



ユーザーガイド

Amazon Security Lake



Amazon Security Lake: ユーザーガイド

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

Amazon の商標およびトレードドレスは、お客様に混乱を招く可能性がある態様、または Amazon の信用を傷つけたり、失わせたりする態様において、Amazon のものではない製品またはサービスに関連して使用してはなりません。Amazon が所有しない他の商標はすべてそれぞれの所有者に帰属します。所有者は必ずしも Amazon との提携や関連があるわけではありません。また、Amazon の支援を受けているとはかぎりません。

Table of Contents

Amazon Security Lake とは何ですか?	1
Security Lakeの概要	1
Security Lake の特徴	2
Security Lakeへのアクセス	3
関連サービス	4
概念と用語	6
開始	8
初期 AWS アカウント セットアップ	8
にサインアップする AWS アカウント	8
管理ユーザーの作成	8
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。	9
Amazon Security Lake を有効にする際の考慮事項	10
コンソールの開始方法	11
ステップ 1: ソースを設定する	11
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)	12
ステップ 3: データレイクを確認して作成する	13
ステップ 4: 独自のデータを表示してクエリする	13
ステップ 5: サブスクライバーを作成する	13
プログラムによる開始方法	14
ステップ 1: IAM ロールを作成する	14
ステップ 2: Amazon Security Lake を有効にする	15
ステップ 3: ソースを設定する	16
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)	17
ステップ 5: 独自のデータを表示してクエリする	18
ステップ 6: サブスクライバーを作成する	19
複数のアカウントの管理	20
委任された Security Lake 管理者に関する重要な考慮事項	20
委任された管理者を指定するには IAM 許可が必要です	21
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。	22
委任された Security Lake 管理者を削除する。	24
Security Lake の信頼できるアクセス	25
リージョンの管理	26
リージョン・ステータスのチェック	26
リージョン設定の変更	27

ロールアップリージョンの設定	29
データレプリケーションの IAM ロール	29
AWS Glue パーティションを登録する IAM ロール	32
ロールアップリージョンの追加	33
ロールアップリージョンの更新または削除	34
ソース管理	37
AWS サービスからのデータ収集	37
前提：アクセス許可	38
CloudTrail イベントログ	39
Amazon EKS 監査ログ	40
Route 53 Resolver クエリログ	41
Security Hub の検出結果	41
VPC Flow Logs	42
AWS のサービスをソースとして追加	43
ロール権限の更新	44
AmazonSecurityLakeMetaStoreManager ロールを削除する。	46
AWS のサービスをソースとして削除	46
ソースコレクションのステータスを取得しています。	48
カスタムソースからのデータ収集	48
カスタムソースの取り込みのベストプラクティス	49
カスタムソースを追加するための前提条件	50
カスタムソースの追加	54
カスタムソースデータを常に最新の状態に保ちます。 AWS Glue	55
カスタムソースの削除	56
サブスクライバー管理	58
サブスクライバーデータアクセス	59
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	59
データにアクセスできるサブスクライバーを作成する。	62
サンプル通知メッセージの例	65
データサブスクライバーの更新	66
データサブスクライバーを削除する。	67
サブスクライバークエリへのアクセス	68
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	68
クエリアクセス権を持つサブスクライバーの作成	71
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)	73
クエリアクセス権を持つサブスクライバーの編集	74

Security Lake クエリ	79
Security Lake クエリバージョン 1	79
ログソーステーブル	79
データベースリージョン	80
パーティション日付	81
CloudTrail データのクエリ例	83
Route 53 Resolver クエリログのクエリ例	85
Security Hub の検出結果に関するクエリ	87
Amazon VPC フローログのクエリ例	90
Security Lake クエリバージョン 2	94
ログソーステーブル	79
データベースリージョン	80
パーティション日付	81
Security Lake オブザーバビリティのクエリ	97
CloudTrail データのクエリ例	83
Route 53 Resolver クエリログのクエリ例	85
Security Hub の検出結果に関するクエリ	87
Amazon VPC フローログのクエリ例	90
Amazon EKS のクエリ例	108
ライフサイクル管理	110
保持管理	110
Security Lake を有効にする際の保存設定を行います。	110
保持設定の更新	112
ロールアップリージョン	113
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)	114
OCSFとは何ですか	114
OCSF イベントクラス	114
OCSFソースの識別	114
統合	118
AWS のサービス インテグレーション	118
AWS AppFabric 統合	119
セキュリティハブの統合	119
サードパーティ統合	121
Accenture – MxDR	122
Aqua Security	122
Barracuda – Email Protection	122

Booz Allen Hamilton	122
ChaosSearch	123
Cisco Security – Secure Firewall	123
Claroty – xDome	123
CMD Solutions	123
Confluent – Amazon S3 Sink Connector	124
Contrast Security	124
Cribl – Search	124
Cribl – Stream	124
CrowdStrike – Falcon Data Replicator	125
CyberArk – Unified Identify Security Platform	125
Darktrace – Cyber AI Loop	125
Datadog	125
Deloitte – MXDR Cyber Analytics and AI Engine (CAE)	126
Devo	126
DXC – SecMon	126
Eviden— Alsaac (旧 Atos)	126
ExtraHop – Reveal(x) 360	127
Falcosidekick	127
Gigamon – Application Metadata Intelligence	127
Hoop Cyber	127
IBM – QRadar	128
Infosys	128
Insbuilt	128
Kyndryl – AIOps	128
Lacework – Polygraph	129
Laminar	129
MegazoneCloud	129
Monad	129
NETSCOUT – Omnis Cyber Intelligence	129
Netskope – CloudExchange	130
New Relic ONE	130
Okta – Workforce Identity Cloud	130
Orca – Cloud Security Platform	131
Palo Alto Networks – Prisma Cloud	131
Ping Identity – PingOne	131

PwC – Fusion center	131
Rapid7 – InsightIDR	132
RipJar – Labyrinth for Threat Investigations	132
Sailpoint	132
Securonix	132
SentinelOne	133
Sentra – Data Lifecycle Security Platform	133
SOC Prime	133
Splunk	133
Stellar Cyber	134
Sumo Logic	134
Swimlane – Turbine	134
Sysdig Secure	134
Talon	135
Tanium	135
TCS	135
Tego Cyber	135
Tines – No-code security automation	136
Torq – Enterprise Security Automation Platform	136
Trellix – XDR	136
Trend Micro – CloudOne	136
Uptycs – Uptycs XDR	137
Vectra AI – Vectra Detect for AWS	137
VMware Aria Automation for Secure Clouds	137
Wazuh	137
Wipro	138
Wiz – CNAPP	138
Zscaler – Zscaler Posture Control	138
セキュリティ	139
ID およびアクセス管理	140
対象者	140
アイデンティティを使用した認証	141
ポリシーを使用したアクセスの管理	144
Amazon Security Lake と IAM との連携方法	147
アイデンティティベースポリシーの例	157
AWS マネージドポリシー	162

サービスリンクロール	182
データ保護	187
保管中の暗号化	188
転送中の暗号化	190
サービス改善のためのデータ使用をオプトアウトする	190
コンプライアンス検証	191
Security Lake のセキュリティのベストプラクティス	192
Security Lake ユーザーへの最小限のアクセス許可の付与	192
概要ページを表示します。	193
Security Hubとの統合	193
セキュリティレイクのイベントを監視してください。	193
耐障害性	193
インフラストラクチャセキュリティ	195
Security Lake での構成と脆弱性の分析	195
モニタリング	195
Amazon Security Lake の CloudWatch メトリクス	196
API コールのログ作成	199
CloudTrail での Security Security Security Information	199
Security Lake のログファイルエントリについて	200
リソースのタグging	202
タグ付けの基本	202
IAMポリシーでタグを使用する	204
リソースに タグを追加する	204
リソースのタグを確認する	207
リソースのタグを編集する	209
リソースからのタグの削除	212
トラブルシューティング	214
データレイクステータスのトラブルシューティング	214
Lake Formation 問題のトラブルシューティング	215
テーブルが見つからない	215
400 AccessDenied	215
SYNTAX_ERROR: 1:8 行目:カラムのないリレーションからの SELECT * は許可されてい ません	215
Security Lakeは、呼び出し元のプリンシパル ARN をレイクフォーマーションデータレイク 管理者に追加できませんでした。現在のデータレイク管理者には、もはや存在しない無効な プリンシパルが含まれている可能性があります。	216

Lake Formation CreateSubscriber を使用した Security Lake が、受け入れる新しい RAM リソース共有招待を作成しなかった	216
Amazon Athena でのクエリのトラブルシューティング	217
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。	217
AWS Glue テーブルにアクセスできない	217
Organizations の問題のトラブルシューティング	218
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。	218
IAM 問題のトラブルシューティング	218
Security Lake でアクションを実行することが認可されていない	218
iam を実行する権限がありません。PassRole	219
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	219
Security Lake の価格	221
使用状況と推定コストの確認	222
サポートされているリージョンおよびエンドポイント	224
Security Lake を無効にする	225
ドキュメント履歴	227
.....	CCXXX

Amazon Security Lake とは何ですか？

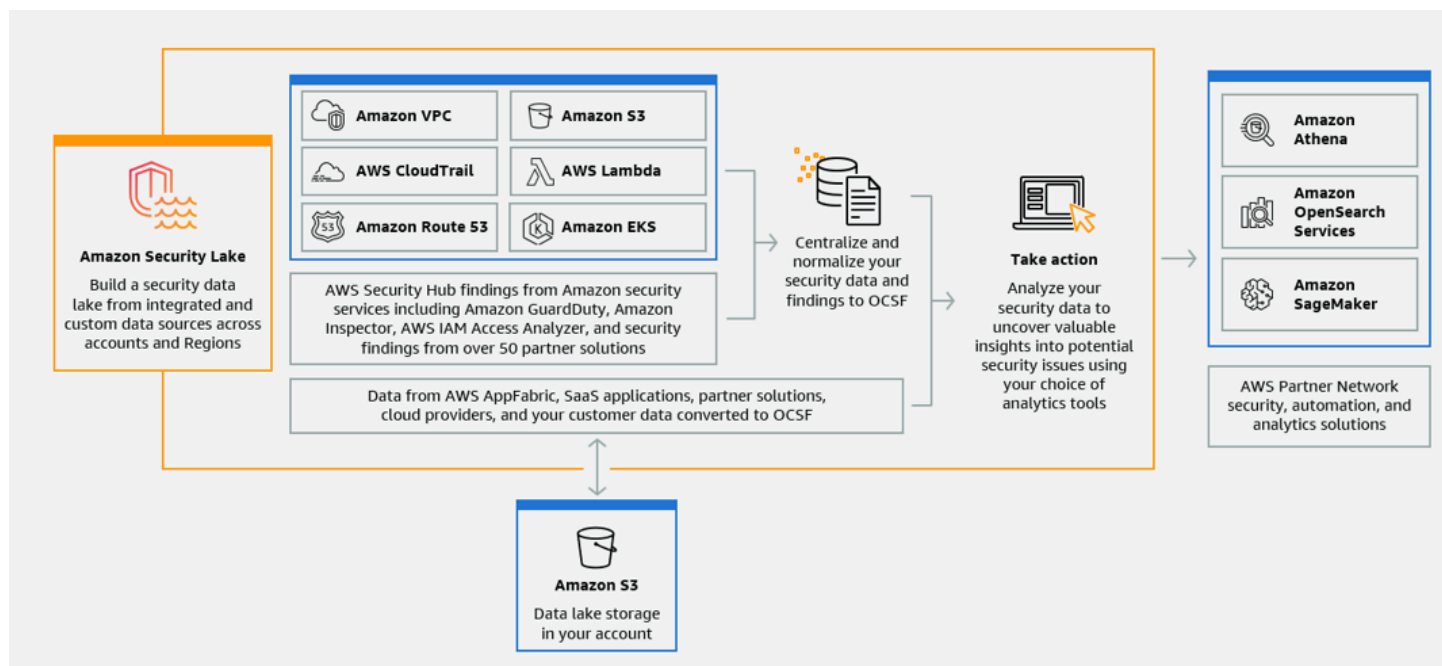
Amazon Security Lake は、完全マネージド型のセキュリティデータレイクサービスです。Security Lake を使用すると、AWS 環境、SaaS プロバイダー、オンプレミス、クラウド ソース、サードパーティ ソースからのセキュリティ データを、AWS アカウント に保存される専用のデータ レイクに自動的に一元化できます。Security Lake はセキュリティデータの分析に役立つため、組織全体のセキュリティ体制をより完全に把握できます。Security Lake を使用すると、ワークロード、アプリケーション、データの保護を強化することもできます。

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

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

他のAWS のサービスやサードパーティのサービスが Security Lake に保存されているデータをサブスクライブして、インシデント対応やセキュリティデータ分析を行うことができます。

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AWS のサービス、第三者のサービスを含む複数のソースからセキュリティログとイベントを収集します。ログを取り込んだ後は、ソースに関係なく、ログに一元的にアクセスしてライフサイクルを管理できます。Security Lake がログとイベントを収集するソースの詳細については、[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 のサービス から Open Cybersecurity Schema Framework (OCSF) オープンソース スキーマにデータを変換します。また、ネイティブにサポートされている AWS のサービス から Open Cybersecurity Schema Framework (OCSF) オープンソース スキーマにデータを変換します。これにより、後処理を必要とせずに他のAWS のサービスや第三者プロバイダーとデータとの互換性が保たれます。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

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 を使用できる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)

AWS CLIを使用すると、システムのコマンドラインでコマンドを発行して Security Lake の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。AWS CLI のインストールおよび使用の詳細については、[AWS Command Line Interface](#) を参照してください。

AWS SDK

AWS は、Java、Go、Python、C++、.NET など、さまざまなプログラミング言語およびプラットフォーム用のライブラリとサンプルコードで構成される SDK を提供します。SDK は、Security Lake および他の AWS のサービス への便利なプログラムによるアクセスを提供します。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 SDK のインストールと使用の詳細については、「[AWS での構築ツール](#)」を参照してください。

関連サービス

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

- [Amazon EventBridge](#) — Security Lakeは、オブジェクトがデータレイクに書き込まれると、EventBridge を使用してサブスクライバーに通知します。
- [AWS Glue](#)— Security Lake 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データカタログに送信します。Security Lake は、Data Catalog 内のAWS Lake Formationテーブルのパーティションメタデータも保存します。
- [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マ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、以下のAWS のサービスに加えてカスタムソースからデータを収集します。

- AWS CloudTrail管理とデータイベント (S3、Lambda)
- Amazon Route 53 Resolver クエリログ

- AWS Security Hub の検出結果
- Amazon Virtual Private Cloud (Amazon VPC) フローログ

これらのソースの詳細については、「[AWS サービスからのデータ収集](#)」を参照してください。OCSF スキーマのデータを読み取ることができるサブスクライバーを作成することで、セキュリティデータレイクの Amazon S3 オブジェクトを使用できます。Amazon Athena、Amazon Redshift、AWS Glueと統合されたサードパーティ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。

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 のサービス 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 のサービスソースは別のサービスでもサードパーティのサービスでもかまいません。サードパーティのソースの場合は、Security Lake に送信する前にデータを OCSF スキーマに変換する必要があります。

サブスクライバー

Security Lake からのログとイベントを利用するサービス。サブスクライバーは別のAWS のサービス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。データレイク設定の構成方法とログ収集の設定方法について説明します。Security Lake を有効にして使用するには、を使用する AWS Management Console か、プログラムを使用します。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。

初期 AWS アカウント セットアップ

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、次のステップを実行して作成します。

にサインアップするには AWS アカウント

1. <https://portal.aws.amazon.com/billing/signup> を開きます。
2. オンラインの手順に従います。

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

にサインアップすると AWS アカウント、AWS アカウントのルートユーザーが作成されます。ルートユーザーには、アカウントのすべての AWS のサービス とリソースへのアクセス権限があります。セキュリティのベストプラクティスとして、[管理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](#)を実行します。

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。

管理ユーザーの作成

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](#) としてサインインします。次のページでパスワードを入力します。

ルートユーザーを使用してサインインする方法については、「AWS サインイン ユーザーガイド」の「[ルートユーザーとしてサインインする](#)」を参照してください。

2. ルートユーザーの多要素認証 (MFA) を有効にします。

手順については、「IAM [ユーザーガイド](#)」の AWS アカウント「[ルートユーザーの仮想 MFA デバイスを有効にする \(コンソール\)](#)」を参照してください。

管理ユーザーを作成する

1. IAM アイデンティティセンターを有効にします。

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

2. IAM アイデンティティセンターで、管理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的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。

管理ユーザーとしてサインインする

- IAM アイデンティティセンターのユーザーとしてサインインするには、IAM アイデンティティセンターのユーザーの作成時に E メールアドレスに送信されたサインイン URL を使用します。

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

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、「[による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。[ル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 の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アクセス許可が含まれます。サービスリンクロールの仕組みの詳細については、「IAM ユーザーガイド」の「[サービスリンクロールの使用](#)」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 を有効にして設定する方法について説明します。AWS Management Console の一部として AWS Management Console、Security Lake コンソールは開始するための合理化されたプロセスを提供し、データレイクの作成に必要なすべての AWS Identity and Access Management (IAM) ロールを作成します。

ステップ 1: ソースを設定する

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

ログソース収集を設定するには

1. Security Hub コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ページの右上隅にある AWS リージョン セレクターを使用して、リージョンを選択します。Security Lake は、オンボーディング中に現在のリージョンと他のリージョンで有効にできます。
3. [使用を開始] 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 ソースを取り込む – 推奨されるオプションを選択すると、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、取り込むログソースとイベントソースを 1 つ以上選択できます。

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。

- バージョン で、ログ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。

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

- 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。ロールアップ地域とそれに貢献する地域を指定します。1 つ以上のロールアップリージョンを設定できます。
- [ストレージクラスを選択] では、Amazon S3 ストレージクラスを選択します。デフォルトのストレージクラスは、S3 Standardです。それ以降にデータを別のストレージクラスに移行する場合は保持期間 (日単位) を指定し、[Add transition] を選択します。保持期間が終了すると、オブジェクトは期限切れになり、Amazon S3 はそれらを削除します。Amazon S3 ストレージ クラスと保持の詳細については、「[保持管理](#)」を参照してください。

3. 最初のステップでサービスアクセス用にロールアップリージョンを選択した場合は、新しい IAM ロールを作成するか、Security Lake に複数のリージョンにデータを複製する権限を付与する既存の IAM ロールを使用してください。
4. [次へ] を選択します。

ステップ 3: データレイクを確認して作成する

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

データレイクを確認して作成するには

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

データレイクを作成すると、Security Lake コンソールに Summary ページが表示されます。このページでは、リージョン数とロールアップリージョン数の概要、サブスクライバーに関する情報、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。権限レベルの詳細については、「[Lake Formation ペルソナ](#)」と「[IAM 権限リファレンス](#)」を参照してください。SELECT 権限を付与する手順については、『AWS Lake Formation 開発者ガイド』の「[名前付きリソースメソッドを使用した Data Catalog 権限の付与](#)」を参照してください。

ステップ 5: サブスクライバーを作成する

データレイクを作成したら、データを使用するサブスクライバーを追加できます。サブスクライバーは、Amazon S3 バケット内のオブジェクトに直接アクセスするか、データレイクにクエリを実行す

ることでデータを使用できます。サブスクライバーの詳細については、「[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方法

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 を使用すると、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になります。または、[AWS Command Line Interface](#)または [AWS Tools for PowerShell](#) などの AWS コマンドラインツール、または [AWS SDKs](#) を使用して Security Lake にアクセスすることもできます。

ステップ 1: IAM ロールを作成する

Security Lake にプログラムでアクセスする場合は、データレイクを設定するためにいくつかの AWS Identity and Access Management (IAM) ロールを作成する必要があります。

Important

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

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

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

前述のロールを作成したら、Security Lake を有効にするために使用しているロールに [AmazonSecurityLakeAdministrator](#) AWS 管理ポリシーをアタッチします。このポリシーにより、プ

リンシパルが Security Lake にオンボーディングし、Security Lake のすべてのアクションにアクセスすることが許可される、管理者許可が付与されます。

[AmazonSecurityLakeMetaStoreManager](#) AWS マネージドポリシーをアタッチして、データレイクを作成するか、Security Lake からデータをクエリします。このポリシーは、Security Lake がソースから受け取る生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のに必要です。

ステップ 2: Amazon Security Lake を有効にする

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

例 1

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

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

例 2

次のコマンド例では、us-east-2 リージョンで Security Lake を有効にします。このデータレイクは、AWS Key Management Service () で作成されたカスタマー マネージドキーで暗号化されます AWS KMS。オブジェクトは 500 日後に期限切れになり、オブジェクトは 30 日後に GLACIER S3

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

```
$ aws securitylake create-data-lake \
--configurations '[{"encryptionConfiguration":
{"kmsKeyId":"1234abcd-12ab-34cd-56ef-1234567890ab"},"region":"us-
east-2","lifecycleConfiguration": {"expiration":{"days":500},"transitions":
[{"days":30,"storageClass":"GLACIER"}]}]' \
--meta-store-manager-role-arn "arn:aws:iam:us-east-1:123456789012:role/service-role/
AmazonSecurityLakeMetaStoreManager"
```

Note

Security Lake を既に有効にしている、リージョンまたはソースの設定を更新する場合は、[UpdateDataLake](#) オペレーションを使用するか、を使用する場合は AWS CLI、[update-data-lake](#) コマンドを使用します。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 のサービスをソースとして追加する場合にのみ使用できます。カスタムソースの追加については、[カスタムソースからのデータ収集](#) を参照してください。

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

Note

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

ます。同様に、sourceVersionパラメータを除外すると、Security Lake は指定されたソースのすべてのバージョンにリクエストを適用します。

Security Lake を有効にしていないリージョンをリクエストで指定すると、エラーが発生します。このエラーに対処するには、regionsアレイに Security Lake を有効にしたリージョンのみを指定するようにしてください。または、リージョンで Security Lake を有効にしてからリクエストを再度送信することもできます。

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

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

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

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

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

- ロールアップリージョンを指定するには、regionフィールドを使用して、ロールアップリージョンにデータを提供するリージョンを指定します。replicationConfiguration オブジェクトの regions配列で、各ロールアップリージョンのリージョンコードを指定します。リージョンコードのリストについては、AWS 全般のリファレンスの「[Amazon Security Lake エンドポイント](#)」を参照してください。
- データの保存設定を指定するには、以下の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 を使用してデータレイクを作成します。us-east-1 リージョンは、ap-northeast-2 リージョンにデータを提供します。この例では、データレイクに追加されたオブジェクトの有効期限を 10 日間設定します。

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

これで、データレイクが作成されました。Security Lake API の [ListDataLakes](#) オペレーションを使用して、各リージョンで Security Lake の有効化とデータレイク設定を確認します。

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しても、データベースビューのアクセス許可は自動的に付与されません。のデータレイク管理者アカウントは、関連するデータベースとテーブルのクエリに使用する IAM ロールにアクセス SELECT 許可を付与 AWS Lake Formation する必要があります。ロールには少なくとも Data Analyst 権限が必要です。権限レベルの詳細については、「[Lake Formation ペルソナ](#)」と「[IAM 権限リファレンス](#)」を参照してください。SELECT 権限を付与する手順については、『AWS Lake Formation 開発者ガイド』の「[名前付きリソースメソッドを使用した Data Catalog 権限の付与](#)」を参照してください。

ステップ 6: サブスクライバーを作成する

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

による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、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のエンドポイントは関係ありません)。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。

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された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 管理者を指定するときは、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。

- 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。
- 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。
- [Delegate (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、Security Lake が新しいアカウントで自動的に有効になります AWS Organizations。委任管理者は、Security Lake API の [CreateDataLakeOrganizationConfiguration](#) オペレーションを使用するか、AWS CLI を使用している場合は、 [create-data-lake-organization-configuration](#) コマンドを実行して、この設定を有効にできます。リクエストでは、新しいアカウントの特定の設定を指定することもできます。

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での Amazon Route 53 Resolver クエリログ、AWS Security Hub 検出結果、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 ログおよび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、「[AWS サービスからのデータ収集](#)」を参照してください。

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

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は維持しつつ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 のサービスで AWS Organizations を利用する](#)」を参照してください。

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

委任された管理者の AWS アカウント が中断、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セキュリティログとイベントを収集できます。AWS リージョン リージョンごとに、データは異なる Amazon S3 バケットに保存されます。リージョンごとに異なるデータレイク設定 (たとえば、異なるソースと保持設定) を指定できます。1 つ以上のロールアップリージョンを定義して、複数のリージョンのデータを統合することもできます。

リージョン・ステータスのチェック

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

Console

リージョンのステータスを確認するには

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

API

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

たとえば、AWS CLI 次のコマンドは、現在のリージョンの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"
```

AWS CLI 次のコマンドは、指定したリージョン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]'
```

あるリージョンで Security Lake が有効になっ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 はセキュリティレイク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設定を返します。

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

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

リージョン設定の変更

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

Console

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

4. <リージョン>のすべてのアカウントのソースを上書きするチェック ボックスをオンにして、ここでの選択がこのリージョンの前の選択をオーバーライドすることを確認します。
5. [ストレージクラスを選択] で [トランジションを追加] を選択し、データ用の新しいストレージクラスを追加します。
6. [タグ] には、必要に応じてリージョンのタグを割り当てたり編集したりします。タグは、特定のリージョンのデータレイク設定など、AWS AWS アカウント 特定のタイプのリソースに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 [UpdateDataLake](#) API の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、コマンドを実行します。[update-data-lake](#) region パラメーターには、設定を変更するリージョンのリージョン コードを指定します。たとえば、米国東部 (バージニア北部) リージョンの場合は us-east-1 です。リージョンコードのリストについては、AWS 全般のリファレンスの「[Amazon Security Lake エンドポイント](#)」を参照してください。

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

たとえば、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。

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

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

Note

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

データレプリケーションの IAM ロール

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

これらのアクセス権限を付与するには、プレフィックス SecurityLake で始まる IAM ロールを作成し、以下のサンプルポリシーをアタッチします。Security Lake でロールアップリージョンを作成する場合、ロールの Amazon リソースネーム (ARN) が必要になります。このポリシーでは、sourceRegions は寄与リージョン、destinationRegions はロールアップリージョンです。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowReadS3ReplicationSetting",
      "Action": [
        "s3:ListBucket",
        "s3:GetReplicationConfiguration",
        "s3:GetObjectVersionForReplication",
```

```

    "s3:GetObjectVersion",
    "s3:GetObjectVersionAcl",
    "s3:GetObjectVersionTagging",
    "s3:GetObjectRetention",
    "s3:GetObjectLegalHold"
  ],
  "Effect": "Allow",
  "Resource": [
    "arn:aws:s3:::aws-security-data-lake-[[sourceRegions]]*",
    "arn:aws:s3:::aws-security-data-lake-[[sourceRegions]]*/*"
  ],
  "Condition": {
    "StringEquals": {
      "s3:ResourceAccount": [
        "{{bucketOwnerAccountId}}"
      ]
    }
  }
},
{
  "Sid": "AllowS3Replication",
  "Action": [
    "s3:ReplicateObject",
    "s3:ReplicateDelete",
    "s3:ReplicateTags",
    "s3:GetObjectVersionTagging"
  ],
  "Effect": "Allow",
  "Resource": [
    "arn:aws:s3:::aws-security-data-lake-[[destinationRegions]]*/*"
  ],
  "Condition": {
    "StringEquals": {
      "s3:ResourceAccount": [
        "{{bucketOwnerAccountId}}"
      ]
    }
  }
}
]
}

```

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

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

AWS Key Management Service (AWS KMS) のカスタマー管理キーを使用して 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権限に加えて次のアクセス権限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「[アクセス権限の設定](#)」を参照してください。

AWS Glue パーティションを登録する IAM ロール

この IAM ロールは、他のリージョンから複製された S3 オブジェクトのパーティションを登録するために Security Lake AWS Lambda AWS Glue が使用するパーティション更新機能の権限を付与します。このロールを作成しないと、サブスクライバーはそれらのオブジェクトからのイベントをクエリできません。

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

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

1. Lake Formation コンソール (<https://console.aws.amazon.com/lakeformation/>) を開きます。
2. 管理ユーザーとしてサインインする
3. [Lake Formation へようこそ] ウィンドウが表示されたら、ステップ 1 で作成または選択したユーザーを選択し、[開始する] を選択します。

4. [Lake Formation へようこそ] ウィンドウが表示されない場合は、以下の手順を実行して Lake Formation 管理者を設定します。
 1. ナビゲーションペインの [許可] で [管理ロールとタスク] を選択します。コンソールページの [データレイク管理者] セクションで、[管理者を選択] を選択します。
 2. データレイク管理者の管理ダイアログボックスの IAM ユーザーとロールで、作成した AmazonSecurityLakeMetaStoreManagerIAM ロールを選択し、[保存] を選択します。

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

ロールアップリージョンの追加

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

Note

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

Console

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

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

API

ロールアップリージョンをプログラマ的に追加するには、Security Lake API [UpdateDataLake](#) の操作を使用します。を使用している場合は AWS CLI、コマンドを実行します。[update-data-lake](#) 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。regionsreplicationConfiguration パラメータの配列で、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。

たとえば、ap-northeast-2 次のコマンドはロールアップリージョンとして設定されます。us-east-1 ap-northeast-2 リージョンはリージョンにデータを提供します。また、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 日に設定しています。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックslash (\) の行継続文字を使用しています。

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

Security Lake へのオンボーディング時にロールアップリージョンを追加することもできます。そのためには、[CreateDataLake](#) オペレーション (またはを使用する場合はコマンド) を使用します。AWS CLI [create-data-lake](#) 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、[参照してください。Amazon Security Lake の開始方法](#)

ロールアップリージョンの更新または削除

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

Console

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

4. ロールアップリージョンのコントリビューションリージョンを変更するには、ロールアップリージョンの行に更新されたコントリビューションリージョンを指定します。
5. ロールアップリージョンを削除するには、ロールアップリージョンの行で [削除] を選択します。
6. 完了したら、保存 を選択します。

API

ロールアップリージョンをプログラマ的に設定するには、Security Lake API [UpdateDataLake](#) のオペレーション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。[update-data-lake](#) 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 インドポイント](#)」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。

たとえば、us-east-1us-east-2 次のコマンドはとをコントリビューションリージョンとして設定します。ap-northeast-3 どちらのリージョンも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 の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

- [AWS サービスからのデータ収集](#)
- [カスタムソースからのデータ収集](#)

AWS サービスからのデータ収集

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

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

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。

i Tip

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

前提：アクセス許可

Security Lake AWS のサービスのソースとしてを追加するには、必要な権限が必要です。ソースの追加に使用するロールにアタッチされている AWS Identity and Access Management (IAM) ポリシーに、以下のアクションを実行する権限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、`s3:PutObject`および権限に関する以下の条件とリソーススコープがあることが推奨されます`S3:getObject`。

```
{
  "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、AWS のサービス AN からログとイベントを収集し、AWS Glue 正しいデータベースとテーブルに送信できます。

AWS KMS データレイクのサーバー側暗号化にキーを使用する場合は、の権限も必要です。kms:DescribeKey

CloudTrail イベントログ

AWS CloudTrail、AWS SDK、コマンドラインツール、AWS および特定のサービスを使用して行われた API 呼び出しを含む、アカウントの API 呼び出しの履歴が表示されます。AWS Management Console CloudTrail また、AWS サポートしているサービスの API を呼び出したユーザーやアカウント CloudTrail、呼び出し元のソース IP アドレス、呼び出しがいつ発生したかを特定できます。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 と Lambda CloudTrail CloudTrail の管理イベントとデータイベントに関連するログを収集できます。CloudTrail 管理イベント、S3 データイベント、および Lambda データイベントは、Security Lake の 3 つの独立したソースです。その結果、これらのいずれかを取り込みログソースとして追加すると、[sourceName](#) の値が異なります。管理イベント (コントロールプレーンイベントとも呼ばれる) は、社内のリソースに対して実行される管理操作に関する洞察を提供します。AWS アカウント CloudTrail データイベントはデータプレーン操作とも呼ばれ、内のリソースに対して、またはリソース内で実行されたリソース操作を示します 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 CloudTrail の管理イベントを収集するには、CloudTrail CloudTrail 読み取り/書き込み管理イベントを収集するマルチリージョン組織トレイルが少なくとも 1 つ必要です。トレイルのロギングが有効になっている必要があります。他のサービスでロギングを設定している場合は、ロギング設定を変更して Security Lake のログソースとして追加する必要はありません。Security Lake は、独立イベントストリームと重複イベントストリームでデータを直接取得します。

マルチリージョン証跡は、複数のリージョンから単一の Amazon Simple Storage Service (Amazon S3) バケットにログファイルを 1 AWS アカウントつのリージョンで配信します。CloudTrail コンソールまたはからマルチリージョンの記録をすでに管理している場合は AWS Control Tower、それ以上の操作は必要ありません。

- トレイルスルーの作成と管理については詳しくは CloudTrail、『AWS CloudTrail ユーザーガイド』の「[組織用の証跡の作成](#)」を参照してください。
- トレイルスルーの作成と管理については AWS Control Tower、『AWS Control Tower ユーザーガイド』AWS CloudTrailの「[AWS Control Tower ロギングアクション](#)」を参照してください。

CloudTrail イベントをソースとして追加すると、Security Lake CloudTrail は直ちに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、CloudTrail 独立した重複したイベントストリームから直接収集されます。

Security Lake CloudTrail はイベントを管理したり、CloudTrail 既存の設定に影響を与えたりすることはありません。CloudTrail イベントのアクセスと保存を直接管理するには、CloudTrail サービスコンソールまたは API を使用する必要があります。詳細については、『AWS CloudTrail ユーザーガイド』の「[CloudTrail イベント履歴によるイベントの表示](#)」を参照してください。

以下のリストは、Security Lake CloudTrail がイベントを OCSF GitHub 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リポジトリリンクを示しています。

GitHub OCSF イベントのリポジトリ - CloudTrail

- [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、セキュリティレイクは Elastic Kubernetes サービス (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 クラスターで発生する可能性のある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、独立した重複する監査ログストリームを通じて、Amazon EKS コントロールプレーンロギング機能から EKS 監査ログイベントを直接消費します。このプロセスでは、追加の設定は不要で、既存の Amazon EKS コントロールプレーンのログ設定 (存在する場合) に影響はありません。

ん。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。

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 OCSF リポジトリにある Amazon EKS 監査ログイベント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。

以下のリストは、セキュリティレイクが Route 53 のログを OCSF GitHub 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リポジトリリンクを示しています。

GitHub Route 53 ログ用の 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 の調査結果は、現在のセキュリティ体制を理解するのに役立ち、セキュリティ業界の標準やベストプラクティスに照らして環境をチェックすることができます。AWS Security Hub は、AWS のサービス他のサードパーティ製品との統合や Security Hub の制御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 セキュリティ結果フォーマット (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 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。

以下のリストは、Security Lake が Security Hub の結果を OCSF GitHub 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リポジトリリンクを示しています。

GitHub Security Hub 調査結果用の 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。Flow Logsの独立した重複ストリームを通じて、Amazon VPC から VPC Flow Logsを直接消費します。

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 GitHub 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リポジトリリンクを示しています。

GitHub VPC フローログ用の 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 のサービスをソースとして追加

AWS のサービスをソースとして追加すると、Security Lake はそこからセキュリティログとイベントの収集を自動的に開始します。以下の手順では、ネイティブにサポートされているものを Security Lake AWS のサービスのソースとして追加する方法を説明します。カスタムソースを追加する手順については、[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. AWS のサービス データを収集したいものを選択し、[Configure] 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、データソースの最新バージョンが Security Lake に取り込まれます。

Important

AWS 指定したリージョンで新しいバージョンのログソースを有効にするのに必要なロール権限がない場合は、Security Lake 管理者に問い合わせてください。詳細については、「[ロール権限の更新](#)」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。サブスクライバーの編集方法の詳細については、「[Amazon Security Lake 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、データ取り込みに使用してい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したい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 のサービスをソースとしてプログラムで追加するには、Security Lake [CreateAwsLogSource](#) API の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、コマンドを実行します。 [create-aws-log-source](#) `sourceName` および `regions` パラメータが必要です。オプションで、accountsソースの範囲を特定のものまたは特定のものに限定することもできます `sourceVersion`。

Important

コマンドに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指していると想定します。たとえば、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、エラーが表示されます。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="1.0"
```

ロール権限の更新

Console

あるバージョンのデータソースからデータを取り込むために必要なロール権限がない場合は、AmazonSecurityLakeMetaStoreManagerV2ソースからのデータを処理するためのロール権限を更新する必要があります。

手順に従ってロール権限を更新し、AWS 指定したリージョンの新しいバージョンのログソースからのデータを処理します。権限はfuture データソースリリースに自動的に適用されるため、これは1回限りのアクションです。

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. [ロール権限の更新] 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

あるバージョンのデータソースからデータを取り込むために必要なロール権限がない場合は、AmazonSecurityLakeMetaStoreManagerV2ソースからのデータを処理するためのロール権限を更新する必要があります。権限はfuture データソースリリースに自動的に適用されるため、これは1回限りのアクションです。

パーミッションをプログラムで更新するには、Security Lake [UpdateDataLakeAPI](#) のオペレーションを使用します。を使用して権限を更新するには AWS CLI、コマンドを実行します。 [update-data-lake](#)

ロールの権限を更新するには、[AmazonSecurityLakeMetastoreManager](#)ポリシーをロールにアタッチする必要があります。

AmazonSecurityLakeMetaStoreManager ロールを削除する。

Important

ロールの権限をに更新したらAmazonSecurityLakeMetaStoreManagerV2、データレイクが正しく機能することを確認してから、AmazonSecurityLakeMetaStoreManager古いロールを削除してください。ロールを削除する前に 4 時間以上待つことをおすすめします。

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

Lake Formation AmazonSecurityLakeMetaStoreManager コンソールから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/> にあるLake Formation の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、[管理者の役割とタスク] を選択します。
3. AmazonSecurityLakeMetaStoreManager各リージョンから削除します。

AWS のサービスをソースとして削除

アクセス方法を選択し、以下の手順に従って Security Lake AWS のサービス ソースとしてネイティブにサポートされているものを削除します。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、AWS のサービス ネイティブにサポートされているソースとして削除する場合にのみ使用できます。カスタムソースの削除については、[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。

- 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

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

Important

コマンドにパラメーターを指定しない場合、Security Lake は欠落しているパラメーターがセット全体を指していると想定します。たとえば、`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="1.0"
```

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="1.0"
```

上記の例は Linux、macOS、または Unix 向け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 (\) の行継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を取得しています。

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状況を確認するには

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

API

現在のリージョンのログ収集状況を確認するには、Security Lake API [GetDataLakeSources](#) の操作を使用してください。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、1 つ以上の AWS アカウント ID 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 は現在のリージョンのアカウントのスナップショットを返します。これには、Security Lake AWS がデータを収集しているソースや各ソースのステータスが含まれます。accounts パラメータを含めない場合、応答には現在のリージョンで Security Lake が設定されているすべてのアカウントのログ収集ステータスが含まれます。

たとえば、AWS CLI 次のコマンドは、現在のリージョンの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 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 (IAM) に作成します。このロールの権限境界は、AWS という管理ポリシーによって設定されます。 [AmazonSecurityLakePermissionsBoundary](#)
- ソースが Security Lake AWS Lake Formation に書き込むオブジェクトを整理するためのテーブルを作成します。
- AWS Glue ソースデータを分割するクローラーを設定します。クローラーはにテーブルを設定します。AWS Glue Data Catalog 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は、それぞれ固有の [Open Cybersecurity Schema Framework \(OCSF\) イベントクラス](#) を個別のソースとして配信する必要があります。Security Lake は、カスタムソースが S3 バケット内の指定された場所を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 — 同じ OCSF イベントクラスを Parquet 形式のオブジェクト内の各レコードに適用する必要があります。

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、および日付で分割する必要があります。パーティションデータパスはという形式です。 `bucket-name/source-location/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/source-location/region=us-west-2/accountId=123456789012/eventDay=20230428/

- 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 バケツの名前。
- source-location— S3 バケツのカスタムソースのプレフィックス。Security Lake は、特定のソースのすべての S3 オブジェクトをこのプレフィックスの下に格納します。プレフィックスは特定のソースに固有のものです。
- region— AWS リージョン データが書き込まれる場所。
- accountId— ソースパーティション内のレコードが属する 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 に書き込まれたオブジェクトは、レコードを 5 分間バッファリングする必要があります。バッファ期間に含まれるデータが多すぎて効率的にクエリできない場合、ファイルの平均サイズが 256 MB 未満であれば、カスタムソースは 5 分間のウィンドウに複数のレコードを書き込むことができます。スループットの低いカスタムソースは、5 分間の取り込み待ち時間を維持するために 5 分ごとに小さなオブジェクトを書き込むことができ、レコードを長期間バッファリングできます。

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、標準が推奨されます。

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。region ロールの名前は AmazonSecurityLake-Provider-{name of the custom source}-{region}、カスタムソースを追加するときの形式に従います。AWS リージョン 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。AWS KMS データレイクをカスタマーマネージドキーで暗号化した場合、Security Lake `kms:Decrypt` `kms:GenerateDataKey` は権限を含むポリシーをロールにアタッチします。このロールの権限境界は、AWS [AmazonSecurityLakePermissionsBoundary](#) という管理ポリシーによって設定されます。

トピック

- [アクセス許可の確認](#)
- [Security Lake バケットロケーションへの書き込みアクセスを許可する IAM ロールを作成する \(API のみのステップ\) AWS CLI](#)

アクセス許可の確認

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

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

- `glue:CreateCrawler`
- `glue:StopCrawler`
- `glue:CreateDatabase`
- `glue:CreateTable`
- `glue:StartCrawlerSchedule`
- `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、AWS Glue 正しい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。

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

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 [カスタムソースの追加](#) 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 を使用する予定がある場合、AWS CLI または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 バケットロケーション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 バケットロケーション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 のみのステップ) AWS CLI

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

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

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

必要な権限を付与するには、カスタムソースからのデータファイルの読み取りと Data Catalog AWS Glue クローラー AWS Glue のテーブルの作成/更新を許可する次のインラインポリシーを作成してロールに埋め込む必要もあります。

```
{
  "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して、AWS アカウント 外部 ID に基づいてロールを引き受けることを許可します。

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

カスタムソースを追加するリージョンの S3 バケットがカスタマーマネージドで暗号化されている場合は AWS KMS key、次のポリシーをロールと KMS キーポリシーに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 クローラーを起動する 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. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. AWS リージョン ページの右上隅にあるセレクターを使用して、カスタムソースを作成するリージョンを選択します。
3. ナビゲーションペインで [カスタムソース] を選択してから、[カスタムソースの作成] を選択します。
4. 「カスタムソースの詳細」セクションに、カスタムソースのグローバルに一意の名前を入力します。次に、カスタムソースが Security Lake に送信するデータの種別を説明する OCSF イベントクラスを選択します。
5. データの書き込み権限を持つAWS アカウントの場合、データレイクにログとイベントを書き込むカスタムソースの AWS アカウント ID と外部 ID を入力します。
6. サービスアクセスの場合は、新しいサービスロールを作成して使用するか、Security Lake に AWS Glueを呼び出すアクセス許可を付与する既存のサービスロールを使用します。
7. [作成] を選択します。

API

カスタムソースをプログラムで追加するには、Security Lake API [CreateCustomLogSource](#)の操作を使用します。AWS リージョン カスタムソースを作成したい場所でオペレーションを使用してください。AWS Command Line Interface (AWS CLI) を使用している場合は、[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 の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、「[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 \(OCSF\)](#)」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。AWS Glue [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`— AWS 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 ID と外部 ID を指定します。

次の例では、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、カスタムソースを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"]
```

カスタムソースデータを常に最新の状態に保ちます。AWS Glue

セキュリティレイクにカスタムソースを追加すると、AWS Glue セキュリティレイクはクローラーを作成します。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。

クローラーの実行方法については、『開発者ガイド』の「[クローラーのスケジュール設定](#)」を参照してください。AWS Glue AWS Glue

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. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. AWS リージョン ページの右上隅にあるセレクターを使用して、カスタムソースを削除したい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 API [DeleteCustomLogSource](#) の操作を使用します。AWS Command Line Interface (AWS CLI) を使用している場合は、コマンドを実行します。[delete-custom-log-source](#) カスタム ソースを削除する AWS リージョン 内の操作を使用します。

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

次の例では、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 [CreateSubscriber](#) API 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 [CreateSubscriber](#) API 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](#)
- [サブスクライバーの外部 ID を取得します。](#)
- [EventBridge API 送信先を呼び出す IAM ロールを作成する \(API および AWS CLI のみのステップ\)](#)

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権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 を使用するか、サブスクリバード AWS CLI を追加する予定がある場合は、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 を通じて Security Lake を使用している場合は AWS CLI、AWS Identity and Access Management (IAM) で API 送信先を呼び出し、オブジェクト通知を正しい HTTPS エンドポイントに送信するアクセス EventBridge 許可を Amazon に付与するロールを作成します。

この IAM ロールの作成が完了したら、サブスクリバード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。サブスクリバードが 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 サブスクリバード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 Hub 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 [Subscribers] (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。

現在選択されている としてリージョンが自動的に入力され AWS リージョン、変更することはできません。

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 EventBridge から HTTPS 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 に付与 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を識別、分類、および管理するのに役立ちます。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では、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へのアクセスに使用する AWS アカウント ID と外部 ID を指定します。
- にはsubscriber-name、サブスクライバーの名前を指定します。
- accessTypes の場合、S3 を指定します。

例 1

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

```
$ aws securitylake create-subscriber \  
--subscriber-identity {"accountID": 1293456789123,"externalId": 123456789012} \  
--sources [{"awsLogSource": {"sourceName": VPC_FLOW, sourceVersion": 1.0}}] \  
--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": 1.0}}] \  
--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 (\) の行連結文字を使用しています。

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

- デフォルトの通知方法 (HTTPS エンドポイント) をオーバーライドして Amazon SQS キューを作成するには、sqsNotificationConfigurationパラメータの値を指定します。

- HTTPS エンドポイントによる通知を希望する場合は、`httpsNotificationConfiguration` パラメータの値を指定します。
- `targetRoleArn` フィールドには、EventBridge API 送信先を呼び出すために作成した IAM ロールの ARN を指定します。

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

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

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

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

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 [Subscribers] (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 の [DeleteSubscriberNotification](#) 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、次の手順をスキップして[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にクエリアクセス
AWS CLI 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。

次の手順に従って、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. [Welcome to Lake Formation] (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、[Get started] (開始する) を選択します。
4. [Welcome to Lake Formation] (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 [Permissions] (許可) で [Administrative Roles and tasks] (管理ロールとタスク) を選択します。[Data lake administrators] (データレイク管理者) セクションで、[Choose administrators] (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リイバーの作成

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リイバーを作成します AWS リージョン。サブスクリイバーは、AWS リージョン それで作成された からのみデータをクエリできます。サブスクリイバーを作成するには、サブスクリイバーの AWS アカウント ID と外部 ID が必要です。外部 ID は、サブスクリ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権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 Hub コンソール <https://console.aws.amazon.com/securitylake/>を開きます。

委任管理者アカウントにサインインします。

2. ページの右上隅にある AWS リージョン セレクターを使用して、サブスクリイバー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 [Subscribers] (サブスクリイバー) を選択します。
4. 「サブスクリイバー」ページで、「サブスクリイバーを作成」を選択します。
5. サブスクリイバーの詳細には、サブスクリイバー名とオプションの説明を入力します。

現在選択されている としてリージョンが自動的に入力され AWS リージョン、変更することはできません。

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リ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を識別、分類、および管理するのに役立ちます。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

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。
- `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、指定し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: 1.0}}] \  
--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 [クエリアクセス権を持つサブスクライバーを作成する](#)と AWS CLI、Security Lake は AWS Resource Access Manager () で [リソース共有](#)を作成して、関連する Lake Formation テーブルに関する情報をサブスクライバーと共有しますAWS RAM。

クエリアクセス権を持つサブスクライバに特定の種類の編集を行うと、Security Lake は新しいリソース共有を作成します。詳細については、「[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、次の手順に従って 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招待の承諾 AWS RAM」](#)を参照してください。
 - API の場合は、[GetResourceShareInvitations](#) API を呼び出します。resourceShareArnとresourceShareNameでフィルタリングして、正しいリソースシェアを見つけてください。[AcceptResourceShareInvitation](#) API を使用して招待を承諾します。

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

クロスアカウント権限の付与については、「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、手順に従って現在の 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手順に従ってください。

サブスクライバー名を編集するには

1. Security Hub 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 [Subscribers] (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 Hub 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 [Subscribers] (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 Hub 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 [Subscribers] (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 \(サブスクライバーステップ\)](#)の手順 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 Hub 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 [Subscribers] (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、以前の AWS RAM リソース共有が自動的に削除されるため、以前のプリンシパルはログソースとイベントソースを使用できません。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および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 Hub 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 [Subscribers] (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

クエリアクセス権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と E メールアドレスを指定します。

- 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** – このアクションは、以前のプリンシパルがログソースとイベントソースを消費できないように、以前の AWS RAM リソース共有を削除します。Security Lake は新しいリソース共有を作成します。サブスクライバーは、新しいプリンシパルの認証情報を使用して、新しいリソース共有を受け入れ、共有テーブルへのリソースリンクを作成する必要があります。これにより、新しいプリンシパルは共有リソースにアクセスできます。ステップについては、「[クロスアカウントテーブル共有セットアップ \(サブスクライバーステップ\)](#)」のステップ 1 と 2 を参照してください。

外部 ID とプリンシパルを更新するには、[クロスアカウントテーブル共有セットアップ \(サブスクライバーステップ\)](#)の手順 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

- [ソースバージョン 1 の Security Lake クエリ](#)
- [ソースバージョン 2 の Security Lake クエリ](#)

ソースバージョン 1 の Security Lake クエリ

次のセクションでは、Security Lake からのデータのクエリに関するガイダンスと、ネイティブにサポートされている AWS ソースのクエリ例をいくつか示します。これらのクエリは、特定の のデータを取得するように設計されています AWS リージョン。これらの例では us-east-1 (米国東部 (バージニア北部)) を使用しています。さらに、サンプルクエリでは最大 25件のレコードを返すLIMIT 25パラメータを使用しています。このパラメータは省略することも、好みに応じて調整することもできます。その他の例については、「[Amazon Security Lake OCSF 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 のデータ CloudTrail 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 アクティビティの一覧表示

次の例では、`20230301` (2023 年 3 月 1 日) 以降に記録されたソース IP `192.0.2.1` からのすべての CloudTrail アクティビティを、`#### 1` のテーブル `cloud_trail_mgmt_1_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_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_0
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_0
WHERE eventDay > cast(date_format(current_timestamp - INTERVAL '30' day, '%Y%m%d
%H') as varchar)
AND src_endpoint.ip = '192.0.2.1'
```

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

CloudTrail データのクエリ例

AWS CloudTrail は、で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 のサービス。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。

CloudTrail データのクエリの例を次に示します。

過去 7 日間の AWS のサービス に対する不正な試み

```
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 日間に失敗した CloudTrail レコードのリスト

```

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

```

Route 53 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 では、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および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
```

ソースバージョン 2 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と、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 のデータ CloudTrail 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

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 年 3 月 1 日) 以降に記録されたソース IP **192.0.2.1** からのすべての Amazon VPC アクティビティを、**#### 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 accountid を使用してパーティショニングを実装するようになりました。一方、Security Lake

1.0 では region、eventDay、および accountid パラメータを使用してパーティショニングを実装しています。

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

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

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

time_dt の一般的な値は次の通りです。

過去 1 年間に発生したイベント

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

過去 1 か月に発生したイベント

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

過去 30 日間に発生したイベント

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

過去 12 時間に発生したイベント

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

過去 5 分間に発生したイベント

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

7 ~ 14 日前に発生したイベント

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

特定の日付以降に発生するイベント

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

例: 2023 年 3 月 1 日 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_0
 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_0
 WHERE eventDay > cast(date_format(current_timestamp - INTERVAL '30' day, '%Y%m%d%H') as varchar)
 AND src_endpoint.ip = '192.0.2.1'
 ORDER BY time desc
 LIMIT 25
```

Security Lake オブザーバビリティのクエリ

オブザーバビリティは、Security Lake 2.0 で利用可能になった新機能です。オブザーバビリティオブジェクトは、イベント内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引き出すことができます。

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、その他のIOCタイプの情報などに制限できます。

これは、オブザーバビリティ配列を使用して、VPC フローと IP 値「172.01.02.03」を含む Route53 テーブル間で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 データのクエリ例

AWS CloudTrail は、で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 のサービス。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。

CloudTrail データのクエリの例を次に示します。

AWS のサービス 過去 7 日間の に対する不正な試み

```
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_mgrm
```

```

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_mgrn
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_mgrn
WHERE time_dt BETWEEN CURRENT_TIMESTAMP - INTERVAL '7' DAY AND CURRENT_TIMESTAMP
AND api.service.name = 'iam.amazonaws.com'
ORDER BY time desc
LIMIT 25

```

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

```

SELECT
    actor.user.uid,
    actor.user.uid_alt,

```

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

過去 7 日間に失敗した CloudTrail レコードのリスト

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

Route 53 Resolver クエリログのクエリ例

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

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 の検出結果に関するクエリ

Security Hub では、 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および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_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_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 のクエリ例

Amazon EKS Logs 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 ログに直接監査ログと診断 CloudWatch ログを提供します。これらのログにより、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。

EKS ログのクエリ例を次に示します。

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 「」に関連付けられた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
```

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-lake](#) lifecycleConfiguration 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 CLI [update-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slash (\) の行継続文字を使用しています。

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

ロールアップリージョン

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

ロールアップリージョン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の 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ン管理基準を定義し、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 [GitHub](#) でホストされています。

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

OCSF イベントクラス

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

OCSFソースの識別

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

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

ソース	metadata. product.name	metadata. product.v endor_name	metadata. product.f eature.name	class_name	metadata. version
CloudTrail Lambda デー タイイベント	CloudTrai l	AWS	Data	API Activity	1.0.0-rc. 2
CloudTrail 管 理イベント	CloudTrai l	AWS	Managemen t	API Activity, Authentic ation , また は Account Change	1.0.0-rc. 2
CloudTrail S3 データイベン ト	CloudTrai l	AWS	Data	API Activity	1.0.0-rc. 2
Route 53	Route 53	AWS	Resolver Query Logs	DNS Activity	1.0.0-rc. 2
Security Hub	Security Hub	AWS	Security Hub ProductNa me 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.v endor_name	metadata. product.f eature.name	class_name	metadata. version
CloudTrail Lambda デー タイイベント	CloudTrai l	AWS	Data	API Activity	1.1.0
CloudTrail 管 理イベント	CloudTrai l	AWS	Managemen t	API Activity, Authentic ation , また は Account Change	1.1.0
CloudTrail S3 データイベン ト	CloudTrai l	AWS	Data	API Activity	1.1.0
Route 53	Route 53	AWS	Resolver Query Logs	DNS Activity	1.1.0
Security Hub	AWS Security Finding 形 式 (ASFF) ProductNa me の値と一 致します	AWS Security Finding 形 式 (ASFF) CompanyNa me の値と一 致します	ASFF featureNa me の値 と一致 ProductFi elds	Vulnerabi lity Finding, Complianc e 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.v endor_name	metadata. product.f eature.name	class_name	metadata. version
EKS 監査ロ グ	Amazon EKS	AWS	Elastic Kubernet e Service	API Activity	1.1.0

Security Lake との統合

Amazon セキュリティレイクは、AWS のサービス 他の製品やサードパーティ製品と統合できます。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、どのサードパーティ製品が Security Lake AWS のサービス と統合されるかを説明します。

トピック

- [AWS のサービス セキュリティレイクとの統合](#)
- [Security Lake とのサードパーティー統合](#)

AWS のサービス セキュリティレイクとの統合

Amazon AWS のサービスセキュリティレイクは他のものと統合されています。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、からのソースデータを直接クエリして読み取ります。 AWS Lake Formation
- Apache Parquet 形式のデータを読み取ることができる (Security Hub やその他の [ネイティブサポートソース](#)では、Security Lake がこれを自動的に処理します)
- OCSF スキーマのデータを読み取ることができる (Security Hub やその他の [ネイティブサポートソース](#)では、Security Lake がこれを自動的に処理します)

次のセクションでは、AWS のサービス どのセキュリティレイクが統合され、各統合がどのように機能するか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、組織全体のサービスとしてのソフトウェア (SaaS) アプリケーションを接続するコード不要のサービスです。これにより、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し、保護できます。

Security Lake が調査結果を受け取る方法 AppFabric

AppFabric 監査ログデータをセキュリティレイクに送信するには、送信先として Amazon Kinesis Data Firehose を選択し、OCSF スキーマと Apache Parquet 形式でデータをセキュリティレイクに配信するように Kinesis Data Firehose を設定します。

前提条件

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

調査結果をセキュリティレイクに送信してください。 AppFabric

前述の前提条件を満たした後に Security Lake AppFabric に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 AppFabric にカスタムソースとして追加する必要があります。カスタムソースを追加する手順については、[カスタムソースからのデータ収集](#)を参照してください。

AppFabric セキュリティレイクでのログの受信を停止します。

AppFabric 監査ログの受信を停止するには、Security Lake コンソール、Security Lake API を使用するか AWS CLI 、 AppFabric カスタムソースとして削除することができます。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。

との統合 AWS Security Hub

統合タイプ: ソース

[AWS Security Hub](#)現在のセキュリティ状態を包括的に把握でき、AWS セキュリティ業界の標準や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、さまざまなサービス、およびサポートされているサードパーティパートナー製品からセキュリティデータを収集し、セキュリティの傾向を分析し、最優先のセキュリティ問題を特定するのに役立ちます。

Amazon Security LakeとSecurity Hub を統合すると、Security Hub 結果がSecurity Lakeに表示されます。Security Hub の調査結果は Security Lake のサブスクライバーが利用できる情報源となり、セキュリティ状況の分析に役立ちます。

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。調査結果の中には、AWS 他のサービスやサードパーティのパートナーによって検出された問題から得られるものもあります。Security Hub には、セキュリティの問題を検出し、結果を生成するために使用するコントロールと呼ばれる一連のルールもあります。

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

Security Lake はSecurity Hub 結果を受け取り、それを[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。

前提条件

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 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 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AWS のサービスをソースとして追加](#)。

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

セキュリティハブの結果の受信を停止するには、セキュリティハブコンソール、Security Hub API、またはを使用できます AWS CLI。

「[統合からの結果のフローの無効化と有効化 \(コンソール\)](#)」または AWS Security Hub ユーザーガイドの「[統合からの結果のフローの無効化 \(Security Hub API、AWS CLI\)](#)」を参照してください。

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、からのソースデータを直接クエリして読み取ります。AWS Lake Formation
- Apache Parquet 形式でデータを読み取ることができます
- OCSF スキーマのデータを読み取ることができる。

サービス統合は、AWS のサービス セキュリティレイクやその他の機能を組織に実装するのに役立ちます。また、レポート、分析、その他のユースケースを支援できます。

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

パートナーインテグレーションとしての追加や Security Lake パートナーになることをリクエストするには、<securitylake-partners@amazon.com> にメールを送信してください。

結果をに送信するサードパーティのインテグレーションを使用している場合 AWS Security Hub、Security Lake の Security Hub インテグレーションが有効になっていれば、Security Lake でそれらの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てください。最新情報については、購読者のマニュアルを参照してください。

Accenture – MxDR

統合タイプ: サブスクライバー、サービス

Accenture's MxDR と Security Lake の統合により、ログやイベントのリアルタイムのデータ取り込み、マネージド型の異常検知、脅威ハンティング、セキュリティ運用が可能になります。これにより、分析やマネージド・ディテクション・アンド・レスポンス (MDR) が容易になります。

サービス統合として、Accentureは組織への Security Lake の実装にも役立ちます。

[統合ドキュメンテーション](#)

Aqua Security

統合タイプ: ソース

Aqua Securityをカスタム ソースとして追加して、監査イベントを Security Lake に送信できます。監査イベントは OCSF スキーマと Parquet 形式に変換されます。

[統合ドキュメンテーション](#)

Barracuda – Email Protection

統合タイプ: ソース

Barracuda Email Protectionは、新しいフィッシングメール攻撃が検出されたときに Security Lake にイベントを送信できます。これらのイベントは、データレイク内の他の Security データと一緒に受信できます。

[統合ドキュメンテーション](#)

Booz Allen Hamilton

統合タイプ: サービス

サービス統合として、Booz Allen Hamilton はデータと分析を Security Lake サービスと融合することにより、サイバーセキュリティに対してデータ駆動型のアプローチを採用しています。

[パートナーリンク](#)

ChaosSearch

統合タイプ: サブスクライバー

ChaosSearchは、Elasticsearch や SQL などのオープン API、またはネイティブに含まれる Kibana UI や Superset UI を使用して、ユーザーにマルチモデル データ アクセスを提供します。Security Lake データを ChaosSearch で保持制限なしで使用して、監視、アラート、脅威の探索を行うことができます。これにより、今日の複雑なセキュリティ環境や持続的な脅威に立ち向かうことができます。

[統合ドキュメンテーション](#)

Cisco Security – Secure Firewall

統合タイプ: ソース

Cisco Secure Firewall を Security Lake と統合することにより、構造化されたスケーラブルな方法でファイアウォールログを保存できます。シスコの eNcore クライアントは、ファイアウォール管理センターからファイアウォールログをストリーミングし、OCSF スキーマへのスキーマ変換を実行して Security Lake に保存します。

[統合ドキュメンテーション](#)

Claroty – xDome

統合タイプ: ソース

Claroty xDome ネットワーク内で検出されたアラートを最小限の設定で Security Lake に送信します。柔軟かつ迅速な導入オプションにより、xDome はネットワーク内の拡張されたモノのインターネット (XIoT) 資産 (IoT、IIoT、BMS 資産で構成される) を保護しながら、脅威の初期兆候を自動的に検出できます。

[統合ドキュメンテーション](#)

CMD Solutions

統合タイプ: サービス

CMD Solutionsは、設計、自動化、継続的保証プロセスを通じてセキュリティを早期かつ継続的に統合することで、企業の俊敏性を高めるのに役立ちます。サービス統合として、CMD Solutionsは、組織に Security Lake を実装するのに役立ちます。

[パートナーリンク](#)

Confluent – Amazon S3 Sink Connector

統合タイプ: ソース

Confluentは、完全に管理された事前構築済みのコネクタを使用して、データ統合を自動的に接続、設定、調整します。これで、Confluent S3 Sink Connectorは、未加工のデータをネイティブの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、API、マイクロサービスの脆弱性をリアルタイムで検出するIASTツールです。Assess は Security Lake と統合されているため、すべてのワークロードを一元的に可視化できます。

[統合ドキュメンテーション](#)

Cribl – Search

統合タイプ: サブスクリイバー

Cribl Search を使用して Security Lake データを検索できます。

[統合ドキュメンテーション](#)

Cribl – Stream

統合タイプ: ソース

Cribl Streamを使用すると、サポートされている Criblのサードパーティ ソースから OCSF スキーマの Security Lake にデータを送信できます。

[統合ドキュメンテーション](#)

CrowdStrike – Falcon Data Replicator

統合タイプ: ソース

この統合は、継続的なストリーミング ベースで CrowdStrike Falcon Data Replicatorからデータを取得し、そのデータを OCSF スキーマに変換して、Security Lake に送信します。

[統合ドキュメンテーション](#)

CyberArk – Unified Identify Security Platform

統合タイプ: ソース

CyberArk Audit Adapterは、CyberArk Identity Security Platformセキュリティイベントを収集し、OCSF スキーマ内の Security Lake AWS Lambda にデータを送信する関数です。

[統合ドキュメンテーション](#)

Darktrace – Cyber AI Loop

統合タイプ: ソース

Darktraceと Security Lake の統合により、Security Lake に Darktrace の自己学習機能が生じます。Cyber AI Loopからの洞察は、他のデータ ストリームや組織のセキュリティ スタックの要素と関連付けることができます。統合により、Darktrace モデル違反がセキュリティの検出結果として記録されます。

[統合ドキュメンテーション \(ドキュメンテーションを確認するには、Darktrace ポータルにサインインします\)](#)

Datadog

統合タイプ: サブスクリイバー

Datadog Cloud SIEMSecurity Lake 内のデータなど、クラウド環境に対する脅威をリアルタイムで検出し、セキュリティ チームを 1 DevOps 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 機能で構成される CAE スイートは、Security Lake 内の OCSF 形式のデータに対して実行されるモデルに基づいて、実用的なインサイトを自動的に提供します。

サービス統合として、Deloitte は組織への Security Lake の実装にも役立ちます。

[統合ドキュメンテーション](#)

Devo

統合タイプ: サブスクライバー

Devoコレクターは Security Lake AWS からの取り込みをサポートします。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— Alsaac (旧 Atos)

統合タイプ: サブスクライバー

Alsaac MDRこのプラットフォームは、Security Lake の OCSF スキーマに取り込まれた VPC フローログを使用し、AI モデルを利用して脅威を検出します。

[統合ドキュメンテーション](#)

ExtraHop – Reveal(x) 360

統合タイプ: ソース

IOC の検出を含むネットワーク データを ExtraHop Reveal(x) 360 から OCSF スキーマの Security Lake に統合することで、ワークロードとアプリケーションのセキュリティを強化できます。

[統合ドキュメンテーション](#)

Falcosidekick

統合タイプ: ソース

Falcosidekickは、Falcoのイベントを収集してSecurity Lakeに送信します。この統合は、OCSF スキーマを使用してセキュリティイベントをエクスポートします。

[統合ドキュメンテーション](#)

Gigamon – Application Metadata Intelligence

統合タイプ: ソース

Gigamon Application Metadata Intelligence (AMI)オブザーバビリティ、SIEM、ネットワークパフォーマンス監視ツールに重要なメタデータ属性を付与します。これにより、アプリケーションの可視性が向上し、パフォーマンスのボトルネック、品質上の問題、潜在的なネットワークセキュリティリスクを特定できます。

[統合ドキュメンテーション](#)

Hoop Cyber

統合タイプ: サービス

Hoop Cyber FastStart には、データソースの評価、優先順位付け、データソースのオンボーディングが含まれ、Security Lake を通じて提供される既存のツールや統合を使用してデータをクエリできるようになっています。

[パートナーリンク](#)

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 コンシューミングログ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は、Kyndryl分析目的で Security Lake AWS CloudTrail から管理イベントを取り込みます。

サービス統合として、Kyndryl は組織に Security Lake を実装するのにも役立ちます。

[統合ドキュメンテーション](#)

Lacework – Polygraph

統合タイプ: ソース

Lacework Polygraph® Data Platformデータソースとして Security Lake と統合し、環境全体の脆弱性、設定ミス、既知および未知の脅威に関するセキュリティ調査結果を提供します。AWS

[統合ドキュメンテーション](#)

Laminar

統合タイプ: ソース

Laminarは、データセキュリティイベントを OCSF スキーマで Security Lake に送信し、インシデント対応や調査など、その他の分析ユースケースで利用できるようにします。

[統合ドキュメンテーション](#)

MegazoneCloud

統合タイプ: サービス

MegazoneCloud はクラウド コンサルティング サービスを専門としており、組織に Security Lake を導入する方法を理解するのに役立ちます。Security Lake を統合 ISV ソリューションと接続して、カスタムタスクを構築し、顧客のニーズに関連するカスタマイズされたインサイトを構築します。

[統合ドキュメンテーション](#)

Monad

統合タイプ: ソース

Monadは、データを自動的に OCSF スキーマに変換し、Security Lake データレイクに送信します。

[統合ドキュメンテーション](#)

NETSCOUT – Omnis Cyber Intelligence

統合タイプ: ソース

Security Lake と統合することで、NETSCOUT は、サイバー脅威、セキュリティリスク、アタックサーフェスの変化など、企業内で何が起きているかについて、セキュリティに関する検出

結果と詳細なセキュリティに関するインサイトのカスタムソースになります。これらの検出結果は、NETSCOUT CyberStreams および Omnis Cyber Intelligence によってお客様のアカウントで生成され、OCSF スキーマで Security Lake に送信されます。取り込まれたデータは、フォーマット、スキーマ、パーティショニング、パフォーマンス関連の要素など、Security Lake ソースのその他の要件やベストプラクティスも満たしています。

[統合ドキュメンテーション](#)

Netskope – CloudExchange

統合タイプ: ソース

Netskopeセキュリティ関連のログや脅威情報を Security Lake と共有することで、セキュリティ体制の強化に役立ちます。Netskope結果はプラグインとともに Security Lake に送信されます。CloudExchangeプラグインは、AWS ローカルデータセンター内またはローカルデータセンター内の docker 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: サブスクリイバー

New Relic ONEは Lambda ベースのサブスクリイバーアプリケーションです。これはアカウントにデプロイされ、Amazon SQS によってトリガーされ、New Relic ライセンスキーを使用してデータを New Relic に送信します。

[統合ドキュメンテーション](#)

Okta – Workforce Identity Cloud

統合タイプ: ソース

OktaAmazon EventBridge インテグレーションを通じて OCSF スキーマのセキュリティレイクにアイデンティティログを送信します。Okta System Logsin OCSF スキーマは、セキュリティチームやデータサイエンティストチームがオープンソース標準によってセキュリティイベントをクエリするのに役立ちます。Okta から標準化された OCSF ログを生成すると、一貫したスキーマのもとで監査活動を行い、認証、承認、アカウント変更、エンティティ変更に関するレポートを生成するのに役立ちます。

[統合ドキュメンテーション](#)

[AWS CloudFormation Security Lake Okta にカスタムソース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

Orcaエージェントレスの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および応答 (CDR) イベントを送信することで Security Lake AWS と統合します。

[統合ドキュメンテーション \(Orca ポータルにサインインしてドキュメンテーションを確認\)](#)

Palo Alto Networks – Prisma Cloud

統合タイプ: ソース

Palo Alto Networks Prisma Cloudクラウドネイティブ環境の VM 全体の脆弱性検出データを集約し、Security Lake に送信しま

す。<#####securitylake-partners@amazon.com #####>

[統合ドキュメンテーション](#)

Ping Identity – PingOne

統合タイプ: ソース

PingOneは、アカウント変更アラートを OCSF スキーマと Parquet 形式で Security Lake に送信し、アカウントの変更を検出して対処できるようにします。

[統合ドキュメンテーション](#)

PwC – Fusion center

統合タイプ: サブスクリイバー、サービス

PwCは知識と専門知識を駆使して、クライアントの個々のニーズを満たすフュージョンセンターの導入を支援します。Amazon Security Lake 上に構築されたフュージョンセンターでは、さまざまなソースからのデータを組み合わせて、一元化されたほぼリアルタイムのビューを作成できます。

[統合ドキュメンテーション](#)

Rapid7 – InsightIDR

統合タイプ: サブスクライバー

InsightIDRRapid7SIEM/XDR ソリューションでは、Security Lakeにログを取り込み、脅威の検出や疑わしいアクティビティの調査を行うことができます。

[統合ドキュメンテーション](#)

RipJar – Labyrinth for Threat Investigations

統合タイプ: サブスクライバー

Labyrinth for Threat Investigationsは、きめ細かなセキュリティ、適応性の高いワークフロー、レポート機能を備えた、データフュージョンに基づく大規模な脅威調査への全社的なアプローチを提供します。

[統合ドキュメンテーション](#)

Sailpoint

統合タイプ: ソース

インテグレーション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: サブスクリイバー

SentinelOne Singularity™ XDR プラットフォームは、Amazon Elastic Compute Cloud (Amazon EC2)、Amazon Elastic Container Service (Amazon ECS)、Amazon Elastic Kubernetes Service (Amazon EKS) を含む、オンプレミスおよびパブリッククラウドインフラストラクチャ上で実行されるエンドポイント、アイデンティティ、クラウドワークロードに対して、リアルタイムの検出と応答を拡張します。

[統合ドキュメンテーション \(SentinelOne ポータルにサインインしてドキュメンテーションを確認\)](#)

Sentra – Data Lifecycle Security Platform

統合タイプ: ソース

Sentra スキャン インフラストラクチャをアカウントにデプロイすると、Sentra は結果を取得して SaaS に取り込みます。これらの検出結果は、Sentra が保存し、後でクエリのために OCSF スキーマの Security Lake にストリーミングするメタデータです。

[統合ドキュメンテーション](#)

SOC Prime

統合タイプ: サブスクリイバー

SOC Prime Amazon OpenSearch Service と Amazon Athena を通じて Security Lake と統合し、ゼロトラストのマイルストーンに基づくスマートなデータオーケストレーションと脅威ハンティングを促進します。SOC Prime セキュリティチームが脅威の可視性を高め、大量のアラートを発生させずにインシデントを調査できるようにします。OCSF スキーマの Athena と OpenSearch Service に自動的に変換できる再利用可能なルールとクエリにより、開発時間を節約できます。

[統合ドキュメンテーション](#)

Splunk

統合タイプ: サブスクリイバー

Amazon Web Services Splunk AWS アドオン (AWS) は、セキュリティレイクからの取り込みをサポートします。この統合により、Security Lake から OCSF スキーマのデータを購読することで、脅威の検出、調査、対応を迅速に行うことができます。

[統合ドキュメンテーション](#)

Stellar Cyber

統合タイプ: サブスクライバー

Stellar Cyber は Security Lake からログを消費し、レコードを Stellar Cyber データ レイクに追加します。このコネクタは OCSF スキーマを使用します。

[統合ドキュメンテーション](#)

Sumo Logic

統合タイプ: サブスクライバー

Sumo Logic Security Lake からのデータを利用して AWS、オンプレミス環境とハイブリッドクラウド環境を幅広く可視化します。Sumo Logic は、セキュリティチームがすべてのセキュリティツールを包括的に可視化し、自動化し、脅威を監視できるようにします。

[統合ドキュメンテーション](#)

Swimlane – Turbine

統合タイプ: サブスクライバー

Swimlane は、OCSF スキーマの Security Lake からデータを取り込み、ローコードプレイブックとケース管理を通じてデータを送信して、より迅速な脅威の検出、調査、インシデント対応を促進します。

[統合ドキュメンテーション \(Swimlane ポータルにサインインしてドキュメンテーションを確認\)](#)

Sysdig Secure

統合タイプ: ソース

Sysdig Secure'sクラウドネイティブアプリケーション保護プラットフォーム (CNAPP) は、セキュリティイベントを Security Lake に送信して、監視を最大化し、調査を効率化し、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: サブスクリイバー

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を関連付けることで、隠れた脅威を発見します。このプラットフォームはコンテキストに即した脅威インテリジェンスで強化されており、脅威の検出と調査において正確性やインサイトを提供します。

[統合ドキュメンテーション](#)

Tines – No-code security automation

統合タイプ: サブスクライバー

Tines No-code security automation は、Security Lake に一元管理されているセキュリティデータを活用することで、より正確な意思決定を支援します。

[統合ドキュメンテーション](#)

Torq – Enterprise Security Automation Platform

統合タイプ: ソース、サブスクライバー

Torq は、カスタム ソースおよびサブスクライバーの両方として Security Lake とシームレスに統合します。Torq は、シンプルなノーコードプラットフォームを使用してエンタープライズ規模の自動化とオーケストレーションを実装するのに役立ちます。

[統合ドキュメンテーション](#)

Trellix – XDR

統合タイプ: ソース、サブスクライバー

オープン XDR プラットフォームとして、Trellix XDR は Security Lake の統合をサポートしています。Trellix XDR は、セキュリティ分析のユースケースに OCSF スキーマのデータを活用できます。また、Trellix XDR のセキュリティ イベントの 1,000 以上のソースを使用して Security Lake データレイクを強化することもできます。これにより、検出機能と対応機能を環境に合わせて拡張できます。AWS 取り込まれたデータは他のセキュリティリスクと相関関係があり、リスクにタイムリーに対応するために必要なプレイブックが得られます。

[統合ドキュメンテーション](#)

Trend Micro – CloudOne

統合タイプ: ソース

Trend Micro CloudOne Workload Securityは、Amazon Elastic Compute Cloud (EC2) インスタンスから Security Lake に以下の情報を送信します。

- DNS クエリアクティビティ

- ファイルアクティビティ
- ネットワークアクティビティ
- プロセスアクティビティ
- レジストリ値アクティビティ
- ユーザーアカウントアクティビティ

[統合ドキュメンテーション](#)

Uptycs – Uptycs XDR

統合タイプ: ソース

Uptycsは、オンプレミスやクラウドのアセットから Security Lake に大量のデータを OCSF スキーマで送信します。データには、エンドポイントやクラウドワークロードからの行動上の脅威の検出、異常検知、ポリシー違反、危険なポリシー、設定ミス、脆弱性が含まれます。

[統合ドキュメンテーション](#)

Vectra AI – Vectra Detect for AWS

統合タイプ: ソース

を使用するとVectra Detect for AWS、専用のテンプレートを使用して、忠実度の高いアラートをカスタムソースとして Security Lake に送信できます。AWS CloudFormation

[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: サブスクリイバー

Wazuh は、ユーザーデータを安全に処理し、各ソースにクエリアクセスを提供し、クエリコストを最適化することを目的としています。

[統合ドキュメンテーション](#)

Wipro

統合タイプ: ソース、サービス

この統合により、Wipro Cloud Application Risk Governance (CARG) プラットフォームからデータを収集して、クラウドアプリケーションと企業全体のコンプライアンス態勢を一元的に把握できます。

サービス統合として、Wiproは、組織に Security Lake を実装するのにも役立ちます。

[統合ドキュメンテーション](#)

Wiz – CNAPP

統合タイプ: ソース

Wiz と Security Lake の統合により、拡張可能で正規化されたセキュリティデータ交換用に設計されたオープンソース標準である OCSF スキーマを活用することで、単一のセキュリティデータレイクでのクラウドセキュリティデータの収集が容易になります。

[統合ドキュメンテーション \(Wiz ポータルにサインインしてドキュメンテーションを確認\)](#)

Zscaler – Zscaler Posture Control

統合タイプ: ソース

クラウドネイティブアプリケーション保護プラットフォームである Zscaler Posture Control™ は、セキュリティ検出結果を OCSF スキーマで Security Lake に送信します。

[統合ドキュメンテーション](#)

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](#)
- [Amazon Security Lake におけるデータ保護](#)
- [Amazon SECURITY Lake のコンプライアンス検証](#)
- [Security Lake のセキュリティのベストプラクティス](#)
- [Amazon Security Lake の耐障害性](#)
- [Amazon Security Lake のインフラストラクチャセキュリティ](#)
- [Security Lake での構成と脆弱性の分析](#)
- [Amazon Security Lakeのモニタリング](#)

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 のサービス するのに役立つです。IAM 管理者は、Security Lake リソースの使用を認証 (サインイン) および承認 (アクセス許可を持つ) できるユーザーを制御します。IAM は、追加料金なしで AWS のサービス 使用できる です。

トピック

- [対象者](#)
- [アイデンティティを使用した認証](#)
- [ポリシーを使用したアクセスの管理](#)
- [Amazon Security Lake と IAM との連携方法](#)
- [Amazon Security Lakeのアイデンティティベースのポリシー例](#)
- [AWS Amazon Security Lake の マネージドポリシー](#)
- [Amazon Security Lake のサービスにリンクされたロール](#)

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、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。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を点検して、IAM の基本概念を理解してください。お客様の会社で Security Lake で IAM を利用する方法の詳細については、「[Amazon Security Lake と IAM との連携方法](#)」を参照してください。

IAM 管理者 - IAM 管理者は、Security Lake へのアクセスを管理するポリシーの作成方法を詳しく確認する必要がある場合があります。IAM で使用できる Security Lake のアイデンティティベースのポ

リシーの例を表示するには、「[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。

アイデンティティを使用した認証

認証は、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 (AWS IAM Identity Center) から提供された認証情報を使用して、フェデレーテッド ID AWS として にサインインできます。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、フェデレーテッド ID の例です。フェデレーテッドアイデンティティとしてサインインする場合、IAM ロールを使用して、前もって管理者により ID フェデレーションが設定されています。フェデレーションを使用して にアクセスすると、間接的 AWS にロールを引き受けます。

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

AWS プログラムで にアクセスする場合、 は Software Development Kit (SDK) とコマンドラインインターフェイス (CLI) AWS を提供し、 認証情報を使用してリクエストに暗号で署名します。AWS ツールを使用しない場合は、リクエストを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、セキュリティ情報の提供を追加でリクエストされる場合もあります。例えば、では、多要素認証 (MFA) を使用してアカウントのセキュリティを高めることを AWS 推奨しています。詳細については、「AWS IAM Identity Center ユーザーガイド」の「[多要素認証](#)」および「IAM ユーザーガイド」の「[AWS での多要素認証 \(MFA\) 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 AWS のサービス およびリソースへの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。このアイデンティティは 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を保護し、それらを使用してルートユーザーのみが実行できるタスクを実行してください。ルートユーザーとしてサインインする必要があります

タスクの完全なリストについては、「IAM ユーザーガイド」の「[ルートユーザー認証情報が必要なタスク](#)」を参照してください。

フェデレーテッドアイデンティティ

ベストプラクティスとして、管理者アクセスを必要とするユーザーを含む人間のユーザーに、一時的な AWS のサービス 認証情報を使用して にアクセスする ID プロバイダーとのフェデレーションの使用を要求します。

フェデレーテッド ID とは、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、Identity Center ディレクトリのユーザー、または ID ソースから提供された認証情報 AWS のサービス を使用して にアクセスするすべてのユーザーです。フェデレーテッド ID が にアクセスすると AWS アカウント、ロールを引き受け、ロールは一時的な認証情報を提供します。

アクセスを一元管理する場合は、AWS IAM Identity Center を使用することをお勧めします。IAM Identity Center でユーザーとグループを作成することも、すべての およびアプリケーションで使用する独自の ID ソースのユーザー AWS アカウント とグループのセットに接続して同期することもできます。IAM アイデンティティセンターの詳細については、「AWS IAM Identity Center ユーザーガイド」の「[What is IAM アイデンティティセンター?](#)」(IAM アイデンティティセンターとは)を参照してください。

IAM ユーザーとグループ

[IAM ユーザー](#)は、単一のユーザーまたはアプリケーションに対して特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を保有する IAM ユーザーを作成する代わりに、一時的な認証情報を使用することをお勧めします。ただし、IAM ユーザーでの長期的な認証情報が必要な特定のユースケースがある場合は、アクセスキーをローテーションすることをお勧めします。詳細については、「IAM ユーザーガイド」の「[長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションする](#)」を参照してください。

[IAM グループ](#)は、IAM ユーザーの集団を指定するアイデンティティです。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、IAMAdmins という名前のグループを設定して、そのグループに IAM リソースを管理するアクセス許可を与えることができます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユー

ザーには永続的な長期の認証情報がありますが、ロールでは一時的な認証情報が提供されます。詳細については、「IAM ユーザーガイド」の「[\(ロールではなく\) IAM ユーザーの作成が適している場合](#)」を参照してください。

IAM ロール

[IAM ロール](#)は、特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。これは IAM ユーザーに似ていますが、特定のユーザーには関連付けられていません。ロールを切り替える AWS Management Console ことで、IAM [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 URL を使用します。ロールを使用する方法の詳細については、「IAM ユーザーガイド」の「[IAM ロールの使用](#)」を参照してください。

一時的な認証情報を持った IAM 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アイデンティティにアクセス許可を割り当てるには、ロールを作成してそのロールのアクセス許可を定義します。フェデレーティッドアイデンティティが認証されると、そのアイデンティティはロールに関連付けられ、ロールで定義されている権限が付与されます。フェデレーションの詳細については、「IAM ユーザーガイド」の「[サードパーティーアイデンティティプロバイダー向けロールの作成](#)」を参照してください。IAM Identity Center を使用する場合、アクセス許可セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM アイデンティティセンターは、アクセス許可セットを IAM のロールに関連付けます。権限セットの詳細については、「AWS IAM Identity Center ユーザーガイド」の「[権限セット](#)」を参照してください。
- 一時的な IAM ユーザーアクセス許可 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なるアクセス許可を一時的に IAM ロールで引き受けることができます。
- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の では AWS のサービス、(ロールをプロキシとして使用する代わりに) リソースにポリシーを直接アタッチできます。クロスアカウントアクセスにおけるロールとリソースベースのポリシーの違いについては、「IAM ユーザーガイド」の「[IAM ロールとリソースベースのポリシーの相違点](#)」を参照してください。
- クロスサービスアクセス — 一部の では、他の の機能 AWS のサービス を使用します AWS のサービス。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスでは、呼び出し元プリンシパルの権限、サービスロール、またはサービスリンクロールを使用してこれを行う場合があります。

- 転送アクセスセッション (FAS) – IAM ユーザーまたはロールを使用してアクションを実行する場合 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、呼び出すプリンシパルのアクセス許可 AWS のサービス を使用し AWS のサービス、ダウンストリームサービスにリクエストを行うリクエストと組み合わせて使用します。FAS リクエストは、他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストをサービスが受信した場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。FAS リクエストを行う際のポリシーの詳細については、「[転送アクセスセッション](#)」を参照してください。
- サービスロール - サービスがユーザーに代わってアクションを実行するために引き受ける [IAM ロール](#)です。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、「IAM ユーザーガイド」の「[AWS のサービスにアクセス許可を委任するロールの作成](#)」を参照してください。
- サービスにリンクされたロール – サービスにリンクされたロールは、にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示できますが、編集することはできません。
- Amazon EC2 で実行されているアプリケーション – IAM ロールを使用して、EC2 インスタンスで実行され、AWS CLI または AWS API リクエストを作成しているアプリケーションの一時的な認証情報を管理できます。これは、EC2 インスタンス内でのアクセスキーの保存に推奨されます。EC2 インスタンスに AWS ロールを割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスにアタッチされたインスタンスプロファイルを作成します。インスタンスプロファイルにはロールが含まれ、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、「IAM ユーザーガイド」の「[Amazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用してアクセス許可を付与する](#)」を参照してください。

IAM ロールと IAM ユーザーのどちらを使用するかについては、「IAM ユーザーガイド」の「[\(ユーザーではなく\) IAM ロールの作成が適している場合](#)」を参照してください。

ポリシーを使用したアクセスの管理

でアクセスを制御するには AWS、ポリシーを作成し、AWS ID またはリソースにアタッチします。ポリシーは のオブジェクト AWS であり、ID またはリソースに関連付けられると、これらのアクセス許可を定義します。は、プリンシパル (ユーザー、ルートユーザー、またはロールセッション) が

リクエストを行うと、これらのポリシー AWS を評価します。ポリシーでのアクセス許可により、リクエストが許可されるか拒否されるかが決まります。ほとんどのポリシーは JSON ドキュメント AWS としてに保存されます。JSON ポリシードキュメントの構造と内容の詳細については、「IAM ユーザーガイド」の「[JSON ポリシーの概要](#)」を参照してください。

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

デフォルトでは、ユーザーやロールに権限はありません。IAM 管理者は、リソースで必要なアクションを実行するためのアクセス許可をユーザーに付与するため、IAM ポリシーを作成できます。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き継ぐことができます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの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。このポリシーを持つユーザーは、AWS Management Console、AWS CLI または AWS API からロール情報を取得できます。

アイデンティティベースのポリシー

アイデンティティベースポリシーは、IAM ユーザー、ユーザーのグループ、ロールなど、アイデンティティにアタッチできる JSON 権限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、および条件を制御します。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、「IAM ユーザーガイド」の「[IAM ポリシーの作成](#)」を参照してください。

アイデンティティベースポリシーは、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、複数のユーザー、グループ、ロールにアタッチできるスタンドアロンポリシーです AWS アカウント。管理ポリシーには、AWS 管理ポリシーとカスタマー管理ポリシーが含まれます。マネージドポリシーまたはインラインポリシーのいずれかを選択する方法については、「IAM ユーザーガイド」の「[マネージドポリシーとインラインポリシーの比較](#)」を参照してください。

リソースベースのポリシー

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには、例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあります。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制御できます。ポリシーが添付されているリソースの場合、指定

されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、[プリンシパルを指定する](#)必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーテッドユーザー、またはを含めることができます AWS のサービス。

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

アクセスコントロールリスト (ACL)

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

Amazon S3、および Amazon VPC は AWS WAF、ACLs。ACL の詳細については、「Amazon Simple Storage Service デベロッパーガイド」の「[アクセスコントロールリスト \(ACL\) の概要](#)」を参照してください。

その他のポリシータイプ

AWS は、追加の一般的でない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る最大の許可を設定できます。

- 権限の境界 - 権限の境界は、アイデンティティベースのポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる許可の上限を設定する高度な機能です。エンティティに権限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権限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、権限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。権限の境界の詳細については、「IAM ユーザーガイド」の「[IAM エンティティの権限の境界](#)」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 (SCP) を一部またはすべてのアカウントに適用できます。SCP は、各を含むメンバーアカウントのエンティティのアクセス許可を制限します AWS アカウントのルートユーザー。Organizations と SCP の詳細については、「AWS Organizations ユーザーガイド」の「[SCP の仕組み](#)」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

セッションの権限の範囲は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合があります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。詳細については、「IAM ユーザーガイド」の「[セッションポリシー](#)」を参照してください。

複数のポリシータイプ

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

Amazon Security Lake と IAM との連携方法

IAM を使用して Security Lake へのアクセスを管理する前に、Security Lake で使用できる IAM 機能を確認してください。

Amazon Security Lake で使用できる IAM の機能

IAM 機能	Security Lake サポート
アイデンティティベースのポリシー	あり
リソースベースのポリシー	はい
ポリシーアクション	あり
ポリシーリソース	はい
ポリシー条件キー	はい
ACL	なし
ABAC (ポリシー内のタグ)	はい
一時的な認証情報	あり
プリンシパル権限	あり

IAM 機能	Security Lake サポート
サービスロール	いいえ
サービスリンクロール	はい

Security Lake およびその他の AWS のサービスがほとんどの IAM 機能と連携する方法の概要を把握するには、「IAM ユーザーガイド」の[AWS 「IAM と連携する のサービス」](#)を参照してください。

Security Lake のアイデンティティベースのポリシー

アイデンティティベースポリシーをサポートする **あり**

アイデンティティベースポリシーは、IAM ユーザー、ユーザーグループ、ロールなど、アイデンティティに添付できる 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、および条件を制御します。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、「IAM ユーザーガイド」の「[IAM ポリシーの作成](#)」を参照してください。

IAM アイデンティティベースのポリシーでは、許可または拒否するアクションとリソース、およびアクションを許可または拒否する条件を指定できます。プリンシパルは、それが添付されているユーザーまたはロールに適用されるため、アイデンティティベースのポリシーでは指定できません。JSON ポリシーで使用できるすべての要素について学ぶには、「IAM ユーザーガイド」の「[IAM JSON ポリシーの要素のリファレンス](#)」を参照してください。

Security Lakeは、アイデンティティベースのポリシーをサポートします。詳細については、「[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、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには、例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあります。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制御できます。ポリシーが添付されているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、[プリンシパルを指定する](#)必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーテッドユーザー、または `iam:*` を含めることができます AWS のサービス。

クロスアカウントアクセスを有効にするには、アカウント全体、または別のアカウントの IAM エンティティをリソースベースのポリシーのプリンシパルとして指定します。リソースベースのポリシーにクロスアカウントのプリンシパルを追加しても、信頼関係は半分しか確立されない点に注意してください。プリンシパルとリソースが異なる がある場合 AWS アカウント、信頼されたアカウントの IAM 管理者は、リソースへのアクセス許可をプリンシパルエンティティ (ユーザーまたはロール) に付与する必要があります。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添付することで許可を付与します。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追加する必要はありません。詳細については、「IAM ユーザーガイド」の「[IAM ロールとリソースベースのポリシーとの相違点](#)」を参照してください。

Security Lake サービスは、データを保存する Amazon S3 バケットのリソースベースのポリシーを作成します。S3 バケットには、これらのリソースベースのポリシーをアタッチしないでください。Security Lake はユーザーに代わってこれらのポリシーを自動的に作成します。

リソースの例としては、Amazon リソースネーム (ARN) の `arn:aws:s3:::aws-security-data-lake-{region}-{bucket-identifier}` S3 バケットがあります。この例では、`region` は Security Lake AWS リージョン を有効にした特定の 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の文字列です。Security Lake は S3 バケットを作成して、そのリージョンのデータを保存します。リソースポリシーは、バケットに対してアクションを実行できるプリンシパルを定義します。Security Lake がバケットにアタッチするリソースベースのポリシー (バケットポリシー) の例を次に示します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Principal": {
        "AWS": "*"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::aws-security-data-lake-{region}-{bucket-identifier}/*",
```

```
        "arn:aws:s3:::aws-security-data-lake-{region}-{bucket-identifier}"
    ],
    "Condition": {
        "Bool": {
            "aws:SecureTransport": "false"
        }
    }
},
{
    "Sid": "PutSecurityLakeObject",
    "Effect": "Allow",
    "Principal": {
        "Service": "securitylake.amazonaws.com"
    },
    "Action": "s3:PutObject",
    "Resource": [
        "arn:aws:s3:::aws-security-data-lake-{region}-{bucket-identifier}/*",
        "arn:aws:s3:::aws-security-data-lake-{region}-{bucket-identifier}"
    ],
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "{DA-AccountID}",
            "s3:x-amz-acl": "bucket-owner-full-control"
        },
        "ArnLike": {
            "aws:SourceArn": "arn:aws:securitylake:us-east-1:{DA-AccountID}:*"
        }
    }
}
]
```

リソースベースのポリシーの詳細については、IAM User Guide 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](#)」を参照してください。

セキュリティレイクのポリシーアクション

ポリシーアクションに対するサポート

はい

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

JSON ポリシーのAction要素には、ポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。ポリシーアクションの名前は通常、関連する AWS API オペレーションと同じです。一致する API オペレーションのない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、ポリシーに複数アクションが必要なオペレーションもあります。これらの追加アクションは、依存アクションと呼ばれます。

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

Security Lake アクションのリストについては、「サービス認可リファレンス」の「[Amazon Security Lake によって定義されたアクション](#)」を参照してください。

Security Lake のポリシーアクションは、アクションの前に以下のプレフィックスを使用します。

```
securitylake
```

例えば、ユーザーに特定のサブスクリ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

ポリシーリソースに対するサポート

はい

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

Resource JSON ポリシーの要素は、オブジェクトあるい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は、[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、リソースレベルのアクセス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、リソースレベルのアクセス許可をサポートしないアクションの場合は、ステートメントがすべてのリソースに適用されることを示すために、ワイルドカード (*) を使用します。

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

Security Lake は、サブスクライバー、および特定の AWS アカウント のデータレイク設定のリソースタイプを定義します AWS リージョン。ARN を使用して、ポリシーでこれらのタイプのリソースを指定できます。

Security Lake リソースタイプのリストとそれぞれの ARN 構文については、「サービス認可リファレンス」の「[Amazon Security Lake によって定義されたリソースタイプ](#)」を参照してください。リソースタイプごとに指定できるアクションについては、「サービス認証リファレンス」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	はい
----------------------	----

管理者は AWS JSON ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどんなリソースにどんな条件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になる条件を指定できます。Condition 要素はオプションです。equal や less than などの[条件演算子](#)を使用して条件式を作成することによって、ポリシーの条件とリクエスト内の値を一致させることができます。

1つのステートメントに複数の Condition 要素が指定されている場合、または 1つの Condition 要素に複数のキーが指定されている場合、AWS では AND 論理演算子を使用してそれら进行评估します。1つの条件キーに複数の値を指定すると、は論理OR演算を使用して条件 AWS を评估します。ステートメントの権限が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば IAM ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる許可を付与できます。詳細については、「IAM ユーザーガイド」の「[IAM ポリシーの要素: 変数およびタグ](#)」を参照してください。

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

Security Lake 条件キーのリストについては、「サービス認可リファレンス」の「[Amazon Security Lake の条件キー](#)」を参照してください。条件キーを使用できるアクションとリソースについては、「サービス認可リファレンス」の「[Amazon Security Lake で定義されるアクション](#)」を参照してください。条件キーを使用するポリシーの例については、[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 (ACL)

ACL のサポート

なし

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするための権限を持つかを制御します。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

セキュリティレイクは ACL をサポートしていないため、セキュリティレイクリソースに ACL をアタッチすることはできません。

Security Lake での属性ベースのアクセスコントロール (ABAC)

ABAC のサポート (ポリシー内のタグ)

はい

属性ベースのアクセス制御 (ABAC) は、属性に基づいてアクセス許可を定義するアクセス許可戦略です。では AWS、これらの属性はタグと呼ばれます。タグは、IAM エンティティ (ユーザーまたはロール) および多くの AWS リソースにアタッチできます。エンティティとリソースのタグ付けは、ABAC の最初の手順です。次に、プリンシパルのタグがアクセスを試行するリソースのタグと一致したときにオペレーションを許可するよう、ABAC ポリシーを設計します。

ABAC は、急成長する環境やポリシー管理が煩雑になる状況で役立ちます。

タグに基づいてアクセスを制御するには、`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 Yes です。サービスが一部のリソースタイプに対してのみ 3 つの条件キーすべてをサポートする場合、値は Partial です。

ABAC の詳細については、「IAM ユーザーガイド」の「[ABAC とは?](#)」を参照してください。ABAC をセットアップするステップを説明するチュートリアルについては、「IAM ユーザーガイド」の「[属性ベースのアクセス制御 \(ABAC\) を使用する](#)」を参照してください。

Security Lake リソース、サブスクライバー、および個々の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

はい

一部の AWS のサービスは、一時的な認証情報を使用してサインインすると機能しません。一時的な認証情報 AWS のサービスを使用するなどの詳細については、IAM ユーザーガイドの「IAM [AWS のサービスと連携する](#)」を参照してください。

ユーザー名とパスワード以外の AWS Management Console 方法でサインインする場合は、一時的な認証情報を使用しています。例えば、会社の Single Sign-On (SSO) リンク AWS を使用してにアクセスすると、そのプロセスは自動的に一時的な認証情報を作成します。また、ユーザーとして

コンソールにサインインしてからロールを切り替える場合も、一時的な認証情報が自動的に作成されます。ロールの切り替えに関する詳細については、「IAM ユーザーガイド」の「[ロール \(コンソール\) への切り替え](#)」を参照してください。

一時的な認証情報は、AWS CLI または AWS API を使用して手動で作成できます。その後、これらの一時的な認証情報を使用して、長期的なアクセスキーを使用する代わりに、動的に一時的な認証情報を生成する AWS. AWS recommends にアクセスできます。詳細については、「[IAM の一時的なセキュリティ認証情報](#)」を参照してください。

Security Lake は、一時的な認証情報の使用をサポートしています。

Security Lake の転送アクセスセッション

フォワードアクセスセッション (FAS) をサポート	はい
----------------------------	----

IAM ユーザーまたはロールを使用してアクションを実行する場合 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使用し AWS のサービス、ダウンストリームサービスにリクエストを行う AWS のサービス リクエストと組み合わせて使用します。FAS リクエストは、他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストをサービスが受信した場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。FAS リクエストを行う際のポリシーの詳細については、「[転送アクセスセッション](#)」を参照してください。

Security Lake のアクションの中には、他の AWS のサービスにおけるアクションに依存する追加のアクションに対する権限が必要なものもあります。これらのアクションのリストについては、「サービス認可リファレンス」の「[Amazon Security Lake によって定義されたアクション](#)」を参照してください。

セキュリティレイクのサービスロール

サービスロールのサポート	いいえ
--------------	-----

サービスロールとは、サービスがユーザーに代わってアクションを実行するために引き受ける [IAM ロール](#) です。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細につい

では、「IAM ユーザーガイド」の「[AWS のサービスにアクセス許可を委任するロールの作成](#)」を参照してください。

Security Lake はサービスロールを引き受けたり使用したりしません。ただし、Security Lake を使用するときには、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 はサービスにリンクされた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 のサービスにリンクされたロール

サービスリンクロールのサポート	はい
-----------------	----

サービスにリンクされたロールは、にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示できますが、編集することはできません。

Security Lake は、AWSServiceRoleForAmazonSecurityLake という名前の IAM サービスにリンクされたロールを使用します。Security Lake サービスにリンクされたロールは、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 に直接リンクされている IAM ロールです。これは Security Lake によって事前定義されており、AWS のサービス Security Lake がユーザーに代わって他の を呼び出すために必要なすべてのアクセス許可が含まれています。Security Lake は、Security Lake AWS リージョン が利用可能なすべてのので、このサービスにリンクされたロールを使用します。

Security Lake サービスにリンクされたロールの作成または管理の詳細については、「[Amazon Security Lake のサービスにリンクされたロール](#)」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AWS APIを使用してタスクを実行することはできません。IAM 管理者は、リソースに必要なアクションを実行するためのアクセス許可をユーザーに付与するため、IAM ポリシーを作成できます。その後、管理者がロールにIAM ポリシーを追加すると、ユーザーはロールを引き受けることができます。

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

各リソースタイプのARNの形式など、Security Lakeによって定義されたアクションとリソースタイプの詳細については、「サービス認可リファレンス」の「[Amazon Security Lake のアクション、リソース、および条件キー](#)」を参照してください。

トピック

- [ポリシー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、または削除できるかどうかを決定します。これらのアクションを実行すると、AWS アカウントに料金が発生する可能性があります。アイデンティティベースの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。これらは使用できます AWS アカウント。ユースケースに固有のAWS カスタマー管理ポリシーを定義して、アクセス許可をさらに減らすことをお勧めします。詳細については、「IAM ユーザーガイド」の「[AWS マネージドポリシー](#)」または「[AWS ジョブ機能の管理ポリシー](#)」を参照してください。

- 最小特権を適用する – IAM ポリシーでアクセス許可を設定する場合、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、最小特権権限とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、「IAM ユーザーガイド」の「[IAM でのポリシーと権限](#)」を参照してください。
- IAM ポリシーで条件を使用してアクセスをさらに制限する - ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。例えば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定できます。条件を使用して、などの特定の を通じてサービスアクションを使用する場合 AWS のサービス、サービスアクションへのアクセスを許可することもできます AWS CloudFormation。詳細については、「IAM ユーザーガイド」の「[IAM JSON ポリシー要素：条件](#)」を参照してください。
- IAM Access Analyzer を使用して IAM ポリシーを検証し、安全で機能的なアクセス許可を確保する - IAM Access Analyzer は、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM アクセスアナライザーは 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーの作成をサポートします。詳細については、「IAM ユーザーガイド」の「[IAM Access Analyzer ポリシーの検証](#)」を参照してください。
- 多要素認証 (MFA) を要求する – IAM ユーザーまたはルートユーザーを必要とするシナリオがある場合は AWS アカウント、セキュリティを強化するために MFA を有効にします。API オペレーションが呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、「IAM ユーザーガイド」の「[MFA 保護 API アクセスの設定](#)」を参照してください。

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

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 リソースの詳細をリストおよび表示できます AWS アカウント。最小限必要な許可よりも制限が厳しいアイデンティティベースのポリシーを作成すると、そのポリシーを持つエンティティ (ユーザーまたはロール) に対してコンソールが意図したとおりに機能しません。

AWS CLI または AWS API のみを呼び出すユーザーには、最小限のコンソールアクセス許可を付与する必要はありません。代わりに、実行しようとしている API オペレーションに一致するアクションのみへのアクセスが許可されます。

ユーザーとロールが Security Lake コンソールを使用できるようにするには、ユーザーにコンソールアクセスを提供する IAM ポリシーを作成します。詳細については、IAM ユーザーガイドの「[IAM アイデンティティ](#)」を参照してください。

ユーザーまたはロールに Security Lake コンソールの使用を許可するポリシーを作成する場合は、そのユーザーまたはロールがコンソールでアクセスする必要のあるリソースに対する適切なアクションがポリシーに含まれていることを確認してください。そうしないと、コンソールでそれらのリソースに移動したり、リソースの詳細を表示したりすることができなくなります。

たとえば、コンソールを使用してカスタムソースを追加するには、ユーザーに以下のアクションの実行を許可する必要があります。

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

例：ユーザーにそれぞれのアクセス権限の表示を許可する

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

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



```

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

```

例:組織管理アカウントに委任された管理者の指定と削除を許可

この例では、AWS Organizations 管理アカウントのユーザーが組織の委任された Security Lake 管理者を指定して削除することを許可するポリシーを作成する方法を示します。

```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "securitylake:RegisterDataLakeDelegatedAdministrator",
                "securitylake:DeregisterDataLakeDelegatedAdministrator"
            ],

```

```

    "Resource": "arn:aws:securitylake:*:*:*"
  }
]
}

```

例: タグに基づいてユーザーが購読者をレビューするのを許可する

ID ベースのポリシーでは、条件を使用して、タグに基づいて Security Lake リソースへのアクセスを制御できます。この例では、Security Lake コンソールまたは Security Lake API を使用して、ユーザーがサブスクライバーを確認することを許可するポリシーを作成する方法を示します。ただし、アクセス許可は、サブスクライバの AA 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せん。

```

{
  "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、購読者には Owner=richard-roe または owner=richard-roe のタグが付けられる必要があります。それ以外の場合、ユーザーはアクセスを拒否されます。条件キー名では大文字と小文字は区別されないため、条件タグキー Owner は Owner と owner に一致します。条件キーの詳細に

については、「IAM ユーザーガイド」の「[IAM JSON ポリシーの要素: Condition](#)」を参照してください。Security Lake リソースのタグ付けの詳細については、[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アクセス許可を付与するように設計されているため、ユーザー、グループ、ロールにアクセス許可の割り当てを開始できます。

AWS 管理ポリシーは、すべての AWS お客様が使用できるため、特定のユースケースに対して最小特権のアクセス許可を付与するわけではない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](#)を定義することで、アクセス許可を絞り込むことをお勧めします。

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

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

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作成できます。この管理ポリシーには、Lambda 関数が S3 パーティションとデータファイルを AWS Glue テーブルにインデックスするためのすべてのアクセス許可が含まれています。

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 にログ記録できるようにします。

- glue – プリンシパルが AWS Glue Data Catalog テーブルに対して特定の書き込みアクションを実行できるようにします。これにより、AWS Glue クローラーはデータ内のパーティションを識別することもできます。
- sqs – プリンシパルが Amazon SQS キューに対して特定の読み取りおよび書き込みアクションを実行し、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}"
        }
    }
}
]
```

```
}
```

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、これらのロールを作成するとき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": [
    {
      "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": "*"
    },
    {
      "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": "*"
},
{
  "Effect": "Deny",
  "Action": [
    "s3:GetObject",
    "s3:GetObjectVersion",
    "s3:ListBucket",
    "s3:ListBucketVersions",
    "s3:PutObject",
    "s3:GetBucketLocation"
  ],
  "NotResource": [
    "arn:aws:s3:::aws-security-data-lake*"
  ]
},
{
  "Effect": "Deny",
  "Action": [
    "sqs:ReceiveMessage",
    "sqs:ChangeMessageVisibility",
    "sqs>DeleteMessage",
    "sqs:GetQueueUrl",
    "sqs:SendMessage",
    "sqs:GetQueueAttributes",
    "sqs:ListQueues"
  ],
  "NotResource": "arn:aws:sqs:*:*:AmazonSecurityLake*"
},
{
  "Effect": "Deny",
  "Action": [
```

```
    "kms:Decrypt",
    "kms:GenerateDataKey"
  ],
  "Resource": "*",
  "Condition": {
    "StringNotLike": {
      "kms:ViaService": [
        "s3.*.amazonaws.com",
        "sqs.*.amazonaws.com"
      ]
    }
  }
},
{
  "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*"
      ]
    }
  }
},
{
  "Effect": "Deny",
  "Action": [
    "kms:Decrypt",
    "kms:GenerateDataKey"
  ],
  "Resource": "*",
  "Condition": {
    "Null": {
      "kms:EncryptionContext:aws:sqs:arn": "false"
    },
    "StringNotLikeIfExists": {
      "kms:EncryptionContext:aws:sqs:arn": [
```



```
        "arn:aws:sqs:*:*:AmazonSecurityLake*"
    ]
}
}
}
]
}
```

AWS マネージドポリシー : AmazonSecurityLakeAdministrator

アカウントのために 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、プリンシパルはセキュリティレイクにオンボーディングし、セキュリティレイクでソースとサブスクライバーを設定できます。

このポリシーには、Security Lake 管理者が Security Lake を通じて他の AWS サービスに対して実行できるアクションが含まれています。

この AmazonSecurityLakeAdministrator ポリシーは、Amazon S3 クロスリージョンレプリケーションの管理、での新しいデータパーティションの登録 AWS Glue、カスタムソースに追加されたデータに対する Glue クローラの実行、または新しいデータの HTTPS エンドポイントサブスクライバーへの通知のために Security Lake が必要とするユーティリティロールの作成をサポートしていません。これらのロールは、[Amazon Security Lake の開始方法](#)で説明されているように事前に作成できます。

Security Lake には、AmazonSecurityLakeAdministrator 管理ポリシーに加えて、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作成できるようにします。これは Amazon EventBridge、これらのサービスを有効にするときの必須ステップです。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` — 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 の管理をプリンシパルに許可します。
- `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` — プリンシパルが、Security Lake ソースの新しいデータをサブスクライバーに通知するために使用する Amazon SQS キューを作成および管理できるようにします。
- `kms`— プリンシパルが Security Lake に顧客管理キーを使用してデータを書き込むためのアクセス権を付与できるようにします。
- `secretsmanager`— プリンシパルは、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ます。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowActionsWithAnyResource",
      "Effect": "Allow",
      "Action": [
        "securitylake:*",
        "organizations:DescribeOrganization",
        "organizations:ListDelegatedServicesForAccount",
        "organizations:ListAccounts",
        "iam:ListRoles",
        "ram:GetResourceShareAssociations"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowActionsWithAnyResourceViaSecurityLake",
      "Effect": "Allow",
      "Action": [
        "glue:CreateCrawler",
        "glue:StopCrawlerSchedule",
        "lambda:CreateEventSourceMapping",

```

```

    "lakeformation:GrantPermissions",
    "lakeformation:ListPermissions",
    "lakeformation:RegisterResource",
    "lakeformation:RevokePermissions",
    "lakeformation:GetDatalakeSettings",
    "events:ListConnections",
    "events:ListApiDestinations",
    "iam:GetRole",
    "iam:ListAttachedRolePolicies",
    "kms:DescribeKey"
  ],
  "Resource": "*",
  "Condition": {
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowManagingSecurityLakeS3Buckets",
  "Effect": "Allow",
  "Action": [
    "s3:CreateBucket",
    "s3:PutBucketPolicy",
    "s3:PutBucketPublicAccessBlock",
    "s3:PutBucketNotification",
    "s3:PutBucketTagging",
    "s3:PutEncryptionConfiguration",
    "s3:PutBucketVersioning",
    "s3:PutReplicationConfiguration",
    "s3:PutLifecycleConfiguration",
    "s3:ListBucket",
    "s3:PutObject",
    "s3:GetBucketNotification"
  ],
  "Resource": "arn:aws:s3::aws-security-data-lake*",
  "Condition": {
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowLambdaCreateFunction",

```

```
"Effect": "Allow",
"Action": [
  "lambda:CreateFunction"
],
"Resource": [
  "arn:aws:lambda:*:*:function:SecurityLake_Glue_Partition_Updater_Lambda*",
  "arn:aws:lambda:*:*:function:AmazonSecurityLake*"
],
"Condition": {
  "ForAnyValue:StringEquals": {
    "aws:CalledVia": "securitylake.amazonaws.com"
  }
}
},
{
  "Sid": "AllowLambdaAddPermission",
  "Effect": "Allow",
  "Action": [
    "lambda:AddPermission"
  ],
  "Resource": [
    "arn:aws:lambda:*:*:function:SecurityLake_Glue_Partition_Updater_Lambda*",
    "arn:aws:lambda:*:*:function:AmazonSecurityLake*"
  ],
  "Condition": {
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    },
    "StringEquals": {
      "lambda:Principal": "securitylake.amazonaws.com"
    }
  }
}
},
{
  "Sid": "AllowGlueActions",
  "Effect": "Allow",
  "Action": [
    "glue:CreateDatabase",
    "glue:GetDatabase",
    "glue:CreateTable",
    "glue:GetTable"
  ],
  "Resource": [
    "arn:aws:glue:*:*:catalog",
```

```

    "arn:aws:glue:*:*:database/amazon_security_lake_glue_db*",
    "arn:aws:glue:*:*:table/amazon_security_lake_glue_db*/*"
  ],
  "Condition": {
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowEventBridgeActions",
  "Effect": "Allow",
  "Action": [
    "events:PutTargets",
    "events:PutRule",
    "events:DescribeRule",
    "events:CreateApiDestination",
    "events:CreateConnection",
    "events:UpdateConnection",
    "events:UpdateApiDestination",
    "events>DeleteConnection",
    "events>DeleteApiDestination",
    "events:ListTargetsByRule",
    "events:RemoveTargets",
    "events>DeleteRule"
  ],
  "Resource": [
    "arn:aws:events:*:*:rule/AmazonSecurityLake*",
    "arn:aws:events:*:*:rule/SecurityLake*",
    "arn:aws:events:*:*:api-destination/AmazonSecurityLake*",
    "arn:aws:events:*:*:connection/AmazonSecurityLake*"
  ],
  "Condition": {
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowSQSActions",
  "Effect": "Allow",
  "Action": [
    "sqs:CreateQueue",
    "sqs:SetQueueAttributes",

```

```
    "sqs:GetQueueURL",
    "sqs:AddPermission",
    "sqs:GetQueueAttributes",
    "sqs>DeleteQueue"
  ],
  "Resource": [
    "arn:aws:sqs:*:*:SecurityLake*",
    "arn:aws:sqs:*:*:AmazonSecurityLake*"
  ],
  "Condition": {
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowKmsCmkGrantForSecurityLake",
  "Effect": "Allow",
  "Action": "kms:CreateGrant",
  "Resource": "arn:aws:kms:*:*:key/*",
  "Condition": {
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    },
    "StringLike": {
      "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::aws-security-data-lake*"
    },
    "ForAllValues:StringEquals": {
      "kms:GrantOperations": [
        "GenerateDataKey",
        "RetireGrant",
        "Decrypt"
      ]
    }
  }
},
{
  "Sid": "AllowEnablingQueryBasedSubscribers",
  "Effect": "Allow",
  "Action": [
    "ram:CreateResourceShare",
    "ram:AssociateResourceShare"
  ],
  "Resource": "*",
```

```

"Condition": {
  "StringLikeIfExists": {
    "ram:ResourceArn": [
      "arn:aws:glue:*:*:catalog",
      "arn:aws:glue:*:*:database/amazon_security_lake_glue_db*",
      "arn:aws:glue:*:*:table/amazon_security_lake_glue_db*/*"
    ]
  },
  "ForAnyValue:StringEquals": {
    "aws:CalledVia": "securitylake.amazonaws.com"
  }
},
{
  "Sid": "AllowConfiguringQueryBasedSubscribers",
  "Effect": "Allow",
  "Action": [
    "ram:UpdateResourceShare",
    "ram:GetResourceShares",
    "ram:DisassociateResourceShare",
    "ram>DeleteResourceShare"
  ],
  "Resource": "*",
  "Condition": {
    "StringLike": {
      "ram:ResourceShareName": "LakeFormation*"
    },
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowConfiguringCredentialsForSubscriberNotification",
  "Effect": "Allow",
  "Action": [
    "secretsmanager:CreateSecret",
    "secretsmanager:GetSecretValue",
    "secretsmanager:PutSecretValue"
  ],
  "Resource": "arn:aws:secretsmanager:*:*:secret:events!connection/
AmazonSecurityLake-*",
  "Condition": {
    "ForAnyValue:StringEquals": {

```



```

        "aws:CalledVia": "securitylake.amazonaws.com"
    }
}
},
{
    "Sid": "AllowPassRoleForUpdatingGluePartitionsSecLakeArn",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": [
        "arn:aws:iam::*:role/service-role/AmazonSecurityLakeMetaStoreManager",
        "arn:aws:iam::*:role/service-role/AmazonSecurityLakeMetaStoreManagerV2"
    ],
    "Condition": {
        "StringEquals": {
            "iam:PassedToService": "lambda.amazonaws.com"
        },
        "StringLike": {
            "iam:AssociatedResourceARN": "arn:aws:securitylake:*:*:data-lake/default"
        }
    }
},
{
    "Sid": "AllowPassRoleForUpdatingGluePartitionsLambdaArn",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": [
        "arn:aws:iam::*:role/service-role/AmazonSecurityLakeMetaStoreManager",
        "arn:aws:iam::*:role/service-role/AmazonSecurityLakeMetaStoreManagerV2"
    ],
    "Condition": {
        "StringEquals": {
            "iam:PassedToService": "lambda.amazonaws.com"
        },
        "StringLike": {
            "iam:AssociatedResourceARN": [
                "arn:aws:lambda:*:*:function:SecurityLake_Glue_Partition_Updater_Lambda*",
                "arn:aws:lambda:*:*:function:AmazonSecurityLake*"
            ]
        }
    },
    "ForAnyValue:StringEquals": {
        "aws:CalledVia": "securitylake.amazonaws.com"
    }
},
},

```

```
{
  "Sid": "AllowPassRoleForCrossRegionReplicationSecLakeArn",
  "Effect": "Allow",
  "Action": "iam:PassRole",
  "Resource": "arn:aws:iam::*:role/service-role/AmazonSecurityLakeS3ReplicationRole",
  "Condition": {
    "StringEquals": {
      "iam:PassedToService": "s3.amazonaws.com"
    },
    "StringLike": {
      "iam:AssociatedResourceARN": "arn:aws:securitylake:*:*:data-lake/default"
    }
  }
},
{
  "Sid": "AllowPassRoleForCrossRegionReplicationS3Arn",
  "Effect": "Allow",
  "Action": "iam:PassRole",
  "Resource": "arn:aws:iam::*:role/service-role/AmazonSecurityLakeS3ReplicationRole",
  "Condition": {
    "StringEquals": {
      "iam:PassedToService": "s3.amazonaws.com"
    },
    "StringLike": {
      "iam:AssociatedResourceARN": "arn:aws:s3:::aws-security-data-lake*"
    },
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowPassRoleForCustomSourceCrawlerSecLakeArn",
  "Effect": "Allow",
  "Action": "iam:PassRole",
  "Resource": "arn:aws:iam::*:role/service-role/AmazonSecurityLakeCustomDataGlueCrawler*",
  "Condition": {
    "StringEquals": {
      "iam:PassedToService": "glue.amazonaws.com"
    },
    "StringLike": {
      "iam:AssociatedResourceARN": "arn:aws:securitylake:*:*:data-lake/default"
    }
  }
}
```

```

    }
  },
  {
    "Sid": "AllowPassRoleForCustomSourceCrawlerGlueArn",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": "arn:aws:iam::*:role/service-role/
AmazonSecurityLakeCustomDataGlueCrawler*",
    "Condition": {
      "StringEquals": {
        "iam:PassedToService": "glue.amazonaws.com"
      },
      "ForAnyValue:StringEquals": {
        "aws:CalledVia": "securitylake.amazonaws.com"
      }
    }
  },
  {
    "Sid": "AllowPassRoleForSubscriberNotificationSecLakeArn",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": "arn:aws:iam::*:role/service-role/
AmazonSecurityLakeSubscriberEventBridge",
    "Condition": {
      "StringEquals": {
        "iam:PassedToService": "events.amazonaws.com"
      },
      "StringLike": {
        "iam:AssociatedResourceARN": "arn:aws:securitylake::*:subscriber/*"
      }
    }
  },
  {
    "Sid": "AllowPassRoleForSubscriberNotificationEventsArn",
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": "arn:aws:iam::*:role/service-role/
AmazonSecurityLakeSubscriberEventBridge",
    "Condition": {
      "StringEquals": {
        "iam:PassedToService": "events.amazonaws.com"
      },
      "StringLike": {
        "iam:AssociatedResourceARN": "arn:aws:events::*:rule/AmazonSecurityLake*"
      }
    }
  }
}

```

```

    },
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowOnboardingToSecurityLakeDependencies",
  "Effect": "Allow",
  "Action": "iam:CreateServiceLinkedRole",
  "Resource": [
    "arn:aws:iam::*:role/aws-service-role/securitylake.amazonaws.com/
AWSServiceRoleForSecurityLake",
    "arn:aws:iam::*:role/aws-service-role/lakeformation.amazonaws.com/
AWSServiceRoleForLakeFormationDataAccess",
    "arn:aws:iam::*:role/aws-service-role/apidestinations.events.amazonaws.com/
AWSServiceRoleForAmazonEventBridgeApiDestinations"
  ],
  "Condition": {
    "StringLike": {
      "iam:AWSServiceName": [
        "securitylake.amazonaws.com",
        "lakeformation.amazonaws.com",
        "apidestinations.events.amazonaws.com"
      ]
    }
  }
},
{
  "Sid": "AllowRolePolicyActionsforSubscribersandSources",
  "Effect": "Allow",
  "Action": [
    "iam:CreateRole",
    "iam:PutRolePolicy",
    "iam>DeleteRolePolicy"
  ],
  "Resource": "arn:aws:iam::*:role/AmazonSecurityLake*",
  "Condition": {
    "StringEquals": {
      "iam:PermissionsBoundary": "arn:aws:iam::aws:policy/
AmazonSecurityLakePermissionsBoundary"
    }
  },
  "ForAnyValue:StringEquals": {
    "aws:CalledVia": "securitylake.amazonaws.com"
  }
}

```

```
    }
  }
},
{
  "Sid": "AllowRegisterS3LocationInLakeFormation",
  "Effect": "Allow",
  "Action": [
    "iam:PutRolePolicy",
    "iam:GetRolePolicy"
  ],
  "Resource": "arn:aws:iam::*:role/aws-service-role/lakeformation.amazonaws.com/
AWSServiceRoleForLakeFormationDataAccess",
  "Condition": {
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowIAMActionsByResource",
  "Effect": "Allow",
  "Action": [
    "iam:ListRolePolicies",
    "iam>DeleteRole"
  ],
  "Resource": "arn:aws:iam::*:role/AmazonSecurityLake*",
  "Condition": {
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "securitylake.amazonaws.com"
    }
  }
},
{
  "Sid": "S3ReadAccessToSecurityLakes",
  "Effect": "Allow",
  "Action": [
    "s3:Get*",
    "s3:List*"
  ],
  "Resource": "arn:aws:s3::aws-security-data-lake-*"
},
{
  "Sid": "S3ReadAccessToSecurityLakeMetastoreObject",
  "Effect": "Allow",
```

```
"Action": [
  "s3:GetObject",
  "s3:GetObjectVersion"
],
"Resource": "arn:aws:s3:::security-lake-meta-store-manager-*"
},
{
  "Sid": "S3ResourcelessReadOnly",
  "Effect": "Allow",
  "Action": [
    "s3:GetAccountPublicAccessBlock",
    "s3:ListAccessPoints",
    "s3:ListAllMyBuckets"
  ],
  "Resource": "*"
}
]
```

AWS マネージドポリシー : SecurityLakeServiceLinkedRole

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にリンクされたロールに添付されます。詳細については、「[Amazon Security Lake のサービスにリンクされたロール](#)」を参照してください。

AWS マネージドポリシー : AWSGlueServiceRole

AWSGlueServiceRole 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のクローリングとパーティションメタデータの識別 AWS Glue を許可します。このメタデータは Data Catalog 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。

Security Lake での AWS マネージドポリシーの更新

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
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した。	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、すべての Security Lake アクションへのフルアクセスをプリンシパルに付与する新しいマネージドポリシーを追加しました。	2023 年 5 月 30 日
Security Lake が変更の追跡を開始	Security Lake が AWS マネージドポリシーの変更の追跡を開始しました。	2022 年 11 月 29 日

Amazon Security Lake のサービスにリンクされたロール

Security Lake は、AWSServiceRoleForSecurityLake 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。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 のサービスを呼び出し、セキュリティ データ レイク サービスを操作するために必要なすべてのアクセス許可が含まれています。

す。Security Lake は、Security Lake が利用可能なすべての AWS リージョン でこのサービスにリンクされたロールを使用します。

サービスにリンクされた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、サービスにリンクされた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 のみは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。

サービスにリンクされたロールの作成、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](#)」(サービスにリンクされたロールのアクセス権限) を参照してください。サービスにリンクされたロールは、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許可の削除が防止され、リソースは保護されます。

サービスにリンクされたロールをサポートする他のサービスについては、「[IAM と連動する AWS のサービス](#)」を参照し、[Service-linked roles] (サービスにリンクされたロール) 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。サービスにリンクされたロールに関するドキュメントをサービスで表示するには、[Yes] (はい) リンクを選択します。

トピック

- [Security Lake のサービスにリンクされたロールの許可](#)
- [Security Lake のサービスにリンクされたロールの作成](#)
- [Security Lake 向けのサービスにリンクされたロールの編集](#)
- [Security Lake 向けのサービスにリンクされたロールの削除](#)
- [Security Lake サービスにリンクされたロールでサポートされる AWS リージョン](#)

Security Lake のサービスにリンクされたロールの許可

Security Lake では、AWSServiceRoleForSecurityLake という名前のサービスにリンクされたロールを使用します。このサービスにリンクされたロールは、ロールを引き受ける上で securitylake.amazonaws.com サービスを信頼します。

ロールのアクセス許可ポリシー (SecurityLakeServiceLinkedRole という名前の AWS 管理ポリシー) により、Security Lake がセキュリティ データ レイクを作成および操作できるようになります。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、関連する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、サービスリンクロールに関する情報を取得する

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"
        }
      }
    }
  ]
}
```

サービスにリンクされたロールの作成、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](#)」(サービスにリンクされたロールのアクセス権限) を参照してください。

Security Lake のサービスにリンクされたロールの作成

Security Lake 用のAWSServiceRoleForSecurityLakeサービスにリンクされたロールを手動で作成する必要はありません。AWS アカウント に対して Security Lake を有効にすると、Security Lake はサービスにリンクされたロールを自動的に作成します。

Security Lake 向けのサービスにリンクされたロールの編集

Security Lake では、AWSServiceRoleForSecurityLake サービスにリンクされたロールの編集は許可されていません。サービスにリンクされた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 を使用したロールの説明の編集はできます。詳細については、「[IAM ユーザーガイド](#)」の「サービスにリンクされたロールの編集」を参照してください。

Security Lake 向けのサービスにリンクされたロールの削除

サービスにリンクされたロールを 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 AWS CLI からサービスにリンクされたロールを削除できます。詳細については、IAM ユーザーガイドの「[サービスにリンクされたロールの削除](#)」を参照してください。

サービスにリンクされた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、数分待つてから操作を再試行してください。

AWSServiceRoleForSecurityLake サービスにリンクされた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にリンクされた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でサポートされるAWS リージョン

Security Lake は、Security Lake が利用可能なすべての場所で 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

AWS [共有責任モデル](#)は、Amazon Security Lake 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、AWS クラウド のすべて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があります。お客様は、このインフラストラクチャでホストされているコンテンツに対する管理を維持する責任があります。また、使用する AWS のサービスのセキュリティ設定と管理タスクもユーザーの責任となります。データプライバシーの詳細については、「[データプライバシーのよくある質問](#)」を参照してください。欧州でのデータ保護の詳細については、「AWS セキュリティブログ」に投稿された「[AWS 責任共有モデルおよび GDPR](#)」のブログ記事を参照してください。

データを保護するため、AWS アカウント の認証情報を保護し、AWS IAM Identity Center または AWS Identity and Access Management (IAM) を使用して個々のユーザーをセットアップすることをお勧めします。この方法により、それぞれのジョブを遂行するために必要な権限のみを各ユーザーに付与できます。また、次の方法でデータを保護することをおすすめします。

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。TLS 1.2 が必須です。TLS 1.3 が推奨されません。
- AWS CloudTrail で API とユーザーアクティビティロギングをセットアップします。
- AWS のサービス内でデフォルトである、すべてのセキュリティ管理に加え、AWS の暗号化ソリューションを使用します。
- 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 API により AWS にアクセスするときに FIPS 140-2 検証済み暗号化モジュールが必要な場合は、FIPS エンドポイントを使用します。利用可能な FIPS エンドポイントの詳細については、「[連邦情報処理規格 \(FIPS\) 140-2](#)」を参照してください。

顧客の E メールアドレスなどの機密情報や重要情報は、タグや Name フィールドなどの自由形式のフィールドに入力しないことを強くお勧めします。これは、コンソール、API、AWS CLI、または AWS SDK を利用して Security Lake または他の AWS のサービスを使用する場合も同様です。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。外部サーバーへの URL を提供する場合は、そのサーバーへのリクエストを検証するための認証情報を URL に含めないように強くお勧めします。

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) からの AWS 所有のキーを使用してこの生データを暗号化します。EE AWS 所有のキーは、AWS サービス (この場合は Security Lake) が複数の AWS アカウントで使用するために所有および管理する AWS KMS キーのコレクションです。

Security Lake は、生のログとイベントデータに対して抽出、変換、ロード (ETL) ジョブを実行します。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の S3 バケット (Security Lake を有効にした AWS リージョンごとに 1 つのバケット) を作成します。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 管理の暗号化キー](#)または[顧客管理のキー](#) (AWS KMS から) を選択できます。どちらの方法も対称暗号化を使用します。

KMS キーを使用してデータを暗号化します。

デフォルトでは、Security Lake によって S3 バケットに配信されるデータは、[Amazon S3 で管理された暗号化キー \(SSE-S3\)](#) を使用した Amazon サーバー側の暗号化によって暗号化されます。直接管理するセキュリティ層を提供するには、代わりに Security Lake データに対して [AWS KMS キー \(SSE-KMS\) を使用したサーバー側暗号化](#)を使用できます。

SSE-KMS は、セキュリティレイクコンソールではサポートされていません。セキュリティレイク API または CLI で SSE-KMS を使用するには、まず [KMS キーを作成する](#)か、既存のキーを使用します。Security Lake データの暗号化と復号化にどのユーザーがキーを使用できるかを決定するポリシーをキーにアタッチします。

顧客管理キーを使用して S3 バケットに書き込まれるデータを暗号化する場合、マルチリージョンキーは選択できません。カスタマー管理キーの場合、Security Lake は CreateGrant リクエストを AWS KMS に送信することで、ユーザーに代わって[許可](#)を作成します。AWS KMS の許可は、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。

- GenerateDataKey リクエストを AWS KMS に送信して、カスタマー管理キーで暗号化されたデータ キーを生成します。
- AWS KMS に RetireGrant リクエストを送信します。データレイクを更新すると、この操作により、ETL 処理用の AWS KMS キーに追加されたグラントの廃止が可能になります。

セキュリティレイクには Decrypt 権限は必要ありません。キーの許可されたユーザーが Security Lake データを読み取ると、S3 が復号化を管理し、許可されたユーザーは暗号化されていない形式でデータを読み取ることができます。ただし、サブスクライバーがソースデータを使用するには Decrypt 権限が必要です。サブスクライバー権限の詳細については、[Security Lake サブスクライバーのデータアクセスの管理](#) を参照してください。

キーポリシーを作成したり、適切な権限を持つ既存のキーポリシーを使用したりすると、KMS キーは付与リクエストを受け付けることができ、Security Lake がキーにアクセスできるようになります。キー ポリシーの作成手順については、AWS Key Management Service 開発者ガイドの[キー ポリシーの作成](#)を参照してください。次のキーポリシーを KMS キーにアタッチします。

```
{
  "Sid": "Allow use of the key",
  "Effect": "Allow",
  "Principal": {"AWS": "arn:aws:iam::111122223333:role/ExampleRole"}
  "Action": [
    "kms:CreateGrant",
    "kms:DescribeKey",
    "kms:GenerateDataKey"
  ],
  "Resource": "*"
}
```

顧客マネージドキーを使用する場合に必要な IAM アクセス許可

Security Lake を使用するために作成する必要がある IAM ロールの概要については、「[はじめに:前提条件](#)」セクションを参照してください。

カスタムソースまたはサブスクライバーを追加すると、Security Lake はアカウントに IAM ロールを作成します。これらのロールは他の IAM ID と共有することを目的としています。これにより、カスタムソースはデータレイクにデータを書き込み、サブスクライバーはデータレイクからのデータを使用できます。AWSAmazonSecurityLakePermissionsBoundary という管理ポリシーがこれらのロールの権限境界を設定します。

Amazon SQS キューの暗号化

データレイクを作成すると、Security Lake は委任されたセキュリティレイク管理者アカウントに 2 つの暗号化されていない Amazon Simple Queue Service (Amazon SQS) キューを作成します。データを保護するには、これらのキューを暗号化する必要があります。Amazon Simple Queue Service が提供するデフォルトのサーバー側の暗号化では十分ではありません。AWS Key Management Service (AWS KMS) でカスタマー管理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「[Amazon S3 イベント通知が、サーバー側の暗号化を使用する Amazon SQS キューに配信されないのはなぜですか?](#)」を参照してください。

Security Lake は AWS Lambda を使用してデータに対する抽出、転送、読み込み (ETL) ジョブをサポートしているため、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を Lambda に付与する必要があります。詳細については、「AWS Lambda デベロッパーガイド」の「[実行ロールのアクセス許可](#)」を参照してください。

転送中の暗号化

Security Lake は、AWS サービス間で転送中のすべてのデータを暗号化します。Security Lake は、トランスポート層セキュリティ (TLS) 1.2 暗号化プロトコルを使用してすべてのネットワーク間データを自動的に暗号化することにより、サービスとの間で送受信されるデータを保護します。Security Lake API に送信されるダイレクト HTTPS リクエストは、[署名バージョン 4 AWS アルゴリズムを使用して署名され](#)、安全な接続を確立します。

サービス改善のためのデータ使用をオプトアウトする

AWS Organizations オプトアウトポリシーを使用して、データが Security Lake の開発および向上、その他の AWS セキュリティサービスに使用されることを拒否することができます。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 は将来、顧客がサードパーティ データ ソースからアップロードしたデータを含むかかるデータを収集する可能性があります。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 AWS Organizations によって一元管理される必要があります。AWS アカウントでまだ組織を作成していない場合は、「AWS Organizations ユーザーガイド」の「[組織の作成と管理](#)」を参照してください。

オプトアウトには次のような効果があります。

- Security Lake は、オプトアウト前に収集して保存したデータ (ある場合のみ) を削除します。
- オプトアウトすると、Security Lake はこのデータを収集または保存しません。

Amazon SECURITY Lake のコンプライアンス検証

AWS のサービスが特定のコンプライアンスプログラムの範囲内にあるかどうかを確認するには、「[コンプライアンスプログラム別の範囲](#)」の「AWS のサービス」と「」の「AWS のサービス」を参照し、関心のあるコンプライアンスプログラムを選択してください。一般的な情報については、「[AWS コンプライアンスプログラム](#)」を参照してください。

AWS Artifact を使用して、サードパーティーの監査レポートをダウンロードできます。詳細については、「[Downloading Reports in AWS Artifact](#)」を参照してください。

AWS のサービスを使用する際のユーザーのコンプライアンス責任は、ユーザーのデータの機密性や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。AWS では、コンプライアンスに役立つ次のリソースを提供しています。

- [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を AWS にデプロイするための手順を示します。
- 「[Amazon Web Services での HIPAA のセキュリティとコンプライアンスのためのアーキテクチャ](#)」 - このホワイトペーパーは、企業が AWS を使用して HIPAA 対象アプリケーションを作成する方法を説明しています。

Note

すべての AWS のサービスが HIPAA 適格であるわけではありません。詳細については、「[HIPAA 対応サービスのリファレンス](#)」を参照してください。

- [AWS コンプライアンスのリソース](#) - このワークブックおよびガイドのコレクションは、顧客の業界と拠点に適用されるものである場合があります。
- [AWS 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見た責任分担モデルを理解してください。このガイドは、AWS のサービス セキュリティを確保するためのベストプラクティスをまとめ、複数のフレームワーク (米国標準技術研究所 (NIST)、ペイメントカード業界セキュリティ標準評議会 (PCI)、国際標準化機構 (ISO) など) にわたるセキュリティ統制へのガイダンスをまとめたものです。
- 「AWS Config デベロッパーガイド」の「[ルールでのリソースの評価](#)」 - AWS Config サービスでは、自社のプラクティス、業界ガイドライン、および規制に対するリソースの設定の準拠状態を評価します。
- [AWS Security Hub](#) - この AWS のサービスは、AWS 内のセキュリティ状態の包括的なビューを提供します。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。
- [AWS Audit Manager](#) - この AWS のサービスは、AWS の使用状況を継続的に監査して、リスクの管理方法や、規制および業界標準へのコンプライアンスの管理方法を簡素化するために役立ちます。

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、ソースやサブスクリ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 の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ゾーンがあります。これらのアベイラビリティゾーンを利用すると、アプリケーションとデータベースを効率的に設計して運用できます。アベイラビリティゾーンは、従来の単一または複数のデータセンターインフラストラクチャよりも可用性、耐障害性、および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ゾーンに分散することで、サービスはどのアベイラビリティゾーンでも障害に耐えることができます。

Security Lake データプレーンの可用性は、どのリージョンの可用性とも関係ありません。ただし、Security Lake コントロールプレーンの可用性は、米国東部 (バージニア北部) リージョンの可用性と密接に関係しています。

AWS リージョン とアベイラビリティゾーンの詳細については、「[AWS グローバルインフラストラクチャ](#)」を参照してください。

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エーションを保持する手段です。バージョニングを使用して、Amazon S3 バケットに格納されたあらゆるオブジェクトのあらゆるバージョンを、格納、取得、復元することができます。バージョニングによって、意図しないユーザーアクションとアプリケーション障害から復旧できます。詳細については、『Amazon S3 ユーザーガイド』の「[S3 バケットでのバージョニングの使用](#)」を参照してください。

ストレージクラス

Amazon S3 では、ワークロードの要件に応じて、幅広いストレージクラスが提供されています。S3 標準 — IA と S3 1 ゾーン — IA ストレージクラスは、月に約 1 回アクセスし、ミリ秒単位のアクセスが必要になるデータ用に設計されています。S3 Glacier インスタント検索ストレージクラスは、四半期に約 1 回アクセスするミリ秒のアクセスでアクセスされる長期間有効なアーカイブデータ用に設計されています。バックアップなど、即時アクセスを必要としないアーカイブデータについては、S3 Glacier Flexible Retrieval または S3 Glacier Deep Archive ストレージクラスを使用できます。詳細については、『Amazon S3 ユーザーガイド』の「[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 AWS がインフラストラクチャ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「[インフラストラクチャ保護](#)」を参照してください。

AWS が公開した API コールを使用して、ネットワーク経由で Security Lake にアクセスします。クライアントは以下をサポートする必要があります。

- Transport Layer Security (TLS) TLS 1.2 および TLS 1.3 をお勧めします。
- DHE (Ephemeral Diffie-Hellman) や ECDHE (Elliptic Curve Ephemeral Diffie-Hellman) などの Perfect Forward Secrecy (PFS) を使用した暗号スイートです。これらのモードは、Java 7 以降など、最近のほとんどのシステムでサポートされています。

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

Security Lake での構成と脆弱性の分析

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 のサービスによって 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 の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、メトリクスのしきい値に到達したときのアラームを設定することもできます。たとえば、ProcessedSize メトリクスにアラームを設定して、特定のソースからのデータ量が特定のしきい値を超えたときに通知を受けることができます。

アラームの設定に関する詳細については、Amazon CloudWatch ユーザーガイドの「[Using Amazon CloudWatch Alarm](#)」を参照してください。

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 によってログに記録され、[API Lake API リファレンス](#)に記載されます。例えば、UpdateDataLake、ListLogSources、CreateSubscriber の各アクションを呼び出すと、CloudTrail ログファイルにエントリが生成されます。

各イベントまたはログエントリには、リクエストの生成者に関する情報が含まれます。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](#)
- [IAMポリシーでタグを使用する](#)
- [Amazon セキュリティレイクリソースへのタグの追加](#)
- [Amazon Security Lake リソースのタグのレビュー](#)
- [Amazon Security Lake リソースのタグを編集する](#)
- [Amazon Security Lake リソースからのタグの削除](#)

タグ付けの基本

リソースには、最大 50 個のタグを含めることができます。タグはそれぞれ、1 つの必須タグキーとオプションの 1 つのタグ値で構成されており、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 のサービス、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 のサービス で作成、使用、または維持する、関連するサポート リソースには適用されません。たとえば、データレイクにタグを割り当てた場合、タグは指定されたリージョンの Security Lake 内のデータレイク設定にのみ適用されます。ログやイベントデータが保存されている Amazon Simple Storage Service (Amazon S3) バケットには適用されません。関連付けられたリソースにもタグを割り当てるには、AWS Resource Groups またはリソース AWS のサービスを保存する を使用できます。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 のサービス、を含む多くの からアクセスできます AWS Billing and Cost Management。それらは機密データに使用することを目的としていません。

Security Lake リソースのタグを追加および管理するには、Security Lake コンソールまたは Security Lake API を使用できます。

IAMポリシーでタグを使用する

リソースのタグ付けを開始した後、タグベースのリソースレベルのアクセス許可を AWS Identity and Access Management (IAM) ポリシーで定義できます。この方法でタグを使用すると、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、keyvalue との引数の両方が必要です。ただし、value 引数の値は空の文字列とすることができます。タグ値をタグキーに関連付けない場合、value 引数の値を指定しないでください。たとえば、以下のコマンドは、関連付けられたタグ値を含まない `Owner` タグキーを追加します。

```
$ aws securitylake tag-resource \  
--resource-arn arn:aws:securitylake:us-  
east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab \  
--tags key=Owner,value=
```

タグ付けオペレーションが正常に実行された場合は Security Lake は空の HTTP 200 レスポンスを返します。それ以外の場合、Security Lake は HTTP 4xx またはオペレーションが失敗した理由を示す 500 レスポンスを返します。

Amazon Security Lake リソースのタグのレビュー

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。

- 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。
- 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 パラメーターを使用して、削除するタグのタグキーを指定します。複数のタグを削除するには、削除する各タグの tagKeys パラメーターと引数をアンパサンド (&) で区切って追加します (例: tagKeys=key1&tagKeys=key2)。リソースから特定のタグ値 (タグキーではない) のみを削除するには、タグを削除する代わりに [タグを編集](#) します。

AWS Command Line Interface (AWS CLI) を使用している場合は、[untag-resource](#) コマンドを実行して、リソースから 1 つ以上のタグを削除します。resource-arn パラメータには、タグを削除するリソースの ARN を指定します。tag-keys パラメータを使用して、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することができません。問題 ページには、過去 14 日間に発生した問題が表示されます。各問題の説明と推奨される修復手順を確認できます。

問題の概要にプログラムでアクセスするには、Security Lake API の [ListDataLakeExceptions](#) オペレーションを使用できます。を使用している場合は AWS CLI、[list-data-lake-exceptions](#) コマンドを実行します。regions パラメータでは、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"
```

Security Lake ユーザーに問題またはエラーを通知するには、Security Lake API の [CreateDataLakeExceptionSubscription](#) オペレーションを使用します。ユーザーは、E メール、Amazon Simple Queue Service (Amazon SQS) キューへの配信、AWS Lambda 関数への配信、またはサポートされている別のプロトコルを通じて通知を受けることができます。

例えば、次の AWS CLI コマンドは、指定されたアカウントに SMS 配信によって Security Lake 例外の通知を送信します。この例は Linux、macOS、または Unix 用にフォーマットされており、読みやすさを向上させるためにバックスラッシュ (\) の行継続文字を使用しています。

```
$ aws securitylake create-data-lake-exception-subscription \  
--notification-endpoint "123456789012" \  
--exception-time-to-live 30 \  
\
```

```
--subscription-protocol "sms"
```

例外サブスクリプションの詳細を表示するには、[GetDataLakeExceptionSubscription](#) オペレーションを使用できます。例外サブスクリプションを更新するには、[UpdateDataLakeExceptionSubscription](#) オペレーションを使用できます。例外サブスクリプションを削除して通知を停止するには、[DeleteDataLakeExceptionSubscription](#) オペレーションを使用できます。

Lake Formation 問題のトラブルシューティング

次の情報は、Security Lake と AWS Lake Formation データベースまたはテーブルの使用時に発生する可能性がある一般的な問題の診断や修復に役立ちます。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 デベロッパガイド」の「[トラブルシューティング](#)」セクションを参照してください。

テーブルが見つからない

サブスクライバーを作成しようとするとき、このエラーが表示されることがあります。

このエラーを解決するには、リージョンにソースを既に追加していることを確認します。Security Lake サービスのプレビューリリース時にソースを追加した場合は、サブスクライバーを作成する前にソースを再度追加する必要があります。ソースの追加についての詳細は、「[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 CreateCustomLogSource API を呼び出すと、このエラーが表示されることがあります。

このエラーを解決するには、Lake Formation の権限を確認してください。API を呼び出す IAM ロールには、Security Lake データベースの Create table 権限が必要です。詳細については、「AWS Lake Formation 開発者ガイド」の「[Lake Formation コンソールと名前付きリソース メソッドを使用したデータベース アクセス許可の付与](#)」を参照してください。

SYNTAX_ERROR: 1:8 行目:カラムのないリレーションからの SELECT * は許可されていません

Lake Formation で初めてソーステーブルをクエリするとき、このエラーが表示されることがあります。

このエラーを解決するには、にサインインしたときに使用している IAM ロールにアクセスSELECT許可を付与します AWS アカウント。SELECT権限を付与する方法については、『AWS Lake Formation 開発者ガイド』の「[Lake Formation コンソールと名前付きリソースメソッドを使用したテーブル権限の付与](#)」を参照してください。

Security Lakeは、呼び出し元のプリンシパル ARN をレイクフォーメーションデータレイク管理者に追加できませんでした。現在のデータレイク管理者には、もはや存在しない無効なプリンシパルが含まれている可能性があります。

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 のサービスとして追加するときに表示されることがあります。

このエラーを解決するには、以下の手順を実行します。

1. Lake Formation コンソール (<https://console.aws.amazon.com/lakeformation/>) を開きます。
2. 管理ユーザーとしてサインインする
3. ナビゲーションペインの [Permissions] (許可) で [Administrative Roles and tasks] (管理ロールとタスク) を選択します。
4. [Data lake administrators] (データレイク管理者) セクションで、[Choose administrators] (管理者を選択) を選択します。
5. 「IAM で見つからない」というラベルの付いたプリンシパルをクリアし、[保存] を選択します。
6. Security Lake操作をもう一度試してください。

Lake Formation CreateSubscriber を使用した Security Lake が、受け入れる新しい RAM リソース共有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 とバージョン 3 のクロスアカウント共有が、複数のクロスアカウント許可の付与を 1 つの AWS RAM リソース共有にマッピングすることで、AWS RAM リソース共有の数を最適化するためです。

リソース共有名がサブスクライバーの作成時に指定した外部 ID であり、リソース共有 ARN がCreateSubscriber応答の ARN と一致することを確認してください。

Amazon Athena でのクエリのトラブルシューティング

次の情報を使用して、Security Lake S3 バケットに保存されているオブジェクトを Athena を使用してクエリする際に発生する可能性のある一般的な問題を診断して修正します。Athena のトラブルシューティングに関するその他のトピックについては、「Amazon Athena ユーザーガイド」の「[Athena でのトラブルシューティング](#)」セクションを参照してください。

クエリを実行しても、データレイク内の新しいオブジェクトは返されません。

Security Lake の S3 バケットにそれらのオブジェクトが含まれていても、Athena クエリはデータレイク内の新しいオブジェクトを返さない場合があります。これは、Security Lake を無効にしてから再度有効にした場合に発生することがあります。その結果、AWS Glue パーティションが新しいオブジェクトを正しく登録できない可能性があります。

このエラーを解決するには、以下の手順を実行します。

1. <https://console.aws.amazon.com/lambda/> で AWS Lambda コンソールを開きます。
2. ナビゲーションバーのリージョンセレクトで、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で、関数 を選択し、SecurityLake_Glue_Partition_Updater_Lambda_#region> 関数を選択します。
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リバラーは、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 \(サブスクリバーステップ\)](#) で説明されている手順を実行していることを確認します。

購読者がまだアクセスできない場合は、次の手順に従ってください。

1. <https://console.aws.amazon.com/glue/> で AWS Glue コンソールを開きます。

2. ナビゲーションペインで、[データカタログ] と [カタログ設定] を選択します。
3. リソースベースのポリシーを使用して AWS Glue テーブルにアクセスする許可をサブスクライバーに付与します。リソースベースのポリシーの作成については、AWS Glue デベロッパーガイドの「[AWS Glue のリソースベースのポリシーの例](#)」を参照してください。

Organizations の問題のトラブルシューティング

次の情報は、Security Lake および AWS Organizations を使用するときが発生する可能性のある一般的な問題の診断と修正に役立ちます。Organizations トラブルシューティングに関するその他のトピックについては、「AWS Organizations ユーザーガイド」の「[トラブルシューティング](#)」セクションを参照してください。

CreateDataLake 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。

このエラーは、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 [CreateDataLake](#) API を使用して Security Lake を設定しようとする时表示されることがあります。

このエラーを解決するには、別の組織の委任管理者アカウントまたはスタンドアロンアカウントを使用してください。

Amazon Security Lake アイデンティティとアクセスのトラブルシューティング

次の情報は、Security Lake と IAM の使用に伴って発生する可能性がある一般的な問題の診断や修復に役立ちます。

Security Lake でアクションを実行することが認可されていない

から、アクションを実行する権限がないと AWS Management Console 通知された場合は、管理者に連絡してサポートを依頼する必要があります。管理者は、認証情報を提供した担当者です。

以下のエラー例は、mateojackson IAM ユーザーがコンソールを使用して架空の *subscriber* に関する詳細情報を表示しようとしているが、架空の SecurityLake:*GetSubscriber* 許可がないという場合に発生します。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
YOURSERVICEPREFIX: GetWidget on resource: my-example-widget
```

この場合、Mateo は、SecurityLake:*GetSubscriber* アクションを使用して *subscriber* 情報へのアクセスが許可されるように、管理者にポリシーの更新を依頼します。

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 のサービスでは、新しいサービスロールまたはサービスにリンクされたロールを作成する代わりに、既存のロールをそのサービスに渡すことができます。そのためには、サービスにロールを渡すアクセス許可が必要です。

以下の例のエラーは、marymajor という IAM ユーザーがコンソールを使用して Security Lake でアクションを実行しようする場合に発生します。ただし、このアクションをサービスが実行するには、サービスロールから付与された権限が必要です。Mary には、ロールをサービスに渡すアクセス許可がありません。

```
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 Mary 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。サインイン資格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、リソースにアクセスするために使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定できます。リソースベースのポリシーまたはアクセスコントロールリスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、以下を参照してください:

- Security Lake でこれらの機能がサポートされるかどうかを確認するには、「[Amazon Security Lake と IAM との連携方法](#)」を参照してください。
- 所有 AWS アカウント する のリソースへのアクセスを提供する方法については、[IAM ユーザーガイドの「所有 AWS アカウント する別の の IAM ユーザーへのアクセスを許可する」](#)を参照してください。
- リソースへのアクセスをサードパーティーの に提供する方法については AWS アカウント、IAM ユーザーガイドの「[第三者 AWS アカウント が所有する へのアクセス権を付与する](#)」を参照してください。
- ID フェデレーションを介してアクセス権を付与する方法については、「IAM ユーザーガイド」の「[外部で認証されたユーザー \(ID フェデレーション\) にアクセス権を付与する](#)」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の詳細については、「IAM ユーザーガイド」の「[IAM ロールとリソースベースのポリシーとの相違点](#)」を参照してください。

Security Lakeの価格の決定方法

Amazon Security Lake の価格は、データ取り込みとデータ変換の 2 つのディメンションに基づいています。Security Lake は他のAWS のサービスと連携してデータを保存および共有するため、これらのアクティビティには別途料金がかかる場合があります。

Security Lake がサポートする AWS リージョンの AWS アカウント で初めてログ収集を有効にすると、そのアカウントは Security Lake の 15 日間の無料トライアルに自動的に登録されます。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。

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、その他のAWS のサービスログ、イベント (Amazon Route 53 リゾルバーのクエリログ、AWS Security Hubの結果、Amazon VPC フローログ) の量から発生します。

データ変換

これらのコストは、Security Lake が [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 のサービス ログとイベントの量から得られます。

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を保存して共有するために、他のAWS のサービスから発生する可能性がある費用は次のとおりです。

- Amazon S3 — Security Lake アカウントの Amazon S3 バケットの維持およびそこへのデータの保存、およびセキュリティとアクセスコントロールのバケットの評価とモニタリングから導出されます。詳細については、「[Amazon S3 の料金](#)」を参照してください。
- Amazon SQS — これらのコストは、メッセージ配信用の Amazon SQS キューを作成することで発生します。詳細については、「[Amazon SQSの料金](#)」を参照してください。
- Amazon EventBridge — これらの費用は、Amazon EventBridge がサブスクリプションエンドポイントにオブジェクト通知を送信することで発生します。詳細については、「[Amazon EventBridge の料金](#)」を参照してください。

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。

詳細については、「[Security Lake の料金](#)」をご覧ください。

Security Lakeの使用状況と推定コストを確認する。

Amazon Security Lake コンソールの「使用状況」ページでは、現在の Security Lake の使用状況だけでなく、将来の使用状況とコストの見積もりを確認できます。現在 15 日間の無料トライアルに参加している場合、トライアル中の使用状況は、無料トライアルの終了後に Security Lake を使用する際のコストを見積もるのに役立ちます。Security Lake の料金の概要については、[を参照してください](#) [Security Lakeの価格の決定方法](#)。詳細情報とコストの例については、「[Amazon Security Lake の料金](#)」を参照してください。

Security Lake では、推定使用コストを米ドルでレポートし、現在のAWS リージョンにのみ適用されます。コストには、組織内のすべてのアカウントの Security Lake の使用がカバーされ、Open Cybersecurity Schema Framework (OCSF) や Apache Parquet 形式への変換が含まれます。ただし、予測コストには、Amazon Simple Storage Service (Amazon S3) や AWS Glue など、Security Lake が連携する他のサービスのコストは含まれていません。

使用状況ページでは、使用状況とコストのデータを表示する期間を選択します。デフォルトの期間は過去 1 暦日です。コスト予測を確認するには、Security Lake の使用日が 1 日以上ある必要があります。

ページの上部には、すべてのアカウントの予測コストが表示されます。これは、選択した期間中の実際の使用量に基づいて、現在の AWS リージョン で今後 30 暦日の Security Lake の予測コストです。実際の使用量と予測コストには、組織内のすべてのアカウントが反映されます。

ページの残りの部分では、使用量とコストのデータが次の 2 つの表に分かれています。

- ソース別の使用量とコスト — データソース別の現在の Security Lake の使用状況と、選択した期間における実際の使用状況に基づく今後 30 日間の推定使用量とコストです。実際の使用量、予測使用量、予測コストには、組織内のすべてのアカウントが反映されます。ソースを選択すると、分割パネルが開き、どのアカウントがそのソースからログとイベントを生成したかが表示されます。各アカウントの分割パネルには、そのソースからの実際の使用状況と、予測される使用量とコストの両方が表示されます。
- アカウント別の使用量とコスト — アカウントごとの現在の Security Lake の使用状況と、選択した期間における実際の使用状況に基づく今後 30 日間の推定使用量とコストが表示されます。アカウントを選択すると、分割パネルが開き、そのアカウントの使用量に貢献したソースが表示されます。貢献しているソースごとに、実際の使用状況と予測される使用量とコストの両方が分割パネルに表示されます。

Security Lake に特定のソースを追加していなくても、サポートされているすべてのAWSデータソースが上の表に示されています。無料トライアルに参加している場合は、すべてのログとイベントのコスト見積もりを取得するために、すべてのAWSソースを追加することをおすすめします。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[AWS サービスからのデータ収集](#)。カスタムソースは使用量やコストの計算には含まれません。

Security Lake コンソールで使用量とコストデータを確認するには、次のステップに従います。

Security Lake の使用状況と予測コストを確認するには (コンソール)

1. Security Lake コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ページの右上の 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 のサービス、に保存したり、他の に公開したデータは引き続き使用できます。Amazon Simple Storage Service (Amazon S3) バケットに保存されたデータは、[Amazon S3 ストレージライフサイクル](#)に従って引き続き使用できます。

Security Lake コンソールの設定ページから Security Lake を無効にすると、Security Lake が現在有効になっているすべての AWS リージョンの AWS ログとイベントの収集が停止します。コンソールの [リージョン] ページを使用して、特定のリージョンのログ収集を停止できます。Security Lake API とは、リクエストで指定したリージョンでのログ収集 AWS CLI 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 管理者だけが Security Lake をそれ自体とメンバーアカウントに対して無効にできます。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。

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 Hub コンソール <https://console.aws.amazon.com/securitylake/> を開きます。
2. ナビゲーションペインで [Settings] の [General] を選択します。
3. [Security Lake を無効にする] を選択します。

4. 確認を求められたら、**Disable**と入力し、Disable (無効化) を選択します。

API

Security Lake をプログラムで無効にするには、Security Lake API の [DeleteDataLake](#) オペレーションを使用します。を使用している場合は AWS CLI、[delete-data-lake](#) コマンドを実行します。リクエストでは、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"
```

「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2月29日

変更	説明	日付
新しいソースバージョン	ロールのアクセス許可を更新 して、新しいデータソースバージョンからデータを取り込みます。	2024年2月29日
新しい AWS ログソース	Security Lake は EKS 監査ログを AWS ログソースとして追加しました。EKS 監査ログは、Amazon Elastic Kubernetes Service 内の EKS クラスターで疑わしいアクティビティの可能性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 で	2024年1月23日

ある 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ためのアクセス許可を付与します。

リージョナルな可用性

Security Lake が、アジアパシフィック (大阪) AWS リージョン、カナダ (中部)、欧州 (パリ)、欧州 (ストックホルム) で利用可能になりました。Security Lake が現在利用可能なリージョンの完全なリストについては、「AWS 全般のリファレンス」の「[Amazon Security Lake エンドポイント](#)」を参照してください。

2023 年 10 月 26 日

新しい特徴

[クエリアクセス権を持つ利用者の特定の設定を編集](#)できるようになりました。自分の [Security Lake リソースにタグを割り当てることもできます](#) AWS アカウント。

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](#) である AmazonSecurityLake Administrator ポリシーを追加しました。このポリシーにより、プリンシパルが Security Lake のすべてのアクションに完全にアクセスすることが許可される、管理者許可が付与されます。

2023 年 5 月 30 日

一般提供	Security Lake は一般的にご利用いただけるようになりました。	2023 年 5 月 30 日
新機能	Security Lake 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) 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、Security Lake がユーザーに代わって AWS Identity and Access Management (IAM) ロールを作成するようになりました。	2023 年 2 月 15 日
初回リリース	これは、Amazon Security Lakeユーザーガイドの初回リリースです。	2022 年 11 月 29 日

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。