



ユーザーガイド

Amazon Security Lake



Amazon Security Lake: ユーザーガイド

Copyright © 2025 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

Amazon Security Lake とは何ですか?	1
Security Lakeの概要	1
Security Lake の特徴	2
Security Lakeへのアクセス	3
関連サービス	4
概念と用語	6
入門	8
のセットアップ AWS アカウント	8
にサインアップする AWS アカウント	8
管理アクセスを持つユーザーを作成する	9
Security Lake を有効にするために使用するアカウントを特定してください。	10
Security Lake を有効にする際の考慮事項	11
コンソールを使用する	12
ステップ 1: ソースを設定する	12
ステップ 2: ストレージ設定とロールアップリージョンを定義する (オプション)	14
ステップ 3: データレイクを確認して作成する	15
ステップ 4: 独自のデータを表示してクエリする	15
ステップ 5: サブスクライバーを作成する	15
AWS CLI または API の使用	16
ステップ 1: IAM ロールを作成する	16
ステップ 2: Amazon Security Lake を有効にする	17
ステップ 3: ソースを設定する	18
ステップ 4: ストレージ設定とロールアップリージョンを設定する (オプション)	19
ステップ 5: 独自のデータを表示してクエリする	20
ステップ 6: サブスクライバーを作成する	20
複数のアカウントの管理	21
委任された Security Lake 管理者に関する重要な考慮事項	22
委任された管理者を指定するには IAM 許可が必要です	23
委任された Security Lake 管理者を指定し、メンバーアカウントを追加します。	24
コンソールでの新しいアカウント設定の編集	26
委任された Security Lake 管理者を削除する。	27
Security Lake の信頼できるアクセス	28
リージョンの管理	29
リージョン・ステータスのチェック	29

リージョン設定の変更	30
ロールアップリージョンの設定	32
データレプリケーションの IAM ロール	32
AWS Glue パーティションを登録する IAM ロール	35
ロールアップリージョンの追加	36
ロールアップリージョンの更新または削除	38
ソース管理	40
からのデータ収集 AWS のサービス	40
前提：アクセス許可	41
ソース AWS のサービス としての の追加	42
ソースコレクションのステータスの取得	44
ロールのアクセス許可の更新	45
ソース AWS のサービス としての の削除	46
CloudTrail のイベントログ	48
Amazon EKS 監査ログ	49
Route 53 Resolver クエリログ	50
Security Hub の検出結果	50
VPC Flow Logs	51
AWS WAF ログ	51
ソース AWS のサービス としての の削除	46
カスタムソースからのデータ収集	54
カスタムソースを取り込むためのパーティション要件	55
カスタムソースを追加するための前提条件	56
カスタムソースの追加	60
カスタムソースの削除	64
サブスクライバー管理	66
サブスクライバーデータアクセス	67
前提条件	67
データにアクセスできるサブスクライバーを作成する。	70
データサブスクライバーの更新	74
データサブスクライバーを削除する。	76
サブスクライバークエリへのアクセス	76
前提条件	77
クエリアクセス権を持つサブスクライバーの作成	79
クエリアクセス権を持つサブスクライバーの編集	82
Security Lake クエリ	88

Security Lake クエリソースバージョン 1	88
ログソーステーブル	89
データベースリージョン	90
パーティション日付	91
CloudTrail データのクエリ	92
Route 53 リゾルバークエリログのクエリ	95
Security Hub の検出結果のクエリ	97
Amazon VPC フローログのクエリ	100
Security Lake クエリソースバージョン 2	103
ログソーステーブル	89
データベースリージョン	90
パーティション日付	91
Security Lake オブザーバビリティのクエリ	107
CloudTrail データのクエリ	108
Route 53 リゾルバークエリログのクエリ	110
Security Hub の検出結果のクエリ	112
Amazon VPC フローログのクエリ	115
Amazon EKS 監査ログのクエリ	118
v2 AWS WAFログのクエリ	120
ライフサイクル管理	123
保持管理	123
Security Lake での保持設定に関する重要な考慮事項	123
Security Lake を有効にする際の保存設定を行います。	124
保持設定の更新	125
ロールアップリージョン	127
オープンサイバーセキュリティスキーマフレームワーク (OCSF)	128
OCSFとは何ですか	128
OCSF イベントクラス	128
OCSFソースの識別	128
統合	132
AWS のサービス 統合	132
Amazon Bedrock の統合	134
Amazon Detective の統合	135
Amazon OpenSearch Service の統合	135
Amazon OpenSearch Service Ingestion パイプライン統合	136
Amazon OpenSearch Service ゼロ ETL ダイレクトクエリ統合	136

Amazon QuickSight の統合	137
Amazon SageMaker AI 統合	138
AWS AppFabric の統合	138
AWS Security Hub 統合	139
サードパーティ統合	140
クエリ統合	141
Accenture – MxDR	142
Aqua Security	142
Barracuda – Email Protection	142
Booz Allen Hamilton	143
Bosch Software and Digital Solutions – AIShield	143
ChaosSearch	143
Cisco Security – Secure Firewall	143
Claroty – xDome	144
CMD Solutions	144
Confluent – Amazon S3 Sink Connector	144
Contrast Security	144
Cribl – Search	145
Cribl – Stream	145
CrowdStrike – Falcon Data Replicator	145
CrowdStrike – Next Gen SIEM	145
CyberArk – Unified Identify Security Platform	146
Cyber Security Cloud – Cloud Fastener	146
DataBahn	146
Darktrace – Cyber AI Loop	146
Datadog	147
Deloitte – MXDR Cyber Analytics and AI Engine (CAE)	147
Devo	147
DXC – SecMon	147
Eviden— Alsaac (旧 Atos)	148
ExtraHop – Reveal(x) 360	148
Falcosidekick	148
Fortinet - Cloud Native Firewall	148
Gigamon – Application Metadata Intelligence	148
Hoop Cyber	149
HTCD – AI-First Cloud Security Platform	149

IBM – QRadar	149
Infosys	149
Insbuilt	150
Kyndryl – AIOps	150
Lacework – Polygraph	150
Laminar	150
MegazoneCloud	151
Monad	151
NETSCOUT – Omnis Cyber Intelligence	151
Netskope – CloudExchange	151
New Relic ONE	152
Okta – Workforce Identity Cloud	152
Orca – Cloud Security Platform	152
Palo Alto Networks – Prisma Cloud	152
Palo Alto Networks – XSOAR	153
Panther	153
Ping Identity – PingOne	153
PwC – Fusion center	153
Query.AI – Query Federated Search	154
Rapid7 – InsightIDR	154
RipJar – Labyrinth for Threat Investigations	154
Sailpoint	154
Securonix	155
SentinelOne	155
Sentra – Data Lifecycle Security Platform	155
SOC Prime	155
Splunk	156
Stellar Cyber	156
Sumo Logic	156
Swimlane – Turbine	156
Sysdig Secure	157
Talon	157
Tanium	157
TCS	157
Tego Cyber	158
Tines – No-code security automation	158

Torq – Enterprise Security Automation Platform	158
Trellix – XDR	158
Trend Micro – CloudOne	159
Uptycs – Uptycs XDR	159
Vectra AI – Vectra Detect for AWS	159
VMware Aria Automation for Secure Clouds	159
Wazuh	160
Wipro	160
Wiz – CNAPP	160
Zscaler – Zscaler Posture Control	160
セキュリティ	162
Identity and Access Management	163
対象者	163
アイデンティティを使用した認証	164
ポリシーを使用したアクセスの管理	167
Security Lake と IAM の連携方法	170
アイデンティティベースのポリシーの例	179
AWS マネージドポリシー	184
サービスにリンクされたロールの使用	208
データ保護	225
保管中の暗号化	226
転送中の暗号化	228
サービス改善のためのデータ使用をオプトアウトする	228
コンプライアンス検証	229
Security Lake のセキュリティのベストプラクティス	230
Security Lake ユーザーへの最小限のアクセス許可の付与	231
概要ページを表示します。	231
Security Hubとの統合	231
削除 AWS Lambda	231
セキュリティレイクのイベントを監視してください。	231
耐障害性	232
インフラストラクチャセキュリティ	233
Security Lake での構成と脆弱性の分析	233
VPC エンドポイントAWS PrivateLink	234
Security Lake VPC エンドポイントに関する考慮事項	234
Security Lake 用のインターフェイス VPC エンドポイントの作成	234

Security Lake 用の VPC エンドポイントポリシーの作成	235
共有サブネット	236
モニタリング	236
Amazon Security Lake の CloudWatch メトリクス	236
API コールのログ作成	240
CloudTrail での Security Security Security Information	240
Security Lake のログファイルエントリについて	241
リソースのタグ付け	243
タグ付けの基本	243
IAMポリシーでタグを使用する	245
リソースに タグを追加する	245
リソースのタグを編集する	248
リソースのタグを確認する	251
リソースからのタグの削除	253
トラブルシューティング	256
データレイクのステータスのトラブルシューティング	256
Lake Formation 問題のトラブルシューティング	257
テーブルが見つからない	257
400 AccessDenied	258
SYNTAX_ERROR	258
Lake Formation に発信者のプリンシパル ARN を追加できませんでした	258
Lake Formation を使用する CreateSubscriber が新しい RAM リソース共有の招待を作成し なかった	259
Amazon Athena でのクエリのトラブルシューティング	259
クエリを実行しても、データレイク内の新しいオブジェクトは返されません。	259
AWS Glue テーブルにアクセスできない	260
Organizations の問題のトラブルシューティング	260
アクセス拒否エラー	261
IAM 問題のトラブルシューティング	261
Security Lake でアクションを実行することが認可されていない	261
管理ポリシーを超えてアクセス許可を拡張したい	261
iam: PassRole を実行する権限がない	262
自分の 以外のユーザーに Security Lake リソース AWS アカウント へのアクセスを許可した い	262
Security Lake の価格	264
使用状況と推定コストの確認	265

サポートされているリージョンおよびエンドポイント	267
Security Lake を無効にする	268
ドキュメント履歴	270
.....	cclxxvi

Amazon Security Lake とは何ですか？

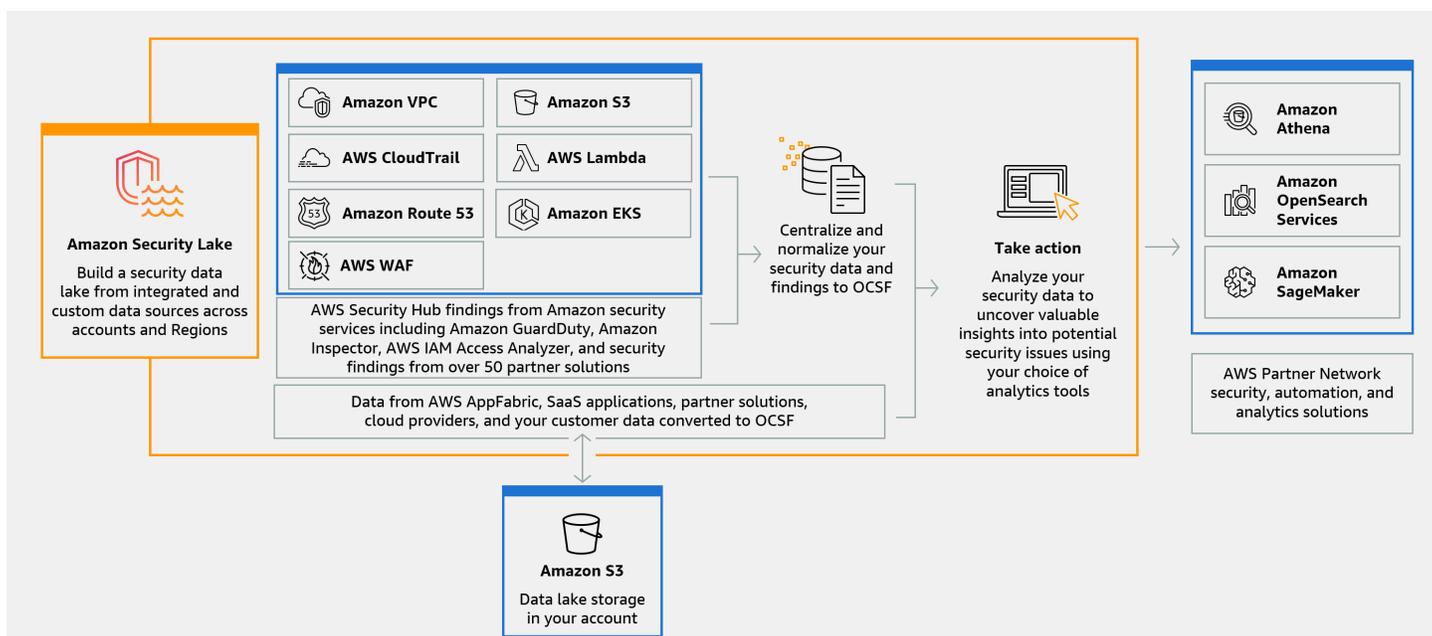
Amazon Security Lake は、完全マネージド型のセキュリティデータレイクサービスです。Security Lake を使用すると、AWS 環境、SaaS プロバイダー、オンプレミス、クラウドソース、サードパーティーソースのセキュリティデータを、に保存されている専用のデータレイクに自動的に一元化できます AWS アカウント。Security Lake はセキュリティデータの分析に役立つため、組織全体のセキュリティ体制をより完全に把握できます。Security Lake を使用すると、ワークロード、アプリケーション、データの保護を強化することもできます。

データレイクは、Amazon Simple Storage Service (Amazon S3) バケットに支えられており、データの所有権はお客様が保持します。

Security Lake は、統合 AWS のサービス や第三者サービスからのセキュリティ関連のログやイベントデータの収集を自動化します。また、保存とレプリケーションの設定をカスタマイズできるため、データのライフサイクル管理にも役立ちます。Security Lake は、取り込んだデータを Apache Parquet 形式と OCSF (OCSF) と呼ばれる標準のオープンソーススキーマに変換します。OCSF サポートにより、Security Lake は のセキュリティデータと幅広いエンタープライズセキュリティデータソースを正規化 AWS し、組み合わせます。

他の AWS のサービス およびサードパーティーサービスは、インシデント対応とセキュリティデータ分析のために Security Lake に保存されているデータをサブスクライブできます。

Security Lakeの概要



Security Lake の特徴

Security Lake がセキュリティ関連のログとイベントデータを一元化、管理、登録するのに役立つ主な方法をいくつか紹介します。

アカウントへのデータ集約

Security Lake は、アカウントに専用のセキュリティデータレイクを作成します。Security Lake は、アカウントやリージョンのクラウド、オンプレミス、カスタムデータソースからログやイベントデータを収集します。データレイクは、Amazon Simple Storage Service (Amazon S3) バケットに支えられており、データの所有権はお客様が保持します。

サポートされているさまざまなログソースとイベントソース

Security Lake は、オンプレミス、サードパーティーのサービスなど AWS のサービス、複数のソースからセキュリティログとイベントを収集します。ログを取り込んだ後は、ソースに関係なく、ログに一元的にアクセスしてライフサイクルを管理できます。Security Lake がログとイベントを収集するソースの詳細については、[Security Lake でのソース管理](#)を参照してください。

データ変換と正規化

また、ネイティブにサポートされている AWS のサービス から Open Cybersecurity Schema Framework (OCSF) オープンソーススキーマにデータを変換します。また、ネイティブにサポートされている から Open Cybersecurity Schema Framework (OCSF) オープンソーススキーマ AWS のサービス にデータを変換します。これにより、後処理を必要とせずに、他のプロバイダー AWS のサービス やサードパーティープロバイダーとデータを互換性を持たせることができます。Security Lake はデータを正規化するため、多くのセキュリティソリューションがこのデータをparallel 使用できます。

サブスクライバーには複数のアクセスレベルがあります。

サブスクライバーは、Security Lake に保存されているデータを使用します。サブスクライバーのデータへのアクセスレベルを選択できます。サブスクライバーは、指定したソースおよび AWS リージョン内のデータのみを使用できます。新しいオブジェクトがデータレイクに書き込まれると、サブスクライバーに自動的に通知される場合があります。または、サブスクライバーはデータレイクからデータをクエリできます。Security Lake は必要な認証情報を自動的に作成し、Security Lake とサブスクライバーの間で交換します。

マルチアカウントおよびマルチリージョンデータ管理

Security Lake は、利用可能なすべてのリージョンや複数の AWS アカウント全体で一元的に有効にできます。Security Lake では、ロールアップリージョンを指定して、複数のリージョンのセキュリティログとイベントデータを統合することもできます。これにより、データレジデンシーのコンプライアンス要件に準拠しやすくなります。

設定とカスタマイズが可能

Security Lake は設定可能でカスタマイズ可能なサービスです。ログ収集を設定するソース、アカウント、リージョンを指定できます。データレイクへのサブスクライバーのアクセスレベルも指定できます。

データライフサイクルの管理と最適化

Security Lake は、カスタマイズ可能な保持設定でデータのライフサイクルを管理し、自動ストレージ階層化によりストレージコストを管理します。Security Lake は、受信するセキュリティデータを自動的に分割し、ストレージとクエリ効率の高い Apache Parquet 形式に変換します。

Security Lakeへのアクセス

Security Lake が現在利用可能なリージョンのリストについては、「[Security Lake のリージョンとエンドポイント](#)」を参照してください。リージョンの詳細については、AWS 全般のリファレンスの「[AWS サービスエンドポイント](#)」を参照してください。

各リージョンでは、次のいずれかの方法で Security Lake にアクセスして使用できます。

AWS Management Console

AWS Management Console は、AWS リソースの作成と管理に使用できるブラウザベースのインターフェイスです。Security Lake コンソールでは、Security Lake アカウントとリソースにアクセスできます。Security Lake のほとんどのタスクは、Security Lake コンソールを使用して実行できます。

Security Lake API

Security Lake にプログラムでアクセスするには、Security Lake API を使用し、HTTPS リクエストをサービスに直接発行します。詳細については、「[Security Lake API リファレンス](#)」を参照してください。

AWS Command Line Interface (AWS CLI)

を使用すると AWS CLI、システムのコマンドラインでコマンドを発行して、Security Lake タスクと AWS タスクを実行できます。コマンドラインを使用すると、コンソールを使用するよりも高速で便利になります。コマンドラインツールは、タスクを実行するスクリプトを作成する場合にも便利です。のインストールと使用の詳細については AWS CLI、「」を参照してください [AWS Command Line Interface](#)。

AWS SDKs

AWS は SDKs を提供します。SDKs は、Security Lake やその他の への便利なプログラムによるアクセスを提供します AWS のサービス。SDK は、暗号署名によるリクエスト、エラーの管理、リクエストの自動再試行などのタスクも処理します。AWS SDKs」を参照してください。 [AWS](#)

関連サービス

Security Lake AWS のサービス が使用するその他のものは次のとおりです。

- [Amazon EventBridge](#) — Security Lakeは、オブジェクトがデータレイクに書き込まれると、EventBridge を使用してサブスクライバーに通知します。
- [AWS Glue](#) – Security Lake AWS Glue はクローラを使用して AWS Glue Data Catalog テーブルを作成し、新しく書き込まれたデータをデータカタログに送信します。Security Lake は AWS Lake Formation 、テーブルのパーティションメタデータもデータカタログに保存します。
- [AWS Lake Formation](#)— Security Lakeは、Security Lakeにデータを提供するソースごとに個別のレイクフォーメーションテーブルを作成します。Lake Formation テーブルには、スキーマ、パーティション、データの場所の情報など、各ソースからのデータに関する情報が含まれています。サブスクライバーは、Lake Formation テーブルにクエリを実行してデータを利用することができます。
- [AWS Lambda](#)— Security Lake は Lambda 関数を使用して、生データの抽出、変換、ロード (ETL) ジョブをサポートし、AWS Glueにソース データのパーティションを登録します。
- [Amazon S3](#) — Security Lakeはデータを Amazon S3 オブジェクトとして保存します。ストレージクラスと保存設定は Amazon S3 サービスに基づいています。Security Lake は、Amazon S3 Select をサポートしていません。
- [Amazon Simple Queue Service](#) – Security Lake は Amazon SQS を使用して、イベント駆動型の処理を有効にし、通知を管理します。

Security Lake は、以下に加えて、カスタムソースからデータを収集します AWS のサービス。

- AWS CloudTrail 管理イベントとデータイベント (S3、Lambda)
- Amazon Elastic Kubernetes Service (Amazon EKS) 監査ログ
- Amazon Route 53 Resolver クエリログ
- AWS Security Hub 検出結果
- Amazon Virtual Private Cloud (Amazon VPC) Flow Logs
- AWS WAF v2 ログ

これらのソースの詳細については、「[Security Lake AWS のサービス でのからのデータ収集](#)」を参照してください。OCSF スキーマのデータを読み取ることができるサブスクライバーを作成することで、セキュリティデータレイクの Amazon S3 オブジェクトを使用できます。と統合されている Amazon Athena、Amazon Redshift、およびサードパーティーのサブスクリプションサービスを使用してデータをクエリすることもできます AWS Glue。

概念と用語

このセクションでは、Amazon Security Lake の使用に役立つ主要な概念と用語を説明します。

コントリビューションリージョン

ロールアップリージョンにデータ AWS リージョン を提供する 1 つ以上の。

データレイク

Amazon Simple Storage Service (Amazon S3) に保存され、セキュリティレイクによって管理されます。Security Lake は AWS Glue を使用して、新しく書き込まれたデータをデータカタログに送信します。Security Lake は、データレイクにデータを提供するソースごとに AWS Lake Formation テーブルも作成します。通常、データレイクには以下のデータが保存されます。

- 構造化データと非構造化データ
- 生データと変換されたデータ

Security Lake は、セキュリティ関連のログとイベントを収集するように設計されたデータレイクサービスです。

オープン・サイバーセキュリティ・スキーマ・フレームワーク (OCSF)

セキュリティログとイベント用の標準化された[オープンソーススキーマ](#)。これは、AWS およびさまざまなセキュリティ分野の他のセキュリティ業界のリーダーによって開発されました。Security Lake は、 から収集したログとイベント AWS のサービスを OCSF スキーマに自動的に変換します。カスタムソースはログとイベントを Security Lake に送信する前に OCSF に変換します。

ロールアップリージョン

1 AWS リージョン つ以上の寄与リージョンのセキュリティログとイベントを統合する。1 つ以上のロールアップリージョンを指定すると、地域のコンプライアンス要件に準拠しやすくなります。

ソース

[OCSF](#) の特定のイベントクラスと一致する、1 つのシステムから生成される一連のログとイベント。Security Lake はソースからデータを収集できます。AWS のサービス ソースは別のサービスでもサードパーティのサービスでもかまいません。サードパーティのソースの場合は、Security Lake に送信する前にデータを OCSF スキーマに変換する必要があります。

サブスクライバー

Security Lake からのログとイベントを利用するサービス。サブスクライバーは、別の AWS のサービス または サードパーティーのサービスである場合があります。

Amazon Security Lake の開始方法

このセクションのトピックでは、Security Lake を有効にして使用を開始する方法について説明します。データレイク設定を設定し、ログ収集を設定する方法について説明します。Security Lake を有効にして使用するには、AWS Management Console またはプログラムを使用します。使用するいずれの方法でも、まず AWS アカウント と管理ユーザーを設定する必要があります。それ以降の手順は、アクセス方法によって異なります。

Security Lake コンソールは、開始するための合理化されたプロセスを提供し、データレイクの作成に必要なすべての AWS Identity and Access Management (IAM) ロールを作成します。

Security Lake にプログラムでアクセスする場合、データレイクを設定するには、一部の AWS Identity and Access Management (IAM) ロールを作成する必要があります。

Important

Security Lake は、Security Lake を有効にする前に生成された既存の AWS raw ログソースイベントのバックアップをサポートしていません。

トピック

- [のセットアップ AWS アカウント](#)
- [Security Lake を有効にする際の考慮事項](#)
- [コンソールを使用した Security Lake の有効化](#)
- [Security Lake をプログラムで有効にする](#)

のセットアップ AWS アカウント

Amazon Security Lake を有効にする前に、[が](#)必要です 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 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/signup> 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。セキュリティベストプラクティスとして、ユーザーに管理アクセス権を割り当て、[ルートユーザーアクセスが必要なタスク](#)の実行にはルートユーザーのみを使用するようにしてください。

AWS サインアッププロセスが完了すると、 から確認メールが送信されます。<https://aws.amazon.com/> の [マイアカウント] をクリックして、いつでもアカウントの現在のアクティビティを表示し、アカウントを管理することができます。

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、日常的なタスクにルートユーザーを使用しないように AWS アカウントのルートユーザー、 を保護し AWS IAM Identity Center、 を有効にして管理ユーザーを作成します。

を保護する 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、IAM [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](#) を有効にする」 を参照してください。

管理アクセスを持つユーザーを作成する

1. IAM アイデンティティセンターを有効にします。

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 アイデンティティセンターで、ユーザーに管理アクセスを付与します。

を 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使用する方法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 アイデンティティセンターのユーザーとしてサインインするには、IAM アイデンティティセンターのユーザーの作成時に E メールアドレスに送信されたサインイン URL 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 「アクセスポータルにサインインする」](#) 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 アイデンティティセンターで、最小特権のアクセス許可を適用するというベストプラクティスに従った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「[グループの結合](#)」を参照してください。

Security Lake を有効にするために使用するアカウントを特定してください。

Security Lake はと統合 AWS Organizations され、組織内の複数のアカウントにわたるログ収集を管理します。Organizations に Security Lake を使用する場合は、組織の管理アカウントを使用して委

任された Security Lake 管理者を指定する必要があります。次に、委任された管理者の認証情報を使用して Security Lake を有効にし、メンバーアカウントを追加し、そのメンバーの Security Lake を有効にする必要があります。詳細については、「[Security Lake AWS Organizations でを使用して複数のアカウントを管理する](#)」を参照してください。

また、Organizations の一部ではないスタンドアロンアカウントでは、組織統合なしで Security Lake を使用することもできます。

Security Lake を有効にする際の考慮事項

Security Lake を有効にする前に、以下の点を考慮してください。

- Security Lake はクロスリージョン管理機能を提供します。つまり、データレイクを作成し、AWS リージョン全体でログ収集を構成できます。[サポートされているすべてのリージョンで](#) Security Lake を有効にするには、サポートされている任意のリージョナルエンドポイントを選択できます。[ロールアップリージョン](#)を追加して、複数のリージョンのデータを1つのリージョンに集約することもできます。
- サポートされているすべての AWS リージョンで Security Lake をアクティブ化することをお勧めします。このように設定することで、Security Lake はアクティブに使用されていないリージョンでも、許可されていないアクティビティや異常なアクティビティに関連するデータを収集できます。Security Lake がサポートされているすべてのリージョンでアクティブになっていない場合、Security Lake は複数のリージョンで使用している他のサービスからデータを収集する機能は低下します。
- 任意のリージョンで初めて Security Lake を有効にすると、アカウントに対して以下のサービスにリンクされたロールが作成されます。
 - [AWSServiceRoleForSecurityLake](#): このロールには、AWS のサービス ユーザーに代わって他の呼び出し、セキュリティデータレイクを運用するアクセス許可が含まれています。Security Lake [を委任された Security Lake 管理者](#)として有効にすると、Security Lake は組織内の各メンバーアカウントに[サービスリンクロール](#)を作成します。
 - [AWSServiceRoleForSecurityLakeResourceManagement](#): Security Lake はこのロールを使用して継続的なモニタリングとパフォーマンスの向上を実行し、レイテンシーとコストを削減する可能性があります。このサービスリンクロールは、ロールを引き受ける上で `resource-management.securitylake.amazonaws.com` サービスを信頼します。このサービスロールを有効にすると、Lake Formation へのアクセスも許可されます。

これが 2025 年 4 月 17 日より前に Security Lake を有効にした既存のアカウントにどのように影響するかについては、「」を参照してください[Update for existing accounts](#)。

サービスにリンクされたロールの仕組みについては、IAM [ユーザーガイドの「サービスにリンクされたロールのアクセス許可の使用」](#)を参照してください。

- Security Lake は Amazon S3 Object Lock をサポートしていません。データレイクバケットが作成されると、S3 Object Lock はデフォルトで無効になります。バケットで Object Lock を有効にすると、データレイクへの正規化されたログデータの配信が中断されます。
- リージョンで Security Lake を再度有効にする場合は、以前の Security Lake の使用からリージョンの対応する AWS Glue データベースを削除する必要があります。

コンソールを使用した Security Lake の有効化

このチュートリアルでは、を使用して Security Lake を有効にして設定する方法について説明します AWS Management Console。の一部として AWS Management Console、Security Lake コンソールは開始するための合理化されたプロセスを提供し、データレイクの作成に必要なすべての AWS Identity and Access Management (IAM) ロールを作成します。

ステップ 1: ソースを設定する

Security Lake は、さまざまなソースから、AWS アカウント および AWS リージョン全体からログとイベントデータを収集します。以下の手順に従って、Security Lake に収集させたいデータを特定してください。これらの手順は、ネイティブにサポートされている AWS のサービスをソースとして追加する場合にのみ使用できます。カスタムソースの追加については、[Security Lake のカスタムソースからデータを収集する](#)を参照してください。

ログソースコレクションを設定するには

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ページの右上隅にある AWS リージョン セレクターを使用して、リージョンを選択します。Security Lake は、オンボーディング中に現在のリージョンと他のリージョンで有効にできます。
3. [開始する] を選択します。
4. ログソースとイベントソースの選択 では、ソース選択に次のいずれかのオプションを選択します。
 - a. デフォルトの AWS ソースの取り込み – 推奨オプションを選択すると、CloudTrail - S3 データイベント および AWS WAF はデフォルトでは取り込みに含まれません。これは、両方のソースタイプを大量に取り込むと、使用コストに大きな影響を与える可能性があるため

す。これらのソースを取り込むには、まず特定の AWS ソースの取り込みオプションを選択し、次にログおよびイベントソースリストからこれらのソースを選択します。

- b. 特定の AWS ソースを取り込む – このオプションでは、取り込む 1 つ以上のログソースとイベントソースを選択できます。

Note

アカウントで Security Lake を初めて有効にすると、選択したすべてのログソースとイベントソースが 15 日間の無料試用期間に含まれます。使用情報の詳細については、「[使用状況と推定コストの確認](#)」を参照してください。

5. バージョンで、ログソースとイベントソースを取り込むデータソースのバージョンを選択します。バージョンの詳細については、「[OCSF ソースの識別](#)」を参照してください。

Important

指定されたリージョンで新しいバージョンの AWS ログソースを有効にするために必要なロールアクセス許可がない場合は、Security Lake 管理者にお問い合わせください。詳細については、「[ロールのアクセス許可の更新](#)」を参照してください。

6. [リージョンの選択] では、サポートされているすべてのリージョンからログとイベントソースを取り込むか、特定のリージョンから取り込むかを選択します。[特定のリージョン] を選択した場合は、データを取り込む地域を選択します。
7. Select アカウントの場合は、次の手順を実行します。
 1. Security Lake が組織内のすべてのアカウントまたは特定のアカウントからデータを取り込むかどうかを選択します。これらのアカウントでは、この設定中に選択した設定で Security Lake が有効になります。
 2. 新しい組織アカウントの Security Lake を自動的に有効にするチェックボックスがデフォルトで選択されています。これらの自動有効化設定は、組織に参加する AWS アカウントときに適用されます。自動有効化設定はいつでも編集できます。

Note

自動有効化設定は、既存のアカウントではなく、組織に参加するアカウントにのみ適用されます。詳細については、「[コンソールでの新しいアカウント設定の編集](#)」を参照してください。

8. サービスアクセスには、新しい IAM ロールを作成するか、ソースからデータを収集してデータレイクに追加する権限を Security Lake に付与する既存の IAM ロールを使用します。Security Lake を有効にしたすべてのリージョンで 1 つのロールが使用されます。
9. [次へ] を選択します。

ステップ 2: ストレージ設定とロールアップリージョンを定義する (オプション)

Security Lake にデータを保存する Amazon S3 ストレージクラスとその期間を指定できます。ロールアップリージョンを指定して、複数のリージョンのデータを統合することもできます。これらはオプションのステップです。詳細については、「[Security Lakeのライフサイクル管理](#)」を参照してください。

ストレージとロールアップの設定を行うには

1. 複数の対象リージョンのデータを 1 つのロールアップリージョンに統合する場合は、[ロールアップリージョンの選択] で [ロールアップリージョンの追加] を選択します。ロールアップリージョンとそれに寄与するリージョンを指定します。1 つ以上のロールアップリージョンを設定できます。
2. [ストレージクラスを選択] では、Amazon S3 ストレージクラスを選択します。デフォルトのストレージクラスは、S3 Standardです。それ以降にデータを別のストレージクラスに移行する場合は保持期間 (日単位) を指定し、[Add transition] を選択します。保持期間が終了すると、オブジェクトは期限切れになり、Amazon S3 はそれらを削除します。Amazon S3 ストレージクラスと保持の詳細については、「[保持管理](#)」を参照してください。
3. 最初のステップでサービスアクセス用にロールアップリージョンを選択した場合は、新しい IAM ロールを作成するか、Security Lake に複数のリージョンにデータを複製する権限を付与する既存の IAM ロールを使用してください。
4. [次へ] を選択します。

ステップ 3: データレイクを確認して作成する

Security Lake がデータを収集するソース、ロールアップ地域、および保持設定を確認してください。次に、データレイクを作成します。

データレイクを確認して作成するには

1. Security Lake を有効にする際には、ログとイベントのソース、リージョン、ロールアップリージョン、ストレージクラスを確認してください。
2. [作成] を選択します。

データレイクを作成すると、Security Lake コンソールに Summary ページが表示されます。このページでは、リージョンとロールアップリージョンの数、サブスクライバーに関する情報、および問題の概要を示します。

問題メニューには、Security Lake サービスまたは Amazon S3 バケットに影響を与えている過去 14 日間の問題の概要が表示されます。各問題の詳細については、Security Lake コンソールの問題ページを参照してください。

ステップ 4: 独自のデータを表示してクエリする

データレイクを作成したら、Amazon Athena または同様のサービスを使用して、AWS Lake Formation データベースやテーブルからデータを表示およびクエリできます。コンソールを使用すると、Security Lake を有効にするために使用するロールに、Security Lake によってデータベース表示アクセス許可が自動的に付与されます。ロールには少なくとも Data Analyst 権限が必要です。権限レベルの詳細については、「[Lake Formation ペルソナ](#)」と「[IAM 権限リファレンス](#)」を参照してください。SELECT 権限を付与する手順については、『AWS Lake Formation 開発者ガイド』の「[名前付きリソースメソッドを使用した Data Catalog 権限の付与](#)」を参照してください。

ステップ 5: サブスクライバーを作成する

データレイクを作成したら、データを使用するサブスクライバーを追加できます。サブスクライバーは、Amazon S3 バケット内のオブジェクトに直接アクセスするか、データレイクにクエリを実行することでデータを使用できます。サブスクライバーの詳細については、「[Security Lake でのサブスクライバー管理](#)」を参照してください。

Security Lake をプログラムで有効にする

このチュートリアルでは、Security Lake をプログラムで有効にして使用を開始する方法について説明します。Amazon Security Lake API は、Security Lake アカウント、データ、リソースへの包括的なプログラムによるアクセスを提供します。または、[AWS Command Line Interface](#)または [AWS Tools for PowerShell](#) などの AWS コマンドラインツール、または [AWS SDKs](#) を使用して Security Lake にアクセスすることもできます。

ステップ 1: IAM ロールを作成する

Security Lake にプログラムでアクセスする場合、データレイクを設定するには、一部の AWS Identity and Access Management (IAM) ロールを作成する必要があります。

Important

Security Lake コンソールを使用して Security Lake を有効にして設定する場合は、これらの IAM ロールを作成する必要はありません。

以下のアクションを 1 つ以上実行する場合は、IAM でロールを作成する必要があります (リンクを選択すると、各アクションの IAM ロールに関する詳細が表示されます)。

- [カスタムソースの作成](#) – カスタム ソースは、Security Lake にデータを送信する、ネイティブにサポートされている AWS のサービス 以外のソースです。
- [データアクセス権限を持つサブスクライバーの作成](#) — 権限を持つサブスクライバーは、データレイクから S3 オブジェクトに直接アクセスできます。
- [クエリアクセス権を持つサブスクライバーの作成](#) — 権限を持つサブスクライバーは、Amazon Athenaなどのサービスを使用して Security Lake からデータをクエリできます。
- [ロールアップ リージョンの構成](#) – ロールアップ リージョンは、複数の AWS リージョンからのデータを統合します。

前述のロールを作成したら、Security Lake を有効にするために使用しているロールに [AmazonSecurityLakeAdministrator](#) AWS 管理ポリシーをアタッチします。このポリシーにより、プリンシパルが Security Lake にオンボーディングし、Security Lake のすべてのアクションにアクセスすることが許可される、管理者許可が付与されます。

[AmazonSecurityLakeMetaStoreManager](#) AWS マネージドポリシーをアタッチして、Security Lake からデータレイクまたはクエリデータを作成します。このポリシーは、Security Lake がソースから

受信する raw ログおよびイベントデータに対する抽出、変換、ロード (ETL) ジョブをサポートするために必要です。

ステップ 2: Amazon Security Lake を有効にする

Security Lake をプログラムで有効にするには、Security Lake API の [CreateDataLake](#) オペレーションを使用します。を使用している場合は AWS CLI、[create-data-lake](#) コマンドを実行します。リクエストでは、configurations オブジェクトの region フィールドを使用して、Security Lake を有効にするリージョンのリージョンコードを指定します。リージョンコードのリストについては、「AWS 全般のリファレンス」の「[Amazon Security Lake エンドポイント](#)」を参照してください。

例 1

次のコマンド例では、us-east-1 および us-east-2 リージョンで Security Lake を有効にします。どちらのリージョンでも、このデータレイクは Amazon S3 マネージドキーで暗号化されます。オブジェクトは 365 日後に期限切れになり、オブジェクトは 60 日後に ONEZONE_IA S3 ストレージクラスに移行します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake create-data-lake \  
--configurations '[{"encryptionConfiguration":  
  {"kmsKeyId": "S3_MANAGED_KEY"}, "region": "us-east-1", "lifecycleConfiguration":  
  {"expiration": {"days": 365}, "transitions": [{"days": 60, "storageClass": "ONEZONE_IA"}]},  
  {"encryptionConfiguration": {"kmsKeyId": "S3_MANAGED_KEY"}, "region": "us-  
east-2", "lifecycleConfiguration": {"expiration": {"days": 365}, "transitions":  
[{"days": 60, "storageClass": "ONEZONE_IA"}]}]\' \  
--meta-store-manager-role-arn "arn:aws:iam:us-east-1:123456789012:role/service-role/  
AmazonSecurityLakeMetaStoreManager"
```

例 2

次のコマンド例では、us-east-2 リージョンで Security Lake を有効にします。このデータレイクは、AWS Key Management Service () で作成されたカスタマーマネージドキーで暗号化されます AWS KMS。オブジェクトは 500 日後に期限切れになり、オブジェクトは 30 日後に GLACIER S3 ストレージクラスに移行します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake create-data-lake \  
--configurations '[{"encryptionConfiguration":  
  {"kmsKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"}, "region": "us-
```

```
east-2", "lifecycleConfiguration": {"expiration":{"days":500}, "transitions": [{"days":30, "storageClass":"GLACIER"}]}] \n --meta-store-manager-role-arn "arn:aws:iam:us-east-1:123456789012:role/service-role/AmazonSecurityLakeMetaStoreManager"
```

Note

Security Lake を既に有効にしている、リージョンまたはソースの設定を更新する場合は、[UpdateDataLake](#) オペレーションを使用するか AWS CLI、を使用する場合は [update-data-lake](#) コマンドを使用します。CreateDataLake オペレーションを使用しないでください。

ステップ 3: ソースを設定する

Security Lake は、さまざまなソースから、AWS アカウント および AWS リージョン全体からログとイベントデータを収集します。以下の手順に従って、Security Lake に収集させたいデータを特定してください。これらの手順は、ネイティブにサポートされている AWS のサービスをソースとして追加する場合にのみ使用できます。カスタムソースの追加については、[Security Lake のカスタムソースからデータを収集する](#) を参照してください。

1 つ以上のコレクションソースをプログラムで定義するには、Security Lake API の [CreateAWSLogSource](#) オペレーションを使用してください。ソースごとに、sourceName パラメータに地域固有の値を指定します。オプションで追加のパラメーターを使用して、ソースの範囲を特定のアカウント (accounts) または特定のバージョン (sourceVersion) に制限します。

Note

リクエストにオプションのパラメータを含めない場合、Security Lake は、除外するパラメータに応じて、指定されたソースのすべてのアカウントまたはすべてのバージョンにリクエストを適用します。たとえば、ある組織の委任された Security Lake 管理者が accounts パラメータを除外した場合、Security Lake はリクエストを組織内のすべてのアカウントに適用します。同様に、sourceVersion パラメータを除外すると、Security Lake は指定されたソースのすべてのバージョンにリクエストを適用します。

Security Lake を有効にしていないリージョンをリクエストで指定すると、エラーが発生します。このエラーに対処するには、regions アレイに Security Lake を有効にしたリージョンのみを指定する

ようにしてください。または、リージョンで Security Lake を有効にしてからリクエストを再度送信することもできます。

アカウントで Security Lake を初めて有効にすると、選択したすべてのログソースとイベントソースが 15 日間の無料試用期間に含まれます。使用情報の詳細については、「[使用状況と推定コストの確認](#)」を参照してください。

ステップ 4: ストレージ設定とロールアップリージョンを設定する (オプション)

Security Lake にデータを保存する Amazon S3 ストレージクラスとその期間を指定できます。ロールアップリージョンを指定して、複数のリージョンのデータを統合することもできます。これらはオプションのステップです。詳細については、「[Security Lake のライフサイクル管理](#)」を参照してください。

Security Lake を有効にするときにプログラムでターゲット目標を定義するには、Security Lake API の [CreateDataLake](#) オペレーションを使用します。Security Lake を既に有効にしている、ターゲット目標を定義する場合は、[UpdateDataLake](#) オペレーションではなく [CreateDataLake](#) オペレーションを使用します。

いずれの操作でも、サポートされているパラメータを使用して必要な設定を指定します。

- ロールアップリージョンを指定するには、`region` フィールドを使用して、ロールアップリージョンにデータを提供するリージョンを指定します。`replicationConfiguration` オブジェクトの `regions` 配列で、各ロールアップリージョンのリージョンコードを指定します。リージョンコードのリストについては、「AWS 全般のリファレンス」の「[Amazon Security Lake エンドポイント](#)」を参照してください。
- データの保存設定を指定するには、`lifecycleConfiguration` パラメータを使用します。
 - `transitions` には、特定の Amazon S3 ストレージクラス (`storageClass`) に S3 オブジェクトを保存する合計日数 (`days`) を指定します。
 - `expiration` には、オブジェクトが作成されてから、任意のストレージクラスを使用して Amazon S3 にオブジェクトを保存する合計日数を指定します。この保持期間が終了すると、オブジェクトは期限切れになり、Amazon S3 はそれらを削除します。

Security Lake は、指定された保持設定を `configurations` オブジェクトの `region` フィールドで指定したリージョンに適用します。

たとえば、次のコマンドは、ロールアップリージョンap-northeast-2としてを使用してデータレイクを作成します。us-east-1リージョンは、ap-northeast-2リージョンにデータを提供します。この例では、データレイクに追加されたオブジェクトの有効期限を10日間設定します。

```
$ aws securitylake create-data-lake \  
--configurations '[{"encryptionConfiguration":  
  {"kmsKeyId": "S3_MANAGED_KEY", "region": "us-east-1", "replicationConfiguration":  
  {"regions": [ap-northeast-2], "roleArn": "arn:aws:iam::123456789012:role/service-  
role/AmazonSecurityLakeS3ReplicationRole"}, "lifecycleConfiguration": {"expiration":  
  {"days": 10}}}]' \  
--meta-store-manager-role-arn "arn:aws:iam::123456789012:role/service-role/  
AmazonSecurityLakeMetaStoreManager"
```

これで、データレイクが作成されました。Security Lake API の [ListDataLakes](#) オペレーションを使用して、各リージョンで Security Lake とデータレイク設定が有効になっていることを確認します。

データレイクの作成中に問題やエラーが発生した場合は、[ListDataLakeExceptions](#) オペレーションを使用して例外のリストを表示し、[CreateDataLakeExceptionSubscription](#) オペレーションで例外をユーザーに通知できます。詳細については、「[データレイクのステータスのトラブルシューティング](#)」を参照してください。

ステップ 5: 独自のデータを表示してクエリする

データレイクを作成したら、Amazon Athena または同様のサービスを使用して、AWS Lake Formation データベースやテーブルからデータを表示およびクエリできます。Security Lake をプログラムで有効にすると、データベースビューのアクセス許可は自動的に付与されません。のデータレイク管理者アカウントは、関連するデータベースとテーブルのクエリに使用する IAM ロールにアクセス SELECT 許可を付与 AWS Lake Formation する必要があります。ロールには少なくとも Data Analyst 権限が必要です。権限レベルの詳細については、「[Lake Formation ペルソナ](#)」と「[IAM 権限リファレンス](#)」を参照してください。SELECT 権限を付与する手順については、『AWS Lake Formation 開発者ガイド』の「[名前付きリソースメソッドを使用した Data Catalog 権限の付与](#)」を参照してください。

ステップ 6: サブスクライバーを作成する

データレイクを作成したら、データを使用するサブスクライバーを追加できます。サブスクライバーは、Amazon S3 バケット内のオブジェクトに直接アクセスするか、データレイクにクエリを実行することでデータを使用できます。サブスクライバーの詳細については、「[Security Lake でのサブスクライバー管理](#)」を参照してください。

Security Lake AWS Organizations で を使用して複数のアカウントを管理する

Amazon Security Lake を使用して、複数の AWS アカウントからセキュリティログとイベントを収集できます。複数のアカウントの管理を自動化および合理化するには、Security Lake を [AWS Organizations](#) と統合することを強くお勧めします。

組織では、組織の作成に使用するアカウントは管理アカウントと呼ばれます。Security Lake を Organizations と統合するには、管理アカウントが組織の委任された Security Lake 管理者アカウントを指定する必要があります。

委任された Security Lake 管理者は、Security Lake を有効にし、メンバー アカウントの Security Lake 設定を構成できます。委任管理者は、Security Lake が有効になってい AWS リージョン すべて (現在使用しているリージョンエンドポイントに関係なく) で、組織全体のログとイベントを収集できます。委任管理者は、新しい組織アカウントのログとイベントデータを自動的に収集するように Security Lake を設定することもできます。

委任された Security Lake 管理者は、関連付けられたメンバー アカウントのログおよびイベントデータにアクセスできます。したがって、関連するメンバー アカウントが所有するデータを収集するように Security Lake を構成できます。また、関連付けられたメンバー アカウントが所有するデータを使用する権限をサブスクライバーに付与することもできます。

組織内の複数のアカウントで Security Lake を有効にするには、まず組織の管理アカウントが組織の委任された Security Lake 管理者アカウントを指定する必要があります。これで委任された管理者は、組織の Security Lake を有効化して設定できます。

Important

Security Lake の [RegisterDataLakeDelegatedAdministrator](#) API を使用して、Security Lake に組織へのアクセスを許可し、Organizations の委任された管理者を登録します。

Organizations の API を使用して委任管理者を登録すると、Organizations のサービスにリンクされたロールが正常に作成されない可能性があります。完全な機能を確認するには、Security Lake APIs を使用します。

Organizations のセットアップについては、「AWS Organizations ユーザーガイド」の「[組織の作成と管理](#)」を参照してください。

❶ 既存の Security Lake アカウントの場合

2025 年 4 月 17 日より前に Security Lake を有効にした場合は、[を有効にすることをお勧めします](#) [リソース管理のためのサービスリンクロール \(SLR\) アクセス許可](#)。この SLR を使用すると、継続的なモニタリングとパフォーマンスの向上を継続できるため、レイテンシーとコストを削減できる可能性があります。この SLR に関連付けられたアクセス許可の詳細については、「」を参照してください [リソース管理のためのサービスリンクロール \(SLR\) アクセス許可](#)。

Security Lake コンソールを使用する場合、[を有効にするよう求める通知](#)が送信されます `AWSServiceRoleForSecurityLakeResourceManagement`。を使用する場合は AWS CLI、[「Security Lake サービスにリンクされたロールの作成」](#)を参照してください。

委任された Security Lake 管理者に関する重要な考慮事項

Security Lake で委任された管理者がどのように動作するかを定義する次の要素に注意してください。

委任管理者はすべてのリージョンで同一です。

委任管理者を作成すると、その委任管理者が Security Lake を有効にするすべてのリージョンの委任管理者になります。

ログアーカイブアカウントを Security Lake 委任管理者として設定することをお勧めします。

ログアーカイブアカウントは AWS アカウント、すべてのセキュリティ関連ログの取り込みとアーカイブ専用です。通常、このアカウントへのアクセスは、コンプライアンス調査を行う監査人やセキュリティチームなど、少数のユーザーに限定されます。Log Archive アカウントを Security Lake の委任管理者として設定して、コンテキストの切り替えを最小限に抑えてセキュリティ関連のログとイベントを表示できるようにすることをお勧めします。

また、Log Archive アカウントに直接アクセスできるのは最小限のユーザーのみにすることをお勧めします。この選択グループ以外で、Security Lake が収集するデータにユーザーがアクセスする必要がある場合は、そのユーザーを Security Lake サブスクライバーとして追加できます。サブスクライバーを追加する方法については、「[Security Lake でのサブスクライバー管理](#)」を参照してください。

AWS Control Tower サービスを使用しない場合、ログアーカイブアカウントがない可能性があります。Log Archive アカウントについて詳しくは、セキュリティリファレンスアーキテクチャの「[Security OU — Log Archive アカウント](#)」を参照してください。AWS

組織は、委任された管理者を 1 名だけ持つことができます。

Security Lake 管理者は、組織あたり 1 名のみです。

組織管理アカウントを代理管理者にすることはできません。

AWS セキュリティのベストプラクティスと最小特権の原則に基づいて、組織管理アカウントを委任管理者にすることはできません。

委任された管理者は、アクティブな組織に属している必要があります。

組織を削除すると、委任された管理者アカウントは Security Lake を管理できなくなります。別の組織の委任管理者を指定するか、組織の一部ではないスタンドアロンアカウントで Security Lake を使用する必要があります。

委任された管理者を指定するには IAM 許可が必要です

委任 Security Lake 管理者を指定する場合、Security Lake を有効にし、次のポリシーステートメントに記載されている特定の AWS Organizations API オペレーションを使用するアクセス許可が必要です。

AWS Identity and Access Management (IAM) ポリシーの末尾に次のステートメントを追加して、これらのアクセス許可を付与できます。

```
{
  "Sid": "Grant permissions to designate a delegated Security Lake administrator",
  "Effect": "Allow",
  "Action": [
    "securitylake:RegisterDataLakeDelegatedAdministrator",
    "organizations:EnableAWSServiceAccess",
    "organizations:RegisterDelegatedAdministrator",
    "organizations:ListAccounts",
    "organizations:ListDelegatedAdministrators",
    "organizations:ListAWSServiceAccessForOrganization",
    "organizations:DescribeOrganizationalUnit",
    "organizations:DescribeAccount",
    "organizations:DescribeOrganization"
  ],
  "Resource": "*"
}
```

委任された Security Lake 管理者を指定し、メンバーアカウントを追加します。

アクセス方法を選択して、組織の委任された Security Lake 管理者アカウントを指定します。組織管理アカウントのみが、組織の委任された管理者アカウントを指定できます。組織の管理アカウントを組織の委任管理者アカウントにすることはできません。

Note

- 組織管理アカウントは、Security Lake の RegisterDataLakeDelegatedAdministrator オペレーションを使用して、委任された Security Hub 管理者アカウントを指定する必要があります。Organizations を通じて委任された Security Lake 管理者の指定はサポートされていません。
- 組織の委任された管理者を変更する場合は、まず 現在の委任された管理者を削除する 必要があります。その後、新しく委任された管理者を指定できます。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
組織の管理アカウントの認証情報を使用してサインインします。
2.
 - Security Lake がまだ有効になっていない場合は、「はじめに」を選択し、「Security Lake を有効にする」ページで Security Lake の委任管理者を指定します。
 - Security Lake がすでに有効になっている場合は、設定ページで委任された Security Lake 管理者を指定します。
3. 別のアカウントに管理を委任で、他の AWS セキュリティサービスの委任管理者として既に機能しているアカウントを選択します (推奨)。または、委任 Security Lake 管理者として指定するアカウントの 12 桁の AWS アカウント ID を入力します。
4. [委任] を選択します。Security Lake がまだ有効になっていない場合は、委任された管理者を指定すると、現在のリージョンでそのアカウントに対して Security Lake が有効になります。

API

プログラムで委任管理者を指定するには、Security Lake API の [RegisterDataLakeDelegatedAdministrator](#) オペレーションを使用します。組織管理アカウントからオペレーションを呼び出す必要があります。を使用している場合は AWS CLI、組織管理アカウントから [register-data-lake-delegated-administrator](#) コマンドを実行します。リクエストで、`accountId` パラメータを使用して、組織の委任管理者アカウントとして AWS アカウント 指定する の 12 桁のアカウント ID を指定します。

たとえば、次の AWS CLI コマンドは委任された管理者を指定します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake register-data-lake-delegated-administrator \  
--account-id 123456789012
```

委任管理者は、新しい組織アカウントの AWS ログとイベントデータの収集を自動化することもできます。この設定では、アカウントが組織に追加されると、新しいアカウントで Security Lake が自動的に有効になります AWS Organizations。委任管理者は、Security Lake API の [CreateDataLakeOrganizationConfiguration](#) オペレーションを使用するか、AWS CLI を使用している場合は [create-data-lake-organization-configuration](#) コマンドを実行して、この設定を有効にできます。リクエストでは、新しいアカウントの特定の設定を指定することもできます。

たとえば、次の AWS CLI コマンドは、新しい組織アカウントで Security Lake と Amazon Route 53 リゾルバークエリログ、AWS Security Hub 検出結果、Amazon Virtual Private Cloud (Amazon VPC) フローログの収集を自動的に有効にします。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake create-data-lake-organization-configuration \  
--auto-enable-new-account '[{"region": "us-east-1", "sources":  
[{"sourceName": "ROUTE53"}, {"sourceName": "SH_FINDINGS"}, {"sourceName": "VPC_FLOW"}]]'
```

組織管理アカウントが委任された管理者を指定すると、管理者は組織に対して Security Lake を有効にして構成できるようになります。これには、組織内の個々のアカウントの AWS ログとイベントデータを収集するための Security Lake の有効化と設定が含まれます。詳細については、「[Security Lake AWS のサービス での データ収集](#)」を参照してください。

[GetDataLakeOrganizationConfiguration](#) オペレーションを使用して、新しいメンバーアカウントの組織の現在の設定に関する詳細を取得できます。

新しい組織アカウントの自動有効化設定の編集

委任 Security Lake 管理者は、アカウントが組織に参加するときに、アカウントの自動有効化設定を表示および編集できます。Security Lake は、既存のアカウントではなく、新しいアカウントに対してのみ、これらの設定に基づいてデータを取り込みます。

次の手順を使用して、新しい組織アカウントの設定を編集します。

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
3. アカウントページで、新しいアカウント設定セクションを展開します。各リージョンからどの Sources Security Lake が取り込むかを表示できます。
4. **編集** を選択して、この設定を編集します。
5. 新しいアカウント設定の編集ページで、次の手順を実行します。
 - a. **Select Regions** で、データを取り込むソースを更新するリージョンを 1 つ以上選択します。[次へ] を選択します。
 - b. **ソースの選択** では、ソースの選択に次のいずれかのオプションを選択します。
 - i. デフォルトの AWS ソースの取り込み – 推奨オプションを選択すると、CloudTrail - S3 データイベント および AWS WAF はデフォルトでは取り込みに含まれません。これは、両方のソースタイプを大量に取り込むと、使用コストに大きな影響を与える可能性があるためです。これらのソースを取り込むには、まず特定の AWS ソースの取り込みオプションを選択し、次にログとイベントソースリストからこれらのソースを選択します。
 - ii. 特定の AWS ソースを取り込む – このオプションでは、取り込む 1 つ以上のログソースとイベントソースを選択できます。
 - iii. ソースを取り込まない – 前のステップで選択したリージョンからソースを取り込まない場合は、このオプションを選択します。
 - iv. [次へ] を選択します。

Note

アカウントで Security Lake を初めて有効にすると、選択したすべてのログソースとイベントソースが 15 日間の無料試用期間に含まれます。使用情報の詳細については、「[使用状況と推定コストの確認](#)」を参照してください。

- c. 変更を確認したら、適用を選択します。

が組織 AWS アカウントに参加すると、これらの設定はデフォルトでそのアカウントに適用されます。

委任された Security Lake 管理者を削除する。

組織管理アカウントのみが、組織の委任された Security Lake 管理者を削除できます。組織の委任管理者を変更する場合は、現在の委任管理者を削除し、新しい委任管理者を指定します。

Important

委任された Security Lake 管理者を削除すると、データレイクが削除され、組織内のアカウントの Security Lake が無効になります。

Security Lake コンソールを使用して委任された管理者を変更または削除することはできません。これらのタスクはプログラムでのみ実行できます。

委任された管理者をプログラムで削除するには、Security Lake API の [DeregisterDataLakeDelegatedAdministrator](#) オペレーションを使用します。組織管理アカウントからオペレーションを呼び出す必要があります。を使用している場合は AWS CLI、組織管理アカウントから [deregister-data-lake-delegated-administrator](#) コマンドを実行します。

たとえば、次の AWS CLI コマンドは、委任 Security Lake 管理者を削除します。

```
$ aws securitylake deregister-data-lake-delegated-administrator
```

委任管理者の指定を保持しながら、新しいメンバーアカウントの自動設定を変更するには、Security Lake API の [DeleteDataLakeOrganizationConfiguration](#) オペレーションを使用するか、を使用してい

る場合は [AWS CLI delete-data-lake-organization-configuration](#) コマンドを使用します。委任された管理者のみが、組織のこれらの設定を変更できます。

たとえば、次の AWS CLI コマンドは、組織に参加する新しいメンバーアカウントからの Security Hub の検出結果の自動収集を停止します。委任管理者がこのオペレーションを呼び出した後、新しいメンバーアカウントは Security Hub の検出結果をデータレイクに提供しません。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックslash (\) の行継続文字を使用しています。

```
$ aws securitylake delete-data-lake-organization-configuration \
--auto-enable-new-account '[{"region":"us-east-1","sources":
[{"sourceName":"SH_FINDINGS"}]]'
```

Security Lake の信頼できるアクセス

組織の Security Lake を設定すると、AWS Organizations 管理アカウントは Security Lake との信頼されたアクセスを有効にできます。信頼されたアクセスを利用すると、Security Lake は IAM サービスにリンクされたロールを作成し、組織およびそのアカウントでタスクを実行することを代理で実行できます。詳細については、AWS Organizations ユーザーガイドの「[他の AWS のサービスで AWS Organizations を利用する](#)」を参照してください。

組織管理アカウントのユーザーは、AWS Organizations の Security Lake に対する信頼されたアクセスを無効にすることができます。信頼できるアクセスを無効にする手順については、『AWS Organizations ユーザーガイド』の「[信頼できるアクセスを有効または無効にする方法](#)」を参照してください。

委任された管理者の AWS アカウント が停止、分離、または閉鎖されている場合は、信頼されたアクセスを無効にすることをお勧めします。

Security Lake でのリージョンの管理

Amazon Security Lake は、サービスを有効にした AWS リージョン 全体でセキュリティログとイベントを収集できます。リージョンごとに、データは異なる Amazon S3 バケットに保存されます。リージョンごとに異なるデータレイク設定 (たとえば、異なるソースと保持設定) を指定できます。1 つ以上のロールアップリージョンを定義して、複数のリージョンのデータを統合することもできます。

リージョン・ステータスのチェック

Security Lakeは複数の AWS リージョンを収集できます。データレイクの状態を追跡するには、各リージョンが現在どのように設定されているかを把握しておくが便利です。希望するアクセス方法を選択し、次の手順に従ってリージョンの現在のステータスを取得します。

Console

リージョンのステータスを確認するには

1. Security Lake コンソール<https://console.aws.amazon.com/securitylake/>を開きます。
2. ナビゲーションペインで [リージョン] を選択します。リージョンページが開き、Security Lakeが現在有効になっているリージョンの概要が表示されます。
3. リージョンを選択し、[編集] を選択すると、そのリージョンの詳細が表示されます。

API

現在のリージョンのログ収集のステータスを取得するには、Security Lake API の [GetDataLakeSources](#) オペレーションを使用します。を使用している場合は AWS CLI、[get-data-lake-sources](#) コマンドを実行します。accounts パラメータには、リストとして 1 つ以上の AWS アカウント IDs を指定します。リクエストが成功すると、Security Lake は、Security Lake がデータを収集している AWS ソースや各ソースのステータスなど、現在のリージョン内のアカウントのスナップショットを返します。accounts パラメータを含めない場合、レスポンスには、Security Lake が現在のリージョンで設定されているすべてのアカウントのログ収集のステータスが含まれます。

例えば、次の AWS CLI コマンドは、現在のリージョンで指定されたアカウントのログ収集ステータスを取得します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake get-data-lake-sources \  
--accounts "123456789012" "111122223333"
```

次の AWS CLI コマンドは、指定されたリージョン内のすべてのアカウントと有効なソースのログ収集ステータスを一覧表示します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake get-data-lake-sources \  
--regions "us-east-1" \  
--query 'dataLakeSources[].[account,sourceName]'
```

リージョンで Security Lake を有効にしているかどうかを確認するには、[ListDataLakes](#) オペレーションを使用します。を使用している場合は AWS CLI、[list-data-lakes](#) コマンドを実行します。regions パラメーターには、リージョンのリージョンコードを指定します。たとえば、米国東部 (バージニア北部) リージョンの場合は us-east-1 です。リージョンコードのリストについては、AWS 全般のリファレンスの「[Amazon Security Lake エンドポイント](#)」を参照してください。ListDataLakes オペレーションは、リクエストで指定した各リージョンのデータレイク設定を返します。リージョンを指定しない場合、Security Lake は、Security Lake が利用可能な各リージョンのデータレイクのステータスと構成設定を返します。

例えば、次の AWS CLI コマンドは、eu-central-1 リージョンのデータレイクのステータスと設定を表示します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake list-data-lakes \  
--regions "us-east-1" "eu-central-1"
```

リージョン設定の変更

好みの方法を選択し、次の手順に従って 1 つ以上の AWS リージョンのデータレイクの設定を更新します。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ナビゲーションペインで [リージョン] を選択します。

3. リージョンを選択し、[編集] を選択します。
4. <リージョン>のすべてのアカウントのソースを上書きするチェック ボックスをオンにして、ここでの選択がこのリージョンの前の選択をオーバーライドすることを確認します。
5. [ストレージクラスを選択] で [トランジションを追加] を選択し、データ用の新しいストレージクラスを追加します。
6. [タグ] には、必要に応じてリージョンのタグを割り当てたり編集したりします。タグは、特定のリージョン AWS アカウント の のデータレイク設定など、特定のタイプの AWS リソースを定義して割り当てることができるラベルです。詳細については、「[Security Lake リソースのタグ付け](#)」を参照してください。
7. リージョンをロールアップ リージョンに変更するには、ナビゲーション ペインで [ロールアップ リージョン] ([設定] の下) を選択します。[Modify (修正)] を選択します。「ロールアップリージョンの選択」セクションで、「ロールアップリージョンを追加」を選択します。関係するリージョンを選択し、Security Lake に複数のリージョンにデータを複製する権限を付与します。完了したら、[Save] を選択して、変更を保存します。

API

データレイクのリージョン設定をプログラムで更新するには、Security Lake API の [UpdateDataLake](#) オペレーションを使用します。を使用している場合は AWS CLI、[update-data-lake](#) コマンドを実行します。region パラメーターには、設定を変更するリージョンのリージョンコードを指定します。たとえば、米国東部 (バージニア北部) リージョンの場合は us-east-1 です。リージョンコードのリストについては、AWS 全般のリファレンスの「[Amazon Security Lake エンドポイント](#)」を参照してください。

追加のパラメータを使用して、変更する設定ごとに新しい値を指定します。たとえば、暗号化キー (encryptionConfiguration) や保存設定 (lifecycleConfiguration) です。

たとえば、次の AWS CLI コマンドは、us-east-1リージョンのデータの有効期限とストレージクラスの移行設定を更新します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ update-data-lake \  
--configurations '[{"region": "us-east-1", "lifecycleConfiguration": {"expiration": {"days": 500}, "transitions": [{"days": 45, "storageClass": "ONEZONE_IA"}]}'
```

Security Lake でのロールアップリージョンの設定

ロールアップリージョンは、1 つ以上の寄与リージョンのデータを統合します。ロールアップリージョンを指定すると、リージョンのコンプライアンス要件に準拠しやすくなります。

Amazon S3 の制限により、カスターマネージドキー (CMK) 暗号化リージョンデータレイクから S3 マネージド暗号化 (デフォルト暗号化) リージョンデータレイクへのレプリケーションはサポートされていません。

Important

カスタムソースを作成した場合、カスタムソースデータが宛先に正しくレプリケートされるように、Security Lake は「[カスタムソースを取り込むためのベストプラクティス](#)」で説明されているベストプラクティスに従うことをお勧めします。ページで説明されているように、S3 パーティションデータパス形式に従わないデータに対してレプリケーションを実行することはできません。

ロールアップリージョンを追加する前に、まず AWS Identity and Access Management (IAM) で 2 つの異なるロールを作成する必要があります。

- [データレプリケーションの IAM ロール](#)
- [AWS Glue パーティションを登録する IAM ロール](#)

Note

Security Lake コンソールを使用すると、ユーザーに代わって Security Lake がこれらの IAM ロールを作成するか、既存のロールを使用します。ただし、Security Lake API または を使用する場合は、これらのロールを作成する必要があります AWS CLI。

データレプリケーションの IAM ロール

この IAM ロールは、ソースログとイベントを複数のリージョンにレプリケートする権限を Amazon S3 に付与します。

これらのアクセス権限を付与するには、プレフィックス SecurityLake で始まる IAM ロールを作成し、以下のサンプルポリシーをアタッチします。Security Lake でロールアップリージョン

を作成する場合、ロールの Amazon リソースネーム (ARN) が必要になります。このポリシーでは、sourceRegions は寄与リージョン、destinationRegions はロールアップリージョンです。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowReadS3ReplicationSetting",
      "Action": [
        "s3:ListBucket",
        "s3:GetReplicationConfiguration",
        "s3:GetObjectVersionForReplication",
        "s3:GetObjectVersion",
        "s3:GetObjectVersionAcl",
        "s3:GetObjectVersionTagging",
        "s3:GetObjectRetention",
        "s3:GetObjectLegalHold"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3::aws-security-data-lake-[[sourceRegions]]*",
        "arn:aws:s3::aws-security-data-lake-[[sourceRegions]]*/*"
      ],
      "Condition": {
        "StringEquals": {
          "s3:ResourceAccount": [
            "{{bucketOwnerAccountId}}"
          ]
        }
      }
    },
    {
      "Sid": "AllowS3Replication",
      "Action": [
        "s3:ReplicateObject",
        "s3:ReplicateDelete",
        "s3:ReplicateTags",
        "s3:GetObjectVersionTagging"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3::aws-security-data-lake-[[destinationRegions]]*/*"
      ]
    }
  ]
}
```

```

    ],
    "Condition": {
      "StringEquals": {
        "s3:ResourceAccount": [
          "{{bucketOwnerAccountId}}"
        ]
      }
    }
  }
]
}

```

次の信頼ポリシーをロールにアタッチして、Amazon S3 がロールを引き受けることを許可します。

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowS3ToAssume",
      "Effect": "Allow",
      "Principal": {
        "Service": "s3.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}

```

AWS Key Management Service (AWS KMS) のカスタマーマネージドキーを使用して Security Lake データレイクを暗号化する場合は、データレプリケーションポリシーのアクセス許可に加えて、次のアクセス許可を付与する必要があります。

```

{
  "Action": [
    "kms:Decrypt"
  ],
  "Effect": "Allow",
  "Condition": {
    "StringLike": {
      "kms:ViaService": [
        "s3.{sourceRegion1}.amazonaws.com",
        "s3.{sourceRegion2}.amazonaws.com"
      ]
    }
  }
}

```

```
    ],
    "kms:EncryptionContext:aws:s3:arn": [
      "arn:aws:s3:::aws-security-data-lake-{sourceRegion1}*",
      "arn:aws:s3:::aws-security-data-lake-{sourceRegion2}*"
    ]
  }
},
"Resource": [
  "{sourceRegion1KmsKeyArn}",
  "{sourceRegion2KmsKeyArn}"
]
},
{
  "Action": [
    "kms:Encrypt"
  ],
  "Effect": "Allow",
  "Condition": {
    "StringLike": {
      "kms:ViaService": [
        "s3.{destinationRegion1}.amazonaws.com",
      ],
      "kms:EncryptionContext:aws:s3:arn": [
        "arn:aws:s3:::aws-security-data-lake-{destinationRegion1}*"
      ]
    }
  },
  "Resource": [
    "{destinationRegionKmsKeyArn}"
  ]
}
```

レプリケーションロールの詳細については、「Amazon Simple Storage Service ユーザーガイド」の「[アクセス許可の設定](#)」を参照してください。

AWS Glue パーティションを登録する IAM ロール

この IAM ロールは、Security Lake が他のリージョンからレプリケートされた S3 オブジェクトの AWS Glue パーティションを登録するために使用するパーティションアップデーター AWS Lambda 関数のアクセス許可を付与します。このロールを作成しないと、サブスクライバーはそれらのオブジェクトからのイベントをクエリできません。

これらの権限を付与するには、AmazonSecurityLakeMetaStoreManager (Security Lake への登録時にこのロールをすでに作成している場合があります) という名前のロールを作成します。サンプルポリシーを含む、このロールの詳細については、「[ステップ 1: IAM ロールを作成する](#)」を参照してください。

また、Lake Formation コンソールで、次の手順に従ってデータレイク管理者の AmazonSecurityLakeMetaStoreManager 権限を付与する必要があります。

1. Lake Formation コンソール (<https://console.aws.amazon.com/lakeformation/>) を開きます。
2. 管理ユーザーとしてサインインする
3. [Lake Formation へようこそ] ウィンドウが表示されたら、ステップ 1 で作成または選択したユーザーを選択し、[開始する] を選択します。
4. [Lake Formation へようこそ] ウィンドウが表示されない場合は、以下の手順を実行して Lake Formation 管理者を設定します。
 1. ナビゲーションペインの [許可] で [管理ロールとタスク] を選択します。コンソールページの [データレイク管理者] セクションで、[管理者を選択] を選択します。
 2. [データレイク管理者の管理] ダイアログ ボックスで、IAM ユーザーとロールに対して、作成した AmazonSecurityLakeMetaStoreManager IAM ロールを選択し、[保存] を選択します。

データレイク管理者の権限変更の詳細については、「AWS Lake Formation デベロッパーガイド」の「[データレイク管理者の作成](#)」を参照してください。

ロールアップリージョンの追加

お好みのアクセス方法を選択し、次の手順に従ってロールアップリージョンを追加します。

Note

1 つのリージョンは複数のロールアップリージョンにデータを提供できます。ただし、あるロールアップリージョンを別のロールアップリージョンの寄与リージョンにすることはできません。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ナビゲーションペインの [設定] で、[ロールアップリージョン] を選択します。

3. [変更] を選択し、[ロールアップリージョンの追加] を選択します。
4. ロールアップリージョンと寄与リージョンを指定します。複数のロールアップリージョンを追加する場合は、このステップを繰り返します。
5. ロールアップリージョンを初めて追加する場合は、サービスアクセス用に新しい IAM ロールを作成するか、複数のリージョンにデータをレプリケートする権限を Security Lake に付与する既存の IAM ロールを使用してください。
6. 完了したら、保存 を選択します。

Security Lake へのオンボーディング時にロールアップリージョンを追加することもできます。詳細については、「[Amazon Security Lake の開始方法](#)」を参照してください。

API

プログラムでロールアップリージョンを追加するには、Security Lake API の [UpdateDataLake](#) オペレーションを使用します。を使用している場合は AWS CLI、[update-data-lake](#) コマンドを実行します。リクエストでは、region フィールドを使用して、ロールアップリージョンにデータを提供するリージョンを指定します。replicationConfiguration パラメータの regions 配列で、各ロールアップリージョンのリージョンコードを指定します。リージョンコードのリストについては、「AWS 全般のリファレンス」の「[Amazon Security Lake エンドポイント](#)」を参照してください。

例えば、次のコマンドは をロールアップリージョン ap-northeast-2 として設定します。us-east-1 リージョンは、ap-northeast-2 リージョンにデータを提供します。この例では、データレイクに追加されたオブジェクトの有効期限を 365 日間設定します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックslash (\) の行継続文字を使用しています。

```
$ aws securitylake update-data-lake \  
--configurations '[{"encryptionConfiguration":  
  {"kmsKeyId": "S3_MANAGED_KEY"}, "region": "us-east-1", "replicationConfiguration":  
  {"regions": ["ap-northeast-2"], "roleArn": "arn:aws:iam::123456789012:role/service-  
role/AmazonSecurityLakeS3ReplicationRole"}, "lifecycleConfiguration": {"expiration":  
  {"days": 365}}}]'
```

Security Lake へのオンボーディング時にロールアップリージョンを追加することもできます。これを行うには、[CreateDataLake](#) オペレーションを使用します (または、を使用している場合は AWS CLI [create-data-lake](#) コマンド)。オンボーディング中のロールアップリージョンの設定の詳細については、「」を参照してください [Amazon Security Lake の開始方法](#)。

ロールアップリージョンの更新または削除

お好みのアクセス方法を選択し、次の手順に従って Security Lake のロールアップリージョン を更新または削除します。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ナビゲーションペインの [設定] で、[ロールアップリージョン] を選択します。
3. [変更] を選択します。
4. ロールアップリージョンのコントリビューションリージョンを変更するには、ロールアップリージョンの行に更新されたコントリビューションリージョンを指定します。
5. ロールアップリージョンを削除するには、ロールアップリージョンの行で [削除] を選択します。
6. 完了したら、保存 を選択します。

API

プログラムでロールアップリージョンを設定するには、Security Lake API の [UpdateDataLake](#) オペレーションを使用します。を使用している場合は AWS CLI、[update-data-lake](#) コマンドを実行します。リクエストでは、サポートされているパラメータを使用してロールアップ設定を指定します。

- 寄与リージョンを追加するには、`region` フィールドを使用して追加するリージョンのリージョンコードを指定します。replicationConfiguration オブジェクトの regions アレイで、データを投稿する各ロールアップリージョンのリージョンコードを指定します。リージョンコードのリストについては、「AWS 全般のリファレンス」の「[Amazon Security Lake エンドポイント](#)」を参照してください。
- 寄与リージョンを削除するには、`region` フィールドを使用して削除するリージョンのリージョンコードを指定します。replicationConfiguration パラメータには値を指定しないでください。

例えば、次のコマンドは、`us-east-1` と `us-east-2` の両方を寄与リージョンとして設定します。両方のリージョンが `ap-northeast-3` ロールアップリージョンにデータを提供します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake update-data-lake \  
--configurations '[{"encryptionConfiguration":  
  {"kmsKeyId":"S3_MANAGED_KEY"},"region":"us-east-1","replicationConfiguration":  
  {"regions": ["ap-northeast-3"],"roleArn":"arn:aws:iam::123456789012:role/service-  
role/AmazonSecurityLakeS3ReplicationRole"},"lifecycleConfiguration": {"expiration":  
{"days":365}}},  
{"encryptionConfiguration": {"kmsKeyId":"S3_MANAGED_KEY"},"region":"us-  
east-2","replicationConfiguration": {"regions": ["ap-  
northeast-3"],"roleArn":"arn:aws:iam::123456789012:role/service-role/  
AmazonSecurityLakeS3ReplicationRole"}, "lifecycleConfiguration": {"expiration":  
{"days":500},"transitions":[{"days":60,"storageClass":"ONEZONE_IA"}]}]'
```

Security Lake でのソース管理

ソースは、[Security Lake でサイバーセキュリティスキーマフレームワーク \(OCSF\) を開くスキーマ](#)の特定のイベントクラスと一致する 1 つのシステムから生成されるログとイベントです。Amazon Security Lake は、ネイティブにサポートされているソースやサードパーティーのカスタムソースなど、さまざまなソースからログ AWS のサービス やイベントを収集できます。

Security Lake は、生のソースデータに対して抽出、変換、ロード (ETL) ジョブを実行し、データが Apache Parquet 形式と OCSF スキーマに変換します。処理後、Security Lake は、データが生成された の がある Amazon Simple Storage Service (Amazon S3) バケット AWS アカウント にソース AWS リージョン データを保存します。Security Lake は、サービスを有効にするリージョンごとに異なる Amazon S3 バケットを作成します。各ソースは S3 バケットに個別のプレフィックスを取得し、Security Lake は各ソースのデータを個別の AWS Lake Formation テーブルセットに整理します。

トピック

- [Security Lake AWS のサービス での からのデータ収集](#)
- [Security Lake のカスタムソースからデータを収集する](#)

Security Lake AWS のサービス での からのデータ収集

Amazon Security Lake では、AWS のサービスネイティブにサポートされているからログとイベントを収集できます。

- AWS CloudTrail 管理イベントとデータイベント (S3、Lambda)
- Amazon Elastic Kubernetes Service (Amazon EKS) 監査ログ
- Amazon Route 53 Resolver クエリログ
- AWS Security Hub 検出結果
- Amazon Virtual Private Cloud (Amazon VPC) Flow Logs
- AWS WAF v2 ログ

Security Lake は、このデータを [Security Lake でサイバーセキュリティスキーマフレームワーク \(OCSF\) を開く](#) および Apache Parquet 形式に自動的に変換します。

i Tip

前述の 1 つ以上のサービスを Security Lake のログソースとして追加するには、CloudTrail 管理イベントを除いて、これらのサービスのロギングを個別に設定する必要はありません。これらのサービスでロギングを設定している場合は、ログ設定を変更して Security Lake のログソースとして追加する必要はありません。Security Lake は、独立イベントストリームと重複イベントストリームでデータを直接取得します。

前提：アクセス許可

Security Lake のソース AWS のサービスとして を追加するには、必要なアクセス許可が必要です。ソースの追加に使用するロールにアタッチされた AWS Identity and Access Management (IAM) ポリシーに、次のアクションを実行するアクセス許可があることを確認します。

- glue:CreateDatabase
- glue:CreateTable
- glue:GetDatabase
- glue:GetTable
- glue:UpdateTable
- iam:CreateServiceLinkedRole
- s3:GetObject
- s3:PutObject

ロールには、 および アクセスs3:PutObject許可の以下の条件S3:getObjectとリソーススコープを設定することをお勧めします。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowUpdatingSecurityLakeS3Buckets",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:PutObject"
      ]
    }
  ]
}
```

```
    ],
    "Resource": "arn:aws:s3:::aws-security-data-lake*",
    "Condition": {
      "StringEquals": {
        "aws:ResourceAccount": "${aws:PrincipalAccount}"
      }
    }
  }
]
}
```

これらのアクションにより、 からログとイベントを収集 AWS のサービス し、正しい AWS Glue データベースとテーブルに送信できます。

データレイクのサーバー側の暗号化に AWS KMS キーを使用する場合は、 のアクセス許可も必要です `kms:DescribeKey`。

ソース AWS のサービス としての の追加

をソース AWS のサービス として追加すると、Security Lake は自動的にセキュリティログとイベントの収集を開始します。これらの手順では、ネイティブにサポートされている を Security Lake のソース AWS のサービス として追加する方法を説明します。カスタムソースを追加する手順については、[Security Lake のカスタムソースからデータを収集する](#)を参照してください。

Console

AWS ログソースを追加するには (コンソール)

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/>を開きます。
2. ナビゲーションペインで [ソース] を選択します。
3. データを収集 AWS のサービス する を選択し、設定を選択します。
4. ソース設定 セクションで、ソースを有効にし、データの取り込みに使用するデータソースのバージョンを選択します。デフォルトでは、最新バージョンのデータソースは Security Lake によって取り込まれます。

⚠ Important

指定されたリージョンで新しいバージョンの AWS ログソースを有効にするために必要なロールのアクセス許可がない場合は、Security Lake 管理者にお問い合わせください。詳細については、[「ロールのアクセス許可の更新」](#)を参照してください。

サブスクライバーが選択したバージョンのデータソースを取り込むには、サブスクライバー設定も更新する必要があります。サブスクライバーを編集する方法の詳細については、[「Amazon Security Lake でのサブスクライバー管理」](#)を参照してください。

オプションで、最新バージョンのみを取り込んで、データインジェストに使用される以前のソースバージョンをすべて無効にすることもできます。

5. リージョン セクションで、ソースのデータを収集するリージョンを選択します。Security Lake は、選択したリージョンのすべてのアカウントからソースからデータを収集します。
6. [有効化] を選択します。

API

AWS ログソースを追加するには (API)

をソース AWS のサービスとしてプログラムで追加するには、Security Lake API の [CreateAwsLogSource](#) オペレーションを使用します。AWS Command Line Interface (AWS CLI) を使用している場合は、[create-aws-log-source](#) コマンドを実行します。sourceName および regions パラメータが必要です。必要に応じて、ソースの範囲を特定の accounts または特定の sourceVersion に制限できます。

⚠ Important

コマンドでパラメータを指定しない場合、Security Lake は欠落しているパラメータがセット全体を参照していると思なします。例えば、パラメータ accounts を指定しない場合、コマンドは組織内のアカウントのセット全体に適用されます。

次の例では、指定されたアカウントとリージョンのソースとして VPC フローログを追加します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

Note

Security Lake を有効にしていないリージョンにこのリクエストを適用すると、エラーが発生します。このエラーを解決するには、そのリージョンで Security Lake を有効にするか、regions パラメータを使用して Security Lake を有効にしたリージョンのみを指定します。

```
$ aws securitylake create-aws-log-source \  
--sources sourceName=VPC_FLOW,accounts='["123456789012",  
"111122223333"]',regions=["us-east-2"],sourceVersion="2.0"
```

ソースコレクションのステータスの取得

アクセス方法を選択し、手順に従って、現在のリージョンでログ収集が有効になっているアカウントとソースのスナップショットを取得します。

Console

現在のリージョンのログ収集のステータスを取得するには

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ナビゲーションペインで、アカウントを選択します。
3. ソース列の番号にカーソルを合わせると、選択したアカウントでどのログが有効になっているかが表示されます。

API

現在のリージョンのログ収集のステータスを取得するには、Security Lake API の [GetDataLakeSources](#) オペレーションを使用します。を使用している場合は AWS CLI、[get-data-lake-sources](#) コマンドを実行します。accounts パラメータでは、1 つ以上の AWS アカウント IDs リストとして指定できます。リクエストが成功すると、Security Lake は、Security Lake がデータを収集している AWS ソースや各ソースのステータスなど、現在のリージョン内のアカウントのスナップショットを返します。accounts パラメータを含めない場合、レスポンスには、Security Lake が現在のリージョンで設定されているすべてのアカウントのログ収集のステータスが含まれます。

例えば、次の AWS CLI コマンドは、現在のリージョンで指定されたアカウントのログ収集ステータスを取得します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake get-data-lake-sources \  
--accounts "123456789012" "111122223333"
```

Security Lake でのロールのアクセス許可の更新

データソースの新しいバージョンからデータを取り込むために必要なロールのアクセス許可またはリソース、つまり新しい AWS Lambda 関数と Amazon Simple Queue Service (Amazon SQS) キューがない場合は、AmazonSecurityLakeMetaStoreManagerV2ロールのアクセス許可を更新し、新しいリソースセットを作成してソースからのデータを処理する必要があります。

任意の方法を選択し、手順に従ってロールのアクセス許可を更新し、新しいリソースを作成して、指定したリージョンの AWS ログソースの新しいバージョンからのデータを処理します。これは、アクセス許可とリソースが将来のデータソースリリースに自動的に適用されるため、1 回限りのアクションです。

Console

ロールのアクセス許可を更新するには (コンソール)

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。

委任された Security Lake 管理者の認証情報を使用してサインインします。

2. ナビゲーションペインで [Settings] の [General] を選択します。

3. ロールのアクセス許可の更新を選択します。

4. サービスアクセスセクションで、次のいずれかを実行します。

- 新しいサービスロールを作成して使用する — Security Lake によって作成された AmazonSecurityLakeMetaStoreManagerV2 ロールを使用できます。
- 既存のサービスロールを使用する — サービスロール名リストから既存のサービスロールを選択できます。

5. [Apply] を選択します。

API

ロールのアクセス許可を更新するには (API)

アクセス許可をプログラムで更新するには、Security Lake API の [UpdateDataLake](#) オペレーションを使用します。を使用してアクセス許可を更新するには AWS CLI、[update-data-lake](#) コマンドを実行します。

ロールのアクセス許可を更新するには、[AmazonSecurityLakeMetastoreManager](#) ポリシーをロールにアタッチする必要があります。

AmazonSecurityLakeMetaStoreManager ロールの削除

Important

ロールのアクセス許可を に更新したら AmazonSecurityLakeMetaStoreManagerV2、古い AmazonSecurityLakeMetaStoreManager ロールを削除する前にデータレイクが正しく機能することを確認します。ロールを削除する前に、少なくとも 4 時間待つことをお勧めします。

ロールを削除する場合は、まず AmazonSecurityLakeMetaStoreManager ロールを削除する必要があります AWS Lake Formation。

Lake Formation コンソールから AmazonSecurityLakeMetaStoreManager ロールを削除するには、次の手順に従います。

1. にサインインし AWS Management Console、<https://console.aws.amazon.com/lakeformation/> で Lake Formation コンソールを開きます。
2. Lake Formation コンソールのナビゲーションペインで、管理ロールとタスクを選択します。
3. 各リージョン AmazonSecurityLakeMetaStoreManager から を削除します。

Security Lake からのソース AWS のサービスとしての の削除

アクセス方法を選択し、以下の手順に従って、Security Lake ソース AWS のサービスとしてのネイティブにサポートされている を削除します。1 つ以上のリージョンのソースを削除できます。ソースを削除すると、Security Lake は指定されたリージョンとアカウントでそのソースからデータを

収集しなくなり、利用者はソースから新しいデータを使用できなくなります。ただし、利用者は Security Lake が削除前にソースから収集したデータを引き続き利用できます。これらの手順は、ネイティブにサポートされている AWS のサービスをソースとして削除する場合にのみ使用できます。カスタムソースの削除については、[Security Lake のカスタムソースからデータを収集する](#)を参照してください。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/>を開きます。
2. ナビゲーションペインで [ソース] を選択します。
3. ソースを選択し、[無効化] を選択します。
4. このソースからのデータ収集を停止したい地域を 1 つまたは複数選択します。Security Lake は、選択したリージョンのすべてのアカウントからのソースからのデータ収集を停止します。

API

をソース AWS のサービスとしてプログラムで削除するには、Security Lake API の [DeleteAwsLogSource](#) オペレーションを使用します。AWS Command Line Interface (AWS CLI) を使用している場合は、[delete-aws-log-source](#) コマンドを実行します。sourceName および regions パラメータが必要です。必要に応じて、削除の範囲を特定の accounts または特定の sourceVersion に制限できます。

Important

コマンドでパラメータを指定しない場合、Security Lake は欠落しているパラメータがセット全体を参照していると見なします。例えば、パラメータ accounts を指定しない場合、コマンドは組織内のアカウントのセット全体に適用されます。

次の例では、指定されたアカウントとリージョンのソースとして VPC フローログを削除します。

```
$ aws securitylake delete-aws-log-source \  
--sources sourceName=VPC_FLOW,accounts='["123456789012",  
"111122223333"]',regions='["us-east-1", "us-east-2"]',sourceVersion="2.0"
```

次の例では、指定されたアカウントとリージョンのソースとして Route 53 を削除します。

```
$ aws securitylake delete-aws-log-source \  
--sources sourceName=ROUTE53,accounts='["123456789012"]',regions='["us-east-1", "us-  
east-2"]',sourceVersion="2.0"
```

前述の例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

Security Lake の CloudTrail イベントログ

AWS CloudTrail は、AWS Management Console、AWS SDKs、コマンドラインツール、特定の AWS サービスを使用して行われた AWS API コールなど、アカウントの API コールの履歴を提供します。CloudTrail では、CloudTrail をサポートするサービス用の AWS API を呼び出したユーザーとアカウント、呼び出し元のソース IP アドレス、および呼び出しの発生日時を特定することもできます。詳細については、[AWS CloudTrail ユーザーガイド](#)をご参照ください。

Security Lake は、S3 および Lambda 用の CloudTrail 管理イベントと CloudTrail データイベントに関連するログを収集できます。CloudTrail 管理イベント、S3 データイベント、および Lambda データイベントは、Security Lake の 3 つの独立したソースです。その結果、これらのいずれかを取り込みログソースとして追加すると、`sourceName` の値が異なります。管理イベントは、コントロールプレーンイベントとも呼ばれ、のリソースで実行される管理オペレーションに関するインサイトを提供します。AWS アカウント。データプレーンオペレーションとして知られる CloudTrail データイベントは、AWS アカウントのリソース上またはリソース内で実行したリソースオペレーションに関するインサイトを提供します。これらの操作は、多くの場合、高ボリュームのアクティビティです。

Security Lake で CloudTrail 管理イベントを収集するには、読み取りと書き込みの CloudTrail 管理イベントを収集する CloudTrail マルチリージョン組織トレイルが少なくとも 1 つ必要です。トレイルのロギングが有効になっている必要があります。他のサービスでロギングを設定している場合は、ロギング設定を変更して Security Lake のログソースとして追加する必要はありません。Security Lake は、独立イベントストリームと重複イベントストリームでデータを直接取得します。

マルチリージョン証跡は、複数のリージョンから単一の Amazon Simple Storage Service (Amazon S3) バケットにログファイルを 1 AWS アカウントつのリージョンで配信します。CloudTrail コンソールまたはで管理されているマルチリージョン証跡がすでにある場合は AWS Control Tower、それ以上のアクションは必要ありません。

- CloudTrail を使用した証跡の作成と管理の詳細については、AWS CloudTrail ユーザーガイドの「[組織の証跡の作成](#)」を参照してください。

- による証跡の作成と管理の詳細については AWS Control Tower、「AWS Control Tower ユーザーガイド」の「[による AWS Control Tower アクションのログ記録 AWS CloudTrail](#)」を参照してください。

CloudTrail イベントをソースとして追加すると、Security Lakeはすぐに CloudTrail イベントログの収集を開始します。CloudTrail 管理および データイベントを CloudTrail から直接、独立イベントストリームと重複イベントストリームで消費します。

Security Lake は CloudTrail イベントを管理したり、既存の CloudTrail 構成に影響を与えたりしません。CloudTrail イベントのアクセスと保持を直接管理するには、CloudTrail サービスコンソールまたは API を使用する必要があります。詳細については、「AWS CloudTrail ユーザーガイド」の「[CloudTrail イベント履歴でのイベントの表示](#)」を参照してください。

次のリストは、Security Lake が CloudTrail イベントを OCSF に正規化する方法のマッピングリファレンスへの GitHub リポジトリリンクを示しています。

CloudTrail イベントの GitHub OCSF リポジトリ

- ソースバージョン 1 ([v1.0.0-rc.2](#))
- ソースバージョン 2 ([v1.1.0](#))

Security Lake の Amazon EKS 監査ログ

Amazon EKS 監査ログをソースとして追加すると、Security Lake は Elastic Kubernetes Service (EKS) クラスタで実行されている Kubernetes リソースで実行されているアクティビティに関する詳細な情報の収集を開始します。EKS 監査ログは、Amazon Elastic Kubernetes Service 内の EKS クラスタで疑わしいアクティビティを検出するのに役立ちます。

Security Lake は、監査ログの独立した重複ストリームを通じて、Amazon EKS コントロールプレーンのログ記録機能から直接 EKS 監査ログイベントを使用します。このプロセスは、追加のセットアップを必要とせず、既存の Amazon EKS コントロールプレーンのログ記録設定に影響を与えるように設計されています。詳細については、Amazon EKS ユーザーガイドの [Amazon EKS クラスタコントロールプレーンのログ](#) を参照してください。

Amazon EKS 監査ログは OCSF v1.1.0 でのみサポートされています。Security Lake が EKS 監査ログイベントを OCSF に正規化する方法については、[Amazon EKS 監査ログイベント \(v1.1.0\) の GitHub OCSF リポジトリ](#) のマッピングリファレンスを参照してください。

Security Lake の Route 53 リゾルバークエリログ

Route 53 リゾルバークエリログは、Amazon Virtual Private Cloud (Amazon VPC) 内のリソースによって実行された DNS クエリを追跡します。これにより、アプリケーションの動作状況を把握し、セキュリティ上の脅威を見抜くことができます。

Route 53 リゾルバークエリログを Security Lake のソースとして追加すると、Security Lake はすぐに、独立した重複したイベントストリームを通じて Route 53 から直接リゾルバークエリログを収集し始めます。

Security Lake は Route 53 ログを管理したり、既存のリゾルバークエリロギング設定に影響を与えたりすることはありません。リゾルバークエリログを管理するには、Route 53 サービスコンソールを使用する必要があります。詳細については、Amazon Route 53 デベロッパーガイドの「[リゾルバークエリログ設定の管理](#)」を参照してください。

次のリストは、Security Lake が Route 53 ログを OCSF に正規化する方法のマッピングリファレンスへの GitHub リポジトリリンクを示しています。

Route 53 ログ用の GitHub OCSF リポジトリ

- ソースバージョン 1 ([v1.0.0-rc.2](#))
- ソースバージョン 2 ([v1.1.0](#))

Security Lake での Security Hub の検出結果

Security Hub の検出結果は、のセキュリティ体制を理解し AWS、セキュリティ業界標準とベストプラクティスに照らして環境をチェックするのに役立ちます。Security Hub は、他のとの統合、サードパーティー製品の統合 AWS のサービス、Security Hub コントロールに対するチェックなど、さまざまなソースから検出結果を収集します。Security Hub は、AWS Security Finding Format (ASFF) と呼ばれる標準形式で検出結果を処理します。

Security Hub の結果を Security Lake のソースとして追加すると、Security Lake はすぐに、独立した重複したイベントストリームを通じて Security Hub から直接調査結果を収集し始めます。また、Security Lake は調査結果を ASFF から [Security Lake でサイバーセキュリティスキーマフレームワーク \(OCSF\) を開く](#) (OCSF) に変換します。

Security Lake は Security Hub 結果を管理したり、Security Hub の設定に影響を与えたりしません。Security Hub の検出結果を管理するには、Security Hub サービスコンソール、API、または

使用する必要があります AWS CLI。詳細については、AWS Security Hub ユーザー ガイドの「[AWS Security Hubの調査結果](#)」を参照してください。

次のリストは、Security Lake が Security Hub の検出結果を OCSF に正規化する方法のマッピングリファレンスへの GitHub リポジトリリンクを示しています。

Security Hub の検出結果の GitHub OCSF リポジトリ

- ソースバージョン 1 ([v1.0.0-rc.2](#))
- ソースバージョン 2 ([v1.1.0](#))

Security Lake の VPC フローログ

Amazon VPC の VPC Flow Logs機能は、環境内のネットワークインターフェイスとの間で送受信される IP トラフィックに関する情報をキャプチャします。

Security Lake のソースとして VPC Flow Logsを追加すると、Security Lakeはすぐに VPC Flow Logs の収集を開始します。フローログの独立した重複ストリームを介して、Amazon VPC から直接 VPC フローログを消費します。

Security Lakeは VPC Flow Logsを管理したり、Amazon VPC の設定に影響を与えたりしません。Flow Logsを管理するには、Amazon VPC サービスコンソールを使用する必要があります。詳細については、Amazon VPC デベロッパーガイドの「[Flow Logsの操作](#)」を参照してください。

次のリストは、Security Lake が VPC フローログを OCSF に正規化する方法のマッピングリファレンスへの GitHub リポジトリリンクを示しています。

VPC フローログ用の GitHub OCSF リポジトリ

- ソースバージョン 1 ([v1.0.0-rc.2](#))
- ソースバージョン 2 ([v1.1.0](#))

AWS WAF Security Lake の ログ

Security Lake でログソース AWS WAF として を追加すると、Security Lake はすぐにログの収集を開始します。AWS WAF は、エンドユーザーがアプリケーションに送信するウェブリクエストをモニタリングし、コンテンツへのアクセスを制御するために使用できるウェブアプリケーションファイアウォールです。ログに記録された情報には、AWS がリソースからウェブリクエストを AWS WAF

受信した時間、リクエストに関する詳細情報、およびリクエストが一致したルールに関する詳細が含まれます。

Security Lake は、独立した重複した AWS WAF ログストリーム AWS WAF を介して から直接ログを使用します。このプロセスは、追加のセットアップを必要とせず、既存の AWS WAF 設定に影響を与えないように設計されています。Security Lake ログは、[ウェブアクセスコントロールリスト \(ウェブ ACL\)](#) 設定で AWS WAF 許可されているデータのみを取得します。Security Lake アカウントのウェブ ACL で[データ保護](#)が有効になっている場合、生成されたデータはウェブ ACL 設定に基づいて編集またはハッシュされます。AWS WAF を使用してアプリケーションリソースを保護する方法については、「AWS WAF デベロッパーガイド」の「[の AWS WAF 仕組み](#)」を参照してください。

Important

Amazon CloudFront ディストリビューションを のリソースタイプとして使用している場合は AWS WAF、米国東部 (バージニア北部) を選択して Security Lake にグローバルログを取り込む必要があります。

AWS WAF ログは OCSF v1.1.0 でのみサポートされています。Security Lake が AWS WAF ログイベントを OCSF に正規化する方法については、[GitHub OCSF リポジトリの AWS WAF ログのマッピングリファレンス \(v1.1.0\)](#) を参照してください。

ソース AWS のサービスとしての の削除

アクセス方法を選択し、以下の手順に従って、Security Lake ソース AWS のサービスとしてネイティブにサポートされている を削除します。1 つ以上のリージョンのソースを削除できます。ソースを削除すると、Security Lake は指定されたリージョンとアカウントでそのソースからデータを収集しなくなり、利用者はソースから新しいデータを使用できなくなります。ただし、利用者は Security Lake が削除前にソースから収集したデータを引き続き利用できます。これらの手順は、ネイティブにサポートされている AWS のサービスをソースとして削除する場合にのみ使用できます。カスタムソースの削除については、[Security Lake のカスタムソースからデータを収集する](#) を参照してください。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ナビゲーションペインで [ソース] を選択します。
3. ソースを選択し、[無効化] を選択します。

- このソースからのデータ収集を停止したい地域を 1 つまたは複数選択します。Security Lake は、選択したリージョンのすべてのアカウントからのソースからのデータ収集を停止します。

API

をソース AWS のサービスとしてプログラムで削除するには、Security Lake API の [DeleteAwsLogSource](#) オペレーションを使用します。AWS Command Line Interface (AWS CLI) を使用している場合は、[delete-aws-log-source](#) コマンドを実行します。sourceName および regions パラメータが必要です。必要に応じて、削除の範囲を特定の accounts または特定の sourceVersion に制限できます。

Important

コマンドでパラメータを指定しない場合、Security Lake は欠落しているパラメータがセット全体を参照していると見なします。例えば、パラメータ accounts を指定しない場合、コマンドは組織内のアカウントのセット全体に適用されます。

次の例では、指定されたアカウントとリージョンのソースとして VPC フローログを削除します。

```
$ aws securitylake delete-aws-log-source \  
--sources sourceName=VPC_FLOW,accounts='["123456789012",  
"111122223333"]',regions='["us-east-1", "us-east-2"]',sourceVersion="2.0"
```

次の例では、指定されたアカウントとリージョンのソースとして Route 53 を削除します。

```
$ aws securitylake delete-aws-log-source \  
--sources sourceName=ROUTE53,accounts='["123456789012"]',regions='["us-east-1", "us-  
east-2"]',sourceVersion="2.0"
```

前述の例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行連続文字を使用しています。

Security Lake のカスタムソースからデータを収集する

Amazon Security Lake はサードパーティのカスタムソースからログとイベントを収集できます。Security Lake カスタムソースは、セキュリティログとイベントを Amazon Security Lake に送信するサードパーティーのサービスです。データを送信する前に、カスタムソースはログとイベントを Open Cybersecurity Schema Framework (OCSF) に変換し、パーティショニング、parquet ファイル形式、オブジェクトのサイズとレートの要件など、Security Lake のソース要件を満たしている必要があります。

Security Lake はカスタムソースごとに以下を処理します。

- Amazon S3 バケットのソースに一意プレフィックスが付けられます。
- (AWS Identity and Access Management IAM) でロールを作成し、カスタムソースがデータレイクにデータを書き込むことを許可します。このロールのアクセス許可の境界は、という AWS マネージドポリシーによって設定されます [AmazonSecurityLakePermissionsBoundary](#)。
- AWS Lake Formation テーブルを作成して、ソースが Security Lake に書き込むオブジェクトを整理します。
- ソースデータをパーティション化するための AWS Glue クローラを設定します。クローラは、AWS Glue Data Catalog に テーブルを入力します。また、新しいソースデータを自動的に検出し、スキーマ定義を抽出します。

Note

アカウントには最大 50 個のカスタムログソースを追加できます。

Security Lake にカスタムソースを追加するには、次の要件を満たす必要があります。これらの要件を満たさないと、パフォーマンスに影響が及び、クエリなどの分析のユースケースに影響する可能性があります。

- 送信先 — カスタムソースは、ソースに割り当てられたプレフィックスの下に S3 オブジェクトのセットとしてデータを Security Lake に書き込むことができる必要があります。複数のカテゴリのデータを含むソースの場合は、それぞれ固有の [Open Cybersecurity Schema Framework \(OCSF\) イベントクラス](#) を個別のソースとして配信する必要があります。Security Lake は、カスタムソースが S3 バケット内の指定された場所書き込むことを許可する IAM ロールを作成します。
- フォーマット — カスタムソースから収集された各 S3 オブジェクトは、Apache Parquet ファイルとしてフォーマットする必要があります。

- スキーマ — 同じ OCSF イベントクラスを Parquet 形式のオブジェクト内の各レコードに適用する必要があります。セキュリティレイクは Parquet のバージョン 1.x と 2.x をサポートします。データページのサイズは 1 MB (非圧縮) に制限する必要があります。行グループのサイズは 256 MB (圧縮) 以下でなければなりません。Parquet オブジェクト内の圧縮には、zstandard が推奨されます。
- パーティショニング — オブジェクトは region、AWS account、eventDay でパーティション化する必要があります。オブジェクトには のプレフィックスを付ける必要があります `source location/region=region/accountId=accountID/eventDay=yyyyMMdd/`。
- オブジェクトのサイズとレート — Security Lake に送信されるファイルは、5 分から 1 イベント日の間で増分で送信する必要があります。ファイルのサイズが 256MB を超える場合、お客様は 5 分以上頻繁にファイルを送信できます。オブジェクトとサイズの要件は、クエリパフォーマンスのために Security Lake を最適化することです。カスタムソース要件に従わないと、Security Lake のパフォーマンスに影響する可能性があります。
- ソート — 各 Parquet 形式のオブジェクト内では、データのクエリコストを削減するために、レコードを時間順に並べる必要があります。

Note

[OCSF 検証ツールを使用して](#)、カスタムソースが と互換性があるかどうかを確認します OCSF Schema。カスタムソースの場合、Security Lake は OCSF バージョン 1.3 以前をサポートしています。

Security Lake でカスタムソースを取り込むためのパーティション要件

効率的なデータ処理とクエリを容易にするには、Security Lake にカスタムソースを追加するときに、パーティショニング、オブジェクト、サイズに関する要件を満たす必要があります。

パーティション

オブジェクトは、ソースの場所、AWS リージョン、AWS アカウントおよび日付でパーティション化する必要があります。

- パーティションデータパスの形式は です。

```
/ext/custom-source-name/region=region/accountId=accountID/  
eventDay=YYYYMMDD.
```

バケット名の例を含むサンプルパーティションは `aws-security-data-lake-us-west-2-lake-uid/ext/custom-source-name/region=us-west-2/accountId=123456789012/eventDay=20230428/`。

次のリストでは、S3 パスパーティションで使用されるパラメータについて説明します。

- Security Lake がカスタムソースデータを保存する Amazon S3 バケットの名前。
- `source-location`— S3 バケットのカスタムソースのプレフィックス。Security Lake は、特定のソースのすべての S3 オブジェクトをこのプレフィックスの下に格納します。プレフィックスは特定のソースに固有のものです。
- `region`— データのアップロード AWS リージョン 先。例えば、US East (N. Virginia) を使用して、米国東部 (バージニア北部) リージョンの Security Lake バケットにデータをアップロードする必要があります。
- `accountId`— ソースパーティション内のレコードが関連する AWS アカウント ID。の外部のアカウントに関連するレコードには AWS、`external` や などの文字列を使用することをお勧めします `external_externalAccountId`。この命名規則を採用することで、外部アカウント IDs の命名があいまいになることを回避し、他の ID 管理システムによって維持されている AWS アカウント IDs や外部アカウント IDs と競合しないようにできます。
- `eventDay`— レコードの UTC タイムスタンプ。8 文字の文字列 () 形式の時間に変換されます `YYYYMMDD`。レコードでイベントタイムスタンプに別のタイムゾーンが指定されている場合は、このパーティションキーのタイムスタンプを UTC に変換する必要があります。

Security Lake でカスタムソースを追加するための前提条件

カスタムソースを追加すると、Security Lake は、ソースがデータレイク内の正しい場所にデータを書き込むことを許可する IAM ロールを作成します。ロールの名前は の形式に従います。ここで `AmazonSecurityLake-Provider-{name of the custom source}-{region}`、`region` はカスタムソースを追加する AWS リージョン です。Security Lake は、データレイクへのアクセスを許可するポリシーをロールにアタッチします。カスタマーマネージド AWS KMS キーでデータレイクを暗号化した場合、Security Lake は `kms:Decrypt` および `kms:GenerateDataKey` 許可を持つポリシーもロールにアタッチします。このロールのアクセス許可の境界は、 という AWS マネージドポリシーによって設定されます [AmazonSecurityLakePermissionsBoundary](#)。

トピック

- [アクセス許可の確認](#)

- [Security Lake バケットの場所への書き込みアクセスを許可する IAM ロールを作成する \(API および AWS CLI 専用ステップ\)](#)

アクセス許可の確認

カスタムソースを追加する前に、次のアクションを実行するアクセス許可があることを確認してください。

権限を確認するには、IAM を使用して IAM ID に添付されている IAM ポリシーを確認してください。次に、これらのポリシーの情報を、カスタム ソースを追加するために実行を許可する必要がある次のアクションのリストと比較します。

- glue:CreateCrawler
- glue:CreateDatabase
- glue:CreateTable
- glue:StopCrawlerSchedule
- iam:GetRole
- iam:PutRolePolicy
- iam>DeleteRolePolicy
- iam:PassRole
- lakeformation:RegisterResource
- lakeformation:GrantPermissions
- s3:ListBucket
- s3:PutObject

これらのアクションにより、カスタムソースからログとイベントを収集し、正しい AWS Glue データベースとテーブルに送信して、Amazon S3 に保存できます。

データレイクのサーバー側の暗号化に AWS KMS キーを使用する場合は、kms:CreateGrant、kms:DescribeKey、およびのアクセス許可も必要ですkms:GenerateDataKey。

Important

Security Lake コンソールを使用してカスタムソースを追加する場合は、次のステップをスキップして「」に進みます[Security Lake でのカスタムソースの追加](#)。Security Lake コン

ソールでは、必要なすべての IAM ロールを作成したり、ユーザーに代わって既存のロールを使用したりできるため、使い始めるためのプロセスが簡略化されています。

Security Lake API または を使用してカスタムソース AWS CLI を追加する場合は、次のステップに進み、Security Lake バケットの場所への書き込みアクセスを許可する IAM ロールを作成します。

Security Lake バケットの場所への書き込みアクセスを許可する IAM ロールを作成する (API および AWS CLI 専用ステップ)

Security Lake API または を使用してカスタムソース AWS CLI を追加する場合は、この IAM ロールを追加して、カスタムソースデータをクローलし、データ内のパーティションを識別する AWS Glue アクセス許可を付与します。これらのパーティションは、データを整理し、Data Catalog 内のテーブルを作成および更新するために必要です。

この IAM ロールを作成したら、カスタムソースを追加するためにロールの Amazon リソースネーム (ARN) が必要になります。

`arn:aws:iam::aws:policy/service-role/AWSGlueServiceRole` AWS 管理ポリシーをアタッチする必要があります。

必要なアクセス許可を付与するには、ロールに次のインラインポリシーを作成して埋め込み、AWS Glue クローラー がカスタムソースからデータファイルを読み取って AWS Glue Data Catalog のテーブルを作成/更新できるようにする必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "S3WriteRead",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:PutObject"
      ],
      "Resource": [
        "arn:aws:s3:::{{bucketName}}/*"
      ]
    }
  ]
}
```

```
}
```

次の信頼ポリシーをアタッチ AWS アカウント して、外部 ID に基づいてロールを引き受けることができる を許可します。

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

カスタムソースを追加するリージョンの S3 バケットがカスタマーマネージド で暗号化されている場合は AWS KMS key、次のポリシーをロールと KMS キーポリシーにアタッチする必要があります。

```
{
  "Effect": "Allow",
  "Action": [
    "kms:GenerateDataKey",
    "kms:Decrypt"
  ],
  "Condition": {
    "StringLike": {
      "kms:EncryptionContext:aws:s3:arn": [
        "arn:aws:s3:::{{name of S3 bucket created by Security Lake}}"
      ]
    }
  },
  "Resource": [
    "{{ARN of customer managed key}}"
  ]
}
```

Security Lake でのカスタムソースの追加

クローラーを呼び出す IAM AWS Glue ロールを作成したら、以下の手順に従って Security Lake にカスタムソースを追加します。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ページの右上隅にある AWS リージョン セレクターを使用して、カスタムソースを作成するリージョンを選択します。
3. ナビゲーションペインで [カスタムソース] を選択してから、[カスタムソースの作成] を選択します。
4. 「カスタムソースの詳細」セクションに、カスタムソースのグローバルに一意の名前を入力します。次に、カスタムソースが Security Lake に送信するデータの種別を説明する OCSF イベントクラスを選択します。
5. データの書き込み権限を持つ AWS アカウント の場合、データ レイクにログとイベントを書き込むカスタム ソースの AWS アカウント ID と外部 ID を入力します。
6. サービス アクセスの場合は、新しいサービス ロールを作成して使用するか、Security Lake に AWS Glue を呼び出すアクセス許可を付与する既存のサービス ロールを使用します。
7. [Create] (作成) を選択します。

API

カスタムソースをプログラムで追加するには、セキュリティレイク API の [CreateCustomLogSource](#) オペレーションを使用してください。カスタムソースを作成する AWS リージョン で オペレーションを使用します。AWS Command Line Interface (AWS CLI) を使用している場合は、[create-custom-log-source](#) コマンドを実行します。

リクエストでは、サポートされているパラメータを使用してカスタムソースの構成設定を指定します。

- `sourceName` – ソースの名前を指定します。名前は地域一意である必要があります。
- `eventClasses` – ソースが Security Lake に送信するデータのタイプを記述する 1 つ以上の OCSF イベントクラスを指定します。Security Lake でソースとしてサポートされている OCSF イベントクラスのリストについては、[「Open Cybersecurity Schema Framework \(OCSF\)」](#) を参照してください。

- `sourceVersion` – オプションで、ログ収集を特定のバージョンのカスタムソースデータに制限する値を指定します。
- `crawlerConfiguration` – クローラーを呼び出す AWS Glue ために作成した IAM ロールの Amazon リソースネーム (ARN) を指定します。IAM ロールを作成する詳細な手順については、[「カスタムソースを追加するための前提条件」](#)を参照してください。
- `providerIdentity` – ソースがデータレイクへのログとイベントの書き込みに使用する AWS ID と外部 ID を指定します。

次の例では、カスタムソースを、指定されたリージョンの指定されたログプロバイダーアカウントにログソースとして追加します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake create-custom-log-source \  
--source-name EXAMPLE_CUSTOM_SOURCE \  
--event-classes ['DNS_ACTIVITY', 'NETWORK_ACTIVITY'] \  
--configuration crawlerConfiguration={"roleArn=arn:aws:iam::XXX:role/service-role/  
RoleName"},providerIdentity={"externalId=ExternalId,principal=principal"} \  
--region=["ap-southeast-2"]
```

でカスタムソースデータを更新する AWS Glue

Security Lake にカスタムソースを追加すると、Security Lake AWS Glue は クローラを作成します。クローラーはカスタムソースに接続し、データ構造を決定し、AWS Glue データカタログにテーブルを入力します。

カスタムソーススキーマを最新の状態に保ち、Athena やその他のクエリサービスのクエリ機能を維持するために、クローラーを手動で実行することをお勧めします。特に、カスタムソースの入力データセットに以下のいずれかの変更が生じた場合は、クローラーを実行する必要があります。

- このデータセットには、1 つ以上の新しい最上位列があります。
- データセットには、structデータ型の列に 1 つ以上の新しいフィールドがあります。

クローラーの実行手順については、「AWS Glue デベロッパーガイド」の [AWS Glue 「クローラーのスケジュール」](#)を参照してください。

Security Lake では、アカウント内の既存のクローラーを削除または更新することはできません。カスタムソースを削除した場合、future 同じ名前のカスタムソースを作成する予定がある場合は、関連するクローラーを削除することをお勧めします。

サポートされている OCSF イベントクラス

Open Cybersecurity Schema Framework (OCSF) イベントクラスは、カスタムソースが Security Lake に送信するデータのタイプについて説明します。サポートされているイベントクラスのリストは次のとおりです。

```
public enum OcsfEventClass {
    ACCOUNT_CHANGE,
    API_ACTIVITY,
    APPLICATION_LIFECYCLE,
    AUTHENTICATION,
    AUTHORIZE_SESSION,
    COMPLIANCE_FINDING,
    DATASTORE_ACTIVITY,
    DEVICE_CONFIG_STATE,
    DEVICE_CONFIG_STATE_CHANGE,
    DEVICE_INVENTORY_INFO,
    DHCP_ACTIVITY,
    DNS_ACTIVITY,
    DETECTION_FINDING,
    EMAIL_ACTIVITY,
    EMAIL_FILE_ACTIVITY,
    EMAIL_URL_ACTIVITY,
    ENTITY_MANAGEMENT,
    FILE_HOSTING_ACTIVITY,
    FILE_SYSTEM_ACTIVITY,
    FTP_ACTIVITY,
    GROUP_MANAGEMENT,
    HTTP_ACTIVITY,
    INCIDENT_FINDING,
    KERNEL_ACTIVITY,
    KERNEL_EXTENSION,
    MEMORY_ACTIVITY,
    MODULE_ACTIVITY,
    NETWORK_ACTIVITY,
    NETWORK_FILE_ACTIVITY,
    NTP_ACTIVITY,
    PATCH_STATE,
    PROCESS_ACTIVITY,
```

```
RDP_ACTIVITY,  
REGISTRY_KEY_ACTIVITY,  
REGISTRY_VALUE_ACTIVITY,  
SCHEDULED_JOB_ACTIVITY,  
SCAN_ACTIVITY,  
SECURITY_FINDING,  
SMB_ACTIVITY,  
SSH_ACTIVITY,  
USER_ACCESS,  
USER_INVENTORY,  
VULNERABILITY_FINDING,  
WEB_RESOURCE_ACCESS_ACTIVITY,  
WEB_RESOURCES_ACTIVITY,  
WINDOWS_RESOURCE_ACTIVITY,  
// 1.3 OCSF event classes  
ADMIN_GROUP_QUERY,  
DATA_SECURITY_FINDING,  
EVENT_LOG_ACTIVITY,  
FILE_QUERY,  
FILE_REMEDIATION_ACTIVITY,  
FOLDER_QUERY,  
JOB_QUERY,  
KERNEL_OBJECT_QUERY,  
MODULE_QUERY,  
NETWORK_CONNECTION_QUERY,  
NETWORK_REMEDIATION_ACTIVITY,  
NETWORKS_QUERY,  
PERIPHERAL_DEVICE_QUERY,  
PROCESS_QUERY,  
PROCESS_REMEDIATION_ACTIVITY,  
REMEDIATION_ACTIVITY,  
SERVICE_QUERY,  
SOFTWARE_INVENTORY_INFO,  
TUNNEL_ACTIVITY,  
USER_QUERY,  
USER_SESSION_QUERY,  
// 1.3 OCSF event classes (Win extension)  
PREFETCH_QUERY,  
REGISTRY_KEY_QUERY,  
REGISTRY_VALUE_QUERY,  
WINDOWS_SERVICE_ACTIVITY  
}
```

Security Lake からのカスタムソースの削除

カスタムソースを削除して、ソースから Security Lake へのデータ送信を停止します。ソースを削除すると、Security Lake は指定されたリージョンとアカウントでそのソースからデータを収集しなくなり、利用者はソースから新しいデータを使用できなくなります。ただし、利用者は Security Lake が削除前にソースから収集したデータを引き続き利用できます。これらの手順は、カスタムソースの削除にのみ使用できます。ネイティブにサポートされている の削除については AWS のサービス、「」を参照してください [Security Lake AWS のサービス での からのデータ収集](#)。

Security Lake でカスタムソースを削除する場合は、Security Lake コンソールの外部にある各ソースをソースで無効にする必要があります。統合を無効にしないと、ソース統合が Amazon S3 にログを送信し続ける可能性があります。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ページの右上隅にある AWS リージョン セレクターを使用して、カスタムソースを削除するリージョンを選択します。
3. ナビゲーションペインで、[Custom source] (カスタムソース) を選択します。
4. 削除する カスタムソース を選択します。
5. [カスタムソースの登録解除] を選択し、[削除] を選択してアクションを確定します。

API

カスタムソースをプログラムで削除するには、セキュリティレイク API の [DeleteCustomLogSource](#) オペレーションを使用してください。AWS Command Line Interface (AWS CLI) を使用している場合は、[delete-custom-log-source](#) コマンドを実行します。カスタムソースを削除する AWS リージョン 内の操作を使用します。

リクエストでは、sourceNameパラメータを使用して削除するカスタムソースの名前を指定します。または、カスタムソースの名前を指定し、sourceVersionパラメーターを使用して削除の範囲をカスタムソースの特定のバージョンのデータだけに制限します。

次の の例では、Security Lake からカスタムログソースを削除します。

この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake delete-custom-log-source \
```

```
--source-name EXAMPLE_CUSTOM_SOURCE
```

Security Lake でのサブスクライバー管理

Amazon セキュリティレイクサブスクライバーは、セキュリティレイクからのログとイベントを使用します。コストを抑え、最小権限アクセスのベストプラクティスを順守するために、サブスクライバーにソースごとにデータへのアクセスを許可します。sources の詳細については、「[Security Lake でのソース管理](#)」を参照してください。

Security Lake は、2 種類のサブスクライバーをサポートしています。

- Amazon Security Lake のソースデータへのデータアクセス権を持つデータアクセスサブスクライバーには、データが S3 バケットに書き込まれると、ソースの新しいオブジェクトが通知されます。デフォルトでは、サブスクライバーは、提供する HTTPS エンドポイントを通じて新しいオブジェクトについて通知されます。また、Amazon Simple Queue Service (Amazon SQS) キューをポーリングすることで、サブスクライバーに新しいオブジェクトについて通知を受け取ることもできます。
- クエリアクセス – クエリアクセスを持つサブスクライバーは、Security Lake が収集するデータをクエリできます。これらのサブスクライバーは、Amazon Athena などのサービスを使用して S3 バケット内の AWS Lake Formation テーブルを直接クエリします。

サブスクライバーは、サブスクライバーの作成時に AWS リージョン 選択した 内のソースデータにのみアクセスできます。サブスクライバーが複数のリージョンのデータにアクセスできるようにするには、サブスクライバーを作成したリージョンをロールアップリージョンとして指定し、他のリージョンにデータを提供してもらうことができます。ロールアップリージョンと貢献リージョンの詳細については、[Security Lake でのリージョンの管理](#)を参照してください。

Important

Security Lake がサブスクライバーごとに追加できるソースの最大数は 10 です。これは、AWS ソースとカスタムソースの組み合わせである場合があります。

トピック

- [Security Lake サブスクライバーのデータアクセスの管理](#)
- [Security Lake サブスクライバーのクエリアクセスの管理](#)

Security Lake サブスクライバーのデータアクセスの管理

Amazon Security Lake のソースデータへのデータアクセス権を持つサブスクライバーには、データが S3 バケットに書き込まれると、ソースの新しいオブジェクトが通知されます。デフォルトでは、サブスクライバーは、提供する HTTPS エンドポイントを通じて新しいオブジェクトについて通知されます。また、Amazon Simple Queue Service (Amazon SQS) キューをポーリングすることで、サブスクライバーに新しいオブジェクトについて通知を受け取ることもできます。

オブジェクトが Security Lake データレイクに書き込まれると、サブスクライバーにソースの新しい Amazon S3 オブジェクトが通知されます。サブスクライバーは、Amazon Simple Queue Service (Amazon SQS) キューをポーリングすることで、S3 オブジェクトに直接アクセスし、新しいオブジェクトの通知を受け取ることができます。このサブスクリプションタイプは、[CreateSubscriber](#) API の `accessTypes` パラメーターで S3 として識別されます。

トピック

- [Security Lake でデータアクセスを持つサブスクライバーを作成するための前提条件](#)
- [Security Lake でデータアクセス権を持つサブスクライバーを作成する](#)
- [Security Lake でのデータサブスクライバーの更新](#)
- [Security Lake からのデータサブスクライバーの削除](#)

Security Lake でデータアクセスを持つサブスクライバーを作成するための前提条件

Security Lake でデータにアクセスできるサブスクライバーを作成する前に、次の必要条件を満たす必要があります。

アクセス許可の確認

権限を確認するには、IAM を使用して IAM ID に添付されている IAM ポリシーを確認してください。次に、それらのポリシーの情報を、データレイクに新しいデータが書き込まれたときにサブスクライバーに通知する必要がある次の (権限) アクションのリストと比較します。

以下のアクションの実行には許可が必要です。

- `iam:CreateRole`
- `iam>DeleteRolePolicy`

- iam:GetRole
- iam:PutRolePolicy
- lakeformation:GrantPermissions
- lakeformation:ListPermissions
- lakeformation:RegisterResource
- lakeformation:RevokePermissions
- ram:GetResourceShareAssociations
- ram:GetResourceShares
- ram:UpdateResourceShare

上記のリストに加えて、以下のアクションを実行する許可も必要です。

- events:CreateApiDestination
- events:CreateConnection
- events:DescribeRule
- events:ListApiDestinations
- events:ListConnections
- events:PutRule
- events:PutTargets
- s3:GetBucketNotification
- s3:PutBucketNotification
- sqs:CreateQueue
- sqs>DeleteQueue
- sqs:GetQueueAttributes
- sqs:GetQueueUrl
- sqs:SetQueueAttributes

サブスクライバーの外部 ID を取得します。

サブスクライバーを作成するには、サブスクライバーの AWS アカウント ID とは別に、外部 ID も取得する必要があります。外部 ID は、サブスクライバーが提供する固有の識別子です。Security Lake

は、作成したサブスクライバー IAM ロールに外部 ID を追加します。外部 ID は、Security Lake コンソール、API、または AWS CLI を使用してサブスクライバを作成するときに使用します。

外部 IDs 「IAM [ユーザーガイド](#)」の「[AWS リソースへのアクセスを第三者に付与するときに外部 ID を使用する方法](#)」を参照してください。

⚠ Important

Security Lake コンソールを使用してサブスクライバーを追加する予定がある場合は、次の手順をスキップして [Security Lake でデータアクセス権を持つサブスクライバーを作成する](#) に進むことができます。Security Lake コンソールでは、必要なすべての IAM ロールを作成したり、ユーザーに代わって既存のロールを使用したりできるため、使い始めるためのプロセスが簡略化されています。

Security Lake API または を使用してサブスクライバー AWS CLI を追加する場合は、次のステップに進み、EventBridge API 送信先を呼び出す IAM ロールを作成します。

EventBridge API 送信先を呼び出す IAM ロールを作成する (API および AWS CLI 専用ステップ)

API または を介して Security Lake を使用している場合は AWS CLI、AWS Identity and Access Management (IAM) でロールを作成し、API 送信先を呼び出してオブジェクト通知を正しい HTTPS エンドポイントに送信するアクセス許可を Amazon EventBridge に付与します。

この IAM ロールの作成が完了したら、サブスクライバーを作成するためにそのロールの Amazon リソースネーム (ARN) が必要になります。サブスクライバーが Amazon Simple Queue Service (Amazon SQS) キューからデータをポーリングする場合、または AWS Lake Formation からデータを直接クエリする場合、この IAM ロールは必要ありません。この種のデータアクセス方法 (アクセスタイプ) の詳細については、「[Security Lake サブスクライバーのクエリアクセスの管理](#)」を参照してください。

次のポリシーを IAM ロールに付けることができます。

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

```
        "Effect": "Allow",
        "Action": [
            "events:InvokeApiDestination"
        ],
        "Resource": [
            "arn:aws:events:{us-west-2}:{123456789012}:api-destination/
AmazonSecurityLake*/*"
        ]
    }
]
```

次の信頼ポリシーを IAM ロールにアタッチして、EventBridge がロールを引き受けることを許可します。

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

Security Lake は、サブスクライバーがデータレイクからデータを読み取ることを許可する IAM ロールを自動的に作成します (または、推奨される通知方法であれば Amazon SQS キューからイベントをポーリングする)。このロールは、という AWS マネージドポリシーで保護されています [AmazonSecurityLakePermissionsBoundary](#)。

Security Lake でデータアクセス権を持つサブスクライバーを作成する

次のいずれかのアクセス方法を選択して、現在の のデータにアクセスできるサブスクライバーを作成します AWS リージョン。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。

2. ページの右上隅にある AWS リージョン セレクターを使用して、サブスクライバーを作成するリージョンを選択します。
3. 左のナビゲーションペインで [サブスクライバー] を選択します。
4. 「サブスクライバー」ページで、「サブスクライバーを作成」を選択します。
5. サブスクライバーの詳細には、サブスクライバー名とオプションで説明を入力します。

リージョンは、現在選択されているとして自動入力 AWS リージョンされ、変更できません。

6. [ログとイベントソース] では、サブスクライバーが使用を許可されているソースを選択します。
7. [データアクセス方法] では、S3 を選択してサブスクライバーのデータアクセスを設定します。
8. サブスクライバーの認証情報には、サブスクライバーの AWS アカウント ID と [外部 ID](#) を指定します。
9. (オプション) 通知の詳細で、Security Lake でサブスクライバーがオブジェクト通知をポーリングできる Amazon SQS キューを作成させたい場合は、SQS キューを選択します。Security Lake が EventBridge 経由で HTTPS エンドポイントに通知を送信するには、[サブスクリプションエンドポイント] を選択します。

[サブスクリプションエンドポイント] を選択した場合は、以下も実行してください。

- a. サブスクリプションエンドポイントを入力します。有効なエンドポイント形式の例には、<http://example.com>があります。オプションで HTTPS キー名と HTTPS キー値を指定することもできます。
- b. サービスアクセスの場合は、新しい IAM ロールを作成するか、API 宛先を呼び出して適切なエンドポイントにオブジェクト通知を送信する権限を EventBridge に付与する既存の IAM ロールを使用します。

新しい IAM ロールを作成する方法については、「[EventBridge API の宛先を呼び出すための IAM ロールの作成](#)」を参照してください。

10. (オプション) [タグ] には、サブスクライバーに割り当てるタグを 50 個まで入力します。

タグは、特定のタイプの AWS リソースを定義して割り当てることができるラベルです。各タグは、必要なタグキーとオプションのタグ値で設定されています。タグは、さまざまな方法でリソースの識別、分類、管理に役立ちます。詳細については、「[Security Lake リソースのタグ付け](#)」を参照してください。

11. [作成] を選択します。

API

データにアクセスできるサブスクライバーをプログラムで作成するには、Security Lake API の [CreateSubscriber](#) オペレーションを使用します。AWS Command Line Interface (AWS CLI) を使用している場合は、[create-subscriber](#) コマンドを実行します。

リクエストでは、これらのパラメータを使用してサブスクライバーに次の設定を指定します。

- `sources` に、サブスクライバーにアクセスさせたいソースをそれぞれ指定します。
- には `subscriberIdentity`、サブスクライバーがソースデータにアクセスするために使用する AWS アカウント ID と外部 ID を指定します。
- には `subscriber-name`、サブスクライバーの名前を指定します。
- `accessTypes` の場合、S3 を指定します。

例 1

次の例では、ソースの指定されたサブスクライバー ID AWS の現在の AWS リージョンのデータにアクセスできるサブスクライバーを作成します。

```
$ aws securitylake create-subscriber \  
--subscriber-identity {"accountID": 1293456789123,"externalId": 123456789012} \  
--sources [{"awsLogSource": {"sourceName": VPC_FLOW, "sourceVersion": 2.0}}] \  
--subscriber-name subscriber name \  
--access-types S3
```

例 2

次の例では、カスタムソースの指定されたサブスクライバー ID の現在の AWS リージョンのデータにアクセスできるサブスクライバーを作成します。

```
$ aws securitylake create-subscriber \  
--subscriber-identity {"accountID": 1293456789123,"externalId": 123456789012} \  
--sources [{"customLogSource": {"sourceName": custom-source-name, \  
"sourceVersion": 2.0}}] \  
--subscriber-name subscriber name \  
--access-types S3
```

前述の例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

(オプション) サブスクライバーを作成したら、[CreateSubscriberNotification](#) オペレーションを使用して、サブスクライバーにアクセスさせたいソースのデータレイクに新しいデータが書き込まれたときにサブスクライバーに通知する方法を指定します。AWS Command Line Interface (AWS CLI) を使用している場合は、[create-subscriber-notification](#) コマンドを実行します。

- デフォルトの通知方法 (HTTPS エンドポイント) をオーバーライドして Amazon SQS キューを作成するには、`sqsNotificationConfiguration` パラメータの値を指定します。
- HTTPS エンドポイントによる通知を希望する場合は、`httpsNotificationConfiguration` パラメータの値を指定します。
- `targetRoleArn` フィールドには、EventBridge API 送信先を呼び出すために作成した IAM ロールの ARN を指定します。

```
$ aws securitylake create-subscriber-notification \  
--subscriber-id "12345ab8-1a34-1c34-1bd4-12345ab9012" \  
--configuration \  
httpsNotificationConfiguration={"targetRoleArn":"arn:aws:iam:XXX:role/service-  
role/RoleName", "endpoint":"https://account-management.$3.$2.securitylake.aws.dev/  
v1/dataLake"}
```

を取得するには `subscriberID`、Security Lake API の [ListSubscribers](#) オペレーションを使用します。AWS Command Line Interface (AWS CLI) を使用している場合は、[list-subscriber](#) コマンドを実行します。

```
$ aws securitylake list-subscribers
```

その後、サブスクライバーの通知方法 (Amazon SQS キューまたは HTTPS エンドポイント) を変更するには、[UpdateSubscriberNotification](#) オペレーションを使用するか、を使用している場合は [update-subscriber-notification](#) コマンド AWS CLI を実行します。Security Lake コンソールを使用して通知方法を変更することもできます。[サブスクライバー] ページでサブスクライバーを選択し、[編集] を選択します。

サンプル通知メッセージの例

次の例は、CreateSubscriberNotificationオペレーションのJSON構造形式のイベント通知を示しています。

```
{
  "source": "aws.s3",
  "time": "2021-11-12T00:00:00Z",
  "account": "123456789012",
  "region": "ca-central-1",
  "resources": [
    "arn:aws:s3:::amzn-s3-demo-bucket"
  ],
  "detail": {
    "bucket": {
      "name": "amzn-s3-demo-bucket"
    },
    "object": {
      "key": "example-key",
      "size": 5,
      "etag": "b57f9512698f4b09e608f4f2a65852e5"
    },
    "request-id": "N4N7GDK58NMKJ12R",
    "requester": "securitylake.amazonaws.com"
  }
}
```

Security Lake でのデータサブスクライバーの更新

サブスクライバーを更新するには、サブスクライバーが消費するソースを変更します。サブスクライバーのタグを割り当てたり編集したりすることもできます。タグは、サブスクライバーを含む特定のタイプのAWSリソースを定義して割り当てることができるラベルです。詳細については、「[Security Lake リソースのタグ付け](#)」を参照してください。

アクセス方法のいずれかを選択し、以下の手順に従って既存のサブスクリプションの新しいソースを定義します。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. 左のナビゲーションペインで [サブスクライバー] を選択します。

3. サブスクライバーを選択します。
4. [編集] を選択し、次のいずれかを実行します。
 - サブスクライバーのソースを更新するには、「ログとイベントのソース」セクションに新しい設定を入力します。
 - サブスクライバーにタグを割り当てたり編集したりするには、「タグ」セクションで必要に応じてタグを変更します。
5. 完了したら、保存 を選択します。

API

サブスクライバのデータアクセスソースをプログラムで更新するには、Security Lake API の [UpdateSubscriber](#) オペレーションを使用します。AWS Command Line Interface (AWS CLI) を使用している場合は、[update-subscriber](#) コマンドを実行します。リクエストでは、sourcesパラメータを使用して、サブスクライバーにアクセスさせたい各ソースを指定します。

```
$ aws securitylake update-subscriber --subscriber-id subscriber ID
```

特定の AWS アカウント または組織に関連付けられているサブスクライバーのリストについては、[ListSubscribers](#) オペレーションを使用します。AWS Command Line Interface (AWS CLI) を使用している場合は、[list-subscribers](#) コマンドを実行します。

```
$ aws securitylake list-subscribers
```

特定のサブスクライバーの現在の設定を確認するには、[GetSubscriber](#) オペレーションを使用します。[get-subscriber](#) コマンドを実行します。その後、Security Lake はサブスクライバーの名前と説明、外部 ID、その他の情報を返します。AWS Command Line Interface (AWS CLI) を使用している場合は、[get-subscriber](#) コマンドを実行します。

サブスクライバーの通知メソッドを更新するには、[UpdateSubscriberNotification](#) オペレーションを使用します。AWS Command Line Interface (AWS CLI) を使用している場合は、[update-subscriber-notification](#) コマンドを実行します。たとえば、サブスクライバーに新しい HTTPS エンドポイントを指定したり、HTTPS エンドポイントから Amazon SQS キューに切り替えたりできます。

Security Lake からのデータサブスクライバーの削除

Security Lake からのデータをサブスクライバーに使用させたくない場合は、以下の手順に従ってサブスクライバーを削除できます。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/>を開きます。
2. 左のナビゲーションペインで [サブスクライバー] を選択します。
3. 削除するサブスクライバーを選択します。
4. [削除] を選択してアクションを確認します。これにより、登録者と関連するすべての通知設定が削除されます。

API

シナリオに基づいて、次のいずれかを実行します。

- サブスクライバーと関連するすべての通知設定を削除するには、Security Lake API の [DeleteSubscriber](#) オペレーションを使用します。AWS Command Line Interface (AWS CLI) を使用している場合は、[delete-subscriber](#) コマンドを実行します。
- サブスクライバーを保持したまま、サブスクライバーへのfuture 通知を停止するには、Security Lake API の [DeleteSubscriberNotification](#) オペレーションを使用してください。AWS Command Line Interface (AWS CLI) を使用している場合は、[delete-subscriber-notification](#) コマンドを実行します。

Security Lake サブスクライバーのクエリアクセスの管理

クエリアクセス権を持つサブスクライバーは、Security Lake が収集するデータをクエリできます。これらのサブスクライバーは、Amazon Athena などのサービスを使用して S3 バケット内の AWS Lake Formation テーブルを直接クエリします。Security Lake の主なクエリ エンジン は Athena ですが、AWS Glue Data Catalog と統合された [Amazon Redshift Spectrum](#) や Spark SQL などの他のサービスも使用できます。

サブスクライバーは、Amazon Athena などのサービスを使用して、S3 バケット内の AWS Lake Formation テーブルからソースデータをクエリします。このサブスクリプション タイプは、[CreateSubscriber](#) API の `accessTypes` パラメーターで LAKEFORMATION として識別されます。

Note

このセクションでは、サードパーティのサブスクライバーにクエリアクセス権を付与する方法について説明します。独自のデータレイクに対してクエリを実行する方法については、「[ステップ 4: 独自のデータを表示してクエリする](#)」を参照してください。

トピック

- [Security Lake でクエリアクセス権を持つサブスクライバーを作成するための前提条件](#)
- [Security Lake でのクエリアクセス権を持つサブスクライバーの作成](#)
- [Security Lake でのクエリアクセス権を持つサブスクライバーの編集](#)

Security Lake でクエリアクセス権を持つサブスクライバーを作成するための前提条件

Security Lake でデータにアクセスできるサブスクライバーを作成する前に、次の必要条件を満たす必要があります。

アクセス許可の確認

クエリアクセス権限を持つサブスクライバーを作成する前に、以下のアクションリストを実行する権限があることを確認してください。

権限を確認するには、IAM を使用して IAM ID に添付されている IAM ポリシーを確認してください。クエリアクセスのあるサブスクライバを作成するために実行が許可される必要がある次のアクションのリストと比較します。

- `glue:PutResourcePolicy`
- `glue>DeleteResourcePolicy`
- `iam:CreateRole`
- `iam>DeleteRolePolicy`
- `iam:GetRole`
- `iam:PutRolePolicy`
- `lakeformation:GrantPermissions`

- lakeformation:ListPermissions
- lakeformation:RegisterResource
- lakeformation:RevokePermissions
- ram:GetResourceShareAssociations
- ram:GetResourceShares
- ram:UpdateResourceShare

Important

権限を確認したら:

- Security Lake コンソールを使用してクエリアクセス権限を持つサブスクライバーを追加する予定がある場合は、次のステップをスキップして [Lake Formation 管理者権限を付与](#)に進んでください。Security Lake は必要なすべての IAM ロールを作成するか、ユーザーに代わって既存のロールを使用します。
- Security Lake API または CLI を使用してクエリアクセス権限を持つサブスクライバーを追加する予定がある場合は、次のステップに進み、Security Lake データをクエリするための IAM ロールを作成します。

Security Lake データをクエリする IAM ロールを作成する (API および AWS CLI 専用ステップ)

Security Lake API または を使用してサブスクライバー AWS CLI にクエリアクセスを許可する場合は、 という名前のロールを作成する必要があります AmazonSecurityLakeMetaStoreManager。Security Lake は、このロールを使用して AWS Glue パーティションを登録し、AWS Glue テーブルを更新します。 [「必要な IAM ロールを作成する」](#) で既にこのロールを作成している場合があります。

Lake Formation 管理者権限を付与

また、Security Lake コンソールにアクセスしてサブスクライバーを追加するために使用する IAM ロールに Lake Formation 管理者権限を追加する必要があります。

次のステップに従って、自分のロールに Lake Formation 管理者権限を付与できます。

1. Lake Formation コンソール (<https://console.aws.amazon.com/lakeformation/>) を開きます。

2. 管理ユーザーとしてサインインする
3. [Lake Formation へようこそ] ウィンドウが表示されたら、ステップ 1 で作成または選択したユーザーを選択し、[開始する] を選択します。
4. [Lake Formation へようこそ] ウィンドウが表示されない場合は、以下の手順を実行して Lake Formation 管理者を設定します。
 1. ナビゲーションペインの [許可] で [管理ロールとタスク] を選択します。[データレイク管理者] セクションで、[管理者を選択] を選択します。
 2. データレイク管理者の管理ダイアログボックスの IAM ユーザーとロールで、Security Lake コンソールにアクセスするときに使用する管理者ロールを選択し、[保存] を選択します。

データレイク管理者の権限変更の詳細については、「AWS Lake Formation デベロッパーガイド」の「[データレイク管理者の作成](#)」を参照してください。

IAM ロールには、サブスクライバーにアクセス権を付与するデータベースとテーブルに対する SELECT 権限が必要です。その方法については、「AWS Lake Formation デベロッパーガイド」の「[名前付きリソースメソッドによる Data Catalog 権限の付与](#)」を参照してください。

Security Lake でのクエリアクセス権を持つサブスクライバーの作成

任意の方法を選択して、現在の でクエリアクセス権を持つサブスクライバーを作成します AWS リージョン。サブスクライバーは、AWS リージョン それで作成された からのみデータをクエリできます。サブスクライバーを作成するには、サブスクライバーの AWS アカウント ID と外部 ID が必要です。外部 ID は、サブスクライバーが提供する固有の識別子です。外部 IDs 「IAM [ユーザーガイド](#)」の「[AWS リソースへのアクセスを第三者に付与するときに外部 ID を使用する方法](#)」を参照してください。

Note

セキュリティレイクは、Lake Formation のクロスアカウントデータ共有バージョン 1 をサポートしていません。Lake Formation のクロスアカウントデータ共有をバージョン 2 またはバージョン 3 に更新する必要があります。AWS Lake Formation コンソールまたは AWS CLI を使用してクロスアカウントバージョン設定を更新する手順については、「[デベロッパーガイド](#)」の「[新しいバージョンを有効にするには](#)」を参照してください。AWS Lake Formation

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。

委任管理者アカウントにサインインします。

2. ページの右上隅にある AWS リージョン セレクターを使用して、サブスクライバーを作成するリージョンを選択します。
3. 左のナビゲーションペインで [サブスクライバー] を選択します。
4. 「サブスクライバー」ページで、「サブスクライバーを作成」を選択します。
5. サブスクライバーの詳細には、サブスクライバー名とオプションの説明を入力します。

リージョンは、現在選択されているとして自動入力 AWS リージョン され、変更できません。

6. [ログとイベントのソース] では、クエリ結果を返すときに Security Lake に含めたいソースを選択します。
7. [データアクセス方法] では、[Lake Formation] を選択してサブスクライバーのクエリアクセスを作成します。
8. サブスクライバーの認証情報には、サブスクライバーの AWS アカウント ID と [外部 ID](#) を指定します。
9. (オプション) [タグ] には、サブスクライバーに割り当てるタグを 50 個まで入力します。

タグは、特定のタイプの AWS リソースを定義して割り当てることができるラベルです。各タグは、必要なタグキーとオプションのタグ値で設定されています。タグは、さまざまな方法でリソースの識別、分類、管理に役立ちます。詳細については、「[Security Lake リソースのタグ付け](#)」を参照してください。

10. [作成] を選択します。

API

クエリアクセス付きのサブスクライバーをプログラムで作成するには、Security Lake API の [CreateSubscriber](#) オペレーションを使用します。AWS Command Line Interface (AWS CLI) を使用している場合は、[create-subscriber](#) コマンドを実行します。

リクエストでは、これらのパラメータを使用してサブスクライバーに次の設定を指定します。

- `accessTypes` の場合、LAKEFORMATION を指定します。

- sourcesでは、クエリ結果を返すときに Security Lake に含めたいソースをそれぞれ指定します。
- にはsubscriberIdentity、サブスクライバーがソースデータのクエリに使用する AWS ID と外部 ID を指定します。

次の の例では、指定されたサブスクライバー ID の現在の AWS リージョンにクエリアクセス権を持つサブスクライバーを作成します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake create-subscriber \  
--subscriber-identity {"accountID": 129345678912,"externalId": 123456789012} \  
--sources [{"awsLogSource": {"sourceName": VPC_FLOW, "sourceVersion": 2.0}}] \  
--subscriber-name subscriber name \  
--access-types LAKEFORMATION
```

クロスアカウントテーブル共有セットアップ (サブスクライバーステップ)

Security Lake は、Lake Formation のクロスアカウントテーブル共有を使用してサブスクライバーのクエリアクセスをサポートします。Security Lake コンソール、API、または でクエリアクセス権を持つサブスクライバーを作成すると AWS CLI、Security Lake は AWS Resource Access Manager () で [リソース共有](#)を作成して、関連する Lake Formation テーブルに関する情報をサブスクライバーと共有しますAWS RAM。

クエリアクセス権を持つサブスクライバに特定の種類の編集を行うと、Security Lake は新しいリソース共有を作成します。詳細については、「[Security Lake でのクエリアクセス権を持つサブスクライバーの編集](#)」を参照してください。

サブスクライバーは、次のステップに従って Lake Formation テーブルからデータを取得する必要があります。

1. リソース共有を受け入れる — サブスクライバーは、サブスクライバーを作成または編集したときに生成される、resourceShareArnとresourceShareNameを含むリソース共有を受け入れる必要があります。次のいずれかのアクセス方法を選択します。
 - コンソールおよび については AWS CLI、[「からのリソース共有の招待の承諾 AWS RAM」](#)を参照してください。

- API の場合は、[GetResourceShareInvitations](#) API を呼び出します。resourceShareArnとresourceShareNameでフィルタリングして、正しいリソースシェアを見つけてください。[AcceptResourceShareInvitation](#) API を使用して招待を受け入れます。

リソース共有の招待は 12 時間で期限切れになるため、12 時間以内に招待を検証して承諾する必要があります。招待の有効期限が切れても、引き続き PENDING 状態で表示されますが、招待を受け入れても共有リソースにアクセスできなくなります。12 時間以上経過したら、Lake Formation サブスクライバーを削除し、サブスクライバーを再作成して新しいリソース共有の招待状を取得します。

2. 共有データベースへのリソースリンクを作成する – サブスクライバーは、共有 Lake Formation データベースへのリソースリンクを AWS Lake Formation (コンソールを使用している場合) または AWS Glue (API/AWS CLI を使用している場合) で作成する必要があります。このリソースリンクは、サブスクライバーのアカウントを共有データベースにポイントします。次のいずれかのアクセス方法を選択します。

- コンソールおよびについては AWS CLI、[「デベロッパーガイド」の「共有 Data Catalog データベースへのリソースリンクの作成」](#)を参照してください。AWS Lake Formation
- 利用者は、[CreateDatabase](#) API を使用してリソースリンクテーブルを格納する独自のデータベースも作成することをお勧めします。

3. 共有テーブルをクエリする — Amazon Athena などのサービスはテーブルを直接参照でき、Security Lake が収集した新しいデータを自動的にクエリに使用できるようになります。クエリはサブスクライバーの で実行され AWS アカウント、クエリによって発生したコストはサブスクライバーに請求されます。自分の Security Lake アカウントのリソースへの読み取りアクセスを制御できます。

クロスアカウント権限の付与については詳しくは、「AWS Lake Formation デベロッパーガイド」の [「Lake Formation でのクロスアカウントデータ共有」](#)を参照してください。

Security Lake でのクエリアクセス権を持つサブスクライバーの編集

Security Lake では、クエリアクセス権を持つサブスクライバーの編集がサポートされます。サブスクライバーの名前、説明、外部 ID、プリンシパル (AWS アカウント ID)、およびサブスクライバーが使用できるログソースを編集できます。希望の方法を選択し、ステップに従って現在の AWS リージョンでクエリ アクセス権を持つサブスクライバを編集します。

Note

セキュリティレイクは、Lake Formation のクロスアカウントデータ共有バージョン 1 をサポートしていません。Lake Formation のクロスアカウントデータ共有をバージョン 2 またはバージョン 3 に更新する必要があります。AWS Lake Formation コンソールまたは AWS CLI を使用してクロスアカウントバージョン設定を更新する手順については、「[デベロッパーガイド](#)」の「[新しいバージョンを有効にする](#)」を参照してください。AWS Lake Formation

Console

編集したい詳細に基づいて、そのアクションにのみ記載されているステップに従ってください。

サブスクライバー名を編集するには

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
委任管理者アカウントにサインインします。
2. ページの右上隅にある AWS リージョン セレクターを使用して、サブスクライバーの詳細を編集するリージョンを選択します。
3. 左のナビゲーションペインで [サブスクライバー] を選択します。
4. サブスクライバーページで、ラジオボタンを使用して編集するサブスクライバーを選択します。選択したサブスクライバーのデータアクセス方法は LAKEFORMATION でなければなりません。
5. [編集] を選択します。
6. 新しいサブスクライバー名を入力し、[保存] を選択します。

サブスクライバーの説明を編集するには

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
委任管理者アカウントにサインインします。
2. ページの右上隅にある AWS リージョン セレクターを使用して、サブスクライバーを編集するリージョンを選択します。
3. 左のナビゲーションペインで [サブスクライバー] を選択します。

4. サブスクライバーページで、ラジオボタンを使用して編集するサブスクライバーを選択します。選択したサブスクライバのデータアクセス方法は LAKEFORMATION でなければなりません。
5. [編集] を選択します。
6. サブスクライバーの新しい説明を入力し、[保存] を選択します。

外部 ID を編集するには

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
委任管理者アカウントにサインインします。
2. ページの右上隅にある AWS リージョン セレクターを使用して、サブスクライバーの詳細を編集するリージョンを選択します。
3. 左のナビゲーションペインで [サブスクライバー] を選択します。
4. サブスクライバーページで、ラジオボタンを使用して編集するサブスクライバーを選択します。選択したサブスクライバのデータアクセス方法は LAKEFORMATION でなければなりません。
5. [編集] を選択します。
6. 加入者が提供した新しい外部 ID を入力し、[Save] を選択します。

新しい外部 ID を保存すると、以前の AWS RAM リソース共有が自動的に削除され、サブスクライバーの新しいリソース共有が作成されます。

7. サブスクライバーは、[クロスアカウントテーブル共有セットアップ \(サブスクライバーステップ\)](#) のステップ 1 に従って新しいリソースシェアを受け入れる必要があります。サブスクライバーの詳細に表示される Amazon リソースネーム (ARN) が Lake Formation コンソールと同じであることを確認してください。共有テーブルへのリソースリンクはそのまま残るため、サブスクライバーは新しいリソースリンクを作成する必要はありません。

プリンシパル (AWS アカウント ID) を編集するには

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
委任管理者アカウントにサインインします。
2. ページの右上隅にある AWS リージョン セレクターを使用して、サブスクライバーの詳細を編集するリージョンを選択します。

3. 左のナビゲーションペインで [サブスクリイバー] を選択します。
4. サブスクリイバーページで、ラジオボタンを使用して編集するサブスクリイバーを選択します。選択したサブスクリイバのデータアクセス方法は LAKEFORMATION でなければなりません。
5. [編集] を選択します。
6. サブスクリイバーの新しい AWS アカウント ID を入力し、[Save] を選択します。

新しいアカウント ID を保存すると、以前の AWS RAM リソース共有が自動的に削除されるため、以前のプリンシパルはログソースとイベントソースを消費できなくなります。Security Lake は新しいリソース共有を作成します。

7. サブスクリイバーは、新しいプリンシパルの認証情報を使用して、新しいリソース共有を受け入れ、共有テーブルへのリソースリンクを作成する必要があります。これにより、新しいプリンシパルは共有リソースにアクセスできます。ステップについては、「[クロスアカウントテーブル共有セットアップ \(サブスクリイバーステップ\)](#)」のステップ1と2を参照してください。サブスクリイバーの詳細に表示される ARN が Lake Formation コンソールに表示される ARN と同じであることを確認してください。

ログソースとイベントソースを編集するには

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。

委任管理者アカウントにサインインします。

2. ページの右上隅にある AWS リージョン セレクターを使用して、サブスクリイバーの詳細を編集するリージョンを選択します。
3. 左のナビゲーションペインで [サブスクリイバー] を選択します。
4. サブスクリイバーページで、ラジオボタンを使用して編集するサブスクリイバーを選択します。選択したサブスクリイバのデータアクセス方法は LAKEFORMATION でなければなりません。
5. [編集] を選択します。
6. 既存のソースを選択解除するか、追加するソースを選択します。ソースを選択解除すると、追加のアクションは必要ありません。ソースを追加することを選択した場合、新しいリソース共有への招待は作成されません。ただし、Security Lake は、追加されたソースに基づいて共有 Lake Formation テーブルを更新します。サブスクリイバーは、ソースデータをクエリできるように、更新された共有テーブルへのリソースリンクを作成する必

必要があります。ステップについては、「[クロスアカウントテーブル共有セットアップ \(サブスクライバーステップ\)](#)」のステップ 2を参照してください。

7. [保存] を選択します。

API

クエリアクセス権限を持つサブスクライバーをプログラムで編集するには、Security Lake API の [UpdateSubscriber](#) オペレーションを使用してください。AWS Command Line Interface (AWS CLI) を使用している場合は、[update-subscriber](#) コマンドを実行します。リクエストでは、サポートされているパラメータを使用して、サブスクライバーの次の設定を指定します。

- `subscriberName`には、新しいサブスクライバー名を指定します。
- `subscriberDescription`には、新しい説明を指定します。
- `identity`には`subscriberIdentity`、サブスクライバーがソースデータのクエリに使用するプリンシパル (AWS アカウント ID) と外部 ID を指定します。プリンシパル ID と外部 ID の両方を指定する必要があります。これらの値のいずれかを同じままにしておきたい場合は、現在の値を渡してください。
- 外部 ID のみの更新 — このアクションは以前の AWS RAM リソース共有を削除し、サブスクライバー用の新しいリソース共有を作成します。サブスクライバーは、[クロスアカウントテーブル共有セットアップ \(サブスクライバーステップ\)](#)のステップ 1 に従って新しいリソースシェアを受け入れる必要があります。共有テーブルへのリソースリンクはそのまま残るため、サブスクライバーは新しいリソースリンクを作成する必要はありません。
- プリンシパルのみの更新 – このアクションは、以前のプリンシパルがログソースとイベントソースを消費できないように、以前の AWS RAM リソース共有を削除します。Security Lake は新しいリソース共有を作成します。サブスクライバーは、新しいプリンシパルの認証情報を使用して、新しいリソース共有を受け入れ、共有テーブルへのリソースリンクを作成する必要があります。これにより、新しいプリンシパルは共有リソースにアクセスできます。ステップについては、「[クロスアカウントテーブル共有セットアップ \(サブスクライバーステップ\)](#)」のステップ 1と2 を参照してください。

外部 ID とプリンシパルを更新するには、[クロスアカウントテーブル共有セットアップ \(サブスクライバーステップ\)](#)のステップ 1 と 2 に従ってください。

- `sources`には、既存のソースを削除するか、追加するソースを指定します。ソースを削除する場合、追加のアクションは必要ありません。ソースを追加しても、新しいリソース共有への招待は作成されません。ただし、Security Lake は、追加されたソースに基づいて共有Lake Formation テーブルを更新します。サブスクライバーは、ソースデータをクエリできるよう

に、更新された共有テーブルへのリソースリンクを作成する必要があります。ステップについては、「[クロスアカウントテーブル共有セットアップ \(サブスクライバーステップ\)](#)」のステップ 2 を参照してください。

Security Lake クエリ

Security Lake が AWS Lake Formation データベースとテーブルに保存するデータをクエリできます。Security Lake コンソール、API、または AWS CLI でサードパーティのサブスクライバーを作成することもできます。サードパーティのサブスクライバーは、指定したソースから Lake Formation データをクエリすることもできます。

Lake Formation データレイク管理者は、データをクエリする IAM ID に、関連するデータベースとテーブルに対する SELECT 権限を付与する必要があります。また、データをクエリする前に Security Lake にサブスクライバーを作成する必要があります。クエリ アクセスを持つサブスクライバを作成する方法の詳細については、「[Security Lake サブスクライバーのクエリアクセスの管理](#)」を参照してください。

保持設定を使用したデータのクエリ

[Amazon S3 ライフサイクル設定](#)は、データの保持期間に影響し、クエリを実行できる時間に影響します。Security Lake で保持設定を設定している場合は、クエリに時間ベースのフィルターを含めて、結果セットの有効期限が切れていないデータファイルの範囲に設定する必要があります。Security Lake のデータ保持の詳細については、「」を参照してください [ライフサイクル管理](#)。

以下のセクションのクエリ例には、このベストプラクティスを示すために `time_dt`、`eventDay` やなどの時間ベースのフィルターが含まれています。

トピック

- [AWS ソースバージョン 1 の Security Lake クエリ \(OCSF 1.0.0-rc.2\)](#)
- [AWS ソースバージョン 2 \(OCSF 1.1.0\) の Security Lake クエリ](#)

AWS ソースバージョン 1 の Security Lake クエリ (OCSF 1.0.0-rc.2)

次のセクションでは、Security Lake からのデータのクエリに関するガイダンスを提供し、AWS ソースバージョン 1 のネイティブにサポートされている AWS ソースのクエリ例をいくつか示します。これらのクエリは、特定の のデータを取得するように設計されています AWS リージョン。これらの例では `us-east-1` (米国東部 (バージニア北部)) を使用しています。さらに、サンプルクエリで

は最大 25 件のレコードを返す LIMIT 25 パラメータを使用しています。このパラメータは省略することも、好みに応じて調整することもできます。その他の例については、[Amazon Security Lake OCSF Queries GitHub ディレクトリ](#)を参照してください。

次のクエリには、クエリが設定された保持設定内であることを確認する eventDay ために を使用する時間ベースのフィルターが含まれています。詳細については、「[Querying data with retention settings](#)」を参照してください。

例えば、60 日以上経過したデータの有効期限が切れている場合、クエリには有効期限が切れたデータにアクセスできないように時間制約を含める必要があります。60 日間の保持期間については、クエリに次の句を含めます。

```
...
WHERE eventDay BETWEEN cast(date_format(current_date - INTERVAL '59' day, '%Y%m%d') AS
  varchar)
      AND cast(date_format(current_date, '%Y%m%d') AS varchar)
...
```

この句は、Amazon S3 と Apache Iceberg 間のデータまたは時間の重複を避けるために 59 日 (60 日ではなく) を使用します。

ログソーステーブル

Security Lake データをクエリするときは、データが存在する Lake Formation テーブルの名前を含める必要があります。

```
SELECT *
FROM
amazon_security_lake_glue_db_<DB_Region>.amazon_security_lake_table_<DB_Region>_SECURITY_LAKE_TABLE
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
LIMIT 25
```

ログソーステーブルの一般的な値には次のようなものがあります。

- cloud_trail_mgmt_1_0 – AWS CloudTrail 管理イベント
- lambda_execution_1_0 – Lambda CloudTrail データイベント

- s3_data_1_0 - S3 CloudTrail データイベント
- route53_1_0 - Amazon Route 53 Resolver クエリログ
- sh_findings_1_0 - 検出 AWS Security Hub 結果
- vpc_flow_1_0 - Amazon Virtual Private Cloud (Amazon VPC) フローログ

例:us-east-1 リージョンのテーブルsh_findings_1_0内のすべてのSecurity Hub 結果

```
SELECT *
  FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_sh_findings_1_0
  WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
  LIMIT 25
```

データベースリージョン

Security Lake データをクエリするときは、データのクエリ元のデータベースリージョンの名前を含める必要があります。Security Lakeが現在利用可能なデータベースリージョンの全リストについては、「[Amazon Security Lake エンドポイント](#)」を参照してください。

例: ソース IP からの AWS CloudTrail アクティビティを一覧表示する

次の例では、us-east-1DB_Regionのテーブル cloud_trail_mgmt_1_0 に、20230301 (2023年3月1日)以降に記録されたソース IP 192.0.2.1 からのすべての CloudTrail アクティビティをリストします。

```
SELECT *
  FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_cloud_trail_mgmt_1_0
  WHERE eventDay > '20230301' AND src_endpoint.ip = '192.0.2.1'
  ORDER BY time desc
  LIMIT 25
```

パーティション日付

データをパーティションすることで、各クエリによってスキャンされるデータの量を制限できるようになるため、パフォーマンスが向上し、コストが削減されます。Security Lake はeventDay、region、accountidパラメータを通じて、パーティショニングを実装します。eventDayパーティションではYYYYMMDD形式を使用します。

これはeventDayパーティションを使ったクエリの例です。

```
SELECT *
  FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_cloud_trail_mgmt_1
 WHERE eventDay > '20230301'
 AND src_endpoint.ip = '192.0.2.1'
 ORDER BY time desc
```

eventDayの一般的な値は次の通りです。

過去1年間に発生したイベント

```
> cast(date_format(current_timestamp - INTERVAL '1' year, '%Y%m%d%H') as
varchar)
```

過去1か月に発生したイベント

```
> cast(date_format(current_timestamp - INTERVAL '1' month, '%Y%m%d%H')
as varchar)
```

過去30日間に発生したイベント

```
> cast(date_format(current_timestamp - INTERVAL '30' day, '%Y%m%d%H') as
varchar)
```

過去12時間に発生したイベント

```
> cast(date_format(current_timestamp - INTERVAL '12' hour, '%Y%m%d%H')
as varchar)
```

過去5分間に発生したイベント

```
> cast(date_format(current_timestamp - INTERVAL '5' minute, '%Y%m%d%H')
as varchar)
```

7 ~ 14 日前に発生したイベント

```
BETWEEN cast(date_format(current_timestamp - INTERVAL '14' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar)
```

特定の日付以降に発生するイベント

```
>= '20230301'
```

例:2023 年 3 月 1 日以降のソース IP**192.0.2.1** からのすべての CloudTrail アクティビティのリスト (表) **cloud_trail_mgmt_1_0**

```
SELECT *
  FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_cloud_trail_mgmt_1_0
 WHERE eventDay >= '20230301'
 AND src_endpoint.ip = '192.0.2.1'
 ORDER BY time desc
 LIMIT 25
```

例: テーブル **cloud_trail_mgmt_1_0** 内の過去 30 日間のソース IP **192.0.2.1** からのすべての CloudTrail アクティビティのリスト

```
SELECT *
  FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_cloud_trail_mgmt_1_0
 WHERE eventDay > cast(date_format(current_timestamp - INTERVAL '30' day, '%Y%m%d%H') as varchar)
 AND src_endpoint.ip = '192.0.2.1'
 ORDER BY time desc
 LIMIT 25
```

CloudTrail データの Security Lake クエリの例

AWS CloudTrail は、でのユーザーアクティビティと API の使用を追跡します AWS のサービス。サブスクライバーは CloudTrail データにクエリを実行して、次の種類の情報を取得できます。

AWS ソースバージョン 1 の CloudTrail データのクエリの例を次に示します。

過去 7 日間の AWS のサービス に対する不正な試行

```
SELECT
    time,
    api.service.name,
    api.operation,
    api.response.error,
    api.response.message,
    unmapped['responseElements'],
    cloud.region,
    actor.user.uuid,
    src_endpoint.ip,
    http_request.user_agent
FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_cloud_trail_mgmt_1
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
AND api.response.error in (
    'Client.UnauthorizedOperation',
    'Client.InvalidPermission.NotFound',
    'Client.OperationNotPermitted',
    'AccessDenied')
ORDER BY time desc
LIMIT 25
```

過去 7 日間のソース IP **192.0.2.1** からのすべての CloudTrail アクティビティのリスト

```
SELECT
    api.request.uid,
    time,
    api.service.name,
    api.operation,
    cloud.region,
    actor.user.uuid,
    src_endpoint.ip,
    http_request.user_agent
FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_cloud_trail_mgmt_1
```

```
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
AND src_endpoint.ip = '127.0.0.1.'
ORDER BY time desc
LIMIT 25
```

過去 7 日間のすべての IAM アクティビティのリスト

```
SELECT *
FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_cloud_trail_mgmt_1
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
AND api.service.name = 'iam.amazonaws.com'
ORDER BY time desc
LIMIT 25
```

過去 7 **AIDACKCEVSQ6C2EXAMPLE** 日間に認証情報が使用されたインスタンス

```
SELECT
actor.user.uid,
actor.user.uuid,
actor.user.account_uid,
cloud.region
FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_cloud_trail_mgmt_1
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
AND actor.user.credential_uid = 'AIDACKCEVSQ6C2EXAMPLE'
LIMIT 25
```

過去 7 日間に失敗した CloudTrail レコードのリスト

```
SELECT
actor.user.uid,
actor.user.uuid,
actor.user.account_uid,
cloud.region
```

```
FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_cloud_trail_mgmt_1
WHERE status='failed' and eventDay BETWEEN cast(date_format(current_timestamp -
INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp -
INTERVAL '0' day, '%Y%m%d%H') as varchar)
ORDER BY time DESC
LIMIT 25
```

Route 53 リゾルバークエリログの Security Lake クエリの例

Amazon Route 53 リゾルバーのクエリログは、Amazon VPC 内のリソースによって行われた DNS クエリを追跡します。利用者は Route 53 リゾルバーのクエリログをクエリして、次の種類の情報を確認できます。

AWS ソースバージョン 1 の Route 53 リゾルバークエリログのクエリの例を次に示します。

過去 7 日間の CloudTrail からの DNS クエリのリスト

```
SELECT
    time,
    src_endpoint.instance_uid,
    src_endpoint.ip,
    src_endpoint.port,
    query.hostname,
    rcode
FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_route53_1_0
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m
%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d
%H') as varchar)
ORDER BY time DESC
LIMIT 25
```

過去 7 日間に **s3.amazonaws.com** と一致した DNS クエリのリスト

```
SELECT
    time,
    src_endpoint.instance_uid,
    src_endpoint.ip,
    src_endpoint.port,
    query.hostname,
```

```

    rcode,
    answers
FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_route53_1_0
WHERE query.hostname LIKE 's3.amazonaws.com.' and eventDay BETWEEN
cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and
cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
ORDER BY time DESC
LIMIT 25

```

過去 7 日間に解決されなかった DNS クエリのリスト

```

SELECT
    time,
    src_endpoint.instance_uid,
    src_endpoint.ip,
    src_endpoint.port,
    query.hostname,
    rcode,
    answers
FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_route53_1_0
WHERE cardinality(answers) = 0 and eventDay BETWEEN
cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and
cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
LIMIT 25

```

過去 7 日間に**192.0.2.1**に解決された DNS クエリのリスト

```

SELECT
    time,
    src_endpoint.instance_uid,
    src_endpoint.ip,
    src_endpoint.port,
    query.hostname,
    rcode,
    answer.rdata
FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_route53_1_0
CROSS JOIN UNNEST(answers) as st(answer)
WHERE answer.rdata='192.0.2.1' and eventDay BETWEEN
cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and
cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)

```

LIMIT 25

Security Hub の検出結果に対する Security Lake クエリの例

Security Hub は、のセキュリティ状態を包括的に把握 AWS し、セキュリティ業界標準とベストプラクティスに照らして環境をチェックするのに役立ちます。Security Hub はセキュリティチェック用の結果を生成し、サードパーティのサービスから結果を受け取ります。

Security Hub 調査結果のクエリの例は次のとおりです。

過去 7 日間の **MEDIUM** と同等かそれ以上の新しい検出結果

```
SELECT
    time,
    finding,
    severity
FROM
    amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_sh_findings_1_0_fi
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m
%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d
%H') as varchar)
    AND severity_id >= 3
    AND state_id = 1
ORDER BY time DESC
LIMIT 25
```

過去 7 日間の重複する検出結果

```
SELECT
    finding.uid,
    MAX(time) AS time,
    ARBITRARY(region) AS region,
    ARBITRARY(accountid) AS accountid,
    ARBITRARY(finding) AS finding,
    ARBITRARY(vulnerabilities) AS vulnerabilities
FROM
    amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_sh_findings_1_0
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d
%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H')
    as varchar)
GROUP BY finding.uid
```

```
LIMIT 25
```

過去 7 日間の情報提供以外のすべての調査結果

```
SELECT
    time,
    finding.title,
    finding,
    severity
FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_sh_findings_1_0
WHERE severity != 'Informational' and eventDay BETWEEN
cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and
cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
LIMIT 25
```

リソースが Amazon S3 バケットである場合の結果 (時間制限なし)

```
SELECT *
FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_sh_findings_1_0
WHERE any_match(resources, element -> element.type = 'amzn-s3-demo-bucket')
LIMIT 25
```

Common Vulnerability Scoring System (CVSS) スコアが 1 (時間制限なし) 以上の検出結果

```
SELECT *
FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_sh_findings_1_0
WHERE any_match(vulnerabilities, element -> element.cve.cvss.base_score > 1.0)
LIMIT 25
```

一般的な脆弱性と露出 (CVE) **CVE-0000-0000** に該当する検出結果 (時間制限なし)

```
SELECT *
FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_sh_findings_1_0
WHERE any_match(vulnerabilities, element -> element.cve.uid = 'CVE-0000-0000')
LIMIT 25
```

過去 7 日間に Security Hub から結果を送信した製品の数

```
SELECT
    metadata.product.feature.name,
    count(*)
FROM
    amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_sh_findings_1_0
    WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
    GROUP BY metadata.product.feature.name
    ORDER BY metadata.product.feature.name DESC
    LIMIT 25
```

過去 7 日間の結果に含まれるリソースタイプの数

```
SELECT
    count(*),
    resource.type
FROM
    amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_sh_findings_1_0
    CROSS JOIN UNNEST(resources) as st(resource)
    WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
    GROUP BY resource.type
    LIMIT 25
```

過去 7 日間の検出結果から得られた脆弱なパッケージ

```
SELECT
    vulnerability
FROM
    amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_sh_findings_1_0,
    UNNEST(vulnerabilities) as t(vulnerability)
    WHERE vulnerabilities is not null
    LIMIT 25
```

過去 7 日間に変更された調査結果

```
SELECT
    finding.uid,
    finding.created_time,
```

```
finding.first_seen_time,  
finding.last_seen_time,  
finding.modified_time,  
finding.title,  
state  
FROM  
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_sh_findings_1_0  
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m  
%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d  
%H') as varchar)  
LIMIT 25
```

Amazon VPC フローログの Security Lake クエリの例

Amazon Virtual Private Cloud (Amazon VPC) は、VPC 内のネットワーク インターフェイスとの間で送受信される IP トラフィックに関する詳細を提供します。

AWS ソースバージョン 1 の Amazon VPC フローログのクエリの例を次に示します。

過去 7 日間の特定の AWS リージョン のトラフィック

```
SELECT *  
FROM  
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_vpc_flow_1_0  
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m  
%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d  
%H') as varchar)  
AND region in ('us-east-1', 'us-east-2', 'us-west-2')  
LIMIT 25
```

過去 7 日間の送信元 IP **192.0.2.1** と送信元ポート**22**のアクティビティのリスト。

```
SELECT *  
FROM  
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_vpc_flow_1_0  
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m  
%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d  
%H') as varchar)  
AND src_endpoint.ip = '192.0.2.1'  
AND src_endpoint.port = 22  
LIMIT 25
```

過去 7 日間の個別の宛先 IP アドレスの数

```
SELECT
  COUNT(DISTINCT dst_endpoint.ip)
FROM
  amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_vpc_flow_1_0
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
LIMIT 25
```

過去 7 日間に 198.51.100.0/24 から発生したトラフィック

```
SELECT *
FROM
  amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_vpc_flow_1_0
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
AND split_part(src_endpoint.ip, '.', 1)='198'AND split_part(src_endpoint.ip, '.', 2)='51'
LIMIT 25
```

過去 7 日間のすべての HTTPS トラフィック

```
SELECT
  dst_endpoint.ip as dst,
  src_endpoint.ip as src,
  traffic.packets
FROM
  amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_vpc_flow_1_0
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
AND dst_endpoint.port = 443
GROUP BY
  dst_endpoint.ip,
  traffic.packets,
  src_endpoint.ip
ORDER BY traffic.packets DESC
LIMIT 25
```

過去 7 日間のポート**443**宛ての接続のパケット数順

```
SELECT
    traffic.packets,
    dst_endpoint.ip
FROM
    amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_vpc_flow_1_0
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
AND dst_endpoint.port = 443
GROUP BY
    traffic.packets,
    dst_endpoint.ip
ORDER BY traffic.packets DESC
LIMIT 25
```

過去 7 日間の IP **192.0.2.1** と **192.0.2.2** 間のすべてのトラフィック

```
SELECT
    start_time,
    end_time,
    src_endpoint.interface_uid,
    connection_info.direction,
    src_endpoint.ip,
    dst_endpoint.ip,
    src_endpoint.port,
    dst_endpoint.port,
    traffic.packets,
    traffic.bytes
FROM
    amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_vpc_flow_1_0
WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
AND(
    src_endpoint.ip = '192.0.2.1'
    AND dst_endpoint.ip = '192.0.2.2')
OR (
    src_endpoint.ip = '192.0.2.2'
    AND dst_endpoint.ip = '192.0.2.1')
ORDER BY start_time ASC
LIMIT 25
```

過去 7 日間のすべてのインバウンドトラフィック

```
SELECT *
  FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_vpc_flow_1_0
  WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
  AND connection_info.direction = 'ingress'
  LIMIT 25
```

過去 7 日間のすべてのアウトバウンドトラフィック

```
SELECT *
  FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_vpc_flow_1_0
  WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
  AND connection_info.direction = 'egress'
  LIMIT 25
```

過去 7 日間に拒否されたすべてのトラフィック

```
SELECT *
  FROM
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_vpc_flow_1_0
  WHERE eventDay BETWEEN cast(date_format(current_timestamp - INTERVAL '7' day, '%Y%m%d%H') as varchar) and cast(date_format(current_timestamp - INTERVAL '0' day, '%Y%m%d%H') as varchar)
  AND type_uid = 400105
  LIMIT 25
```

AWS ソースバージョン 2 (OCSF 1.1.0) の Security Lake クエリ

次のセクションでは、Security Lake からのデータのクエリに関するガイダンスを提供し、AWS ソースバージョン 2 のネイティブにサポートされている AWS ソースのクエリ例をいくつか示します。これらのクエリは、特定の のデータを取得するように設計されています AWS リージョン。これらの例では us-east-1 (米国東部 (バージニア北部)) を使用しています。さらに、サンプルクエリで

は最大 25 件のレコードを返す LIMIT 25 パラメータを使用しています。このパラメータは省略することも、好みに応じて調整することもできます。その他の例については、[Amazon Security Lake OCSF Queries GitHub ディレクトリ](#)を参照してください。

Security Lake が AWS Lake Formation データベースとテーブルに保存するデータをクエリできます。Security Lake コンソール、API、または AWS CLI でサードパーティのサブスクライバーを作成することもできます。サードパーティのサブスクライバーは、指定したソースから Lake Formation データをクエリすることもできます。

Lake Formation データレイク管理者は、データをクエリする IAM ID に、関連するデータベースとテーブルに対する SELECT 権限を付与する必要があります。また、データをクエリする前に Security Lake にサブスクライバーを作成する必要があります。クエリ アクセスを持つサブスクライバを作成する方法の詳細については、「[Security Lake サブスクライバーのクエリアクセスの管理](#)」を参照してください。

次のクエリには、クエリが設定された保持設定内であることを確認する eventDay ために を使用する時間ベースのフィルターが含まれています。詳細については、「[Querying data with retention settings](#)」を参照してください。

例えば、60 日以上経過したデータの有効期限が切れている場合、クエリには有効期限が切れたデータにアクセスできないように時間制約を含める必要があります。60 日間の保持期間については、クエリに次の句を含めます。

```
...
WHERE time_dt > DATE_ADD('day', -59, CURRENT_TIMESTAMP)
...
```

この句は、Amazon S3 と Apache Iceberg 間のデータまたは時間の重複を避けるために 59 日 (60 日ではなく) を使用します。

ログソーステーブル

Security Lake データをクエリするときは、データが存在する Lake Formation テーブルの名前を含める必要があります。

```
SELECT *
FROM
  "amazon_security_lake_glue_db_DB_Region"."amazon_security_lake_table_DB_Region_SECURITY_LAKE_T
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
LIMIT 25
```

ログソーステーブルの一般的な値には次のようなものがあります。

- `cloud_trail_mgmt_2_0` – AWS CloudTrail 管理イベント
- `lambda_execution_2_0` – Lambda CloudTrail データイベント
- `s3_data_2_0` – S3 CloudTrail データイベント
- `route53_2_0` – Amazon Route 53 Resolver クエリログ
- `sh_findings_2_0` – 検出 AWS Security Hub 結果
- `vpc_flow_2_0` – Amazon Virtual Private Cloud (Amazon VPC) フローログ
- `eks_audit_2_0` – Amazon Elastic Kubernetes Service (Amazon EKS) 監査ログ
- `waf_2_0` – AWS WAF v2 ログ

例: `us-east-1` リージョンのテーブル `sh_findings_2_0` 内のすべての Security Hub 結果

```
SELECT *
  FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_sh_findings_2_0"
  WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
 LIMIT 25
```

データベースリージョン

Security Lake データをクエリするときは、データのクエリ元のデータベースリージョンの名前を含める必要があります。Security Lake が現在利用可能なデータベースリージョンの全リストについては、「[Amazon Security Lake エンドポイント](#)」を参照してください。

例: ソース IP から Amazon Virtual Private Cloud アクティビティを一覧表示する

次の例では、`us-west-2 ##### vpc_flow_2_0 ####(2023 # 3 # 1 #) #####`
`# IP 192.0.2.1` からのすべての Amazon VPC アクティビティを一覧表示します DB_Region。
`20230301`

```
SELECT *
  FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_vpc_flow_2_0"
  WHERE time_dt > TIMESTAMP '2023-03-01'
  AND src_endpoint.ip = '192.0.2.1'
  ORDER BY time_dt desc
 LIMIT 25
```

パーティション日付

データをパーティションすることで、各クエリによってスキャンされるデータの量を制限できるようになるため、パフォーマンスが向上し、コストが削減されます。Security Lake 2.0 では、Security Lake 1.0 とパーティションの動作が若干異なります。Security Lake は、time_dt、region および accountid によるパーティショニングを実装するようになりました。一方、Security Lake 1.0 では、eventDay、region および accountid パラメータによるパーティショニングを実装しました。

クエリ time_dt を実行すると、S3 から自動的に日付パーティションが生成され、Athena の任意の時間ベースのフィールドと同様にクエリを実行できます。

これは、2023 年 3 月 1 日以降に time_dt パーティションを使用してログをクエリするクエリの例です。

```
SELECT *
  FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_vpc_flow_2_0"
 WHERE time_dt > TIMESTAMP '2023-03-01'
 AND src_endpoint.ip = '192.0.2.1'
 ORDER BY time desc
 LIMIT 25
```

time_dt の一般的な値は次の通りです。

過去 1 年間に発生したイベント

```
WHERE time_dt > CURRENT_TIMESTAMP - INTERVAL '1' YEAR
```

過去 1 か月に発生したイベント

```
WHERE time_dt > CURRENT_TIMESTAMP - INTERVAL '1' MONTH
```

過去 30 日間に発生したイベント

```
WHERE time_dt > CURRENT_TIMESTAMP - INTERVAL '30' DAY
```

過去 12 時間に発生したイベント

```
WHERE time_dt > CURRENT_TIMESTAMP - INTERVAL '12' HOUR
```

過去 5 分間に発生したイベント

```
WHERE time_dt > CURRENT_TIMESTAMP - INTERVAL '5' MINUTE
```

7 ~ 14 日前に発生したイベント

```
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '14' DAY AND  
CURRENT_TIMESTAMP - INTERVAL '7' DAY
```

特定の日付以降に発生するイベント

```
WHERE time_dt >= TIMESTAMP '2023-03-01'
```

例:2023 年 3 月 1 日以降のソース IP**192.0.2.1** からのすべての CloudTrail アクティビティのリスト (表) **cloud_trail_mgmt_1_0**

```
SELECT *  
FROM  
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_cloud_trail_mgmt_1_0  
WHERE eventDay >= '20230301'  
AND src_endpoint.ip = '192.0.2.1'  
ORDER BY time desc  
LIMIT 25
```

例: テーブル **cloud_trail_mgmt_1_0** 内の過去 30 日間のソース IP **192.0.2.1** からのすべての CloudTrail アクティビティのリスト

```
SELECT *  
FROM  
amazon_security_lake_glue_db_us_east_1.amazon_security_lake_table_us_east_1_cloud_trail_mgmt_1_0  
WHERE eventDay > cast(date_format(current_timestamp - INTERVAL '30' day, '%Y%m%d  
%H') as varchar)  
AND src_endpoint.ip = '192.0.2.1'  
ORDER BY time desc  
LIMIT 25
```

Security Lake オブザーバビリティのクエリ

オブザーバブルは、Security Lake 2.0 で利用可能になった新機能です。オブザーバブルオブジェクトは、イベントの多くの場所で見つかった関連情報を含むピボット要素です。オブザーバビリティをクエリすると、ユーザーはデータセット全体で から高レベルのセキュリティインサイトを取得できます。

オブザーバブル内の特定の要素をクエリすることで、特定のユーザー名、リソース UUIDs、IPs、ハッシュ、その他の IOC タイプ情報などのものにデータセットを制限できます。

これは、observables 配列を使用して VPC Flow と Route53 テーブル全体のログをクエリするクエリの例で、IP 値172.01.02.03「」が含まれています。

```
WITH a AS
  (SELECT
    time_dt,
    observable.name,
    observable.value
  FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_vpc_flow_2_0",
    UNNEST(observables) AS t(observable)
  WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
  AND observable.value='172.01.02.03'
  AND observable.name='src_endpoint.ip'),
b as
  (SELECT
    time_dt,
    observable.name,
    observable.value
  FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_route53_2_0",
    UNNEST(observables) AS t(observable)
  WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
  AND observable.value='172.01.02.03'
  AND observable.name='src_endpoint.ip')
SELECT * FROM a
LEFT JOIN b ON a.value=b.value and a.name=b.name
LIMIT 25
```

CloudTrail データの Security Lake クエリの例

AWS CloudTrail は、でのユーザーアクティビティと API の使用を追跡します AWS のサービス。サブスクライバーは CloudTrail データにクエリを実行して、次の種類の情報を取得できます。

AWS ソースバージョン 2 の CloudTrail データに対するクエリの例を次に示します。

過去 7 日間の AWS のサービス に対する不正な試行

```
SELECT
  time_dt,
  api.service.name,
  api.operation,
```

```
    api.response.error,  
    api.response.message,  
    api.response.data,  
    cloud.region,  
    actor.user.uid,  
    src_endpoint.ip,  
    http_request.user_agent  
FROM  
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_cloud_trail_mgm  
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP  
AND api.response.error in (  
    'Client.UnauthorizedOperation',  
    'Client.InvalidPermission.NotFound',  
    'Client.OperationNotPermitted',  
    'AccessDenied')  
ORDER BY time desc  
LIMIT 25
```

過去 7 日間のソース IP **192.0.2.1** からのすべての CloudTrail アクティビティのリスト

```
SELECT  
    api.request.uid,  
    time_dt,  
    api.service.name,  
    api.operation,  
    cloud.region,  
    actor.user.uid,  
    src_endpoint.ip,  
    http_request.user_agent  
FROM  
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_cloud_trail_mgm  
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP  
AND src_endpoint.ip = '192.0.2.1.'  
ORDER BY time desc  
LIMIT 25
```

過去 7 日間のすべての IAM アクティビティのリスト

```
SELECT *  
FROM  
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_cloud_trail_mgm  
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP  
AND api.service.name = 'iam.amazonaws.com'
```

```
ORDER BY time desc
LIMIT 25
```

過去 7 AIDACKCEVSQ6C2EXAMPLE 日間に認証情報が使用されたインスタンス

```
SELECT
    actor.user.uid,
    actor.user.uid_alt,
    actor.user.account.uid,
    cloud.region
FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_cloud_trail_mgm
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND actor.user.credential_uid = 'AIDACKCEVSQ6C2EXAMPLE'
LIMIT 25
```

過去 7 日間に失敗した CloudTrail レコードのリスト

```
SELECT
    actor.user.uid,
    actor.user.uid_alt,
    actor.user.account.uid,
    cloud.region
FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_cloud_trail_mgm
WHERE status='failed' and time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND
CURRENT_TIMESTAMP
ORDER BY time DESC
LIMIT 25
```

Route 53 Resolver クエリログのクエリ例

Amazon Route 53 リゾルバーのクエリログは、Amazon VPC 内のリソースによって行われた DNS クエリを追跡します。利用者は Route 53 リゾルバーのクエリログをクエリして、次の種類の情報を確認できます。

AWS ソースバージョン 2 の Route 53 リゾルバークエリログのクエリの例を次に示します。

過去 7 日間の CloudTrail からの DNS クエリのリスト

```
SELECT
```

```
time_dt,  
src_endpoint.instance_uid,  
src_endpoint.ip,  
src_endpoint.port,  
query.hostname,  
rcode  
FROM  
"amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_route53_2_0"  
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP  
ORDER BY time DESC  
LIMIT 25
```

過去 7 日間に **s3.amazonaws.com** と一致した DNS クエリのリスト

```
SELECT  
time_dt,  
src_endpoint.instance_uid,  
src_endpoint.ip,  
src_endpoint.port,  
query.hostname,  
rcode,  
answers  
FROM  
"amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_route53_2_0"  
WHERE query.hostname LIKE 's3.amazonaws.com.' and time_dt BETWEEN CURRENT_TIMESTAMP -  
INTERVAL '7' DAY AND CURRENT_TIMESTAMP  
ORDER BY time DESC  
LIMIT 25
```

過去 7 日間に解決されなかった DNS クエリのリスト

```
SELECT  
time_dt,  
src_endpoint.instance_uid,  
src_endpoint.ip,  
src_endpoint.port,  
query.hostname,  
rcode,  
answers  
FROM  
"amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_route53_2_0"  
WHERE cardinality(answers) = 0 and time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY  
AND CURRENT_TIMESTAMP
```

```
LIMIT 25
```

過去 7 日間に**192.0.2.1**に解決された DNS クエリのリスト

```
SELECT
    time_dt,
    src_endpoint.instance_uid,
    src_endpoint.ip,
    src_endpoint.port,
    query.hostname,
    rcode,
    answer.rdata
FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_route53_2_0",
    UNNEST(answers) as st(answer)
WHERE answer.rdata='192.0.2.1'
AND time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
LIMIT 25
```

Security Hub の検出結果に対する Security Lake クエリの例

Security Hub は、 のセキュリティ状態を包括的に把握 AWS し、セキュリティ業界標準とベストプラクティスに照らして環境をチェックするのに役立ちます。Security Hub はセキュリティチェック用の結果を生成し、サードパーティのサービスから結果を受け取ります。

AWS ソースバージョン 2 の Security Hub の検出結果に対するクエリの例を次に示します。

過去 7 日間の**MEDIUM**と同等かそれ以上の新しい検出結果

```
SELECT
    time_dt,
    finding_info,
    severity_id,
    status
FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_sh_findings_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
    AND severity_id >= 3
    AND status = 'New'
ORDER BY time DESC
LIMIT 25
```

過去 7 日間の重複する検出結果

```
SELECT
  finding_info.uid,
  MAX(time_dt) AS time,
  ARBITRARY(region) AS region,
  ARBITRARY(accountid) AS accountid,
  ARBITRARY(finding_info) AS finding,
  ARBITRARY(vulnerabilities) AS vulnerabilities
FROM
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_sh_findings_2_0
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
GROUP BY finding_info.uid
LIMIT 25
```

過去 7 日間の情報提供以外のすべての調査結果

```
SELECT
  time_dt,
  finding_info.title,
  finding_info,
  severity
FROM
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_sh_findings_2_0
WHERE severity != 'Informational' and time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7'
  DAY AND CURRENT_TIMESTAMP
LIMIT 25
```

リソースが Amazon S3 バケットである場合の結果 (時間制限なし)

```
SELECT *
FROM
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_sh_findings_2_0
WHERE any_match(resources, element -> element.type = 'amzn-s3-demo-bucket')
LIMIT 25
```

Common Vulnerability Scoring System (CVSS) スコアが 1 (時間制限なし) 以上の検出結果

```
SELECT
  DISTINCT finding_info.uid
  time_dt,
```

```
    metadata,  
    finding_info,  
    vulnerabilities,  
    resource  
FROM  
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_sh_findings_2_0  
UNNEST(vulnerabilities) AS t(vulnerability),  
UNNEST(vulnerability.cve.cvss) AS t(cvss)  
WHERE cvss.base_score > 1.0  
AND vulnerabilities is NOT NULL  
LIMIT 25
```

一般的な脆弱性と露出 (CVE) **CVE-0000-0000** に該当する検出結果 (時間制限なし)

```
SELECT *  
  FROM  
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_sh_findings_2_0  
WHERE any_match(vulnerabilities, element -> element.cve.uid = 'CVE-0000-0000')  
LIMIT 25
```

過去 7 日間に Security Hub から結果を送信した製品の数

```
SELECT  
  metadata.product.name,  
  count(*)  
FROM  
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_sh_findings_2_0  
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP  
GROUP BY metadata.product.name  
ORDER BY metadata.product.name DESC  
LIMIT 25
```

過去 7 日間の結果に含まれるリソースタイプの数

```
SELECT  
  count(*) AS "Total",  
  resource.type  
FROM  
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_sh_findings_2_0  
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP  
GROUP BY resource.type  
ORDER BY count(*) DESC
```

```
LIMIT 25
```

過去 7 日間の検出結果から得られた脆弱なパッケージ

```
SELECT
    vulnerabilities
FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_sh_findings_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND vulnerabilities is NOT NULL
LIMIT 25
```

過去 7 日間に変更された調査結果

```
SELECT
    status,
    finding_info.title,
    finding_info.created_time_dt,
    finding_info,
    finding_info.uid,
    finding_info.first_seen_time_dt,
    finding_info.last_seen_time_dt,
    finding_info.modified_time_dt
FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_sh_findings_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
LIMIT 25
```

Amazon VPC フローログの Security Lake クエリの例

Amazon Virtual Private Cloud (Amazon VPC) は、VPC 内のネットワーク インターフェイスとの間で送受信される IP トラフィックに関する詳細を提供します。

AWS ソースバージョン 2 の Amazon VPC フローログのクエリの例を次に示します。

過去 7 日間の特定の AWS リージョン のトラフィック

```
SELECT *
FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_vpc_flow_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
```

```
AND region in ('us-east-1','us-east-2','us-west-2')
LIMIT 25
```

過去 7 日間の送信元 IP **192.0.2.1** と送信元ポート**22**のアクティビティのリスト。

```
SELECT *
FROM
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_vpc_flow_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND src_endpoint.ip = '192.0.2.1'
AND src_endpoint.port = 22
LIMIT 25
```

過去 7 日間の個別の宛先 IP アドレスの数

```
SELECT
  COUNT(DISTINCT dst_endpoint.ip) AS "Total"
FROM
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_vpc_flow_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
LIMIT 25
```

過去 7 日間に 198.51.100.0/24 から発生したトラフィック

```
SELECT *
FROM
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_vpc_flow_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND split_part(src_endpoint.ip, '.', 1)='198'AND split_part(src_endpoint.ip, '.', 2)='51'
LIMIT 25
```

過去 7 日間のすべての HTTPS トラフィック

```
SELECT
  dst_endpoint.ip as dst,
  src_endpoint.ip as src,
  traffic.packets
FROM
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_vpc_flow_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND dst_endpoint.port = 443
```

```
GROUP BY
  dst_endpoint.ip,
  traffic.packets,
  src_endpoint.ip
ORDER BY traffic.packets DESC
LIMIT 25
```

過去 7 日間のポート**443**宛ての接続のパケット数順

```
SELECT
  traffic.packets,
  dst_endpoint.ip
FROM
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_vpc_flow_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND dst_endpoint.port = 443
GROUP BY
  traffic.packets,
  dst_endpoint.ip
ORDER BY traffic.packets DESC
LIMIT 25
```

過去 7 日間の IP **192.0.2.1** と **192.0.2.2** 間のすべてのトラフィック

```
SELECT
  start_time_dt,
  end_time_dt,
  src_endpoint.interface_uid,
  connection_info.direction,
  src_endpoint.ip,
  dst_endpoint.ip,
  src_endpoint.port,
  dst_endpoint.port,
  traffic.packets,
  traffic.bytes
FROM
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_vpc_flow_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND(
  src_endpoint.ip = '192.0.2.1'
AND dst_endpoint.ip = '192.0.2.2')
OR (
  src_endpoint.ip = '192.0.2.2'
```

```
AND dst_endpoint.ip = '192.0.2.1')
ORDER BY start_time_dt ASC
LIMIT 25
```

過去 7 日間のすべてのインバウンドトラフィック

```
SELECT *
FROM
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_vpc_flow_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND connection_info.direction = 'Inbound'
LIMIT 25
```

過去 7 日間のすべてのアウトバウンドトラフィック

```
SELECT *
FROM
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_vpc_flow_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND connection_info.direction = 'Outbound'
LIMIT 25
```

過去 7 日間に拒否されたすべてのトラフィック

```
SELECT *
FROM
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_vpc_flow_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND action = 'Denied'
LIMIT 25
```

Amazon EKS 監査ログの Security Lake クエリの例

Amazon EKS ログは、コントロールプレーンのアクティビティを追跡し、Amazon EKS コントロールプレーンからアカウントの CloudWatch Logs に監査ログと診断ログを直接提供します。これらのログを使用すると、クラスターの保護と実行が容易になります。サブスクライバーは EKS ログをクエリして、次のタイプの情報を学習できます。

AWS ソースバージョン 2 の Amazon EKS 監査ログのクエリの例を次に示します。

過去 7 日間の特定の URL へのリクエスト

```
SELECT
    time_dt,
    actor.user.name,
    http_request.url.path,
    activity_name
FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_eks_audit_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND activity_name = 'get'
and http_request.url.path = '/apis/coordination.k8s.io/v1/'
LIMIT 25
```

過去7日間の「10.0.97.167」からのリクエストの更新

```
SELECT
    activity_name,
    time_dt,
    api.request,
    http_request.url.path,
    src_endpoint.ip,
    resources
FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_eks_audit_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND src_endpoint.ip = '10.0.97.167'
AND activity_name = 'Update'
LIMIT 25
```

過去7日間のリソース「kube-controller-manager」に関連するリクエストとレスポンス

```
SELECT
    activity_name,
    time_dt,
    api.request,
    api.response,
    resource.name
FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_eks_audit_2_0",
    UNNEST(resources) AS t(resource)
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND resource.name = 'kube-controller-manager'
LIMIT 25
```

v2 ログの Security Lake AWS WAFクエリの例

AWS WAF は、エンドユーザーがアプリケーションに送信するウェブリクエストをモニタリングし、コンテンツへのアクセスを制御するために使用できるウェブアプリケーションファイアウォールです。

AWS ソースバージョン 2 AWS WAFの v2 ログのクエリの例を以下に示します。

過去 7 日間の特定のソース IP からのリクエストを投稿する

```
SELECT
    time_dt,
    activity_name,
    src_endpoint.ip,
    http_request.url.path,
    http_request.url.hostname,
    http_request.http_method,
    http_request.http_headers
FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_waf_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND src_endpoint.ip = '100.123.123.123'
AND activity_name = 'Post'
LIMIT 25
```

過去 7 日間にファイアウォールタイプ MANAGED_RULE_GROUP に一致したリクエスト

```
SELECT
    time_dt,
    activity_name,
    src_endpoint.ip,
    http_request.url.path,
    http_request.url.hostname,
    http_request.http_method,
    firewall_rule.uid,
    firewall_rule.type,
    firewall_rule.condition,
    firewall_rule.match_location,
    firewall_rule.match_details,
    firewall_rule.rate_limit
FROM
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_waf_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
```

```
AND firewall_rule.type = 'MANAGED_RULE_GROUP'  
LIMIT 25
```

過去 7 日間にファイアウォールルールで REGEX に一致したリクエスト

```
SELECT  
    time_dt,  
    activity_name,  
    src_endpoint.ip,  
    http_request.url.path,  
    http_request.url.hostname,  
    http_request.http_method,  
    firewall_rule.uid,  
    firewall_rule.type,  
    firewall_rule.condition,  
    firewall_rule.match_location,  
    firewall_rule.match_details,  
    firewall_rule.rate_limit  
FROM  
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_waf_2_0"  
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP  
AND firewall_rule.condition = 'REGEX'  
LIMIT 25
```

過去 7 日間に AWS WAF ルールをトリガーした AWS 認証情報のリクエストの取得を拒否しました

```
SELECT  
    time_dt,  
    activity_name,  
    action,  
    src_endpoint.ip,  
    http_request.url.path,  
    http_request.url.hostname,  
    http_request.http_method,  
    firewall_rule.uid,  
    firewall_rule.type  
FROM  
    "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_waf_2_0"  
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP  
AND http_request.url.path = '/.aws/credentials'  
AND action = 'Denied'  
LIMIT 25
```

過去 7 AWS 日間に国ごとにグループ化された認証情報のリクエストを取得する

```
SELECT count(*) as Total,
       src_endpoint.location.country AS Country,
       activity_name,
       action,
       src_endpoint.ip,
       http_request.url.path,
       http_request.url.hostname,
       http_request.http_method
FROM
  "amazon_security_lake_glue_db_us_east_1"."amazon_security_lake_table_us_east_1_waf_2_0"
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY
      AND CURRENT_TIMESTAMP
      AND activity_name = 'Get'
      AND http_request.url.path = '/.aws/credentials'
GROUP BY src_endpoint.location.country,
         activity_name,
         action,
         src_endpoint.ip,
         http_request.url.path,
         http_request.url.hostname,
         http_request.http_method
```

Security Lakeのライフサイクル管理

Security Lake をカスタマイズして、任意の期間 AWS リージョン、任意の にデータを保存できます。ライフサイクル管理は、さまざまなコンプライアンス要件への準拠に役立ちます。

保持管理

データをコスト効率良く保存するように管理するには、Security Lake のライフサイクル設定を使用してデータの保持を設定できます。これらの保持設定は、希望する [Amazon S3 ストレージクラス](#)と、Amazon S3 オブジェクトが別のストレージクラスに移行して有効期限が切れる前にそのストレージクラスに留まる期間を指定するのに役立ちます。

Warning

Security Lake コンソール、API、または CLI を使用して保持設定を管理することをお勧めします。これは、Amazon S3 サービスで Amazon S3 ライフサイクル設定を直接変更すると、メタデータが削除され、データにアクセスできなくなる可能性があるためです。

Security Lake での保持設定に関する重要な考慮事項

Security Lake でデータ保持を管理するときは、次の考慮事項を確認してください。

- Security Lake は [Amazon S3 オブジェクトロック](#)をサポートしていません。データレイクバケットが作成されると、S3 Object Lock はデフォルトで無効になります。デフォルトの保持モードで S3 オブジェクトロックを有効にすると、正規化されたログデータのデータレイクへの配信が中断されます。
- デフォルトの Amazon S3 ストレージ クラスは S3 Standardです。保持設定を設定しない場合、Security Lake は Amazon S3 ライフサイクル設定のデフォルト設定を使用します。SS3 Standard ストレージクラスを使用してデータを無期限に保存します。
- Security Lake では、リージョンレベルで保持設定を指定します。たとえば、特定の 内のすべての S3 オブジェクトを、データレイクに書き込まれた 30 日後に S3 Standard-IA ストレージクラスに移行する AWS リージョン ように設定できます。
- 保持設定は S3 バケットに保存されているデータにのみ適用されますが、Apache Iceberg メタデータは保持ポリシーから除外されます。

Security Lake を有効にする際の保存設定を行います。

Security Lake にオンボーディングするときに、以下の手順に従って 1 つ以上のリージョンの保存設定を行います。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. 「ステップ 2: オンボーディングワークフローの目標を定義」に到達したら、「ストレージクラスの選択」で「移行を追加」を選択します。次に、S3 オブジェクトを移行する Amazon S3 ストレージクラスを選択します。(リストされていないデフォルトのストレージ クラスは S3 Standard です。) また、そのストレージクラスの保持期間 (日数) を指定します。それ以降にオブジェクトを別のストレージクラスに移行するには、[Add transition] を選択し、次のストレージクラスと保存期間の設定を入力します。
3. S3 オブジェクトの有効期限を指定するには、[Add transition] を選択します。次に、[ストレージクラス] に [期限切れ] を選択します。保持期間には、オブジェクトが作成されてから、任意のストレージクラスを使用して Amazon S3 にオブジェクトを保存する合計日数を入力します。この期間が終了すると、オブジェクトは期限切れになり、Amazon S3 はそれらを削除します。
4. 終了したら、次へ を選択します。

変更は、以前のオンボーディングステップで Security Lake を有効にしたすべてのリージョンに適用されます。

API

Security Lake へのオンボーディング時に保持設定をプログラムで設定するには、Security Lake API の [CreateDataLake](#) オペレーションを使用します。AWS CLI を使用している場合は、[create-data-lake](#) コマンドを実行します。lifecycleConfiguration パラメータに必要な保持設定を次のように指定します。

- transitions には、特定の Amazon S3 ストレージクラス (storageClass) に S3 オブジェクトを保存する合計日数 (days) を指定します。
- expiration には、オブジェクトが作成されてから、任意のストレージクラスを使用して Amazon S3 にオブジェクトを保存する合計日数を指定します。この期間が終了すると、オブジェクトは期限切れになり、Amazon S3 はそれらを削除します。

Security Lake は、configurations オブジェクトの region フィールドで指定したリージョンに設定を適用します。

たとえば、次のコマンドは、us-east-1 リージョンで Security Lake を有効にします。このリージョンでは、オブジェクトは 365 日後に期限切れになり、オブジェクトは 60 日後に ONEZONE_IA S3 ストレージクラスに移行します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake create-data-lake \  
--configurations '[{"encryptionConfiguration":  
{"kmsKeyId":"S3_MANAGED_KEY"},"region":"us-east-1","lifecycleConfiguration":  
{"expiration":{"days":365},"transitions":  
[{"days":60,"storageClass":"ONEZONE_IA"}]}]' \  
--meta-store-manager-role-arn "arn:aws:securitylake:ap-  
northeast-2:123456789012:data-lake/default"
```

保持設定の更新

Security Lake を有効にしたら、以下の手順に従って 1 つ以上のリージョンの保存設定を更新してください。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ナビゲーションペインで、[リージョン] を選択します。
3. リージョンを選択し、[編集] を選択します。
4. [ストレージクラスの選択] セクションで、必要な設定を入力します。ストレージクラスで、S3 オブジェクトを転送する Amazon S3 ストレージクラスを選択します。(リストされていないデフォルトのストレージクラスは S3 Standard です。) 保持期間には、そのストレージクラスに格納する日数を入力します。複数のトランジションを指定できます。

S3 オブジェクトをいつ期限切れにするかも指定するには、ストレージクラスの [期限切れ] を選択します。次に、保持期間には、オブジェクトが作成されてから、任意のストレージクラスを使用して Amazon S3 にオブジェクトを保存する合計日数を入力します。この期間が終了すると、オブジェクトは期限切れになり、Amazon S3 はそれらを削除します。

5. 完了したら、保存 を選択します。

API

プログラムで保持設定を更新するには、Security Lake API の [UpdateDataLake](#) オペレーションを使用します。を使用している場合は AWS CLI、[update-data-lake](#) コマンドを実行します。リクエストで、`lifecycleConfiguration` パラメータを使用して新しい設定を指定します。

- 移行設定を変更するには、`transitions` パラメータを使用して、特定の Amazon S3 ストレージクラス (`storageClass`) に S3 オブジェクトを保存する新しい期間を日数 (`days`) 単位で指定します。
- 全体の保持期間を変更するには、`expiration` パラメータを使用して、オブジェクトが作成されてから、任意のストレージクラスを使用して S3 オブジェクトを保存する合計日数を指定します。この保持期間が終了すると、オブジェクトは期限切れになり、Amazon S3 はそれらを削除します。

Security Lake は、`configurations` オブジェクトの `region` フィールドで指定したリージョンに設定を適用します。

Security Lake API の `UpdateDataLake` オペレーションは、指定された項目またはレコードが存在しない場合は挿入を実行する「アップサート」オペレーションとして機能し、既に存在する場合は更新を実行します。Security Lake は、AWS 暗号化ソリューションを使用して保管中のデータを安全に保存します。

現在 KMS を使用している更新呼び出しに含まれているリージョン `encryptionConfiguration` からキーを省略すると、そのリージョンの KMS キーはそのまま残りますが、キーを指定すると、同じリージョンでキーがリセットされます。

たとえば、次の AWS CLI コマンドは、`us-east-1` リージョンのデータの有効期限設定とストレージ移行設定を更新します。このリージョンでは、オブジェクトは 500 日後に期限切れになり、オブジェクトは 30 日後に `ONEZONE_IA` S3 ストレージクラスに移行します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックslash (\) の行継続文字を使用しています。

```
$ aws securitylake update-data-lake \
--configurations '[{"encryptionConfiguration":
{"kmsKeyId": "S3_MANAGED_KEY", "region": "us-east-1", "lifecycleConfiguration":
{"expiration": {"days": 500}, "transitions":
[{"days": 30, "storageClass": "ONEZONE_IA"}]}]' \
--meta-store-manager-role-arn "arn:aws:securitylake:ap-
northeast-2:123456789012:data-lake/default"
```

ロールアップリージョン

ロールアップリージョンは、1つ以上の寄与リージョンのデータを統合します。これは、地域のデータコンプライアンス要件に準拠するのに役立ちます。

ロールアップリージョンを設定する手順については、「」を参照してください[Security Lake でのロールアップリージョンの設定](#)。

Security Lake でサイバーセキュリティスキーマフレームワーク (OCSF) を開く

OCSFとは何ですか

[Open Cybersecurity Schema Framework \(OCSF\)](#) は、サイバーセキュリティ業界における AWS および主要なパートナーによる共同のオープンソースの取り組みです。OCSF は、一般的なセキュリティイベントの標準スキーマを提供し、スキーマの進化を容易にするバージョニング基準を定義し、セキュリティログの作成者と利用者を対象とした自己管理プロセスを組み込んでいます。OCSF の公開ソースコードは [GitHub](#) でホストされています。

Security Lake は、ネイティブにサポートされているログとイベントを OCSF スキーマに自動的に変換 AWS のサービスします。OCSF に変換すると、Security Lake はデータを の Amazon Simple Storage Service (Amazon S3) バケット (ごとに 1 つのバケット AWS リージョン) に保存します AWS アカウント。カスタムソースからセキュリティレイクに書き込まれるログとイベントは、OCSF スキーマと Apache Parquet 形式に準拠している必要があります。サブスクライバーは、ログとイベントを汎用の Parquet レコードとして扱うことも、OCSF スキーマのイベントクラスを適用してレコードに含まれる情報をより正確に解釈することもできます。

OCSF イベントクラス

特定の Security Lake [ソース](#)からのログとイベントは、OCSF で定義された特定のイベントクラスと一致します。[OCSFのイベントクラス](#)には、DNS アクティビティ、SSH アクティビティ、認証などがあります。特定のソースがどのイベントクラスと一致するかを指定できます。

OCSFソースの識別

OCSF は、さまざまなフィールドを使用して、特定のログやイベントの発生元を特定するのに役立ちます。これらは、Security Lake のソースとしてネイティブにサポートされている AWS のサービスの関連フィールドの値です。

The OCSF source identification for AWS log sources (Version 1) are listed in the following table.

ソース	metadata. product.name	metadata. product.v endor_name	metadata. product.f eature.name	class_name	metadata. version
CloudTrail Lambda デー タイイベント	CloudTrai l	AWS	Data	API Activity	1.0.0-rc. 2
CloudTrail 管 理イベント	CloudTrai l	AWS	Managemen t	API Activity, Authentic ation , また は Account Change	1.0.0-rc. 2
CloudTrail S3 データイベン ト	CloudTrai l	AWS	Data	API Activity	1.0.0-rc. 2
Route 53	Route 53	AWS	Resolver Query Logs	DNS Activity	1.0.0-rc. 2
Security Hub	Security Hub	AWS	Security Hub ProductName 値と一致 します	Security Finding	1.0.0-rc. 2
VPC フロー ログ	Amazon VPC	AWS	Flowlogs	Network Activity	1.0.0-rc. 2

The OCSF source identification for AWS log sources (Version 2) are listed in the following table.

ソース	metadata. product.name	metadata. product.v endor_name	metadata. product.f eature.name	class_name	metadata. version
CloudTrail Lambda デー タイイベント	CloudTrai l	AWS	Data	API Activity	1.1.0
CloudTrail 管 理イベント	CloudTrai l	AWS	Managemen t	API Activity, Authentic ation , また は Account Change	1.1.0
CloudTrail S3 データイベン ト	CloudTrai l	AWS	Data	API Activity	1.1.0
Route 53	Route 53	AWS	Resolver Query Logs	DNS Activity	1.1.0
Security Hub	AWS Security Finding 形 式 (ASFF) ProductNa me の値と一 致する	AWS Security Finding 形 式 (ASFF) CompanyNa me の値と一 致する	ASFF featureNa me の値 と一致 ProductFi elds	Vulnerabi lity Finding, Complianc e Finding, or Detection Finding	1.1.0
VPC フロー ログ	Amazon VPC	AWS	Flowlogs	Network Activity	1.1.0

ソース	metadata. product.name	metadata. product.v endor_name	metadata. product.f eature.name	class_name	metadata. version
EKS 監査ロ グ	Amazon EKS	AWS	Elastic Kubernet e Service	API Activity	1.1.0
AWS WAF v2 ログ	AWS WAF	AWS	–	HTTP Activity	1.1.0

Security Lake との統合

Amazon Security Lake は、他の AWS のサービス およびサードパーティー製品と統合されています。統合では、ソースとして Security Lake にデータを送信したり、サブスクライバーとして Security Lake のデータを利用したりできます。以下のトピックでは、Security Lake と統合する AWS のサービス およびサードパーティー製品について説明します。

トピック

- [AWS のサービス Security Lake との統合](#)
- [Security Lake とのサードパーティー統合](#)

AWS のサービス Security Lake との統合

Amazon Security Lake は他の と統合されています AWS のサービス。サービスは、ソース統合、サブスクライバー統合、またはその両方として動作する場合があります。

ソース統合には以下のプロパティがあります。

- Security Lakeにデータを送信します。
- データがスキーマに到着します。 [Security Lake でサイバーセキュリティスキーマフレームワーク \(OCSF\) を開く](#)
- データは Apache Parquet 形式で届きます

サブスクライバー統合は、次のいずれかの方法で Security Lake データにアクセスできます。

- HTTPS エンドポイントを介して Security Lake からソースデータを読み取る
- Amazon Simple Queue Service (Amazon SQS) を介して Security Lake からソースデータを読み取る
- を使用してソースデータを直接クエリする AWS Lake Formation

次の表に、Security Lake がサポートする AWS のサービス 統合のリストを示します。

AWS のサービス	統合タイプ	説明	統合の仕組み
Amazon Bedrock	サブスクライバー	AI を活用したインサイトを生成して Security Lake データを分析します。	Amazon Bedrock の統合
Amazon Detective	サブスクライバー	Security Lake にクエリを実行して、セキュリティ検出結果または疑わしいアクティビティの根本原因を分析し、調査し、迅速に特定します。	Amazon Detective の統合
Amazon OpenSearch Service	サブスクライバー	OpenSearch Service の取り込みを使用して、Security Lake データからセキュリティインサイトを生成します。	Amazon OpenSearch Service の統合
Amazon OpenSearch Service 取り込みパイプライン	サブスクライバー、ソース	ログ、メトリクス、トレースデータを OpenSearch Service と Security Lake にストリーミングします。	Amazon OpenSearch Service Ingestion パイプライン統合
Amazon OpenSearch Service ゼロ ETL	サブスクライバー (クエリ)	Security Lake のデータをゼロ ETL でクエリします。	Amazon OpenSearch Service ゼロ ETL ダイレクトクエリ統合
Amazon QuickSight	サブスクライバー	QuickSight を使用して Security Lake のク	Amazon QuickSight の統合

AWS のサービス	統合タイプ	説明	統合の仕組み
		データを視覚化、探索、解釈します。	
Amazon SageMaker AI	サブスクライバー	AI を活用したインサイトを生成して Security Lake データを分析します。	Amazon SageMaker AI 統合
AWS AppFabric	ソース	Software as a Service (SaaS) アプリケーションログを Security Lake 標準形式に取り込み、正規化します。	AWS AppFabric の統合
AWS Security Hub	ソース	Security Hub からのセキュリティ検出結果を Security Lake 標準形式で一元化して保存します。	AWS Security Hub 統合

Amazon Bedrock との統合

[Amazon Bedrock](#) は、主要な AI スタートアップと Amazon からの高性能な基盤モデル (FMs) を統合 API を通じて使用できるようにするフルマネージドサービスです。Amazon Bedrock のサーバーレスエクスペリエンスを使用すると、インフラストラクチャを管理することなく、すばやく開始し、独自のデータを使用して基盤モデルをプライベートにカスタマイズし、AWS ツールを使用して簡単かつ安全に統合してアプリケーションにデプロイできます。

生成 AI

SageMaker AI Studio の Amazon Bedrock の生成 AI 機能と自然言語入力を使用して、Security Lake 内のデータを分析し、組織のリスクを軽減し、セキュリティ体制を強化できます。適切なデータソースを自動的に特定し、SQL クエリを生成して呼び出し、調査からデータを視覚化することで、調査の実行に必要な時間を短縮できます。詳細については、[「Amazon SageMaker AI Studio と Amazon](#)

[Bedrock を使用して Amazon Security Lake の AI を活用したインサイトを生成する](#) を参照してください。

Amazon Detective を使用した調査

統合タイプ: サブスクライバー

[Amazon Detective](#) を使用すると、セキュリティに関する検出結果や疑わしいアクティビティの根本原因を分析、調査、および迅速に特定できます。Detective は、AWS リソースからログデータを自動的に収集します。その後、機械学習、統計分析、グラフ理論を使用して、セキュリティ調査をより迅速かつ効率的に行うのに役立つビジュアライゼーションを生成します。Detective の事前に作成されたデータの集計、要約、およびコンテキストは、考えられるセキュリティ問題の性質と範囲を迅速に分析および特定するのに役立ちます。

Security Lake と Detective を統合すると、Security Lake によって保存されている raw ログデータを Detective からクエリできます。詳細については、[「Amazon Security Lake との統合」](#) を参照してください。

Amazon OpenSearch Service との統合

統合タイプ: サブスクライバー

[Amazon OpenSearch Service](#) は、で OpenSearch Service クラスターのデプロイ、運用、スケーリングを容易にするマネージドサービスです AWS クラウド。OpenSearch Service Ingestion を使用して OpenSearch Service クラスターにデータを取り込むと、時間的制約のあるセキュリティ調査のためにインサイトをすばやく取得できます。セキュリティインシデントに迅速に対応できるため、ビジネスクリティカルなデータとシステムを保護するのに役立ちます。

OpenSearch Service ダッシュボード

OpenSearch Service を Security Lake と統合したら、サーバーレス OpenSearch Service Ingestion を介してさまざまなソースから OpenSearch Service にセキュリティデータを送信するように Security Lake を設定できます。セキュリティデータを処理するように OpenSearch Service 取り込みを設定する方法の詳細については、[「Amazon OpenSearch Service Ingestion を使用して Amazon Security Lake データからセキュリティインサイトを生成する」](#) を参照してください。

OpenSearch Service Ingestion が OpenSearch Service ドメインへのデータの書き込みを開始した後。構築済みのダッシュボードを使用してデータを視覚化するには、ダッシュボードに移動し、インストールされているダッシュボードのいずれかを選択します。

Amazon OpenSearch Service Ingestion パイプラインとの統合

統合タイプ：サブスクライバー、ソース

Amazon OpenSearch Service Ingestion は、ログ、メトリクス、トレースデータを OpenSearch Service と Security Lake にストリーミングするフルマネージド型のサーバーレスデータコレクターです。

OpenSearch Ingestion パイプラインを使用して Security Lake にデータを送信する

OpenSearch Ingestion の Amazon Simple Storage Service (Amazon S3) シンクプラグインを使用して、サポートされている任意のソースから Security Lake にデータを送信できます。Security Lake は、AWS 環境、オンプレミス環境、SaaS プロバイダーのセキュリティデータを専用のデータレイクに自動的に一元化します。詳細については、[「Amazon Security Lake をシンクとして OpenSearch Ingestion パイプラインを使用する」](#)を参照してください。

OpenSearch Ingestion パイプラインを使用して Security Lake から OpenSearch にデータを送信する

Amazon S3 ソースプラグインを使用して、OpenSearch Ingestion パイプラインにデータを取り込むことができます。詳細については、[「Amazon Security Lake をソースとして OpenSearch Ingestion パイプラインを使用する」](#)を参照してください。

Amazon OpenSearch Service ゼロ ETL 直接クエリとの統合

統合タイプ：サブスクライバー (クエリ)

OpenSearch Service ダイレクトクエリを使用して、Amazon Security Lake 内のデータを分析できます。OpenSearch Service は、取り込みパイプラインの構築や分析ツール間の切り替えの摩擦を発生させることなく、OpenSearch SQL または OpenSearch Piped Processing Language (PPL) を使用して Security Lake 内のデータを直接クエリする方法としてゼロ ETL 統合を提供します。このアプローチにより、データの移動や重複が不要になり、OpenSearch Service Dashboards の Discover エクスペリエンスを使用して、保管されているデータを分析できます。保管中のデータのクエリからダッシュボードによるアクティブモニタリングに切り替える場合は、クエリ結果にインデックス付きビューを構築し、OpenSearch Service インデックスに取り込むことができます。直接クエリの詳細については、「Amazon OpenSearch Service デベロッパーガイド」の[「直接クエリの使用」](#)を参照してください。

OpenSearch Service は OpenSearch Serverless コレクションを使用して、データを Security Lake に直接クエリし、インデックス付きビューを保存します。これを行うには、Security Lake データ

で OpenSearch ゼロ ETL 機能を使用できるようにするデータソースを作成します。データソースを作成すると、Security Lake に保存されているデータを直接検索、インサイトの取得、分析できます。オンデマンドインデックス作成を使用して、クエリのパフォーマンスを加速し、一部の Security Lake データセットで高度な OpenSearch 分析を使用できます。

- OpenSearch Service データソース統合の作成の詳細については、[「Amazon OpenSearch Service デベロッパーガイド」](#)の[「Amazon Security Lake データソース統合の作成」](#)を参照してください。 OpenSearch
- OpenSearch Service での Security Lake データソースの設定の詳細については、Amazon OpenSearch Service デベロッパーガイド OpenSearchの[OpenSearch Service Dashboards での Security Lake データソースの設定](#)を参照してください。

Security Lake で OpenSearch Service を使用する方法の詳細については、次のリソースを使用します。

- [セキュリティ分析を簡素化するための Amazon OpenSearch Service と Amazon Security Lake の統合の紹介](#)
- Amazon Security Lake を使用した OpenSearch Service でのゼロ ETL の概要

[Amazon Security Lake を使用した OpenSearch Service でのゼロ ETL の概要](#)

Amazon QuickSight との統合

統合タイプ: サブスクライバー

[Amazon QuickSight](#) は、クラウド規模のビジネスインテリジェンス (BI) サービスです。このサービスを使用すると、どこで仕事をしていても、相手に easy-to-understand インサイトを提供できます。QuickSight はクラウド内のデータに接続し、さまざまなソースからのデータを結合します。QuickSight は、意思決定者がインタラクティブなビジュアル環境で情報を探求し、解釈する機会を提供します。ネットワーク上の任意のデバイスおよびモバイルデバイスから、ダッシュボードに安全にアクセスできます。

Amazon QuickSight ダッシュボード

Amazon QuickSight で Amazon Security Lake データを視覚化するには、必要な AWS オブジェクトを作成し、Security Lake に関して基本的なデータソース、データセット、分析、ダッシュボー

ド、ユーザーグループを Amazon QuickSight にデプロイします。詳細な手順については、[Amazon QuickSight との統合](#)」を参照してください。

Amazon QuickSight による Security Lake データの視覚化の詳細については、以下のリソースを参照してください。

[Amazon QuickSight を使用した Security Lake データの視覚化: 2024 Amazon QuickSight 学習シリーズ](#)

[Security Lake による AWS WAF ウェブ ACL ログの運用](#)

Amazon SageMaker AI との統合

統合タイプ: サブスクライバー

[Amazon SageMaker AI](#) は、フルマネージド機械学習 (ML) サービスです。Security Lake を使用すると、データサイエンティストとデベロッパーは、本番環境に対応したホスト環境に ML モデルをすばやく自信を持って構築、トレーニング、デプロイできます。複数の統合開発環境 (IDEs) で SageMaker AI ML ツールを利用できるようにする ML ワークフローを実行するための UI エクスペリエンスを提供します。

SageMaker AI インサイト

SageMaker AI Studio を使用して、Security Lake の機械学習インサイトを生成できます。この Studio は、データサイエンティストが機械学習モデルを準備、構築、トレーニング、デプロイするためのツールを提供する、機械学習用のウェブ統合開発環境 (IDE) です。このソリューションを使用すると、Security Lake の検出 [AWS Security Hub](#) 結果に焦点を当てた Python ノートブックの基本セットをすばやくデプロイできます。また、Security Lake に他の AWS ソースやカスタムデータソースを組み込むように拡張することもできます。詳細については、「[Amazon SageMaker AI を使用して Amazon Security Lake データの機械学習インサイトを生成する](#)」を参照してください。

との統合 AWS AppFabric

統合タイプ: ソース

[AWS AppFabric](#) は、組織全体の Software as a Service (SaaS) アプリケーションを接続するノーコードサービスであるため、IT およびセキュリティアプリケーションは標準スキーマと中央リポジトリを使用します。

Security Lakeが AppFabric の調査結果を受け取る方法

AppFabric 監査ログデータをSecurity Lakeに送信するには、送信先として Amazon Kinesis Data Firehose を選択し、OCSF スキーマと Apache Parquet 形式のデータをSecurity Lakeに配信するように Kinesis Data Firehose を設定します。

前提条件

AppFabric 監査ログをSecurity Lakeに送信する前に、OCSF の正規化された監査ログを Kinesis Data Firehose ストリームに出力する必要があります。その後、出力をSecurity Lakeの Amazon S3 バケットに送信するように Kinesis Data Firehose を設定できます。詳細については、『Amazon Kinesis 開発者ガイド』の「[宛先として Amazon S3 を選択する](#)」を参照してください。

AppFabric の結果をSecurity Lakeに送信します。

前述の前提条件を満たした後に AppFabric 監査ログを Security Lake に送信するには、両方のサービスを有効にし、AppFabric をカスタムソースとして追加する必要があります。カスタムソースを追加する手順については、[を参照してください](#)。 [Security Lake のカスタムソースからデータを収集する](#)

Security Lakeでの AppFabric ログの受信を停止します。

AppFabric 監査ログの受信を停止するには、Security Lake コンソール、Security Lake API、または AWS CLI を使用して、カスタムソースとして AppFabric を削除できます。手順については、「[Security Lake からのカスタムソースの削除](#)」を参照してください。

との統合 AWS Security Hub

統合タイプ: ソース

[AWS Security Hub](#) は、のセキュリティ状態を包括的に把握 AWS し、セキュリティ業界標準とベストプラクティスに照らして環境を支援します。Security Hub は AWS アカウント、のサービス、およびサポートされているサードパーティーパートナー製品からセキュリティデータを収集し、セキュリティの傾向を分析し、最も優先度の高いセキュリティ問題を特定するのに役立ちます。

Security Hub を有効にし、Security Hub の結果を Security Lake のソースとして追加すると、Security Hub は新しい結果と既存の結果に対する更新をSecurity Lakeに送信し始めます。

Security Lake が Security Hub 調査結果を受け取る方法

Security Hub では、セキュリティの問題が調査結果として追跡されます。一部の検出結果は、他の AWS のサービス またはサードパーティーパートナーによって検出された問題から発生しま

す。Security Hub は、ルールに対して自動的かつ継続的なセキュリティチェックを実行することで、独自の検出結果も生成します。ルールはセキュリティコントロールによって表されます。

Security Hub のすべての調査結果で、[AWS Security Finding 形式 \(ASFF\)](#) と呼ばれる標準の JSON 形式が使用されます。

Security Lake は Security Hub 結果を受け取り、それを [Security Lake でサイバーセキュリティスキーマフレームワーク \(OCSF\) を開く](#) に変換します。

Security Hub の調査結果を Security Lake に送信する

Security Hub の検出結果を Security Lake に送信するには、両方のサービスを有効にし、Security Hub の検出結果を Security Lake のソースとして追加する必要があります。AWS ソースを追加する手順については、「」を参照してください [ソース AWS のサービスとしての追加](#)。

Security Hub で [制御結果](#) を生成し、Security Lake に送信する場合は、関連するセキュリティ標準を有効にし、AWS Config においてリージョンベースでリソースの記録をオンにする必要があります。詳細については、AWS Security Hub ユーザーガイドの「[AWS Config の有効化と設定](#)」を参照してください。

Security Lake で Security Hub の検出結果の受信を停止する

Security Hub の検出結果の受信を停止するには、「AWS Security Hub ユーザーガイド」の以下のトピック AWS CLI で Security Hub コンソール、Security Hub API、または を使用できます。

- [統合からの検出結果のフローの無効化と有効化 \(コンソール\)](#)
- [統合からの検出結果のフローの無効化 \(Security Hub API、AWS CLI\)](#)

Security Lake とのサードパーティー統合

Amazon Security Lake は、複数のサードパーティプロバイダーと統合できます。プロバイダーは、ソース統合、サブスクライバー統合、またはサービス統合を提供する場合があります。プロバイダーは 1 つ以上の統合タイプを提供する場合があります。

ソース統合には以下のプロパティがあります。

- Security Lake にデータを送信します。
- データは Apache Parquet 形式で届きます

- データがスキーマに到着します。[Security Lake でサイバーセキュリティスキーマフレームワーク \(OCSF\) を開く](#)

サブスクライバー統合には以下のプロパティがあります。

- HTTPS エンドポイントまたは Amazon Simple Queue Service (Amazon SQS) キューで Security Lake からソースデータを読み取るか、 から直接ソースデータをクエリします。AWS Lake Formation
- Apache Parquet 形式でデータを読み取ることができます
- OCSF スキーマのデータを読み取ることができる。

サービス統合は、Security Lake やその他の AWS のサービス を組織に実装するのに役立ちます。また、レポート、分析、その他のユースケースを支援できます。

特定のパートナープロバイダーを検索するには、Partner [Solutions Finder](#) を参照してください。サードパーティ製品を購入するには、[AWS Marketplace](#) を参照してください。

パートナー統合としての追加をリクエストするか、Security Lake パートナーになるには、<securitylake-partners@amazon.com> に E メールを送信します。

検出結果を に送信するサードパーティー統合を使用する場合は AWS Security Hub、Security Lake の Security Hub 統合が有効になっている場合、Security Lake でそれらの検出結果を確認することもできます。統合を有効にする手順については、「[との統合 AWS Security Hub](#)」を参照してください。検出結果を Security Hub に送信するサードパーティ統合のリストについては、『AWS Security Hub ユーザー ガイド』の「[利用可能なサードパーティ パートナー製品の統合](#)」を参照してください。

サブスクライバーを設定する前に、サブスクライバーの OCSF ログのサポートを確認します。詳細については、サブスクライバーのドキュメントを参照してください。

クエリ統合

Security Lake が AWS Lake Formation データベースとテーブルに保存するデータをクエリできます。Security Lake コンソール、API、または でサードパーティーのサブスクライバーを作成することもできます AWS Command Line Interface。

Lake Formation データレイク管理者は、データをクエリする IAM ID に、関連するデータベースとテーブルに対する SELECT 権限を付与する必要があります。データをクエリする前に、Security Lake

でサブスクライバーを作成する必要があります。クエリ アクセスを持つサブスクライバを作成する方法の詳細については、「[Security Lake サブスクライバーのクエリアクセスの管理](#)」を参照してください。

以下のサードパーティーパートナーに対して、Security Lake とのクエリ統合を設定できます。

- Cribl – Search
- IBM – QRadar
- Palo Alto Networks – XSOAR
- Query.AI – Query Federated Search
- SOC Prime
- [Splunk](#) – Federated Analytics
- Tego Cyber

Accenture – MxDR

統合タイプ: サブスクライバー、サービス

Accenture's MxDR と Security Lake の統合により、ログやイベントのリアルタイムのデータ取り込み、マネージド型の異常検知、脅威ハンティング、セキュリティ運用が可能になります。これにより、分析やマネージド・ディテクション・アンド・レスポンス (MDR) が容易になります。

サービス統合として、Accentureは組織への Security Lake の実装にも役立ちます。

[統合ドキュメンテーション](#)

Aqua Security

統合タイプ: ソース

Aqua Securityをカスタム ソースとして追加して、監査イベントを Security Lake に送信できます。監査イベントは OCSF スキーマと Parquet 形式に変換されます。

[統合ドキュメンテーション](#)

Barracuda – Email Protection

統合タイプ: ソース

Barracuda Email Protectionは、新しいフィッシングメール攻撃が検出されたときに Security Lake に イベントを送信できます。これらのイベントは、データレイク内の他のSecurity データと一緒に受信 できます。

[統合ドキュメンテーション](#)

Booz Allen Hamilton

統合タイプ: サービス

サービス統合として、Booz Allen Hamilton はデータと分析を Security Lake サービスと融合すること により、サイバーセキュリティに対してデータ駆動型のアプローチを採用しています。

[パートナーリンク](#)

Bosch Software and Digital Solutions – AIShield

統合タイプ: ソース

AIShield を搭載した Boschは、Security Lake との統合を通じて、AI アセットの自動脆弱性分析とエ ンドポイント保護を提供します。

[統合ドキュメンテーション](#)

ChaosSearch

統合タイプ: サブスクライバー

ChaosSearchは、Elasticsearch や SQL などのオープン API、またはネイティブに含まれる Kibana UI や Superset UI を使用して、ユーザーにマルチモデル データ アクセスを提供します。Security Lake データを ChaosSearch で保持制限なしで使用して、監視、アラート、脅威の探索を行うこと ができます。これにより、今日の複雑なセキュリティ環境や持続的な脅威に立ち向かうことができま す。

[統合ドキュメンテーション](#)

Cisco Security – Secure Firewall

統合タイプ: ソース

Cisco Secure Firewall を Security Lake と統合することにより、構造化されたスケーラブルな方法でファイアウォールログを保存できます。シスコの eNcore クライアントは、ファイアウォール管理センターからファイアウォールログをストリーミングし、OCSF スキーマへのスキーマ変換を実行して Security Lake に保存します。

[統合ドキュメンテーション](#)

Claroty – xDome

統合タイプ: ソース

Claroty xDome ネットワーク内で検出されたアラートを最小限の設定で Security Lake に送信します。柔軟かつ迅速な導入オプションにより、xDome はネットワーク内の拡張されたモノのインターネット (XIoT) 資産 (IoT、IIoT、BMS 資産で構成される) を保護しながら、脅威の初期兆候を自動的に検出できます。

[統合ドキュメンテーション](#)

CMD Solutions

統合タイプ: サービス

CMD Solutions は、設計、自動化、継続的保証プロセスを通じてセキュリティを早期かつ継続的に統合することで、企業の俊敏性を高めるのに役立ちます。サービス統合として、CMD Solutions は、組織に Security Lake を実装するのに役立ちます。

[パートナーリンク](#)

Confluent – Amazon S3 Sink Connector

統合タイプ: ソース

Confluent は、完全に管理された事前構築済みのコネクタを使用して、データ統合を自動的に接続、設定、調整します。これで、Confluent S3 Sink Connector は、未加工のデータをネイティブの Parquet 形式で大規模に Security Lake に取り込むことができます。

[統合ドキュメンテーション](#)

Contrast Security

統合タイプ: ソース

統合のパートナー製品: コントラストアセスメント

Contrast Security Assessは、ウェブアプリ、API、マイクロサービスの脆弱性をリアルタイムで検出するIASTツールです。Assess は Security Lake と統合されているため、すべてのワークロードを一元的に可視化できます。

[統合ドキュメンテーション](#)

Cribl – Search

統合タイプ: サブスクライバー

Cribl Search を使用して Security Lake データを検索できます。

[統合ドキュメンテーション](#)

Cribl – Stream

統合タイプ: ソース

Cribl Streamを使用すると、サポートされている Criblのサードパーティ ソースから OCSF スキーマの Security Lake にデータを送信できます。

[統合ドキュメンテーション](#)

CrowdStrike – Falcon Data Replicator

統合タイプ: ソース

この統合は、継続的なストリーミング ベースで CrowdStrike Falcon Data Replicatorからデータを取得し、そのデータを OCSF スキーマに変換して、Security Lake に送信します。

[統合ドキュメンテーション](#)

CrowdStrike – Next Gen SIEM

統合タイプ: サブスクライバー

ネイティブ OCSF スキーマパーサーを搭載したCrowdStrike Falcon Next-Gen SIEMデータコネクタを使用して、Security Lake データの取り込みを簡素化します。は、比類のないセキュリティの深さ

と幅を1つの統合プラットフォームにまとめることで、脅威の検出、調査、対応にFalcon NG SIEM 変革をもたらします。

[統合ドキュメンテーション](#)

CyberArk – Unified Identify Security Platform

統合タイプ: ソース

CyberArk Audit Adapter AWS Lambda 関数である は、 からセキュリティイベントを収集CyberArk Identity Security Platformし、OCSF スキーマの Security Lake にデータを送信します。

[統合ドキュメンテーション](#)

Cyber Security Cloud – Cloud Fastener

統合タイプ: サブスクライバー

CloudFastener は Security Lake を活用して、クラウド環境のセキュリティデータの統合を容易にします。

[統合ドキュメンテーション](#)

DataBahn

統合タイプ: ソース

Security Data Fabric を使用して Security Lake DataBahn's のセキュリティデータを一元化します。

[統合ドキュメンテーション \(DataBahn ポータルにサインインしてドキュメンテーションを確認\)](#)

Darktrace – Cyber AI Loop

統合タイプ: ソース

Darktraceと Security Lake の統合により、Security Lake に Darktrace の自己学習機能が生じます。Cyber AI Loopからの洞察は、他のデータ ストリームや組織のセキュリティ スタックの要素と関連付けることができます。統合により、Darktrace モデル違反がセキュリティの検出結果として記録されます。

[統合ドキュメンテーション \(ドキュメンテーションを確認するには、Darktrace ポータルにサインインします\)](#)

Datadog

統合タイプ: サブスクリイバー

Datadog Cloud SIEMは、Security Lake のデータを含むクラウド環境に対する脅威をリアルタイムで検出し、DevOps チームとセキュリティ チームを 1 つのプラットフォームに統合します。

[統合ドキュメンテーション](#)

Deloitte – MXDR Cyber Analytics and AI Engine (CAE)

統合タイプ: サブスクリイバー、サービス

Deloitte MXDR CAEは、標準化されたセキュリティデータを迅速に保存、分析、視覚化するのに役立ちます。カスタマイズされた分析、AI、ML 機能で構成される CAE スイートは、Security Lake 内の OCSF 形式のデータに対して実行されるモデルに基づいて、実用的なインサイトを自動的に提供します。

サービス統合として、Deloitte は組織への Security Lake の実装にも役立ちます。

[統合ドキュメンテーション](#)

Devo

統合タイプ: サブスクリイバー

Devoコレクターは、Security Lake からの取り込み AWS をサポートします。この統合は、脅威の検出、調査、インシデント対応など、さまざまなセキュリティユースケースの分析と対処に役立ちます。

[統合ドキュメンテーション](#)

DXC – SecMon

統合タイプ: サブスクリイバー、サービス

DXC SecMon は、Security Lake からセキュリティイベントを収集して監視し、潜在的なセキュリティ脅威を検出して警告します。これにより、組織は自社のセキュリティ体制をよりよく理解し、脅威を事前に特定して対応することができます。

サービス統合として、DXC は組織に Security Lake を実装するのに役立ちます。

[統合ドキュメンテーション](#)

Eviden— Alsaac (旧 Atos)

統合タイプ: サブスクライバー

Alsaac MDRこのプラットフォームは、Security Lake の OCSF スキーマに取り込まれた VPC フローログを使用し、AI モデルを利用して脅威を検出します。

[統合ドキュメンテーション](#)

ExtraHop – Reveal(x) 360

統合タイプ: ソース

IOC の検出を含むネットワーク データを ExtraHop Reveal(x) 360 から OCSF スキーマの Security Lake に統合することで、ワークロードとアプリケーションのセキュリティを強化できます。

[統合ドキュメンテーション](#)

Falcosidekick

統合タイプ: ソース

Falcosidekickは、Falcoのイベントを収集してSecurity Lakeに送信します。この統合は、OCSF スキーマを使用してセキュリティイベントをエクスポートします。

[統合ドキュメンテーション](#)

Fortinet - Cloud Native Firewall

統合タイプ: ソース

で CNF FortiGate インスタンスを作成するときに AWS、ログ出力先として Amazon Security Lake を指定できます。

[統合ドキュメンテーション](#)

Gigamon – Application Metadata Intelligence

統合タイプ: ソース

Gigamon Application Metadata Intelligence (AMI)オブザーバビリティ、SIEM、ネットワークパフォーマンス監視ツールに重要なメタデータ属性を付与します。これにより、アプリケーションの可視性が向上し、パフォーマンスのボトルネック、品質上の問題、潜在的なネットワークセキュリティリスクを特定できます。

[統合ドキュメンテーション](#)

Hoop Cyber

統合タイプ: サービス

Hoop Cyber FastStart には、データソースの評価、優先順位付け、データソースのオンボーディングが含まれ、Security Lake を通じて提供される既存のツールや統合を使用してデータをクエリできるようになっています。

[パートナーリンク](#)

HTCD – AI-First Cloud Security Platform

統合タイプ: サブスクリイバー

コンプライアンスの即時自動化、セキュリティ検出結果の優先順位付け、カスタマイズされたパッチを取得します。HTCD は Security Lake にクエリを実行して、自然言語クエリと AI 主導のインサイトで脅威を発見するのに役立ちます。

[統合ドキュメンテーション](#)

IBM – QRadar

統合タイプ: サブスクリイバー

IBM Security QRadar SIEM with UAXは、Security Lakeをハイブリッドクラウド全体の脅威を特定して防止する分析プラットフォームと統合します。この統合は、データアクセスとクエリアクセスの両方をサポートします。

[AWS CloudTrail ログの消費に関する統合ドキュメント](#)

[Amazon Athena をクエリに使用する方法に関する統合ドキュメンテーション](#)

Infosys

統合タイプ: サービス

Infosys は、組織のニーズに合わせて Security Lake の実装をカスタマイズするのに役立ち、カスタムインサイトも提供します。

[パートナーリンク](#)

Insbuilt

統合タイプ: サービス

Insbuilt はクラウド コンサルティングサービスを専門としており、組織に Security Lake を導入する方法を理解するのに役立ちます。

[パートナーリンク](#)

Kyndryl – AIOps

統合タイプ: サブスクライバー、サービス

Kyndryl は Security Lake と統合することで、サイバーデータ、脅威インテリジェンス、AI を活用した分析の相互運用性を実現します。データアクセスサブスクライバーとして、は分析目的で Security Lake Kyndryl から AWS CloudTrail 管理イベントを取り込みます。

サービス統合として、Kyndryl は組織に Security Lake を実装するのにも役立ちます。

[統合ドキュメンテーション](#)

Lacework – Polygraph

統合タイプ: ソース

Lacework Polygraph® Data Platform はデータソースとして Security Lake と統合され、AWS 環境全体の脆弱性、設定ミス、既知および未知の脅威に関するセキュリティ検出結果を提供します。

[統合ドキュメンテーション](#)

Laminar

統合タイプ: ソース

Laminarは、データセキュリティイベントを OCSF スキーマで Security Lake に送信し、インシデント対応や調査など、その他の分析ユースケースで利用できるようにします。

[統合ドキュメンテーション](#)

MegazoneCloud

統合タイプ: サービス

MegazoneCloud はクラウド コンサルティング サービスを専門としており、組織に Security Lake を導入する方法を理解するのに役立ちます。Security Lake を統合 ISV ソリューションと接続して、カスタムタスクを構築し、顧客のニーズに関連するカスタマイズされたインサイトを構築します。

[統合ドキュメンテーション](#)

Monad

統合タイプ: ソース

Monadは、データを自動的に OCSF スキーマに変換し、Security Lake データレイクに送信します。

[統合ドキュメンテーション](#)

NETSCOUT – Omnis Cyber Intelligence

統合タイプ: ソース

Security Lake と統合することで、NETSCOUT は、サイバー脅威、セキュリティリスク、アタックサーフェスの変化など、企業内で何が起きているかについて、セキュリティに関する検出結果と詳細なセキュリティに関するインサイトのカスタムソースになります。これらの検出結果は、NETSCOUT CyberStreams および Omnis Cyber Intelligence によってお客様のアカウントで生成され、OCSF スキーマで Security Lake に送信されます。取り込まれたデータは、フォーマット、スキーマ、パーティショニング、パフォーマンス関連の要素など、Security Lake ソースのその他の要件やベストプラクティスも満たしています。

[統合ドキュメンテーション](#)

Netskope – CloudExchange

統合タイプ: ソース

Netskope セキュリティ関連のログと脅威情報を Security Lake と共有することで、セキュリティ体制を強化するのに役立ちます。Netskope の検出結果は、CloudExchange プラグインを使用して

Security Lake に送信されます。プラグインは、ローカルデータセンター内 AWS またはローカルデータセンター内のドッカーベースの環境として起動できます。

[統合ドキュメンテーション](#)

New Relic ONE

統合タイプ: サブスクライバー

New Relic ONEは Lambda ベースのサブスクライバーアプリケーションです。これはアカウントにデプロイされ、Amazon SQS によってトリガーされ、New Relic ライセンスキーを使用してデータを New Relic に送信します。

[統合ドキュメンテーション](#)

Okta – Workforce Identity Cloud

統合タイプ: ソース

Okta は、Amazon EventBridge 統合を通じて OCSF スキーマの Security Lake に ID ログを送信します。OCSF スキーマの Okta System Logs は、セキュリティ チームとデータ サイエンティスト チームがオープン ソース標準によってセキュリティ イベントをクエリするのに役立ちます。Okta から標準化された OCSF ログを生成すると、一貫したスキーマのもとで監査活動を行い、認証、認可、アカウント変更、エンティティ変更に関するレポートを生成するのに役立ちます。

[統合ドキュメンテーション](#)

[AWS CloudFormation Security Lake のカスタムソースOktaとして追加する テンプレート](#)

Orca – Cloud Security Platform

統合タイプ: ソース

のOrcaエージェントレスクラウドセキュリティプラットフォームは、OCSF スキーマで Cloud Detection and Response (CDR) イベントを送信することで Security Lake と AWS 統合します。

[統合ドキュメンテーション \(Orca ポータルにサインインしてドキュメンテーションを確認\)](#)

Palo Alto Networks – Prisma Cloud

統合タイプ: ソース

Palo Alto Networks Prisma Cloudクラウドネイティブ環境の VM 全体の脆弱性検出データを集約し、Security Lake に送信します。

[統合ドキュメンテーション](#)

Palo Alto Networks – XSOAR

統合タイプ: サブスクリイバー

Palo Alto Networks XSOAR は、XSOAR および Security Lake とのサブスクリイバー統合を構築しました。

[統合ドキュメンテーション](#)

Panther

統合タイプ: サブスクリイバー

Panther は、検索と検出に使用する Security Lake ログの取り込みをサポートしています。

[統合ドキュメンテーション](#)

Ping Identity – PingOne

統合タイプ: ソース

PingOneは、アカウント変更アラートを OCSF スキーマと Parquet 形式で Security Lake に送信し、アカウントの変更を検出して対処できるようにします。

[統合ドキュメンテーション](#)

PwC – Fusion center

統合タイプ: サブスクリイバー、サービス

PwCは知識と専門知識を駆使して、クライアントの個々のニーズを満たすフュージョンセンターの導入を支援します。Amazon Security Lake 上に構築されたフュージョンセンターでは、さまざまなソースからのデータを組み合わせて、一元化されたほぼリアルタイムのビューを作成できます。

[統合ドキュメンテーション](#)

Query.AI – Query Federated Search

統合タイプ: サブスクライバー

Query Federated Search は、Amazon Athena 経由で任意の Security Lake テーブルを直接クエリして、OCSF スキーマ内のさまざまなオブザーバビリティ、イベント、オブジェクトにわたるインシデント対応、調査、脅威ハンティング、一般的な検索をサポートします。

[統合ドキュメンテーション](#)

Rapid7 – InsightIDR

統合タイプ: サブスクライバー

InsightIDRRapid7SIEM/XDR ソリューションでは、Security Lakeにログを取り込み、脅威の検出や疑わしいアクティビティの調査を行うことができます。

[統合ドキュメンテーション](#)

RipJar – Labyrinth for Threat Investigations

統合タイプ: サブスクライバー

Labyrinth for Threat Investigationsは、きめ細かなセキュリティ、適応性の高いワークフロー、レポート機能を備えた、データフュージョンに基づく大規模な脅威調査への全社的なアプローチを提供します。

[統合ドキュメンテーション](#)

Sailpoint

統合タイプ: ソース

統合のためのパートナー製品: SailPoint IdentityNow

この統合により、顧客は SailPoint IdentityNow からのイベントデータを変換できるようになります。この統合は、自動化されたプロセスを提供し、IdentityNow ユーザー アクティビティとガバナンス イベントを Security Lake に取り込み、セキュリティ インシデントとイベント監視製品からの洞察を向上させることが目的です。

[統合ドキュメンテーション](#)

Securonix

統合タイプ: サブスクリイバー

Securonix Next-Gen SIEM は Security Lake と統合されており、セキュリティ チームがより迅速にデータを取り込み、検出および対応能力を拡張できるようになります。

[統合ドキュメンテーション](#)

SentinelOne

統合タイプ: サブスクリイバー

SentinelOne Singularity™ XDR プラットフォームは、Amazon Elastic Compute Cloud (Amazon EC2)、Amazon Elastic Container Service (Amazon ECS)、Amazon Elastic Kubernetes Service (Amazon EKS) を含む、オンプレミスおよびパブリッククラウドインフラストラクチャ上で実行されるエンドポイント、アイデンティティ、クラウドワークロードに対して、リアルタイムの検出と応答を拡張します。

[統合ドキュメンテーション \(SentinelOne ポータルにサインインしてドキュメンテーションを確認\)](#)

Sentra – Data Lifecycle Security Platform

統合タイプ: ソース

Sentra スキャン インフラストラクチャをアカウントにデプロイすると、Sentra は結果を取得して SaaS に取り込みます。これらの検出結果は、Sentra が保存し、後でクエリのために OCSF スキーマの Security Lake にストリーミングするメタデータです。

[統合ドキュメンテーション](#)

SOC Prime

統合タイプ: サブスクリイバー

SOC Prime は、Amazon OpenSearch Service および Amazon Athena を通じて Security Lake と統合し、ゼロトラストマイルストーンに基づいたスマートなデータオーケストレーションと脅威ハンティングを促進します。SOC Prime を使用すると、セキュリティチームは大量のアラートを生成することなく、脅威の可視性を高め、インシデントを調査できるようになります。OCSF スキーマで Athena と OpenSearch Service に自動的に変換できる再利用可能なルールとクエリにより、開発時間を節約できます。

[統合ドキュメンテーション](#)

Splunk

統合タイプ: サブスクライバー

Amazon Web Services の Splunk AWS アドオン (AWS) は、Security Lake からの取り込みをサポートしています。この統合により、Security Lake から OCSF スキーマのデータを購読することで、脅威の検出、調査、対応を迅速に行うことができます。

[統合ドキュメンテーション](#)

Stellar Cyber

統合タイプ: サブスクライバー

Stellar Cyber は Security Lake からログを消費し、レコードを Stellar Cyber データ レイクに追加します。このコネクタは OCSF スキーマを使用します。

[統合ドキュメンテーション](#)

Sumo Logic

統合タイプ: サブスクライバー

Sumo Logic は Security Lake からのデータを消費し AWS、オンプレミス、ハイブリッドクラウド環境全体で幅広い可視性を提供します。Sumo Logic は、セキュリティチームがすべてのセキュリティツールを包括的に可視化し、自動化し、脅威を監視できるようにします。

[統合ドキュメンテーション](#)

Swimlane – Turbine

統合タイプ: サブスクライバー

Swimlane は、OCSF スキーマの Security Lake からデータを取り込み、ローコード プレイブックとケース管理を通じてデータを送信して、より迅速な脅威の検出、調査、インシデント対応を促進します。

[統合ドキュメンテーション \(Swimlane ポータルにサインインしてドキュメンテーションを確認\)](#)

Sysdig Secure

統合タイプ: ソース

Sysdig Secure's クラウドネイティブアプリケーション保護プラットフォーム (CNAPP) は、セキュリティイベントを Security Lake に送信して、監視を最大化し、調査を合理化し、コンプライアンスを簡素化します。

[統合ドキュメンテーション](#)

Talon

統合タイプ: ソース

統合用パートナー製品: Talon エンタープライズブラウザ

安全で隔離されたブラウザベースのエンドポイント環境である Talon's Enterprise Browser は、Talon アクセス、データ保護、SaaS アクション、セキュリティイベントを Security Lake に送信し、検出、フォレンジック、調査のためにイベントを相互に関連付けるための可視性とオプションを提供します。

[統合ドキュメンテーション \(Talon ポータルにサインインしてドキュメンテーションを確認\)](#)

Tanium

統合タイプ: ソース

Tanium Unified Cloud Endpoint Detection, Management, and Securityプラットフォームは OCSF スキーマで Security Lake にインベントリデータを提供します。

[統合ドキュメンテーション](#)

TCS

統合タイプ: サービス

TCS AWS Business Unitはイノベーション、経験、才能を提供します。この統合は、10年にわたる共同価値創造、深い業界知識、技術的専門知識、そして提供に関する知恵によって支えられています。サービス統合として、TCS は組織への Security Lake の実装を支援します。

[統合ドキュメンテーション](#)

Tego Cyber

統合タイプ: サブスクライバー

Tego Cyber は Security Lake と統合されているため、潜在的なセキュリティ脅威を迅速に検出して調査するために役立ちます。Tego Cyber は、広範囲の期間とログソースにわたり多様な脅威インジケーターを関連付けることで、隠れた脅威を発見します。このプラットフォームはコンテキストに即した脅威インテリジェンスで強化されており、脅威の検出と調査において正確性やインサイトを提供します。

[統合ドキュメンテーション](#)

Tines – No-code security automation

統合タイプ: サブスクライバー

Tines No-code security automation は、Security Lake に一元管理されているセキュリティデータを活用することで、より正確な意思決定を支援します。

[統合ドキュメンテーション](#)

Torq – Enterprise Security Automation Platform

統合タイプ: ソース、サブスクライバー

Torq は、カスタム ソースおよびサブスクライバーの両方として Security Lake とシームレスに統合します。Torq は、シンプルなノーコードプラットフォームを使用してエンタープライズ規模の自動化とオーケストレーションを実装するのに役立ちます。

[統合ドキュメンテーション](#)

Trellix – XDR

統合タイプ: ソース、サブスクライバー

オープン XDR プラットフォームとして、Trellix XDR は Security Lake の統合をサポートしています。Trellix XDR は、セキュリティ分析のユースケースに OCSF スキーマのデータを活用できます。また、Trellix XDR のセキュリティ イベントの 1,000 以上のソースを使用して Security Lake データレイクを強化することもできます。これにより、AWS 環境の検出および対応機能を拡張できます。取り込まれたデータは他のセキュリティリスクと相関関係があり、リスクにタイムリーに対応するために必要なプレイブックが得られます。

[統合ドキュメンテーション](#)

Trend Micro – CloudOne

統合タイプ: ソース

Trend Micro CloudOne Workload Securityは、Amazon Elastic Compute Cloud (EC2) インスタンスから Security Lake に以下の情報を送信します。

- DNS クエリアクティビティ
- ファイルアクティビティ
- ネットワークアクティビティ
- プロセスアクティビティ
- レジストリ値アクティビティ
- ユーザーアカウントアクティビティ

[統合ドキュメンテーション](#)

Uptycs – Uptycs XDR

統合タイプ: ソース

Uptycsは、オンプレミスやクラウドのアセットから Security Lake に大量のデータを OCSF スキーマで送信します。データには、エンドポイントやクラウドワークロードからの行動上の脅威の検出、異常検知、ポリシー違反、危険なポリシー、設定ミス、脆弱性が含まれます。

[統合ドキュメンテーション](#)

Vectra AI – Vectra Detect for AWS

統合タイプ: ソース

を使用すると Vectra Detect for AWS、専用 AWS CloudFormation テンプレートを使用して、カスタムソースとして Security Lake に忠実度の高いアラートを送信できます。

[統合ドキュメンテーション](#)

VMware Aria Automation for Secure Clouds

統合タイプ: ソース

この統合により、クラウドの設定ミスを検出して Security Lake に送信し、高度な分析を行うことができます。

[統合ドキュメンテーション](#)

Wazuh

統合タイプ: サブスクライバー

Wazuh は、ユーザーデータを安全に処理し、各ソースにクエリアクセスを提供し、クエリコストを最適化することを目的としています。

[統合ドキュメンテーション](#)

Wipro

統合タイプ: ソース、サービス

この統合により、Wipro Cloud Application Risk Governance (CARG) プラットフォームからデータを収集して、クラウドアプリケーションと企業全体のコンプライアンス態勢を一元的に把握できます。

サービス統合として、Wiproは、組織に Security Lake を実装するのにも役立ちます。

[統合ドキュメンテーション](#)

Wiz – CNAPP

統合タイプ: ソース

Wiz と Security Lake の統合により、拡張可能で正規化されたセキュリティデータ交換用に設計されたオープンソース標準である OCSF スキーマを活用することで、単一のセキュリティデータレイクでのクラウドセキュリティデータの収集が容易になります。

[統合ドキュメンテーション \(Wiz ポータルにサインインしてドキュメンテーションを確認\)](#)

Zscaler – Zscaler Posture Control

統合タイプ: ソース

クラウドネイティブアプリケーション保護プラットフォームである Zscaler Posture Control™ は、セキュリティ検出結果を OCSF スキーマで Security Lake に送信します。

統合ドキュメンテーション

Security Lake のセキュリティ

で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 とお客様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、で AWS サービスを実行する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。AWS また、は、お客様が安全に使用できるサービスも提供します。[AWS コンプライアンスプログラム](#)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。Amazon Security Lake 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 による対象範囲内のサービスコンプライアンスプログラム](#)」を参照してください。
- クラウド内のセキュリティ – お客様の責任は、使用する AWS サービスによって決まります。また、ユーザーは、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、Security Lake を使用する際に責任共有モデルを適用する方法を理解するのに役立ちます。以下のトピックでは、セキュリティおよびコンプライアンスの目的を達成するために Security Lake を設定する方法を説明します。また、Security Lake リソースのモニタリングや保護に役立つ他の AWS サービスの使用方法についても説明します。

トピック

- [Security Lake の Identity and Access Management](#)
- [Amazon Security Lake におけるデータ保護](#)
- [Amazon SECURITY Lake のコンプライアンス検証](#)
- [Security Lake のセキュリティのベストプラクティス](#)
- [Amazon Security Lake の耐障害性](#)
- [Amazon Security Lake のインフラストラクチャセキュリティ](#)
- [Security Lake での構成と脆弱性の分析](#)
- [Amazon Security Lake とインターフェイス VPC エンドポイント \(AWS PrivateLink\)](#)
- [Amazon Security Lake のモニタリング](#)

Security Lake の Identity and Access Management

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、Security Lake 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) できるユーザーをコントロールします。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](#)
- [Security Lake と IAM の連携方法](#)
- [Security Lake のアイデンティティベースのポリシーの例](#)
- [AWS Security Lake の マネージドポリシー](#)
- [Security Lake のサービスにリンクされたロールの使用](#)

対象者

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

サービスユーザー - ユーザーがジョブを実行するために Security Lake サービスを使用する場合は、管理者から必要な認証情報と許可がそのユーザーに提供されます。作業を行うためにさらに多くの Security Lake の機能を使用する場合、追加の許可が必要になる可能性があります。アクセスの管理方法を理解すると、管理者に適切なアクセス許可をリクエストするのに役に立ちます。Security Lake の機能にアクセスできない場合は、「[Amazon Security Lake アイデンティティとアクセス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 - お客様が社内の Security Hub リソースを担当している場合は、通常 Security Lake に完全にアクセスすることができます。サービスユーザーがどの Security Lake 機能およびリソースにアクセスする必要があるかを決定するのは管理者の仕事です。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を点検して、IAM の基本概念を理解してください。お客様の会社で Security Lake で IAM を利用する方法の詳細については、「[Security Lake と IAM の連携方法](#)」を参照してください。

IAM 管理者 - IAM 管理者は、Security Lake へのアクセスを管理するポリシーの作成方法を詳しく確認する必要がある場合があります。IAM で使用できる Security Lake の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、「[Security Lake のアイデンティティベースのポリシーの例](#)」を参照してください。

アイデンティティを使用した認証

認証とは、ID 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 (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、フェデレーテッド ID AWS として にサインインできます。AWS IAM Identity Center (IAM アイデンティティセンター) ユーザー、会社のシングルサインオン認証、Google または Facebook 認証情報は、フェデレーテッド ID の例です。フェデレーテッド ID としてサインインする場合、IAM ロールを使用して、前もって管理者により ID フェデレーションが設定されています。フェデレーションを使用して にアクセスすると、間接的 AWS にロールを引き受けます。

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「[にサインインする方法 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 Software Development Kit (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で署名します。AWS ツールを使用しない場合は、リクエストに自分で署名する必要があります。リクエストに自分で署名する推奨方法の使用については、「IAM ユーザーガイド」の「[API リクエストに対する AWS Signature Version 4](#)」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強化す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](#)」および「IAM ユーザーガイド」の「[IAM の AWS 多要素認証](#)」を参照してください。

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 つのサインインアイデンティティから始めます。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することでアクセスできます。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

については、「IAM ユーザーガイド」の「[ルートユーザー認証情報が必要なタスク](#)」を参照してください。

フェデレーテッドアイデンティティ

ベストプラクティスとして、管理者アクセスを必要とするユーザーを含む人間のユーザーに、一時的な認証情報 AWS のサービス を使用して にアクセスするために ID プロバイダーとのフェデレーションを使用することを要求します。

フェデレーテッド ID は、エンタープライズユーザーディレクトリ、ウェブ ID プロバイダー、AWS Directory Service、アイデンティティセンターディレクトリのユーザー、または ID ソースを通じて提供された認証情報 AWS のサービス を使用して にアクセスするユーザーです。フェデレーテッド ID が にアクセスすると AWS アカウント、ロールを引き受け、ロールは一時的な認証情報を提供します。

アクセスを一元管理する場合は、AWS IAM Identity Centerを使用することをお勧めします。IAM Identity Center でユーザーとグループを作成することも、独自の ID ソース内のユーザーとグループのセットに接続して同期し、すべての AWS アカウント とアプリケーションで使用することもできます。IAM Identity Center の詳細については、「AWS IAM Identity Center ユーザーガイド」の「[What is IAM Identity Center?](#)」(IAM Identity Center とは) を参照してください。

IAM ユーザーとグループ

[IAM ユーザー](#)は、単一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 内のアイデンティティです。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保有する IAM ユーザーを作成する代わりに、一時的な認証情報を使用することをお勧めします。ただし、IAM ユーザーでの長期的な認証情報が必要な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ために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 ユーザーの集団を指定するアイデンティティです。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、IAMAdmins という名前のグループを設定して、そのグループに IAM リソースを管理する許可を与えることができます。

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「[IAM ユーザーに関するユースケース](#)」を参照してください。

IAM ロール

[IAM ロール](#)は、特定のアクセス許可 AWS アカウント を持つ内のアイデンティティです。これは IAM ユーザーに似ていますが、特定のユーザーには関連付けられていません。IAM ロールを一時的に引き受けるには AWS Management Console、[ユーザーから IAM ロールに切り替えることができます \(コンソール\)](#)。ロールを引き受けるには、または AWS API オペレーションを AWS CLI 呼び出すか、カスタム URL を使用します。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ための各種方法](#)」を参照してください。

IAM ロールと一時的な認証情報は、次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 \(フェデレーション\) のロールを作成する](#)」を参照してください。IAM Identity Center を使用する場合は、許可セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM Identity Center は、権限セットを IAM のロールに関連付けます。アクセス許可セットの詳細については、「AWS IAM Identity Center User Guide」の「[Permission sets](#)」を参照してください。
- 一時的な IAM ユーザー権限 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なる権限を一時的に IAM ロールで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、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の AWS サービス、(ロールをプロキシとして使用する代わりに) 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における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「[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。
- クロスサービスアクセス — 一部の AWS サービスでは、他の AWS の機能 AWS のサービスを使用します AWS のサービス。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。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リクエストする を組み合わせて使用します。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アクセス許可が必要です。FAS リクエストを行う際のポリシーの詳細については、「[転送アクセスセッション](#)」を参照してください。
- サービスロール - サービスがユーザーに代わってアクションを実行するために引き受ける [IAM ロール](#)です。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除することができます。詳細については、「IAM ユーザーガイド」の「[AWS のサービスに許可を委任するロールを作成する](#)」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、サービスリンク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。
- Amazon EC2 で実行されているアプリケーション – 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 リクエストを実行している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 インスタンス内でのアクセスキーの保存に推奨されます。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、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、「IAM ユーザーガイド」の「[Amazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用して許可を付与する](#)」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 AWS、ポリシーを作成し、ID AWS またはリソースにアタッチします。ポリシーは のオブジェクト AWS であり、アイデンティティまたはリソースに関連付けられると、そのアクセス許可を定義します。 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、IAM ユーザーガイドの [JSON ポリシー概要](#) を参照してください。

管理者は JSON AWS ポリシーを使用して、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。IAM 管理者は、リソースに必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。その後、管理者はロールに IAM ポリシーを追加し、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方法を問わず、アクションの許可を定義します。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 API からロール情報を取得できます。

アイデンティティベースのポリシー

アイデンティティベースポリシーは、IAM ユーザーグループ、ユーザーのグループ、ロールなど、アイデンティティにアタッチできる JSON 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。アイデンティティベースポリシーの作成方法については、「IAM ユーザーガイド」の [「カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する」](#) 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。管理ポリシーは、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のいずれかを選択する方法については、「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のいずれかを選択する」](#) 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、リソースに添付する 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には例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。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制御できます。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では、IAM の AWS マネージドポリシーを使用できません。

アクセスコントロールリスト (ACL)

アクセスコントロールリスト (ACL) 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ための許可を持つかを制御します。ACL 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しません。

Amazon S3、AWS WAF、および Amazon VPC は、ACLs。ACL の詳細については、「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 \(ACL\) の概要](#)」を参照してください。

その他のポリシータイプ

AWS は、一般的ではない追加のポリシータイプをサポートしています。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ポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる権限の上限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 AWS アカウント をグループ化して一元管理するためのサービスです。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 (SCP) を一部またはすべてのアカウントに適用できます。SCP 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を制限します AWS アカウントのルートユーザー。Organizations と SCP の詳細については、「AWS Organizations ユーザーガイド」の「[サービスコントロールポリシー \(SCP\)](#)」を参照してください。
- **リソースコントロールポリシー (RCP)** – RCP は、所有する各リソースにアタッチされた IAM ポリシーを更新することなく、アカウント内のリソースに利用可能な最大数のアクセス許可を設定する

ために使用できる JSON ポリシーです。RCP は、メンバーアカウントのリソースのアクセス許可を制限し、組織に属しているかどうかにかかわらず AWS アカウントのルートユーザー、を含む ID の有効なアクセス許可に影響を与える可能性があります。RCP をサポートする のリストを含む Organizations と RCP の詳細については、AWS Organizations RCPs [「リソースコントロールポリシー \(RCPs\)」](#) を参照してください。AWS のサービス

- 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があります。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、「IAM 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。

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。複数のポリシータイプが関係する場合に、ガリクエストを許可するかどうかが AWS を決定する方法については、「IAM 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。

Security Lake と IAM の連携方法

IAM を使用して Security Lake へのアクセスを管理する前に、Security Lake で使用できる IAM 機能を確認してください。

Amazon Security Lake で使用できる IAM の機能

IAM 機能	Security Lake サポート
アイデンティティベースポリシー	はい
リソースベースのポリシー	はい
ポリシーアクション	はい
ポリシーリソース	あり
ポリシー条件キー	Yes

IAM 機能	Security Lake サポート
ACL	いいえ
ABAC (ポリシー内のタグ)	あり
一時的な認証情報	はい
プリンシパル権限	はい
サービスロール	いいえ
サービスリンクロール	はい

Security Lake およびその他の AWS のサービスがほとんどの IAM 機能と連携する方法の概要を把握するには、「IAM ユーザーガイド」の[AWS 「IAM と連携する のサービス」](#)を参照してください。

Security Lake のアイデンティティベースのポリシー

アイデンティティベースのポリシーのサポート: あり

アイデンティティベースポリシーは、IAM ユーザーグループ、ユーザーのグループ、ロールなど、アイデンティティにアタッチできる JSON 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 ベースのポリシーの作成方法については、「IAM ユーザーガイド」の「[カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する](#)」を参照してください。

IAM アイデンティティベースのポリシーでは、許可または拒否するアクションとリソース、およびアクションを許可または拒否する条件を指定できます。プリンシパルは、それが添付されているユーザーまたはロールに適用されるため、アイデンティティベースのポリシーでは指定できません。JSON ポリシーで使用できるすべての要素について学ぶには、「IAM ユーザーガイド」の「[IAM JSON ポリシーの要素のリファレンス](#)」を参照してください。

Security Lakeは、アイデンティティベースのポリシーをサポートします。詳細については、「[Security Lake のアイデンティティベースのポリシーの例](#)」を参照してください。

Security Lake 内のリソースベースのポリシー

リソースベースのポリシーのサポート: あり

リソースベースのポリシーは、リソースに添付する 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には例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。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制御できます。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。

クロスアカウントアクセスを有効にするには、アカウント全体、または別のアカウントの IAM エンティティをリソースベースのポリシーのプリンシパルとして指定します。リソースベースのポリシーにクロスアカウントのプリンシパルを追加しても、信頼関係は半分しか確立されない点に注意してください。プリンシパルとリソースが異なる場合 AWS アカウント、信頼されたアカウントの IAM 管理者は、リソースにアクセスするためのアクセス許可をプリンシパルエンティティ (ユーザーまたはロール) に付与する必要もあります。IAM 管理者は、アイデンティティベースのポリシーをエンティティにアタッチすることで権限を付与します。ただし、リソースベースのポリシーで、同じアカウントのプリンシパルへのアクセス権が付与されている場合は、アイデンティティベースのポリシーをさらに付与する必要はありません。詳細については、「IAM ユーザーガイド」の「[IAM でのクロスアカウントリソースアクセス](#)」を参照してください。

Security Lake サービスは、データを保存する Amazon S3 バケットのリソースベースのポリシーを作成します。S3 バケットには、これらのリソースベースのポリシーをアタッチしないでください。Security Lake はユーザーに代わってこれらのポリシーを自動的に作成します。

リソースの例としては、arn:aws:s3:::aws-security-data-lake-{region}-{bucket-identifier} の Amazon リソースネーム (ARN) の S3 バケットがあります。この例では、region は Security Lake を有効にした特定の AWS リージョンであり、bucket-identifier は Security Lake がバケットに割り当てるリージョン固有の英数字文字列です。Security Lake は S3 バケットを作成して、そのリージョンのデータを保存します。リソースポリシーは、バケットに対してアクションを実行できるプリンシパルを定義します。Security Lake がバケットにアタッチするリソースベースのポリシー (バケットポリシー) の例を次に示します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Principal": {
        "AWS": "*"
      }
    }
  ]
}
```

```
    },
    "Action": "s3:*",
    "Resource": [
      "arn:aws:s3::aws-security-data-lake-{region}-{bucket-identifier}/*",
      "arn:aws:s3::aws-security-data-lake-{region}-{bucket-identifier}"
    ],
    "Condition": {
      "Bool": {
        "aws:SecureTransport": "false"
      }
    }
  },
  {
    "Sid": "PutSecurityLakeObject",
    "Effect": "Allow",
    "Principal": {
      "Service": "securitylake.amazonaws.com"
    },
    "Action": "s3:PutObject",
    "Resource": [
      "arn:aws:s3::aws-security-data-lake-{region}-{bucket-identifier}/*",
      "arn:aws:s3::aws-security-data-lake-{region}-{bucket-identifier}"
    ],
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "{DA-AccountID}",
        "s3:x-amz-acl": "bucket-owner-full-control"
      },
      "ArnLike": {
        "aws:SourceArn": "arn:aws:securitylake:us-east-1:{DA-AccountID}:*"
      }
    }
  }
]
```

リソースベースのポリシーの詳細については、IAM ユーザーガイドの「[アイデンティティベースおよびリソースベースのポリシー](#)」を参照してください。

Security Lake のポリシーアクション

ポリシーアクションのサポート:あり

管理者は JSON AWS ポリシーを使用して、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 要素にはポリシー内の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が記述されます。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 オペレーションのない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アクセス許可を付与するポリシーで使用されます。

Security Lake アクションのリストについては、「サービス認可リファレンス」の「[Amazon Security Lake によって定義されたアクション](#)」を参照してください。

Security Lake のポリシーアクションは、アクションの前に以下のプレフィックスを使用します。

```
securitylake
```

例えば、ユーザーに特定のサブスクライバーに関する情報へのアクセス許可を付与するには、ユーザーに割り当てるポリシーに `securitylake:GetSubscriber` 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。Security Lake では、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のセットを定義します。

単一のステートメントで複数のアクションを指定するには、アクションをカンマで区切ります。

```
"Action": [  
  "securitylake:action1",  
  "securitylake:action2"  
]
```

Security Lake 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、[Security Lake のアイデンティティベースのポリシーの例](#) を参照してください。

Security Lake のポリシーリソース

ポリシーリソースのサポート: あり

管理者は JSON AWS ポリシーを使用して、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。

Resource JSON ポリシー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": "*"
```

Security Lake は、サブスクライバー、および AWS アカウント 特定の のデータレイク設定のリソースタイプを定義します AWS リージョン。ARN を使用して、ポリシーでこれらのタイプのリソースを指定できます。

Security Lake リソースタイプのリストとそれぞれの ARN 構文については、「サービス認可リファレンス」の「[Amazon Security Lake によって定義されたリソースタイプ](#)」を参照してください。リソースタイプごとに指定できるアクションについては、「サービス認可リファレンス」の「[Amazon Security Lake で定義されるアクション](#)」を参照してください。

Security Lake 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、[Security Lake のアイデンティティベースのポリシーの例](#) を参照してください。

Security Lake のポリシー条件キー

サービス固有のポリシー条件キーのサポート: あり

管理者は JSON AWS ポリシーを使用して、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して、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进行评估します。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 を评估します。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 ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる権限を付与することができます。詳細については、「IAM ユーザーガイド」の「[IAM ポリシーの要素: 変数およびタグ](#)」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、「IAM ユーザーガイド」の[AWS 「グローバル条件コンテキストキー」](#)を参照してください。

Security Lake 条件キーのリストについては、「サービス認可リファレンス」の「[Amazon Security Lake の条件キー](#)」を参照してください。条件キーを使用できるアクションとリソースについては、「サービス認可リファレンス」の「[Amazon Security Lake で定義されるアクション](#)」を参照してください。条件キーを使用するポリシーの例については、「[Security Lake のアイデンティティベースのポリシーの例](#)」を参照してください。

Security Lake のアクセスコントロールリスト (ACL)

ACL のサポート: なし

アクセスコントロールリスト (ACL) 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ための許可を持つかを制御します。ACL 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しません。

Security Lake は ACL をサポートしていないため、Security Lake リソースに ACL をアタッチすることはできません。

Security Lake での属性ベースのアクセスコントロール (ABAC)

ABAC (ポリシー内のタグ) のサポート: あり

属性ベースのアクセス制御 (ABAC) は、属性に基づいてアクセス許可を定義する認可戦略です。では AWS、これらの属性はタグと呼ばれます。タグは、IAM エンティティ (ユーザーまたはロール) および多くの AWS リソースにアタッチできます。エンティティとリソースのタグ付けは、ABAC の最初の手順です。その後、プリンシパルのタグがアクセスしようとしているリソースのタグと一致した場合にオペレーションを許可するように ABAC ポリシーをします。

ABAC は、急成長する環境やポリシー管理が煩雑になる状況で役立ちます。

タグに基づいてアクセスを管理するには、`aws: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。

サービスがすべてのリソースタイプに対して 3 つの条件キーすべてをサポートする場合、そのサービスの値はありです。サービスが一部のリソースタイプに対してのみ 3 つの条件キーのすべてをサポートする場合、値は「部分的」になります。

ABAC の詳細については、「IAM ユーザーガイド」の「[ABAC 認可でアクセス許可を定義する](#)」を参照してください。ABAC をセットアップする手順を説明するチュートリアルについては、「IAM ユーザーガイド」の「[属性ベースのアクセスコントロール \(ABAC\) を使用する](#)」を参照してください。

Security Lake リソース、サブスクライバー、および個々のデータレイク設定 AWS アカウントにタグをアタッチできます AWS リージョン。ポリシーの Condition 要素にタグ情報を指定することで、これらの種類のリソースへのアクセスを制御することもできます。Security Lake リソースのタグ付けの詳細については、[Security Lake リソースのタグ付け](#) を参照してください。リソースのタグに基づいてリソースへのアクセスを制御する ID ベースのポリシーの例については、「[Security Lake のアイデンティティベースのポリシーの例](#)」を参照してください。

Security Lake で一時的なセキュリティ認証情報を使用する

一時的な認証情報のサポート: あり

一部の AWS のサービスは、一時的な認証情報を使用してサインインすると機能しません。一時的な認証情報 AWS のサービスを使用する機能などの詳細については、[AWS のサービス「IAM ユーザーガイド」の「IAM と連携する](#)」を参照してください。

ユーザー名とパスワード以外の AWS Management Console 方法でサインインする場合は、一時的な認証情報を使用します。例えば、会社のシングルサインオン (SSO) リンク AWS を使用してアクセスすると、そのプロセスによって一時的な認証情報が自動的に作成されます。また、ユーザーとしてコンソールにサインインしてからロールを切り替える場合も、一時的な認証情報が自動的に作成されます。ロールの切り替えに関する詳細については、「IAM ユーザーガイド」の「[ユーザーから IAM ロールに切り替える \(コンソール\)](#)」を参照してください。

一時的な認証情報は、AWS CLI または AWS API を使用して手動で作成できます。その後、これらの一時的な認証情報を使用してアクセスすることができます AWS。長期的なアクセスキーを使用

する代わりに、一時的な認証情報 AWS を動的に生成することをお勧めします。詳細については、「[IAM の一時的セキュリティ認証情報](#)」を参照してください。

Security Lake は、一時的な認証情報の使用をサポートしています。

Security Lake の転送アクセスセッション

転送アクセスセッション (FAS) のサポート: あり

IAM ユーザーまたはロールを使用して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リクエストを組み合わせ使用します。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アクセス許可が必要です。FAS リクエストを行う際のポリシーの詳細については、「[転送アクセスセッション](#)」を参照してください。

Security Lake のアクションの中には、他の AWS のサービスにおけるアクションに依存する追加のアクションに対する権限が必要なものもあります。これらのアクションのリストについては、「サービス認可リファレンス」の「[Amazon Security Lake によって定義されたアクション](#)」を参照してください。

Security Lake のサービスロール

サービスロールのサポート: なし

サービスロールとは、サービスがユーザーに代わってアクションを実行するために引き受ける [IAM ロール](#) です。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、「IAM ユーザーガイド」の「[AWS のサービスに許可を委任するロールを作成する](#)」を参照してください。

Security Lake はサービスロールを引き受けたり使用したりしません。ただし、Security Lake を使用するときには AWS Lambda、Amazon EventBridge や Amazon S3 などの関連サービスがサービスロールを引き受けます。ユーザーに代わってアクションを実行するために、Security Lake はサービスリンクロールを使用します。

⚠ Warning

サービスロールの権限を変更すると、Security Lake を使用する際に運用上の問題が発生する可能性があります。Security Lake がそのためのガイダンスを提供している場合にのみ、サービスロールを編集してください。

Security Lake のサービスリンクロール

サービスリンクロールのサポート: あり

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、サービスリンクロールの許可を表示できますが、編集することはできません。

Security Lake は、AWSServiceRoleForAmazonSecurityLake という名前の IAM サービスリンクロールを使用します。Security Lake のサービスリンクロールは、顧客に代わってセキュリティデータレイクサービスを運用する権限を付与します。このサービスリンクロールは、Security Lake に直接リンクされている IAM ロールです。これは Security Lake によって事前定義されており、AWS のサービス Security Lake がユーザーに代わって他の 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。Security Lake は、Security Lake AWS リージョン が利用可能なすべてのので、このサービスにリンクされたロールを使用します。

Security Lake サービスリンクロールの作成または管理の詳細については、「[Security Lake のサービスにリンクされたロールの使用](#)」を参照してください。

Security Lake のアイデンティティベースのポリシーの例

デフォルトでは、ユーザーおよびロールには、Security Lake リソースを作成または変更する許可はありません。また、AWS Command Line Interface (AWS CLI) AWS Management Console、または AWS API を使用してタスクを実行することはできません。IAM 管理者は、リソースに必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き継ぐことができます。

これらサンプルの JSON ポリシードキュメントを使用して、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「[IAM ポリシーを作成する \(コンソール\)](#)」を参照してください。

各リソースタイプの ARN の形式など、Security Lake によって定義されたアクションとリソースタイプの詳細については、「サービス認可リファレンス」の「[Amazon Security Lake のアクション、リソース、および条件キー](#)」を参照してください。

トピック

- [ポリシーに関するベストプラクティス](#)
- [Security Lake コンソールの使用](#)
- [例: ユーザーにそれぞれのアクセス権限の表示を許可する](#)
- [例: 組織管理アカウントに委任された管理者の指定と削除を許可](#)
- [例: タグに基づいてユーザーが購読者をレビューするのを許可する](#)

ポリシーに関するベストプラクティス

ID ベースのポリシーは、あるユーザーがアカウント内で Security Lake 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 管理ポリシーの使用を開始し、最小特権のアクセス許可に移行する - ユーザーとワークロードにアクセス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にアクセス許可を付与する AWS 管理ポリシーを使用します。これらはで使用できます AWS アカウント。ユースケースに固有の AWS カスタマー管理ポリシーを定義して、アクセス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「[ジョブ機能のAWS マネージドポリシー](#)」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場合は、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できます。条件を使用して、サービスアクションがなどの特定のを通じて使用される場合に AWS のサービス、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 AWS CloudFormation。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。

- IAM Access Analyzer を使用して IAM ポリシーを検証し、安全で機能的な権限を確保する - 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 アクセスマナライザーは 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の作成をサポートします。詳細については、「IAM ユーザーガイド」の「[IAM Access Analyzer でポリシーを検証する](#)」を参照してください。
- 多要素認証 (MFA) を要求する - で IAM ユーザーまたはルートユーザーを必要とするシナリオがある場合は AWS アカウント、セキュリティを強化するために MFA を有効にします。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 を使用した安全な API アクセス](#)」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。

Security Lake コンソールの使用

Amazon Security Lake コンソールにアクセスするには、許可の最小限のセットが必要です。これらのアクセス許可により、内の Security Lake リソースの詳細を一覧表示および表示できます AWS アカウント。最小限必要な許可よりも制限が厳しいアイデンティティベースのポリシーを作成すると、そのポリシーを持つエンティティ (ユーザーまたはロール) に対してコンソールが意図したとおりに機能しません。

AWS CLI または AWS API のみを呼び出すユーザーには、最小限のコンソールアクセス許可を付与する必要はありません。代わりに、実行しようとしている API オペレーションに一致するアクションのみへのアクセスが許可されます。

ユーザーとロールが Security Lake コンソールを使用できるようにするには、それらにコンソールアクセスを提供する IAM ポリシーを作成します。詳細については、IAM ユーザーガイドの「[IAM アイデンティティ](#)」を参照してください。

ユーザーまたはロールに Security Lake コンソールの使用を許可するポリシーを作成する場合は、そのユーザーまたはロールがコンソールでアクセスする必要のあるリソースに対する適切なアクションをポリシーに必ず含めます。そうしないと、コンソールでそれらのリソースに移動したり、リソースの詳細を表示したりすることができなくなります。

たとえば、コンソールを使用してカスタムソースを追加するには、ユーザーに以下のアクションの実行を許可する必要があります。

- glue:CreateCrawler
- glue:CreateDatabase
- glue:CreateTable
- glue:StartCrawlerSchedule
- iam:GetRole
- iam:PutRolePolicy
- iam>DeleteRolePolicy
- iam:PassRole
- lakeformation:RegisterResource
- lakeformation:GrantPermissions
- s3:ListBucket
- s3:PutObject

例: ユーザーにそれぞれのアクセス権限の表示を許可する

この例では、ユーザーアイデンティティにアタッチ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の作成方法を示します。このポリシーには、コンソールで、または AWS CLI または AWS API を使用してプログラムでこのアクションを実行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:ListGroupsWithUser",
        "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": "*"
  }
]
```

例: 組織管理アカウントに委任された管理者の指定と削除を許可

この例では、AWS Organizations 管理アカウントのユーザーが組織の委任された Security Lake 管理者を指定して削除することを許可するポリシーを作成する方法を示します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "securitylake:RegisterDataLakeDelegatedAdministrator",
        "securitylake:DeregisterDataLakeDelegatedAdministrator"
      ],
      "Resource": "arn:aws:securitylake:*:*:*"
    }
  ]
}
```

例: タグに基づいてユーザーが購読者をレビューするのを許可する

ID ベースのポリシーでは、条件を使用して、タグに基づいて Security Lake リソースへのアクセスを制御できます。この例では、Security Lake コンソールまたは Security Lake API を使用して、ユーザーがサブスクライバーを確認することを許可するポリシーを作成する方法を示します。ただし、アクセス許可は、サブスクライバの `owner` タグの値がユーザーのユーザー名である場合にのみ付与されます。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ReviewSubscriberDetailsIfOwner",
      "Effect": "Allow",
      "Action": "securitylake:GetSubscriber",
      "Resource": "arn:aws:securitylake:*:*:subscriber/*",
      "Condition": {
        "StringEquals": {"aws:ResourceTag/Owner": "${aws:username}"}
      }
    },
    {
      "Sid": "ListSubscribersIfOwner",
      "Effect": "Allow",
      "Action": "securitylake:ListSubscribers",
      "Resource": "*",
      "Condition": {
        "StringEquals": {"aws:ResourceTag/Owner": "${aws:username}"}
      }
    }
  ]
}
```

この例では、ユーザー名 richard-roe を持つユーザーが個々の購読者の詳細を確認しようとする
と、購読者には Owner=richard-roe または owner=richard-roe のタグが付けられる必要が
あります。それ以外の場合、ユーザー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
は区別されないため、条件タグキー Owner は Owner と owner に一致します。条件キーの詳細に
ついては、「IAM ユーザーガイド」の [IAM JSON ポリシーの要素: Condition](#) を参照してくださ
い。Security Lake リソースのタグ付けの詳細については、[Security Lake リソースのタグ付け](#) を参照
してください。

AWS Security Lake の マネージドポリシー

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、ロールに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ります。ユースケースに固有の[カスタマー管理ポリシー](#)を定義して、アクセス許可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) に影響します。AWS 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 オペレーションが利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。

AWS マネージドポリシー: AmazonSecurityLakeMetastoreManager

Amazon Security Lake は、AWS Lambda 関数を使用してデータレイク内のメタデータを管理します。この関数を使用することで、Security Lake は、データとデータファイルを含む Amazon Simple Storage Service (Amazon S3) パーティションを AWS Glue Data Catalog テーブルにインデックス化できます。この管理ポリシーには、Lambda 関数が S3 パーティションとデータファイルを AWS Glue テーブルにインデックスするためのすべてのアクセス許可が含まれています。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- logs – プリンシパルが Lambda 関数の出力を Amazon CloudWatch Logs にログ記録できるようにします。
- glue – プリンシパルが AWS Glue Data Catalog テーブルに対して特定の書き込みアクションを実行できるようにします。これにより、AWS Glue クローラーはデータ内のパーティションを識別することもできます。
- sqs – プリンシパルが Amazon SQS キューに対して特定の読み取りおよび書き込みアクションを実行し、オブジェクトがデータレイクに追加または更新されたときにイベント通知を送信できるようにします。
- s3 – プリンシパルがデータを含む Amazon S3 バケットに対して特定の読み取りおよび書き込みアクションを実行できるようにします。

```
{
```

```
"Version": "2012-10-17",
"Statement": [
  {
    "Sid": "AllowWriteLambdaLogs",
    "Effect": "Allow",
    "Action": [
      "logs:CreateLogStream",
      "logs:PutLogEvents",
      "logs:CreateLogGroup"
    ],
    "Resource": [
      "arn:aws:logs:*:*:log-group:/aws/lambda/AmazonSecurityLake*",
      "arn:aws:logs:*:*/aws/lambda/AmazonSecurityLake*"
    ],
    "Condition": {
      "StringEquals": {
        "aws:ResourceAccount": "${aws:PrincipalAccount}"
      }
    }
  },
  {
    "Sid": "AllowGlueManage",
    "Effect": "Allow",
    "Action": [
      "glue:CreatePartition",
      "glue:BatchCreatePartition",
      "glue:GetTable",
      "glue:UpdateTable"
    ],
    "Resource": [
      "arn:aws:glue:*:*:table/amazon_security_lake_glue_db*/*",
      "arn:aws:glue:*:*:database/amazon_security_lake_glue_db*",
      "arn:aws:glue:*:*:catalog"
    ],
    "Condition": {
      "StringEquals": {
        "aws:ResourceAccount": "${aws:PrincipalAccount}"
      }
    }
  },
  {
    "Sid": "AllowToReadFromSqs",
    "Effect": "Allow",
    "Action": [
```

```

    "sqs:ReceiveMessage",
    "sqs>DeleteMessage",
    "sqs:GetQueueAttributes"
  ],
  "Resource": [
    "arn:aws:sqs:*:*:AmazonSecurityLake*"
  ],
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
  }
},
{
  "Sid": "AllowMetaDataReadWrite",
  "Effect": "Allow",
  "Action": [
    "s3:ListBucket",
    "s3:PutObject",
    "s3:GetObject"
  ],
  "Resource": [
    "arn:aws:s3:::aws-security-data-lake*"
  ],
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
  }
},
{
  "Sid": "AllowMetaDataCleanup",
  "Effect": "Allow",
  "Action": [
    "s3>DeleteObject"
  ],
  "Resource": [
    "arn:aws:s3:::aws-security-data-lake*/metadata/*.avro",
    "arn:aws:s3:::aws-security-data-lake*/metadata/*.metadata.json"
  ],
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
  }
}

```

```
    }
  }
]
}
```

AWS マネージドポリシー: AmazonSecurityLakePermissionsBoundary

Amazon Security Lake は、サードパーティのカスタムソースがデータレイクにデータを書き込むためのものと、サードパーティのカスタムサブスクライバーがデータレイクからデータを消費するための IAM ロールを作成し、その際にこのポリシーを使用して権限の境界を定義します。このポリシーを使用するために、お客様が実行する必要があるアクションはありません。データレイクがカスタマーマネージド AWS KMS キーで暗号化されている場合、`kms:Decrypt`および `アクセスkms:GenerateDataKey`許可が追加されます。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowActionsForSecurityLake",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion",
        "s3:ListBucket",
        "s3:ListBucketVersions",
        "s3:PutObject",
        "s3:GetBucketLocation",
        "kms:Decrypt",
        "kms:GenerateDataKey",
        "sqs:ReceiveMessage",
        "sqs:ChangeMessageVisibility",
        "sqs>DeleteMessage",
        "sqs:GetQueueUrl",
        "sqs:SendMessage",
        "sqs:GetQueueAttributes",
        "sqs:ListQueues"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyActionsForSecurityLake",
      "Effect": "Deny",
```

```
"NotAction": [
  "s3:GetObject",
  "s3:GetObjectVersion",
  "s3:ListBucket",
  "s3:ListBucketVersions",
  "s3:PutObject",
  "s3:GetBucketLocation",
  "kms:Decrypt",
  "kms:GenerateDataKey",
  "sqs:ReceiveMessage",
  "sqs:ChangeMessageVisibility",
  "sqs>DeleteMessage",
  "sqs:GetQueueUrl",
  "sqs:SendMessage",
  "sqs:GetQueueAttributes",
  "sqs:ListQueues"
],
"Resource": "*"
},
{
  "Sid": "DenyActionsNotOnSecurityLakeBucket",
  "Effect": "Deny",
  "Action": [
    "s3:GetObject",
    "s3:GetObjectVersion",
    "s3:ListBucket",
    "s3:ListBucketVersions",
    "s3:PutObject",
    "s3:GetBucketLocation"
  ],
  "NotResource": [
    "arn:aws:s3:::aws-security-data-lake*"
  ]
},
{
  "Sid": "DenyActionsNotOnSecurityLakeSQS",
  "Effect": "Deny",
  "Action": [
    "sqs:ReceiveMessage",
    "sqs:ChangeMessageVisibility",
    "sqs>DeleteMessage",
    "sqs:GetQueueUrl",
    "sqs:SendMessage",
    "sqs:GetQueueAttributes",
```

```
    "sqs:ListQueues"
  ],
  "NotResource": "arn:aws:sqs:*:*:AmazonSecurityLake*"
},
{
  "Sid": "DenyActionsNotOnSecurityLakeKMSS3SQS",
  "Effect": "Deny",
  "Action": [
    "kms:Decrypt",
    "kms:GenerateDataKey"
  ],
  "Resource": "*",
  "Condition": {
    "StringNotLike": {
      "kms:ViaService": [
        "s3.*.amazonaws.com",
        "sqs.*.amazonaws.com"
      ]
    }
  }
},
{
  "Sid": "DenyActionsNotOnSecurityLakeKMSForS3",
  "Effect": "Deny",
  "Action": [
    "kms:Decrypt",
    "kms:GenerateDataKey"
  ],
  "Resource": "*",
  "Condition": {
    "Null": {
      "kms:EncryptionContext:aws:s3:arn": "false"
    },
    "StringNotLikeIfExists": {
      "kms:EncryptionContext:aws:s3:arn": [
        "arn:aws:s3:::aws-security-data-lake*"
      ]
    }
  }
},
{
  "Sid": "DenyActionsNotOnSecurityLakeKMSForS3SQS",
  "Effect": "Deny",
  "Action": [
```

```
    "kms:Decrypt",
    "kms:GenerateDataKey"
  ],
  "Resource": "*",
  "Condition": {
    "Null": {
      "kms:EncryptionContext:aws:sqs:arn": "false"
    },
    "StringNotLikeIfExists": {
      "kms:EncryptionContext:aws:sqs:arn": [
        "arn:aws:sqs:*:*:AmazonSecurityLake*"
      ]
    }
  }
}
```

AWS マネージドポリシー: AmazonSecurityLakeAdministrator

アカウントで Amazon Security Lake を有効にする前に、プリンシパルに AmazonSecurityLakeAdministrator ポリシーをアタッチできます。このポリシーにより、プリンシパルが Security Lake のすべてのアクションに完全にアクセスすることが許可される、管理者許可が付与されます。その後、プリンシパルは Security Lake にオンボーディングし、Security Lake でソースとサブスクライバーを設定できます。

このポリシーには、Security Lake 管理者が Security Lake を通じて他の AWS サービスに対して実行できるアクションが含まれています。

このAmazonSecurityLakeAdministratorポリシーは、Amazon S3 クロスリージョンレプリケーションの管理、での新しいデータパーティションの登録 AWS Glue、カスタムソースに追加されたデータに対する Glue クローラの実行、または新しいデータの HTTPS エンドポイントサブスクライバーへの通知を行うために Security Lake が必要とするユーティリティロールの作成をサポートしていません。これらのロールは、「[Amazon Security Lake の開始方法](#)」で説明されているように事前に作成できます。

AmazonSecurityLakeAdministrator 管理ポリシーに加えて、Security Lake にはオンボーディングと設定機能のための lakeformation:PutDataLakeSettings 権限が必要です。PutDataLakeSettings は、アカウント内のすべてのリージョンの Lake Formation リソースの管理者として IAM プリンシパルを設定できます。このロールには iam:CreateRole

permission と AmazonSecurityLakeAdministrator のポリシーが添付されている必要があります。

Lake Formation 管理者は Lake Formation コンソールへのフルアクセス権を持ち、初期データ設定とアクセス権限を制御できます。Security Lake は、Security Lake を有効にするプリンシパルと AmazonSecurityLakeMetaStoreManager ロール (またはその他の指定されたロール) を Lake Formation 管理者として割り当てます。これにより、管理者はテーブルの作成、テーブルスキーマの更新、新しいパーティションの登録、テーブルに対する権限の設定を行うことができます。Security Lake 管理者ユーザーまたはロールのポリシーには、次のアクセス許可を含める必要があります。

Note

Lake Formation ベースのサブスクライバーアクセスを許可するのに十分なアクセス許可を提供するために、Security Lake では次のアクセス `glue:PutResourcePolicy` 許可を追加することをお勧めします。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowPutLakeFormationSettings",
      "Effect": "Allow",
      "Action": "lakeformation:PutDatalakeSettings",
      "Resource": "*",
      "Condition": {
        "ForAnyValue:StringEquals": {
          "aws:CalledVia": "securitylake.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowGlueActions",
      "Effect": "Allow",
      "Action": ["glue:PutResourcePolicy", "glue>DeleteResourcePolicy"],
      "Resource": [
        "arn:aws:glue:*:*:catalog",
        "arn:aws:glue:*:*:database/amazon_security_lake_glue_db*",
        "arn:aws:glue:*:*:table/amazon_security_lake_glue_db*/*"
      ],
      "Condition": {
```

```
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
}
]
```

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `securitylake` - すべての Security Lake アクションへの完全なアクセスをプリンシパルに許可します。
- `organizations` - プリンシパルが AWS Organizations から組織内のアカウントに関する情報を取得できるようにします。アカウントが組織に属している場合、これらの許可は、Security Lake コンソールがアカウント名とアカウント番号を表示することを許可します。
- `iam` - プリンシパルが Security Lake、AWS Lake Formation のサービスにリンクされたロールを Amazon EventBridge、これらのサービスを有効にするために必要なステップとして作成できるようにします。また、サブスクライバーロールとカスタムソースロールのポリシーを作成および編集できます。これらのロールの権限は、AmazonSecurityLakePermissionsBoundary ポリシーで許可されているものに限定されます。
- `ram` - プリンシパルが Security Lake ソースへのサブスクライバーによる Lake Formation ベースのクエリアクセスを設定できるようにします。
- `s3` - プリンシパルが Security Lake バケットを作成および管理し、それらのバケットの内容を読み取ることを許可します。
- `lambda` - プリンシパルが、AWS ソース配信とクロスリージョンレプリケーション後にテーブルパーティションを更新する Lambda AWS Glue ために使用される を管理できるようにします。
- `glue` - プリンシパルが Security Lake のデータベースとテーブルを作成および管理できるようにします。
- `lakeformation` - プリンシパルが Security Lake テーブルの Lake Formation アクセス許可を管理できるようにします。
- `events` - プリンシパルが Security Lake ソース内の新しいデータをサブスクライバーに通知するためのルールを管理できるようにします。

- sqs — プリンシパルが Security Lake ソースの新しいデータをサブスクライバーに通知するために使用する Amazon SQS キューを作成および管理できるようにします。
- kms — プリンシパルが Security Lake に顧客管理キーを使用してデータを書き込むためのアクセス権を付与できるようにします。
- secretsmanager — プリンシパルが HTTPS エンドポイント経由で、Security Lake ソース内の新しいデータをサブスクライバーに通知するために使用されるシークレットを管理できるようにします。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowActionsWithAnyResource",
      "Effect": "Allow",
      "Action": [
        "securitylake:*",
        "organizations:DescribeOrganization",
        "organizations:ListDelegatedServicesForAccount",
        "organizations:ListAccounts",
        "iam:ListRoles",
        "ram:GetResourceShareAssociations"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowActionsWithAnyResourceViaSecurityLake",
      "Effect": "Allow",
      "Action": [
        "glue:CreateCrawler",
        "glue:StopCrawlerSchedule",
        "lambda:CreateEventSourceMapping",
        "lakeformation:GrantPermissions",
        "lakeformation:ListPermissions",
        "lakeformation:RegisterResource",
        "lakeformation:RevokePermissions",
        "lakeformation:GetDatalakeSettings",
        "events:ListConnections",
        "events:ListApiDestinations",
        "iam:GetRole",
        "iam:ListAttachedRolePolicies",

```

```
    "kms:DescribeKey"
  ],
  "Resource": "*",
  "Condition": {
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowManagingSecurityLakeS3Buckets",
  "Effect": "Allow",
  "Action": [
    "s3:CreateBucket",
    "s3:PutBucketPolicy",
    "s3:PutBucketPublicAccessBlock",
    "s3:PutBucketNotification",
    "s3:PutBucketTagging",
    "s3:PutEncryptionConfiguration",
    "s3:PutBucketVersioning",
    "s3:PutReplicationConfiguration",
    "s3:PutLifecycleConfiguration",
    "s3:ListBucket",
    "s3:PutObject",
    "s3:GetBucketNotification"
  ],
  "Resource": "arn:aws:s3:::aws-security-data-lake*",
  "Condition": {
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowLambdaCreateFunction",
  "Effect": "Allow",
  "Action": [
    "lambda:CreateFunction"
  ],
  "Resource": [
    "arn:aws:lambda:*:*:function:SecurityLake_Glue_Partition_Updater_Lambda*",
    "arn:aws:lambda:*:*:function:AmazonSecurityLake*"
  ],
  "Condition": {
```

```
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  },
  {
    "Sid": "AllowLambdaAddPermission",
    "Effect": "Allow",
    "Action": [
      "lambda:AddPermission"
    ],
    "Resource": [
      "arn:aws:lambda:*:*:function:SecurityLake_Glue_Partition_Updater_Lambda*",
      "arn:aws:lambda:*:*:function:AmazonSecurityLake*"
    ],
    "Condition": {
      "ForAnyValue:StringEquals": {
        "aws:CalledVia": "securitylake.amazonaws.com"
      },
      "StringEquals": {
        "lambda:Principal": "securitylake.amazonaws.com"
      }
    }
  },
  {
    "Sid": "AllowGlueActions",
    "Effect": "Allow",
    "Action": [
      "glue:CreateDatabase",
      "glue:GetDatabase",
      "glue:CreateTable",
      "glue:GetTable"
    ],
    "Resource": [
      "arn:aws:glue:*:*:catalog",
      "arn:aws:glue:*:*:database/amazon_security_lake_glue_db*",
      "arn:aws:glue:*:*:table/amazon_security_lake_glue_db*/*"
    ],
    "Condition": {
      "ForAnyValue:StringEquals": {
        "aws:CalledVia": "securitylake.amazonaws.com"
      }
    }
  },
},
```

```
{
  "Sid": "AllowEventBridgeActions",
  "Effect": "Allow",
  "Action": [
    "events:PutTargets",
    "events:PutRule",
    "events:DescribeRule",
    "events:CreateApiDestination",
    "events:CreateConnection",
    "events:UpdateConnection",
    "events:UpdateApiDestination",
    "events>DeleteConnection",
    "events>DeleteApiDestination",
    "events:ListTargetsByRule",
    "events:RemoveTargets",
    "events>DeleteRule"
  ],
  "Resource": [
    "arn:aws:events:*:*:rule/AmazonSecurityLake*",
    "arn:aws:events:*:*:rule/SecurityLake*",
    "arn:aws:events:*:*:api-destination/AmazonSecurityLake*",
    "arn:aws:events:*:*:connection/AmazonSecurityLake*"
  ],
  "Condition": {
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowSQSActions",
  "Effect": "Allow",
  "Action": [
    "sqs:CreateQueue",
    "sqs:SetQueueAttributes",
    "sqs:GetQueueURL",
    "sqs:AddPermission",
    "sqs:GetQueueAttributes",
    "sqs>DeleteQueue"
  ],
  "Resource": [
    "arn:aws:sqs:*:*:SecurityLake*",
    "arn:aws:sqs:*:*:AmazonSecurityLake*"
  ],
}
```

```
"Condition": {
  "ForAnyValue:StringEquals": {
    "aws:CalledVia": "securitylake.amazonaws.com"
  }
},
{
  "Sid": "AllowKmsCmkGrantForSecurityLake",
  "Effect": "Allow",
  "Action": "kms:CreateGrant",
  "Resource": "arn:aws:kms:*:*:key/*",
  "Condition": {
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    },
    "StringLike": {
      "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::aws-security-data-lake*"
    },
    "ForAllValues:StringEquals": {
      "kms:GrantOperations": [
        "GenerateDataKey",
        "RetireGrant",
        "Decrypt"
      ]
    }
  }
},
{
  "Sid": "AllowEnablingQueryBasedSubscribers",
  "Effect": "Allow",
  "Action": [
    "ram:CreateResourceShare",
    "ram:AssociateResourceShare"
  ],
  "Resource": "*",
  "Condition": {
    "StringLikeIfExists": {
      "ram:ResourceArn": [
        "arn:aws:glue:*:*:catalog",
        "arn:aws:glue:*:*:database/amazon_security_lake_glue_db*",
        "arn:aws:glue:*:*:table/amazon_security_lake_glue_db*/*"
      ]
    },
    "ForAnyValue:StringEquals": {
```

```

        "aws:CalledVia": "securitylake.amazonaws.com"
    }
}
},
{
    "Sid": "AllowConfiguringQueryBasedSubscribers",
    "Effect": "Allow",
    "Action": [
        "ram:UpdateResourceShare",
        "ram:GetResourceShares",
        "ram:DisassociateResourceShare",
        "ram>DeleteResourceShare"
    ],
    "Resource": "*",
    "Condition": {
        "StringLike": {
            "ram:ResourceShareName": "LakeFormation*"
        },
        "ForAnyValue:StringEquals": {
            "aws:CalledVia": "securitylake.amazonaws.com"
        }
    }
}
},
{
    "Sid": "AllowConfiguringCredentialsForSubscriberNotification",
    "Effect": "Allow",
    "Action": [
        "secretsmanager:CreateSecret",
        "secretsmanager:GetSecretValue",
        "secretsmanager:PutSecretValue"
    ],
    "Resource": "arn:aws:secretsmanager:*:*:secret:events!connection/
AmazonSecurityLake-*",
    "Condition": {
        "ForAnyValue:StringEquals": {
            "aws:CalledVia": "securitylake.amazonaws.com"
        }
    }
}
},
{
    "Sid": "AllowPassRoleForUpdatingGluePartitionsSecLakeArn",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": [

```

```

    "arn:aws:iam::*:role/service-role/AmazonSecurityLakeMetaStoreManager",
    "arn:aws:iam::*:role/service-role/AmazonSecurityLakeMetaStoreManagerV2"
  ],
  "Condition": {
    "StringEquals": {
      "iam:PassedToService": "lambda.amazonaws.com"
    },
    "StringLike": {
      "iam:AssociatedResourceARN": "arn:aws:securitylake:*:*:data-lake/default"
    }
  }
},
{
  "Sid": "AllowPassRoleForUpdatingGluePartitionsLambdaArn",
  "Effect": "Allow",
  "Action": "iam:PassRole",
  "Resource": [
    "arn:aws:iam::*:role/service-role/AmazonSecurityLakeMetaStoreManager",
    "arn:aws:iam::*:role/service-role/AmazonSecurityLakeMetaStoreManagerV2"
  ],
  "Condition": {
    "StringEquals": {
      "iam:PassedToService": "lambda.amazonaws.com"
    },
    "StringLike": {
      "iam:AssociatedResourceARN": [
        "arn:aws:lambda:*:*:function:SecurityLake_Glue_Partition_Updater_Lambda*",
        "arn:aws:lambda:*:*:function:AmazonSecurityLake*"
      ]
    }
  },
  "ForAnyValue:StringEquals": {
    "aws:CalledVia": "securitylake.amazonaws.com"
  }
},
{
  "Sid": "AllowPassRoleForCrossRegionReplicationSecLakeArn",
  "Effect": "Allow",
  "Action": "iam:PassRole",
  "Resource": "arn:aws:iam::*:role/service-role/AmazonSecurityLakeS3ReplicationRole",
  "Condition": {
    "StringEquals": {
      "iam:PassedToService": "s3.amazonaws.com"
    }
  },

```

```

    "StringLike": {
      "iam:AssociatedResourceARN": "arn:aws:securitylake:*:*:data-lake/default"
    }
  },
  {
    "Sid": "AllowPassRoleForCrossRegionReplicationS3Arn",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": "arn:aws:iam:*:*:role/service-role/AmazonSecurityLakeS3ReplicationRole",
    "Condition": {
      "StringEquals": {
        "iam:PassedToService": "s3.amazonaws.com"
      },
      "StringLike": {
        "iam:AssociatedResourceARN": "arn:aws:s3:::aws-security-data-lake*"
      },
      "ForAnyValue:StringEquals": {
        "aws:CalledVia": "securitylake.amazonaws.com"
      }
    }
  },
  {
    "Sid": "AllowPassRoleForCustomSourceCrawlerSecLakeArn",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": "arn:aws:iam:*:*:role/service-role/AmazonSecurityLakeCustomDataGlueCrawler*",
    "Condition": {
      "StringEquals": {
        "iam:PassedToService": "glue.amazonaws.com"
      },
      "StringLike": {
        "iam:AssociatedResourceARN": "arn:aws:securitylake:*:*:data-lake/default"
      }
    }
  },
  {
    "Sid": "AllowPassRoleForCustomSourceCrawlerGlueArn",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": "arn:aws:iam:*:*:role/service-role/AmazonSecurityLakeCustomDataGlueCrawler*",
    "Condition": {

```

```

    "StringEquals": {
      "iam:PassedToService": "glue.amazonaws.com"
    },
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowPassRoleForSubscriberNotificationSecLakeArn",
  "Effect": "Allow",
  "Action": "iam:PassRole",
  "Resource": "arn:aws:iam::*:role/service-role/
AmazonSecurityLakeSubscriberEventBridge",
  "Condition": {
    "StringEquals": {
      "iam:PassedToService": "events.amazonaws.com"
    },
    "StringLike": {
      "iam:AssociatedResourceARN": "arn:aws:securitylake:*:*:subscriber/*"
    }
  }
},
{
  "Sid": "AllowPassRoleForSubscriberNotificationEventsArn",
  "Effect": "Allow",
  "Action": "iam:PassRole",
  "Resource": "arn:aws:iam::*:role/service-role/
AmazonSecurityLakeSubscriberEventBridge",
  "Condition": {
    "StringEquals": {
      "iam:PassedToService": "events.amazonaws.com"
    },
    "StringLike": {
      "iam:AssociatedResourceARN": "arn:aws:events:*:*:rule/AmazonSecurityLake*"
    },
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowOnboardingToSecurityLakeDependencies",
  "Effect": "Allow",

```

```

    "Action": "iam:CreateServiceLinkedRole",
    "Resource": [
      "arn:aws:iam::*:role/aws-service-role/securitylake.amazonaws.com/
AWSServiceRoleForSecurityLake",
      "arn:aws:iam::*:role/aws-service-role/lakeformation.amazonaws.com/
AWSServiceRoleForLakeFormationDataAccess",
      "arn:aws:iam::*:role/aws-service-role/apidestinations.events.amazonaws.com/
AWSServiceRoleForAmazonEventBridgeApiDestinations"
    ],
    "Condition": {
      "StringLike": {
        "iam:AWSServiceName": [
          "securitylake.amazonaws.com",
          "lakeformation.amazonaws.com",
          "apidestinations.events.amazonaws.com"
        ]
      }
    }
  },
  {
    "Sid": "AllowRolePolicyActionsforSubscribersandSources",
    "Effect": "Allow",
    "Action": [
      "iam:CreateRole",
      "iam:PutRolePolicy",
      "iam>DeleteRolePolicy"
    ],
    "Resource": "arn:aws:iam::*:role/AmazonSecurityLake*",
    "Condition": {
      "StringEquals": {
        "iam:PermissionsBoundary": "arn:aws:iam::aws:policy/
AmazonSecurityLakePermissionsBoundary"
      },
      "ForAnyValue:StringEquals": {
        "aws:CalledVia": "securitylake.amazonaws.com"
      }
    }
  },
  {
    "Sid": "AllowRegisterS3LocationInLakeFormation",
    "Effect": "Allow",
    "Action": [
      "iam:PutRolePolicy",
      "iam:GetRolePolicy"
    ]
  }
}

```

```
    ],
    "Resource": "arn:aws:iam::*:role/aws-service-role/lakeformation.amazonaws.com/
AWSServiceRoleForLakeFormationDataAccess",
    "Condition": {
      "ForAnyValue:StringEquals": {
        "aws:CalledVia": "securitylake.amazonaws.com"
      }
    }
  },
  {
    "Sid": "AllowIAMActionsByResource",
    "Effect": "Allow",
    "Action": [
      "iam:ListRolePolicies",
      "iam:DeleteRole"
    ],
    "Resource": "arn:aws:iam::*:role/AmazonSecurityLake*",
    "Condition": {
      "ForAnyValue:StringEquals": {
        "aws:CalledVia": "securitylake.amazonaws.com"
      }
    }
  },
  {
    "Sid": "S3ReadAccessToSecurityLakes",
    "Effect": "Allow",
    "Action": [
      "s3:Get*",
      "s3:List*"
    ],
    "Resource": "arn:aws:s3:::aws-security-data-lake-*"
  },
  {
    "Sid": "S3ReadAccessToSecurityLakeMetastoreObject",
    "Effect": "Allow",
    "Action": [
      "s3:GetObject",
      "s3:GetObjectVersion"
    ],
    "Resource": "arn:aws:s3:::security-lake-meta-store-manager-*"
  },
  {
    "Sid": "S3ResourcelessReadOnly",
    "Effect": "Allow",
```

```
"Action": [  
  "s3:GetAccountPublicAccessBlock",  
  "s3:ListAccessPoints",  
  "s3:ListAllMyBuckets"  
],  
"Resource": "*" ]  
}
```

AWS マネージドポリシー: SecurityLakeServiceLinkedRole

Security Lake は、 という名前のサービスにリンクされたロール `AWSServiceRoleForSecurityLake` を使用して、セキュリティデータレイクを作成および運用します。

`SecurityLakeServiceLinkedRole` マネージド ポリシーを IAM エンティティに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 Security Lake 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添付されます。詳細については、[「Security Lake のサービスにリンクされたロールのアクセス許可」](#)を参照してください。

AWS マネージドポリシー: SecurityLakeResourceManagementServiceRolePolicy

Security Lake は、 という名前のサービスにリンクされたロール `AWSServiceRoleForSecurityLakeResourceManagement` を使用して、継続的なモニタリングとパフォーマンスの向上を実行します。これにより、レイテンシーとコストを削減できます。

`SecurityLakeResourceManagementServiceRolePolicy` マネージド ポリシーを IAM エンティティに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 Security Lake 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添付されます。詳細については、[「リソース管理のためのサービスにリンクされたロールのアクセス許可」](#)を参照してください。

AWS マネージドポリシー: AWS GlueServiceRole

`AWS GlueServiceRole` 管理ポリシーは AWS Glue クローラーを呼び出し、カスタムソースデータをクローリングしてパーティションメタデータを識別 AWS Glue することをに許可します。このメタデータはデータカタログでテーブルを作成および更新するために必要です。

詳細については、[「Security Lake のカスタムソースからデータを収集する」](#)を参照してください。

AWS マネージドポリシーに対する Security Lake の更新

Security Lake 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については、Security Lake の [Document history] (ドキュメントの履歴) ページの RSS フィードをサブスクライブしてください。

変更	説明	日付
Amazon Security Lake のサービスにリンクされたロール – 新しいサービスにリンクされたロール	新しいサービスにリンクされたロールを追加しました。AWSServiceRoleForSecurityLakeResourceManagement。このサービスにリンクされたロールは、継続的なモニタリングとパフォーマンスの向上を実行するアクセス許可を Security Lake に提供し、レイテンシーとコストを削減できます。	2024 年 11 月 14 日
Amazon Security Lake のサービスにリンクされたロール – 既存のサービスにリンクされたロールのアクセス許可の更新	SecurityLakeServiceLinkedRole ポリシーの AWS 管理ポリシーに AWS WAF アクションを追加しました。追加のアクションにより、Security Lake で AWS WAF ログソースとして有効になっている場合、Security Lake はログを収集できます。	2024 年 5 月 22 日
AmazonSecurityLakePermissionsBoundary – 既存ポリシーへの更新	Security Lake がポリシーに SID アクションを追加しました。	2024 年 5 月 13 日

変更	説明	日付
AmazonSecurityLake MetastoreManager – 既存ポリシーへの更新	Security Lake は、データレイク内のメタデータを削除できるメタデータクリーンアップアクションを追加するようにポリシーを更新しました。	2024 年 3 月 27 日
AmazonSecurityLakeAdministrator – 既存ポリシーへの更新	Security Lake は、新しいAmazonSecurityLake MetastoreManagerV2 ロールiam:PassRole でを許可するようにポリシーを更新し、Security Lake がデータレイクコンポーネントをデプロイまたは更新できるようにしました。	2024 年 2 月 23 日
AmazonSecurityLake MetastoreManager - 新しいポリシー	Security Lake は、Security Lake がデータレイク内のメタデータを管理するためのアクセス許可を付与する新しいマネージドポリシーを追加しました。	2024 年 1 月 23 日
AmazonSecurityLakeAdministrator - 新しいポリシー	Security Lake は、すべての Security Lake アクションへのフルアクセスをプリンシパルに付与する新しいマネージドポリシーを追加しました。	2023 年 5 月 30 日
Security Lake が変更の追跡を開始	Security Lake 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 29 日

Security Lake のサービスにリンクされたロールの使用

Security Lake 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、Security Lake に直接リンクされた IAM ロールです。これは Security Lake によって事前定義されており、Security Lake がユーザーに代わって他の AWS のサービス を呼び出し、セキュリティ データ レイク サービスを操作するために必要なすべてのアクセス許可が含まれています。Security Lake は、Security Lake AWS リージョン が利用可能なすべてのこのサービスにリンクされたロールを使用します。

サービスリンクロールを使用することで、Security Lake の設定時に必要な許可を手動で追加する必要がなくなります。Security Lake は、サービスリンクロールの許可を定義します。その許可が特別に定義されていない限り、Security Lake のみがそのロールを引き受けます。定義された許可には信頼ポリシーと許可ポリシーが含まれ、その許可ポリシーを他の IAM エンティティにアタッチすることはできません。

サービスリンクロールの作成、編集、削除をIAM エンティティ (ユーザー、グループ、ロールなど) に許可するには、許可を設定する必要があります。詳細については、「IAM User Guide」(IAM ユーザーガイド) の [「Service-linked role permissions」](#) (サービスリンクロールのアクセス権限) を参照してください。サービスリンクロールを削除するには、その関連リソースを削除します。これにより、リソースへの意図しないアクセスによる権限の削除が防止され、リソースは保護されます。

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、[AWS 「IAM と連携するサービス」](#) を参照し、「サービスにリンクされたロール」列で「はい」があるサービスを探します。サービスリンク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。

トピック

- [Security Lake のサービスにリンクされたロール \(SLR\) アクセス許可](#)
- [リソース管理のためのサービスリンクロール \(SLR\) アクセス許可](#)

Security Lake のサービスにリンクされたロール (SLR) アクセス許可

Security Lake では、AWSServiceRoleForSecurityLake という名前のサービスリンクロールを使用します。このサービスリンクロールは、ロールを引き受ける上で securitylake.amazonaws.com サービスを信頼します。Amazon Security Lake AWS の管理ポリシーの詳細については、[AWS 「Amazon Security Lake の管理ポリシー」](#) を参照してください。

という名前の AWS マネージドポリシーであるロールのアクセス許可ポリ

シーSecurityLakeServiceLinkedRoleにより、Security Lake はセキュリティデータレイクを作成して運用できます。また、Security Lake は指定されたリソースに対して次のようなタスクを実行できるようになります。

- AWS Organizations アクションを使用して、関連付けられたアカウントに関する情報を取得する
- Amazon Elastic Compute Cloud (Amazon EC2) を使用して、Amazon VPC フローログに関する情報を取得します
- AWS CloudTrail アクションを使用して、サービスにリンクされたロールに関する情報を取得する
- Security Lake で AWS WAF ログソースとして有効になっている場合、AWS WAF アクションを使用してログを収集する
- LogDelivery アクションを使用して、AWS WAF ログ配信サブスクリプションを作成または削除します。

そのロールは、次のアクセス許可ポリシーで設定する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "OrganizationsPolicies",
    "Effect": "Allow",
    "Action": [
      "organizations:ListAccounts",
      "organizations:DescribeOrganization"
    ],
    "Resource": [
      "*"
    ]
  },
  {
    "Sid": "DescribeOrgAccounts",
    "Effect": "Allow",
    "Action": [
      "organizations:DescribeAccount"
    ],
    "Resource": [
      "arn:aws:organizations::*:account/o-*/*"
    ]
  },
  {
```

```

        "Sid": "AllowManagementOfServiceLinkedChannel",
        "Effect": "Allow",
        "Action": [
            "cloudtrail:CreateServiceLinkedChannel",
            "cloudtrail>DeleteServiceLinkedChannel",
            "cloudtrail:GetServiceLinkedChannel",
            "cloudtrail:UpdateServiceLinkedChannel"
        ],
        "Resource": "arn:aws:cloudtrail:*:*:channel/aws-service-channel/security-
lake/*"
    },
    {
        "Sid": "AllowListServiceLinkedChannel",
        "Effect": "Allow",
        "Action": [
            "cloudtrail:ListServiceLinkedChannels"
        ],
        "Resource": "*"
    },
    {
        "Sid": "DescribeAnyVpc",
        "Effect": "Allow",
        "Action": [
            "ec2:DescribeVpcs"
        ],
        "Resource": "*"
    },
    {
        "Sid": "ListDelegatedAdmins",
        "Effect": "Allow",
        "Action": [
            "organizations:ListDelegatedAdministrators"
        ],
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "organizations:ServicePrincipal": "securitylake.amazonaws.com"
            }
        }
    },
    {
        "Sid": "AllowWafLoggingConfiguration",
        "Effect": "Allow",
        "Action": [

```

```

        "wafv2:PutLoggingConfiguration",
        "wafv2:GetLoggingConfiguration",
        "wafv2:ListLoggingConfigurations",
        "wafv2>DeleteLoggingConfiguration"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "wafv2:LogScope": "SecurityLake"
        }
    }
},
{
    "Sid": "AllowPutLoggingConfiguration",
    "Effect": "Allow",
    "Action": [
        "wafv2:PutLoggingConfiguration"
    ],
    "Resource": "*",
    "Condition": {
        "ArnLike": {
            "wafv2:LogDestinationResource": "arn:aws:s3:::aws-waf-logs-
security-lake-*"
        }
    }
},
{
    "Sid": "ListWebACLs",
    "Effect": "Allow",
    "Action": [
        "wafv2:ListWebACLs"
    ],
    "Resource": "*"
},
{
    "Sid": "LogDelivery",
    "Effect": "Allow",
    "Action": [
        "logs:CreateLogDelivery",
        "logs>DeleteLogDelivery"
    ],
    "Resource": "*",
    "Condition": {
        "ForAnyValue:StringEquals": {

```

```
        "aws:CalledVia": [
            "wafv2.amazonaws.com"
        ]
    }
}
]
}
```

サービスリンクロールの作成、編集、削除をIAM エンティティ (ユーザー、グループ、ロールなど) に許可するには、許可を設定する必要があります。詳細については、「IAM User Guide」(IAM ユーザーガイド) の [「Service-linked role permissions」](#) (サービスリンクロールのアクセス権限) を参照してください。

Security Lake のサービスリンクロールの作成

Security Lake 用の `AWSServiceRoleForSecurityLake` 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 Security Lake を有効にすると AWS アカウント、Security Lake によってサービスにリンクされたロールが自動的に作成されます。

Security Lake 向けのサービスリンクロールの編集

Security Lake では、`AWSServiceRoleForSecurityLake` サービスリンクロールの編集は許可されていません。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、IAM を使用してロールの説明を編集することはできます。詳細については、「[IAM ユーザーガイド](#)」の「サービスリンクロールの編集」を参照してください。

Security Lake 向けのサービスリンクロールの削除

サービスリンクロールを Security Lake から削除することはできません。代わりに、IAM コンソール、API、または からサービスにリンクされたロールを削除できます AWS CLI。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削除」](#) を参照してください。

サービスリンクロールを削除する前に、まずロールにアクティブなセッションがないことを確認し、`AWSServiceRoleForSecurityLake` が使用しているリソースを削除する必要があります。

Note

リソースを削除しようとするときに Security Lake が `AWSServiceRoleForSecurityLake` ロールを使用している場合、削除は失敗する可能性があります。失敗した場合は、数分待ってから操作を再試行してください。

`AWSServiceRoleForSecurityLake` サービスリンクロールを削除したが、再作成する必要がある場合は、アカウントで Security Lake を有効にすることで、ロールを再作成することができます。Security Lake を再作成すると、Security Lake は、ユーザーのためにサービスリンクロールを再作成します。

Security Lake サービスにリンクされたロール AWS リージョン でサポート

Security Lake は、Security Lake AWS リージョン が利用可能なすべての `AWSServiceRoleForSecurityLake` サービスにリンクされたロールの使用をサポートしています。Security Lake が現在利用可能なリージョンのリストについては、「[Security Lake のリージョンとエンドポイント](#)」を参照してください。

リソース管理のためのサービスリンクロール (SLR) アクセス許可

Security Lake は、という名前のサービスにリンクされたロール `AWSServiceRoleForSecurityLakeResourceManagement` を使用して継続的なモニタリングとパフォーマンスの向上を実行し、レイテンシーとコストを削減できます。このサービスリンクロールは、ロールを引き受ける上で `resource-management.securitylake.amazonaws.com` サービスを信頼します。を有効にする `AWSServiceRoleForSecurityLakeResourceManagement`、Lake Formation へのアクセス権も付与され、セキュリティを強化するために、すべてのリージョンで Security Lake マネージド S3 バケットを Lake Formation に自動的に登録します。

という名前の AWS マネージドポリシーであるロールのアクセス許可ポリシーは `SecurityLakeResourceManagementServiceRolePolicy`、データレイク内のメタデータの管理など、Security Lake によって作成されたリソースを管理するアクセスを許可します。Amazon Security Lake AWS の管理ポリシーの詳細については、[AWS 「Amazon Security Lake の管理ポリシー」](#)を参照してください。

このサービスにリンクされたロールにより、Security Lake は、Security Lake によってアカウントにデプロイされたリソース (S3 バケット、AWS Glue テーブル、Amazon SQS キュー、Metastore

Manager (MSM) Lambda 関数、EventBridge ルール) の状態をモニタリングできます。Security Lake がこのサービスにリンクされたロールで実行できるオペレーションの例を次に示します。

- Apache Iceberg マニフェストファイルの圧縮により、クエリのパフォーマンスが向上し、Lambda MSM の処理時間とコストを削減できます。
- Amazon SQS の状態をモニタリングして、取り込みの問題を検出します。
- メタデータファイルを除外するために、クロスリージョンデータレプリケーションを最適化します。

Note

`AWSServiceRoleForSecurityLakeResourceManagement` サービスにリンクされたロールをインストールしない場合、Security Lake は引き続き機能しますが、Security Lake がアカウントのリソースをモニタリングおよび最適化できるように、このサービスにリンクされたロールを受け入れることを強くお勧めします。

アクセス許可の詳細

そのロールは、次のアクセス許可ポリシーで設定する必要があります。

- `events` – プリンシパルがログソースとログサブスクライバーに必要な EventBridge ルールを管理できるようにします。
- `lambda` – AWS プリンシパルがソース配信とクロスリージョンレプリケーションの後に AWS Glue テーブルパーティションを更新するために使用される Lambda を管理できるようにします。
- `glue` – プリンシパルが AWS Glue Data Catalog テーブルに対して特定の書き込みアクションを実行できるようにします。これにより AWS Glue、クローラはデータ内のパーティションを識別でき、Security Lake は Apache Iceberg テーブルの Apache Iceberg メタデータを管理できます。
- `s3` – プリンシパルがログデータと Glue テーブルメタデータを含む Security Lake バケットに対して特定の読み取りおよび書き込みアクションを実行できるようにします。
- `logs` – プリンシパルが Lambda 関数の出力を CloudWatch Logs にログ記録するための読み取りアクセスを許可します。
- `sqs` – プリンシパルが、データレイクでオブジェクトが追加または更新されたときにイベント通知を受け取る Amazon SQS キューに対して特定の読み取りおよび書き込みアクションを実行できるようにします。

- lakeformation – プリンシパルが Lake Formation 設定を読み取って、設定ミスモニタリングできるようにします。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ReadEventBridgeRules",
      "Effect": "Allow",
      "Action": [
        "events:ListRules"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "${aws:PrincipalAccount}"
        }
      }
    },
    {
      "Sid": "ManageSecurityLakeEventRules",
      "Effect": "Allow",
      "Action": [
        "events:PutRule"
      ],
      "Resource": "arn:aws:events:*:*:rule/AmazonSecurityLake-*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "${aws:PrincipalAccount}"
        }
      }
    },
    {
      "Sid": "ManageSecurityLakeLambdaConfigurations",
      "Effect": "Allow",
      "Action": [
        "lambda:GetEventSourceMapping",
        "lambda:GetFunction",
        "lambda:PutFunctionConcurrency",
        "lambda:GetProvisionedConcurrencyConfig",
        "lambda:GetFunctionConcurrency",
        "lambda:GetRuntimeManagementConfig",

```

```
    "lambda:PutProvisionedConcurrencyConfig",
    "lambda:PublishVersion",
    "lambda>DeleteFunctionConcurrency",
    "lambda>DeleteEventSourceMapping",
    "lambda:GetAlias",
    "lambda:GetPolicy",
    "lambda:GetFunctionConfiguration",
    "lambda:UpdateFunctionConfiguration"
  ],
  "Resource": [
    "arn:aws:lambda:*:*:function:SecurityLake_Glue_Partition_Updater_Lambda*",
    "arn:aws:lambda:*:*:function:AmazonSecurityLakeMetastoreManager-*-*"
  ],
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
  }
},
{
  "Sid": "AllowListLambdaEventSourceMappings",
  "Effect": "Allow",
  "Action": [
    "lambda:ListEventSourceMappings"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
  }
},
{
  "Sid": "AllowUpdateLambdaEventSourceMapping",
  "Effect": "Allow",
  "Action": [
    "lambda:UpdateEventSourceMapping"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
    },
    "StringLike": {
```

```
        "lambda:FunctionArn":
"arn:aws:lambda:*:*:function:AmazonSecurityLakeMetastoreManager-*-*"
    }
}
},
{
  "Sid": "AllowUpdateLambdaConfigs",
  "Effect": "Allow",
  "Action": [
    "lambda:UpdateFunctionConfiguration"
  ],
  "Resource": "arn:aws:lambda:*:*:function:AmazonSecurityLakeMetastoreManager-*-*",
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
  }
},
{
  "Sid": "ManageSecurityLakeGlueResources",
  "Effect": "Allow",
  "Action": [
    "glue:CreatePartition",
    "glue:BatchCreatePartition",
    "glue:GetTable",
    "glue:GetTables",
    "glue:UpdateTable",
    "glue:GetDatabase"
  ],
  "Resource": [
    "arn:aws:glue:*:*:table/amazon_security_lake_glue_db*/*",
    "arn:aws:glue:*:*:database/amazon_security_lake_glue_db*",
    "arn:aws:glue:*:*:catalog"
  ],
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
  }
},
{
  "Sid": "AllowDataLakeConfigurationManagement",
  "Effect": "Allow",
  "Action": [
```

```

    "s3:ListBucket",
    "s3:PutObject",
    "s3:GetObjectAttributes",
    "s3:GetBucketNotification",
    "s3:PutBucketNotification",
    "s3:GetLifecycleConfiguration",
    "s3:PutLifecycleConfiguration",
    "s3:GetEncryptionConfiguration",
    "s3:GetReplicationConfiguration"
  ],
  "Resource": [
    "arn:aws:s3:::aws-security-data-lake*"
  ],
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
  }
},
{
  "Sid": "AllowMetaDataCompactionAndManagement",
  "Effect": "Allow",
  "Action": [
    "s3:GetObject",
    "s3:DeleteObject",
    "s3:RestoreObject"
  ],
  "Resource": [
    "arn:aws:s3:::aws-security-data-lake*/metadata/*.avro",
    "arn:aws:s3:::aws-security-data-lake*/metadata/*.metadata.json"
  ],
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
  }
},
{
  "Sid": "ReadSecurityLakeLambdaLogs",
  "Effect": "Allow",
  "Action": [
    "logs:DescribeLogStreams",
    "logs:StartQuery",
    "logs:GetLogEvents",

```

```

    "logs:GetQueryResults",
    "logs:GetLogRecord"
  ],
  "Resource": [
    "arn:aws:logs:*:*:log-group:/aws/lambda/AmazonSecurityLakeMetastoreManager-*-*"
  ],
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
  }
},
{
  "Sid": "ManageSecurityLakeSQSQueue",
  "Effect": "Allow",
  "Action": [
    "sqs:StartMessageMoveTask",
    "sqs>DeleteMessage",
    "sqs:GetQueueUrl",
    "sqs:ListDeadLetterSourceQueues",
    "sqs:ChangeMessageVisibility",
    "sqs:ListMessageMoveTasks",
    "sqs:ReceiveMessage",
    "sqs:SendMessage",
    "sqs:GetQueueAttributes",
    "sqs:SetQueueAttributes"
  ],
  "Resource": [
    "arn:aws:sqs:*:*:SecurityLake_*",
    "arn:aws:sqs:*:*:AmazonSecurityLakeManager-*"
  ],
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
  }
},
{
  "Sid": "AllowDataLakeManagement",
  "Effect": "Allow",
  "Action": [
    "lakeformation:GetDataLakeSettings",
    "lakeformation:ListPermissions"
  ],

```

```
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "aws:ResourceAccount": "${aws:PrincipalAccount}"
      }
    }
  }
]
```

サービスリンク役割の作成、編集、削除を IAM エンティティ (ユーザー、グループ、役割など) に許可するにはアクセス許可を設定する必要があります。詳細については、「IAM User Guide」(IAM ユーザーガイド) の [「Service-linked role permissions」](#) (サービスリンクロールのアクセス権限) を参照してください。

Security Lake のサービスリンクロールの作成

Security Lake `AWSServiceRoleForSecurityLakeResourceManagement` のサービスにリンクされたロールは、Security Lake コンソールまたは `aws iam create-service-linked-role` を使用して作成できます AWS CLI。

サービスにリンクされたロールを作成するには、IAM ユーザーまたは IAM ロールに次のアクセス許可を付与する必要があります。IAM ロールは、Security Lake が有効なすべてのリージョンで Lake Formation 管理者である必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowLakeFormationActionsViaSecurityLakeConsole",
      "Effect": "Allow",
      "Action": [
        "lakeformation:GrantPermissions",
        "lakeformation:ListPermissions",
        "lakeformation:ListResources",
        "lakeformation:RegisterResource",
        "lakeformation:RevokePermissions"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowIamActionsViaSecurityLakeConsole",
      "Effect": "Allow",
      "Action": [
```

```

    "iam:CreateServiceLinkedRole",
    "iam:GetPolicyVersion",
    "iam:GetRole",
    "iam:PutRolePolicy"
  ],
  "Resource": [
    "arn:*:iam:*:role/aws-service-role/resource-
management.securitylake.amazonaws.com/AWSServiceRoleForSecurityLakeResourceManagement",
    "arn:*:iam:*:role/*AWSServiceRoleForLakeFormationDataAccess",
    "arn:*:iam::aws:policy/service-role/AWSGlueServiceRole",
    "arn:*:iam::aws:policy/service-role/AmazonSecurityLakeMetastoreManager",
    "arn:*:iam::aws:policy/aws-service-role/
SecurityLakeResourceManagementServiceRolePolicy"
  ],
  "Condition": {
    "StringLikeIfExists": {
      "iam:AWSServiceName": [
        "securitylake.amazonaws.com",
        "resource-management.securitylake.amazonaws.com",
        "lakeformation.amazonaws.com"
      ]
    }
  }
},
{
  "Sid": "AllowGlueActionsViaConsole",
  "Effect": "Allow",
  "Action": [
    "glue:GetDatabase",
    "glue:GetTables"
  ],
  "Resource": [
    "arn:*:glue:*:*:catalog",
    "arn:*:glue:*:*:database/amazon_security_lake_glue_db*",
    "arn:*:glue:*:*:table/amazon_security_lake_glue_db*/*"
  ]
}
]
}

```

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。

2. 概要ページの情報バーでサービスにリンクされたロールを有効にするをクリックして、新しいサービスにリンクされたロールを受け入れます。

サービスにリンクされたロールを有効にすると、Security Lake を将来使用するためにこのプロセスを繰り返す必要はありません。

CLI

AWSServiceRoleForSecurityLakeResourceManagement サービスにリンクされたロールをプログラムで作成するには、次の CLI コマンドを使用します。

```
$ aws iam create-service-linked-role
--aws-service-name resource-management.securitylake.amazonaws.com
```

を使用してAWSServiceRoleForSecurityLakeResourceManagementサービスにリンクされたロールを作成する場合 AWS CLI、テーブルメタデータを管理し、データにアクセスするために、Security Lake Glue データベース上のすべてのテーブルに Lake Formation テーブルレベルのアクセス許可 (ALTER、DESCRIBE) も付与する必要があります。任意のリージョンの Glue テーブルが以前の Security Lake 有効化の S3 バケットを参照する場合は、サービスにリンクされたロールへの DATA_LOCATION_ACCESS アクセス許可を一時的に許可して、Security Lake がこの状況を修復できるようにする必要があります。

また、Lake Formation のアクセス許可をアカウントのAWSServiceRoleForSecurityLakeResourceManagementサービスにリンクされたロールに付与する必要があります。

次の例は、指定されたリージョンのサービスにリンクされたロールに Lake Formation アクセス許可を付与する方法を示しています。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws lakeformation grant-permissions --region {region} --principal
DataLakePrincipalIdentifier={AWSServiceRoleForSecurityLakeResourceManagement ARN} \
--permissions ALTER DESCRIBE --resource '{ "Table": { "DatabaseName":
"amazon_security_lake_glue_db_{region}", "TableWildcard": {} } }'
```

次の例は、ロール ARN がどのように表示されるかを示しています。リージョンに合わせてロール ARN を編集する必要があります。

```
"AWS": "arn:[partition]:iam::[accountid]:role/aws-service-  
role/resource-management.securitylake.amazonaws.com/  
AWSServiceRoleForSecurityLakeResourceManagement"
```

[CreateServiceLinkedRole](#) API コールを使用することもできます。

リクエストで、`AWSServiceName`として指定します `resource-
management.securitylake.amazonaws.com`。

`AWSServiceRoleForSecurityLakeResourceManagement` ロールを有効にした後、暗号化に AWS KMS カスタマーマネージドキー (CMK) を使用している場合は、サービスにリンクされたロールが CMK が存在するリージョンの S3 バケットに暗号化されたオブジェクトを AWS 書き込むことを許可する必要があります。AWS KMS コンソールで、CMK が存在する AWS リージョンの KMS キーに次のポリシーを追加します。KMS キーポリシーを変更する方法の詳細については、「AWS Key Management Service デベロッパーガイド」の「[のキーポリシー AWS KMS](#)」を参照してください。

```
{  
  "Sid": "Allow SLR",  
  "Effect": "Allow",  
  "Principal": {  
    "AWS": "arn:[partition]:iam::[accountid]:role/aws-service-  
management.securitylake.amazonaws.com/AWSServiceRoleForSecurityLakeResourceManagement"  
  },  
  "Action": [  
    "kms:Decrypt",  
    "kms:GenerateDataKey*"  
  ],  
  "Resource": "*",  
  "Condition": {  
    "StringEquals": {  
      "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::[regional-datalake-s3-  
bucket-name]"  
    },  
    "StringLike": {  
      "kms:ViaService": "s3.[region].amazonaws.com"  
    }  
  }  
},
```

Security Lake 向けのサービスリンクロールの編集

Security Lake では、`AWSServiceRoleForSecurityLakeResourceManagement` サービスリンクロールの編集は許可されていません。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、IAM を使用してロールの説明を編集することはできます。詳細については、「[IAM ユーザーガイド](#)」の「サービスリンクロールの編集」を参照してください。

Security Lake 向けのサービスリンクロールの削除

サービスリンクロールを Security Lake から削除することはできません。代わりに、IAM コンソール、API、または `awscli` からサービスにリンクされたロールを削除できます AWS CLI。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削除](#)」を参照してください。

サービスリンクロールを削除する前に、まずロールにアクティブなセッションがないことを確認し、`AWSServiceRoleForSecurityLakeResourceManagement` が使用しているリソースを削除する必要があります。

Note

リソースを削除しようとするときに Security Lake が `AWSServiceRoleForSecurityLakeResourceManagement` ロールを使用している場合、削除は失敗する可能性があります。失敗した場合は、数分待ってから操作を再試行してください。

`AWSServiceRoleForSecurityLakeResourceManagement` サービスリンクロールを削除したが、再作成する必要がある場合は、アカウントで Security Lake を有効にすることで、ロールを再作成することができます。Security Lake を再作成すると、Security Lake は、ユーザーのためにサービスリンクロールを再作成します。

Security Lake サービスにリンクされたロール AWS リージョン でサポート

Security Lake は、Security Lake AWS リージョン が利用可能なすべての `AWSServiceRoleForSecurityLakeResourceManagement` サービスにリンクされたロールの使用をサポートしています。Security Lake が現在利用可能なリージョンのリストについては、「[Security Lake のリージョンとエンドポイント](#)」を参照してください。

Amazon Security Lake におけるデータ保護

Amazon Security Lake でのデータ保護には、AWS [責任共有モデル](#)が適用されます。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。ユーザーは、このインフラストラクチャでホストされ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に関するよくある質問](#)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 セキュリティブログに投稿された [AWS 責任共有モデルおよび GDPR](#) 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 (IAM) 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。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。TLS 1.2 が必須で、TLS 1.3 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。CloudTrail 証跡を使用して 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 証跡の使用](#)」を参照してください。
- AWS 暗号化ソリューションと、その中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 などの高度な管理された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 API AWS を介して にアクセスするときに FIPS 140-3 検証済み暗号化モジュールが必要な場合は、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、「[連邦情報処理規格 \(FIPS\) 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を、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含めないことを強くお勧めします。これは、コンソール、API、または SDK を使用して Security Lake AWS CLI または他の AWS のサービス を使用する場合も同様です。AWS SDKs タグ、または名前に使用される自由記述のテキストフィールドに入力したデータは、請求または診断ログに使用される場合があります。外部サーバーに URL を提供する場合、そのサーバーへのリクエストを検証できるように、認証情報を URL に含めないことを強くお勧めします。

保管中の暗号化

Amazon Security Lake は、AWS 暗号化ソリューションを使用して保管中のデータを安全に保存します。Raw セキュリティログとイベントデータは、Security Lake が管理するアカウントのソース固有の [マルチテナント Amazon Simple Storage Service \(Amazon S3\) バケット](#) に保存されます。各ログソースには、独自のマルチテナントバケットがあります。Security Lake は、AWS Key Management Service (AWS KMS) の [AWS 所有キー](#) を使用してこの raw データを暗号化します。AWS 所有キーは、AWS サービス、この場合は Security Lake が所有および管理する AWS KMS キーのコレクションで、複数の AWS アカウントで使用します。

Security Lake は、生のログとイベントデータに対して抽出、変換、ロード (ETL) ジョブを実行します。

ETL ジョブが完了すると、Security Lake はアカウントにシングルテナント S3 バケット (Security Lake を有効に AWS リージョンした各に 1 つのバケット) を作成します。データは、Security Lake がシングルテナント S3 バケットにデータを確実に配信できるようになるまで、一時的にのみマルチテナント S3 バケットに保存されます。シングルテナントバケットには、ログとイベントデータをバケットに書き込む権限を Security Lake に付与するリソースベースのポリシーが含まれています。S3 バケット内のデータを暗号化するには、[S3-managed暗号化キー](#) または [カスタマーマネージドキー](#) (から) のいずれかを選択できます AWS KMS。どちらの方法も対称暗号化を使用します。

KMS キーを使用してデータを暗号化します。

デフォルトでは、Security Lake によって S3 バケットに配信されるデータは、[Amazon S3 で管理された暗号化キー \(SSE-S3\)](#) を使用した Amazon サーバー側の暗号化によって暗号化されます。直接管理するセキュリティレイヤーを提供するには、代わりに Security Lake [データの AWS KMS キーによるサーバー側の暗号化 \(SSE-KMS\)](#) を使用できます。

SSE-KMS は、セキュリティレイクコンソールではサポートされていません。セキュリティレイク API または CLI で SSE-KMS を使用するには、まず [KMS キーを作成する](#) か、既存のキーを使用します。Security Lake データの暗号化と復号化にどのユーザーがキーを使用できるかを決定するポリシーをキーにアタッチします。

顧客管理キーを使用して S3 バケットに書き込まれるデータを暗号化する場合、マルチリージョンキーは選択できません。カスタマー管理キーの場合、Security Lake は CreateGrant リクエストを AWS KMS に送信することで、ユーザーに代わって [許可](#) を作成します。の許可 AWS KMS は、Security Lake に顧客アカウントの KMS キーへのアクセスを許可するために使用されます。

Security Lake では、次の内部操作に顧客管理キーを使用するための許可を必要とします。

- カスタマーマネージドキーで暗号化されたデータキーを生成する AWS KMS には、GenerateDataKey リクエストを送信します。
- RetireGrant リクエストの送信先 AWS KMS。データレイクを更新すると、この操作により、ETL 処理用の AWS KMS キーに追加されたグラントの廃止が可能になります。

セキュリティレイクには Decrypt 権限は必要ありません。キーの許可されたユーザーが Security Lake データを読み取ると、S3 が復号化を管理し、許可されたユーザーは暗号化されていない形式でデータを読み取ることができます。ただし、サブスクライバーがソースデータを使用するには Decrypt 権限が必要です。サブスクライバー権限の詳細については、[Security Lake サブスクライバーのデータアクセスの管理](#) を参照してください。

既存の KMS キーを使用して Security Lake データを暗号化する場合は、KMS キーのキーポリシーを変更する必要があります。キーポリシーは、Lake Formation データレイクの場所に関連付けられた IAM ロールが KMS キーを使用してデータを復号することを許可する必要があります。KMS キーのキーポリシーを変更する方法については、「AWS Key Management Service デベロッパーガイド」の「[キーポリシーの変更](#)」を参照してください。

キーポリシーを作成したり、適切な権限を持つ既存のキーポリシーを使用したりすると、KMS キーは付与リクエストを受け付けることができ、Security Lake がキーにアクセスできるようになります。キーポリシーの作成手順については、AWS Key Management Service 開発者ガイドの[キーポリシーの作成](#)を参照してください。

次のキーポリシーを KMS キーにアタッチします。

```
{
  "Sid": "Allow use of the key",
  "Effect": "Allow",
  "Principal": {"AWS": "arn:aws:iam::111122223333:role/ExampleRole"},
  "Action": [
    "kms:CreateGrant",
    "kms:DescribeKey",
    "kms:GenerateDataKey"
  ],
  "Resource": "*"
}
```

顧客マネージドキーを使用する場合に必要な IAM アクセス許可

Security Lake を使用するために作成する必要がある IAM ロールの概要については、「[はじめに:前提条件](#)」セクションを参照してください。

カスタムソースまたはサブスクライバーを追加すると、Security Lake はアカウントに IAM ロールを作成します。これらのロールは他の IAM ID と共有することを目的としています。これにより、カスタムソースはデータレイクにデータを書き込み、サブスクライバーはデータレイクからのデータを使用できます。という AWS マネージドポリシーは、これらのロールのアクセス許可の境界 AmazonSecurityLakePermissionsBoundary を設定します。

Amazon SQS キューの暗号化

データレイクを作成すると、Security Lake は委任されたセキュリティレイク管理者アカウントに 2 つの暗号化されていない Amazon Simple Queue Service (Amazon SQS) キューを作成します。データを保護するには、これらのキューを暗号化する必要があります。Amazon Simple Queue Service が提供するデフォルトのサーバー側の暗号化では十分ではありません。(AWS Key Management Service AWS KMS) でカスタマーマネージドキーを作成してキューを暗号化し、暗号化されたキューを操作するためのアクセス許可を Amazon S3 サービスプリンシパルに付与する必要があります。これらのアクセス許可を付与する手順については、AWS ナレッジセンターの「[サーバー側の暗号化を使用する Amazon SQS キューに Amazon S3 イベント通知が配信されないのはなぜですか？](#)」を参照してください。

Security Lake は AWS Lambda を使用してデータの抽出、転送、ロード (ETL) ジョブをサポートするため、Amazon SQS キュー内のメッセージを管理するためのアクセス許可も Lambda に付与する必要があります。詳細については、「AWS Lambda デベロッパーガイド」の「[実行ロールのアクセス許可](#)」を参照してください。

転送中の暗号化

Security Lake は、AWS サービス間で転送中のすべてのデータを暗号化します。Security Lake は、トランスポート層セキュリティ (TLS) 1.2 暗号化プロトコルを使用してすべてのネットワーク間データを自動的に暗号化することにより、サービスとの間で送受信されるデータを保護します。Security Lake API に送信されるダイレクト HTTPS リクエストは、[署名バージョン 4 AWS アルゴリズムを使用して署名され](#)、安全な接続を確立します。

サービス改善のためのデータ使用をオプトアウトする

オプトアウトポリシーを使用して、Security Lake およびその他の AWS セキュリティサービスの開発と改善にデータを使用することを AWS Organizations オプトアウトできます。Security Lake が現

在那のようなデータを収集していない場合でも、オプトアウトすることができます。オプトアウトする方法の詳細については、「AWS Organizations ユーザーガイド」の「[AI サービスのオプトアウトポリシー](#)」を参照してください。

現在、Security Lake は、ユーザーに代わって処理するセキュリティデータや、このサービスによって作成されたセキュリティデータレイクにユーザーがアップロードしたセキュリティデータを収集することはありません。Security Lake サービスおよび他の AWS セキュリティサービスの機能を開発および改善するために、Security Lake は、サードパーティーのデータソースからアップロードしたデータを含め、将来そのようなデータを収集することがあります。Security Lake がそのようなデータを収集する予定がある場合は、このページを更新し、その仕組みについて説明します。ただし、いつでもオプトアウトすることができます。

Note

オプトアウトポリシーを使用するには、AWS アカウントが によって一元管理されている必要があります AWS Organizations。AWS アカウント用の組織をまだ作成していない場合は、「AWS Organizations ユーザーガイド」の「[組織の作成と管理](#)」を参照してください。

オプトアウトには次のような効果があります。

- Security Lake は、オプトアウト前に収集して保存したデータ (ある場合のみ) を削除します。
- オプトアウトすると、Security Lake はこのデータを収集または保存しません。

Amazon SECURITY Lake のコンプライアンス検証

AWS のサービス が特定のコンプライアンスプログラムの範囲内にあるかどうかを確認するには、コンプライアンス [AWS のサービス プログラムによる範囲内コンプライアンス](#) を参照し、関心のあるコンプライアンスプログラムを選択します。一般的な情報については、[AWS 「Compliance Programs Assurance」](#) を参照してください。

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 AWS Artifact。詳細については、「[Downloading AWS Artifact reports](#)」を参照してください。

を使用する際のお客様のコンプライアンス責任 AWS のサービス は、お客様のデータの機密性、貴社のコンプライアンス目的、適用される法律および規制によって決まります。では、コンプライアンスに役立つ以下のリソース AWS を提供しています。

- [セキュリティのコンプライアンスとガバナンス](#) – これらのソリューション実装ガイドでは、アーキテクチャ上の考慮事項について説明し、セキュリティとコンプライアンスの機能をデプロイする手順を示します。
- [HIPAA 対応サービスのリファレンス](#) – HIPAA 対応サービスの一覧が提供されています。すべてが HIPAA 対応 AWS のサービスであるわけではありません。
- [AWS コンプライアンスリソース](#) – このワークブックとガイドのコレクションは、お客様の業界や地域に適用される場合があります。
- [AWS カスタマーコンプライアンスガイド](#) – コンプライアンスの観点から責任共有モデルを理解します。このガイドは、複数のフレームワーク (米国国立標準技術研究所 (NIST)、Payment Card Industry Security Standards Council (PCI)、国際標準化機構 (ISO) を含む) のセキュリティコントロールを保護し、そのガイダンスに AWS のサービス マッピングするためのベストプラクティスをまとめたものです。
- 「[デベロッパーガイド](#)」の「[ルールによるリソースの評価](#)」 – この AWS Config サービスは、リソース設定が社内プラクティス、業界ガイドライン、および規制にどの程度準拠しているかを評価します。AWS Config
- [AWS Security Hub](#) – これにより AWS のサービス、セキュリティ状態を包括的に把握できます。AWS Security Hub では、セキュリティコントロールを使用して AWS リソースを評価し、セキュリティ業界標準とベストプラクティスに対するコンプライアンスをチェックします。サポートされているサービスとコントロールの一覧については、[Security Hub のコントロールリファレンス](#)を参照してください。
- [Amazon GuardDuty](#) – 環境をモニタリングして AWS アカウント不審なアクティビティや悪意のあるアクティビティがないか調べることで、ワークロード、コンテナ、データに対する潜在的な脅威 AWS のサービスを検出します。GuardDuty を使用すると、特定のコンプライアンスフレームワークで義務付けられている侵入検知要件を満たすことで、PCI DSS などのさまざまなコンプライアンス要件に対応できます。
- [AWS Audit Manager](#) – これにより AWS のサービス、AWS 使用状況を継続的に監査し、リスクの管理方法と規制や業界標準への準拠を簡素化できます。

Security Lake のセキュリティのベストプラクティス

Amazon Security Lake を使用するには、次のベストプラクティスを確認します。

Security Lake ユーザーへの最小限のアクセス許可の付与

AWS Identity and Access Management (IAM) ユーザー、ユーザーグループ、およびロールに最小限のアクセスポリシー許可を付与することで、最小特権の原則に従います。たとえば、IAM ユーザーに Security Lake のログソースのリストの表示を許可するが、ソースやサブスクライバーの作成は許可しない場合があります。詳細については、「[Security Lake のアイデンティティベースのポリシーの例](#)」を参照してください。

AWS CloudTrail を使用して、Security Lake で API の使用状況を追跡することもできます。CloudTrail は、Security Lake のユーザー、グループ、またはロールによって実行された API アクションの記録を提供します。詳細については、「[CloudTrail を使用した Security Lake API コールのログ記録](#)」を参照してください。

概要ページを表示します。

Security Lake コンソールの概要ページには、Security Lake サービスとデータが保存されている Amazon S3 バケットに影響を与えている過去 14 日間の問題の概要が表示されます。これらの問題をさらに調査して、起こり得るセキュリティ関連の影響を軽減するのに役立ちます。

Security Hubとの統合

Security Lake とを統合し AWS Security Hub で、Security Hub の検出結果を受信します。Security Hub は、さまざまな AWS のサービスおよびサードパーティーの統合から結果を生成します。Security Hub の検出結果を受け取ると、コンプライアンス体制の概要と、AWS セキュリティのベストプラクティスを満たしているかどうかを把握できます。

詳細については、「[との統合 AWS Security Hub](#)」を参照してください。

削除 AWS Lambda

AWS Lambda 関数を削除するときは、最初に無効にしないことをお勧めします。削除前に Lambda 関数を無効にすると、データクエリ機能が妨げられ、他の機能に影響を与える可能性があります。Lambda 関数は無効にせずに直接削除することをお勧めします。Lambda 関数の削除の詳細については、[AWS Lambda 「デベロッパーガイド」](#)を参照してください。

セキュリティレイクのイベントを監視してください。

Amazon CloudWatch メトリクスを使用して Security Lake をモニタリングできます。CloudWatch は、Security Lake から生データを毎分収集し、それをメトリクスに処理します。メトリックスが指定したしきい値に一致したときに通知をトリガーするアラームを設定できます。

詳細については、「[Amazon Security Lake の CloudWatch メトリクス](#)」を参照してください。

Amazon Security Lake の耐障害性

AWS グローバルインフラストラクチャは、AWS リージョン およびアベイラビリティゾーンを中心に構築されています。は、低レイテンシー、高スループット、および高度に冗長なネットワークで接続された、物理的に分離および分離された複数のアベイラビリティゾーン AWS リージョンを提供します。これらのアベイラビリティゾーンを利用すると、アプリケーションとデータベースを効率的に設計して運用できます。アベイラビリティゾーンは、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、耐障害性、およびスケーラビリティが優れています。

セキュリティレイクの可用性は、リージョンの可用性と結びついています。複数のアベイラビリティゾーンに分散することで、サービスはどのアベイラビリティゾーンでも障害に耐えることができます。

Security Lake データプレーンの可用性は、どのリージョンの可用性とも関係ありません。ただし、Security Lake コントロールプレーンの可用性は、米国東部 (バージニア北部) リージョンの可用性と密接に関係しています。

AWS リージョン およびアベイラビリティ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。

AWS グローバルインフラストラクチャである Security Lake に加えて、Amazon Simple Storage Service (Amazon S3) によってデータがバックアップされます。には、データの耐障害性とバックアップのニーズをサポートするのに役立ついくつかの機能があります。

ライフサイクル設定

ライフサイクル設定は、Amazon S3 がオブジェクトのグループに適用するアクションを定義するルールのセットです。ライフサイクル設定ルールを使用すると、オブジェクトのより安価なストレージクラスへの移行、アーカイブ、削除を Amazon S3 に指定できます。詳細については、Amazon S3 ユーザーガイドの「[ストレージのライフサイクルの管理](#)」をご参照ください。

バージョニング

バージョニングとは、同じバケット内でオブジェクトの複数のバリエーションを保持する手段です。バージョニングを使用して、Amazon S3 バケットに格納されたあらゆるオブジェクトのあらゆるバージョンを、格納、取得、復元することができます。バージョニングによって、意図しないユーザーアクションとアプリケーション障害から復旧できます。詳細については、『Amazon S3 ユーザーガイド』の「[S3 バケットでのバージョニングの使用](#)」を参照してください。

ストレージクラス

Amazon S3 では、ワークロードの要件に応じて、幅広いストレージクラスが提供されています。S3 標準 — IA と S3 1 ゾーン — IA ストレージクラスは、月に約 1 回アクセスし、ミリ秒単位のアクセスが必要になるデータ用に設計されています。S3 Glacier インスタント検索ストレージクラスは、四半期に約 1 回アクセスするミリ秒のアクセスでアクセスされる長期間有効なアーカイブデータ用に設計されています。バックアップなど、即時アクセスを必要としないアーカイブデータについては、S3 Glacier Flexible Retrieval または S3 Glacier Deep Archive ストレージクラスを使用できます。詳細については、『Amazon S3 ユーザーガイド』の「[Amazon S3 ストレージクラスの使用](#)」を参照してください。

Amazon Security Lake のインフラストラクチャセキュリティ

マネージドサービスである Amazon Security Lake は グローバル AWS 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 [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、「セキュリティの柱 AWS Well-Architected フレームワーク」の「[インフラストラクチャの保護](#)」を参照してください。

AWS が公開した API コールを使用して、ネットワーク経由で Security Lake 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)。TLS 1.2 が必須で、TLS 1.3 をお勧めします。
- DHE (楕円ディフィー・ヘルマン鍵共有) や ECDHE (楕円曲線ディフィー・ヘルマン鍵共有) などの完全前方秘匿性 (PFS) による暗号スイート。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。

また、リクエストにはアクセスキー ID と、IAM プリンシパルに関連付けられている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、一時的な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。

Security Lake での構成と脆弱性の分析

設定と IT コントロールは、AWS とお客様の間で責任を共有します。詳細については、AWS [「責任共有モデル」](#) を参照してください。

Amazon Security Lake とインターフェイス VPC エンドポイント (AWS PrivateLink)

インターフェイス VPC エンドポイントを作成することで、VPC と Amazon Security Lake 間のプライベート接続を確立できます。インターフェイスエンドポイントは、インターネットゲートウェイ [AWS PrivateLink](#)、NAT デバイス、VPN 接続、AWS Direct Connect 接続のいずれも必要とせずに Security Lake APIs にプライベートにアクセスできるテクノロジーである [AWS PrivateLink](#) を利用しています。VPC 内のインスタンスは、Security Lake APIs と通信するためにパブリック IP アドレスを必要としません。VPC と Security Lake 間のトラフィックは Amazon ネットワークを離れません。

各インターフェイスエンドポイントは、サブネット内の 1 つ以上の [Elastic Network Interface](#) によって表されます。

詳細については、「AWS PrivateLink ガイド」の「[インターフェイス VPC エンドポイント \(AWS PrivateLink\)](#)」を参照してください。

Security Lake VPC エンドポイントに関する考慮事項

Security Lake のインターフェイス VPC エンドポイントを設定する前に、「AWS PrivateLink ガイド」の「[インターフェイスエンドポイントのプロパティと制限](#)」を確認してください。

Security Lake は、VPC からのすべての API アクションの呼び出しをサポートしています。

Security Lake は、FIPS が存在する次のリージョンでのみ FIPS VPC エンドポイントをサポートします。

- 米国東部 (バージニア北部)
- 米国東部 (オハイオ)
- 米国西部 (北カリフォルニア)
- 米国西部 (オレゴン)

Security Lake 用のインターフェイス VPC エンドポイントの作成

Security Lake サービスの VPC エンドポイントは、Amazon VPC コンソールまたは AWS Command Line Interface () を使用して作成できます。AWS CLI。詳細については、「AWS PrivateLink ガイド」の「[インターフェイスエンドポイントを作成](#)」を参照してください。

次のサービス名を使用して、Security Lake の VPC エンドポイントを作成します。

- com.amazonaws.*region*.securitylake
- com.amazonaws.*region*.securitylake-fips (FIPS エンドポイント)

エンドポイントのプライベート DNS を有効にすると、などのリージョンのデフォルトの DNS 名を使用して Security Lake に API リクエストを行うことができます securitylake.us-east-1.amazonaws.com。

詳細については、AWS PrivateLink ガイドの [インターフェイスエンドポイントを介したサービスへのアクセス](#) を参照してください。

Security Lake 用の VPC エンドポイントポリシーの作成

Security Lake へのアクセスを制御するエンドポイントポリシーを VPC エンドポイントにアタッチできます。このポリシーでは、以下の情報を指定します。

- アクションを実行できるプリンシパル。
- 実行可能なアクション。
- このアクションを実行できるリソース。

詳細については、「AWS PrivateLink ガイド」の「[VPC エンドポイントでサービスへのアクセスを制御する](#)」を参照してください。

例: Security Lake アクションの VPC エンドポイントポリシー

以下は、Security Lake のエンドポイントポリシーの例です。このポリシーは、エンドポイントにアタッチされると、すべてのリソースのすべてのプリンシパルに対して、リストされている Security Lake アクションへのアクセスを許可します。

```
{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "securitylake:ListDataLakes",
        "securitylake:ListLogSources",
        "securitylake:ListSubscribers"
      ],
      "Resource": "*"
    }
  ]
}
```

```
}  
  ]  
}
```

共有サブネット

自分と共有されているサブネットで VPC エンドポイントを作成、説明、変更、または削除することはできません。ただし、自分と共有されているサブネットで VPC エンドポイントを使用することはできます。VPC 共有の詳細については、「Amazon VPC ユーザーガイド」の「[VPC を他のアカウントと共有する](#)」を参照してください。

Amazon Security Lake のモニタリング

Security Lake は と統合されています。これは AWS CloudTrail、ユーザー、ロール、または別の によって Security Lake で実行されたアクションを記録するサービスです AWS のサービス。これには、Security Lake コンソールからのアクションと、Security Lake API オペレーションへのプログラムによる呼び出しが含まれます。CloudTrail によって収集された情報を使用すると、どのリクエストが Security Lake に対して行われたかを判断できます。リクエストごとに、リクエスト日時、リクエスト元の IP アドレス、作成者、その他の詳細を確認できます。詳細については、「[CloudTrail を使用した Security Lake API コールのログ記録](#)」を参照してください。

Security Lake と Amazon CloudWatch は統合されているため、Security Lake が収集するログについてメトリクスを収集、表示、分析できます。Security Lake データレイクの CloudWatch メトリクスは自動的に収集され、1 分間隔で CloudWatch にプッシュされます。Security Lake メトリクスの指定のしきい値に達した場合に、通知が送信されるよう、アラームを設定することもできます。Security Lake が CloudWatch に送信するすべてのメトリクスのリストについては、「[Security Lake のメトリクスとディメンション](#)」を参照してください。

Amazon Security Lake の CloudWatch メトリクス

Amazon CloudWatch を使用して Security Lake を監視できます。Amazon CloudWatch は生データを毎分収集し、それを読み取り可能なほぼリアルタイムのメトリクスに処理します。これらの統計は 15 か月間保存されるため、履歴情報にアクセスして、データレイク内のデータをより正確に把握できます。また、特定のしきい値を監視するアラームを設定し、これらのしきい値に達したときに通知を送信したりアクションを実行したりできます。

トピック

- [Security Lake のメトリクスとディメンション](#)

- [Security Lake の CloudWatch メトリクスの表示](#)
- [Security Lake メトリクスの CloudWatch アラームの設定](#)

Security Lake のメトリクスとディメンション

AWS/SecurityLake 名前空間には、次のメトリクスが含まれます。

メトリクス	説明
ProcessedSize	<p>現在データレイクに保存 AWS のサービス されているネイティブにサポートされている からのデータボリューム。</p> <p>単位: バイト</p>

Security Lakeメトリクス では以下のディメンションが利用可能です。

ディメンション	説明
Account	特定の AWS アカウントの Processed Size メトリクス。このディメンションは、CloudWatch で Per-Account Source Version Metrics を表示する場合にのみ使用できます。
Region	特定の AWS リージョンの Processed Size メトリクス。
Source	ProcessedSize 特定の AWS ログソースのメトリクス。
SourceVersion	ProcessedSize AWS ログソースの特定バージョンのメトリクス。

specific AWS アカウント (Per-Account Source Version Metrics) または組織内のすべてのアカウント () のメトリクスを表示できますPer-Source Version Metrics。

Security Lake の CloudWatch メトリクスの表示

Security Lake のメトリクスは、CloudWatch コンソール、CloudWatch 独自のコマンドラインインターフェイス (CLI) を使用するか、CloudWatch API を使用してプログラムで監視できます。ご希望の方法を選択し、手順に従って Security Lake メトリクスにアクセスします。

CloudWatch console

1. CloudWatch コンソール (<https://console.aws.amazon.com/cloudwatch/>) を開きます。
2. ナビゲーションペインで、[Metrics](メトリクス)、[All metrics] (すべてのメトリクス) を選択します。
3. 「ブラウズ」タブで「Security Lake」を選択します。
4. [アカウントごとのソースバージョンメトリクス] または [ソースバージョンごとのメトリクス] を選択します。
5. メトリクスを選択して詳細を表示します。また、以下を実行することもできます。
 - メトリクスを並べ替えるには、列見出しを使用します。
 - メトリクスをグラフ表示するには、メトリクス名を選択し、グラフ表示オプションを選択します。
 - メトリクスでフィルタするには、メトリクスの名前を選択し、[Add to search] を選択します。

CloudWatch API

CloudWatch APIを使用して Security Lakeメトリクスにアクセスするには、[GetMetricStatistics](#) アクションを使用します。

AWS CLI

を使用して Security Lake メトリクスにアクセスするには AWS CLI、[get-metric-statistics](#) コマンドを実行します。

CloudWatch メトリクスを使用したモニタリング方法の詳細については、[Amazon CloudWatch ユーザーガイド] の [\[Amazon CloudWatch Metrics の使用\]](#) を参照してください。

Security Lake メトリクスの CloudWatch アラームの設定

CloudWatch では、メトリクスのしきい値に到達したときのアラームを設定することもできます。たとえば、ProcessedSize メトリクスにアラームを設定して、特定のソースからのデータ量が特定のしきい値を超えたときに通知を受けることができます。

アラームの設定に関する詳細については、Amazon CloudWatch ユーザーガイドの「[Using Amazon CloudWatch Alarm](#)」を参照してください。

CloudTrail を使用した Security Lake API コールのログ記録

Amazon Security Lake は AWS CloudTrail、Security Lake のユーザー、ロール、または サービスによって実行されたアクションを記録する AWS サービスであると統合されます。CloudTrail は、Security Hub の API コールをイベントとしてキャプチャします。キャプチャされたコールには、Security Lake コンソールからのコールと、Security Lake API オペレーションへのコードコールが含まれます。追跡を作成すると、Security Lake のイベントなどを含んだ Amazon S3 バケットへの CloudTrail イベントの継続的な送信を有効にすることができます。証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表示できます。CloudTrail によって収集された情報を使用して、Security Lake 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、リクエストが行われた日時、および追加の詳細を確認することができます。

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 での Security Security Security Information

アカウントを作成する AWS アカウントと、で CloudTrail が有効になります。Security Lake 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の他の AWS サービスイベントとともに CloudTrail イベントに記録されます。で最近のイベントを表示、検索、ダウンロードできます AWS アカウント。詳細については、[CloudTrail イベント履歴でのイベントの表示](#)を参照してください。

Security Lake のイベントなど AWS アカウント、のイベントの継続的な記録については、証跡を作成します。追跡を使用すると、CloudTrail はイベントをログ ファイルとして指定した 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 パーティション内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、それに基づいて行動するように他の AWS サービスを設定できます。詳細については、次を参照してください：

- [追跡を作成するための概要](#)
- 「[CloudTrail がサポートされているサービスと統合](#)」
- 「[CloudTrail の Amazon SNS 通知の設定](#)」
- 「[複数のリージョンから CloudTrail ログファイルを受け取る](#)」 および 「[複数のアカウントから CloudTrail ログファイルを受け取る](#)」

Amazon Translate のすべてのアクションは CloudTrail によってログに記録され、[API Lake API リファレンス](#)に記載されます。例えば、UpdateDataLake、ListLogSources、CreateSubscriber の各アクションを呼び出すと、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、リクエストの生成者に関する情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます。

- リクエストがルート認証情報と AWS Identity and Access Management ユーザー認証情報のどちらを使用して行われたか。
- リクエストがロールまたはフェデレーションユーザーのテンポラリなセキュリティ認証情報を使用して行われたかどうか。
- リクエストが別の AWS サービスによって行われたかどうか。

詳細については、[CloudTrail userIdentity 要素](#)を参照してください。

Security Lake のログファイルエントリについて

CloudTrail のログファイルは、単一か複数のログエントリを含みます。イベントは任意ソースからの単一リクエストを表し、リクエストされたアクション、アクションの日時、リクエストパラメータなどの情報を含みます。CloudTrail ログファイルは、パブリック API 呼び出し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。

次の例は、GetSubscriber アクションの CloudTrail ログエントリを示しています。

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE:user",
    "arn": "arn:aws:sts::123456789012:assumed-role/Admin/user",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::123456789012:role/Admin",
        "accountId": "123456789012",
```

```
    "userName": "Admin"
  },
  "webIdFederationData": {
  },
  "attributes": {
    "creationDate": "2023-05-30T13:27:19Z",
    "mfaAuthenticated": "false"
  }
}
},
"eventTime": "2023-05-30T17:29:17Z",
"eventSource": "securitylake.amazonaws.com",
"eventName": "GetSubscriber",
"awsRegion": "us-east-1",
"sourceIPAddress": "198.51.100.1",
"userAgent": "console.amazonaws.com",
"requestParameters": {
  "subscriberId": "30ed17a3-0cac-4997-a41f-f5a6bexample"
},
"responseElements": null,
"requestID": "d01f0f32-9ec6-4579-af50-e9f14example",
"eventID": "9c1bff41-0f48-4ee6-921c-ebfd8example",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "123456789012",
"eventCategory": "Management"
}
```

Security Lake リソースのタグ付け

タグは、特定のタイプの Amazon Security Lake AWS リソースなど、リソースを定義して割り当てることのできるオプションのラベルです。タグは、目的、所有者、環境、その他の基準など、さまざまな方法でリソースを識別、分類、管理するのに役立ちます。たとえば、タグを使用して、ポリシーの適用、リソースの区別、特定のコンプライアンス要件やワークフローをサポートするリソースの識別を行うことができます。

サブスクライバーと AWS アカウント 個々の のデータレイク設定の種類の Security Lake リソースにタグを割り当てることができます AWS リージョン。

トピック

- [タグ付けの基本](#)
- [IAMポリシーでタグを使用する](#)
- [Amazon セキュリティレイクリソースへのタグの追加](#)
- [Amazon Security Lake リソースのタグを編集する](#)
- [Amazon Security Lake リソースからのタグの削除](#)

タグ付けの基本

リソースには、最大 50 個のタグを含めることができます。タグはそれぞれ、1 つの必須タグキーとオプションの 1 つのタグ値で設定されており、どちらもお客様側が定義します。タグキーは、より具体的なタグ値のカテゴリとして動作する一般的なラベルです。タグ値は、タグキーの記述子として機能します。

たとえば、さまざまな環境 (クラウドデータ用とオンプレミスデータ用のサブスクライバーセット) のセキュリティデータを分析するサブスクライバーを追加する場合、それらのサブスクライバーに Environment タグキーを割り当てることができます。関連するタグ値は、からのデータを分析するサブスクライバー Cloud の場合は AWS のサービス、他のサブスクライバー On-Premises の場合はです。

Amazon Security Lake リソースにタグを定義して割り当てる際、以下の点に注意してください。

- 各リソースには、最大 50 個のタグを設定できます。
- リソースごとに、各タグ キーは一意である必要があり、タグ値は 1 つだけ持つことができます。

- タグのキーと値では、大文字と小文字が区別されます。タグに大文字を使用する場合の戦略を決定し、その戦略をすべてのリソースにわたって一貫して実装することをベストプラクティスとして推奨します。
- タグキーは最大 128 文字 (UTF-8) です。タグ値は最大 256 文字 (UTF-8) です。文字には、文字、数字、スペース、または記号 (_ . : / = + - @) を使用できます。
- aws: プレフィックスは が使用するために予約されています AWS。定義したどのタグキーやタグ値にも使用できません。さらに、このプレフィックスを使用するタグキーまたは値は変更または削除できません。このプレフィックスを使用するタグは、リソースあたりのタグ数のクォータ (50 個) にはカウントされません。
- 割り当てたタグは、AWS アカウント および割り当てた AWS リージョン でのみ使用できます。
- Security Lake を使用してリソースにタグを割り当てる場合、タグは該当する AWS リージョンの Security Lake に直接保存されているリソースにのみ適用されます。これらは、Security Lake が他の AWS のサービスで作成、使用、または維持する、関連するサポート リソースには適用されません。たとえば、データレイクにタグを割り当てた場合、タグは指定されたリージョンの Security Lake 内のデータレイク設定にのみ適用されます。ログやイベントデータが保存されている Amazon Simple Storage Service (Amazon S3) バケットには適用されません。関連付けられたリソースにタグを割り当てるには、AWS Resource Groups またはリソース AWS のサービス を保存する を使用します。たとえば、SAmazon S3S3 などです。関連するリソースにタグを割り当てると、データレイクをサポートするリソースを特定しやすくなります。
- リソースを削除すると、そのリソースに割り当てられているタグも削除されます。

その他の制限、ヒント、ベストプラクティスについては、[「AWS リソースのタグ付け」](#)ユーザーガイドの「AWS リソースのタグ付け」を参照してください。

Important

機密データやその他の機密データをタグに保存しないでください。タグには AWS のサービス、を含む多くの からアクセスできます AWS Billing and Cost Management。それらは機密データに使用することを目的としていません。

Security Lake リソースのタグを追加および管理するには、Security Lake コンソールまたは Security Lake API を使用できます。

IAMポリシーでタグを使用する

リソースのタグ付けを開始した後、タグベースのリソースレベルのアクセス許可を AWS Identity and Access Management (IAM) ポリシーで定義できます。この方法でタグを使用することで、内のどのユーザーとロールにリソースの作成とタグ付けのアクセス許可 AWS アカウント があり、どのユーザーとロールにタグの追加、編集、削除のアクセス許可があるかを詳細に制御できます。タグに基づいてアクセスを制御するには、IAM ポリシーの [条件要素](#) で [タグ関連の条件キー](#) を使用できます。

たとえば、リソースの Owner タグでユーザー名が指定されている場合、ユーザーにすべての Amazon Security Lake リソースへのフルアクセスを許可するポリシーを作成できます。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ModifyResourceIfOwner",
      "Effect": "Allow",
      "Action": "securitylake:*",
      "Resource": "*",
      "Condition": {
        "StringEqualsIgnoreCase": {"aws:ResourceTag/Owner": "${aws:username}"}
      }
    }
  ]
}
```

タグをベースにしてリソースレベルでアクセス許可を定義した場合、そのアクセス許可は即座に反映されます。つまり、リソースが作成されるとすぐにリソースの安全性が増し、新しいリソースにタグの使用をすぐに強制できるようになります。リソースレベルのアクセス許可を使用して、新しいリソースと既存のリソースに、どのタグキーと値を関連付けるかを制御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 [「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。

Amazon セキュリティレイクリソースへのタグの追加

Amazon セキュリティレイクリソースにタグを追加するには、セキュリティレイクコンソールまたはセキュリティレイク API を使用できます。

⚠ Important

リソースにタグを追加すると、リソースへのアクセスに影響を与える可能性があります。リソースにタグを追加する前に、タグを使用してリソースへのアクセスを制御する可能性のある AWS Identity and Access Management (IAM) ポリシーを確認してください。

Console

の Security Lake を有効にする AWS リージョン か、サブスクライバーを作成すると、Security Lake コンソールには、リージョンまたはサブスクライバーのデータレイク設定などのリソースにタグを追加するためのオプションが表示されます。リソースを作成したら、コンソールの指示に従ってリソースにタグを追加します。

Security Lake コンソールを使用して、既存のリソースに 1 つ以上のタグを追加するには、以下のステップに従います。

リソースにタグを追加する

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. タグを追加するリソースのタイプに応じて、次のいずれかを実行します。
 - データレイク設定は、ナビゲーションペインで [リージョン] を選択します。次に、「リージョン」テーブルで「リージョン」を選択します。
 - サブスクライバーの場合は、ナビゲーションペインで [サブスクライバー] を選択します。次に、「マイサブスクライバー」テーブルでサブスクライバーを選択します。

サブスクライバーがテーブルに表示されない場合は、ページの右上隅の AWS リージョンセレクターを使用して、サブスクライバーを作成したリージョンを選択します。表には、現在のリージョンに関してのみの既存のサブスクライバーのみが表示されます。

3. [編集] を選択します。
4. [タグ] セクションを展開します。このセクションには、リソースに現在割り当てられているすべてのタグが一覧表示されます。
5. In the Tags section, choose Add new tag.
6. Key ボックスに、リソースに追加するタグのタグキーを入力します。次に、[値] ボックスに、任意でキーのタグ値を入力します。

タグキーには最大 128 文字を含めることができます。タグ値は最大 256 文字を含めることができます。文字には、文字、数字、スペース、または次の記号を使用できます: _ . : / = + - @

7. リソースに別のタグを追加するには、[Add new tag] を選択し、前のステップを繰り返します。1 つのリソースには、最大 50 個のタグを割り当てることができます。
8. タグの追加を完了したら、保存 を選択します。

API

リソースを作成して 1 つ以上のタグをプログラムで追加するには、作成するリソースのタイプに適した Create 操作を使用します。

- データレイク設定 — [CreateDataLake](#) オペレーションを使用するか、AWS Command Line Interface (AWS CLI) を使用している場合は [create-data-lake](#) コマンドを実行します。
- サブスクライバー — [CreateSubscriber](#) オペレーションを使用するか、AWS CLI を使用している場合は [create-subscriber](#) コマンドを実行します。

リクエストでは、tags パラメータを使用して、リソースに追加する各タグのタグキー (key) とオプションのタグ値 (value) を指定します。tags パラメータは、オブジェクトの配列を指定します。各オブジェクトはタグキーとそれに関連するタグ値を指定します。

既存のリソースに 1 つ以上のタグを追加するには、Security Lake API の [TagResource](#) オペレーションを使用するか、を使用している場合は [tag-resource](#) コマンド AWS CLI を実行します。リクエストでは、タグを追加するリソースの Amazon リソースネーム (ARN) を指定します。tags パラメータを使用して、追加する各タグのタグキー (key) とオプションのタグ値 (value) を指定します。Create 操作やコマンドの場合と同様に、tags パラメータはオブジェクトの配列、つまり各タグキーとそれに関連するタグ値に 1 つのオブジェクトを指定します。

例えば、次の AWS CLI コマンドは、指定されたサブスクライバーに Environment タグ値を持つ Cloud タグキーを追加します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake tag-resource \  
--resource-arn arn:aws:securitylake:us-  
east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab \  
--tags key=Environment,value=Cloud
```

コードの説明は以下のとおりです。

- `resource-arn` タグを追加するサブスクライバーの ARN を指定します。
- `Environment` サブスクライバーに追加するタグのタグキーです。
- `Cloud` は、指定されたタグ キー (`Environment`) のタグ値です。

次の例では、コマンドはサブスクライバーに複数のタグを追加します。

```
$ aws securitylake tag-resource \  
--resource-arn arn:aws:securitylake:us-  
east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab \  
--tags key=Environment,value=Cloud key=CostCenter,value=12345 key=Owner,value=jane-  
doe
```

tags 配列内の各オブジェクトには、keyvalue との引数の両方が必要です。ただし、value 引数の値は空の文字列とすることができます。タグ値をタグキーに関連付けない場合、value 引数の値を指定しないでください。たとえば、以下のコマンドは、関連付けられたタグ値を含まない `Owner` タグキーを追加します。

```
$ aws securitylake tag-resource \  
--resource-arn arn:aws:securitylake:us-  
east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab \  
--tags key=Owner,value=
```

タグ付けオペレーションが正常に実行された場合は Security Lake は空の HTTP 200 レスポンスを返します。それ以外の場合、Security Lake は HTTP 4xx またはオペレーションが失敗した理由を示す 500 レスポンスを返します。

Amazon Security Lake リソースのタグを編集する

Amazon Security Lake リソースのタグ (タグキーまたはタグ値) を編集するには、セキュリティレイクコンソールまたは Security Lake API を使用できます。

⚠ Important

リソースのタグを編集すると、リソースへのアクセスに影響する可能性があります。リソースのタグキーまたは値を編集する前に、タグを使用してリソースへのアクセスを制御する可能性のある AWS Identity and Access Management (IAM) ポリシーを確認してください。

Console

Security Lake コンソールを使用してリソースのタグを編集するには、以下のステップに従います。

リソースのタグを編集するには

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. タグを編集するリソースのタイプに応じて、次のいずれかの操作を行います。
 - データレイク設定は、ナビゲーションペインで [リージョン] を選択します。次に、「リージョン」テーブルで「リージョン」を選択します。
 - サブスクライバーの場合は、ナビゲーションペインで [サブスクライバー] を選択します。次に、「マイサブスクライバー」テーブルでサブスクライバーを選択します。

サブスクライバーがテーブルに表示されない場合は、ページの右上隅の AWS リージョンセレクターを使用して、サブスクライバーを作成したリージョンを選択します。表には、現在のリージョンに関してのみの既存のサブスクライバーのみが表示されます。

3. [編集] を選択します。
4. [タグ] セクションを展開します。Tags セクションには、現在リソースに割り当てられているすべてのタグが一覧表示されます。
5. 次のいずれかを実行します。
 - 既存のタグキーにタグ値を追加するには、タグキーの横にある Value ボックスに値を入力します。
 - 既存のタグキーを変更するには、タグの横にある [削除] を選択します。次に、[新しいタグを追加] をクリックします。表示されるキーボックスに、新しいタグキーを入力します。値 ボックスに、必要に応じて関連するタグ値を入力します。
 - 既存のタグ値を変更するには、その値を含む [値] ボックスで [X] を選択します。次に、値ボックスにその新しいタグの値を入力します。

- 既存のタグ値を削除するには、その値を含む [値] ボックスで [X] を選択します。
- 既存のタグ (タグキーとタグ値の両方) を削除するには、タグの横の [削除] を選択します。

リソースには、最大 50 個のタグを含めることができます。タグキーには最大 128 文字を含めることができます。タグ値は最大 256 文字を含めることができます。文字には、文字、数字、スペース、または記号 (_ . : / = + - @) を使用できます。

6. タグの編集を完了したら、保存 を選択します。

API

リソースのタグをプログラムで編集すると、既存のタグが新しい値に上書きされます。したがって、タグを編集する最適な方法は、タグキー、タグ値、またはその両方を編集するかどうかによって異なります。タグキーを編集するには、[現在のタグを削除](#)して、[新しいタグを追加](#)します。

タグキーに関連付けられているタグ値のみを編集または削除するには、Security Lake API の [TagResource](#) オペレーションを使用して既存の値を上書きします。AWS Command Line Interface (AWS CLI) を使用している場合は、[tag-resource](#) コマンドを実行します。リクエストで、タグ値を編集または削除するリソースの Amazon リソースネーム (ARN) を指定します。

タグ値を編集するには、tagsパラメータを使用して、タグ値を変更したいタグキーを指定します。キーの新しいタグ値も指定します。例えば、次の AWS CLI コマンドは、指定されたサブスクライバーに割り当てられたタグキー `On-Premises` の `Environment` タグ値を Cloud からに変更します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake tag-resource \  
--resource-arn arn:aws:securitylake:us-  
east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab \  
--tags key=Environment,value=On-Premises
```

コードの説明は以下のとおりです。

- `resource-arn` は、サブスクライバの ARN を指定します。
- `Environment` は、変更するタグ値に関連付けられているタグキーです。
- `On-Premises` は、指定したタグキー (`Environment`) に使用する新しいタグ値です。

タグキーからタグ値を削除するには、tagsパラメーターのキーのvalue引数の値を指定しないでください。以下に例を示します。

```
$ aws securitylake tag-resource \  
--resource-arn arn:aws:securitylake:us-east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab \  
--tags key=Owner,value=
```

オペレーションが正常に実行された場合は、Security Lake は空の HTTP 200 レスポンスを返します。それ以外の場合、Security Lake は HTTP 4xx またはオペレーションが失敗した理由を示す 500 レスポンスを返します。

Amazon Security Lake リソースのタグのレビュー

Amazon Security Lake リソースのタグ (タグキーとタグ値の両方) は、Security Lake コンソールまたは Security Lake API を使用して確認できます。

Console

Security Lake コンソールを使用してリソースのタグを確認するには、次の手順に従います。

リソースのタグを確認するには

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/>を開きます。
2. タグを確認するリソースのタイプに応じて、次のいずれかのリンクを実行します。
 - データレイク設定は、ナビゲーションペインで [リージョン] を選択します。リージョンテーブルでリージョンを選択し、[編集] を選択します。次に、[タグ] セクションを展開します。
 - サブスクライバーの場合は、ナビゲーションペインで [サブスクライバー] を選択します。次に、「マイサブスクライバー」テーブルで、サブスクライバーの名前を選択します。

サブスクライバーがテーブルに表示されない場合は、ページの右上隅の AWS リージョンセレクターを使用して、サブスクライバーを作成したリージョンを選択します。表には、現在のリージョンに関してのみの既存のサブスクライバーのみが表示されます。

Tags セクションには、現在リソースに割り当てられているすべてのタグが一覧表示されます。

API

既存のリソースのタグをプログラムで取得して確認するには、Security Lake API の [ListTagsForResource](#) オペレーションを使用します。リクエストで、`resourceArn`パラメータを使用して、リソースの Amazon リソースネーム (ARN) を指定します。

AWS Command Line Interface (AWS CLI) を使用している場合は、[list-tags-for-resource](#) コマンドを実行し、`resource-arn`パラメータを使用してリソースの ARN を指定します。以下に例を示します。

```
$ aws securitylake list-tags-for-resource --resource-arn arn:aws:securitylake:us-east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab
```

前の例では、`arn:aws:securitylake:us-east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab`は既存のサブスクライバーの ARN です。

オペレーションが正常に実行された場合は Security Lake はtags配列を返します。配列内の各オブジェクトは、現在リソースに割り当てられているタグ (タグキーとタグ値の両方) を指定します。以下に例を示します。

```
{
  "tags": [
    {
      "key": "Environment",
      "value": "Cloud"
    },
    {
      "key": "CostCenter",
      "value": "12345"
    },
    {
      "key": "Owner",
      "value": ""
    }
  ]
}
```

ここでEnvironment、CostCenter、Ownerは、リソースに割り当てられるタグキーです。Cloudは、Environmentタグキーに関連付けられているタグ値です。

12345は、CostCenterタグキーに関連付けられているタグ値です。Ownerタグキーには、関連するタグ値はありません。

Amazon Security Lake リソースからのタグの削除

Amazon セキュリティレイクリソースからタグを削除するには、セキュリティレイクコンソールまたはセキュリティレイク API を使用できます。

Important

リソースからタグを削除すると、リソースへのアクセスに影響を与える可能性があります。タグを削除する前に、タグを使用してリソースへのアクセスを制御する可能性のある AWS Identity and Access Management (IAM) ポリシーを確認してください。

Console

Security Lake コンソールを使用して、リソースから 1 つ以上のタグを削除するには、以下のステップに従います。

リソースからタグを削除する

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. タグを削除するリソースのタイプに応じて、次のいずれかを実行します。
 - データレイク設定は、ナビゲーションペインで [リージョン] を選択します。次に、「リージョン」テーブルで「リージョン」を選択します。
 - サブスクライバーの場合は、ナビゲーションペインで [サブスクライバー] を選択します。次に、「マイサブスクライバー」テーブルでサブスクライバーを選択します。

サブスクライバーがテーブルに表示されない場合は、ページの右上隅の AWS リージョンセレクターを使用して、サブスクライバーを作成したリージョンを選択します。表には、現在のリージョンに関してのみの既存のサブスクライバーのみが表示されます。
3. [編集] を選択します。
4. [タグ] セクションを展開します。Tags セクションには、現在リソースに割り当てられているすべてのタグが一覧表示されます。
5. 次のいずれかを実行します。

- タグに対しタグ値のみを削除するには、削除する値を含む [値] ボックスで [X] を選択します。
 - タグのタグキーとタグ値の両方を (ペアで) 削除するには、削除するタグの横にある [削除] を選択します。
6. リソースから追加のタグを削除するには、削除するタグを追加するたびに前の手順を繰り返します。
 7. タグの削除を完了したら、[Save (保存)] を選択します。

API

リソースから 1 つ以上のタグをプログラムから削除するには、Security Lake API の [UntagResource](#) オペレーションを使用します。リクエストで、resourceArn パラメーターを使用して、タグを削除するリソースの Amazon リソースネーム (ARN) を指定します。tagKeys パラメーターを使用して、削除するタグのタグキーを指定します。複数のタグを削除するには、削除する各タグの tagKeys パラメーターと引数をアンパサンド (&) で区切って追加します (例: tagKeys=key1&tagKeys=key2)。リソースから特定のタグ値 (タグキーではない) のみを削除するには、タグを削除する代わりに [タグを編集](#) します。

AWS Command Line Interface (AWS CLI) を使用している場合は、[untag-resource](#) コマンドを実行して、リソースから 1 つ以上のタグを削除します。resource-arn パラメータには、タグを削除するリソースの ARN を指定します。tag-keys パラメーターを使用して、削除するタグのタグキーを指定します。たとえば、次のコマンドは、指定したサブスクライバーからタグ (Environment タグキーとタグ値の両方) を削除します。

```
$ aws securitylake untag-resource \  
--resource-arn arn:aws:securitylake:us-  
east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab \  
--tag-keys Environment
```

ここで、resource-arn はタグを削除するサブスクライバの ARN を指定し、Environment は削除するタグのタグ キーです。

リソースから複数のタグを削除するには、追加の各タグ キーを tag-keys パラメーターの引数として追加します。以下に例を示します。

```
$ aws securitylake untag-resource \  
--resource-arn arn:aws:securitylake:us-  
east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab \  
--tag-keys Environment,AnotherTag
```

```
--tag-keys Environment Owner
```

オペレーションが正常に実行された場合は、Security Lake は空の HTTP 200 レスポンスを返します。それ以外の場合、Security Lake は HTTP 4xx またはオペレーションが失敗した理由を示す 500 レスポンスを返します。

Security Lake の問題のトラブルシューティング

Amazon Security Lake の使用中に問題が発生した場合は、次のトラブルシューティングリソースを使用します。

以下のトピックでは、データレイクのステータス、Lake Formation、Amazon Athena でのクエリ、IAM に関連して発生する可能性のあるエラー AWS Organizations や問題のトラブルシューティングに関するアドバイスを提供します。ここに記載されていない問題が見つかった場合は、このページのFeedbackボタンを使用して報告できます。

Security Lake の使用に伴って問題が発生した場合は、以下のトピックを参照してください。

トピック

- [データレイクのステータスのトラブルシューティング](#)
- [Lake Formation 問題のトラブルシューティング](#)
- [Amazon Athena でのクエリのトラブルシューティング](#)
- [Organizations の問題のトラブルシューティング](#)
- [Amazon Security Lake アイデンティティとアクセスのトラブルシューティング](#)

データレイクのステータスのトラブルシューティング

Security Lake コンソールの問題ページには、データレイクに影響を与えている問題の概要が表示されます。例えば、組織の CloudTrail 証跡を作成していない場合、Security Lake は AWS CloudTrail 管理イベントのログ収集を有効にできません。問題ページには、過去 14 日間に発生した問題が表示されます。各問題の説明と推奨される修復手順を確認できます。

問題の概要にプログラムでアクセスするには、Security Lake API の [ListDataLakeExceptions](#) オペレーションを使用できます。を使用している場合は AWS CLI、[list-data-lake-exceptions](#) コマンドを実行します。regions パラメータでは、us-east-1 米国東部 (バージニア北部) リージョンなど、1 つ以上のリージョンコードを指定して、それらのリージョンに影響する問題を確認できます。regions パラメータを含めない場合、すべてのリージョンに影響する問題が返されます。リージョンコードのリストについては、「AWS 全般のリファレンス」の「[Amazon Security Lake エンドポイント](#)」を参照してください。

例えば、次の AWS CLI コマンドは、us-east-1 および eu-west-3 リージョンに影響を与えている問題を一覧表示します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake list-data-lake-exceptions \  
--regions "us-east-1" "eu-west-3"
```

Security Lake ユーザーに問題やエラーを通知するには、Security Lake API の [CreateDataLakeExceptionSubscription](#) オペレーションを使用します。ユーザーは、E メール、Amazon Simple Queue Service (Amazon SQS) キューへの配信、AWS Lambda 関数への配信、またはサポートされている別のプロトコルを通じて通知を受けることができます。

例えば、次の AWS CLI コマンドは、指定されたアカウントに Security Lake 例外の通知を SMS 配信で送信します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake create-data-lake-exception-subscription \  
--notification-endpoint "123456789012" \  
--exception-time-to-live 30 \  
--subscription-protocol "sms"
```

例外サブスクリプションの詳細を表示するには、[GetDataLakeExceptionSubscription](#) オペレーションを使用できます。例外サブスクリプションを更新するには、[UpdateDataLakeExceptionSubscription](#) オペレーションを使用できます。例外サブスクリプションを削除して通知を停止するには、[DeleteDataLakeExceptionSubscription](#) オペレーションを使用できます。

Lake Formation 問題のトラブルシューティング

次の情報は、Security Lake と AWS Lake Formation データベースまたはテーブルを操作するとき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。Lake Formation のトラブルシューティングに関するその他のトピックについては、「AWS Lake Formation デベロッパーガイド」の「[トラブルシューティング](#)」セクションを参照してください。

テーブルが見つからない

サブスクライバーを作成しようとするとき、このエラーが表示されることがあります。

このエラーを解決するには、リージョンにソースを既に追加していることを確認します。Security Lake サービスのプレビューリリース時にソースを追加した場合は、サブスクライバーを作成する前にソースを再度追加する必要があります。ソースの追加についての詳細は、「[Security Lake でのソース管理](#)」を参照してください。

400 AccessDenied

[カスタムソースを追加](#)して CreateCustomLogSource API を呼び出すと、このエラーが表示されることがあります。

このエラーを解決するには、Lake Formation の権限を確認してください。API を呼び出す IAM ロールには、Security Lake データベースの Create table 権限が必要です。詳細については、「AWS Lake Formation 開発者ガイド」の「[Lake Formation コンソールと名前付きリソース メソッドを使用したデータベース アクセス許可の付与](#)」を参照してください。

SYNTAX_ERROR: 1:8 行目:カラムのないリレーションからの SELECT * は許可されていません

Lake Formation で初めてソーステーブルをクエリするときに、このエラーが表示されることがあります。

このエラーを解決するには、にサインインしたときに使用している IAM ロールにアクセスSELECT許可を付与します AWS アカウント。SELECT権限を付与する方法については、『AWS Lake Formation 開発者ガイド』の「[Lake Formation コンソールと名前付きリソースメソッドを使用したテーブル権限の付与](#)」を参照してください。

Security Lakeは、呼び出し元のプリンシパル ARN をレイクフォーメーションデータレイク管理者に追加できませんでした。現在のデータレイク管理者には、もはや存在しない無効なプリンシパルが含まれている可能性があります。

Security Lake を有効にするとき、または をログソース AWS のサービス として追加するときに、このエラーが表示されることがあります。

このエラーを解決するには、以下の手順を実行します。

1. Lake Formation コンソール (<https://console.aws.amazon.com/lakeformation/>) を開きます。
2. 管理ユーザーとしてサインインする
3. ナビゲーションペインの [許可] で [管理ロールとタスク] を選択します。
4. [データレイク管理者] セクションで、[管理者を選択] を選択します。
5. 「IAM で見つからない」というラベルの付いたプリンシパルをクリアし、[保存] を選択します。
6. Security Lake操作をもう一度試してください。

Lake Formation を使用した Security Lake CreateSubscriber は、承認してもらうための新しい RAM リソース共有の招待を作成しませんでした

Security Lake で Lake Formation サブスクライバーを作成する前に、[Lake Formation バージョン 2 またはバージョン 3 のクロスアカウントデータ共有](#)でリソースを共有していた場合、このエラーが表示されることがあります。これは、Lake Formation バージョン 2 およびバージョン 3 のクロスアカウント共有が、複数のクロスアカウントアクセス許可の付与を 1 つの AWS RAM リソース共有にマッピングすることで AWS、RAM リソース共有の数を最適化するためです。

リソース共有名がサブスクライバーの作成時に指定した外部 ID であり、リソース共有 ARN が CreateSubscriber 応答の ARN と一致することを確認してください。

Amazon Athena でのクエリのトラブルシューティング

次の情報を使用して、Security Lake S3 バケットに保存されているオブジェクトを Athena を使用してクエリする際に発生する可能性のある一般的な問題を診断して修正します。Athena のトラブルシューティングに関するその他のトピックについては、「Amazon Athena ユーザーガイド」の「[Athena でのトラブルシューティング](#)」セクションを参照してください。

クエリを実行しても、データレイク内の新しいオブジェクトは返されません。

Security Lake の S3 バケットにそれらのオブジェクトが含まれていても、Athena クエリはデータレイク内の新しいオブジェクトを返さない場合があります。これは、Security Lake を無効にしてから再度有効にした場合に発生することがあります。その結果、AWS Glue パーティションは新しいオブジェクトを適切に登録しない可能性があります。

このエラーを解決するには、以下の手順を実行します。

1. <https://console.aws.amazon.com/lambda/> で AWS Lambda コンソールを開きます。
2. ナビゲーションバーのリージョンセレクタで、Security Lake が有効になっているのに Athena クエリが結果を返さないリージョンを選択します。
3. ナビゲーションペインから関数を選択し、ソースバージョンに応じて次のリストから関数を選択します。
 - Source version 1 (OCSF 1.0.0-rc.2) - SecurityLake_Glue_Partition_Updater_Lambda_#region> 関数。

- Source version 2 (OCSF 1.1.0) – AmazonSecurityLakeMetastoreManager_#region> 関数。
4. [設定] タブで、[トリガー] を選択します。
 5. 関数の横にあるオプションを選択し、[編集] を選択します。
 6. [トリガーを有効化] を選択し、[保存] を選択します。これにより、機能の状態が [有効] に変わります。

AWS Glue テーブルにアクセスできない

クエリアクセスサブスクライバーは、Security Lake データを含む AWS Glue テーブルにアクセスできない場合があります。

まず、[クロスアカウントテーブル共有セットアップ \(サブスクライバーステップ\)](#) で説明されている手順を実行していることを確認します。

購読者がまだアクセスできない場合は、次の手順に従ってください。

1. <https://console.aws.amazon.com/glue/> で AWS Glue コンソールを開きます。
2. ナビゲーションペインで、[データカタログ] と [カタログ設定] を選択します。
3. リソースベースのポリシーを使用して AWS Glue テーブルにアクセスするアクセス許可をサブスクライバーに付与します。リソースベースのポリシーの作成については、AWS Glue デベロッパーガイドの「[AWS Glueのリソースベースのポリシーの例](#)」を参照してください。

Organizations の問題のトラブルシューティング

次の情報は、Security Lake および AWS Organizations を使用するときが発生する可能性のある一般的な問題の診断と修正に役立ちます。Organizations トラブルシューティングに関するその他のトピックについては、「AWS Organizations ユーザーガイド」の「[トラブルシューティング](#)」セクションを参照してください。

CreateDataLake 操作を呼び出しているときに、アクセス拒否エラーが発生しました。アカウントは組織の委任管理者アカウントまたはスタンドアロンアカウントでなければなりません。

委任された管理者アカウントが所属する組織を削除し、Security Lake コンソールまたは CreateDataLake API を使用してそのアカウントを使用して Security Lake を設定しようとする、このエラーが表示されることがあります。

このエラーを解決するには、別の組織の委任管理者アカウントまたはスタンドアロンアカウントを使用してください。

Amazon Security Lake アイデンティティとアクセスのトラブルシューティング

次の情報は、Security Lake と IAM の使用に伴って発生する可能性がある一般的な問題の診断や修復に役立ちます。

Security Lake でアクションを実行することが認可されていない

からアクションを実行する権限がないと AWS Management Console 通知された場合は、管理者に連絡してサポートを依頼する必要があります。管理者は、認証情報を提供した担当者です。

以下のエラー例は、mateojackson IAM ユーザーがコンソールを使用して架空の *subscriber* に関する詳細情報を表示しようとしているが、架空の SecurityLake:*GetSubscriber* 許可がないという場合に発生します。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
YOURSERVICEPREFIX:GetWidget on resource: my-example-widget
```

この場合、Mateo は、SecurityLake:*GetSubscriber* アクションを使用して *subscriber* 情報へのアクセスが許可されるように、管理者にポリシーの更新を依頼します。

管理ポリシーを超えてアクセス許可を拡張したい

サブスクライバーまたはカスタムログソース APIs によって作成されたすべての IAM ロールは、AmazonSecurityLakePermissionsBoundary 管理ポリシーによってバインドされます。管理ポ

リシーを超えてアクセス許可を拡張する場合は、ロールのアクセス許可の境界から管理ポリシーを削除できます。ただし、dataLakes とサブスクライバーの Security Lake APIs の変更を操作する場合は、IAM が IAM ロールを変更するためにアクセス許可の境界をアタッチする必要があります。

iam: PassRole を実行する権限がない

iam:PassRole アクションを実行する権限がないというエラーが表示された場合は、ポリシーを更新して Security Lake にロールを渡すことができるようにする必要があります。

一部の AWS のサービスでは、新しいサービスロールまたはサービスにリンクされたロールを作成する代わりに、既存のロールをそのサービスに渡すことができます。そのためには、サービスにロールを渡す権限が必要です。

以下の例のエラーは、marymajor という IAM ユーザーがコンソールを使用して Security Lake でアクションを実行しようする場合に発生します。ただし、このアクションをサービスが実行するには、サービスロールから付与された権限が必要です。メアリーには、ロールをサービスに渡す許可がありません。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

この場合、Mary のポリシーを更新してメアリーに iam:PassRole アクションの実行を許可する必要があります。

サポートが必要な場合は、AWS 管理者にお問い合わせください。サインイン認証情報を提供した担当者が管理者です。

自分の 以外のユーザーに Security Lake 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:

- Security Lake でこれらの機能がサポートされるかどうかを確認するには、「[Security Lake と IAM の連携方法](#)」を参照してください。

- 所有 AWS アカウント する 全体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 する別の 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「[サードパーティーが所有する へのアクセスを提供する AWS アカウント](#)」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。

Security Lakeの価格の決定方法

Amazon Security Lake の価格は、データ取り込みとデータ変換の 2 つのディメンションに基づいています。Security Lake は他の AWS のサービス と連携してデータを保存および共有するため、これらのアクティビティには別途料金がかかる場合があります。

Security Lake がサポート AWS リージョン する AWS アカウント ので初めてログ収集を有効にすると、そのアカウントは自動的に Security Lake の 15 日間の無料トライアルに登録されます。無料トライアル中も、他のサービスから料金が発生する可能性があります。

Security Lake の料金の背後にある方法論を理解するには、Amazon [Security Lake pricing](#)--> の動画をご覧ください。

データ取り込み

これらのコストは、取り込まれた AWS CloudTrail ログやその他の AWS のサービス ログやイベント (Amazon Route 53 リゾルバーのクエリログ、AWS Security Hub 結果、Amazon VPC フローログ) の量から発生します。

データ変換

これらのコストは、Security Lake が[Security Lake でサイバーセキュリティスキーマフレームワーク \(OCSF\) を開く](#)スキーマに正規化し、Apache Parquet 形式に変換する AWS のサービス ログとイベントのボリュームから発生します。

関連サービスのコスト

セキュリティデータレイクにデータ AWS のサービス を保存および共有するために、他の から発生する可能性のあるコストを以下に示します。

- Amazon S3 — Security Lake アカウントの Amazon S3 バケットの維持およびそこへのデータの保存、およびセキュリティとアクセスコントロールのバケットの評価とモニタリングから導出されます。詳細については、「[Amazon S3 の料金](#)」を参照してください。
- Amazon SQS — これらのコストは、メッセージ配信用の Amazon SQS キューを作成することで発生します。詳細については、「[Amazon SQS の料金](#)」を参照してください。
- Amazon EventBridge — これらの費用は、Amazon EventBridge がサブスクリプションエンドポイントにオブジェクト通知を送信することで発生します。詳細については、「[Amazon EventBridge の料金](#)」を参照してください。

- AWS Glue – 月額コストは、ギガバイトあたりの AWS サービスから取り込まれたログとイベントデータのボリュームによって決まります。データは Amazon Simple Storage Service に保存され、標準の Amazon S3 料金が適用されます。Security Lake は、ユーザーに代わって他の AWS のサービスもオーケストレーションします。セキュリティデータレイクの一部として設定された AWS サービスとリソースには、個別の料金が発生します。[AWS Glue](#)、[Amazon EventBridge](#)、[AWS Lambda](#)[Amazon SQS](#)、[Amazon Simple Notification Service](#) の料金を参照してください。Security Lake からデータをクエリし、クエリ結果を保存することで発生するコストは、お客様の負担となります。

Security Lake からデータをクエリし、クエリ結果を保存することでサブスクライバーが負担する費用は、サブスクライバーの負担となります。

コストと補助サービスの詳細なリストについては、[「Security Lake の料金」](#)を参照してください。

Security Lakeの使用状況と推定コストを確認する。

Amazon Security Lake コンソールの「使用状況」ページでは、現在の Security Lake の使用状況だけでなく、将来の使用状況とコストの見積もりを確認できます。現在 15 日間の無料トライアルに参加している場合、トライアル中の使用状況は、無料トライアルの終了後に Security Lake を使用する際のコストを見積もるのに役立ちます。Security Lake の料金の概要については、[を参照してください](#)[Security Lakeの価格の決定方法](#)。詳細情報とコストの例については、[「Amazon Security Lake の料金」](#)を参照してください。

Security Lake では、推定使用コストを米ドルでレポートし、現在の AWS リージョンにのみ適用されます。コストには、組織内のすべてのアカウントの Security Lake の使用がカバーされ、Open Cybersecurity Schema Framework (OCSF) や Apache Parquet 形式への変換が含まれます。ただし、予測コストには、Amazon Simple Storage Service (Amazon S3) や AWS Glueなど、Security Lake が連携する他のサービスのコストは含まれていません。

使用状況ページでは、使用状況とコストのデータを表示する期間を選択します。デフォルトの期間は過去 1 暦日です。コスト予測を確認するには、Security Lake の使用日が 1 日以上ある必要があります。

ページの上部には、すべてのアカウントの予測コストが表示されます。これは、選択した時間枠内の実際の使用量に基づいて AWS リージョン、今後 30 暦日間における現在の Security Lake の予測コストです。実際の使用量と予測コストには、組織内のすべてのアカウントが反映されます。

ページの残りの部分では、使用量とコストのデータが次の 2 つの表に分かれています。

- ソース別の使用量とコスト — データソース別の現在の Security Lake の使用状況と、選択した期間における実際の使用状況に基づく今後 30 日間の推定使用量とコストです。実際の使用量、予測使用量、予測コストには、組織内のすべてのアカウントが反映されます。ソースを選択すると、分割パネルが開き、どのアカウントがそのソースからログとイベントを生成したかが表示されます。各アカウントの分割パネルには、そのソースからの実際の使用状況と、予測される使用量とコストの両方が表示されます。
- アカウント別の使用量とコスト — アカウントごとの現在の Security Lake の使用状況と、選択した期間における実際の使用状況に基づく今後 30 日間の推定使用量とコストが表示されます。アカウントを選択すると、分割パネルが開き、そのアカウントの使用量に貢献したソースが表示されます。貢献しているソースごとに、実際の使用状況と予測される使用量とコストの両方が分割パネルに表示されます。

Security Lake に特定のソースを追加していない場合でも、サポートされているすべての AWS データソースが上記の表に表示されます。無料トライアルに参加している場合は、すべての AWS ソースを追加して、ログとイベントのフルセットのコスト見積もりを取得することをお勧めします。AWS ソースを追加する手順については、「」を参照してください[Security Lake AWS のサービスでのからのデータ収集](#)。カスタムソースは使用量やコストの計算には含まれません。

Security Lake コンソールで使用量とコストデータを確認するには、次のステップに従います。

Security Lake の使用状況と予測コストを確認するには (コンソール)

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ページの右上隅にある AWS リージョン セレクターを使用して、使用状況とコストを確認するリージョンを選択します。
3. ナビゲーションペインで、「設定」、「使用状況」の順に選択します。
4. 使用状況とコストデータを表示したい期間を選択します。デフォルトは過去 1 日です。
5. [データソース別] または [アカウント別] タブを選択して、使用状況とコストを詳細に確認します。

Security Lake のリージョンとエンドポイント

セキュリティレイクでサポートされているリージョンと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「[Amazon Security Lake エンドポイント](#)」を参照してください。

Security Lake は、サポートされているすべての AWS リージョンで有効にすることをお勧めします。このように設定することで、Security Lake はアクティブに使用されていないリージョンでも、許可されていないアクティビティや異常なアクティビティを検出して調査できます。

Security Lake を無効にする

Amazon Security Lake を無効にすると、Security Lake は AWS ソースからのログとイベントの収集を停止します。既存の Security Lake 設定と AWS アカウント で作成されたリソースは保持されます。さらに、AWS Lake Formation テーブルや AWS CloudTrail ログの機密データなど AWS のサービス、他の に保存または公開したデータは引き続き使用できます。Amazon Simple Storage Service (Amazon S3) バケットに保存されたデータは、[Amazon S3 ストレージライフサイクル](#)に従って引き続き使用できます。

Security Lake コンソールの設定ページから Security Lake を無効にすると、Security Lake が現在有効になっているすべての AWS リージョンで AWS ログとイベントの収集が停止します。コンソールの [リージョン] ページを使用して、特定のリージョンのログ収集を停止できます。Security Lake API とは、リクエストで指定したリージョンのログ収集 AWS CLI も停止します。

との統合を使用し AWS Organizations、アカウントが複数の Security Lake アカウントを一元管理する組織の一部である場合、委任 Security Lake 管理者のみが、それ自体とメンバーアカウントの Security Lake を無効にできます。ただし、組織を離れると、そのメンバーアカウントのログ収集は停止します。

組織の Security Lake を無効にしても、このページに記載されている無効化手順に従えば、委任管理者の指定は保持されます。Security Lake を再度有効にする前に、委任管理者を再度指定する必要はありません。

カスタムソースの場合、Security Lake を非アクティブ化するときに、Security Lake コンソールの外部にある各ソースを無効にする必要があります。統合を無効にしないと、ソースの統合により、引き続き Amazon S3 にログが送信されます。さらに、サブスクライバーの統合を無効にする必要があります。無効にしないと、サブスクライバーは引き続き Security Lake からのデータを使用できます。カスタムソースまたはサブスクライバーの統合を削除する方法の詳細については、各プロバイダーのドキュメントを参照してください。

Important

Security Lake を再度有効にする前に AWS Glue データベースを削除して、クエリが正しく動作するようにする必要があります。

Security Lake が再び有効になると、新しいデータレイク Amazon S3 バケットが作成され、データがこの新しい S3 バケットに収集されます。以前に AWS Glue テーブルを削除したことがある場合は、新しい AWS Glue テーブルセットが作成されます。

Security Lake を無効にする前に収集されたすべてのデータは、古い Amazon S3 バケットに残ります。古いデータをクエリする場合は、Amazon S3 Sync コマンドを使用して新しいバケットに移動する必要があります。詳細については、[「コマンドリファレンス」の「Sync コマンド AWS CLI」](#)を参照してください。

このトピックでは、Security Lake コンソール、Security Lake API、または `aws` を使用して Security Lake を無効にする方法について説明します AWS CLI。

Console

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ナビゲーションペインで [Settings] の [General] を選択します。
3. [Security Lakeを無効にする] を選択します。
4. 確認を求められたら、**Disable**と入力し、Disable (無効化) を選択します。

API

Security Lake をプログラムで無効にするには、Security Lake API の [DeleteDataLake](#) オペレーションを使用します。を使用している場合は AWS CLI、[delete-data-lake](#) コマンドを実行します。リクエストで、regions リストを使用して、Security Lake を無効にする各リージョンのリージョンコードを指定します。リージョンコードのリストについては、「AWS 全般のリファレンス」の [「Amazon Security Lake エンドポイント」](#)を参照してください。

を利用する Security Lake デプロイでは AWS Organizations、組織の委任 Security Lake 管理者のみが、組織内のアカウントの Security Lake を無効にすることができます。

たとえば、次の AWS CLI コマンドは、ap-northeast-1 および eu-central-1 リージョンで Security Lake を無効にします。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake delete-data-lake \  
--regions "ap-northeast-1" "eu-central-1"
```

「Amazon Security Lake ユーザーガイド」のドキュメント履歴

次のテーブルに、Amazon Security Lake の前回のリリース以後に行われたドキュメントの重要な変更を示します。このドキュメントの更新に関する通知を受け取るには、RSS フィードにサブスクライブできます。

ドキュメントの最終更新日：2025年4月24日

変更	説明	日付
更新された機能 - サービスにリンクされたロール	Security Lake は、データレイクの作成中に <code>AWSServiceRoleForSecurityLakeResourceManagement SLR</code> を自動的に作成するようになりました。詳細については、「 考慮事項 」をご参照ください。	2025年4月24日
大幅に書き換えられたトピック - AWS 統合	Security Lake と特定の の統合を指定するコンテンツを更新しました AWS のサービス。詳細については、「 AWS のサービス 統合 」を参照してください。	2025年3月31日
更新された機能 - 複数のアカウントの管理	Security Lake コンソールで、アカウントが組織に加わったときの自動有効化設定の管理がサポートされるようになりました。詳細については、「 コンソールでの新しいアカウント設定の編集 」を参照してください。	2025年3月10日
更新された機能 - AWS WAF ログ内のデータ保護	Security Lake アカウントのウェブ ACL で有効になって	2025年2月17日

いる場合のデータ保護のサポートが追加されました。詳細については、[AWS WAF「Security Lake のログ」](#)を参照してください。

[新機能 - VPC エンドポイントのサポートの追加](#)

Security Lake がと統合 AWS PrivateLink され、VPC エンドポイントがサポートされるようになりました。AWS PrivateLink 統合の詳細については、[「Amazon Security Lake とインターフェイス VPC エンドポイント \(AWS PrivateLink \)」](#)を参照してください。

2025 年 2 月 4 日

[新機能](#)

Security Lake は、Security Lake 内のデータを分析するための OpenSearch Service ダイレクトクエリをサポートするようになりました。詳細については、[OpenSearch Service との統合](#)」を参照してください。

2024 年 12 月 1 日

[新しいサービスにリンクされたロール](#)

新しいサービスにリンクされたロールを追加しました [AWSServiceRoleForSecurityLakeResourceManagement](#) 。このサービスにリンクされたロールは、継続的なモニタリングとパフォーマンスの向上を実行するアクセス許可を Security Lake に提供し、レイテンシーとコストを削減できます。

2024 年 11 月 14 日

リージョナルな可用性

Security Lake が、AWS GovCloud (米国東部) および AWS GovCloud (米国西部) で利用可能になりました AWS リージョン。Security Lake が現在利用可能なリージョンの完全なリストについては、「[AWS 全般のリファレンス](#)」の「[Amazon Security Lake エンドポイント](#)」を参照してください。

2024 年 6 月 10 日

既存のマネージドポリシーの更新

[SecurityLakeServiceLinkedRole](#) ポリシーの AWS 管理ポリシーに AWS WAF アクションを追加しました。追加のアクションにより、Security Lake で AWS WAF ログソースとして有効になっている場合、Security Lake はログを収集できます。

2024 年 5 月 22 日

新しい AWS ログソース

Security Lake は、[AWS WAF ログ](#)を AWS ログソースとして追加しました。は、エンドユーザーがアプリケーションに送信するウェブリクエストをモニタリングする AWS WAF のに役立ちます。

2024 年 5 月 22 日

既存のマネージドポリシーの更新

[AmazonSecurityLakePermissionsBoundary](#) ポリシーに SID アクションを追加しました。

2024 年 5 月 13 日

[既存のマネージドポリシーの更新](#)

[AmazonSecurityLake MetastoreManager](#) ポリシーを更新して、データレイク内のメタデータを削除できるメタデータクリーンアップアクションを追加しました。

2024 年 3 月 27 日

[新しいソースバージョン](#)

[ロールのアクセス許可を更新](#)して、新しいデータソースバージョンからデータを取り込みます。

2024 年 2 月 29 日

[新しい AWS ログソース](#)

Security Lake は [EKS 監査ログをログソース](#)として追加しました。AWS EKS 監査ログは、Amazon Elastic Kubernetes Service 内の EKS クラスター内の潜在的に疑わしいアクティビティを検出するのに役立ちます。

2024 年 2 月 29 日

[既存のマネージドポリシーの更新](#)

新しいAmazonSecurityLakeMetastoreManagerV2 [ロールiam:PassRole](#) を許可するようにポリシーを更新し、Security Lake がデータレイクコンポーネントをデプロイまたは更新できるようにしました。

2024 年 2 月 23 日

新しいマネージドポリシー

新しい[AWS マネージドポリシー](#)、Amazon Security Lake Metastore Manager ポリシーを追加しました。このポリシーは、Security Lake がデータレイク内のメタデータを管理するアクセス許可を付与します。

2024 年 1 月 23 日

リージョナルな可用性

Security Lake が AWS リージョン、アジアパシフィック (大阪)、カナダ (中部)、欧州 (パリ)、欧州 (ストックホルム) で利用可能になりました。Security Lake が現在利用可能なリージョンの完全なリストについては、「AWS 全般のリファレンス」の「[Amazon Security Lake エンドポイント](#)」を参照してください。

2023 年 10 月 26 日

新しい特徴

[クエリアクセス権を持つサブスクライバーの特定の設定を編集](#)できるようになりました。AWS アカウントの [Security Lake リソースにタグを割り当てる](#)こともできます。

2023 年 7 月 20 日

新しいマネージドポリシー	Security Lake は、新しい AWS 管理ポリシー である AmazonSecurityLake Administrator ポリシーを追加しました。このポリシーにより、プリンシパルが Security Lake のすべてのアクションに完全にアクセスすることが許可される、管理者許可が付与されます。	2023 年 5 月 30 日
一般提供	Security Lake は一般的にご利用いただけるようになりました。	2023 年 5 月 30 日
新機能	Security Lake は Amazon CloudWatch にメトリクスを送信 するようになりました。	2023 年 5 月 4 日
リージョナルな可用性	Security Lake が、AWS リージョンアジアパシフィック (シンガポール)、欧州 (ロンドン)、南米 (サンパウロ) で利用可能になりました。	2023 年 3 月 22 日
新機能	Security Lake コンソールを使用して Security Lake を有効にし、使用を開始するときに、Security Lake がユーザーに代わって AWS Identity and Access Management (IAM) ロールを作成するようになりました。	2023 年 2 月 15 日
初回リリース	これは、Amazon Security Lakeユーザーガイドの初回リリースです。	2022 年 11 月 29 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。