

デバイスオンボーディング用の Secure Connector の展開および管理

- セキュアコネクタの概要 （1 ページ）

セキュアコネクタの概要

セキュアコネクタは、ネットワークまたはサポートされているクラウド環境で実行され、Security Cloud Control Firewall Management と管理対象リソースとの間にセキュアな通信経路を提供する、コネクタインスタンスです。セキュアコネクタは、Security Cloud Control Firewall Management がインターネットから直接到達できないデバイスを管理するのに役立ち、ロギングおよび分析ワークフローでのイベント転送をサポートします。

[セキュアコネクタ（Secure Connector）]ページを使用して、テナントのコネクタインスタンスを表示、展開、および管理します。

セキュアコネクタのタイプ

[セキュアコネクタ（Secure Connector）]ページには、次のコネクタタイプを含めることができます。

- セキュアデバイスコネクタ（SDC）：Security Cloud Control Firewall Management とサポートされている管理対象デバイスとの間にセキュアな通信パスを提供します。
- セキュアイベントコネクタ（SEC）：サポートされているデバイスからイベントを受信し、ロギングおよび分析のために、それらのイベントを Cisco Cloud に転送します。

Secure Device Connector について

Secure Device Connector（SDC）は、デバイスがインターネットから直接到達できない場合にシスコデバイスが Security Cloud Control Firewall Management と通信できるようにするインテリジェントプロキシです。デバイスログイン情報を使用してデバイスをオンボーディングする場合、ネットワークに SDC を展開して、デバイスと Security Cloud Control Firewall Management 間の通信をプロキシできます。デバイスがインターネットから直接到達可能である場合、SDC

を使用する代わりにデバイスの外部インターフェイスを介した直接通信を許可できます。SDC は、次の機能を実行します。

- 管理対象デバイスのコマンドとメッセージを Security Cloud Control Firewall Management でモニターします。
- Security Cloud Control Firewall Management の代わりにコマンドを実行します。
- 管理対象デバイスにメッセージをリレーします。
- デバイス応答を返します。

SDC と Security Cloud Control Firewall Management 間の通信では、TLS 1.3 および AES-128-GCM を使用した HTTPS が使用されます。オンボードされたデバイスとサービスのクレデンシャルは、ブラウザから SDC に暗号化され、AES-128-GCM を使用して保存時にも暗号化されます。SDC のみがこれらのログイン情報にアクセスできます。

ネットワーク管理者ロールを持つユーザーは、Secure Device Connector またはセキュアイベントコネクタ（SEC）を作成する必要があります。

これらのデバイスを SDC を介して Secure Device Connector にオンボーディングできます。

- ログイン情報メソッドを使用した オンプレミス Firewall Management Center
- Meraki MX デバイス
- 汎用 SSH デバイス
- Cisco IOS デバイス

クラウド提供型 Firewall Management Center によって管理される Cisco Secure Firewall Threat Defense デバイスは SDC を必要とせず、プロキシを介したオンボーディングをサポートしません。これらのデバイスが クラウド提供型 Firewall Management Center に接続できるように、適切な DNS 設定とアウトバウンド インターネット接続があることを確認してください。

SDC と Security Cloud Control 間の通信を許可する方法については、直接クラウド接続のインバウンドアクセスを許可 （4 ページ）を参照してください。

詳細については、Secure Device Connector および Secure Event Connector を実行するための VM の展開 （5 ページ）を参照してください。

デバイス接続の計画

Security Cloud Control Firewall Management はクラウドコネクタを介して、または SDC を介して管理対象デバイスに接続します。

表 *1:*

接続方法	次の場合に使用	必要なネットワークアクセス
ダイレクトクラウドコネクタ	デバイスはインターネットから直接到達可能です。	ポート443、またはデバイス管理に使用するポートで、クラウドリージョンの Security Cloud Control Firewall Management IPアドレスからのインバウンドアクセスを許可します。
SDC	デバイスがインターネットから直接到達できないか、送信元にオンプレミス SDC が明示的に必要です。	ポート 443 またはデバイス管理に使用するポートで、SDC ホストからのフルインバウンドアクセスを許可します。SDC VM がデバイス管理インターフェイスに到達できることを確認します。

FDM による管理 デバイスは、インターネットから直接アクセスできるかどうかに関係なく、デバイスのログイン情報、登録キー、またはシリアル番号を使用して Security Cloud Control Firewall Management へのオンボーディングを実行できます。デバイスがインターネットに直接アクセスできないものの、インターネットに直接アクセスできるネットワーク上に存在する場合、このデバイスの一部として提供される Security Services Exchange コネクタは Security Services Exchange クラウドに到達できるため、FDM による管理 デバイスのオンボーディングが可能になります。

送信元は、以下をオンボーディングするにはオンプレミス SDC が必要であると明示的に記載しています。

- クラウドからアクセスできない ASA デバイス
- ログイン情報のオンボード方式を使用する場合にクラウドからアクセスできない FDM による管理 デバイス
- Cisco IOS デバイス
- SSH でアクセスするデバイス

Security Cloud Control Firewall Management はクラウドコネクタを使用して接続するため、他のすべてのデバイスとサービスにはオンプレミス SDC は必要ありません。

SDC に関する Docker ネットワークの考慮事項

SDC は Docker コンテナとして実行されます。Docker は通常、デフォルトのブリッジネットワークである docker0 に `172.17.0.0/16` サブネットを使用します。

`172.17.0.0/16`または Docker によって割り当てられたネットワークスペースの一部が既存のネットワーク IP スペースと重複している場合は、SDC を展開する前に Docker ネットワークのデ

フォルト値を変更します。代替ネットワークを使用するには、/etc/docker/daemon.json の設定を編集します。

たとえば、Docker ネットワークアドレスプールでスーパーネット `10.198.0.1/24` を使用し、サブネット `10.198.0.0/24` から、IP アドレス `10.198.128.0/17` をデフォルト Docker ブリッジインターフェイスである docker0 として使用します。

```
sudo vi /etc/docker/daemon.json

{
  "log-driver": "json-file",
  "log-opts": {
    "max-size": "100m",
    "max-file": "5"
  },
    "bip": "10.198.0.1/24",
    "default-address-pools": [
        {
            "base": "10.198.128.0/17",
            "size": 24
        }
    ]
}

(save and exit)

sudo reboot (required)
```

詳細については、「ネットワークの概要」を参照してください。

直接クラウド接続のインバウンドアクセスを許可

Security Cloud Control は、クラウドコネクタまたは Secure Device Connector（SDC）を介して管理対象デバイスに接続します。

クラウドコネクタを介して Security Cloud Control Firewall Management をデバイスに直接接続する場合、EMEA、米国、または APJ 地域のさまざまな IP アドレスに、ポート 443（またはデバイス管理用に設定したポート）でのインバウンドアクセスを許可する必要があります。

Table 2:

地域	URL	次からのインバウンドアクセスを許可
アジア太平洋日本（APJ）	https://security.cisco.com	54.199.195.111 52.199.243.0
オーストラリア（AUS）	https://security.cisco.com	13.55.73.159 13.238.226.118
ヨーロッパ、中東、アフリカ（EMEA）	https://security.cisco.com	35.157.12.126 35.157.12.15
インド（IN）	https://security.cisco.com	35.154.115.175 13.201.213.99

地域	URL	次からのインバウンドアクセスを許可
アメリカ合衆国（US）	https://security.cisco.com	52.34.234.2 52.36.70.147

SDC キャパシティとホストの要件を計画

各 Security Cloud Control Firewall Management 組織は SDC の数に制限はありません。SDC は 1 つの組織にのみ属しており、組織間で共有されることはありません。展開プランでは、1 つの SDC は約 500 台のデバイスをサポートすると予想されます。実際のキャパシティは、該当デバイスで有効になっている機能と、構成ファイルのサイズによって異なります。

複数の SDC を展開すると、次のことが可能になります。

- パフォーマンスを低下させることなく、デバイス管理を拡張できます。
- SDC を隔離されたネットワークセグメントに配置し、デバイスを同じ Security Cloud Control Firewall Management 組織内に配置します。
- 追加の SDC ごとにブートストラップ手順に従い、ホスト上で複数の SDC を実行します。

最初の SDC の名前にはテナント名と数値 1 が含まれています。SDC が追加されるたびに、シーケンスが番号付けされます。

Secure Device Connector および Secure Event Connector を実行するための VM の展開

デバイスのログイン情報を使用して Security Cloud Control をデバイスに接続する場合、Security Cloud Control とデバイス間の通信を管理するために、ネットワークに SDC をダウンロードして展開することがベストプラクティスです。通常、これらのデバイスは、非境界ベースでパブリック IP アドレスを持たないか、外部インターフェイスに開かれたポートを持っています。

SDC は、管理対象デバイスで実行する必要があるコマンドと、管理対象デバイスに送信する必要があるメッセージについて、Security Cloud Control を監視します。SDC は、Security Cloud Control に代わってこのコマンドを実行し、管理対象デバイスに代わって Security Cloud Control にメッセージを送信し、管理対象デバイスからの応答を Security Cloud Control に返します。

1 つの SDC が管理できるデバイスの数は、それらのデバイスに実装されている機能と、構成ファイルのサイズによって異なります。ただし、展開を計画するために、1 つの SDC で約 500 台のデバイスをサポートすることを想定しています。詳細については、「単一の Security Cloud Control テナントで複数の SDC を使用する」を参照してください。

この手順では、Security Cloud Control の VM イメージを使用してネットワークに SDC をインストールする方法について説明します。これは、SDC を作成するために推奨される、最も信頼できる方法です。

はじめる前に

- Security Cloud Control は、厳密な証明書チェックを必要とし、Secure Device Connector（SDC）とインターネットの間の Web またはコンテンツプロキシ検査をサポートしていません。

プロキシサーバーを使用している場合は、SDC と Security Cloud Control の間のトラフィックの検査を無効にします。

- SDC には、TCP ポート 443 またはデバイス管理用に設定したポートでのインターネットへの完全なアウトバウンドアクセスが必要です。
- Security Cloud Control によって管理されるデバイスは、SDC VM の IP アドレスからのインバウンドトラフィックを許可する必要があります。
- 適切なネットワークアクセスを確保するため、「直接クラウド接続のインバウンドアクセスを許可 （4 ページ） の Secure Device Connector への接続」を参照してください。
- ネットワークでプロキシを使用している場合は、ホストセットアップコマンドを実行する前に、必要なすべての詳細情報があることを確認します。ほとんどの問題は、誤ったプロキシ設定に関連しています。重要な詳細情報は次のとおりです。
  - プロキシの IP/ホスト名。
  - プロキシが、トラフィックを代行受信し、独自の証明書を使用してそれを再暗号化するかどうか。この詳細情報が、SDC VM セットアップに関する複雑さのほとんどの原因です。
    - プロキシがトラフィックを代行受信する場合は、VM を設定するときにルート証明書を準備します。プロンプトが表示されたら、それを貼り付けることで、プロキシによって生成された証明書をホストと SDC が認識して信頼できるようになります。
    - プロキシがトラフィックを代行受信しない場合、ここでは他に何もする必要はありません。
  - 以下の項目は、ほとんどの場合、プロキシされる HTTP および HTTPS 接続で同じです。ただし、プロトコルごとに異なるプロキシを使用する場合は、それぞれに次のすべてが必要になります。
    - プロキシの IPアドレス
    - プロキシが使用するポート
    - プロキシが、プロキシ自体への接続が HTTPS を介したものである必要があるかどうか（通常はそうではありません）。たとえば、プロキシのアドレスが https://proxy.corp.com:80 と表示された場合は、「はい」と応答します。表示されるアドレスが http://proxy.corp.com:80 の場合は、「いいえ」と応答します。どちらの URL もポート 80 を使用しますが、プロトコルが異なることに注意してください。
    - 以下を含むプロキシの認証の詳細情報：
      - プロキシに認証が必要かどうか（ほとんどの場合、必要ありません）
      - 必要な場合は、ホストを設定するときに使用できるユーザー名とパスワードが必要です。

サポートされているインストール

- Security Cloud Control は、vSphere Web クライアントまたは ESXi Web クライアントを使用した SDC VM OVF イメージのインストールをサポートしています。
- Security Cloud Control は、vSphere デスクトップクライアントを使用した SDC VM OVF イメージのインストールをサポートしていません。
- Security Cloud Control は、独自の Ubuntu インスタンスへの SDC のインストールをサポートしています。現在、バージョン 20LTS ～ 24LTS がサポートされています。
- ESXi 5.1 ハイパーバイザ。

システム要件

- 1 つの SDC を持つ VM のシステム要件は、次のとおりです。
  - 2 vCPU
  - 2 GB のメモリ
  - 64 GB のディスク容量
- ホストに追加する各 SDC には、追加の 1 つの vCPU と 1 GB の RAM が必要です。
- 1 つの SEC（Cisco Security Analytics and Logging で使用されるコンポーネント）を持つ VM のシステム要件は、次のとおりです。
  - 4 つの vCPU（最小）
  - 8 GB のメモリ
- ホストに追加する SEC ごとにリソースを倍にする必要があります。そのため、これらは 1 つの SDC と 1 つの SEC を持つ VMware ESXi ホストの要件です。
  - 6 vCPU
  - 10 GB のメモリ
  - 64 GB のディスク容量

設置の準備

- ホストにおいてネットワーキングを手動で設定するには、次の情報を収集します。
  - VM に使用する静的 IP アドレス
  - `cdo` ユーザー（または sudo アクセスを持つユーザー）と sdc ユーザー（Docker を実行するユーザー）に使用するパスワード
  - 組織で使用する DNS サーバーの IP アドレス
  - SDC アドレスが存在するネットワークのゲートウェイ IP アドレス

- タイム/NTP サーバーの FQDN または IP アドレス

- SDC 仮想マシンは、セキュリティパッチを定期的にインストールするように設定されており、これを行うには、ポート 80 のアウトバウンドを開く必要があります。

  ネットワークでアウトバウンド接続に許可/拒否リストを使用している場合は、ubuntu.com への接続を許可して、それらのセキュリティ更新を適用できるようにする必要があります。



（注） Ubuntu はチェックサムで更新を保護し、HTTPS ではなく HTTP のみを使用します。セキュリティ更新をプルするには、ubuntu.com への HTTP 接続を許可する必要があります。

VM の展開

SDC と SEC の実行に使用される VM を展開するには、2 つのオプションがあります。

1. Security Cloud Control が提供する VMware イメージをダウンロードするには、次の手順を実行します。
2. Ubuntu 20、22、または 24 を自分で展開する場合、独自の Ubuntu インスタンスを展開するときは、次のセクションをスキップして、「VM の設定」セクションに進むことができます。

手順

1. Security Cloud Control のホームページで、[ファイアウォール（Firewall）] をクリックします。
2. 左側のペインで **管理（Administration）** > **統合（Integrations）** > **セキュアコネクタ（Secure Connectors）** をクリックします。
3. [Services] ページの [Secure Connectors] タブを選択し、青いプラスボタンをクリックして、[Secure Device Connector] を選択します。
4. **[Download the SDC VM]** のイメージをクリックします。新しいタブで表示されます。
5. .zip ファイルからすべてのファイルを抽出します。これらは次のようなものです。
   - CDO-SDC-VM-ddd50fa.ovf
   - CDO-SDC-VM-ddd50fa.mf
   - CDO-SDC-VM-ddd50fa-disk1.vmdk
6. vSphere Web クライアントを使用して、管理者として VMware サーバーにログインします。

（注） ESXi Web クライアントは使用しないでください。

プロンプトに従って、OVF テンプレートから Secure Device Connector 仮想マシンを展開します。

7. セットアップが完了したら、SDC VM の電源を入れます。
8. 新しい SDC VM のコンソールを開きます。
9. `cdo` というユーザー名でログインします。デフォルトのパスワードは `adm123` です。

VM の設定

これで、展開した VM イメージのコンソールを起動できます（また、VM を自作しており、SSH を有効にしている場合は、その VM に SSH 接続できます）。ホストで SDC または SEC Docker コンテナを実行できるようにするには、設定スクリプトを実行する必要があります。

1. Security Cloud Control 提供の VM をダウンロードした場合は、CLI がすでにインストールされているため、手順 2 に進むことができます。独自の VM を展開した場合は、その VM に SSH 接続し、次のコマンドを実行して CLI をインストールします。

   ```
   curl -O
   https://s3.us-west-2.amazonaws.com/download.defenseorchestrator.com/sdc-cli/sdc-cli-package-latest.tgz
    && tar -xvf sdc-cli-package-latest.tgz && chmod +x ./install.sh && ./install.sh
   ```

2. 次のコマンドを実行して、ホストの構成を開始します：

   ```
   sudo sdc host configure
   ```

3. パスワードの入力を求められたら、Security Cloud Control 提供の VM 場合は adm123 を入力し、独自の VM の場合は、選択した管理者パスワードを入力します。
4. プロンプトに従って、`sdc` ユーザーを設定します。
5. ネットワークの設定を求められたら、次のいずれかを選択します。
   - [このホストを静的 IP で手動設定する（Manually configure this host with a static IP）]：このホストの IP、ゲートウェイ、DNS サーバーなどを指定し、それを VM 上のシステム設定に書き込みます。
   - [DHCP]：VM に静的 IP を割り当てる DHCP サーバーがある場合。
   - [静的 IP が設定済みで、今すぐネットワーキングを変更したくない。（Static IP is already configured and I don't want to change my networking now.）]
6. プロンプトが表示されたら、プロキシ設定に関する質問に回答します。このトピックの上部にある詳細なリストで、すべての前提条件と、可能なプロキシ設定オプションを確認してください。

7. プロキシが設定済みである場合は、すべてのプロキシ設定を有効にするために VM を再起動することを求められます。設定していない場合は、再起動することを求められず、手順 8 に進むことができます。

8. カスタム インターネット アクセス テスト URL を設定します。これを行う必要があるのは、デフォルトですべてのアウトバウンド接続を拒否する場合のみです。その場合は、許可リストにある https://google.com などのパブリックにアクセス可能な Web URL を指定します。

9. 最新のセキュリティパッチをインストールします。一部のパッチにはOS ツールと Docker サーバーが必要です。

10. プロンプトが表示されたら、スクリプトでSSH設定を強化するかどうかを指定します。

    シスコの VM を使用している場合は、続行します。独自の VM を使用しており、SSH を自分で設定している場合は、現在の設定を変更しないように、この手順をスキップすることができます。

11. SDC、SEC、CLI 自体の自動更新を有効にするように求められた場合は、これを実行して、バグ修正、パッチ、および新機能を最新の状態に保つことを推奨します。ポリシーによって自動更新が許可されない場合は、「Secure Device Connector の更新」を参照してください。

自身の VM 上での Secure Device Connector の展開

デバイスのログイン情報を使用して Security Cloud Control をデバイスに接続する場合、Security Cloud Control とデバイス間の通信を管理するために、ネットワークに Secure Device Connector（SDC）をダウンロードして展開することがベストプラクティスです。通常、これらのデバイスは非境界ベースであり、パブリック IP アドレスを持たないか、外部インターフェイスに開かれたポートを持っています。適応型セキュリティアプライアンス（ASA）、FDM による管理デバイス、および Firepower Management Center（FMC）デバイスはすべて、デバイスのログイン情報を使用して Security Cloud Control に対してオンボーディングすることができます。

SDC は、管理対象デバイスで実行する必要があるコマンドと、管理対象デバイスに送信する必要があるメッセージについて、Security Cloud Control を監視します。SDC は、Security Cloud Control に代わってこのコマンドを実行し、管理対象デバイスに代わって Security Cloud Control にメッセージを送信し、管理対象デバイスからの応答を Security Cloud Control に返します。

1 つの SDC が管理できるデバイスの数は、それらのデバイスに実装されている機能と、構成ファイルのサイズによって異なります。ただし、展開計画の目安として、1 つの SDC で約 500 台のデバイスをサポートできることを想定しています。詳細については、単一の組織で複数の Secure Device Connector を使用 （25 ページ）を参照してください。

この手順では、独自の仮想マシンイメージを使用してネットワークに SDC をインストールする方法について説明します。

（注） SDC をインストールするために推奨される、最も簡単で信頼できる方法は、Security Cloud Control の SDC OVA イメージをダウンロードしてインストールすることです。

始める前に

- Security Cloud Control は、厳密な証明書チェックを必要とし、SDC とインターネットの間の Web/コンテンツプロキシをサポートしていません。
- SDC が Security Cloud Control と通信するためには、TCP ポート 443 でのインターネットへの完全なアウトバウンドアクセスが必要です。
- SDC を介して Security Cloud Control に到達するデバイスは、ポート 443 で SDC からのインバウンドアクセスを許可する必要があります。
- ネットワークのガイドラインについては、「Secure Device Connector を使用した Security Cloud Control への接続」を参照してください。
- vCenter Web クライアントまたはr ESXi Web クライアントを使用してインストールされた VMware ESXi ホスト。

（注） vSphere デスクトップクライアントを使用したインストールはサポートしていません。

- ESXi 5.1 ハイパーバイザ。
- Ubuntu 22.04 および Ubuntu 24.04 オペレーティングシステム。
- SDC のみを持つ VM のシステム要件：
  - VMware ESXi ホストには 2 つの CPU が必要です。
  - VMware ESXi ホストには 2 GB 以上のメモリが必要です。
  - VMware ESXi では、プロビジョニングの選択に応じて、仮想マシンをサポートするために 64 GB のディスク容量が必要です。これは、必要に応じてディスク領域を拡張できるように、パーティションで論理ボリューム管理（LVM）を使用していることを想定した値です。
- VM の CPU とメモリを更新したら、VM の電源を入れ、[セキュアコネクタ（Secure Connectors）] ページに SDC が「アクティブ」状態であることが示されていることを確認します。
- この手順を実行するユーザーは、Linux 環境の操作に親しんでおり、vi ビジュアルエディタを使用してファイルを編集している必要があります。

- オンプレミスの SDC を CentOS 仮想マシンにインストールする場合は、Yum セキュリティパッチを定期的にインストールすることをお勧めします。Yum の更新を取得するための設定に応じて、ポート 443 だけでなくポート 80 でもアウトバウンドアクセスを開く必要がある場合があります。また、更新をスケジュールするために yum-cron または crontab も設定する必要があります。セキュリティ運用チームと連携して、Yum の更新を取得するためにセキュリティポリシーを変更する必要があるかどうかを判断します。

（注） **始める前に**：手順内のコマンドは、コピーして端末ウィンドウに貼り付けるのではなく入力するようにしてください。一部のコマンドに含まれる「n ダッシュ」は、カットアンドペーストのプロセスで「m ダッシュ」として適用される場合があり、コマンドが失敗する原因となります。

手順

ステップ 1 SDC を作成する Security Cloud Control テナントにログオンします。

ステップ 2 Security Cloud Control のホームページで、[ファイアウォール（Firewall）] をクリックします。

ステップ 3 左側のペインで **管理（Administration）** > **統合（Integrations）** > **セキュアコネクタ（Secure Connectors）** をクリックします。

ステップ 4 [サービス（Services）] ページの [セキュアコネクタ（Secure Connectors）] タブで、青いプラスボタンをクリックし、[Secure Device Connector] を選択します。

ステップ 5 ウィンドウの手順 2 のブートストラップデータをメモ帳にコピーします。

ステップ 6 少なくとも次の RAM とディスク領域が SDC に割り当てられている **CentOS 7 仮想マシン**をインストールします。

- 8 GB の RAM
- 10 GB のディスクスペース

ステップ 7 インストールしたら、SDC の IP アドレス、サブネットマスク、ゲートウェイの指定など、ネットワークの基本設定を行います。

ステップ 8 DNS（ドメインネームサーバー）を設定します。

ステップ 9 NTP（ネットワーク タイム プロトコル）サーバーを設定します。

ステップ 10 SDC の CLI と簡単にやり取りできるように、CentOS に SSH サーバーをインストールします。

ステップ 11 Yum の更新を実行し、**open-vm-tools**、**nettools**、および **bind-utils** パッケージをインストールします。

```
[root@sdc-vm ~]# yum update -y
                [root@sdc-vm ~]# yum install -y open-vm-tools net-tools bind-utils
```

ステップ 12 AWS CLI パッケージをインストールします。https://docs.aws.amazon.com/cli/latest/userguide/awscli-install-linux.htmlを参照してください。

（注） **--user** フラグは使用しないでください。

ステップ 13 Docker CE パッケージをインストールします。https://docs.docker.com/install/linux/docker-ce/centos/#install-docker-ceを参照してください。

（注） 「リポジトリを使用したインストール」方法を使用します。

ステップ 14 Docker サービスを開始し、起動時に開始できるようにします。

```
[root@sdc-vm ~]# systemctl start docker
[root@sdc-vm ~]# systemctl enable docker
Created symlink from /etc/systemd/system/multiuser.target.wants/docker.service to
    /usr/lib/systemd/system/docker.service.
```

ステップ 15 cdo と sdc の 2 つのユーザーを作成します。cdo ユーザーを使用すれば、ルートユーザーに直接アクセスせずに管理タスクを実行できます。sdc ユーザーは、SDC docker コンテナを実行するために指定します。

```
[root@sdc-vm ~]# useradd cdo
[root@sdc-vm ~]# useradd sdc -d /usr/local/cdo
```

ステップ 16 sdc ユーザーのパスワードを設定します。

```
[root@sdc-vm ~]# passwd cdo
Changing password for user cdo.
New password: <type password>
Retype new password: <type password>
passwd: all authentication tokens updated successfully.
```

ステップ 17 sdc ユーザーを wheel グループに追加して、管理者（sudo）権限を付与します。

```
[root@sdc-vm ~]# usermod -aG wheel cdo
[root@sdc-vm ~]#
```

ステップ 18 Docker がインストールされると、ユーザーグループが作成されます。CentOS/Docker のバージョンに応じて、「docker」または「dockerroot」と呼ばれます。/etc/group ファイルでどのグループが作成されたかを確認したら、sdc ユーザーをそのグループに追加します。

```
[root@sdc-vm ~]# grep docker /etc/group
 docker:x:993:
[root@sdc-vm ~]#
[root@sdc-vm ~]# usermod -aG docker sdc
[root@sdc-vm ~]#
```

ステップ 19 /etc/docker/daemon.json ファイルが存在しない場合は作成し、以下の内容を入力します。作成したら、docker デーモンを再起動します。

（注） 「group」キーに入力したグループ名が、前の手順の /etc/group ファイルで見つけたグループと一致していることを確認してください。

```
[root@sdc-vm ~]# cat /etc/docker/daemon.json
        {
           "live-restore": true,
           "group": "docker"
        }
        [root@sdc-vm ~]# systemctl restart docker
        [root@sdc-vm ~]#
```

ステップ 20 現在 vSphere コンソールセッションを使用している場合は、SSH に切り替えて、cdo ユーザーでログインします。ログインしたら、sdc ユーザーに切り替えます。パスワードの入力を求められたら、cdo ユーザーのパスワードを入力します。

```
[CDO@sdc-vm ~]$ sudo su sdc
              [sudo] password for cdo: <type password for cdo user>
              [sdc@sdc-vm ~]$
```

ステップ 21 ディレクトリを **/usr/local/CDO** に変更します。

ステップ 22 bootstrapdata という新しいファイルを作成し、[オンプレミスの Secure Device Connector の展開（Deploy an On-Premises Secure Device Connector）] ウィザードの手順2 のブートストラップデータを、このファイルに貼り付けます。[保存（Save）] をクリックしてファイルを保存します。[vi] または [nano] を使用してファイルを作成できます。

ステップ 23 ブートストラップデータは base64 でエンコードされていますので、復号して extractedbootstrapdata というファイルにエクスポートします。

```
[sdc@sdc-vm ~]$ base64 -d /usr/local/ CDO/bootstrapdata > /usr/local/CDO/extractedbootstrapdata
             [sdc@sdc-vm ~]$
```

cat コマンドを実行して復号したデータを表示します。コマンドおよび復号したデータは次のようになります。

```
[sdc@sdc-vm ~]$ cat /usr/local/ CDO/extractedbootstrapdata
               CDO_TOKEN="<token string>"
               CDO_DOMAIN="www.defenseorchestrator.com"
               CDO_TENANT="<tenant-name>"

CDO_BOOTSTRAP_URL="https://www.defenseorchestrator.com/sdc/bootstrap/tenant-name/<tenant-name-SDC>"
```

ステップ 24 以下のコマンドを実行して、復号したブートストラップデータの一部を環境変数にエクスポートします。

```
[sdc@sdc-vm ~]$ sed -e 's/^/export /g' extractedbootstrapdata > sdcenv && source sdcenv
             [sdc@sdc-vm ~]$
```

ステップ 25 Security Cloud Control からブートストラップバンドルをダウンロードします。

```
[sdc@sdc-vm ~]$ curl -O -H "Authorization: Bearer $CDO_TOKEN" "$CDO_BOOTSTRAP_URL"
               100 10314 100 10314 0 0 10656 0 --:--:-- --:--:-- --:--:-- 10654
               [sdc@sdc-vm ~]$ ls -l /usr/local/ CDO/*SDC
               -rw-rw-r--. 1 sdc sdc 10314 Jul 23 13:48 /usr/local/CDO/tenant-name-SDC
```

ステップ 26 SDC tarball を展開し、bootstrap.sh ファイルを実行して SDC パッケージをインストールします。

```
[sdc@sdc-vm ~]$ tar xzvf /usr/local/CDO/tenant-name-SDC
                <snipped – extracted files>
                [sdc@sdc-vm ~]$
                [sdc@sdc-vm ~]$ /usr/local/ CDO/bootstrap/bootstrap.sh
                [2018-07-23 13:54:02] environment properly configured
                download: s3://onprem-sdc/toolkit/prod/toolkit.tar to toolkit/toolkit.tar
                toolkit.sh
                common.sh
                [2018-07-23 13:54:04] startup new container
                Unable to find image 'ciscodefenseorchestrator/sdc_prod:latest' locally
                sha256:d98f17101db10e66db5b5d6afda1c95c29ea0004d9e4315508fd30579b275458: Pulling
from
                ciscodefenseorchestrator/sdc_prod
                08d48e6f1cff: Pull complete
                ebbd10b629b1: Pull complete
                d14d580ef2ed: Pull complete
                45421d451ab8: Pull complete
                <snipped – downloads>
                no crontab for sdc
```

すると、Security Cloud Control で SDC が「アクティブ」と表示されるはずです。

展開されたホストでの Secure Device Connector のブートストラップ

Procedure

ステップ 1 Security Cloud Control のホームページで、[ファイアウォール（Firewall）] をクリックします。

ステップ 2 左側のペインで **管理（Administration）** > **統合（Integrations）** > **セキュアコネクタ（Secure Connectors）** をクリックします。

ステップ 3 [Services] ページの [Secure Connectors] タブで、[+] アイコンをクリックし、[Secure Device Connector] を選択します。

ステップ 4 ウィンドウの手順 2 のブートストラップデータをメモ帳にコピーします。

ステップ 5 管理者ユーザー（通常は `cdo`）と選択したパスワードを使用して、VM に SSH で接続します。

ステップ 6 「sudo su - sdc」コマンドを使用して `sdc` ユーザーに切り替えます。

```
sudo su - sdc
```

ステップ 7 次のコマンドを使用して、新しい SDC をブートストラップします。

```
sdc bootstrap <paste-your-bootstrap-data-here>
```

ステップ 8 使用する SDC のバージョンを選択します。

SDC バージョンには 3 つのオプションがあります。

- SDC 2024：これは、ほとんどの場合に実行を希望されるバージョンです。
- FIPS が有効な SDC 2024：FedRamp 準拠の条件を満たす場合は、このバージョンを選択します。
- SDC レガシー：このバージョンは機能の更新を受信しなくなったため、代わりに SDC 2024 を実行することを推奨します。

ステップ 9 CLI はコンテナイメージをプルして SDC を起動します。また、以下を実行して、SDC がアクティブでユーザーインターフェイスおよびホスト上で動作していることを検証できます。

```
sdc show running
```

テナントの SDC が表示されます。

Terraform を使用した vSphere への Secure Device Connector の展開

始める前に

この手順では、vSphere 用 Security Cloud Control SDC Terraform モジュールを Security Cloud Control Terraform プロバイダーと組み合わせて使用して、vSphere に SDC を展開する方法について詳しく説明します。このタスク手順を実行する前に、次の前提条件を確認してください。

- vSphere データセンターバージョン 7 以降が必要です
- 次を実行する権限を持つデータセンターの管理者アカウントが必要です。
  - VM の作成
  - フォルダの作成
  - コンテンツライブラリの作成
  - コンテンツライブラリへのファイルのアップロード
- Terraform の知識

手順

ステップ 1 Security Cloud Control で API のみのユーザーを作成し、API トークンをコピーします。API のみのユーザーの作成方法については、「API のみのユーザーを作成する」を参照してください。

ステップ 2 「Security Cloud Control Terraform Provider」の手順に従って、Terraform リポジトリで Security Cloud Control Terraform プロバイダーを構成します。

例：

```
terraform {
  required_providers {
    cdo = {
      source = "CiscoDevNet/cdo"
      version = "0.7.0"
    }
  }
}

provider "cdo" {
  base_url = "<the CDO URL you use to access CDO>"
  api_token = "<the API Token generated in step 1>"
}
```

ステップ 3 Security Cloud Control Terraform プロバイダーを使用して `cdo_sdc` リソースを作成するための Terraform コードを記述します。詳細については、Security Cloud Control-sdc リソースの Terraform レジストリを参照してください。

例：

```
Resource “cdo_sdc” “my-sdc” {
  name = “my-sdc-in-vsphere”
}
```

このリソースの `bootstrap_data` 属性には、Security Cloud Control ブートストラップデータの値が入力され、次のステップで `cdo_sdc` Terraform モジュールに提供されます。

ステップ 4 Security Cloud Control_sdc Terraform モジュールを使用して、vSphere で SDC を作成するための Terraform コードを記述します。

例：

```
data "cdo_tenant" "current" {}

module "vsphere-cdo-sdc" {
  source               = "CiscoDevNet/cdo-sdc/vsphere"
  version              = "1.0.0"
  vsphere_username     = "<replace-with-username-with-admin-privileges>"
  vsphere_password     = "<super-secure-password>"
  vsphere_server       = "<replace-with-address-of-vsphere-server>"
  datacenter           = "<replace-with-datacenter-name>"
  resource_pool        = "<replace-with-resource-pool-name>"
  cdo_tenant_name      = data.cdo_tenant.current.human_readable_name
  datastore            = "<replace-with-name-of-datastore-to-deploy-vm-in>"
  network              = "<replace-with-name-of-network-to-deploy-vm-in>"
  host                 = "<replace-with-esxi-host-address>"
  allow_unverified_ssl = <boolean; set to true if your vsphere server does not have a valid SSL
certificate>
  ip_address          = "<sdc-vm-ip-address; must be in the subnet of the assigned network for the
 VM>"
  gateway              = "<replace-with-network-gateway-address>"
  cdo_user_password    = "<replace-with-password-for-cdo-user-in-sdc-vm>"
  root_user_password   = "<replace-with-password-for-root-user-in-sdc-vm>"
  cdo_bootstrap_data   = cdo_sdc.sdc-in-vsphere.bootstrap_data
}
```

作成された VM には 2 人のユーザー（`root` ユーザーと `cdo` というユーザー）があり、VM の IP アドレスは静的に設定されていることに注意してください。`cdo_bootstrap_data` 属性には、`cdo_sdc` リソースの作成時に生成された `bootstrap_data` 属性の値が指定されます。

ステップ 5 通常どおり、`terraform plan` と `terraform apply` を使用して Terraform を計画および適用します。

完全な例については、CiscoDevNet の「Security Cloud Control Automation Repository」[英語] を参照してください。

SDC がオンボーディング状態のままである場合は、リモートコンソールを使用して vSphere VM に接続し、`CDO` ユーザーとしてログインして、次のコマンドを実行します。

```
sdc host status
```

資料によっては、以下を手動で実行する必要がある場合があります。

```
sdc host configure
```

（注） Security Cloud Control Terraform モジュールは、Apache 2.0 ライセンスの下でオープンソースソフトウェアとして公開されています。サポートが必要な場合は、GitHub で問題を報告できます。

Terraform モジュールを使用した AWS VPC 上での Secure Device Connector の展開

始める前に

AWS VPC に SDC を展開する前に、次の前提条件を確認してください。

- Security Cloud Control は、厳密な証明書チェックを必要とし、SDC とインターネットの間の Web/コンテンツプロキシ検査をサポートしていません。プロキシサーバーを使用している場合は、Secure Device Connector（SDC）と Security Cloud Control の間のトラフィックの検査を無効にします。
- 適切なネットワークアクセスを確保するため、「Security Cloud Controlの Secure Device Connector への接続」を参照してください。
- AWS アカウント、少なくとも 1 つのサブネットを持つ AWS VPC、および AWS Route53 でホストされるゾーンが必要です。
- Security Cloud Control ブートストラップデータ、AWS VPC ID、およびそのサブネット ID が手元にあることを確認します。
- SDC を展開するプライベートサブネットに NAT ゲートウェイが接続されていることを確認します。
- ファイアウォール管理HTTPインターフェイスが実行されているポートで、ファイアウォールから NAT ゲートウェイに接続された Elastic IP へのトラフィックを開きます。

手順

ステップ 1 Terraform ファイルに次のコード行を追加します。変数の入力は手動で入力してください。

```
module "example-sdc" {
  source            = "git::https://github.com/cisco-lockhart/terraform-aws-cdo-sdc.git?ref=v0.0.1"

  env                = "example-env-ci"
  instance_name      = "example-instance-name"
  instance_size      = "r5a.xlarge"
  cdo_bootstrap_data = "<replace-with-cdo-bootstrap-data>"
  vpc_id             = <replace-with-vpc-id>
  subnet_id          = <replace-with-private-subnet-id>
}
```

入力変数と説明のリストについては、「Secure Device Connector Terraform module」を参照してください。

ステップ 2 Terraform コードの出力として `instance_id` を登録します。

```
output "example_sdc_instance_id" {
  value = module. example-sdc.instance_id
}
```

`instance_id` を使用して SDC インスタンスに接続し、AWS Systems Manager Session Manager（SSM）を使用してトラブルシューティングを行うことができます。使用可能な出力のリストについては、「Secure Device Connector Terraform module」の「Outputs」を参照してください。

次に行う作業：

SDC のトラブルシューティングでは、AWS SSM を使用して SDC インスタンスに接続する必要があります。インスタンスへの接続方法の詳細については、「AWS Systems Manager Session Manager」を参照してください。SSH を使用して SDC インスタンスに接続するためのポートは、セキュリティ上の理由により公開されないことに注意してください。

（注） Security Cloud Control Terraform モジュールは、Apache 2.0 ライセンスの下でオープンソースソフトウェアとして公開されています。サポートが必要な場合は、GitHub で問題を報告できます。

オンプレミス Secure Device Connector および Secure Event Connector の CentOS 7 仮想マシンから Ubuntu 仮想マシンへの移行

Security Cloud Control のオンプレミス Secure Device Connector（SDC）は、これまで CentOS 7 仮想マシンにインストールされていました。CentOS 7 はサポートが終了し、Security Cloud Control によって廃止されたため、すべての SDC を CentOS 7 から Ubuntu 仮想マシンに移行できるように、こちらの移行プロセスを準備しました。

移行する前に

- SDC には TCP ポート 443 でのインターネットへの完全なアウトバウンドアクセスが必要です。
- SDC を実行している Ubuntu 仮想マシンには、ASA や Cisco IOS デバイスなど、通信するデバイスの管理インターフェイスへのネットワークアクセスが必要です。
- デバイスに到達するため、元の SDC VM の IP アドレスまたは FQDN で作成されたネットワークルールを、新しい SDC VM の IP アドレスまたは FQDN を使用して作成し直す必要があります。
- 移行には 10 ～ 15 分かかります。移行中、デバイスは引き続きセキュリティポリシーを適用し、ネットワークトラフィックを回送しますが、SDC を介した通信はできません。

前提条件

「Secure Device Connector および Secure Event Connector を実行するための VM の展開」の手順に従って新しいホストを展開します。

ホストの設定

SDC や SEC を移行する場合は、次の手順に従います。

1. こちらから新しい VM イメージをダウンロードします。
2. CDO-SDC_VM.zip ファイルを解凍します。次のような名前の 3 つの VM ファイルが表示されます。
   - CDO-SDC-VM-708cd33-2024-05-30-2031-disk1.vmdk
   - CDO-SDC-VM-708cd33-2024-05-30-2031.mf
   - CDO-SDC-VM-708cd33-2024-05-30-2031.ovf
3. ダウンロードした VM を展開します。
4. 新しい VM に割り当てられた静的 IP アドレスまたは FQDN をメモします。
5. SSH を使用して、CDO ユーザーとして新しい VM にログインします。
6. プロンプトに次のコマンドを入力します。

```
sudo sdc host configure
```

（注）
- 移行スクリプトのプロンプトに厳密に従ってください。このスクリプトは適切に文書化されており、手順ごとの説明で移行プロセスをガイドします。
- 移行スクリプトの最後に、SDC が新しい VM に移行されたことを示すメッセージが表示されます。SDC の名前は、移行後も保持されます。

SDC の移行

手順：

1. SSH を使用して、CDO ユーザーとして元の（CentOS）SDC にログインします。
2. 次のコマンドを使用して CLI をインストールします。

```
curl -O
https://s3.us-west-2.amazonaws.com/download.defenseorchestrator.com/sdc-cli/sdc-cli-package-latest.tgz
 && tar -xvf sdc-cli-package-latest.tgz && chmod +x ./install.sh && ./install.sh
```

3. 次のコマンドを実行し、プロンプトに従います。

```
sudo sdc migrate now
```

確認：

1. Security Cloud Control テナントにログインします。
2. 移行先の SDC を選択して、[操作（Actions）] ペインで [ハートビートの要求（Request Heartbeat）] をクリックします。

（注） SDC が [アクティブ（Active）] 状態であることを確認します。

SEC の移行

手順：

1. SSH を使用して、CDO ユーザーとして元の（CentOS）SDC にログインします。
2. 次のコマンドを使用して CLI をインストールします。

```
curl -O
https://s3.us-west-2.amazonaws.com/download.defenseorchestrator.com/sdc-cli/sdc-cli-package-latest.tgz
 && tar -xvf sdc-cli-package-latest.tgz && chmod +x ./install.sh && ./install.sh
```

3. 次のコマンドを実行し、プロンプトに従います。

```
sudo sdc eventing migrate
```

4. SEC の新しい IP アドレスをポイントするようにデバイスを設定するか、元のホストをシャットダウンし、新しいホストに元のホストと同じ IP アドレスを割り当てます（この場合、デバイスの更新は不要です）。

確認：

SEC の状態に関する詳細は、「Secure Event Connector の状態を把握するためのヘルスチェックの使用」を参照してください。

追加手順

古い SDC を再起動しない

移行が完了したら、元の仮想マシンで古い SDC を再起動しないでください。

失敗した移行を元に戻す

何らかの理由で移行が失敗した場合、または期待どおりの結果でなかったため、元の SDC に戻す場合は、次の手順に従います。

1. 新しい VM にログインし、**SDC** ユーザーにスイッチします。
2. 次のコマンドを使用して、SDC が現在新しい VM で実行されていないことを確認します。

```
docker ps
```

3. SDC が実行されている場合は、次のコマンドを実行します。

```
sdc stop
```

4. 再度 `docker ps` を実行して、SDC が実行を停止したことを確認します。
5. 古い VM にログインし、次のコマンドを実行します。

```
sdc migrate revert
```

6. 古い SDC がアクティブで UI に表示されたら、新しい VM に戻り、次のコマンドを実行します。

   ```
   sdc delete <your-tenant-name-here>
   ```

7. ブラウザを完全に更新し、SDC をクリックして、古いホストの IP がサイドバーに表示されていることを確認します。

   これらの手順を実行しても、新しいIPが表示される場合は、新しいヘルスチェックを要求し、ブラウザを更新して、もう一度確認してください。

8. SEC 移行を元に戻すには、以下のコマンドを実行します。

   ```
   sdc eventing revert
   ```

Secure Device Connector の IP アドレスの変更

始める前に

- このタスクを実行するには、管理者である必要があります。
- SDC には、TCP ポート 443 またはデバイス管理用に設定したポートでのインターネットへの完全なアウトバウンドアクセスが必要です。

（注） SDC の IP アドレスを変更した後、デバイスを Security Cloud Control に再度オンボーディングする必要はありません。

手順

ステップ 1 SDC への SSH 接続を作成するか、仮想マシンのコンソールを開き、**CDO** ユーザーとしてログインします。

ステップ 2 IP アドレスを変更する前に SDC VM のネットワーク インターフェイス設定情報を表示するには、コマンドを使用します。

```
[cdo@localhost ~]$ ip addr
```

ステップ 3 インターフェイスの IP アドレスを変更するには、次のコマンドを使用してホスト設定を再度開始します。

```
[cdo@localhost ~]$ sdc host configure network
```

ステップ 4 プロンプトが表示されたら、パスワードを入力します。

ステップ 5 設定スクリプトは、ネットワーク設定について尋ね、新しい IP を使用して新しい設定ファイルを書き込み、その設定を適用します。

（注） この時点で、SSH 接続は失われます。

ステップ 6 SDC に割り当てた新しい IP アドレスを使用して SSH 接続を作成し、ログインします。

ステップ 7 SDC は自動的に起動しますが、起動しない場合は以下のコマンドを実行します。

```
[cdo@localhost ~]$ sudo su - sdc
[cdo@localhost ~]$ sdc start
```

（注） VM のコンソールでこの手順を実行している場合、値が正しいことを確認すると、接続ステータスのテストが自動的に実行され、ステータスが表示されます。

ステップ 8 Security Cloud Control ユーザーインターフェイスを介して SDC の接続を確認することもできます。確認するには、Security Cloud Control アプリケーションを開き、**管理（Administration）**>**統合（Integrations）**>**セキュアコネクタ（Secure Connectors）** ページに移動します。

ステップ 9 ページを一度更新し、IP アドレスを変更したセキュアコネクタを選択します。

ステップ 10 [操作（Actions）]ペインで、[ハートビートの要求（Request heartbeat）]をクリックします。「**ハートビートが正常に要求された**」というメッセージが表示され、**[Last Heartbeat]** に現在の日付と時刻が表示されるはずです。

ある Secure Device Connector SDC から別の SDC への ASA の移動

Security Cloud Control Firewall Management では、1 つのテナントあたり 1 つ以上の SDC の使用がサポートされます。次の手順を使用して、管理対象 ASA を、ある SDC から別の SDC に移動できます。

手順

ステップ 1 左側のペインで **セキュリティデバイス** をクリックします。

ステップ 2 [ASA] タブをクリックします。

ステップ 3 別の SDC に移動する 1 つ以上の ASA を選択します。

ステップ 4 [デバイスアクション（Device Actions）] ペインで、[資格情報の更新（Update Credentials）] をクリックします。

ステップ 5 [Secure Device Connector] ボタンをクリックし、デバイスの移動先の SDC を選択します。

ステップ 6 Security Cloud Control がデバイスにログインするために使用する管理者のユーザー名とパスワードを入力し、[更新（Update）] をクリックします。変更されていない限り、管理者のユーザー名とパスワードは、ASA のオンボードに使用したログイン情報と同じです。これらの変更をデバイスに展開する必要はありません。

（注） すべての ASA が同じログイン情報を使用している場合、複数の ASA を、ある SDC から別の SDC に一括で移動できます。複数の ASA のログイン情報が異なる場合、各 ASA をある SDC から別の SDC に 1つずつ移動する必要があります。

Secure Device Connector の名前変更

手順

ステップ 1 左側のペインで、**管理（Administration）**>**統合（Integrations）**>**セキュアコネクタ（Secure Connectors）**を選択します。

ステップ 2 名前を変更する SDC を選択します。

ステップ 3 詳細ペインで、SDC の名前の横にある編集アイコン をクリックします。

ステップ 4 SDC の名前を変更します。

この新しい名前は、Security Cloud Control インターフェース内の SDC 名が表示される場所に、**[Security Devices]** ペインの Secure Device Connector フィルタを含めて表示されます。

Secure Device Connector を手動で更新

このタスクは、障害対応ツールとして使用します。この送信元では、SDC は通常自動的に更新され、エラーが発生しない限り手動で更新する必要はないと説明しています。

Procedure

ステップ 1 SDC に接続します。SSH を使用して接続するか、ハイパーバイザのコンソールビューを使用できます。

ステップ 2 SDC に管理者ユーザー（通常は **cdo**）としてログインします。

ステップ 3 SDC ユーザーに切り替えて、SDC Docker コンテナを更新します。

```
[cdo@sdc-vm ~]$ sudo su sdc
                [sudo] password for cdo: <type password for cdo user>
                [sdc@sdc-vm ~]$
```

ステップ 4 SDC をアップグレードします。

```
sdc upgrade
```

Note **SDC 仮想マシンで推奨される更新およびメンテナンス**

必ず、組織の内部 IT セキュリティおよびパッチ管理ポリシーに従って、Ubuntu Linux で動作する SDC VM をモニターし、更新を適用してください。ネットワーク環境内で SDC VM のセキュリティ保護と最適な機能が維持されるように、関連するセキュリティパッチを定期的に確認して適用することを強くお勧めします。

単一の組織で複数の Secure Device Connector を使用

テナントにインストールできる SDC の数に制限はありません。複数の SDC により、より多くのデバイスを管理し、それらのデバイスを同じ Security Cloud Control Firewall Management テナントに配置したまま、コネクタを異なるネットワークセグメントに配置できます。2 つ目以降の SDC を展開する手順は、最初の SDC の手順と同じです。最初の SDC 名前にはテナント名と数字の「1」が含まれています。追加の各 SDC には、順番に番号が付けられます。

- 2 番目以降の SDC を展開する手順は、最初の SDC を展開する手順と同じです。
- テナントの最初の SDC には、テナントの名前と番号 1 が組み込まれています。追加の各 SDC には、順番に番号が付けられます。

同じ Secure Device Connector を介するデバイスを検索

同じ SDC を介して接続するデバイスを検索するには、次の手順を実行します。

Procedure

ステップ 1 左側のペインで **セキュリティデバイス** をクリックします。

ステップ 2 [デバイス（Devices）] タブをクリックしてデバイスを見つけます。

ステップ 3 適切なデバイスタイプのタブをクリックします。

ステップ 4 フィルタ基準がすでに指定されている場合は、[セキュリティデバイス（Security Devices）] ページの上部にある [クリア（Clear）] ボタンをクリックして、Security Cloud Control で管理しているすべてのデバイスとサービスを表示します。

ステップ 5 フィルタボタン をクリックして、[フィルタ（Filter）] メニューを展開します。

ステップ 6 フィルタの [Secure Device Connector] セクションで、必要な SDC の名前をクリックします。[セキュリティデバイス（Security Devices）] ページには、フィルタでチェックした SDC を使用して Security Cloud Control に接続しているデバイスのみが表示されます。

ステップ 7 （オプション）検索をさらに絞り込むには、フィルタメニューで追加のフィルタをチェックします。

ステップ 8 （オプション）完了したら、[セキュリティデバイス（Security Devices）] ページの上部にある [クリア（Clear）] ボタンをクリックして、Security Cloud Control で管理しているすべてのデバイスとサービスを表示します。

Secure Device Connector を削除

SDC の削除は元に戻せません。削除後、新しい SDC をインストールしてデバイスを再接続するまで、その SDC に接続されていたデバイスは管理できません。デバイスを再接続する場合、管理者のログイン情報を再度入力する必要がある場合があります。

SDC を削除する前に、接続されているデバイスを別の SDC に移動するか、Security Cloud Control Firewall Management から削除してください。

Procedure

ステップ 1 削除する SDC に接続されているデバイスをすべて削除します。この操作は、次の 2 つの方法で実行できます。

- 削除する SDC に接続されているすべてのデバイスを Security Cloud Control から削除します。
  - **a.** SDC で使用されるすべてのデバイスを特定するには、「同一 SDC を使用した Security Cloud Control に接続するすべてのデバイスを見つける」を参照してください。
  - **b.** [セキュリティデバイス（Security Devices）] ページで、識別したすべてのデバイスを選択します。
  - **c.** [デバイス アクション（Device Actions）] ウィンドウで [削除（Remove）] をクリックし、[OK] をクリックして操作を確定します。

ステップ 2 Security Cloud Control のホームページで、[ファイアウォール（Firewall）] をクリックします。

ステップ 3 左側のペインで **管理（Administration）** > **統合（Integrations）** > **セキュアコネクタ（Secure Connectors）** をクリックします。

ステップ 4 [サービス（Services）] ページの [セキュアコネクタ（Secure Connectors）] タブが選択された状態で、青いプラスボタンをクリックし、[Secure Device Connector] を選択します。

ステップ 5 [セキュアコネクタ（Secure Connectors）] テーブルで、削除する SDC を選択します。これで、デバイス数はゼロになっているはずです。

ステップ 6 [アクション（Actions）] ペインで、[削除（Remove）] アイコン をクリックします。次の警告が表示されます。

Warning <sdc_name> を削除しようとしています。SDC の削除は元に戻せません。SDC を削除すると、デバイスをオンボーディングまたは再オンボーディングする前に、新しい SDC を作成してオンボーディングする必要があります。

現在オンボーディング済みのデバイスがあるため、SDC を削除するには、これらのデバイスを再接続し、新しい SDC を設定した後にログイン情報を再度入力する必要があります。

- ご質問や懸念事項がある場合は、[キャンセル（Cancel）] をクリックして、Security Cloud Control サポートにお問い合わせください。
- 続行するには、下のテキストボックスに <sdc_name> を入力して、[OK] をクリックします。

ステップ 7 続行する場合は、警告メッセージに記載されている SDC の名前を確認ダイアログ ボックスに入力します。

ステップ 8 [OK] をクリックして、SDC の削除を確定します。

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。