」を参照してください。

ビューにタグが含まれていない

タグは Resource Explorer ウィジェットで必要です。ビューにタグが含まれていない場合、リソースは結果に含まれません。詳細については、「ビューへのタグの追加」を参照してください。

検索で間違った検索クエリ構文を使用している

Resource Explorer での検索は、このサービスに固有です。正しい構文がないと、必要なリソースは見つかりません。詳細については、「Resource Explorer の検索クエリ構文リファレンス」を参照してください。

リソースに最近タグ付けした

リソースにタグを付けると、リソースが検索結果に表示されるまでに 30 秒の遅延が発生します。

リソースタイプはタグフィルターをサポートしていません

タグフィルターがリソースタイプでサポートされていない場合、Resource Explorer ウィジェットには表示されません。タグフィルターをサポートしていないリソースタイプは次のとおりです。

- `cloudfront:cache-policy`
- `cloudfront:origin-access-identity`
- `cloudfront:function`
- `cloudfront:origin-request-policy`
- `cloudfront:realtime-log-config`
- `cloudfront:response-headers-policy`
- `cloudwatch:dashboard`
- `docdb:globalcluster`
- `elasticache:globalreplicationgroup`
- `iam:group`
- `lambda:code-signing-config`
- `lambda:event-source-mapping`
- `ssm:windowtarget`
- `ssm:windowtask`
- `rds:auto-backup`

- `rds:global-cluster`
- `s3:accesspoint`

コンソールの統合検索結果に自分のリソースが表示されない

統合検索の結果は、 AWS Management Console の各ページの上部にある検索バーに表示されます。ただし、次の設定オプションが完了するまでは、検索結果のクエリと一致するリソースは返されません。

- アカウント内のいずれかのリージョンにアグリゲーターインデックスが存在する必要があります。
- そのリージョンに、アグリゲーターインデックスを含むデフォルトビューが設定されている必要があります。
- すべてのプリンシパル (IAM ロールとユーザー) には、そのデフォルトビュー を使用して検索するアクセス許可が必要です。

コンソールと Resource Explorer の統合検索の結果が異なることがある

統合検索の結果は、各 AWS Management Console ページの上部の検索バーに表示されます。統合検索を使用すると、統合検索プロセスにより、クエリ文字列に最初に入力した用語の末尾にワイルドカード文字 (*) が自動的に挿入されます。このワイルドカード文字は統合検索ボックスには表示されませんが、結果には影響します。

> **Important**
>
> 統合検索では、文字列の最初のキーワードの末尾にワイルドカード文字 (*) 演算子が自動的に挿入されます。つまり、統合検索結果には、指定されたキーワードで始まる任意の文字列と一致するリソースが含まれます。
> これに対し、Resource Explorer コンソールの [リソース検索] ページの [クエリ] テキストボックスから実行する検索では、ワイルドカード文字は自動的には追加されません。検索文字列の任意の用語の後に、* を手動で挿入できます。

リソースを検索するのに必要なアクセス許可

検索を実行するには、操作を呼び出したリージョンにあるビューに対して次の操作の両方を実行する権限が必要です。

- `resource-explorer-2:GetView`
- `resource-explorer-2:Search`

これは、プリンIAMシパルに割り当てられたポリシーに次の例のようなステートメントを追加することで実行できます。

```
        {
            "Effect": "Allow",
            "Action": [
                "resource-explorer-2:GetView",
                "resource-explorer-2:Search"
            ],
            "Resource": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-
View-Name/1a2b3c4d-5d6e-7f8a-9b0c-abcd11111111"
        }
```

特定のビューの Amazon リソースナンバー (ARN) をワイルドカード (*) ARNを含む に置き換えて、一致するすべてのビューにアクセス許可を付与できます。

リクエストでビューを指定しない場合、Resource Explorer はリクエストを行ったリージョンのデフォルトビューを自動的に使用します。デフォルトビューを使用するアクセス許可が付与されていない場合は、管理者に問い合わせてください。

> **Note**
>
> Resource Explorer の検索クエリの結果にリソースが表示される場合でも、そのリソースを操作するにはリソース自体に対する権限が必要です。

Resource Explorer のクォータ

AWS アカウント には、各 AWS のサービス に対してデフォルトのクォータがあります。特に明記されていない限り、クォータはリージョンごとに存在します。一部のクォータについては引き上げをリクエストできますが、その他のクォータについてはリクエストできません。

AWS Resource Explorer のクォータを表示するには、[Service Quotas コンソール] を開きます。ナビゲーションペインで [AWS のサービス] を選択し、[Resource Explorer] を選択します。

クォータの引き上げをリクエストするには、「Service Quotas ユーザーガイド」の「Requesting a quota increase」(クォータ引き上げリクエスト) を参照してください。Service Quotas でクォータがまだ利用できない場合は、[制限の引き上げ] のフォームを使用してください。

次のクォータはリソースエクスプローラーのデフォルトです。

最大クォータ値	デフォルト値
AWS リージョン のビュー数	10

オペレーションのレート制限	デフォルト値
1 秒あたりの最大検索オペレーションの最大数	5
1 秒あたりの非検索オペレーションの最大数	3
アグリゲーターリージョンの 1 か月あたりの検索オペレーションの最大数	10,000
ローカルリージョンの 1 か月あたりの検索オペレーションの最大数	500

AWS Resource Explorer で を使用する AWS SDK

AWS ソフトウェア開発キット (SDKs) は、多くの一般的なプログラミング言語でご利用いただけます。それぞれSDKにAPI、開発者が好みの言語でアプリケーションを簡単に構築できるようにする 、コード例、ドキュメントが用意されています。

SDK ドキュメント	コードの例
AWS SDK for C++	AWS SDK for C++ コード例
AWS CLI	AWS CLI コード例
AWS SDK for Go	AWS SDK for Go コード例
AWS SDK for Java	AWS SDK for Java コード例
AWS SDK for JavaScript	AWS SDK for JavaScript コード例
AWS SDK for Kotlin	AWS SDK for Kotlin コード例
AWS SDK for .NET	AWS SDK for .NET コード例
AWS SDK for PHP	AWS SDK for PHP コード例
AWS Tools for PowerShell	PowerShell コード例のツール
AWS SDK for Python (Boto3)	AWS SDK for Python (Boto3) コード例
AWS SDK for Ruby	AWS SDK for Ruby コード例
AWS SDK for Rust	AWS SDK for Rust コード例
AWS SDK for SAP ABAP	AWS SDK for SAP ABAP コード例
AWS SDK for Swift	AWS SDK for Swift コード例

可用性の例

必要なものが見つからなかった場合。このページの下側にある [Provide feedback (フィードバックを送信)] リンクから、コードの例をリクエストしてください。

Resource Explorer ユーザーガイドのドキュメント履歴

次の表に、 のドキュメントリリースを示します AWS Resource Explorer。このドキュメントの更新に関する通知については、RSSフィードをサブスクライブできます。

変更	説明	日付
新しい検索フィルターの追加	Resource Explorer に新しい`tag:all`検索クエリフィルターが追加され、リソースタイプが Resource Explorer でサポートされていない場合でも、ユーザーが作成したタグが 1 つ以上のアタッチされているリソースを検索できるようになりました。	2024 年 9 月 6 日
コンテンツ組織の改善	トピックタイトルを更新し、コンテンツを再編成して読みやすさと検出可能性を向上させました。	2024 年 8 月 29 日
IAMポリシーを にアップグレードする通知 IPv6	を含むASPENポリシーでデュアルアドレッシングを使用しているお客様は`aws:sourceIp` 、このアップグレードの影響を受けます。デュアルアドレス指定は、ネットワークが IPv4と の両方をサポートすることを意味しますIPv6。	2024 年 7 月 15 日
3 つのリソースタイプのサポートの中止	Resource Explorer は、、`ecs:task`、`ssm:automation-execution` の 3 つのリソースタイプのサポートを終了しました`ssm:patchbaseline` 。	2024 年 7 月 9 日

新しいリソースタイプのサポートを追加	Resource Explorer は、、Amazon Route 53 AWS Key Management Service、Amazon Fraud Detector AWS のサービス など、65 の新しいリソースのサポートを追加しました。	2024 年 2 月 20 日
マネージドポリシーの更新	Resource Explorer は、追加のリソースタイプを表示するサポートを追加しました。`AWSResourceExplorerServiceRolePolicy` AWS マネージドポリシーが更新され、追加のリソースタイプを表示するための Resource Explorer アクセスが付与されました。	2023 年 12 月 12 日
新しい検索フィルターの追加	Resource Explorer で、アプリケーション別のリソース検索ができるようになりました。	2023 年 11 月 16 日
新しいリソースタイプのサポートを追加	Resource Explorer は、 AWS CloudFormation、、 AWS Glue Amazon AWS のサービス を含む 86 の新しいリソースのサポートを追加しました SageMaker。	2023 年 11 月 15 日

Resource Explorer でマルチアカウント検索をサポート	Resource Explorer を使用して、組織内または部署内の AWS アカウント 全体のリソースを検索および発見できるようになりました。詳細については、「マルチアカウント検索を有効にする」を参照してください。	2023 年 11 月 14 日
新しいマネージドポリシーと更新されたマネージドポリシー	Resource Explorer に AWS Organizationsのサポートが追加されました。「AWS マネージドポリシー」が追加および更新され、組織、組織構造、アカウント、および委任された管理者に Resource Explorer へのアクセス権が付与されるようになりました。	2023 年 11 月 14 日
新しいリソースタイプのサポートを追加	Resource Explorer に AWS Organizationsのサポートが追加されました。「AWS マネージドポリシー」が更新され、組織、組織構造、アカウント、および委任された管理者に Resource Explorer へのアクセス権が付与されるようになりました。	2023 年 11 月 14 日
新しいリソースタイプのサポートを追加	Resource Explorer に、Amazon Cognito、 AWS Elastic Beanstalk、Amazon Elastic File System などのサービスからの 12 の新しいリソースタイプのサポートが追加されました。	2023 年 10 月 18 日

新しいリソースタイプのサポートを追加	Resource Explorer に 164 個のリソースのサポートが追加されました。Resource Explorer にインデックスリソースへのアクセスを許可する「AWS マネージドポリシー」が更新され、これらの新しいリソースタイプが含まれるようになりました。	2023 年 10 月 17 日
Resource Explorer が特定のオプトインリージョンで利用可能に	BAH および のお客様は、Resource Explorer にオプトインCGKできるようになりました。	2023 年 10 月 5 日
新しいリソースタイプのサポートを追加	Resource Explorer では AWS のサービス AWS CodeBuild 、、 AWS CodePipeline、Amazon CognitoAmazon Elastic Container Registry、AWS Elastic Beanstalk Amazon Elastic File System、AWS IoT、および からのリソースのサポートが追加されました AWS Step Functions 。Resource Explorer にインデックスリソースへのアクセスを許可する「AWS マネージドポリシー」が更新され、これらの新しいリソースタイプが含まれるようになりました。	2023 年 8 月 1 日
Resource Explorer が、 への検索結果のエクスポートをサポートするようになりました。 CSV	リソース検索ページで検索の結果を 形式のファイルにエクスポートできるようになりました。 CSV	2023 年 4 月 4 日

チャットアプリケーションでの Amazon Q Developer を使用して AWS リソースを検索および検出する	チャットアプリケーションでの Amazon Q Developer を使用して、自然言語の質問を使用してリソースを検索できるようになりました。詳細については、「チャットアプリケーションでの Amazon Q Developer を用いたリソースの検索」を参照してください。	2023 年 3 月 30 日
新しいリソースタイプのサポートを追加	Resource Explorer では、AWS のサービス Amazon ElastiCache、 AWS Lambda および Amazon Simple Queue Service (Amazon) からのリソースのサポートが追加されましたSQS。Resource Explorer にインデックスリソースへのアクセスを許可する「AWS マネージドポリシー」が更新され、これらの新しいリソースタイプが含まれるようになりました。	2023 年 3 月 7 日
IAM ベストプラクティスの更新	IAM ベストプラクティスに合わせてガイドを更新しました。詳細については、「」のセキュリティのベストプラクティスIAMを参照してください。	2022 年 12 月 6 日

新しい AWS マネージドポリシー	Resource Explorer は AWSResourceExplorerFullAccess、、 AWSResourceExplorerReadOnlyAccess、AWSResourceExplorerServiceRolePolicy マネージドポリシーを追加します。	2022 年 11 月 7 日
初回リリース	Resource Explorer ユーザーガイドの初回リリース	2022 年 11 月 7 日

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。