



Cisco Secure Cloud Insights

スタートアップガイド



目次

Cisco Secure Cloud Insights について	5
Secure Cloud Insights : 統合	5
Secure Cloud Insights : アプリ	5
アセット	5
ポリシー	6
アラート	6
コンプライアンス	6
グラフビューア	6
インサイト	6
クエリライブラリ	7
Ask Anything 検索バー	8
JupiterOne Query Language	8
マネージド統合の設定	9
検索の設定	10
質問する	10
全文検索	10
JupiterOne Query Language (J1QL)	11
全文検索と J1QL の組み合わせ	12
グラフの操作	13
ズームおよび移動	15
JupiterOne クエリ言語について	16
パート 1 - 単純なルートクエリ	16
パート 2 - インフラストラクチャの分析	18
2a - SSH Key Usage Examples	18
2b - EBS ボリュームの例	19
2c - 暗号化されていないデータ	20
2d - Tagging Resources	21
2e - ネットワークリソースと構成	21
2f - サーバーレス関数	24
パート 3 - ユーザーとアクセスの分析	26
3a - IdP ユーザーとアクセス	26
3b - クラウドユーザーとアクセス	27
3c - Combined Users and Access Across All Environments	27

パート 4 – Cross Account Analysis	28
パート 5 – エンドポイントコンプライアンス	29
アセットインベントリアプリでのフィルタの使用	31
重要なアセットのクイックフィルタ	31
クラス/タイプ別のクイックフィルタ	32
プロパティ別の詳細フィルタ	33
アラートの設定	35
ルールパックからのアラートルールのインポート	35
カスタムアラートルールの作成	35
追加のアラートオプションの設定	36
アラートの管理	37
日次通知 E メールの設定	37
Visual Query Builder の使用	38
権限	38
前提条件	38
VQB を使用したクエリの作成	38
ワイルドカードの使用	40
フィルタリング	41
セキュリティ調査結果の管理	42
結果の表示	42
調査結果に対するアラートを作成する	43
例:	43
J1QL クエリとグラフを使用して調査結果のビューを生成	45
Secure Cloud Insights のポリシーと手順の管理	46
テンプレートからのポリシーと手順の生成	46
変数	46
バージョン管理	46
ポリシーと手順書のダウンロード/エクスポート	47
ポリシービルダー CLI	47
独自の既存ポリシーを使用する	47
Secure Cloud Insights アカウント/組織のユーザーの招待	48
Secure Cloud Insights API の使用	49
エンドポイント	49
認証	49
API キー	49

アカウント ID (Account ID)	49
サポートへの問い合わせ	50


Cisco Secure Cloud Insights について

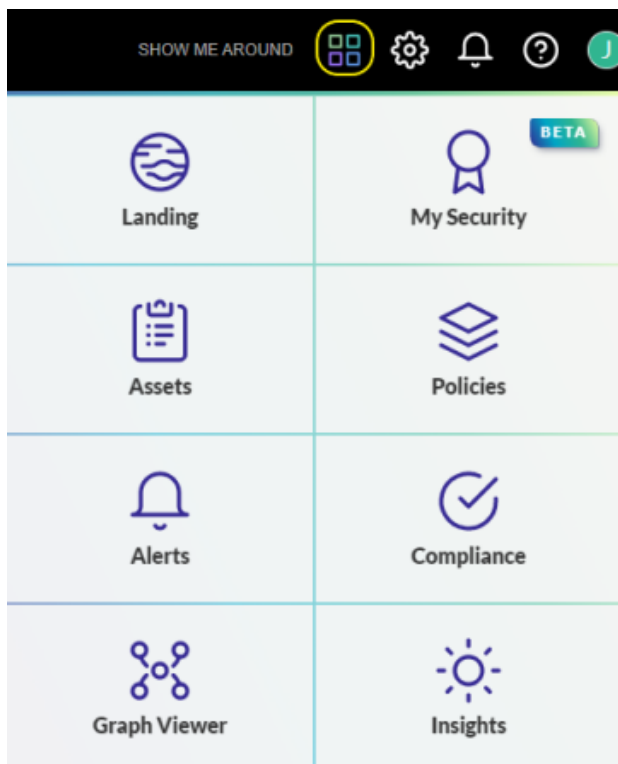
Secure Cloud Insights は、サイロ化されたセキュリティツール間を接続するクラウドネイティブのセキュリティプラットフォームであり、サイバーアセットユニバース全体のセキュリティリスクに対する障害物のない可視性を実現します。この拡張可能なプラットフォームは、複雑な関係とデータの間を点でつなぎ合わせ、環境、インフラストラクチャ、および運用に究極の可視性を提供します。

Secure Cloud Insights: 統合

Secure Cloud Insights を使用するための最初のステップは、Secure Cloud Insights にデータを取り込むことです。インストールが簡単で使いやすい、既成の統合が数多く用意されており、デジタルユニバースのあらゆる次元でエンドツーエンドのサイバーアセットの可視性、コンテキスト、自動化を実現することができます。Secure Cloud Insights では、Secure Cloud Insights にデータをインポートし、データモデルとマッピングを理解する手順を示します。

Secure Cloud Insights: アプリ

Secure Cloud Insights には、セキュリティ管理のすべての主要コンポーネントに役立つ個別のアプリがあります。 をクリックしてアプリを確認してください。



アセット

データをインポートしたら、Assets アプリを使用して、インフラストラクチャとセキュリティに関するサイバーアセットインベントリ全体を分析して可視化できます。さらに、Assets アプリは、所有しているサイバーアセットのタイプとクラス、およびそれらの関係に関する情報を提供します。

ポリシー

Policies アプリを使用すると、組織のポリシーを明確にして、ポリシーをコンプライアンス要件に関連付けることができます。

各ポリシーと手順のドキュメントは、それぞれ個別の Markdown ファイルに記述されており、他のファイルにリンクするように各ポリシーファイルを設定できます。テンプレートは、Policies アプリを使用してオンラインで直接編集できるオープンソースです。

簡単に利用を開始できるように、Secure Cloud Insights には、組織がセキュリティプログラムと運用を構築するのに役立つ 120 以上の [ポリシーと手順のテンプレート](#) が用意されています。これらのテンプレートは、Secure Cloud Insights の内部ポリシーと手順から派生したものであり、コンプライアンス評価を何度も経た上で作成されています。

アラート

Secure Cloud Insights では、継続的な監査と脅威の監視のため、任意のクエリを使用して Alerts アプリでアラートルールを構成できます。アラートをトリガーするには、少なくとも 1 つのアクティブなアラートルールが必要です。ルールをアラートに追加する最も簡単な方法は、Secure Cloud Insights で用意されているルールパックをインポートすることです。カスタムルールを作成することもできます。

コンプライアンス

Secure Cloud Insights は、コンプライアンス標準またはフレームワークを一連のコントロールまたは要件として管理するための柔軟なプラットフォームを提供します。このプラットフォームでは、次のことができます。

- コンプライアンス標準またはセキュリティに関するアンケートをインポートする
- ポリシー手順を各コントロールまたは要件にマッピングする
- クエリの質問によってデータ主導型のコンプライアンスの証拠をマッピングする
- クエリ結果に基づいて自動ギャップ分析を実行する
- コンプライアンス アーティファクトをエクスポートする (要約または完全な証拠パッケージ)

グラフビューア

Secure Cloud Insights はデータ主導型のグラフプラットフォーム上に構築されています。JupiterOne Query Language (J1QL) は、このグラフをトラバースしてサブグラフを返す、またはサブグラフのエンティティおよびエッジ (関係など) からのデータを返すように設計されています。任意の J1QL クエリ結果からサブグラフを表示して操作できます。


インサイト

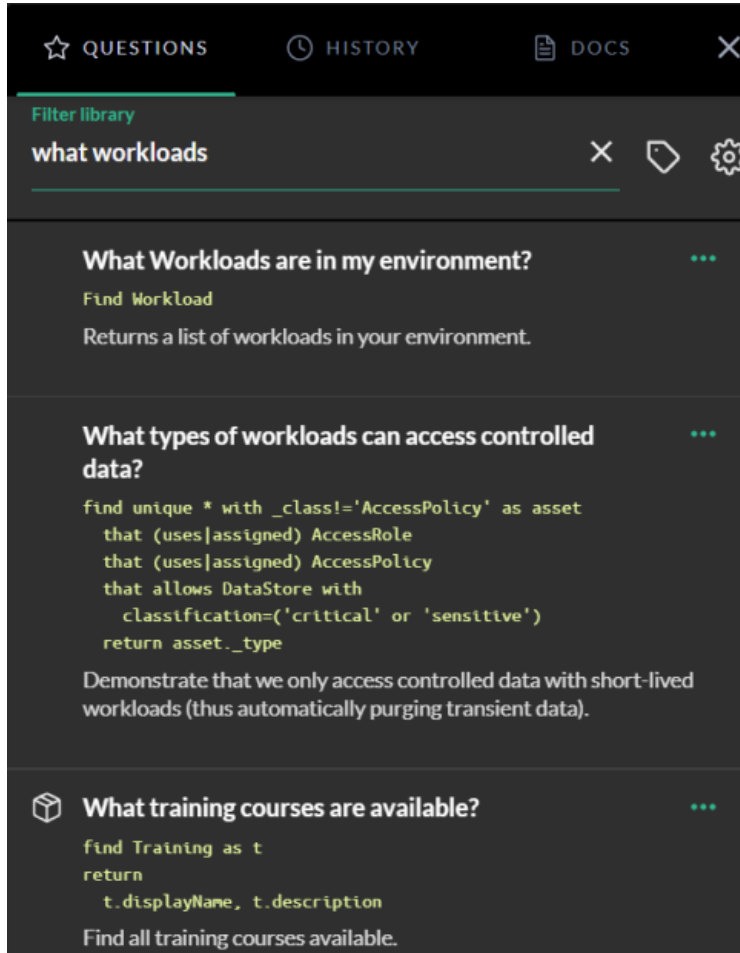
Insights アプリでは、J1QL クエリを使用してレポートダッシュボードを構築できます。

各ダッシュボードは、他のアカウントのユーザーと共有するチームボード、または個々のユーザーの個人ボードとして設定できます。各ボードのレイアウトは、チームボードのレイアウトを含めてユーザーごとに個別に保存されるため、各ユーザーは他のユーザーに影響を与えることなく、自分の好みに応じてレイアウトを設定できます。管理者は、チームボードのレイアウトを他のユーザーのデフォルトとして保存できます。

独自のカスタムダッシュボードを構築することも、Secure Cloud Insights すでに構築されている既存のボードを利用することもできます。

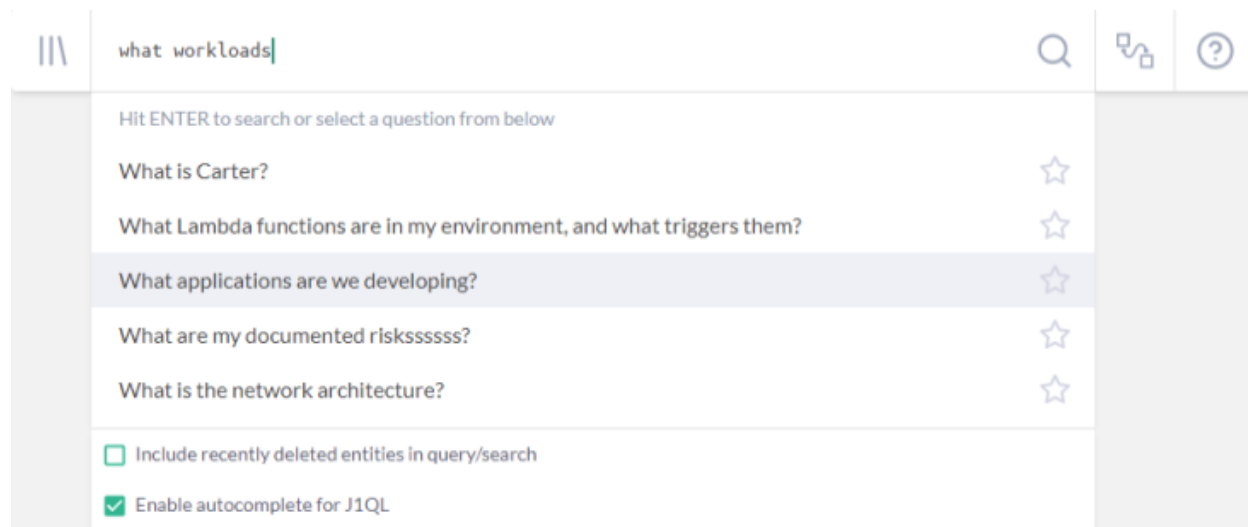
クエリライブラリ

Secure Cloud Insights には、サイバーアセットの現在の状態を評価するための、事前に構築および分類された何百ものクエリがあります。特定のトピックに関するクエリをフィルタ処理したり、既存のクエリを複製して独自のカスタムクエリを作成したり、よく使用する検索を保存して後で簡単に参照したりできます。クエリライブラリにアクセスするには、ランディングページの  をクリックします。



Ask Anything 検索バー

クエリライブラリを使用するだけでなく、Secure Cloud Insights の任意のページまたはアプリから検索バーに質問を入力することもできます。デフォルトで、Secure Cloud Insights はテキストをオートコンプリートし、ユーザーまたはユーザーの組織内の他の誰かが尋ねた関連する質問をリストします。



JupiterOne Query Language

JupiterOne Query Language (J1QL) は、デジタル環境内のエンティティと関係を検索するためのクエリ言語です。J1QL では、質問、全文検索の実行、または複雑なエンティティ関係グラフのクエリといった機能を組み合わせて使用することができます。

J1QL は複雑ですが、学習に役立つチュートリアルが用意されています。さらに、Secure Cloud Insights ではビジュアルクエリ構築アプリである J1VQB を用意しています。これはコード不要でクエリを作成できるツールです。

マネージド統合の設定

Secure Cloud Insights プラットフォームの機能を利用するには、このプラットフォームにデータが存在する必要があります。データが多いほど、このプラットフォームの機能は強力になります。

すぐに使用できるターンキー設定のマネージド統合は 10 を超えており、さらに定期的に追加されています。プロバイダーの必要に応じて、統合ごとに認証と設定のメカニズムが多少異なる場合があります。たとえば、AWS 統合では、アクセス用に IAM ロールとロールの信頼ポリシーを使用します。他の統合では、API キー/トークン、OAuth、または基本認証を使用する場合があります。

加えて、Secure Cloud Insights API クライアントまたは CLI を使用して、これらのマネージド統合の外部にデータをアップロードすることができます。これにより、オンプレミスシステムやセキュリティ/コンプライアンス アーティファクトなどのあらゆるデータを一元的に追跡、監視、視覚化できます。

検索の設定

ランディングゾーンから直接、Secure Cloud Insights と統合されたデジタル環境全体をすばやく検索してインサイトを得ることができます。検索には次の 3 つのモードがあります。

1. キーワードを入力して質問し、パッケージ化された/保存されたすべての質問を検索する
2. プロパティ値に基づいてすべてのエンティティを全文検索する
3. JupiterOne Query Language (J1QL) を使用してエンティティと関係を正確にクエリする

表、グラフ、Raw JSON、または Pretty JSON の 4 つの異なる結果表示モードを切り替えることができます。



パフォーマンス上の理由から、検索結果は最大 250 アイテムを返すように制限されていることに注意してください。大量の結果セットから見落としが生じると思われる場合は、クエリを調整してより正確な結果を生成してみてください。

質問する

次のようなキーワード(またはキーワードの組み合わせ)を入力するだけです(引用符は不要)。

- 準拠
- アクセス
- トラフィック
- ssh
- data encrypted
- 実稼働

または、次のような質問をします。

- Who are my vendors? (自分のベンダーは?)
- What lambda functions do I have in AWS? (AWS にはどのようなラムダ関数がありますか?)
- What is connected to the Internet? (インターネットに接続中のものは?)
- Who has access to ...? (誰が...にアクセスできますか?)

全文検索

全文検索を開始するには、キーワードを引用符で囲みます(“キーワード”)。または、キーワードを入力して Enter キーを押します。次に例を示します。

- “sg-123ab45c” と入力すると、このグループ ID を持つ AWS EC2 セキュリティグループを検索します。
- “Charlie” と入力すると、この名前の人物/ユーザー、およびその人物/ユーザーに関連する他のリソースを検索します。

JupiterOne Query Language (J1QL)

ここでは、JupiterOne Query Language (J1QL) を使用して、すべてのエンティティと関係の全体を検索します。

基本的なクエリ構造は次のとおりです。

- エンティティから始めます。

```
FIND {class or type of an Entity}
```
- 必要に応じていくつかのプロパティフィルタを追加します。

```
WITH {property}={value} AND|OR {property}={value}
```
- その関係を取得します。

```
THAT {relationship_verb}|RELATES TO {class/type of another Entity}
```

次に例を示します。

```
FIND * WITH tag.Production='true'
```

(上記のワイルドカード * はすべてを含めるために使用されていることに注意してください)

人物であるユーザーを探す

正確な関係がわからない場合は、キーワード `RELATES TO` を使用すると、あらゆる関係を対象に含めることができます。

```
FIND User THAT RELATES TO Person
```

エイリアスと `AS {something}` を一緒に使用して、エンティティまたは関係に名前を付けることができます。エイリアスは、`WHERE` で使用すると追加のフィルタリングまたは比較が実行され、`RETURN` で使用すると特定のプロパティを返すことができます。

次に例を示します。

```
FIND Firewall AS fw
  THAT ALLOWS AS rule (Network|Host) AS n
WHERE
  rule.ingress=true and rule.fromPort=22
RETURN
  fw._type, fw.displayName, fw.tag.AccountName,
  n._type, n.displayName, n.tag.AccountName
```

クエリ言語では、次の場合を除き、大文字と小文字が区別されません。

- Find の後の TitleCase エンティティキーワードと {relationship verb} は、そのクラスのエンティティを検索します (例: CodeRepo)。
- Find の後の lowercase エンティティキーワードと {relationship verb} は、そのタイプのエンティティを検索します。複数の単語を含むエンティティタイプは、通常、snake_case にあります (例: github_repo)。
- エンティティのプロパティの名前と値、およびクエリの一部として定義されたエイリアス名は、大文字と小文字が区別されます。

全文検索と J1QL の組み合わせ

まず全文検索を行い、次に J1QL を使用して、最初の検索結果をさらにフィルタリングすることもできます。次に例を示します。

```
Find "Administrator" with _class='AccessPolicy' that ASSIGNED (User|AccessRole)
```

```
Find 'security officer' with _type='employee'
```

```
Find 'roles responsibilities' with _class=('Policy' or 'Procedure')
```

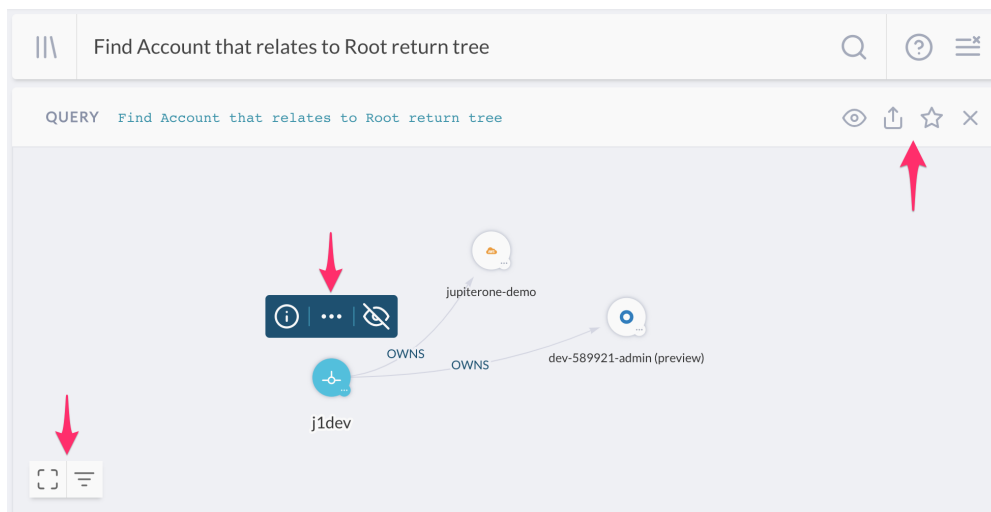
一重引用符 (') または二重引用符 (") は、全文検索キーワードとプロパティ文字列値の両方で機能することに注意してください。

グラフの操作

Secure Cloud Insights はデータ主導型のグラフプラットフォーム上に構築されています。開発のきっかけとなったストーリーについては、こちらのブログをご覧ください。

JupiterOne クエリ言語 (J1QL) は、このグラフをトラバースしてサブグラフを返す、またはサブグラフのノード (エンティティなど) とエッジ (関係など) からのデータを返すように設計されています。クエリ結果からサブグラフを表示して操作できます。





以下のスクリーンショットは、Ask Anything 検索バーのクエリからの結果グラフの例を示しています。




右上隅の最初のコントロールセットは、次のことを行います。

Control	機能
👁️	ビューの切り替え: テーブル、グラフ、Raw JSON、Pretty JSON を切り替えます。
📄	クエリを共有: コピーおよび共有用の Web リンクを含むモーダルポップアップが表示されます。
☆	クエリを保存: モーダルポップアップが表示されるので、タイトルや説明、必要に応じてタグを指定して、独自のクエリライブラリに保存できます。
✕	結果を削除: この特定のクエリ/質問の結果をページビューから削除します。

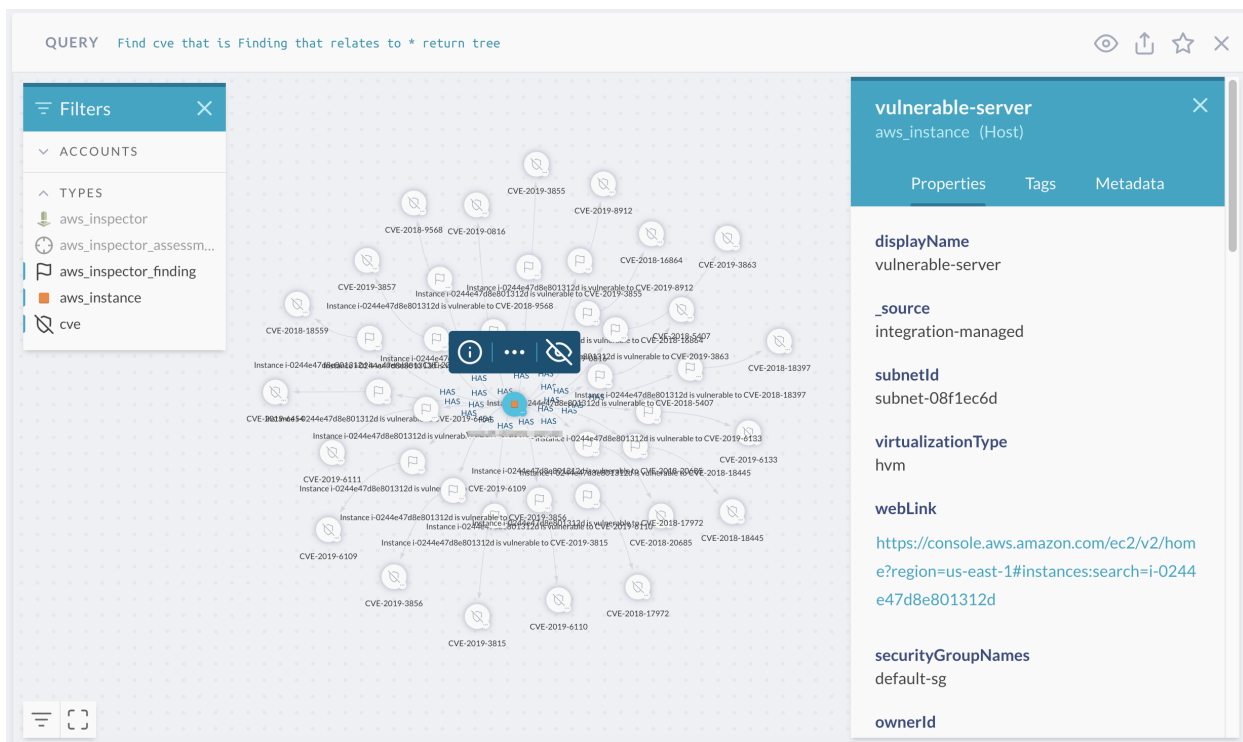
グラフ上の任意のノード (エンティティなど) を選択すると、そのノードと対話できるコントロールのセットがそのすぐ上に表示されます。これらのコントロールは次の機能を提供します。

Control	機能
	サイドパネルを開く: 選択したエンティティの詳細なプロパティ、タグ、およびメタデータを表示します。 サイドパネルでエッジを選択し、そのプロパティを確認することもできることに注意してください。
...	ネイバーをロード: コピーおよび共有用の Web リンクを含むモーダルポップアップが表示されます。
	選択したノードを非表示: グラフから煩雑さを軽減します。左下のコントロールから、すべての非表示のノードを再表示できます。
	グループ化されたノードを展開: 同じ親ノードを持つ同じタイプのノードを展開します。グラフのデータによっては、このオプションは常に使用できるとは限りません。
	ノードを折りたたむ: 同じ親ノードを持つ同じタイプのノードをグループに折りたたみます。グラフのデータによっては、このオプションは常に使用できるとは限りません。




最後のコントロールセットはグラフの左下隅にあり、次の操作を実行します。

Control	機能
	全画面モードでグラフを 最大化 します。
	クエリ結果コンポーネントのグラフを 復元 します。
	フィルタパネルを開く: アカウントやタイプでグラフ上のノードをフィルタ処理 (表示/非表示) できます。
	非表示のノードを再表示: このコントロールアイコンは、グラフに非表示のノードがある場合にのみ表示されます。

Property パネルと Filter パネルを開いた状態のグラフのスクリーンショットを次に示します。



ズームおよび移動

Control	機能
 	マウス/タッチパッドを使用してスクロールして、グラフを拡大/縮小します。
	マウス/タッチパッドを使用してグラフ上の空白の場所をクリックしてドラッグし、グラフを移動します。選択したノードをクリックしてドラッグすると、その特定のノードが移動します。

スタンドアロンの Galaxy/Graph Viewer アプリは、同じコントロールセットを使用します。

JupiterOne クエリ言語について

クエリの作成は最も難易度の高い作業ですが、Secure Cloud Insights エクスペリエンスの中でも最も楽しく、やりがいのある箇所です。クエリ言語に慣れれば、これまで発見されていなかったあらゆる種類のインサイトをデータから発見できることでしょう。

JupiterOne Query Language (J1QL) は、デジタル環境内のエンティティと関係を検索するためのクエリ言語です。J1QL では、質問、全文検索の実行、または複雑なエンティティ関係グラフのクエリといった機能を組み合わせて使用することができます。

予めパッケージ化されたクエリが豊富にあり、Ask Anything 検索バーで簡単に使用することも、クエリライブラリで参照することもできます。このチュートリアルでは、それらの代わりに、カスタムクエリを自分で作成できるようにすることに重点を置きます。

このチュートリアルは、J1QL の詳細なマニュアルに基づいて、いくつかの一般的なユースケースを使用して構築されています。

i ここで紹介するサンプルクエリは、データソースの特定の構造に基づいて変更できます。

パート 1 – 単純なルートクエリ

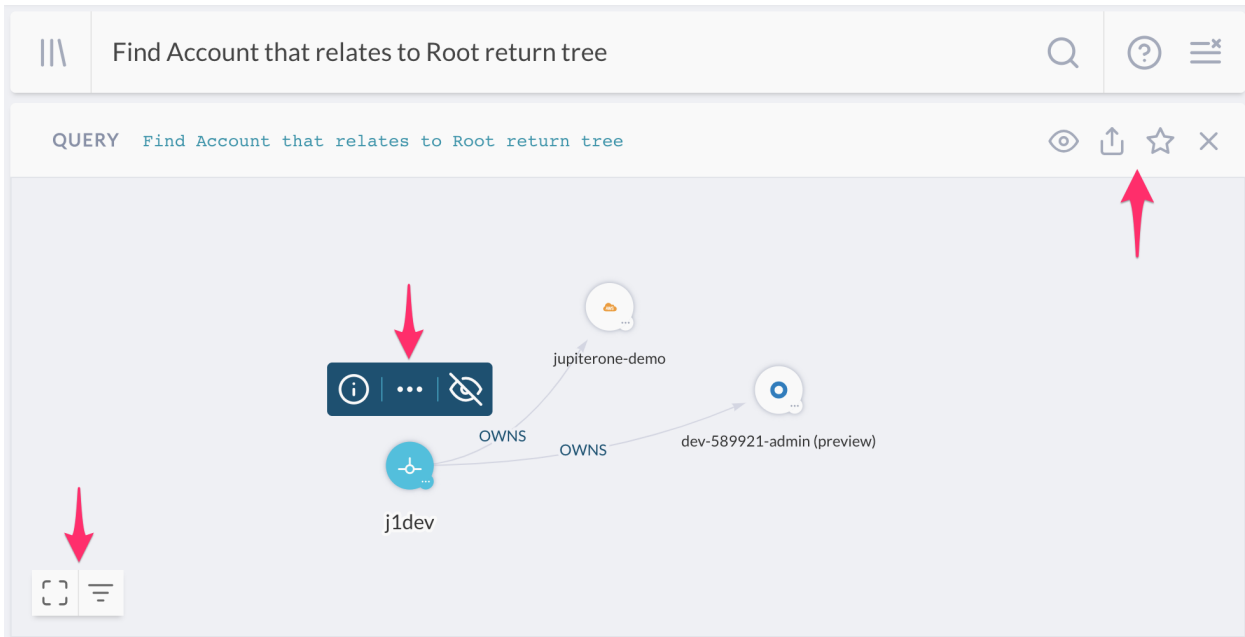
まず、次のクエリを試してみましょう。

```
Find Account that relates to Root return tree
```

動詞の直後に続く名詞は、大文字と小文字が区別されることに注意してください。

- TitleCase の単語は、そのクラスのエンティティを検索するようにクエリに指示します (例: Account、Firewall、Gateway、Host、User、Root、Internet など)。
- snake_case の単語は、そのタイプのエンティティを検索するようにクエリに指示します (例: aws_account、aws_security_group、aws_internet_gateway、aws_instance、aws_iam_user、okta_user、user_endpoint など)。

次のような結果が得られるはずですが (クエリの `return tree` の部分は、デフォルトでグラフビューを表示するように指示します)。



上記の例で選択されているノードは、組織を表す特別な Root ノードです。所有している統合構成の数に応じて、表示される接続中アカウントの数は異なり、Root エンティティがこれらの Account エンティティを所有 (OWN) していることを示します。

結果パネルの 3 つのコントロールセットをご覧ください。右上から左下への順に見ていきます。

最初のコントロールセット(クエリの隣)では、次のことができます。

- テーブル、グラフ、Raw JSON、Pretty JSON の間でビューを切り替えます。
- クエリを共有: コピーおよび共有用の Web リンクを含むポップアップボックスが表示されます。
- クエリを保存: タイトルや説明、必要に応じてタグを付けて、独自のクエリライブラリに保存します。
- この結果パネルを閉じるか、ページから削除します。

2 番目のコントロールセット(選択したエンティティノードの上)では、次のことができます。

- 選択したエンティティの詳細なプロパティ、タグ、およびメタデータを表示します。
- エンティティを展開して、接続されているネイバーをさらに表示します。これにより、元のクエリでは返されなかった可能性のある追加のデータが表示され、検索と分析をさらに進めることができます。
- 選択したエンティティノードをグラフビューから非表示にします。エンティティを非表示にすると、グラフの左下にある 3 番目のコントロールセットに再表示ボタンが表示されます。このボタンを使用して、現在非表示になっているすべてのエンティティを再表示できます。

最後のコントロールセット(左下隅)では、次のことができます。

- 全画面表示を切り替えます。
- フィルタパネルを開き、アカウントまたはエンティティタイプ別にグラフ内のエンティティを表示/非表示にします。

- 現在非表示のエンティティをすべて再表示します(上のスクリーンショットでは表示されていません。少なくとも1つの非表示のエンティティがある場合にのみ表示されます)。

グラフコントロールの詳細については、「[グラフの操作](#)」を参照してください。

パート 2 – インフラストラクチャの分析

i このセクションの例では、少なくとも1つの AWS 統合設定が必要です。

AWS 統合の設定が完了したら、さらに興味深い事柄を試してみましょう。次のクエリを入力するか、コピーして貼り付けます。

2a – SSH Key Usage Examples

```
Find AccessKey with usage='ssh'
```

すると、EC2 インスタンスへの SSH アクセスに使用される `aws_access_key` エンティティのセットが見つかるはずです(そのようなエンティティがいくつかあり、SSH アクセスを許可するように設定されていると仮定します)。

クラスではなくエンティティタイプでクエリを実行することもできます。次のクエリでも同じ結果が得られます(他の統合(AWS 以外)または UI / API から追加した SSH キーもある場合は除きます)。

```
Find aws_key_pair
```

では、次のように検索を少し拡張しましょう。

```
Find Host as h
  that uses AccessKey with usage='ssh' as k
  return
  h.tag.AccountName,
  h._type,
  h.displayName,
  h.instanceId,
  h.region,
  h.availabilityZone,
  h.publicIpAddress,
  h.privateIpAddress,
  h.platform,
  h.instanceType,
  h.state,
  k._type,
  k.displayName
```

これにより、各 `AccessKey` を使用(USE)する `Host` エンティティが検出され、特定のプロパティのセットが返されます。必要に応じて、返されたプロパティを追加または削除できます。

i キーワード `that` は、その後に関係クラスを表す動詞が続いて、グラフをトラバースしてエンティティ間の接続/関係を見つけるようにクエリに指示するものであることに注意してください。

また、グラフビューに切り替えると、より視覚的な結果を得て、インタラクティブにドリルダウンを続けることができます。

ここでも同様に、より具体的なエンティティタイプを使用してクエリを実行できます。次に例を示します。

```
Find aws_instance that uses aws_key_pair
```

または、これらを組み合わせて一致させることもできます。

```
Find Host that uses aws_key_pair
```

i 関係のキーワード/動詞では大文字と小文字が区別されないことに注意してください。

2b – EBS ボリュームの例

まず、暗号化されていない EBS ボリュームがあるかどうかを確認しましょう。

```
Find aws_ebs_volume with encrypted != true
```

i 上記のクエリでは、`with` キーワードはそのすぐ左側のエンティティの名詞にバインドし、そのエンティティのプロパティ値で結果をフィルタリングできることに注意してください。

上記のクエリで暗号化されていない EBS ボリュームが見つかった場合、何がそれらを使用しているかを確認してみましょう。

```
Find Host that uses aws_ebs_volume with encrypted != true
```

`aws_ebs_volume` エンティティとその関係をグラフモードで表示し、各エンティティノードまたは関係エッジのプロパティをさらに調べることができます。展開し、接続されたエンティティと関係をさらに表示することもできます。

これらのエンティティはアクティブに使用されているでしょうか。本番環境ではどうでしょう。

```
Find Host with active = true and tag.Production = true
that uses aws_ebs_volume with encrypted != true
```

これらのインスタンスはどのサブネットにあるでしょうか。また、この検索に関連するエンティティのタイプからいくつかの重要なプロパティを返してみましょう。

```
Find Network as n
  that has Host as h
  that uses aws_ebs_volume with encrypted != true and tag.Production = true as e return
  n.displayName, h._type, h.displayName, e.displayName, e.encrypted
```

もちろん、アクティブに使用されていないEBS ボリュームはどうでしょうか。一部は削除できるかもしれません。

```
Find aws_ebs_volume that !uses Host
```

上記のクエリは逆方向だと感じられるかもしれませんが、問題ありません。クエリは、関係の方向を問わず同じように機能します。クエリはデフォルトでエンティティの初期セットからすべてのプロパティを返すように設定されているため、クエリの方向を反対にする方が探しているデータをより簡単に取得できる場合があります。

技術的には、`Find Host that !uses aws_ebs_volume as v return v.*`の方が正しいと思われるかもしれませんが、入力が少し増えるのは確かです。

2c - 暗号化されていないデータ

AWS にはさまざまな種類のデータストアがあります。たとえば、EBS ボリューム、S3 バケット、RDS クラスターとインスタンス、DynamoDB テーブル、Redshift クラスターなどです。機密データを保存する場合、これらのデータストアを暗号化することをお勧めします。

その場合、どのように検索すればよいでしょうか。

```
Find (aws_s3_bucket|aws_rds_cluster|aws_db_instance|aws_dynamodb_table|aws_redshift_
cluster) with encrypted!=true
```

上記のクエリは確かに機能しますが、かなり複雑です。ここで、Secure Cloud Insights によって自動的に割り当てられた抽象クラスのラベル付けがその目的を果たすこととなります。クラスごとにクエリを実行する方がはるかに簡単です。

```
Find DataStore with encrypted != true
```

これで、プロパティフィルタをいくつか追加して結果をさらに絞り込み、ノイズを減らしたり、修復を優先したりすることができます。次に例を示します。

```
Find DataStore with
  encrypted != true and
  tag.Production = true and
```

```
(classification = 'confidential' or classification = 'restricted')
```

2d – Tagging Resources

前述のいくつかの例からわかるように、リソースをタグ付けすると操作上非常に便利です。そのため、ソースでリソースにタグを付けることを強くお勧めします。これらのタグは Secure Cloud Insights によって取り込まれ、カスタムクエリで使用できます。

デフォルトで、クエリライブラリに表示され、Compliance アプリで使用される Secure Cloud Insights が提供するパッケージ化されたクエリは、次のタグに依存します。

- 分類 (Classification)
- [オーナー (Owner)]
- PII または PHI または PCI (boolean タグでデータ型を示す)
- アカウント名 (AccountName)
- 実稼動

Secure Cloud Insights の統合によって取り込まれたすべてのカスタムタグには、tag.<TagName> がプレフィックスとして付けられます。これらのプレフィックスはクエリでそのまま使用する必要があります。

Classification タグと Owner タグはプロパティとして自動的にキャプチャされるため、tag. プレフィックスなしでクエリで直接使用できます。classification = '...' または owner = '...' のようにすべて小文字で使います。

tag.AccountName (string) および tag.Production (boolean) タグは、各統合構成の詳細オプションの一部として追加できます。Secure Cloud Insights

2e – ネットワークリソースと構成

ネットワークリソースおよびその構成について、質問や確認事項がある場合があります。次に例を示します。

ネットワークリソースとその接続の検索から始めましょう。

```
Find (Gateway|Firewall) with category='network'
  that relates to *
  return tree
```

必要に応じて、結果をテーブルビューに切り替えることができることに注意してください。

ネットワークとサブネットはどうでしょうか。

```
Find Network that contains Network return tree
```

VPC 内のリソースはどうでしょうか。

Find Network that has (Host|Cluster|Database) return tree

結果は次のようになります(ここでデモ環境から表示されるものよりも多数の結果が表示される場合があります)。

QUERY Find Network that HAS (Host|Cluster|Database) return tree

subnet-ddcb48f1 (172.31.80.0/20) ×

aws_subnet (Network)

Properties Tags Metadata

subnetId
subnet-ddcb48f1

internal
true

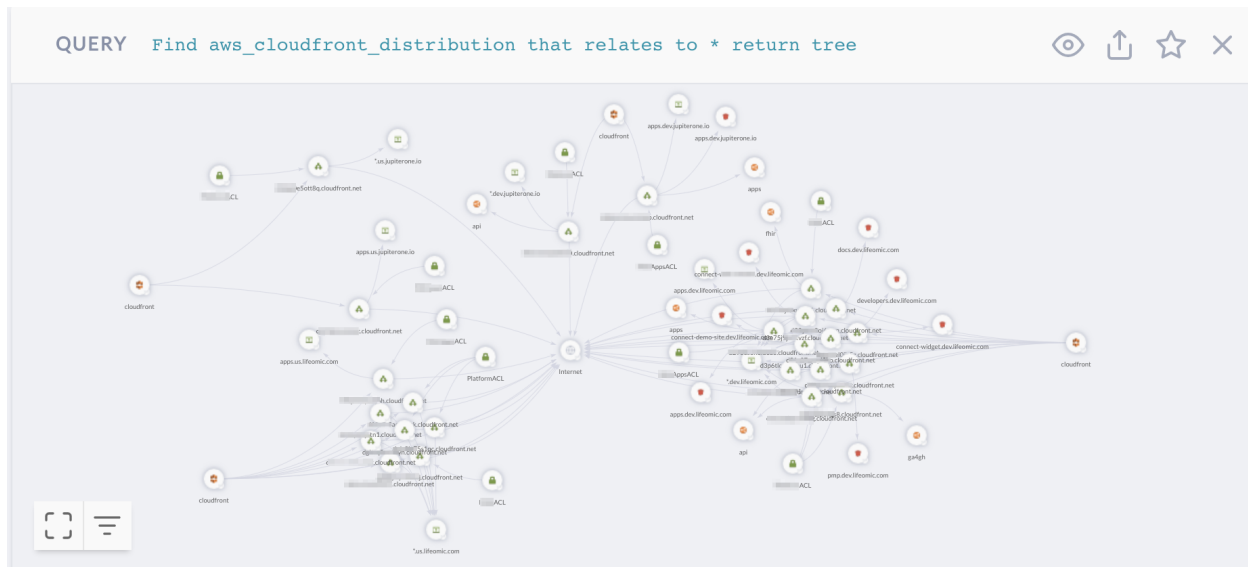
webLink
<https://console.aws.amazon.com/vpc/home?reg>

i 選択した `aws_subnet` のプロパティパネルには、AWS Web コンソールで直接ソースにすばやくアクセスできる `webLink` があることに注意してください。

AWS では、ほとんどの場合、S3 でホストされている API ゲートウェイまたは静的ウェブサイトにはトラフィックを分散するように、**CloudFront ディストリビューション**を設定しています。これはどのように表示されるでしょうか。

Find `aws_cloudfront_distribution` that relates to * return tree

今回の結果は少し複雑なようです。アカウントには複数の AWS 統合構成があり、かなりの数の `aws_cloudfront_distribution` エンティティおよび関係があります。



このグラフは、S3 バケット（静的 Web サイト/コンテンツ用）と API ゲートウェイの両方のディストリビューションに接続された起点を示しています。さらに、これらの起点によって使用されている ACM 証明書と、これらを保護するように構成されている WAF ACL（あれば）が表示されます。

グラフ内の任意のエンティティノードを選択してその詳細なプロパティを調べたり、Web リンクを検索して AWS Web コンソールのソースにすばやくアクセスしたりできることに注意してください。

AWS Transfer for SFTP を使用する場合、転送サーバー、ユーザー、それらに割り当てられている IAM ロール、およびユーザーがアクセスできる S3 バケット 検索することができます。

```
Find aws_account
that HAS aws_transfer
that HAS Host
that HAS User
that RELATES TO *
return tree
```

次のようなビジュアルが表示されます。



2f- サーバーレス関数

サーバーレス(ラムダ関数)を使用していますか？使用している場合、その設定方法を理解するのに役立つクエリをいくつかご紹介します。

ラムダ関数を一覧表示することから始めましょう。

```
Find aws_lambda_function
```

[Simple]。では、各関数をトリガーするものは何でしょうか。

```
find aws_lambda_function as function
  that TRIGGERS * as trigger
  return
  trigger._type, trigger.displayName, trigger.arn, trigger.webLink, function.functionName,
  function.arn, function.webLink
```

VPC 内のリソースにアクセスできるラムダ関数はあるでしょうか。

```
Find aws_lambda_function that has aws_vpc return tree
```

上記のクエリは、ラムダ関数と、それらのラムダ関数を内部で実行するように構成されている VPC の視覚的なグラフを提供します。

実際には、VPC 内のリソース(EG2 インスタンス、RDS データベース、ElasticSearch/ElastiCache など)に直接アクセスする必要がない限り、VPC へのアクセスなしでラムダ関数を実行しないことがベストプラクティスです。

外部ホストまたはネットワークからインバウンド SSH は直接許可されているでしょうか。


```
Find Firewall as fw
  that ALLOWS as rule (Host|Network)
  with internal=false or internal=undefined as src
  where rule.ingress=true and (rule.fromPort<=22 and rule.toPort>=22)
  return
  fw._type,
  fw.displayName,
  rule.fromPort,
  rule.toPort,
  src.displayName,
  src.ipAddress,
  src.CIDR
```

i 上記のクエリでは、関係のプロパティ値をフィルタ処理するために `where` が使用されていることに注意してください。`with` と `where` の両方を使用して、エンティティのプロパティ値をフィルタ処理できます。詳細については、J1QL の詳細なマニュアルを参照してください。また、**グラフビュー**に切り替えることで、上記の結果をより視覚的かつインタラクティブに表示できることにも注意してください。

インターネット(全員)に直接接続(公開)されている本番リソースはどれでしょうか。

```
Find (Internet|Everyone)
  that relates to *
  with tag.Production=true and _class!='Firewall' and _class!='Gateway'
  return tree
```

自分のネットワークレイヤのリソースはどれでしょうか。

```
Find (Firewall | Gateway) with category='network'
```

セキュリティグループの保護はどうでしょうか。

```
Find aws_security_group that PROTECTS aws_instance return tree
```

i ヒントグラフ内のエッジを選択すると、セキュリティグループルールの詳細(エッジのプロパティなど)が表示されます。

パート 3 – ユーザーとアクセスの分析

Okta または OneLogin の統合を設定したら、次に示すクエリの例をいくつか試してみてください。

3a – IdP ユーザーとアクセス

i このセクションの例では、ID プロバイダーの統合 (Okta または OneLogin) が必要です。

個々の従業員/ユーザーに属さないシステムアカウントはあるでしょうか。

```
Find User that !is Person
```

Secure Cloud Insights の User エンティティは、自動的に対応する Person (type: 'employee') エンティティにマッピングされます (Okta または OneLogin など、少なくとも 1 つの ID プロバイダー (IdP) 統合構成が存在する場合)。

i ヒント IdP アカウントのユーザープロファイルの userType プロパティを「system」または「generic」または「bot」に設定すると、そのユーザーの Person エンティティが作成されなくなります。Secure Cloud Insights

i ヒント aws_iam_user またはその他の非 IdP ユーザーの username を個人/従業員の電子メールアドレスに設定すると、そのユーザーを対応する個人に自動的にマッピングできません。Secure Cloud Insights は、マッピングが機能するように aws_iam_user に email タグを追加することもできます。

多要素認証が有効でないアクティブなユーザーアカウントはどれでしょうか。

```
Find User with active = true and mfaEnabled != true
that !(ASSIGNED|USES|HAS) mfa_device
```

特定の IdP 統合によっては、User エンティティは、mfaEnabled フラグをプロパティとして直接指定する代わりに、mfa_device への関係マッピングがある場合があります。

したがって、上記のクエリは、プロパティで active フラグが設定されているが、mfaEnabled フラグが true に設定されていないすべての User エンティティを検索し、さらに、その User と割り当てられているか使用中の mfa_device との間に関係が存在するかどうかをチェックします。

MFA を使用せずに「AWS」アプリケーションにアクセスしているユーザーはいるでしょうか。

```
Find User with active = true and mfaEnabled != true
that ASSIGNED Application with displayName = 'Amazon Web Services'
```

別のアプリケーションを確認するには、displayName の文字列値を置き換えます。

IdP で複数の AWS SAML アプリが設定されている場合、`shortName = 'aws'` を使用して、すべての AWS アプリケーションインスタンスをチェックすることもできます。

環境内のすべての請負業者と外部ユーザーを検索しましょう。

```
Find User that IS Person that !EMPLOYS Root
```

上記のクエリは、組織 (Root エンティティ) に直接雇用されていない個人に属するユーザーアカウントを検索します。

```
Find User as u that IS Person as p
  where u.userType='contractor' or p.employeeType='contractor'
```

上記のクエリは、請負業者のユーザーを検索します。

3b – クラウドユーザーとアクセス

i このセクションの例では、少なくとも 1 つの AWS 統合設定が必要です。

AWS で管理者のフルアクセス権が割り当てられているのは誰でしょうか。

```
find (aws_iam_role|aws_iam_user|aws_iam_group)
  that ASSIGNED AccessPolicy with policyName='AdministratorAccess'
```

どの IAM ロールにどの IAM ポリシーが割り当てられているのでしょうか。

```
find aws_iam_role as role
  that ASSIGNED AccessPolicy as policy
  return
  role._type as RoleType,
  role.roleName as RoleName,
  policy._type as PolicyType,
  policy.policyName as PolicyName
```

3c – Combined Users and Access Across All Environments

i このセクションの例は、IdP と AWS の両方の統合設定が有効になっている場合に最も適切に機能します。Secure Cloud Insights

誰がどのシステム/リソースにアクセスできるでしょうか。

```
Find (User|Person) as u
  that (ASSIGNED|TRUSTS|HAS|OWNS)
    (Application|AccessPolicy|AccessRole|Account|Device|Host) as a
  return
  u.displayName, u._type, u.username, u.email,
  a._type, a.displayName, a.tag.AccountName
  order by u.displayName
```

パート 4 – Cross Account Analysis

i このセクションの多くの例では、Secure Cloud Insights での Okta と AWS の両方の統合設定と、Okta アカウントで設定された AWS SAML アプリが必要です。一部のクエリは、複数の AWS 構成がある場合に最も適切に機能します。

シングルサインオン(SSO)を介して AWS アカウントにアクセスできるのは誰でしょうか。

```
Find User as U
  that ASSIGNED Application as App
  that CONNECTS aws_account as AWS
  return
  U.displayName as User,
  App.tag.AccountName as IdP,
  App.displayName as ssoApplication,
  App.signOnMode as signOnMode,
  AWS.name as awsAccount
```

ある AWS アカウントから他の外部エンティティへのロールの信頼が想定されているでしょうか。

```
Find aws_account
  that HAS aws_iam
  that HAS aws_iam_role
  that TRUSTS (Account|AccessRole|User|UserGroup) with _source='system-mapper'
  return tree
```

i 上記のクエリから、`_source='system-mapper'` は、信頼できるエンティティが統合設定によって取り込まれたものではなく、アカウントの IAM ロールの想定ロールポリシーの分析中に Secure Cloud Insights によってマッピングおよび作成されたことを示すインジケータであることに注意してください。したがって、これらのエンティティは外部エンティティである可能性が最も高くなります。

たとえば、Secure Cloud Insights AWS アカウントへの `TRUSTS` 関係を持つ Secure Cloud Insights 統合 IAM ロールがほぼ確実に表示されるでしょう。

パート5 – エンドポイントコンプライアンス

i このセクションの例では、Stethoscope アプリが提供する少なくとも1つのエンドポイントコンプライアンス エージェントをアクティブ化している必要があります。

エンドユーザーデバイスでローカルファイアウォールが有効になっているでしょうか。

```
Find HostAgent as agent
  that MONITORS user_endpoint as device
  return
  device.displayName,
  device.platform,
  device.osVersion,
  device.hardwareModel,
  device.owner,
  agent.firewall,
  agent.compliant,
  agent._type,
  agent.displayName
```

どのユーザーのエンドポイントが準拠していないでしょうか。

```
Find Person as person
  that OWNS (Host|Device) as device
  that MONITORS HostAgent with compliant!=true as agent
  return
  person.displayName,
  person.email,
  device.displayName,
  device.platform,
  device.osVersion,
  device.hardwareModel,
  device.owner,
  agent.compliant,
  agent._type,
  agent.displayName
```

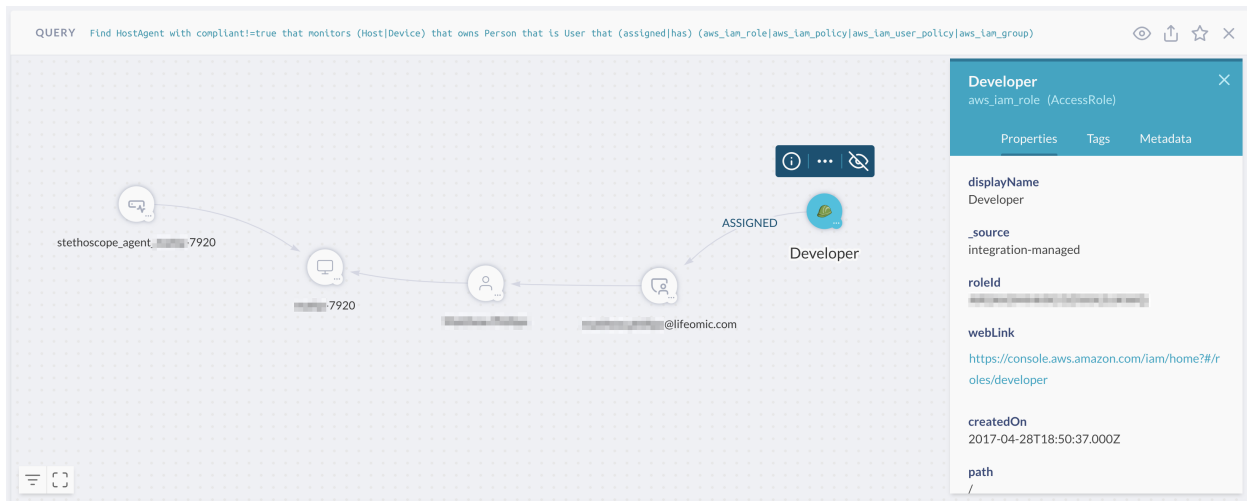
それらのユーザーはどのアプリケーションにアクセスできるでしょうか。

```
Find HostAgent with compliant!=true
  that MONITORS (Host|Device)
  that OWNS Person
  that IS User
  that Assigned Application
  return tree
```

上記のうち、AWS にアクセスできるユーザーはいるでしょうか。

```
Find HostAgent with compliant!=true
that MONITORS (Host|Device)
that OWNS Person
that IS User
that (ASSIGNED|HAS) (aws_iam_role|aws_iam_policy|aws_iam_user_policy|aws_iam_group)
return tree
```

結果のグラフは次のようになります。



アセットインベントリアプリでのフィルタの使用

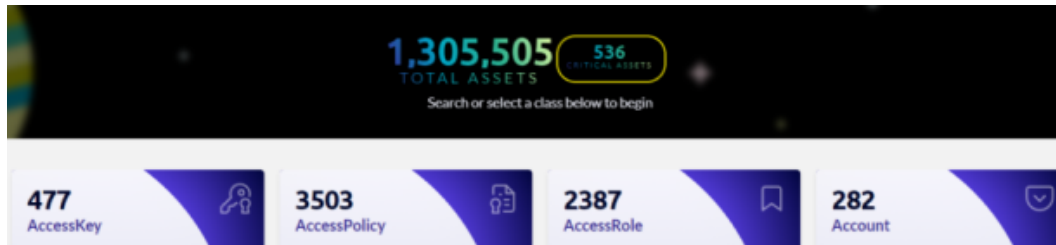
所有しているすべてのデジタルアセット(エンティティ)を Secure Cloud Insights で表示するには、[アプリ(Apps)]>[アセット(Assets)]に移動します。

Assets アプリに表示される大量のエンティティのリストをフィルタリングするには、いくつかの方法があります。

- 重要なアセットのクイックフィルタ
- クラス/タイプ別の追加フィルタ
- プロパティ別の詳細フィルタ

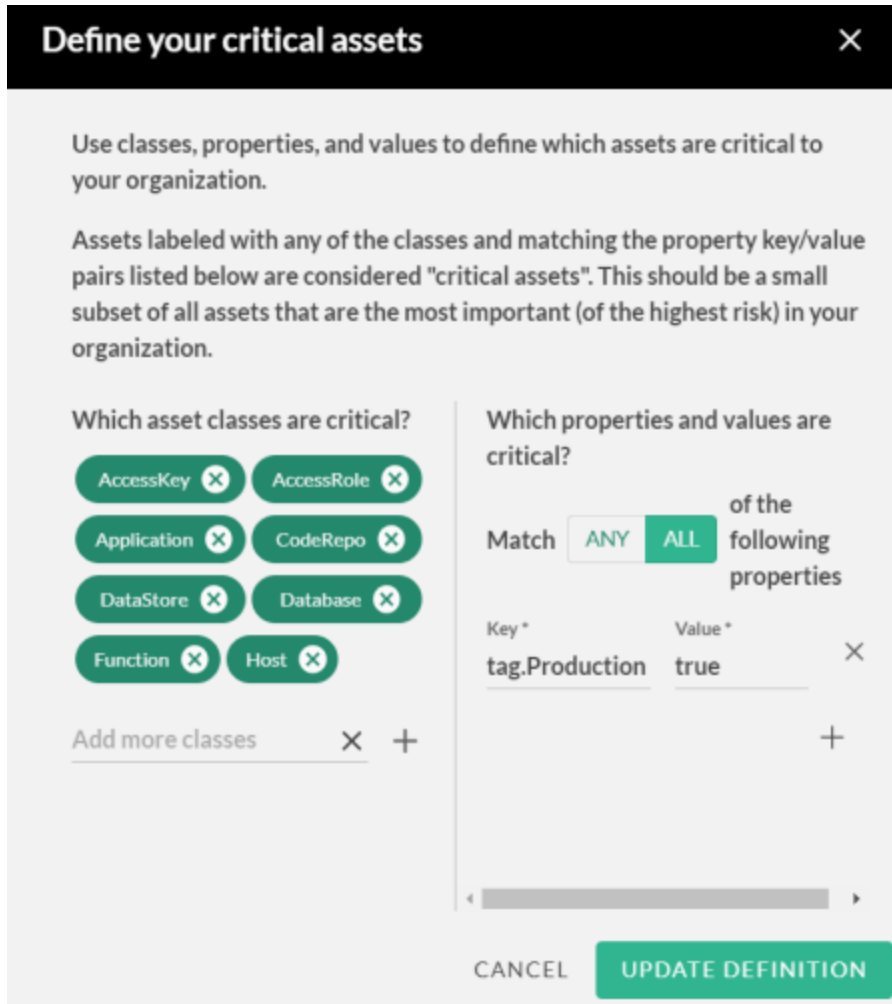
重要なアセットのクイックフィルタ

トップバナーの [重要なアセット(Critical Assets)] をクリックして、最も重要なエンティティに直接移動します。



重要なアセット機能を使用すると、クエリとアラートを作成して、最も重要なデータにすばやくアクセスできます。デフォルトでは、アセットを最も重要で、したがって最もリスクが高いものとして定義する基準は Secure Cloud Insights が決定しますが、管理者はこの定義を編集することができます。

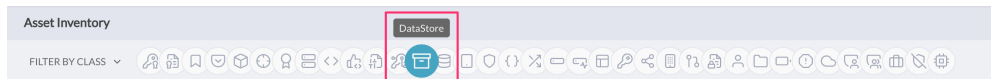
⚙️ をクリックして、重要なアセットを定義するデフォルト値を編集します。クラス、プロパティ、および値を使用して、何を重要とするかを定義できます。



組織にとって重要なアセットとなるアセットクラスとプロパティを追加し、[定義の更新 (UPDATE DEFINITION)] をクリックします。

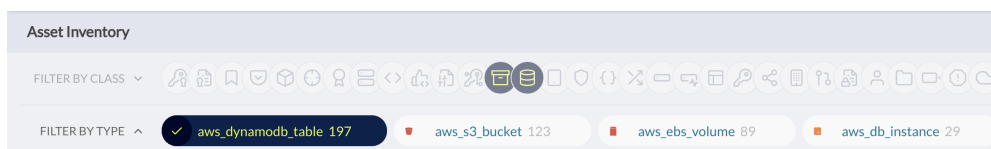
クラス/タイプ別のクイックフィルタ

各クラスを表す 1 つ以上のアイコンを選択することにより、クラス別にアセットをすばやくフィルタリングできます。ツールチップ上にカーソルを移動すると、クラスラベルが表示されます。



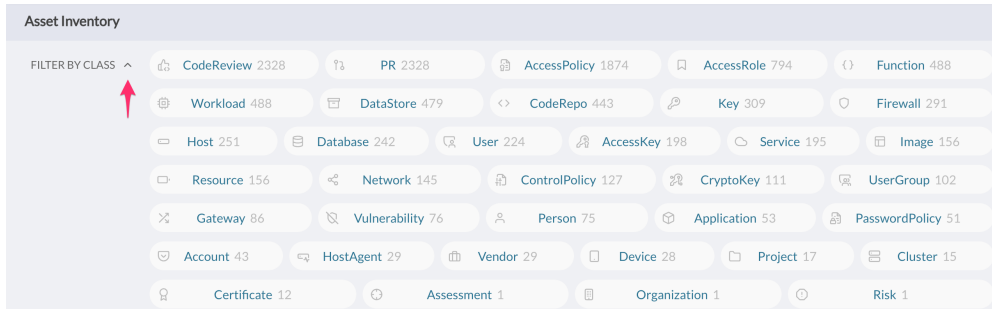
エンティティのクラスは、セキュリティ運用の概念においてエンティティを定義する抽象ラベルです。詳細については、データモデルのドキュメントを参照してください。

1 つ以上のクラスを選択した後、エンティティ/アセットを [タイプ (Type)] ごとにさらにフィルタリングできます。

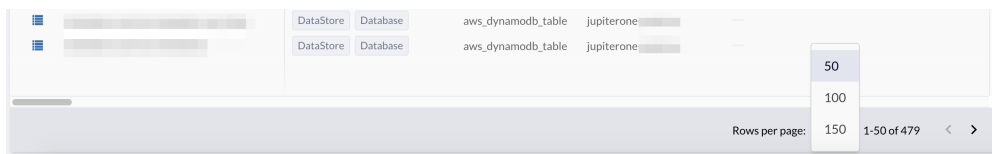


i エンティティのタイプは、そのソースによって定義される特定のタイプのエンティティを表します。詳細については、データモデルのドキュメントを参照してください。

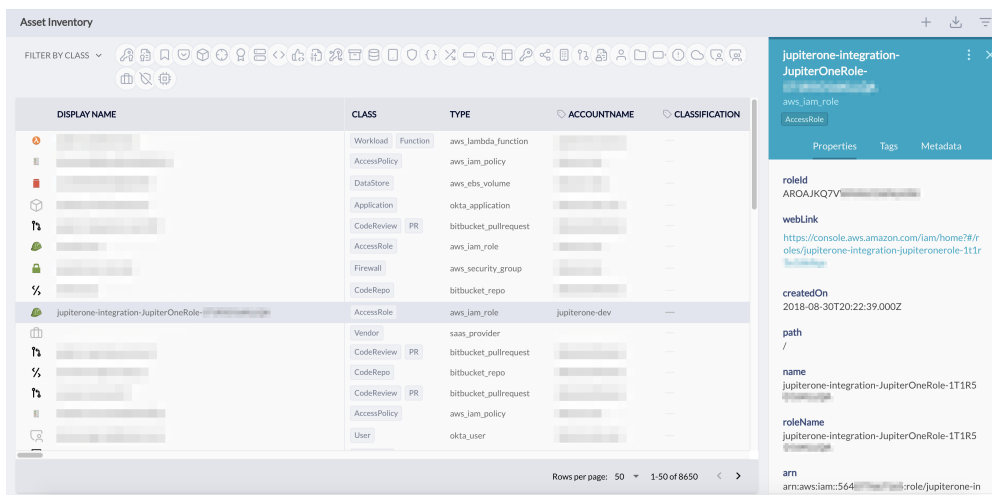
[クラス(Class)] フィルタを展開して、各クラスのカウンを含む、ダッシュボードに似たエンティティ/アセットの詳細ビューを表示することもできます。



データは、クイックフィルタの下での表での選択に対応します。表の下部にあるページネーションコントロールに注意してください。

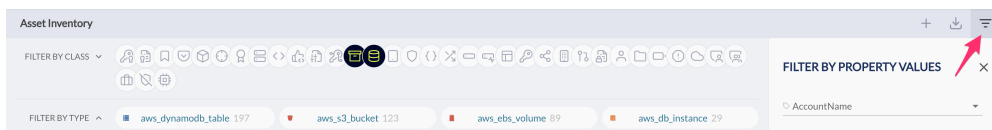


テーブルでエンティティを選択すると、右側のサイドパネルに詳細なプロパティが表示されます。

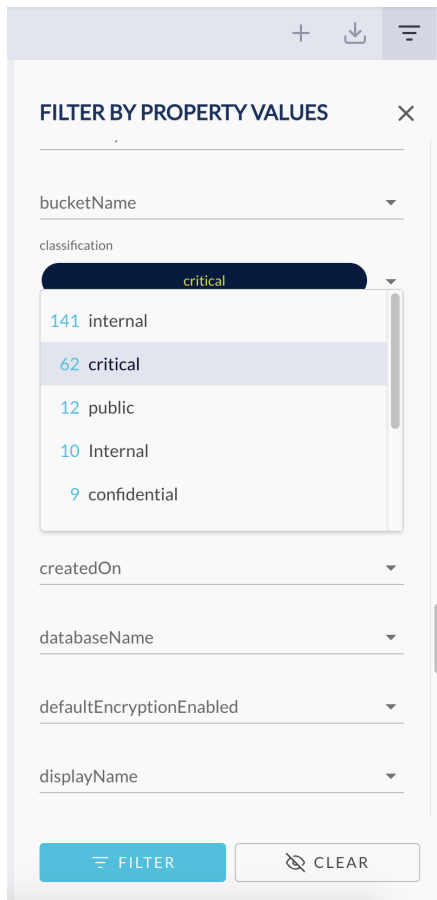


プロパティ別の詳細フィルタ

特定のプロパティ値を選択することで、詳細なフィルタ処理を適用できます。右上隅にあるコントロールアイコンを使用して、[フィルタ(Filter)] パネルを開きます。



フィルタを適用するプロパティを検索し、フィルタを適用する 1 つまたは複数の値を選択します。[プロパティ(Property)]ドロップダウンボックスで以前に選択した値をクリックすると、選択が解除されます。



i ヒント: より詳細なプロパティフィルタを適用する前に、まずクイックフィルタを使用してクラス/タイプを選択することをお勧めします。これによりプロパティ/値の数が減り、選択しやすくなります。

アラートの設定

Secure Cloud Insights では、継続的な監査と脅威の監視のために、任意の J1QL クエリを使用してアラートルールを設定できます。これは Alerts アプリで行います。

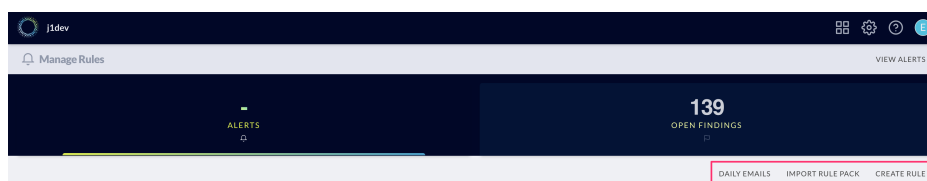
ルールパックからのアラートルールのインポート

アラートをトリガーするには、少なくとも 1 つのアクティブなアラートルールが必要です。複数のルールを追加する最も簡単な方法は、次の手順に従ってルールパックをインポートすることです。

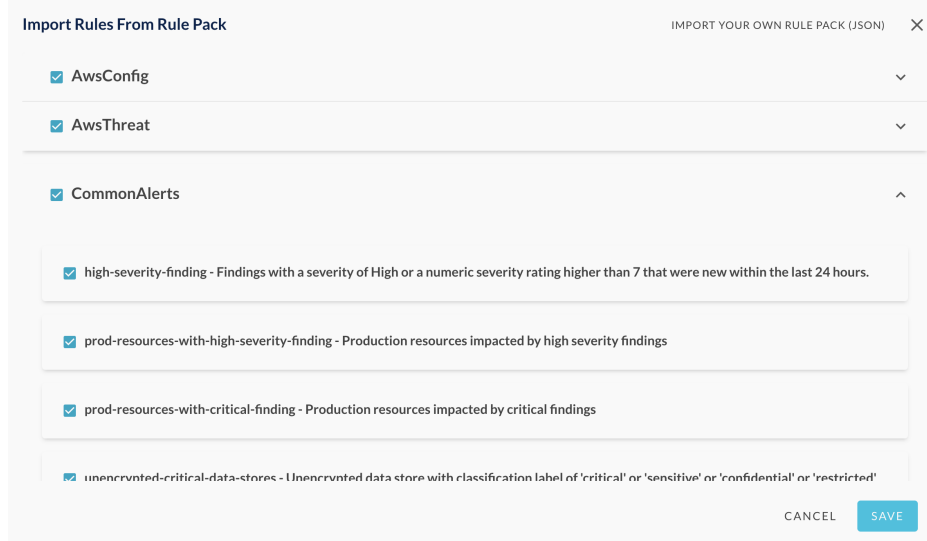
1. Alerts アプリから [ルールの管理 (Manage Rules)] に移動します。



2. [ルールパックのインポート (Import Rule Pack)] アクションボタンをクリックします。



3. すると [ルールパックからルールをインポート (Import Rules from Rule Pack)] モーダルウィンドウが表示されるので、ルールパックまたはルールパック内の個々のルールを選択できます。[保存 (Save)] をクリックして、選択したルールをインポートします。



カスタムアラートルールの作成

独自のカスタムアラートルールを作成するには、次の手順を実行します。

1. Alerts アプリから [ルール管理 (Manage Rules)] に移動します。
2. [ルールの作成 (Create Rule)] アクションボタンをクリックして、モーダルウィンドウを表示します。
3. カスタムルールに次の詳細情報を入力し、[保存 (Save)] をクリックします。
 - [名前 (Name)]
 - 説明
 - [シビラティ (重大度) (Severity)] (ドロップダウンリストから選択)
 - [クエリ (Query)] (任意の J1QL クエリ)

カスタムルールが追加され、毎日、毎時、あるいはエンタープライズ顧客の場合はストリーミング評価によって評価されます。ルールで指定したクエリが少なくとも1つの一致を返す場合、アラートがトリガーされます。

追加のアラートオプションの設定

アラートからワークフローをトリガーする機能があります。利用したいオプションのチェックボックスをオンにし、ドロップダウンやフィールドから必要な情報を入力します。

一部のアラートオプションには、追加の統合/権限が要求されます。

1. Slack: [次の手順](#)に従って、Secure Cloud Insights Slack との統合を設定する必要があります。必ず #channel 形式でチャンネルを指定してください。
2. JIRA: [次の手順](#)に従って、Secure Cloud Insights JIRA との統合を設定する必要があります。
3. SNS: 送信先の AWS アカウントは AWS 統合として設定されている必要があり、発行先の AWS アカウントの IAM ロールには SNS:Publish 権限が必要です Secure Cloud Insights
4. SQS: 送信先の AWS アカウントは AWS 統合として設定されている必要があり、発行先の AWS アカウントの IAM ロールには SQS:SendMessage 権限が必要です Secure Cloud Insights

アラートの管理

アラートルールの評価はデフォルトで毎日行われ、または特定のルールで指定したカスタム間隔（毎時または 30 分ごと）で評価されます。

アクティブなアラートがアラートルールの評価基準に一致すると、Alerts アプリの次のようなデータグリッドに表示されます。

TYPE	SEVERITY	ALERT TITLE / MESSAGE	COUNT	LAST ALERTED ON	
Alert	HIGH	s3-bucket-replication-enabled S3 buckets should enable cross-region replication.		05/3/19 7:01 am	DISMISS
Alert	HIGH	config-rule-noncompliant AWS Config rule evaluation found non-compliant resource configurations.		05/3/19 7:01 am	DISMISS
Alert	INFO	acm-certificate-expiration-check ACM certificate set to expire within 30 days.		05/3/19 6:58 am	DISMISS
Alert	HIGH	s3-bucket-versioning-enabled S3 buckets should enable versioning.		05/3/19 6:55 am	DISMISS
Alert	MEDIUM	unclassified-data-stores Data stores without a classification tag assigned		05/3/19 6:48 am	DISMISS
Alert	HIGH	s3-bucket-replication-enabled S3 buckets should enable cross-region replication.		05/2/19 7:01 am	DISMISS

- 個々のアラート行をクリックして展開すると、アラートの詳細が表示されます。
- 警告を閉じるには、[却下 (DISMISS)] ボタンをクリックします。

アラートが却下されないと、クエリ結果に変更がない限り、フォローアップアラート通知は送信されません。

日次通知 Eメールの設定

新しい/アクティブなアラートの通知を毎日受け取るには、次を選択します。

- Manage Rules
- [日次メール (Daily Emails)]
- [受信者 (Recipients)] フィールドにユーザーまたはチームの電子メールアドレスを入力します。

Visual Query Builder の使用

Visual Query Builder (VQB) は、JupiterOne Query Language (J1QL) 構文を学ばなくてもクエリを作成することができる、コード不要でドラッグアンドドロップで操作できるビジュアルインターフェイスを提供します。Secure Cloud Insights Secure Cloud Insights

権限

VQB を使用するには、accessLanding および readGraph 以上のロールが必要です。

前提条件


組織はデータを Secure Cloud Insights にインポートするためにすでに統合を使用している必要があります。

VQB を使用したクエリの作成

VQB にアクセスするには、Ask Anything 検索バーから  をクリックします。

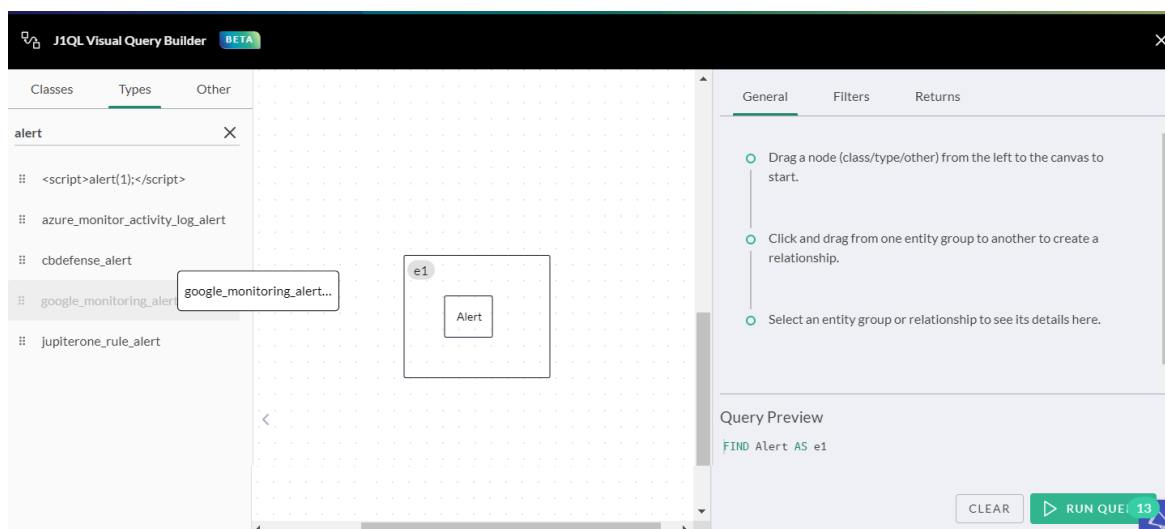
VQB ワークスペースの構成を次に示します。

- 左ペインには以下が含まれます。
 - デフォルトのアセットクラス
 - J1QL 固有のデータモデルアセットタイプ
 - テキスト検索や * ワイルドカードを含むその他のエンティティ

 同じクエリでテキスト検索アセットとワイルドカードアセットを使用することはできません。

- 中央キャンバスペイン
- 右側の情報ペインには、クエリプレビューウィンドウなど、中央キャンバスの内容に基づいたコンテキスト情報が含まれます。

1. アセットを左ペインから中央キャンバスにドラッグして開始します。



中央キャンバスにアイテムを配置すると、右側のペインの [クエリプレビュー (Query Preview)] ウィンドウでクエリの作成が開始されます。

クエリを使用してカスタムテキスト文字列でグラフ全体の全文検索を実行する場合は、左側のペインで [その他 (Other)] をクリックし、テキストアセット上にドラッグします。カスタムテキストを入力するように求められます。カスタムテキスト文字列はルートアセットである必要があるため、この手順はクエリの作成の最初のみ実行できます。

2. 関係を構築するすべてのクラスアセットとタイプアセットをドラッグします。

- アセットをグループ化する場合は、アセットを互いの上にドラッグしてグループを形成します。
- 各アセットまたはアセットグループの左上隅に、エイリアス ID があります。
- アセットグループをクリックすると、右側の情報ペインに詳細が表示されます。
- 最初にドラッグしたアセット、または最初に作成したアセットグループが、[その他 (Other)] メニューのテキストアセットでない限り、ルートアセットまたはアセットグループになります。

3. あるアセットまたはアセットグループボックスをクリックして別のアセットまたはアセットグループボックスにドラッグして、関係を作成します。

- 各関係には識別子があります。
- 関係識別子をクリックすると、右側の情報ペインにその詳細が表示されます。
- 関係クラスは、デフォルトで使用可能なすべての動詞に対して設定されます。J1QL は、その関係に適用できるすべての動詞を示します。関係クラスを切り替えて、クエリの検索方法を決定できます。

i 動詞 RELATES TO は、あらゆる関係動詞をカバーします。ただし、クエリのパフォーマンスを向上させるには、特定の関係動詞を使用することを強くお勧めします。

The screenshot shows the J1QL Visual Query Builder interface. On the left, a sidebar lists classes under the 'slack' group: slack_channel, slack_team, and slack_user. The main canvas displays a query diagram with two entity groups: e1 (containing User and slack_user) and e3 (containing Account and slack_team). A relationship r1 is being established between e1 and e3. On the right, the configuration panel shows relationship classes defaulting to all available verbs, with 'HAS' and 'MANAGES' selected. The Query Preview shows the following query:

```
FIND (User | slack_user) AS e1
  THAT (HAS | MANAGES) AS r1 (Account |
    slack_team) AS e3
```

4. 引き続きアセットをドラッグし、グループとそれらの間の関係を作成します。

ワイルドカードの使用

任意の関係を表すために使用できるワイルドカードアセットがあります。*(任意のエンティティ)をクリックして、キャンバスにドラッグします。次に、右側のペインで、クエリに含めない関係を選択解除します。ワイルドカードアセットを使用すると、そのワイルドカードアセットがメンバーになっているアセットグループ内のすべての子アセットがオーバーライドされます。

たとえば、次の例では、テキストアセット「Security」がルートアセットです。クエリによって、いずれかのプロパティに「security」という単語を使用するすべてのアラートとすべてのアラートアセットに対して、全文検索を実行します。Secure Cloud Insightsワイルドカードアセットには、考えられるすべての関係がリストされています。検索する情報に応じて、右側のペインにリストされている選択肢のオンとオフを切り替えることができます。

The screenshot shows the J1QL Visual Query Builder interface. On the left, a sidebar lists classes under the 'Other' group: Text Node. The main canvas displays a query diagram with three entity groups: e4 (containing *(any entity)), e1 (containing *Securi...), and e2 (containing Alert and jupiterone_rule_alert). Relationships r1 and r2 are shown between e1 and e2. On the right, the configuration panel shows relationship classes defaulting to all available verbs, with 'HAS', 'ALLOWS', and 'DENIES' selected. The Query Preview shows the following query:

```
FIND "Security" AS e1
  THAT RELATES TO AS r1 (Alert | jupiterone_rule_alert)
    AS e2
```


i 注: アセットまたはアセットグループ間の関係を接続するときは、ルートアセット(e1 など)から始めて、それを関連するアセット(e4 など)にドラッグしてから、必要に応じて他のアセットを接続する必要があります。

フィルタリング

キャンバスで選択したアセットのすべてのプロパティに基づいたフィルタを作成できます。フィルタには AND OR 条件を適用することもできます。

General Filters Returns

Filters for the **e1** group of entities

Property name Operator

Attribute value OR

e1.level = warning

OR e1._createdOn <= undefined

AND

Query view

```
FIND Alert
WITH ( level = "warning"
OR _createdOn <= undefined ) AS e1
```

セキュリティ調査結果の管理

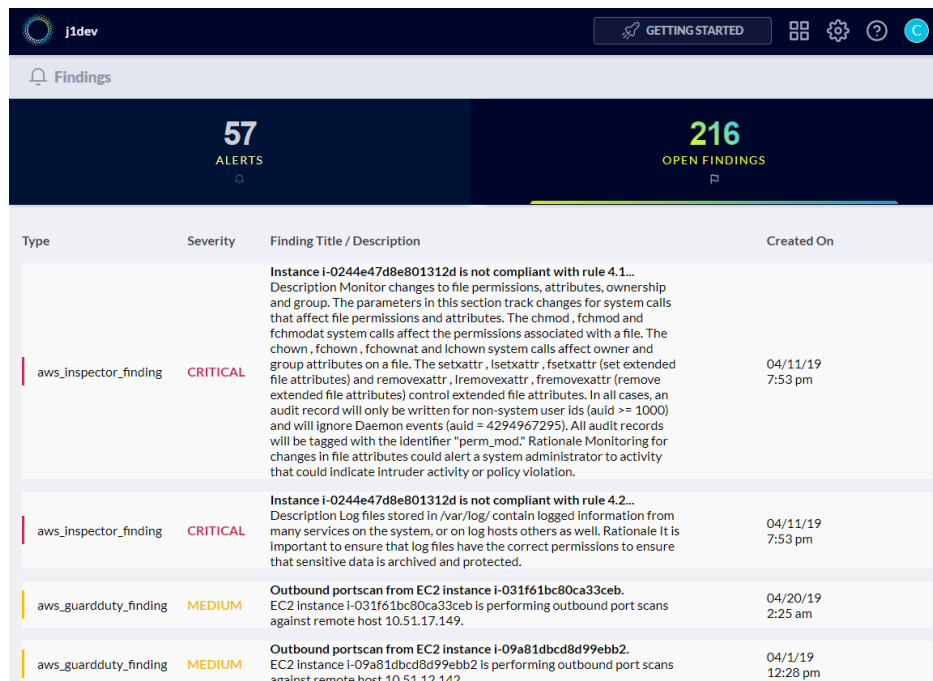
Secure Cloud Insights では、一元化されたリポジトリとダッシュボードが用意されているため、次のようなさまざまなソースからのセキュリティに関する調査結果を簡単に管理できます。

- AWS Inspector の調査結果
- AWS GuardDuty の調査結果
- Veracode の静的分析と動的分析の調査結果
- WhiteHat アプリケーションのセキュリティに関する調査結果
- Tenable Cloud スキャンの調査結果
- HackerOne レポートの調査結果
- CVE およびその他の脆弱性の調査結果
- 手動侵入テストの調査結果 (API 経由でインポート)

i 脆弱性スキャナの統合がさらに追加されています。現在のロードマップには、Rapid7、Qualys、Bugcrowd、White Source、Source Clear、および Snyk が含まれます。

結果の表示

調査結果のまとめは、Alerts アプリの [調査結果 (Findings)] タブでアクセスできます。ヘッダータブには、現在開いている調査結果の合計数が表示されます。選択すると、詳細な調査結果ビューが表示されます。



Type	Severity	Finding Title / Description	Created On
aws_inspector_finding	CRITICAL	Instance i-0244e47d8e801312d is not compliant with rule 4.1... Description Monitor changes to file permissions, attributes, ownership and group. The parameters in this section track changes for system calls that affect file permissions and attributes. The chmod, fchmod and fchmodat system calls affect the permissions associated with a file. The chown, fchown, fchownat and lchown system calls affect owner and group attributes on a file. The setxattr, lsetxattr, fsetxattr (set extended file attributes) and removexattr, lremovexattr, fremovexattr (remove extended file attributes) control extended file attributes. In all cases, an audit record will only be written for non-system user ids (audit >= 1000) and will ignore Daemon events (audit = 4294967295). All audit records will be tagged with the identifier "perm_mod." Rationale Monitoring for changes in file attributes could alert a system administrator to activity that could indicate intruder activity or policy violation.	04/11/19 7:53 pm
aws_inspector_finding	CRITICAL	Instance i-0244e47d8e801312d is not compliant with rule 4.2... Description Log files stored in /var/log/ contain logged information from many services on the system, or on log hosts others as well. Rationale It is important to ensure that log files have the correct permissions to ensure that sensitive data is archived and protected.	04/11/19 7:53 pm
aws_guardduty_finding	MEDIUM	Outbound portscan from EC2 instance i-031f61bc80ca33ceb. EC2 instance i-031f61bc80ca33ceb is performing outbound port scans against remote host 10.51.17.149.	04/20/19 2:25 am
aws_guardduty_finding	MEDIUM	Outbound portscan from EC2 instance i-09a81dbcd8d99ebb2. EC2 instance i-09a81dbcd8d99ebb2 is performing outbound port scans against remote host 10.51.12.142.	04/11/19 12:28 pm

i Secure Cloud Insights は、調査結果のソースから入手できる属性に基づいて、各調査結果の影響を受けるリソースや各調査結果に関連するリソースを自動的にマッピングします。

リストから調査結果を選択すると、これらの関係を示すグラフが表示されます。これにより、コンテキストを視覚化して、調査結果の影響をさらに分析し、修復のためのアクションコースを決定することができます。

TYPE	SEVERITY	FINDING TITLE / DESCRIPTION	CREATED ON
aws_inspector_finding	CRITICAL	Instance I-0244e47d8e801312d is not compliant with rule 5.2... Description: The PermitUserEnvironment option allows users to present environment options to the ssh daemon. Rationale: Permitting users the ability to set environment variables through the SSH daemon could potentially allow users to bypass security controls (e.g. setting an execution path that has ssh executing trojan/d programs)	04/11/19 9:54 pm
QUERY Find Finding with _key="z0Rksu00(LuLb5f6R55HTMB0V2n4zsqhvR8sv8uQLE=" that relates to * return tree			
<pre> graph TD Inspector((Inspector)) -- HAS --> Instance[Instance I-0244e47d8e801312d is not compliant with rule 5.2...] Instance -- HAS --> Wazu[wazu-elastic-server] Wazu -- HAS --> Run[Run - Assessment-Template-Default-All-Rules - 2019-04-12T00:52:20.495Z] </pre>			
On instance I-0244e47d8e801312d, TCP port 9200 which is associat... On this instance, TCP port 9200, which is associated with Elasticsearch, is reachable from the internet with no process listening. The instance I-0244e47d8e801312d is located in VPC vpc-43fe713a and has an attached ENI eni-0157f58c02f051db which uses network ACL acl-9aah99c7. The port is reachable from the internet through Security Group sg-00820034cb9961b2 and IGW igw-baf-80dc			
aws_inspector_finding	INFO		04/11/19 9:54 pm

調査結果に対するアラートを作成する

J1QL を使用してフィルタリングと相関を行い、特定の結果を通知するカスタムアラートルールを作成できます。

例:

Common Alerts ルールパックには、次の 3 つのルールが含まれています。

- シビラティ(重大度)の高い調査結果

過去 24 時間以内に新たに検出された、シビラティ(重大度)が「高」または 7 を超える調査結果に対するアラート。

```
Find Finding with
(severity='High' or severity='high' or numericSeverity>7) and
_createdOn > date.now-24hours
```

- prod-resources-with-high-severity-finding

シビラティ(重大度)の高い調査結果によって本番リソースが影響を受ける場合にアラートを出します。

```
Find (Host|DataStore|Application|CodeRepo|Account|Service|Network)
with tag.Production=true
that has Finding with severity=('High' or 'high') or numericSeverity=(7 or 8)
```

- **prod-resources-with-critical-finding**

クリティカルな調査結果によって本番リソースが影響を受ける場合にアラートを出します。

```
Find (Host|DataStore|Application|CodeRepo|Account|Service|Network)
  with tag.Production=true
  that has Finding with severity=('Critical' or 'critical') or numericSeverity=(9 or
  10)
```

AWS Treat ルールパックには、次のルールが含まれています。

- **aws-guardduty-inspector-finding-instance-correlation**

脆弱な EC2 インスタンス (中または高レート of オープンなインスペクタの調査結果) を識別します。これは疑わしいアクティビティのターゲットでもあります (中または高レート of オープンな GuardDuty の調査結果)。

```
Find aws_guardduty_finding with numericSeverity>5 and open=true as guardduty
  that relates to aws_instance as i
  that has aws_inspector_finding with numericSeverity>5 and open=true as inspector
  return i.*, guardduty.*, inspector.*
```

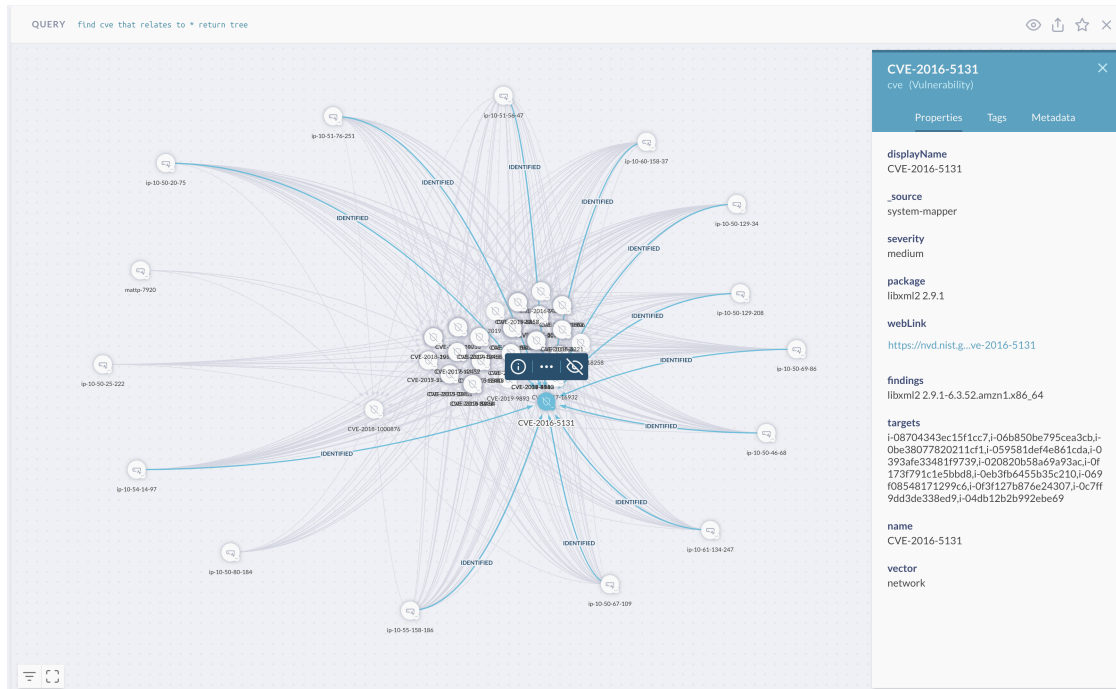
J1QL クエリとグラフを使用して調査結果のビューを生成

J1QL クエリを実行して、調査結果、それらを識別したエージェント/スキャナ/サービス、およびそれらが影響を与えるリソース間の関係を分析するのに役立つ視覚的なグラフを生成できます。

次に例を示します。

```
Find cve that relates to (Host|HostAgent) with active=true return tree
```

これにより、次のようなビジュアルが得られます(ノードを移動させて位置を調整する必要がある場合があります)。



Secure Cloud Insights のポリシーと手順の管理

Secure Cloud Insights は、ユーザーが会社のセキュリティポリシーと手順を生成および管理できるようにする **Policies** アプリを提供します。このアプリには次の機能があります。

- テンプレートからのポリシーと手順の生成
- Web アプリを介したオンラインでのポリシーと手順の管理
- コンプライアンス要件へのコントロール/手順のマッピング
- ポリシービルダー CLI

テンプレートからのポリシーと手順の生成

Policies アプリには、組織がセキュリティプログラムと運用をゼロから構築するのに役立つ、120 を超えるポリシーと手順のテンプレートセットが用意されています。これらのテンプレートは、当社独自の内部ポリシーと手順から派生したものであり、コンプライアンス評価を幾度も経た上で作成されています。

開始するには、**Policies** アプリに移動し、Web フォームの次の 3 つの情報セクションに入力します。

- 会社情報
- 主要な個人情報(セキュリティおよびプライバシー担当者など)
- セキュリティと DevOps ツールに関する情報

ポリシーおよび手順書を初めて生成する場合、数分かかる場合があります。

変数

Markdown テキストには、`{variableName}` の形式でグローバル変数とローカル変数の両方が含まれていることに注意してください。テンプレート内の変数は、関連するテキストに自動的に置き換えられるため、編集しないことをお勧めします。

手順書には、オプションのローカル `{{provider}}` 変数を含めることができます。これにより、その手順を実装する、またはその手順を実行する責任が任された制御プロバイダーを設定できます。たとえば、「シングルサインオン」のプロバイダーは、「Okta」、「OneLogin」、「JumpCloud」、「Google」などです。この `provider` 値は、ドキュメントエディターを開いたときに上部近くに表示されるドキュメントタイトルの下に入力できます。

手順エディターには、短く要約されたガイダンスの説明も表示されます。さらに、特定の手順を採用する準備ができていないかどうかに応じて、「採用済み」フラグのオンとオフを切り替えることができます。

バージョン管理

ポリシーと手順書を編集した場合、保存時に自動的にバージョン管理されます。

`{{defaultRevision}}` 変数には、ドキュメントが最後に編集された日付が入力されます。

現在、Web アプリには、以前のバージョンのドキュメントを表示するための UI がありません。

ポリシーと手順書のダウンロード/エクスポート

画面の右上隅にある [Zipのエクスポート/ダウンロード(Export / Download Zip)] ボタンをクリックすると、次の 3 つのファイルセットを含む zip ファイルが生成されます。

- Markdown 形式のテンプレート
- Markdown 形式の最終的なポリシーと手順
- HTML 形式の最終的なポリシーと手順

ポリシービルダー CLI

Secure Cloud Insights は、ポリシーと手順をオフラインで(たとえば git リポジトリのコードとして)管理し、必要に応じてアカウントに公開できるオフライン CLI を提供します。Secure Cloud Insights





独自の既存ポリシーを使用する

Policies アプリは、プラットフォームのオプションのコンポーネントです。それ以外のプラットフォームの前提条件ではありません。Compliance アプリは、コンプライアンス フレームワークの要件とコントロールへの適切なマッピングのために Policies アプリに依存する唯一のアプリです。

Secure Cloud Insights で提供されているポリシー/手順テンプレートを使用する必要はありません。組織にセキュリティポリシーと手順に関するドキュメントがすでにあり、コンプライアンスアプリとそのマッピング機能を利用したい場合は、既存のポリシーを変換して Secure Cloud Insights に公開することができます。

Secure Cloud Insightsアカウント/組織のユーザーの招待

アカウント/組織に他のユーザーを追加するには、簡単な招待プロセスを使用します。Secure Cloud Insights招待を送信するには、次の手順に従います。

1.  [設定 (Settings)] に移動し、 [ユーザーとアクセス (Users & Access)] を選択します。
2. モーダル画面がポップアップ表示され、現在のユーザーグループが表示されます。[グループの追加 (Add Group)] ボタンをクリックして新しいグループを追加し、新しいグループに名前を付け、オプションで説明を入力することができます。
3.  [ユーザーの追加 (Add User)] ボタンをクリックし、ユーザーの電子メールアドレスを入力して、 [招待の送信 (Send Invitation)] をクリックします。
4. アカウント/組織に参加するための招待メールがユーザーに送信され、ログイン時に招待を承諾するように求められます。新しいユーザーには Secure Cloud Insights ユーザーアカウントの作成が求められます。

- 他のユーザーを招待するには、**管理者** グループのメンバーである必要があります。
- **企業・官公庁** のお客様で SAML SSO を使用している場合は、[こちら](./configure-ss-integration.md)の手順を参照してください。

Secure Cloud Insights API の使用

次の情報は、Secure Cloud Insights API の使用を開始するために必要な基本事項の一部を提供します。

エンドポイント

Secure Cloud Insights プラットフォームは、多くのパブリック GraphQL エンドポイントを公開しています。ベース URL は `https://api.us.securecloudinsights.cisco.com` で、次のエンドポイントがあります。


- **クエリとグラフの操作**: `/graphql`
- **アラートとルールの操作**: `/rules/graphql`

認証

Secure Cloud Insights API は、認証にベアトークンを使用します。必ず、API キーをベアトークンとしてヘッダーに含め、JupiterOne-Account はヘッダーパラメーターとして含めてください。

API キー

API キーを生成するには、次の手順を実行します。

1. [設定 (Settings)] > [アカウント管理 (Account Management)] に移動します。
2. 左側のパネルで、 ([キー (Key)]) アイコンをクリックします。
3. [ユーザー API キー (User API Keys)] ページで、[追加 (Add)] をクリックします。
4. API キー モーダルで、キーの名前と有効期限が切れるまでの日数を入力し、[作成] をクリックします。
5. API キーは 1 回しか利用できないため、コピーまたは保存してから、[完了 (Done)] をクリックします。

アカウント ID (Account ID)

次の J1QL クエリを実行して、アカウントのアカウント ID (JupiterOne-Account 値) を見つけることができます。

```
FIND jupiterone_account as a return a._accountId
```

次に、認証付きの cURL コマンドの例を示します。

```
curl --location --request POST
'https://api.us.securecloudinsights.cisco.com/graphql' \
--header 'JupiterOne-Account: accountId' \
--header 'Authorization: Bearer 123456abcdef' \
--header 'Content-Type: application/json' \
--data-raw '{"query":...}'
```

サポートへの問い合わせ

テクニカルサポートが必要な場合は、次のいずれかを実行してください。

- 最寄りのシスコパートナーにご連絡ください。
- シスコサポートの連絡先
- Web でケースを開く場合：<http://www.cisco.com/c/en/us/support/index.html>
- 電子メールでケースを開く場合：tac@cisco.com
- 電話でサポートを受ける場合：800-553-2447(米国)
- ワールドワイド サポート番号：
<https://www.cisco.com/c/en/us/support/web/tsd-cisco-worldwide-contacts.html>

著作権情報

Cisco および Cisco ロゴは、シスコまたはその関連会社の米国およびその他の国における商標または登録商標です。シスコの商標の一覧については、URL: <https://www.cisco.com/go/trademarks> をご覧ください。記載されている第三者機関の商標は、それぞれの所有者に帰属します。「パートナー」という用語の使用はシスコと他社との間のパートナーシップ関係を意味するものではありません。(1721R)