aws

ユーザーガイド

Amazon Security Lake

Amazon Security Lake: ユーザーガイド

Table of Contents

Amazon Security Lake とは何ですか?

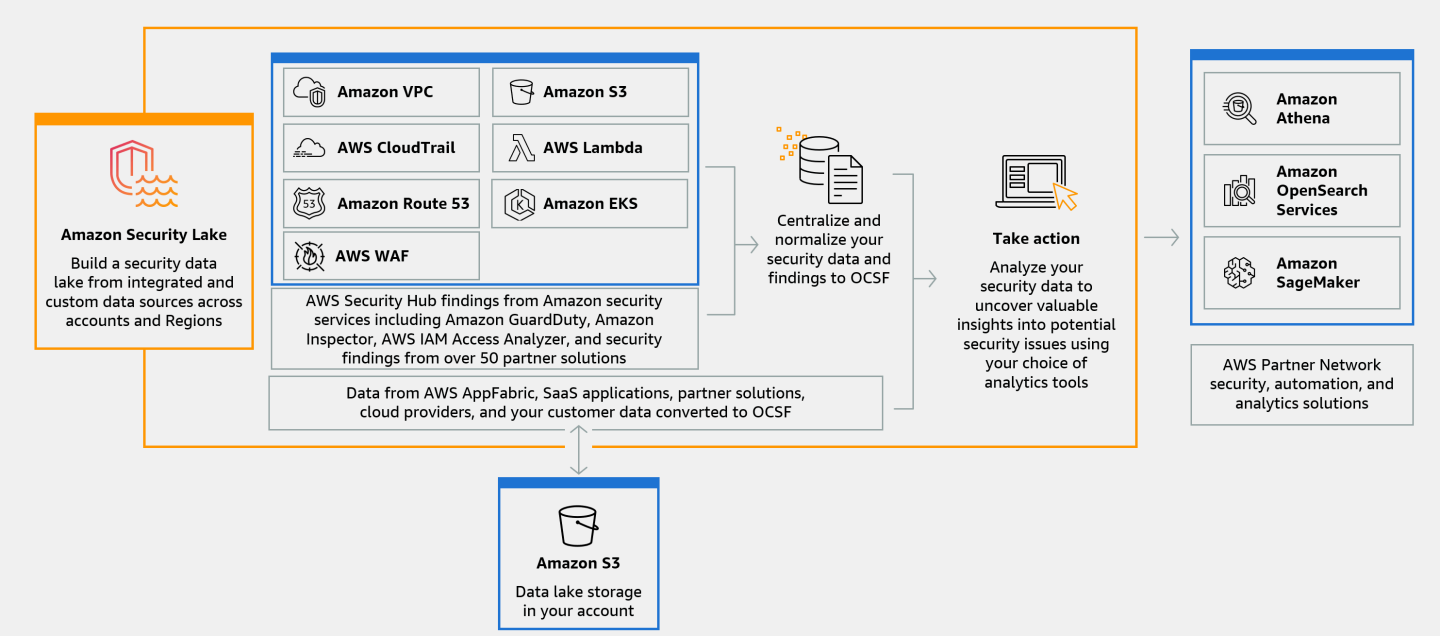
Amazon Security Lake は、完全マネージド型のセキュリティデータレイクサービスです。Security Lake を使用すると、 AWS 環境、SaaS プロバイダー、オンプレミス、クラウドソース、およびサードパーティーソースのセキュリティデータを、 に保存されている専用のデータレイクに自動的に一元化できます AWS アカウント。Security Lake はセキュリティデータの分析に役立つため、組織全体のセキュリティ体制をより完全に把握できます。Security Lake を使用すると、ワークロード、アプリケーション、データの保護を強化することもできます。

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 サービス や第三者サービスからのセキュリティ関連の口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 から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 サービス およびサードパーティーのサービスは、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、サードパーティーサービスなど、複数のソースからセキュリティログ AWS サービス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、Amazon 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 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 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 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、「Amazon Security Lake リージョンおよびエンドポイント」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、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 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で構築するツール AWS」を参照してください。

関連サービス

Security Lake AWS 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 EventBridge 、オブジェクトがデータレイクに書き込まれるときにサブスクライバーに通知するために を使用します。
- AWS Glue – Security Lake は AWS Glue クローラーを使用して 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 Data Catalog 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。

Security Lake は、次の に加えて、カスタムソースからデータを収集します AWS 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、「からのデータ収集 AWS サービス」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

AWS リージョンロールアップリージョンにデータを提供する 1 つまたは複数のユーザー。

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、AWS Lake Formationデータレイクにデータを提供するソースごとに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。それは、AWSとさまざまなセキュリティドメインにわたるセキュリティ業界のリーダーたちによって開発されました。Security Lake は、AWS サービス から収集したログとイベントを 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 つ以上の寄与リージョンからのセキュリティ ログとイベントを統合する AWS リージョン。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 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 サービスでも第三者のサービスでもかまいません。

Amazon Security Lake の開始方法

このセクションでは、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。データレイクの設定方法と収集の設定方法について説明します。Security Lake は、 を通じて、 AWS Management Console またはプログラムで有効にして使用できます。どの方法を使用する場合でも、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。

> **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フィルをサポートしていません。

AWS アカウント 初期設定

にサインアップする AWS アカウント

がない場合は AWS アカウント、次の手順を実行して作成します。

にサインアップするには AWS アカウント

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

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

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

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

のセキュリティ保護 AWS アカウントのルートユーザー

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

   ルートユーザーを使用してサインインする方法については、AWS サインイン ユーザーガイドのルートユーザーとしてサインインするを参照してください。
2. ルートユーザーの多要素認証 (MFA) を有効にします。

   手順については、「 ユーザーガイド」の AWS アカウント 「ルートユーザーの仮想MFAデバイスを有効にする (コンソール) IAM」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 Identity Center を有効にします。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「AWS IAM Identity Centerの有効化」を参照してください。
2. IAM Identity Center で、ユーザーに管理アクセス権を付与します。

   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、「 ユーザーガイド」の「デフォルト で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AWS IAM Identity Center 」を参照してください。

管理アクセス権を持つユーザーとしてサインインする

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

IAM Identity Center ユーザーを使用してサインインする方法については、「 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。 AWS サインイン

追加のユーザーにアクセス権を割り当てる

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

   手順については、「AWS IAM Identity Center ユーザーガイド」の「権限設定を作成する」を参照してください。
2. グループにユーザーを割り当て、そのグループにシングルサインオンアクセス権を割り当てます。

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

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 して、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、「を使用した複数のアカウントの管理 AWS Organizations」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。

Amazon 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。
- どのリージョンでも Security Lake を初めて有効にすると、アカウントに `AWSServiceRoleForSecurityLake` というサービスリンクロールが作成されます。このロールには、 AWS サービス ユーザーに代わって他の を呼び出し、セキュリティデータレイクを運用するためのアクセス許可が含まれています。サービスにリンクされたロールの仕組みの詳細については、「 ユーザーガイド」の「サービスにリンクされたロールの使用IAM」を参照してください。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。
- 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。

コンソールの開始方法

このチュートリアルでは、 を使用して Security Lake を有効化および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 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、カスタムソースからのデータ収集 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. で Security Lake コンソールを開きますhttps://console.aws.amazon.com/securitylake/。

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. [ログとイベントソースを選択] で、次のオプションのいずれかを選択します。

   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した場合、 CloudTrail - S3 データイベントは取り込みに含まれません。これは、大量の CloudTrail - S3 データイベントを取り込むと、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。このソースを取り込むには、[特定の AWS ソースの取り込み] オプションを選択します。

   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。

> **Important**
>
> 指定し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。

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. サービスアクセス では、新しいIAMロールを作成するか、Security Lake にソースからデータを収集してデータレイクに追加するアクセス許可を付与する既存の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。
8. [Next (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. 最初のステップでロールアップリージョンを選択した場合は、サービスアクセス で新しい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レプリケートするアクセス許可を付与する既存のIAMロールを使用します。 IAM
4. [Next (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の概要、サブスクライバーに関する情報、および問題について説明します。

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。アクセス許可レベルの詳細については、「Lake Formation ペルソナ」およびIAM「 アクセス許可リファレンス」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、「Amazon Security Lake におけるサブスクライバー管理」を参照してください。

プログラムによる開始

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では、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が可能です。または、 AWS コマンドラインツール、つまり AWS Command Line Interfaceまたは AWS Tools for PowerShell を使用して Security Lake AWS SDKs 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は、データレイクを設定するには AWS Identity and Access Management 、 (IAM) ロールをいくつか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、これらのIAMロールを作成する必要はありません。

次のアクションを 1 つ以上実行するIAM場合は、 でロールを作成する必要があります (各アクションのIAMロールに関する詳細情報を表示するには、リンクを選択してください) 。

- 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 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 がソースから受信した未加工の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 の CreateDataLakeオペレーションを使用しますAPI。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 ONEZONE_IA 60 日後に 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 GLACIER 30 日後に 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"}]}}]' \
--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て、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、 を使用する場合は update-data-lake コマンド AWS CLIを使用します。CreateDataLake オペレーションは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 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、カスタムソースからのデータ収集 を参照してください。

プログラムで 1 つ以上のコレクションソースを定義するには、Security Lake の CreateAwsLogSourceオペレーションを使用しますAPI。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 の CreateDataLakeオペレーションを使用しますAPI。Security Lake を既に有効にしていて、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。

例えば、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、その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 の ListDataLakesオペレーションを使用して API、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時に問題または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。アクセス許可レベルの詳細については、「Lake Formation ペルソナ」およびIAM「 アクセス許可リファレンス」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、「Amazon Security Lake におけるサブスクライバー管理」を参照してください。

を使用した複数のアカウントの管理 AWS Organizations

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。委任された管理者は、 AWS リージョン Security Lake が有効になっているすべての で、組織全体のログとイベントを収集できます (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) 。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 Organization 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 APIs を使用して委任された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場合があります。完全な機能を確保するには、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。

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の 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、「Amazon Security Lake における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は、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

委任された 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ための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. 管理を別のアカウント に委任 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、委任された 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

委任された管理者をプログラムで指定するには、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。リクエストでは、`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。

例えば、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 の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。委任管理者として、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。

例えば、次の AWS CLI コマンドは、Security Lake と、新しい組織アカウントの Amazon Route 53 リゾルバークエリログ、検出 AWS Security Hub 結果、および 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ように Security Lake を有効にして設定することが含まれます。詳細については、「からのデータ収集 AWS サービス」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。

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。

例えば、次の AWS CLI コマンドは委任された Security Lake 管理者を削除します。

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

委任された管理者の指定を保持し、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 delete-data-lake-organization-configuration コマンド AWS CLIを使用します。組織のこれらの設定を変更できるのは、委任された管理者のみです。

例えば、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スラッシュ (\) 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 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 が一時停止、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。

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と便利です。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. で Security Lake コンソールを開きますhttps://console.aws.amazon.com/securitylake/。
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 の GetDataLakeSourcesオペレーションを使用しますAPI。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、指定し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. で Security Lake コンソールを開きますhttps://console.aws.amazon.com/securitylake/。
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、「Amazon 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 の UpdateDataLakeオペレーションを使用しますAPI。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。

例えば、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"}]}}]'
```

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。

> **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の送信先に正しくレプリケートされるように、Security Lake では、「カスタムソースの取り込みに関する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 () で 2 つの異なるロールを作成する必要がありますIAM。

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

> **Note**
>
> Security Lake コンソールを使用するときに、Security Lake はこれらの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。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。

IAM データレプリケーションの 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アクセス許可を Amazon S3 に付与します。

これらのアクセス許可を付与するには、プレフィックス で始まる IAMロールを作成しSecurityLake、次のサンプルポリシーをロールにアタッチします。Security Lake でロールアップリージョンを作成するときは、ロールの Amazon リソースネーム (ARN) が必要です。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。

IAM AWS Glue パーティションを登録する 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. で Lake Formation コンソールを開きますhttps://console.aws.amazon.com/lakeformation/。
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のIAMユーザーとロールで、作成したAmazonSecurityLakeMetaStoreManagerIAMロールを選択し、 の保存を選択します。

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. で Security Lake コンソールを開きますhttps://console.aws.amazon.com/securitylake/。
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 で新しい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レプリケートするアクセス許可を付与する既存のIAMロールを使用します。 IAM
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 の UpdateDataLakeオペレーションを使用しますAPI。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、その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-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 (を使用する場合は AWS CLIcreate-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. で Security Lake コンソールを開きますhttps://console.aws.amazon.com/securitylake/。
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

ロールアップリージョンをプログラムで設定するには、Security Lake の UpdateDataLakeオペレーションを使用しますAPI。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 の両方を寄与リージョンus-east-2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"}]}}]'
```

Amazon Security Lake でのソース管理

ソースは、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)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のカスタムソースなど、AWS サービスさまざまなソースからログ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 AWS リージョン 内の AWS アカウント 内の Amazon Simple Storage Service (Amazon S3) バケットにソース 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 は各ソースからのデータを別々のAWS Lake Formationテーブルセットに整理します。

トピック

- からのデータ収集 AWS サービス
- カスタムソースからのデータ収集

からのデータ収集 AWS サービス

Amazon Security Lake では、 AWS 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 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) および Apache Parquet 形式に自動的に変換します。

> **Tip**
>
> 上記の 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 管理イベントを除き、これらのサービスのログ記録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

を Security Lake のソース AWS サービス として追加するには、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`とリソース範囲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 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。

CloudTrail イベントログ

AWS CloudTrail は、、 SDK、コマンドラインツール AWS Management Console、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 また、 では、 をサポートするサービスの AWS APIs を呼び出したユーザーとアカウント CloudTrail、呼び出し元のソース IP アドレス、および呼び出しが発生した日時を特定することもできます。 AWS SDKs 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。

Security Lake は、S3 および Lambda. CloudTrail management イベント、S3 CloudTrail データイベント、および Lambda データイベント CloudTrail の管理イベントおよびデータイベントに関連するログを収集できます。これらのログは、Security Lake の S3 つの異なるソースです。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、 内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。 CloudTrail データイベントは、データプレーンオペレーションとも呼ばれ、 内のリソースで実行されたリソースオペレーションを表示します AWS アカウント。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、読み取りおよび書き込み CloudTrail 管理イベントを収集するマルチリージョン組織の証跡が少なくとも 1 CloudTrail 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。

マルチリージョント証跡は、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の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。

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

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 管理イベントとデータイベントは、イベントの独立した重複ストリーム CloudTrail を介して から直接消費されます。

Security Lake は CloudTrail イベントを管理したり、既存の CloudTrail 設定に影響を与えたりしません。 CloudTrail イベントへ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「イベント履歴を含む CloudTrail 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 に正規化する方法に関する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。

GitHub CloudTrail イベントの 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)

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) クラスターで実行されている 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。

Security Lake は、監査ログの独立した重複ストリームを介して、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 クラスター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、GitHub Amazon EKS 監査ログイベント (v1.1.0) の 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。

Route 53 Resolver 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 に正規化する方法に関する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。

GitHub Route 53 ログの 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 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 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 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) (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 を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 に正規化する方法に関する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。

GitHub Security Hub の検出結果の 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)

VPC Flow Logs

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。

GitHub VPC フローログの 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 ログ

Security Lake のログソース 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。 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 AWS WAF です。ログに記録された情報には、 がリソースから AWS 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 から直接ログを消費します。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。 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の詳細については、「 AWS WAF デベロッパーガイド」の「 AWS WAF の仕組み」を参照してください。

> **Important**
>
> のリソースタイプとして Amazon CloudFront ディストリビューションを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。

をソース AWS サービス として追加する

をソース AWS サービス として追加すると、Security Lake はセキュリティログとイベントからの収集を自動的に開始します。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 サービス として追加する方法について説明します。カスタムソースを追加する手順については、カスタムソースからのデータ収集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 サービス する を選択し、 の設定 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**
>
> 指定し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。

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み、データインジェストに使用した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. [Enable (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 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。オプションで、ソースの範囲を特定の `accounts`または特定の に制限できます`sourceVersion`。

> **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、 `accounts`パラメータを指定しない場合、 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することで解決できます。

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

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やリソース、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、指示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. [適用] 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動作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 から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。

ソース AWS 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 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、ソース AWS サービス としてネイティブにサポートされている の削除にのみ使用できます。カスタムソースの削除については、カスタムソースからのデータ収集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 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 に制限できます`sourceVersion`。

> **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、 `accounts`パラメータを指定しない場合、 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、読みやすくす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。

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を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"
```

カスタムソースからのデータ収集

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 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) に、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するロールを作成します。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 クローラーを設定します。クローラーは、 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。

Security Lake にカスタムソースを追加するには、次の要件を満たしている必要があります。

1. 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、一意の各 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に書き込むことを許可する IAMロールを作成します。

> **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 1.1。

2. 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。
3. スキーマ – Parquet 形式のオブジェクト内の各レコードには、同じOCSFイベントクラスを適用する必要があります。

カスタムソースの取り込みのベストプラクティス

効率的なデータ処理とクエリを容易にするために、Security Lake にカスタムソースを追加するときは以下のベストプラクティスに従うことをお勧めします。

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。

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

  サンプルパーティションはです。aws-security-data-lake-*us-west-2-lake-uid*/ext/*custom-source-name*/region=*us-west-2*/accountId=*123456789012*/eventDay=*20230428*/
- ソースバージョンをカスタムソースに追加した場合、パーティションデータパスは の形式になります。

  bucket-name/ext/*custom-source-name*/*custom-source-version*/region=*us-west-2*/accountId=*123456789012*/eventDay=*20230428*/

  ソースバージョンを含むサンプルパーティションは ですaws-security-data-lake-*us-west-2-lake-uid*/ext/*custom-source-name*/*custom-source-version*/region=*us-west-2*/accountId=*123456789012*/eventDay=*20230428*/。

次のリストは、 パーティションで使用されるパラメータを示しています。

- bucket-name— セキュリティレイク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。

- source-version – カスタムソースのソースバージョン。
- region - データが書き込まれる AWS リージョン 宛先。
- accountId –ソースパーティション内のレコードが関係する AWS アカウント ID。
- eventDay— 8 文字の文字列 () YYYYMMDD としてフォーマットされた、イベントが発生した日付。

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。

Parquetの設定

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。

ソート

Parquet 形式の各オブジェクト内では、データのクエリにかかる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。

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ている場合、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のみのステップ)

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。

アクセス許可を確認するには、 IAMを使用して、IAMID にアタッチされているIAMポリシーを確認します。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 に進みますカスタムソースの追加。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、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 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のみのステップ)

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ルし、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ロールを作成した後、カスタムソースを追加するには、ロールの Amazon リソースネーム (ARN) が必要です。

`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管理の で暗号化されている場合は AWS KMS key、ロールとKMSキーポリシーに次のポリシーも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}}"
    ]
}
```

カスタムソースの追加

AWS Glue クローラーを呼び出すIAM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. で Security Lake コンソールを開きますhttps://console.aws.amazon.com/securitylake/。
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タイプを記述する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

プログラムでカスタムソースを追加するには、Security Lake の CreateCustomLogSourceオペレーションを使用しますAPI。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` – ソースがログとイベントをデータレイクに書き込むために使用する 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 は AWS Glue 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、「 デベロAWS Glue 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。

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。

Console

1. で Security Lake コンソールを開きますhttps://console.aws.amazon.com/securitylake/。
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、Security Lake の DeleteCustomLogSourceオペレーションを使用しますAPI。 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のみに制限します。

次の例では、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
```

Amazon Security Lake における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、「Amazon 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。

- データアクセス — オブジェクトが Security Lake データレイクに書き込まれると、ソースの新しい Amazon S3 オブジェクトがサブスクライバーに通知されます。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 CreateSubscriberAPI S3 accessTypes のパラメータで識別されます。
- クエリアクセス — サブスクライバーは、Amazon Athena などのサービスを使用して S3 AWS Lake Formation バケット内の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 CreateSubscriberAPI LAKEFORMATION accessTypes のパラメータで識別されます。

サブスクライバーは、サブスクライバーを作成したときに選択したのソースデータにのみアクセスできます。 AWS リージョン 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、 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 1 人あたりに追加できるソースの最大数は 10 です。AWS ソースとカスタムソースの組み合わせでもかまいません。

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。

データにアクセスできる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。

トピック

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 データにアクセスできる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のみのステップ)

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 EventBridge アクセス許可を Amazon 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。

データにアクセスできる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 から HTTPS EventBridge エンドポイントに通知を送信する場合は、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 では、新しい IAM ロールを作成するか、API 送信先を呼び出してオブジェクト通知を正しいエンドポイントに送信するアクセス許可を付与 EventBridgeする既存の IAM ロールを使用します。

      新しい IAM ロールの作成については、 EventBridge 「API 送信先 を呼び出す 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、「Amazon 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

プログラムでデータアクセスを持つサブスクライバーを作成するには、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 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 。

```
$ 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 (\) の行連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、CreateSubscriber通知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、 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、UpdateSubscriber通知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 update-subscriber-notification コマンドを実行します。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

```
{
  "source": "aws.s3",
  "time": "2021-11-12T00:00:00Z",
  "account": "123456789012",
  "region": "ca-central-1",
  "resources": [
    "arn:aws:s3:::example-bucket"
  ],
  "detail": {
    "bucket": {
      "name": "example-bucket"
    },
```

```
    "object": {
      "key": "example-key",
      "size": 5,
      "etag": "b57f9512698f4b09e608f4f2a65852e5"
    },
    "request-id": "N4N7GDK58NMKJ12R",
    "requester": "securitylake.amazonaws.com"
  }
}
```

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、「Amazon 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

サブスクライバーのデータ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方法を更新するには、UpdateSubscriber通知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。

データサブスクライバーを削除する。

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、サブスクライバーへの今後の通知を停止するには、Security Lake API DeleteSubscriberの通知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-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。

> **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。

クエリアクセスのある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。

トピック

- アクセス許可の確認
- 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のみのステップ)
- Lake Formation 管理者権限を付与

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。

- `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のみのステップ)

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。

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 それが作成された 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、「Amazon 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

プログラムでクエリアクセスを持つサブスクライバーを作成するには、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。

クエリアクセス権限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、「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。
   - コンソールと 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 で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. 共有テーブルへのリソース リンクを作成する – サブスクライバーは、 AWS Lake Formation (コンソールを使用する場合) または AWS Glue (API/AWS CLIを使用する場合) で共有 Lake Formation 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テーブルに誘導します。次のいずれかのアクセス方法を選択します。
   - コンソールと については AWS CLI、「 デベロッパーガイド」の「共有データカタログテーブルへのリソースリンクの作成」を参照してください。 AWS Lake Formation
   - API の場合は、 AWS Glue CreateTable API を呼び出します。サブスクライバーは、リソースリンクテーブルを保存するために CreateDatabase API を使用して一意のデータベースを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。

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、クエリアクセス権限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には」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 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、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、前の AWS RAM リソース共有が自動的に削除されます。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。サブスクラ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. 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。

7. [保存] を選択します。

API

クエリアクセス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、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。
- 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 – このアクションでは、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、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。

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、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。このパラメーターは省略することも、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 クエリ GitHub ディレクトリ」を参照してください。

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_TABL
```

```
   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 のデータイベント
- `s3_data_1_0` –S3 CloudTrail のデータイベント
- `route53_1_0` - Amazon Route 53 Resolver クエリログ
- `sh_findings_1_0` - AWS Security Hub 検出結果
- `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、*20230301* (2023 年 3 月 1 日) 以降に記録された CloudTrail ソース IP 192.0.2.1 からのアクティビティを、*us-east*-1 の *cloud_trail_mgmt_1_0* テーブルに一覧表示しますDB_Region。

```
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
    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 **192.0.2.1**日以降のソース IP からのすべての CloudTrail アクティビティのリスト **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
    WHERE eventDay >= '20230301'
    AND src_endpoint.ip = '192.0.2.1'
    ORDER BY time desc
    LIMIT 25
```

例: テーブル内の**192.0.2.1**過去 30 日間のソース IP からのすべての CloudTrail アクティビティのリスト **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
    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 データに対するクエリの例

AWS CloudTrail は、 のユーザーアクティビティと API の使用状況を追跡します AWS サービス。サブスクライバーは CloudTrail データをクエリして、次のタイプの情報を学習できます。

データクエリの例を次に示します CloudTrail 。

過去 7 日間の AWS 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
```

192.0.2.1 過去 7 日間のソース IP 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 日間の失敗した 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 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。

Route 53 Resolver クエリログの例をいくつか挙げます。

CloudTrail 過去 7 日間の 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 の検出結果に関するクエリ

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 = 'AwsS3Bucket')
   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 フローログのクエリ例

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。

Amazon VPC フローログのクエリの例は次のとおりです。

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

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、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。このパラメーターは省略することも、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 クエリ GitHub ディレクトリ」を参照してください。

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 のデータイベント
- `s3_data_2_0` - S3 CloudTrail のデータイベント
- `route53_2_0` - Amazon Route 53 Resolver クエリログ

- `sh_findings_2_0` - AWS Security Hub 検出結果
- `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、*20230301* (*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、*us-west*-2 のテーブル *vpc_flow_2_0* に一覧表示しますDB_Region。

```
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 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accountid。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。

クエリを実行すると time_dt 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　time_dt3 月 1 日以降に 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 **192.0.2.1**日以降のソース IP からのすべての CloudTrail アクティビティのリスト **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
    WHERE eventDay >= '20230301'
    AND src_endpoint.ip = '192.0.2.1'
    ORDER BY time desc
    LIMIT 25
```

例: テーブル内の**192.0.2.1**過去 30 日間のソース IP からのすべての CloudTrail アクティビティのリスト **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
    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 オブザーバブルのクエリ

オブザーバビリティは、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。

オブザーバビリティ内の特定の要素をクエリすることで、データセットを特定のユーザー名、リソース UIDs、IPs、ハッシュ、その他の IOC タイプの情報などに制限できます。

これは、オブザーバブル配列を使用して VPC フローと IP 値 '172.01.02.03' を含む Route53 テーブル全体のログをクエリするクエリの例です。

```
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 データのクエリ

AWS CloudTrail は、 のユーザーアクティビティと API の使用状況を追跡します AWS サービス。サブスクライバーは CloudTrail データをクエリして、次のタイプの情報を学習できます。

データに対するクエリの例を次に示します CloudTrail 。

過去 7 日間の AWS 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
```

192.0.2.1 過去 7 日間のソース IP 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 日間の失敗した 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 リゾルバークエリログのクエリ

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。

Route 53 Reesolver クエリログのクエリ例を次に示します。

CloudTrail 過去 7 日間の 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 の検出結果のクエリ

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 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 = 'AwsS3Bucket')
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(cvs)
WHERE cvs.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 フローログのクエリ

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。

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 監査ログのクエリ

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 ログに直接監査ログと診断ログを提供します。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。

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」からリクエストを更新する

```
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
```

AWS WAF v2 ログのクエリ

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 WAF v2 ログのクエリの例をいくつか示します。

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 日間に国ごとにグループ化された AWS 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、 AWS リージョン 希望する時間だけ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よく保存できるように管理するには、データの保持設定を行います。Security Lake はデータをオブジェクトとして Amazon Simple Storage Service (Amazon S3) バケットに保存するため、保持設定は Amazon S3 ライフサイクル設定に対応します。これらの設定を行うことで、希望する Amazon S3 ストレージクラスと、S3 オブジェクトが別のストレージクラスに移行または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できます。Amazon S3 ライフサイクル設定の詳細については、Amazon Simple Storage Service ユーザーガイドの「ストレージライフサイクルの管理」を参照してください。

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、 AWS リージョン 特定のすべての S3 オブジェクトをデータレイクに書き込まれてから 30 日後に S3 標準 IA ストレージクラスに移行するように選択できます。デフォルトの Amazon S3 ストレージ クラスは S3 Standardです。

> **Important**
>
> 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。デフォルト保持モードで S3 オブジェクトロックを有効にすると、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。

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。保持設定を行わない場合、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。つまり、S3 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。

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構成するには、Security Lake API CreateDataLakeの操作を使用してください。を使用している場合は AWS CLI、コマンドを実行します。create-data-lakelifecycleConfiguration必要な保存設定を以下のようにパラメータに指定します。

- 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、us-east-1以下のコマンドはリージョンのセキュリティレイクを有効にします。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

保持設定をプログラムで更新するには Security Lake API UpdateDataLakeの操作を使用し、を使用している場合はコマンドを実行します。 AWS CLIupdate-data-lakeリクエストでは、`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。

たとえば、 AWS CLI us-east-1次のコマンドはリージョンのデータ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","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の設定方法については、を参照してください。ロールアップリージョンの設定

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)

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 のパブリックソースコードは でホストされますGitHub。

Security Lake は、ネイティブにサポートされている から取得したログとイベント AWS サービス を OCSF スキーマに自動的に変換します。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 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.vendor_name	metadata.product.feature.name	class_name	metadata.version
CloudTrail Lambda データイベント	CloudTrail	AWS	Data	API Activity	1.0.0-rc.2
CloudTrail 管理イベント	CloudTrail	AWS	Management	API Activity, Authentication , または Account Change	1.0.0-rc.2
CloudTrail S3 データイベント	CloudTrail	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 Flow Logs	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.vendor_name	metadata.product.feature.name	class_name	metadata.version
CloudTrail Lambda データイベント	CloudTrail	AWS	Data	API Activity	1.1.0
CloudTrail 管理イベント	CloudTrail	AWS	Management	API Activity, Authentication , または Account Change	1.1.0
CloudTrail S3 データイベント	CloudTrail	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) ProductName の値と一致します	AWS Security Finding Format (ASFF) CompanyName の値と一致します	ASFF featureName の値と一致 ProductFields	Vulnerability Finding, Compliance Finding, or Detection Finding	1.1.0
VPC Flow Logs	Amazon VPC	AWS	Flowlogs	Network Activity	1.1.0

ソース	metadata.product.name	metadata.product.vendor_name	metadata.product.feature.name	class_name	metadata.version
EKS 監査ログ	Amazon EKS	AWS	Elastic Kubernetes 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 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 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 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 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。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)
- データは Apache Parquet 形式で届きます

サブスクライバー統合には、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、 からソースデータを直接クエリできる次のプロパティがあります。 AWS Lake Formation

次のセクションでは、 AWS サービス Security Lake がどの と統合されるか、および各統合の仕組みについて説明します。

との統合 AWS AppFabric

統合タイプ: ソース

AWS AppFabric は、Software as a Service (SaaS) アプリケーションを組織全体に接続するノーコードサービスです。そのため、IT チームとセキュリティチームは、標準スキーマと中央リポジトリを使用してアプリケーションを管理および保護できます。

Security Lake が AppFabric 結果を受信する方法

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

前提条件

Security Lake に AppFabric 監査ログを送信する前に、OCSF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 結果を Security Lake に送信する

上記の前提条件を完了した後に Security Lake に AppFabric 監査ログを送信するには、両方のサービスを有効にし、Security Lake のカスタムソース AppFabric として を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、カスタムソースからのデータ収集を参照してください。

Security Lake での AppFabric ログの受信を停止する

AppFabric 監査ログの受信を停止するには、Security Lake コンソール、Security Lake 、または を使用してAPI、 AWS CLI をカスタムソース AppFabric として削除します。手順については、カスタムソースの削除 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、 のサービス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 OpenSearch サービスです AWS クラウド。Service OpenSearch 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 サービスダッシュボード

OpenSearch Service を Security Lake と統合したら、サーバーレス OpenSearch サービス取り込みを介してさまざまなソースから OpenSearch Service 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 サービス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。

Service Ingestion OpenSearch が OpenSearch Service 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 との統合 QuickSight

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、どこで作業していても、相手に easy-to-understand インサイトを提供できます。Amazon 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組み合わせます。Amazon QuickSight は、意思決定者にインタラクティブなビジュアル環境で情報を探索して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 で Amazon Security Lake データを視覚化するには QuickSight、必要な AWS オブジェクトを作成し、Security Lake QuickSight 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、およびユーザーグループを Amazon にデプロイします。詳細な手順については、「Amazon との統合 QuickSight」を参照してください。

Amazon との統合 SageMaker

統合タイプ: サブスクライバー

Amazon SageMaker はフルマネージド型の機械学習 (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 モデルを迅速かつ確実に構築、トレーニング、デプロイできます。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、複数の統合開発環境 () で SageMaker ML ツールを利用できるようにしますIDEs。

SageMaker インサイト

SageMaker 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。 SageMaker 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 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 SageMaker」を参照してください。

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、統合された 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API。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

Studio で Amazon Bedrock の生成 AI 機能と自然言語入力を使用して、Security Lake SageMaker 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、調査からの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 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。

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チェックするのに役立ちます。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全体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。検出結果の中には、他の AWS のサービスやサードパーティーパートナーによって検出された問題に由来するものもあります。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 のすべての検出結果は、Security Finding JSON形式 () と呼ばれる標準形式を使用します。 AWS ASFF

Security Lake はSecurity Hub 結果を受け取り、それを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)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をソース AWS サービス として追加する。

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、Security Hub コンソール、Security Hub API、または を使用できます AWS CLI。

「 ユーザーガイド」の「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)」または「統合からの検出結果のフローの無効化 (Security Hub API、AWSCLI)」を参照してください。 AWS Security Hub

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。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)

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 スキーマ内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 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。サードパーティー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の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に、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アイデンティティに、関連するデータベースとテーブルに対するSELECTアクセス許可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
- Palo Alto Networks – XSOAR
- IBM – QRadar
- Query.AI – Query Federated Search
- SOC Prime
- Tego Cyber

Accenture – MxDR

統合タイプ: サブスクライバー、サービス

Accenture'sMxDR 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、分析とマネージド型の検出と対応 () が容易になりますMDR。

サービス統合として、Accenture 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

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 APIsなどのオープンなユーザーSQL、または Kibana と Superset がネイティブUIsに含まれている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。Cisco 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、スキーマへのOCSFスキーマ変換を実行し、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。柔軟で迅速なデプロイオプションは、IoT 、、IIoTおよび アセットで構成される拡張されたモノのインターネット (XIoT) BMSアセットをネットワーク内でxDome保護し、脅威の早期指標を自動的に検出するのに役立ちます。

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寄木細工形式で大規模に 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、ウェブアプリケーション、、APIsマイクロサービスでリアルタイムの脆弱性検出を提供する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

を使用してCribl Stream、Criblサポートされている任意のサードパーティーソースからOCSFスキーマ内の 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CrowdStrike Falcon Data Replicatorで からデータを取得し、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

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スキーマで 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: サブスクライバー

Datadog Cloud SIEM は、Security Lake 内のデータを含むクラウド環境に対するリアルタイムの脅威を検出し、 DevOps 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: サブスクライバー、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 機能のCAEスイートは、Security Lake 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

Devo

統合タイプ: サブスクライバー

の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: サブスクライバー、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 を実装するのにも役立ちます。

統合ドキュメンテーション

Eviden— AIsaac (旧 Atos)

統合タイプ: サブスクライバー

AIsaac MDR 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フローログを消費し、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

OCSF スキーマ内の IOCsから Security Lake への の検出を含むネットワークデータを統合することでExtraHop Reveal(x) 360、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FortiGateCNF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) は、重要なメタデータ属性を使用して、オブザーバビリティSIEM、、およびネットワークパフォーマンスモニタリングツールを強化します。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

IBM – QRadar

統合タイプ: サブスクライバー

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、 Kyndryl は分析目的で Security Lake から AWS CloudTrail Management Events 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、および既知の脅威と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の追加の分析ユースケースに使用できます。

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ソリューションに接続してカスタムタスクを構築し、お客様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OCSFスキーマに自動的に変換し、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、企業内で何が起こっているかについて、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と によってお客様のアカウントで生成されOmnis Cyber Intelligence、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、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で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。アカウントにデプロイされ、Amazon によってトリガーされSQS、New Relicライセンスキー New Relicを使用して にデータを送信します。

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 ID ログをOCSFスキーマの Security Lake に送信します。 OCSFスキーマOkta System Logsの 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で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、一貫したスキーマで監査アクティビティを実行し、認証、認可、アカウント変更、エンティティ変更に関連するレポートを生成できます。

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スキーマでクラウド検出とレスポンス (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 は、クラウドネイティブ環境VMsの 全体で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： Suscriber

Palo Alto Networks XSOAR は、 XSOARおよび Security Lake とのサブスクラ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: サブスクライバー

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スキーマおよび Parquet 形式で Security Lake に送信し、アカウントの変更を検出して対応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: サブスクライバー、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 テーブルに直接クエリを実行して、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、および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

InsightIDRSIEM/ Rapid7 XDR ソリューションである は、Security Lake にログを取り込んで、脅威を検出し、疑わしいアクティビティを調査することができます。

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

統合用のパートナー製品: 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: サブスクライバー

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: サブスクライバー

SentinelOne Singularity™ XDR プラットフォームは、Amazon Elastic Compute Cloud (Amazon)、Amazon Elastic Container Service (Amazon EC2) 、Amazon Elastic Kubernetes Service (Amazon ECS) など、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で実行されているエンドポイント、アイデンティティ、クラウドワークロードに対するリアルタイムの検出と対応を拡張しますEKS。

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 Lifecyle 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、クエリ用のOCSFスキーマで Security Lake に保存 Sentraされ、それ以降にストリーミングされるメタデータです。

統合ドキュメンテーション

SOC Prime

統合タイプ: サブスクライバー

SOC Prime は、Amazon OpenSearch Service および Amazon Athena を介して Security Lake と統合され、ゼロトラストマイルストーンに基づくスマートデータオーケストレーションと脅威ハンティングを容易にします。 は、セキュリティチームが大量のアラートを必要とせずに、脅威の可視性を高め、インシデントを調査するSOC Primeことを可能にします。OCSF スキーマ内の Athena と OpenSearch サービスに自動的に変換できる再利用可能なルールとクエリを使用して、開発時間を短縮できます。

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スキーマ内のデータをサブスクライブすることで、脅威の検出、調査、対応を高速化できます。

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、Security Lake からOCSFスキーマにデータを取り込んで、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、脅威の検出、調査、インシデント対応を高速化します。

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、 は Security Lake 統合Trellix XDR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スキーマ内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 環境の検出と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 に次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、OCSFスキーマ内の豊富なデータをオンプレミスおよびクラウドアセットから Security Lake に送信します。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

Zscaler Posture Control™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 は、セキュリティ検出結果をOCSFスキーマの Security Lake に送信します。

統合ドキュメンテーション

Amazon Security Lakeのセキュリティ

AWS では、クラウドセキュリティを最優先事項としています。AWS のユーザーは、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を利用できます。

セキュリティは、AWS とユーザーの間の責任共有です。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して説明しています。

- クラウドのセキュリティ - AWS は、AWS クラウド で AWS の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を担います。また、AWS は、ユーザー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 の一環として、サードパーティーの監査が定期的にセキュリティの有効性をテストおよび検証しています。Amazon Security Lakeに適用するコンプライアンスプログラムについては、「AWSコンプライアンスプログラムによる対象範囲サービス」の「 」を参照してください。
- クラウド内のセキュリティ - ユーザーの責任は、使用する AWS のサービスに応じて異なります。またお客様は、データの機密性、企業要件、適用法令と規制などのその他の要因に対しても責任を担います。

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。Security Lake リソースのモニタリングやセキュリティ確保に役立つ他の AWS サービスの使用方法についても説明します。

トピック

Amazon Security Lake 用の 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 サービス するのに役立つ です。IAM 管理者は、誰を認証 (サインイン) し、誰に Security Lake リソースの使用を承認する (アクセス許可を付与する) かを制御します。IAM は追加料金なしで AWS サービス 使用できる です。

トピック

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しておくと、管理者に適切な許可をリクエストするうえで役立ちます。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。次に、サービスユーザーのアクセス許可を変更するリクエストをIAM管理者に送信する必要があります。このページの情報を確認して、 の基本概念を理解してくださいIAM。会社で Security Lake を使用する方法の詳細については、IAM「」を参照してくださいAmazon Security Lake と の連携方法 IAM。

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の詳細について確認する場合があります。で使用できる Security Lake アイデンティティベースのポリシーの例

を表示するにはIAM、「」を参照してくださいAmazon Security Lakeのアイデンティティベースのポリシー例。

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ロールを引き受けることによって認証（ にサインイン AWS) される必要があります。

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

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

AWS プログラムで にアクセスする場合、 はソフトウェア開発キット (SDK) とコマンドラインインターフェイス (CLI) AWS を提供し、認証情報を使用してリクエストに暗号で署名します。 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。推奨される方法を使用してリクエストを自分で署名する方法の詳細については、「 IAMユーザーガイド」のAWS API「リクエストの署名」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、 AWS では、アカウントのセキュリティを高めるために多要素認証 (MFA) を使用する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「 ユーザーガイド」の「 での多要素認証 (MFA) AWS IAM の使用」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 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

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

フェデレーティ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 AWS サービス します。

フェデレーティッド 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 サービス を使用して にアクセスするユーザーです。フェデレーティッド 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て、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、「 ユーザーガイド」のIAM「Identity Center とはAWS IAM Identity Center 」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持つIAMユーザーを作成するのではなく、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ユーザーとの長期的な認証情報を必要とする特定のユースケースがある場合は、アクセスキーをローテーションすることをお勧めします。詳細については、「 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でアクセスキーを定期的にローテーションするIAM」を参照してください。

IAM グループは、IAMユーザーのコレクションを指定する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、 という名前のグループを作成しIAMAdmins、そのグループにIAMリソースを管理するアクセス許可を付与できます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユーザーには永続的な長期の認証情報がありますが、ロールでは一時認証情報が提供されます。詳細につ

いては、「 ユーザーガイド」のIAM「 (ロールの代わりに) ユーザーを作成する場合IAM」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。ユーザーと似ていますがIAM、特定のユーザーに関連付けられていません。IAM ロール を切り替える AWS Management Console ことで、 で ロールを一時的に引き受けることができます。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 を使用しますURL。ロールの使用方法の詳細については、「 ユーザーガイド」のIAM「ロールの使用IAM」を参照してください。

IAM 一時的な認証情報を持つ ロールは、以下の状況で役立ちます。

- フェデレーションユーザーアクセス – フェデレーティッド ID に許可を割り当てるには、ロールを作成してそのロールの許可を定義します。フェデレーティッド ID が認証されると、その ID はロールに関連付けられ、ロールで定義されている許可が付与されます。フェデレーションのロールの詳細については、「 ユーザーガイド」の「サードパーティー ID プロバイダーのロールの作成IAM」を参照してください。IAM Identity Center を使用する場合は、アクセス許可セットを設定します。ID が認証後にアクセスできる内容を制御するために、IAMIdentity Center はアクセス許可セットを のロールに関連付けますIAM。アクセス許可セットの詳細については、「AWS IAM Identity Center ユーザーガイド」の「アクセス許可セット」を参照してください。
- 一時的なIAMユーザーアクセス許可 – IAM ユーザーまたはロールは、 IAMロールを引き受けて、特定のタスクに対して異なるアクセス許可を一時的に引き受けることができます。
- クロスアカウントアクセス – IAMロールを使用して、別のアカウントのユーザー (信頼されたプリンシパル) が自分のアカウントのリソースにアクセスすることを許可できます。クロスアカウントアクセスを許可する主な方法は、ロールを使用することです。ただし、一部の では AWS サービス、 (ロールをプロキシとして使用する代わりに) ポリシーをリソースに直接アタッチできます。クロスアカウントアクセスのロールとリソースベースのポリシーの違いについては、「 ユーザーガイド」の「 でのクロスアカウントリソースアクセスIAMIAM」を参照してください。
- クロスサービスアクセス — 一部の は、他の の機能 AWS サービス を使用します AWS サービス。例えば、サービスで呼び出しを行うと、そのサービスが Amazon でアプリケーションを実行EC2したり、Amazon S3 にオブジェクトを保存したりするのが一般的です。サービスでは、呼び出し元プリンシパルの許可、サービスロール、またはサービスリンクロールを使用してこれを行う場合があります。
  - 転送アクセスセッション (FAS) – IAM ユーザーまたはロールを使用して でアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行

することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS サービス、ダウンストリームサービス AWS サービス へのリクエストのリクエストと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 サービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するための権限が必要です。FAS リクエストを行う際のポリシーの詳細については、「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 – サービスロールは、ユーザーに代わってアクションを実行するためにサービスが引き受ける IAMロールです。IAM 管理者は、 内からサービスロールを作成、変更、削除できますIAM。詳細については、「 ユーザーガイド」の「 にアクセス許可を委任するロールの作成 AWS サービスIAM」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 サービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント 、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示できますが、編集することはできません。

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

IAM ロールとIAMユーザーのどちらを使用するかについては、「 ユーザーガイド」の「 (ユーザーではなく) IAMロールを作成する場合IAM」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 AWS には、ポリシーを作成し、 AWS ID またはリソースにアタッチします。ポリシーは、アイデンティティまたはリソースに関連付けられているときにアクセス許可を定義する のオブジェクトです。 は、プリンシパル (ユーザー、ルートユーザー、またはロールセッション) AWS がリクエストを行うときに、これらのポリシー AWS を評価します。ポリシーでの権限により、リクエストが許可されるか拒否されるかが決まります。ほとんどのポリシーはJSONドキュメ

ント AWS として に保存されます。JSON ポリシードキュメントの構造と内容の詳細については、「 ユーザーガイド」のJSON「ポリシーの概要IAM」を参照してください。

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

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

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

アイデンティティベースのポリシー

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

リソースベースのポリシー

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

リソースベースのポリシーは、そのサービス内にあるインラインポリシーです。リソースベースのポリシーIAMでは、 の AWS 管理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 (ACLs)

アクセスコントロールリスト (ACLs) は、リソースへのアクセス許可を持つプリンシパル (アカウントメンバー、ユーザー、またはロール) を制御します。ACLs はリソースベースのポリシーに似ていますが、JSONポリシードキュメント形式を使用しません。

Amazon S3、 AWS WAF、および Amazon VPCは、 をサポートするサービスの例ですACLs。の詳細についてはACLs、Amazon Simple Storage Service デベロッパーガイドの「アクセスコントロールリスト (ACL) 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ます。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- アクセス許可の境界 – アクセス許可の境界は、アイデンティティベースのポリシーがIAMエンティティ (IAMユーザーまたはロール) に付与できるアクセス許可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) に対する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、 AWS アカウント ビジネスが所有する複数の をグループ化して一元管理するためのサービスです。組織内のすべての機能を有効にすると、サービスコントロールポリシー (SCPs) をアカウントの一部またはすべてに適用できます。は、各 を含むメンバーアカウントのエンティティのアクセス許可SCPを制限します AWS アカウントのルートユーザー。Organizations と の詳細についてはSCPs、「 AWS Organizations ユーザーガイド」の「サービスコントロールポリシー」を参照してください。
- セッションポリシー - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果としてセッションの権限は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合もありま

す。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。詳細については、「 ユーザーガイド」の「セッションポリシーIAM」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リクエストを許可するかどうか AWS を決定する方法については、「 ユーザーガイド」の「ポリシー評価ロジックIAM」を参照してください。

Amazon Security Lake と の連携方法 IAM

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。

IAM Amazon Security Lake で使用できる の機能

IAM 機能	Security Lake サポート
アイデンティティベースのポリシー	あり
リソースベースのポリシー	あり
ポリシーアクション	あり
ポリシーリソース	Yes
ポリシー条件キー	あり
ACLs	なし
ABAC (ポリシー内のタグ)	あり
一時的な認証情報	あり
プリンシパル権限	あり
サービスロール	いいえ
サービスリンクロール	あり

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 「 と連携する のサービスIAM」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

ID ベースのポリシーは、IAMユーザー、ユーザーのグループ、ロールなどの ID にアタッチできるJSONアクセス許可ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。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、「 ユーザーガイド」のIAM「ポリシーの作成IAM」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、許可または拒否されたアクションとリソース、およびアクションが許可または拒否される条件を指定できます。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は、「 ユーザーガイド」の「 IAMJSONポリシー要素のリファレンスIAM」を参照してください。

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、「Amazon Security Lakeのアイデンティティベースのポリシー例」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: はい

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

クロスアカウントアクセスを有効にするには、リソースベースのポリシーのプリンシパルとして、アカウント全体または別のアカウントのIAMエンティティを指定できます。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 にある場合 AWS アカウント、信頼されたアカウントのIAM管理者は、プリンシパルエンティティ (ユーザーまたはロール) にリソースへのアクセス許可も付与する必要があります。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、「 ユーザーガイド」の「 でのクロスアカウントリソースアクセスIAMIAM」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。

リソースの例は、Amazon リソースネーム (ARN) が の S3 バケットですarn:aws:s3:::aws-security-data-lake-{region}-{bucket-identifier}。この例では、 regionは Security Lake を有効に AWS リージョン した特定の 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、「 ユーザーガイド」の「アイデンティティベースのポリシーとリソースベースのポリシーIAM」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

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

JSON ポリシーの `Action`要素は、ポリシーでアクセスを許可または拒否するために使用できるアクションを記述します。ポリシーアクションの名前は通常、関連する AWS APIオペレーションと同じです。一致するAPIオペレーションを持たないアクセス許可のみのアクションなど、いくつかの例外があります。また、ポリシーに複数のアクションが必要なオペレーションもあります。これらの追加アクションは、依存アクションと呼ばれます。

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

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、Amazon Security Lakeのアイデンティティベースのポリシー例 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

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

Policy ResourceJSON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 では、サブスクライバー、および特定の AWS アカウント 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。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ARNs。

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、「サービス認証リファレンス」の「Amazon Security Lake で定義されるリソースタイプ」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、Amazon Security Lakeのアイデンティティベースのポリシー例 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

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

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

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

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

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

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、「Amazon Security Lakeのアイデンティティベースのポリシー例」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 (ACLs)

をサポートACLs： いいえ

アクセスコントロールリスト (ACLs) は、リソースへのアクセス許可を持つプリンシパル (アカウントメンバー、ユーザー、またはロール) を制御します。ACLs はリソースベースのポリシーに似ていますが、JSONポリシードキュメント形式を使用しません。

Security Lake は をサポートしていません。つまりACLs、Security Lake リソースACLに 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)

サポート ABAC (ポリシー内のタグ): はい

属性ベースのアクセスコントロール (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、の最初のステップですABAC。次に、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する場合に、オペレーションを許可するABACポリシーを設計します。

ABAC は、急速に成長している環境や、ポリシー管理が煩雑になる状況に役立ちます。

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。

の詳細についてはABAC、「 IAMユーザーガイド」の「 とはABAC」を参照してください。のセットアップ手順を含むチュートリアルを表示するにはABAC、「 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IAM」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 の 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、Amazon 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、「Amazon Security Lakeのアイデンティティベースのポリシー例」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 は、一時的な認証情報を使用してサインインすると機能 AWS サービス しません。一時的な認証情報 AWS サービス を使用する などの詳細については、 ユーザーガイドのAWS サービス 「 と連携する IAM IAM 」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 にサインインする場合、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 にアクセスすると、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。ロールの切り替えの詳細については、「 IAMユーザーガイド」の「ロールへの切り替え (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 を使用して手動で作成できます AWS API。その後、これらの一時的な認証情報を使用して . AWS recommends にアクセスできます AWS。これは、長期的なアクセスキーを使用する代わりに、一時的な認証情報を動的に生成することを推奨しています。詳細については、「」の「一時的なセキュリティ認証情報IAM」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をサポート (FAS): はい

IAM ユーザーまたはロールを使用して でアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS サービス、ダウンストリームサービス AWS サービス へのリクエストのリクエストと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 サービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するための権限が必要です。FAS リクエストを行う際のポリシーの詳細については、「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 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

サービスロールは、ユーザーに代わってアクションを実行するためにサービスが引き受ける IAM ロールです。IAM 管理者は、 内からサービスロールを作成、変更、削除できますIAM。詳細については、「 ユーザーガイド」の「 にアクセス許可を委任するロールの作成 AWS サービスIAM」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 を使用するときは、Amazon EventBridge、 AWS Lambda、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 サービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント 、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示できますが、編集することはできません。

Security Lake は、 という名前IAMのサービスにリンクされたロールを使用しますAWSServiceRoleForAmazonSecurityLake。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 AWS サービス 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 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、「Amazon Security Lake のサービスリンクロール」を参照してください。

Amazon Security Lakeのアイデンティティベースのポリシー例

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 を使用してタスクを実行することはできません AWS API。必要なリソースに対してアクションを実行するアクセス許可をユーザーに付与するために、IAM管理者はIAMポリシーを作成できます。その後、管理者はIAMポリシーをロールに追加し、ユーザーはロールを引き受けることができます。

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

各リソースタイプの の形式など、Security Lake で定義されるアクションとリソースタイプの詳細については、「サービス認証リファレンスARNs」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

- ポリシーの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

ポリシーの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を付与するには、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、「 ユーザーガイド」の「 AWS 管理ポリシーAWS 」または「 ジョブ機能の 管理ポリシーIAM」を参照してください。

- 最小特権のアクセス許可を適用する – IAMポリシーでアクセス許可を設定する場合は、タスクの実行に必要なアクセス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 を使用してアクセス許可を適用する方法の詳細については、「 ユーザーガイド」の「 のポリシーとアクセス許可IAMIAM」を参照してください。
- IAM ポリシーの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、アクションと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 を使用して送信する必要があることを指定できますSSL。条件を使用して、 などの特定の を介してサービスアクションが使用される場合に AWS 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、「 ユーザーガイド」のIAMJSON「ポリシー要素: 条件IAM」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、安全で機能的なアクセス許可を確保する – IAM Access Analyzer は、ポリシーがポリシー言語 (JSON) とIAMベストプラクティスに準拠するように、新規および既存のIAMポリシーを検証します。IAM Access Analyzer には、安全で機能的なポリシーの作成に役立つ 100 を超えるポリシーチェックと実用的な推奨事項が用意されています。詳細については、「 ユーザーガイド」のIAM「Access Analyzer ポリシーの検証IAM」を参照してください。
- 多要素認証を要求する (MFA) – 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MFAするために をオンにします。API オペレーションが呼び出されるMFAタイミングを要求するには、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「 で保護されたAPIアクセスの設定」を参照してください。

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「 のセキュリティのベストプラクティスIAMIAM」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、 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 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 AWS API。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を許可します。

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、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、「 ユーザーガイド」のIAM「 IDIAM」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、IAMユーザーがユーザー ID にアタッチされているインラインポリシーと管理ポリシーを表示できるようにするポリシーを作成する方法を示します。このポリシーには、コンソールで、または AWS CLI または 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 AWS API。

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

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

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 を使用してサブスクライバーのレビューをユーザーに許可するポリシーを作成する方法を示しますAPI。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。条件キーの使用の詳細については、「 ユーザーガイド」のIAMJSON「ポリシー要素: 条件IAM」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、Amazon Security Lake リソースのタグ付け を参照してください。

AWS Amazon 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対するアクセス許可を付与するように設計されているため、ユーザー、グループ、ロールへのアクセス許可の割り当てを開始できます。

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

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

詳細については、「IAM ユーザーガイド」の「AWS 管理ポリシー」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、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 Lambda 関数のすべてのアクセス許可が含まれています。

許可の詳細

このポリシーには、以下の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 に口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、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 管理ポリシー： 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、Security Lake が 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に必要な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。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPutLakeFormationSettings",
            "Effect": "Allow",
            "Action": "lakeformation:PutDatalakeSettings",
            "Resource": "*",
            "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および 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 — Security Lake テーブルの Lake Formation アクセス許可を管理することをプリンシパルに許可します。
- 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 — 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することをプリンシパルに許可します。
- 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": "AllowRolePolicyActionsforSubscibersandSources",
    "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

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、「Amazon Security Lake のサービスリンクロール」を参照してください。

AWS マネージドポリシー： AWS GlueServiceロール

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ルしてパーティションメタデータを識別 AWS Glue することを許可します。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、「カスタムソースからのデータ収集」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 — 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 マネージドポリシーに 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

変更	説明	日付
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
AmazonSecurityLakeMetastoreManager – 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`AmazonSecurityLakeMetastoreManagerV2` 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す。	2024 年 2 月 23 日
AmazonSecurityLakeMetastoreManager - 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、プリンシパルにすべての Security Lake アクションへのフルアクセスを許可する新しい 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

Amazon Security Lake のサービスリンクロール

Security Lake は、 という名前の 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AWSServiceRoleForSecurityLake。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、Security Lake がユーザーに代わって他の AWS 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 で、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は、その関連リソースを削除した後にのみ削除できます。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。

サービスリンクロールをサポートする他のサービスについては、「IAM と連動するAWS のサービス」を参照し、[Service-linked roles] (サービスリンクロール) の列内で [Yes] (はい) と表記されたサービスを確認してください。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 のサービスリンクロールの許可
- Security Lake のサービスリンクロールの作成
- Security Lake 向けのサービスリンクロールの編集
- Security Lake 向けのサービスリンクロールの削除
- Security Lake サービスにリンクされたロール AWS リージョン でサポートされます

Security Lake のサービスリンクロールの許可

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 の AWS マネージドポリシーの詳細については、AWS 「Amazon Security Lake のポリシーの管理」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は、SecurityLakeServiceLinkedRoleSecurity Lake がセキュリティデータレイクを作成および運用することを許可します。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します。
- 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 を使用したロール記述の編集はできます。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「サービスリンクロールの削除」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 でサポートされます

Security Lake は、Security Lake AWS リージョン が利用可能なすべての で、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、「Amazon Security Lake リージョンおよびエンドポイント」を参照してください。

Amazon Security Lake におけるデータ保護

責任 AWS 共有モデル、Amazon Security Lake でのデータ保護に適用されます。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。お客様は、このインフラストラクチャでホストされているコンテンツに対する管理を維持する責任があります。また、使用する AWS サービス のセキュリティ設定と管理タスクもユーザーの責任となります。データプライバシーの詳細については、「データプライバシー FAQ」を参照してください。欧州におけるデータ保護の詳細については、AWS 「 セキュリティブログ」のAWS 「責任共有モデル」とGDPRブログ記事を参照してください。

データ保護の目的で、認証情報を保護し AWS アカウント 、 AWS IAM Identity Center または AWS Identity and Access Management () を使用して個々のユーザーを設定することをお勧めしますIAM。この方法により、それぞれのジョブを遂行するために必要な権限のみが各ユーザーに付与されます。また、次の方法でデータを保護することもお勧めします:

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。1TLS.2 が必要で、1.3 TLS をお勧めします。
- を使用して 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。
- AWS 暗号化ソリューションと、 内のすべてのデフォルトのセキュリティコントロールを使用します AWS サービス。
- Amazon Macie などの高度なマネージドセキュリティサービスを使用します。これらは、Amazon S3 に保存されている機密データの検出と保護を支援します。
- コマンドラインインターフェイスまたは AWS を介して にアクセスするときに FIPS 140-3 検証済みの暗号化モジュールが必要な場合はAPI、FIPSエンドポイントを使用します。利用可能なFIPSエンドポイントの詳細については、「連邦情報処理規格 (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は、タグ、または名前フィールドなどの自由形式のテキストフィールドに配置しないことを強くお勧めします。これは、コンソール、、またはを使用して Security Lake または他の AWS サービス を使用する場合API AWS CLIも同様です AWS SDKs。名前に使用する自由記述のテキストフィールドやタグに入力したデータは、課金や診断ログに使用される場合があります。URL を外部サーバーに提供する場合は、そのサーバーへのリクエストを検証URLするために認証情報を に含めないことを強くお勧めします。

保管中の暗号化

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。生の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マルチテナントの Amazon Simple Storage Service (Amazon S3) バケットに保存されます。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あり、複数の AWS アカウントで使用できます。

Security Lake は、未加工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。処理されたデータは Security Lake サービスアカウントで暗号化されたままです。

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 バケットを作成します (Security Lake を有効に AWS リージョン した各 に 1 つのバケット)。Security Lake がデータ

をシングルテナントの S3 バケットに確実に配信できるようになるまで、データはマルチテナントの S3 バケットに一時的にのみ保存されます。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の暗号化キーまたはカスタマー管理のキー (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 キーを使用したデータの暗号化

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 がS3-managedによる Amazon サーバー側の暗号化によって暗号化SSE-S3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 (SSE-KMS) によるサーバー側の暗号化を使用できます。

SSE-KMS Security Lake コンソールではサポートされていません。Security Lake APIまたは で SSE-KMS を使用するにはCLI、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、に`GenerateDataKey`リクエストを送信します。
- に`RetireGrant`リクエストを送信します AWS KMS。データレイクを更新すると、このオペレーションにより、ETL処理のためにAWSKMSキーに追加されたグラントの廃止が有効になります。

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、キーのKMS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。

KMS キーポリシーを作成するとき、または適切なアクセス許可を持つ既存のキーポリシーを使用するときに、キーは許可リクエストを受け入れることができ、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": "*"
}
```

カスタマー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、「開始方法: 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 ID IAM と共有され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 Security Lake 管理者アカウントに暗号化されていない 2 つの 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 (SSE) では不十分です。 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 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？SQS」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、Transport Layer Security (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で、サービスとの間で送受信される転送中のデータを保護します。Security Lake に送信される直接HTTPSリクエストAPIsは、AWS 署名バージョン 4 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 が現在そ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、「 ユーザーガイド」の「組織の作成と管理AWS Organizations 」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 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、コンプライアンスプログラムAWS サービス による対象範囲内のコンプライアンスプログラムを参照し、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 「コンプライアンスプログラム」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、「 でのレポートのダウンロード AWS Artifact」の」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 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。 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。

- セキュリティとコンプライアンスのクイックスタートガイド – これらのデプロイガイドでは、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に重点を置いたベースライン環境 AWS を にデプロイする手順について説明します。
- アマゾン ウェブ サービスHIPAAのセキュリティとコンプライアンスのためのアーキテクチャ – このホワイトペーパーでは、企業が AWS を使用して HIPAA対象アプリケーションを作成する方法について説明します。

> **Note**
>
> すべての AWS サービス がHIPAA対象となるわけではありません。詳細については、HIPAA「対象サービスリファレンス」を参照してください。

- 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。このガイドでは、ガイダンスを保護し AWS サービス 、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) など) のセキュリティコントロールにマッピングするためのベストプラクティスをまとめています。
- 「 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 サービス 、 内の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リストについては、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 – これにより AWS アカウント、疑わしいアクティビティや悪意のあるアクティビティがないか環境を監視することで、、、ワークロード、コンテナ、データに対する潜在的な脅威 AWS サービス を検出します。 GuardDuty は、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DSS、 PCI などのさまざまなコンプライアンス要件への対応に役立ちます。
- AWS Audit Manager – これにより AWS 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

最小特権の原則に従い、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 ポリシー権限セットを付与します。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、「Amazon Security Lakeのアイデンティティベースのポリシー例」を参照してください。

AWS CloudTrailを使用して Security Lake での API 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、「AWS CloudTrail を使用した Amazon 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

セキュリティレイクとAWS Security Hubを統合し、Security Hub 調査結果をセキュリティレイクで受け取ることができます。Security Hub は、さまざまなAWS サービスとサードパーティインテグレーションから結果を生成します。Security Hub の調査結果を受け取ると、コンプライアンス状況の概要や、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がわかります。

詳細については、「との統合 AWS Security Hub」を参照してください。

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ーゾーンを中心に構築されています。AWS リージョン には、低レイテンシー、高いスループット、そして高度の冗長ネットワークで接続されている複数の物理的に独立・隔離されたアベイラビリティーゾーンがあります。これらのアベイラビリティー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。アベイラビリティーゾーンは、従来の単一または複数のデータセンターインフラストラクチャよりも可用性、耐障害性、および拡張性が優れています。

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ー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ー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 とアベイラビリティーゾーンの詳細については、「AWS グローバルインフラストラクチャ」を参照してください。

Amazon Security Lake では、AWSグローバルインフラストラクチャに加えて、Simple Storage Service (Amazon S3); データの耐障害性とバックアップのニーズに対応できるように複数の機能を提供しています。

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アント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「EFS 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 AWS グローバルネットワークセキュリティで保護されています。 AWS セキュリティサービスと がインフラストラクチャ AWS を保護する方法については、AWS 「 クラウドセキュリティ」を参照してください。インフラストラクチャセキュリティのベストプラクティスを使用して AWS 環境を設計するには、「セキュリティの柱 AWS Well-Architected Framework」の「Infrastructure Protection」を参照してください。

が AWS 公開したAPI呼び出しを使用して、ネットワーク経由で 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) 。1TLS.2 が必要で、1.3 TLS をお勧めします。
- (Ephemeral Diffie-HellmanPFS) や DHE (Elliptic Curve Ephemeral Diffie-Hellman) などの完全前方秘匿性 ECDHE () を備えた暗号スイート。これらのモードは、Java 7 以降など、ほとんどの最新システムでサポートされています。

さらに、リクエストは、アクセスキー ID とプリンIAMシパルに関連付けられたシークレットアクセスキーを使用して署名する必要があります。または、AWS Security Token Service (AWS STS) を使用して、一時セキュリティ認証情報を生成し、リクエストに署名することもできます。

Security Lake での構成と脆弱性の分析

構成および IT 管理は、AWS とお客様の間で共有される責任です。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のモニタリング

Security Lake は、ユーザー、ロール、または別の AWS サービス によって Security Lake で実行されたアクションの記録を提供するサービスである AWS CloudTrail と統合されます。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、「AWS CloudTrail を使用した Amazon 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、「セキュリティレイク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、特定のしきい値をモニタリングするアラームを設定し、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

- セキュリティレイク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
- セキュリティレイクメトリクスの CloudWatch アラームの設定

セキュリティレイク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	データ レイクに現在保存され、ネイティブにサポートされている AWS サービス からのデータの量。 単位: バイト

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

ディメンション	説明
	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ProcessedSize メトリクス。
Source	特定のAWSログソースのProcessedSize メトリクス。
SourceVersion	特定のバージョンのAWSログソースのProcessedSize メトリクス。

特定の 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 (Per-Source Version Metrics) のメトリクスを表示できます。

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

AWS CLI を使用して メトリクスにアクセスするには、 `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 User Guide] (Amazon CloudWatch ユーザーガイド) の[Use Amazon CloudWatch Metrics] (Amazon CloudWatch メトリクスの使用) を参照してください。

セキュリティレイクメトリクスの CloudWatch アラームの設定

CloudWatch では、メトリクス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。

AWS CloudTrail を使用した Amazon Security Lake API コールのログ記録

Amazon Security Lake は、Security Lake のユーザー、ロール、または AWS サービスによって実行されたアクションの記録を提供するサービスである AWS CloudTrail と統合します。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。追跡を設定しない場合でも、CloudTrail コンソールの [Event history] (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

CloudTrail は、アカウント作成時に AWS アカウント で有効になります。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。最近のイベントは、AWS アカウント で表示、検索、ダウンロードできます。詳細については、 CloudTrail 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 のイベントを継続的に記録する場合は、追跡を作成します。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 ログで収集したイベントデータをより詳細に分析し、それに基づく対応するためにその他の AWS の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 によって口グに記録され、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。同一性情報は次の判断に役立ちます。

- リクエストが、ルートと AWS Identity and Access Management ユーザー認証情報のどちらを使用して送信されたか。
- 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一時的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のか。

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 のログファイルには、単一か複数のログエントリがあります。イベントはあらゆるソースからの単一のリクエストを表し、リクエストされたアクション、アクションの日時、リクエストのパラメータなどの情報が含まれます。CloudTrail ログファイルは、パブリック API コールの順序付けられたスタックトレースではないため、特定の順序では表示されません。

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"
}
```

Amazon 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が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。

Security Lake リソースには、サブスクライバーと個々の AWS アカウント の のデータレイク設定の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

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 つのタグ値で構成されており、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機能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 サービス、その他のタグ値`On-Premises`の場合は である場合があります。

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。ベスト プラクティスとして、タグを大文字にする戦略を定義し、その戦略をリソース全体で一貫して実装することをお勧めします。
- タグキーは最大 128 文字 (UTF-8) です。タグ値には最大 256 文字の UTF-8 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: _ 。 : / = + - @
- aws: プレフィックスは、 が使用できるように予約されています AWS。定義したどのタグキーや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を変更または削除することはできません。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 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もタグを割り当てるには、 AWS Resource Groups またはリソース AWS サービス を保存する を使用できます。例えば、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 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と、 内のどのユーザーとロールが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 を持ち、どのユーザーとロールがより一般的にタグを追加、編集、削除するアクセス許可を持つかをきめ細かく制御できます。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タグをリソースに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 Hub 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。
   - データレイク設定の場合は、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、「My subscribers」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、[Save (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、 を使用している場合は create-subscriber コマンド AWS CLI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、`key`と`value`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 Hub 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。

   - データレイク設定の場合は、ナビゲーションペインの [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、「My subscribers」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。例:

```
$ 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: Security Lake: us-east-1:1289012: subscriber/12abcd-12ab-34ab-34ab-34ab-34cd-56eb-12ab-34cd-56ab*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。例:

```
{
    "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 Hub 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。

   - データレイク設定の場合は、ナビゲーションペインの [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、「My subscribers」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、値を含む「値」ボックスで X を選択します。次に、[Value] ボックスに新しいタグ値を入力します。
   - 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、タグの横にある [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、または次の記号を使用できます: _ 。 : / = + - @

6. タグの編集が完了したら、[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で上書きされます。したがって、タグを編集する最適な方法は、タグキーまたはタグ値を編集するのか、またはその両方を編

集するの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。例:

```
$ 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 Hub 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。
   - データレイク設定の場合は、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、「My subscribers」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。
   - タグの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

プログラムでリソースから 1 つ以上のタグを削除するには、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、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。例:

```
$ aws securitylake untag-resource \
--resource-arn arn:aws:securitylake:us-
east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab \
--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。

Amazon Security Lake のトラブルシューティング

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。

データレイク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 の ListDataLakeExceptionsオペレーションを使用できますAPI。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` パラメータには、米国東部 (バージニア北`us-east-1`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"
```

問題またはエラーを Security Lake ユーザーに通知するには、Security Lake の CreateDataLakeExceptionSubscriptionオペレーションを使用しますAPI。ユーザーには、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取ることができます。

例えば、次の AWS CLI コマンドは、Security Lake 例外の通知を指定されたアカウントに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 および AWS Lake Formation データベースまたはテーブルの使用時に発生する可能性がある一般的な問題の診断と修正に役立ちます。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、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、「Amazon 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 を呼び出すと、このエラーが表示されることがありますCreateCustomLogSourceAPI。

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。を呼び出すIAMロールには、Security Lake データベースのテーブル作成アクセス許可API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: SELECT * 列がないリレーションからは許可されません

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 Lake Formation 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。

このエラーは、Security Lake を有効にするとき、または をログソース AWS サービス として追加するときに発生することがあります。

このエラーを解決するには、以下の手順を実行します。

1. で Lake Formation コンソールを開きますhttps://console.aws.amazon.com/lakeformation/。
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. 「Not found in IAM」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。

Security Lake CreateSubscriber with Lake Formation が、受け入れるための新しいRAMリソース共有の招待を作成しなかった

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 AWS RAM つのリソース共有にマッピングすることで、リソース共有の数 AWS RAMを最適化するためです。

リソース共有名にサブスクライバーの作成時に指定した外部 ID があり、リソース共有が`CreateSubscriber`レスポンスARN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. で AWS Lambda コンソールを開きますhttps://console.aws.amazon.com/lambda/。
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. で AWS Glue コンソールを開きますhttps://console.aws.amazon.com/glue/。
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を使用するときに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 オペレーションを呼び出す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ある必要があります。

委任された管理者アカウントが属していた組織を削除し、そのアカウントを使用して Security Lake コンソールまたは を使用して Security Lake CreateDataLake をセットアップしようとすると、このエラーが表示されることがありますAPI。

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 の使用時に発生する可能性がある一般的な問題の診断と修正に役立ちますIAM。

Security Lake でアクションを実行することが認可されていない

がアクションを実行する権限がないと AWS Management Console 通知した場合は、管理者に連絡してサポートを依頼する必要があります。管理者は、認証情報を自分に提供した人物です。

次の例のエラーは、mateojacksonIAMユーザーが コンソールを使用して架空の の詳細を表示しようとしている*subscriber*が、架空の SecurityLake:*GetSubscriber* アクセス許可がない場合に発生します。

```
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。

iam を実行する権限がありません。PassRole

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 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、そのサービスに既存のロールを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。

次の例のエラーは、 というIAMユーザーがコンソールを使用して Security Lake marymajor 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 (ACLs) をサポートするサービスでは、これらのポリシーを使用して、ユーザーにリソースへのアクセスを許可できます。

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、「Amazon Security Lake と の連携方法 IAM」を参照してください。
- 所有している のリソースへのアクセスを提供する方法については、 AWS アカウント 「 ユーザーガイド」の「所有 AWS アカウント している別の のIAMユーザーへのアクセスを提供するIAM」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の に提供する方法については AWS アカウント、「 ユーザーガイド」の「サードパーティー AWS アカウント が所有する へのアクセスを提供するIAM」を参照してください。
- ID フェデレーションを通じ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「外部認証されたユーザーへのアクセスの提供 (ID フェデレーション)」を参照してください。
- クロスアカウントアクセスでのロールとリソースベースのポリシーの使用の違いについては、ユーザーガイドの「 でのクロスアカウントリソースアクセスIAMIAM」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 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 の料金体系を理解するには、次の動画をご覧ください。Amazon Security Lake の料金表

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 サービス ログとイベント (Amazon Route 53 リゾルバーのクエリログ、 AWS Security Hub 検出結果、Amazon VPC Flow Logs) 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)スキーマに正規化し、Apache Parquet 形式に変換する AWS サービス ログとイベントの量から算出されます。

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 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次に示します。

- 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キューの作成から発生します。詳細については、「Amazon のSQS料金」を参照してください。
- Amazon EventBridge – これらのコストは、Amazon 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 EventBridge を送信することから発生します。詳細については、「Amazon EventBridge の料金」を参照してください。

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。

補助サービスの完全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。コストは、組織内のすべてのアカウントによる Security Lake の使用を対象とし、Open Cybersecurity Schema Framework (OCSF) および 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、今後 30 暦日間 AWS リージョン における現在の で予測される Security Lake 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からのデータ収集 AWS サービス。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. で Security Lake コンソールを開きますhttps://console.aws.amazon.com/securitylake/。
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。

Amazon Security Lake リージョンおよびエンドポイント

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。

Amazon 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 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 の AWS ログとイベントの収集 AWS リージョン 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 AWS CLI は、リクエストで指定したリージョンのログ収集も停止します。

との統合を使用して AWS Organizations いて、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、委任された Security Lake 管理者のみが、それ自体とメンバーアカウントに対して 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。

Security Lake を再度有効にする前に AWS Glue テーブルを削除して、サブスクライバーのクエリアクセスが正しく動作することを確認することをお勧めします。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 AWS CLI コマンド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 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-date-lake コマンドを実行します。リクエストでは、 `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、組織の委任された Security Lake 管理者のみが、組織内のアカウントの Security Lake を無効にできます。

例えば、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"
```

よくある質問

最新バージョンの Parquet への Security Lake の更新

5/20/2024 に、Amazon Security Lake は最新バージョンの parquet に更新されます。

Security Lake がこの更新を行う理由

Amazon では、お客様に安全で効率的なサービスを提供する継続的な取り組みの一環として、Security Lake は依存関係、サードパーティーライブラリ、APIsツールを定期的に更新しています。また、Security Lake は、お客様が Parquet 仕様を含むすべての標準の最新拡張機能を使用していることを保証します。

まれに、データの保存や処理方法にわずかな変更が生じることがあります。変更は、確立されたコミュニティ標準内で常に下位互換性があります。

Security Lake は、お客様のセキュリティログファイルを OCSF 形式に正規化し、クエリ効率の高い Parquet 形式で公開します。Security Lake は、最新の Parquet 形式をシームレスに採用するために、この変更を加えています。詳細については、「Parquet 形式」を参照してください。

Parquet 仕様の変更の詳細については、どこで確認できますか？

詳細については、parquet 形式 GitHub リポジトリの「非推奨のタイムスタンプ ConvertedType」を参照してください。

このアップグレードは Security Lake の統合に影響しますか？

Amazon Athena または Apache ツール (Spark、Hive、Impala、Hadoop) のみを使用して Security Lake テーブルにアクセスする場合、変更はありません。アップグレードに関連する変更は、クライアントツールと APIsによって透過的に自動的に処理されます。

他のクライアントツールを使用する場合、Security Lake では、日付/時刻フィールドの保存と処理の新しい方法を理解することをお勧めします。次の表に、古い合成データと新しい合成データの間に見られる可能性のある小さな違いを示します。

合成データの変更

AWS サービス	タイプ	Current	New
Amazon Athena	日付/時刻	1970-01-20 03:04:05.399000	変更なし
Apache Spark	日付/時刻	1970-01-20T00:04:05.000-03:00	変更なし
PyArrow	日付/時刻	1970-01-20 03:04:05	1970-01-20 03:04:05+00:00 UTC タイムゾーンマーカーの導入が変更されました。

Parquet 形式処理の変更を特定するにはどうすればよいですか？

zip ファイル parquet_format.zip をダウンロードします。zip ファイルは 2 つのファイルで構成されます。

- 古いフレームワークによって生成された合成テストデータ – `parquet_format_old.parquet`
- 新しいフレームワークによって生成された合成テストデータ – `parquet_format_new.parquet`

クライアントツールをテストし、古いフレームワークによって生成された合成テストデータと新しいフレームワークによって生成されたデータを比較します。

顕著な変更が見られる場合は、`Changes in synthetic data`表の推奨事項を使用してください。さらにサポートが必要な場合は、 AWS サポートにお問い合わせください。

「 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： 2024 年 6 月 10 日

変更	説明	日付
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
既存の管理ポリシーの更新	Security Lake は、ポリシーの AWS マネージドSecurityLakeServiceLinkedRole 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 口グソースとして有効になっているときに、Security Lake が口グを収集できるようになります。	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。 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するのに役立ちます。	2024 年 5 月 22 日
既存の マネージドポリシーの更新	Security Lake が 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	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ために、AmazonSecurityLakeMetastoreマネージャーポリシーを更新しました。	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、ログ AWS ソースとして EKS 監査ログを追加しました。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	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
新しい マネージドポリシー	Security Lake は、新しい AWS マネージドポリシー 、 AmazonSecurityLake MetastoreManager 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) の 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 マネージドポリシー、 AmazonSecurityLakeAdministrator 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 が Amazon にメトリクスを送信する CloudWatch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) の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。 https://docs.aws.amazon.com/security-lake/latest/userguide/getting-started.html	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。