AWS での Threat Defense Virtual の展開

この章では、AWS ポータルから Threat Defense Virtual を展開する方法について説明します。

Threat Defense Virtual と AWS クラウドについて

AWS はパブリッククラウド環境です。Threat Defense Virtual は、次のインスタンスタイプの AWS 環境でゲストとして実行されます。

表 *1 : AWS* でサポートされているインスタンス *Threat Defense Virtual*

インスタンス タイプ	Threat Defense Virtual	vCPU	メモリ（RAM（GB））	vNIC
C5.xlarge	6.6.0 以上	4	8	4
C 5.2 xlarge		8	16	4
C5.4xlarge		16	32	8

インスタンス タイプ	Threat Defense Virtual	vCPU	メモリ（RAM（GB））	vNIC
C4.xlarge	6.6.0 以前	4	7.5	4
C3.xlarge	6.6.0 以前（利用制限あり）	4	7.5	4

インスタンス タイプ	Threat Defense Virtual	vCPU	メモリ（RAM（GB））	vNIC
c5a.xlarge	7.1.0 以上	4	8	4
c5a.2xlarge		8	16	4
c5a.4xlarge		16	32	8
c5ad.xlarge		4	8	4
c5ad.2xlarge		8	16	4
c5ad.4xlarge		16	32	8
c5d.xlarge		4	8	4
c5d.2xlarge		8	16	4
c5d.4xlarge		16	32	8
c5n.xlarge		4	10.5	4
c5n.2xlarge		8	21	4
c5n.4xlarge		16	54	8
i3en.xlarge		4	32	4
i3en.2xlarge		8	64	4
i3en.3xlarge		12	96	4
inf1.xlarge		4	8	4
inf1.2xlarge		8	16	4
m5.xlarge		4	16	4
m5.2xlarge		8	32	4
m5.4xlarge		16	64	8
m5a.xlarge		4	16	4
m5a.2xlarge		8	32	4
m5a.4xlarge		16	64	8
m5ad.xlarge		4	16	4
m5ad.2xlarge		8	32	4
m5ad.4xlarge		16	64	8
m5d.xlarge		4	16	4

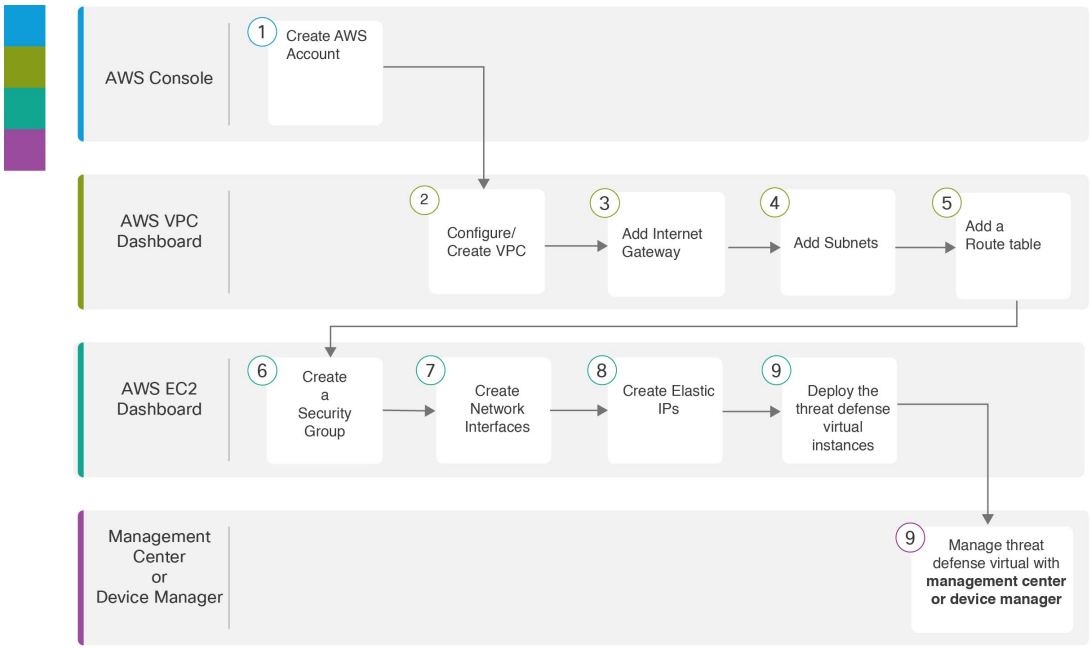
インスタンス タイプ	Threat Defense Virtual	vCPU	メモリ（RAM（GB））	vNIC
m5d.2xlarge		8	32	4
m5d.4xlarge		16	64	8
m5dn.xlarge		4	16	4
m5dn.2xlarge		8	32	4
m5dn.4xlarge		16	64	8
m5n.xlarge		4	16	4
m5n.2xlarge		8	32	4
m5n.4xlarge		16	64	8
m5zn.xlarge		4	16	4
m5zn.2xlarge		8	32	4
m5zn.3xlarge		12	48	8
r5.xlarge		4	32	4
r5.2xlarge		8	64	4
r5.4xlarge		16	128	8
r5a.xlarge		4	32	4
r5a.2xlarge		8	64	4
r5a.4xlarge		16	128	8
r5ad.xlarge		4	32	4
r5ad.2xlarge		8	64	4
r5ad.4xlarge		16	128	8
r5b.xlarge		4	32	4
r5b.2xlarge		8	64	4
r5b.4xlarge		16	128	8
r5d.xlarge		4	32	4
r5d.2xlarge		8	64	4
r5d.4xlarge		16	128	8
r5dn.xlarge		4	32	4

インスタンス タイプ	Threat Defense Virtual	vCPU	メモリ（RAM（GB））	vNIC
r5dn.2xlarge		8	64	4
r5dn.4xlarge		16	128	8
r5n.xlarge		4	32	4
r5n.2xlarge		8	64	4
r5n.4xlarge		16	128	8
z1d.xlarge		4	32	4
z1d.2xlarge		8	64	4
z1d.3xlarge		12	96	8

AWS マーケットプレイスにリストされている NGFWv でサポートされている EC2 インスタンスタイプについては、https://aws.amazon.com/marketplace/pp/prodview-p2336sqyya34e#pdp-overview を参照してください。

エンドツーエンドの手順

次のフローチャートは、Amazon Web Services（AWS）に Threat Defense Virtual を展開する際のワークフローを示しています。



	ワークスペース	手順
1	AWS コンソール	www.amazon.com：AWS コンソールでユーザーアカウントを作成します。
2	AWS VPC ダッシュボード	VPC の作成：AWS アカウント専用の VPC を作成および設定します。
3	AWS VPC ダッシュボード	インターネット ゲートウェイの追加：VPC をインターネットに接続するために、インターネットゲートウェイを追加します。
4	AWS VPC ダッシュボード	サブネットの追加：VPC にサブネットを追加します。
5	AWS VPC ダッシュボード	ルート テーブルの追加：VPC 用に設定したゲートウェイにルートテーブルを接続します。
6	AWS EC2 ダッシュボード	セキュリティ グループの作成：許可されるプロトコル、ポート、送信元 IP 範囲を指定するルールを使用して、セキュリティグループを作成します。
7	AWS EC2 ダッシュボード	ネットワーク インターフェイスの作成：静的 IP アドレスを使用して、Threat Defense Virtual のネットワーク インターフェイスを作成します。

	ワークスペース	手順
8	AWS EC2 ダッシュボード	Elastic IP の作成：Elastic IP は、Threat Defense Virtual および他のインスタンスへのリモートアクセスに使用されるパブリック IP 用に予約されます。
9	AWS EC2 ダッシュボード	Threat Defense Virtual インスタンスの展開：AWS ポータルから Threat Defense Virtual を展開します。
10	Management Centerまたは Device Manager	Threat Defense Virtual を次のように管理します。 • Firepower Management Center を使用した Firepower Threat Defense Virtual の管理 • Firepower Device Manager を使用した Firepower Threat Defense Virtual の管理

Secure Firewall Threat Defense Virtual デバイスの管理方法

Secure Firewall Threat Defense Virtual デバイスの管理には次の 2 つのオプションを選択できます。

Secure Firewall Management Center

多数のデバイスを管理している場合、または Threat Defense で許可される、より複雑な機能や設定を使用したい場合は、組み込みの Device Manager の代わりに Management Center を使用してデバイスを設定します。

重要 Device Manager と Management Center の両方を使用して Threat Defense デバイスを管理することはできません。いったん Device Manager の統合管理を有効にすると、ローカル管理を無効にして、Management Center を使用するように管理を再設定しない限り、Management Center を使用して Threat Defense デバイスを管理することはできなくなります。一方、Threat Defense デバイスを Management Center に登録すると、Device Manager のオンボード管理サービスは無効になります。

注意 現在、シスコには Device Manager の設定を Management Center に移行するオプションはありません。その逆も同様です。Threat Defense デバイス用に設定する管理のタイプを選択する際は、このことを考慮してください。

Secure Firewall Device Manager

Device Manager はオンボード統合マネージャです。

Device Manager は一部の Threat Defense デバイスに搭載された Web ベースの設定インターフェイスです。Device Manager では、小規模ネットワークで最も一般的に使用されるソフトウェアの基本機能を設定できます。また、これは多くの Threat Defense デバイスを含む大規模なネットワークを制御するために強力な複数デバイスのマネージャを使用することがない、単一のデバイスまたは限られた数のデバイスを含むネットワークのために特に設計されています。

（注） Device Manager をサポートする Threat Defense デバイスのリストについては、「Cisco Secure Firewall Device Manager Configuration Guide」を参照してください。

AWS ソリューションの概要

AWS は、Amazon.com によって提供されるリモート コンピューティング サービスの集合で、Web サービスとも呼ばれており、クラウド コンピューティング プラットフォームを構成します。これらのサービスは、世界の 11 の地理的地域で運用されます。通常、Secure Firewall Management Center Virtual（旧称 Firepower Management Center Virtual）および Threat Defense Virtual を展開する際には、以下の AWS サービスに精通している必要があります。

- Amazon Elastic Compute Cloud（EC2）：仮想コンピュータをレンタルして、お客様独自のアプリケーションおよびサービス（ファイアウォールなど）を Amazon のデータセンターで起動および管理できるようにする Web サービス。
- Amazon Virtual Private Cloud（VPC）：Amazon パブリック クラウド内の隔離されたプライベート ネットワークを設定できるようにする Web サービス。EC2 インスタンスは VPC 内で実行されます。
- Amazon Simple Storage Service（S3）：データ ストレージ インフラストラクチャを提供する Web サービス。

AWS でアカウントを作成し、VPC および EC2 コンポーネントを（AWS ウィザードまたは手動設定のいずれかを使用して）設定し、Amazon Machine Image（AMI）インスタンスを選択します。AMI は、インスタンスを起動するために必要なソフトウェア構成を含むテンプレートです。

（注） AMI イメージは AWS 環境の外部ではダウンロードできません。

前提条件

- AWS アカウントhttp://aws.amazon.com/ で 1 つ作成できます。
- Threat Defense Virtual コンソールにアクセスするには、SSH クライアント（例：Windows の場合は PuTTY、MacOS の場合はターミナル）が必要です。
- Cisco スマートアカウント。Cisco Software Central で 1 つ作成できます。https://software.cisco.com/
- Threat Defense Virtual へのライセンス付与。

  Cisco Secure Firewall Management Center

  - Management Center からセキュリティ サービスのすべてのライセンス資格を設定します。
  - ライセンスの管理方法の詳細については、『Firepower Management Center Configuration Guide』の「Licensing the System」を参照してください。

  Secure Firewall デバイスマネージャ

  - Secure Firewall デバイスマネージャ からセキュリティサービスのすべてのパフォーマンス階層型ライセンス資格を設定します。
  - ライセンスの管理方法の詳細については、「Threat Defense Virtual のライセンス」を参照してください。

- Threat Defense Virtual インターフェイスの要件：
  - 管理インターフェイス（2）：1 つは Threat Defense Virtual を Management Center に接続するために使用されます。もう 1 つは診断目的に使用され、通過トラフィックには使用できません。

    6.7 以降では、必要に応じて、管理インターフェイスの代わりにデータインターフェイスを Management Center の管理に使用できます。管理インターフェイスはデータインターフェイス管理の前提条件であるため、初期設定でこれを設定する必要があります。データインターフェイスから Management Center へのアクセスは、高可用性の展開ではサポートされません。Management Center へのアクセスに関するデータインターフェイス設定の詳細については、『FTD command reference』の **configure network management-data-interface** コマンドを参照してください。
  - トラフィック インターフェイス（2）：Threat Defense Virtual を内部のホストおよびパブリック ネットワークに接続するために使用されます。
- 通信パス：
  - Threat Defense Virtual にアクセスするためのパブリック IP/Elastic IP。

サポートされるソフトウェア プラットフォーム

Threat Defense Virtual Auto Scale ソリューションは、Management Center によって管理される Threat Defense Virtual に適用可能です。ソフトウェアバージョンには依存しません。『Cisco Firepower Compatibility Guide』には、オペレーティングシステムとホスティング環境の要件を含む、シスコのソフトウェアとハードウェアの互換性が記載されています。

- Firepower Management Centers: Virtual の表には、AWS 上の Management Center Virtual における互換性および仮想ホスティング環境の要件が一覧表示されています。
- Firepower Threat Defense Virtual Compatibility の表には、AWS 上の Threat Defense Virtual における互換性および仮想ホスティング環境の要件が一覧表示されています。

（注） AWS Auto Scale ソリューションを導入するためには、AWS 上で Threat Defense Virtual バージョン 6.4 以上を使用する必要があります。メモリベースのスケーリングを使用するには、Management Center バージョン 6.6 以降を実行している必要があります。

Threat Defense Virtual および AWS のガイドラインと制限事項

サポートされる機能

- 仮想プライベート クラウド（VPC）への導入
- 拡張ネットワーク（SR-IOV）。
- Amazon マーケットプレイスからの導入
- L3 ネットワークの導入
- ルーテッドモード（デフォルト）
- ERSPAN を使用するパッシブモード
- クラスタリング (バージョン 7.2 以降) 詳細については、『パブリッククラウドにおける Threat Defense Virtual のクラスタリング』を参照してください。
- Amazon CloudWatch によって記録されたヘルスモニタリングのメトリクス
- ジャンボ フレーム
- スナップショット (バージョン 7.2 以降)

サポートされない機能

- 複製

- IPv6
- トランスペアレントモード、インラインモード、パッシブモード

ライセンシング

- シスコ スマート ライセンス アカウントを使用する BYOL（Bring Your Own License）がサポートされています。
- PAYG（Pay As You Go）ライセンス。顧客がシスコ スマート ライセンシングを購入せずに Threat Defense Virtual を実行できる従量制課金モデル。登録された PAYG Threat Defense Virtualデバイスでは、ライセンス供与されたすべての機能（マルウェア、脅威、URL フィルタリング、VPNなど）が有効になっています。ライセンス供与された機能は、Management Center から編集または変更することはできません（バージョン 6.5 以上）。

（注） PAYG ライセンスは、Device Manager モードで展開されている Threat Defense Virtual デバイスではサポートされていません。

Threat Defense Virtual デバイスのライセンス取得のガイドラインについては、『Firepower Management Center Administration Guide』の「Licenses」の章を参照してください。

Threat Defense Virtual スマートライセンスのパフォーマンス階層

Threat Defense Virtual のバージョン 7.0.0 リリース以降では、Threat Defense Virtual は導入要件に基づいて異なるスループットレベルと VPN 接続制限を提供するパフォーマンス階層型ライセンスをサポートしています。

表 *2 : Threat Defense Virtual* 権限付与に基づくライセンス機能の制限

パフォーマンス階層	デバイス仕様（コア/RAM）	レート制限	RA VPN セッション制限
FTDv5	4 コア/8 GB	100 Mbps	50
FTDv10	4 コア/8 GB	1 Gbps	250
FTDv20	4 コア/8 GB	3Gbps	250
FTDv30	8 コア/16 GB	5 Gbps	250
FTDv50	12 コア/24 GB	10 Gbps	750
FTDv100	16 コア/34 GB	16 Gbps	10,000

パフォーマンスの最適化

Threat Defense Virtual の最高のパフォーマンスを実現するために、VM とホストの両方を調整することができます。詳細については、「AWS での仮想化の調整と最適化」を参照してください。

Receive Side Scaling：Threat Defense Virtual は Receive Side Scaling（RSS）をサポートしています。これは、ネットワークアダプタによって複数のプロセッサコアにネットワーク受信トラフィックを分散するために使用されるテクノロジーです。バージョン 7.0 以降でサポートされています。詳細については、「Receive Side Scaling（RSS）用の複数の RX キュー」を参照してください。

Threat Defense Virtual の制限事項

- 推奨されるインスタンスは c5.xlarge です。c3.xlarge インスタンスでは AWS リージョンでの可用性が制限されます。
- 起動時には、2 つの管理インターフェイスが構成されている必要があります。
- 起動するには、2 つのトラフィック インターフェイスと 2 つの管理インターフェイス（合計 4 つのインターフェイス）が必要です。

（注） Threat Defense Virtual はこの 4 つのインターフェイスがなければ起動しません。

- AWS でトラフィック インターフェイスを設定する場合、[送信元/宛先の変更の確認（Change Source/Dest. Check）] オプションを無効にする必要があります。
- IP アドレス設定は（CLI から設定したものでも Management Center から設定したものでも）AWS コンソールで作成されたものと一致する必要があります。展開時に設定を書き留めてください。
- Threat Defense Virtual を登録した後、インターフェイスを編集し、Management Center で有効にする必要があります。IP アドレスは、AWS で設定されたインターフェイスと一致している必要があることに注意してください。
- トランスペアレント モード、インライン モード、パッシブ モードは現時点でサポートされていません。
- インターフェイスを変更するには、AWS コンソールから変更を行う必要があります。AWS コンソールで、Management Center からインターフェイスの登録を解除し、AWS AMI ユーザーインターフェイスを使用しているインスタンスを停止します。次に、変更するインターフェイスを切り離し、新しいインターフェイスを接続します（起動するには、2 つのトラフィック インターフェイスと 2 つの管理インターフェイスが必要であることに注意してください）。ここで、インスタンスを起動し、Management Center に再登録します。

  Management Center から、デバイスインターフェイスを編集し、AWS コンソールから行った変更と一致するように、IP アドレスと他のパラメータを変更します。
- ブート後にインターフェイスを追加することはできません。

- Snort のシャットダウンに時間がかかったり、VM が全体的に遅くなったりといった異常な動作が見られる場合や、特定のプロセスが実行されるときには、Threat Defense Virtual および VM ホストからログを収集します。全体的な CPU 使用率、メモリ、I/O 使用率、および読み取り/書き込み速度のログの収集は、問題のトラブルシューティングに役立ちます。
- Snort のシャットダウン時には、CPU と I/O の使用率が高くなります。十分なメモリがなく、専用の CPU がない単一のホスト上に多数の Threat Defense Virtual インスタンスが作成されている場合は、Snort のシャットダウンに時間がかかって Snort コアが作成されます。

AWS 環境の設定

Threat Defense Virtual を AWS に展開するには、展開に固有の要件および設定を使用して Amazon VPC を設定する必要があります。ほとんどの環境では、セットアップ ウィザードに従ってセットアップを実行できます。AWS では、概要から詳細機能に至るまで、サービスに関する有用な情報を扱ったオンライン ドキュメントを提供しています。詳細については、https://aws.amazon.com/documentation/gettingstarted/ を参照してください。

AWS のセットアップを適切に制御するために、続くセクションでは、Threat Defense Virtual インスタンスの起動前の VPC および EC2 構成について説明します。

- VPC の作成 （13 ページ）
- インターネット ゲートウェイの追加 （14 ページ）
- サブネットの追加 （15 ページ）
- ルート テーブルの追加 （16 ページ）
- セキュリティ グループの作成 （16 ページ）
- ネットワーク インターフェイスの作成 （17 ページ）
- Elastic IP の作成 （18 ページ）

はじめる前に

- AWS アカウントを作成します。
- AMI を Threat Defense Virtual インスタンスに使用できることを確認します。

VPC の作成

仮想プライベート クラウド（VPC）は、AWS アカウント専用の仮想ネットワークです。これは、AWS クラウド内の他の仮想ネットワークから論理的に分離されています。Management Center Virtual や Threat Defense Virtual インスタンスなどの AWS リソースを VPC に起動できま

す。VPCを設定できます。さらに、そのIPアドレス範囲を選択し、サブネットを作成し、ルートテーブル、ネットワーク ゲートウェイ、およびセキュリティ設定を作成できます。

ステップ1 http://aws.amazon.com/ にログインし、地域を選択します。

AWSは互いに分かれた複数の地域に分割されています。地域は、画面の右上隅に表示されます。ある地域内のリソースは、別の地域には表示されません。目的の地域内に存在していることを定期的に確認してください。

ステップ2 [サービス（Services）] > [VPC] の順にクリックします。

ステップ3 [VPCダッシュボード（VPC Dashboard）] > [使用するVPC（Your VPCs）] の順にクリックします。

ステップ4 [VPCの作成（Create VPC）] をクリックします。

ステップ5 [VPCの作成（Create VPC）] ダイアログボックスで、次のものを入力します。

a) VPC を識別するユーザー定義の [名前タグ（Name tag）]。

b) IP アドレスの **IPv4 CIDR ブロック**。CIDR（クラスレス ドメイン間ルーティング）の表記法は、IP アドレスとそれに関連付けられているルーティング プレフィクスのコンパクトな表現です。たとえば、「10.0.0.0/24」と入力します。

c) [デフォルト（Default）] の [テナント（Tenancy）] 設定。この VPC で起動されたインスタンスが、起動時に指定されたテナント属性を使用するようにします。

ステップ6 [はい、作成します（Yes, Create）] をクリックして、VPC を作成します。

次のタスク

次のセクションで説明されているように、VPCにインターネットゲートウェイを追加します。

インターネット ゲートウェイの追加

VPCをインターネットに接続するために、インターネットゲートウェイを追加できます。VPCの外部のIPアドレスのトラフィックをインターネットゲートウェイにルーティングできます。

はじめる前に

- Threat Defense Virtual のインスタンスの VPC を作成します。

ステップ1 [サービス（Services）] > [VPC] の順にクリックします。

ステップ2 [VPC ダッシュボード（VPC Dashboard）] > [インターネットゲートウェイ（Internet Gateway）] の順にクリックしてから、[インターネットゲートウェイの作成（Create Internet Gateway）] をクリックします。

ステップ3 ユーザー定義の [名前タグ（Name tag）] を入力してゲートウェイを特定し、[はい、作成します（Yes, Create）] をクリックしてゲートウェイを作成します。

ステップ4 前のステップで作成したゲートウェイを選択します。

ステップ5 [VPCに接続（Attach to VPC）] をクリックして、以前に作成した VPC を選択します。

ステップ 6 [はい、接続します（Yes, Attach）] をクリックして、ゲートウェイを VPC に追加します。

デフォルトでは、ゲートウェイが作成されて VPC に接続されるまで、VPC で起動されたインスタンスはインターネットと通信できません。

次のタスク

次のセクションで説明されているように、VPC にサブネットを追加します。

サブネットの追加

Threat Defense Virtual のインスタンスが接続できる VPC の IP アドレス範囲をセグメント化することができます。セキュリティおよび運用のニーズに応じて、インスタンスをグループ化するためのサブネットを作成できます。Threat Defense Virtual では、管理用のサブネットとトラフィック用のサブネットを作成する必要があります。

はじめる前に

- Threat Defense Virtual のインスタンスの VPC を作成します。

ステップ 1 [サービス（Services）] > [VPC] の順にクリックします。

ステップ 2 [VPCダッシュボード（VPC Dashboard）]>[サブネット（Subnets）] の順にクリックして、[サブネットの作成（Create Subnet）] をクリックします。

ステップ 3 [サブネットの作成（Create Subnet）] ダイアログボックスで、次のものを入力します。

a) サブネットを識別するユーザー定義の [名前タグ（Name tag）]。
b) このサブネットに使用する [VPC]。
c) このサブネットが存在する [可用性ゾーン（Availability Zone）]。 [設定なし（No Preference）] を選択して、Amazon が選択するゾーンを選びます。
d) IP アドレスの [CIDRブロック（CIDR block）]。サブネットの IP アドレスの範囲は、VPC の IP アドレス範囲のサブセットである必要があります。ブロック サイズは、/16 ネットワーク マスクから /28 ネットワーク マスクの範囲で指定する必要があります。サブネットのサイズは VPC のサイズと同じにすることができます。

ステップ 4 [はい、作成します（Yes, Create）] をクリックして、サブネットを作成します。

ステップ 5 必要な数のサブネットについて、手順を繰り返します。管理トラフィックには別のサブネットを作成し、データ トラフィックに必要な数のサブネットを作成します。

次のタスク

次のセクションで説明されているように、VPC にルート テーブルを追加します。

ルート テーブルの追加

VPC 用に設定したゲートウェイにルート テーブルを接続できます。また、複数のサブネットを 1 つのルート テーブルに関連付けることができます。しかし、1 つのサブネットは一度に 1 つのルート テーブルにしか関連付けることができません。

ステップ 1 [サービス（Services）] > [VPC] の順にクリックします。

ステップ 2 [VPCダッシュボード（VPC Dashboard）] > [ルートテーブル（Route Tables）] の順にクリックしてから、[ルートテーブルの作成（Create Route Table）] をクリックします。

ステップ 3 ルート テーブルを識別するユーザー定義の [名前タグ（Name tag）] を入力します。

ステップ 4 このルート テーブルを使用する [VPC] をドロップダウン リストから選択します。

ステップ 5 [はい、作成します（Yes, Create）] をクリックして、ルート テーブルを作成します。

ステップ 6 作成したルート テーブルを選択します。

ステップ 7 [ルート（Routes）] タブをクリックして、詳細ペインにルート情報を表示します。

ステップ 8 [編集（Edit）] をクリックして、[別のルートを追加（Add another route）] をクリックします。

a) [宛先（Destination）] 列に、「0.0.0.0/0」を入力します。

b) [ターゲット（Target）] 列で、ゲートウェイを選択します。

ステップ 9 [保存（Save）] をクリックします。

次のタスク

次のセクションで説明するように、セキュリティ グループを作成します。

セキュリティ グループの作成

許可されるプロトコル、ポート、送信元 IP 範囲を指定するルールを使用して、セキュリティ グループを作成できます。各インスタンスに割り当てることができる、さまざまな異なるルールを使用して、複数のセキュリティ グループを作成できます。

ステップ 1 [サービス（Services）] > [EC2] の順にクリックします。

ステップ 2 [EC2ダッシュボード（EC2 Dashboard）] > [セキュリティグループ（Security Groups）] の順にクリックします。

ステップ 3 [セキュリティグループの作成（Create Security Group）] をクリックします。

ステップ 4 [セキュリティグループの作成（Create Security Group）] ダイアログボックスで、次の内容を入力します。

a) セキュリティ グループを識別するユーザー定義の [セキュリティグループ名（Security group name）]。

b) このセキュリティ グループの [説明（Description）]。

c) このセキュリティ グループに関連付けられた VPC。

ステップ 5 [セキュリティグループルール（Security group rules）] を設定します。

a) [インバウンド（Inbound）] タブをクリックして、[ルールの追加（Add Rule）] をクリックします。

（注） Management Center Virtual を AWS の外部から管理するには、HTTPS および SSH アクセスが必要です。それに基づいて、送信元 IP アドレスを指定する必要があります。また、Management Center Virtual と Threat Defense Virtual の両方を AWS VPC 内で設定している場合、プライベート IP 管理サブネットアクセスを許可する必要があります。

b) [アウトバウンド（Outbound）] タブをクリックしてから、[ルールの追加（Add Rule）] をクリックして、アウトバウンドトラフィックのルールを追加するか、デフォルトの [すべてのトラフィック（All traffic）]（[タイプ（Type）] の場合）および [任意の宛先（Anywhere）]（[宛先（Destination）] の場合）のままにします。

ステップ 6 セキュリティ グループを作成するには、[作成（Create）] をクリックします。

次のタスク

次のセクションで説明されているように、ネットワーク インターフェイスを作成します。

ネットワーク インターフェイスの作成

Threat Defense Virtual のネットワーク インターフェイスは、静的 IP アドレスまたは DHCP を使用して作成できます。具体的な展開の必要に応じてネットワーク インターフェイス（内部および外部）を作成します。

ステップ 1 [サービス（Services）] > [EC2] の順にクリックします。

ステップ 2 [EC2ダッシュボード（EC2 Dashboard）] > [ネットワークインターフェイス（Network Interfaces）] の順にクリックします。

ステップ 3 [ネットワークインターフェイスの作成（Create Network Interface）] をクリックします。

ステップ 4 [ネットワークインターフェイスの作成（Create Network Interface）] ダイアログボックスで、次のものを入力します。

a) ネットワーク インターフェイスに関するオプションのユーザー定義の [説明（Description）]。

b) ドロップダウン リストから [サブネット（Subnet）] を選択します。Threat Defense Virtual インスタンスを作成する VPC のサブネットが選択されていることを確認します。

c) [プライベートIP（Private IP）] アドレスを入力します。静的 IP アドレスまたは自動生成（DHCP）を使用できます。

d) [セキュリティグループ（Security groups）] を 1 つ以上選択します。セキュリティ グループの必要なポートがすべて開いていることを確認します。

ステップ 5 [ネットワーク インターフェイスの作成（Create network interface）] をクリックして、ネットワーク インターフェイスを作成します。

ステップ 6 作成したネットワーク インターフェイスを選択します。

ステップ 7 右クリックして、[送信元/宛先の変更の確認（Change Source/Dest. Check）]を選択します。

ステップ 8 [送信元または送信先の確認（Source/destination check）] の下にある [有効化（Enable）] チェックボックスをオフにして、[保存（Save）] をクリックします。

次のタスク

次のセクションで説明するように、Elastic IP アドレスを作成します。

Elastic IP の作成

インスタンスが作成されると、パブリック IP アドレスはそのインスタンスに関連付けられます。インスタンスを停止してから開始すると、そのパブリック IP アドレスは自動的に変更されます。この問題を解決するには、Elastic IP アドレッシングを使用して、永続的なパブリック IP アドレスをそのインスタンスに割り当てます。Elastic IP は、Threat Defense Virtual および他のインスタンスへのリモート アクセスに使用されるパブリック IP 用に予約されます。

（注） 少なくとも、Threat Defense Virtual 管理インターフェイス用と診断インターフェイス用の Elastic IP アドレスを作成してください。

ステップ 1 [サービス（Services）] > [EC2] の順にクリックします。
ステップ 2 [EC2ダッシュボード（EC2 Dashboard）] > [Elastic IP（Elastic IPs）] の順にクリックします。
ステップ 3 [新規アドレスの割り当て（Allocate New Address）] をクリックします。
ステップ 4 必要な数の Elastic IP およびパブリック IP について、この手順を繰り返します。
ステップ 5 [はい、割り当てます（Yes, Allocate）] をクリックして、Elastic IP を作成します。
ステップ 6 展開に必要な数の Elastic IP について、この手順を繰り返します。

次のタスク

次のセクションで説明されているように、Threat Defense Virtual を展開します。

Threat Defense Virtual インスタンスの展開

始める前に

次のことを推奨します。

- AWS 環境の設定 （13 ページ）の説明に従って、AWS VPC および EC2 のエレメントを設定します。
- AMI が Threat Defense Virtual インスタンスで使用できることを確認します。

ステップ 1 https://aws.amazon.com/marketplace（Amazon マーケットプレイス）に移動してサインインします。

ステップ 2 Amazon マーケットプレイスにログイン後、Threat Defense Virtual（Cisco Firepower NGFW Virtual（NGFWv）： BYOL）用に提供されているリンクをクリックします。

（注） すでに AWS を使用していた場合、リンクを有効にするには、いったんサインアウトしてから、サインインし直す必要があります。

ステップ 3 [続行（Continue）] をクリックしてから、[手動起動（Manual Launch）] タブをクリックします。

ステップ 4 [条件に同意する（Accept Terms）] をクリックします。

ステップ 5 [EC2コンソールを使用して起動する（Launch with EC2 Console）] をクリックします。

ステップ 6 Threat Defense Virtual でサポートされる [インスタンスタイプ（Instance Type）] を選択します。推奨タイプは c4.xlarge です。

ステップ 7 画面下部にある [次：インスタンスの詳細の設定（Next: Configure Instance Details）] ボタンをクリックします。

- 前に作成した VPC に一致するように [ネットワーク（Network）] を変更します。
- 前に作成した管理サブネットに一致するように [サブネット（Subnet）] を変更します。IP アドレスを指定するか、または自動生成を使用できます。
- [パブリックIP（Public IP）]の [自動生成（Auto-generate）] を有効にすることができます。
- [ネットワーク インターフェイス（Network Interfaces）] の下にある [デバイスの追加（Add Device）] ボタンをクリックして、eth1 ネットワーク インターフェイスを追加します。
- eth0 に使用される、事前に作成した管理サブネットに一致するように、[サブネット（Subnet）] を変更します。

（注） Threat Defense Virtual には 2 つの管理インターフェイスが必要です。

- [高度な詳細（Advanced Details）] の下で、デフォルトのログイン情報を追加します。デバイス名とパスワードの要件に合わせて、以下の例を変更してください。

注意：[高度な詳細（Advanced Details）] フィールドにデータを入力する際には、プレーンテキストのみを使用してください。テキストエディタからこの情報をコピーする場合、プレーンテキストとしてのみコピーしてください。[高度な詳細（Advanced Details）] フィールドに Unicode データ（空白を含む）をコピーする場合、インスタンスが破損する可能性があります。破損した場合は、インスタンスを終了して、作成し直す必要があります。

Management Center を使用して Threat Defense Virtual を管理するためのサンプルログイン設定：

```
#Sensor
{
            "AdminPassword": "<your_password>",
            "Hostname": "<your_hostname>",

            "ManageLocally": "No",
            "FmcIp":  "<IP address of FMC>",
            "FmcRegKey":"<registration_passkey>",
```

```
            "FmcNatId":"<NAT_ID_if_required>"
}
```

Device Manager を使用して Threat Defense Virtual を管理するためのサンプルログイン設定：

```
#Sensor
{
            "AdminPassword": "<your_password>",
            "Hostname": "<your_hostname>",
            "ManageLocally": "Yes"
}
```

ステップ 8 [次：ストレージの追加（Next: Add Storage）] をクリックします。

デフォルトを受け入れることも、ボリュームを変更することもできます。

ステップ 9 [次：タグ インスタンス（Next: Tag Instance）] をクリックします。

タグは大文字と小文字を区別するキーと値のペアで構成されます。たとえば、[キー（Key）]=名前、[値（Value）] = ファイアウォールでタグを定義できます。

ステップ 10 [次：セキュリティ グループの設定（Next: Configure Security Group）] を選択します。

ステップ 11 [既存のセキュリティグループを選択する（Select an existing Security Group）] をクリックして、以前に設定されたセキュリティグループを選択するか、または新しいセキュリティグループを作成できます。セキュリティ グループの作成の詳細については、AWS の資料を参照してください。

ステップ 12 [確認して起動する（Review and Launch）] をクリックします。

ステップ 13 [起動（Launch）] をクリックします。

ステップ 14 既存のキー ペアを選択するか、新しいキー ペアを作成します。

（注） 既存のキー ペアを選択することも、新しいキー ペアを作成することもできます。キー ペアは、AWS が保存する公開キーと、ユーザーが保存する秘密キーファイルで構成されます。これらを一緒に使用すると、インスタンスに安全に接続できます。キー ペアはインスタンスへの接続に必要となる場合があるため、必ず既知の場所に保存してください。

ステップ 15 [インスタンスの起動（Launch Instances）] をクリックします。

ステップ 16 [起動の表示（View Launch）] をクリックし、プロンプトに従います。

ステップ 17 [EC2ダッシュボード（EC2 Dashboard）]>[ネットワーク インターフェイス（Network Interfaces）] の順にクリックします。

ステップ 18 AWS 環境の設定 （13 ページ）で以前に作成したインターフェイス トラフィックを特定し、[接続（Attach）] をクリックします。これは、Threat Defense Virtual インスタンス上の **eth2** インターフェイスになります。

ステップ 19 AWS 環境の設定 （13 ページ）で以前に作成したインターフェイス トラフィックを特定し、[接続（Attach）] をクリックします。これは、Threat Defense Virtual インスタンス上の **eth3** インターフェイスになります。

（注） 4 つのインターフェイスを設定する必要があります。設定しないと、Threat Defense Virtual の起動プロセスが完了しません。

ステップ 20 [EC2ダッシュボード（EC2 Dashboard）] > [インスタンス（Instances）] の順にクリックします。

ステップ 21 インスタンスを右クリックし、[インスタンスの設定（Instance Settings）] > [システムログの取得（Get System Log）] の順に選択して、ステータスを表示します。

（注） 接続の問題に関する警告が表示される可能性があります。これが予想されるのは、EULA が完了するまで eth0 インターフェイスがアクティブにならないためです。

ステップ 22 20 分後、Threat Defense Virtual を Management Center に登録します。

次のタスク

次の手順は、選択した管理モードによって異なります。

- [ローカルマネージャを有効にする（Enable Local Manager）] で [いいえ（No）] を選択した場合は、Management Center を使用して Threat Defense Virtual を管理します。「Secure Firewall Management Center を使用した Secure Firewall Threat Defense Virtual の管理」を参照してください。
- [ローカルマネージャを有効にする（Enable Local Manager）] で [はい（Yes）] を選択した場合は、統合されている Device Manager を使用して Threat Defense Virtual を管理します。「Secure Firewall Device Manager を使用した Secure Firewall Threat Defense Virtual の管理」を参照してください。

管理オプションの選択方法の概要については、「Secure Firewall Threat Defense Virtual デバイスの管理方法」を参照してください。

AWS での Threat Defense Virtual の Auto Scale ソリューション

次のセクションでは、Auto Scale ソリューションのコンポーネントが AWS の Threat Defense Virtual でどのように機能するかについて説明します。

Auto Scale ソリューションについて

シスコでは、Lambda、Auto Scaling グループ、Elastic Load Balancing（ELB）、Amazon S3 バケット、SNS、CloudWatch などの複数の AWS サービスを使用して、Threat Defense Virtual ファイアウォールの Auto Scaling グループを導入するための CloudFormation テンプレートとスクリプトを提供しています。

AWS の Threat Defense Virtual Auto Scale は、AWS 環境の Threat Defense Virtual インスタンスに水平 Auto Scaling 機能を追加する、完全なサーバーレス実装です（つまり、この機能の自動化に関与するヘルパー VM はありません）。バージョン 6.4 以降、Threat Defense Virtual Auto Scale ソリューションは、Management Center によって管理される Threat Defense Virtual でサポートされます。

Threat Defense Virtual Auto Scale ソリューションは、以下の内容を提供する CloudFormation テンプレートベースの導入です。

- Management Center による Threat Defense Virtual インスタンスの登録と登録解除の完全な自動化。
- スケールアウトされた Threat Defense Virtual インスタンスへの NAT ポリシー、アクセスポリシー、およびルートの自動適用。
- ロードバランサとマルチ可用性ゾーンのサポート。
- Auto Scale 機能の有効化と無効化をサポート。
- Management Center でのみ動作し、Device Manager はサポート対象外。

Auto Scale の機能拡張（バージョン 6.7）

- カスタム指標パブリッシャ：新しい Lambda 関数は、Auto Scale グループ内のすべての Threat Defense Virtual インスタンスのメモリ消費量について Management Center を 2 分ごとにポーリングし、その値を CloudWatch メトリックにパブリッシュします。詳細については、「入力パラメータ （30 ページ）」を参照してください。
- メモリ消費に基づく新しいスケーリングポリシーを使用できます。
- Management Center への SSH およびセキュアトンネル用の Threat Defense Virtual プライベート IP 接続。
- Management Center 設定の検証.
- ELB でより多くのリスニングポートを開くためのサポート。
- シングルスタック展開に変更。すべての Lambda 関数と AWS リソースは、合理化された展開のためにシングルスタックから展開されます。

Auto Scale の導入例

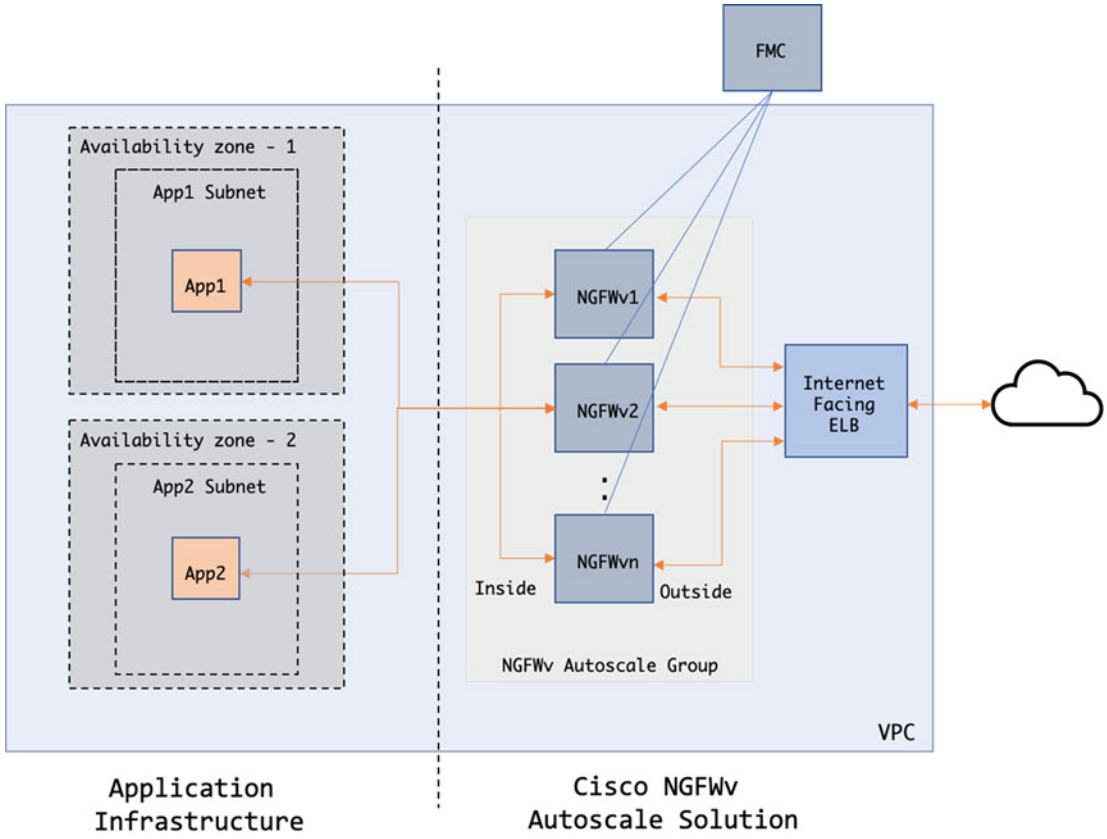
この Threat Defense Virtual AWS Auto Scale ソリューションの導入例は、「図 1 : Threat Defense Virtual Auto Scale の導入例の図 （23 ページ）」に示されています。AWS ロードバランサはインバウンドで開始された接続のみを許可するため、外部で生成されたトラフィックのみが Cisco Threat Defense Virtual ファイアウォール経由で内部を通過できます。

（注） 前提条件のSSL サーバー証明書 （28 ページ）で説明されているように、セキュアなポートには SSL/TLS 証明書が必要です。

インターネットに面したロードバランサは、ネットワークロードバランサまたはアプリケーション ロードバランサです。いずれの場合も、AWS のすべての要件と条件が適用されます。導入例の図に示されているように、点線の右側部分は Threat Defense Virtual テンプレートを介して展開されます。左側は完全にユーザー定義の部分です。

（注） アプリケーションが開始したアウトバウンドトラフィックは Threat Defense Virtual を通過しません。

図 *1 : Threat Defense Virtual Auto Scale* の導入例の図



トラフィックのポートベースの分岐が可能です。この分岐は、NAT ルールによって実現できます。Management Center でのオブジェクト、デバイスグループ、NAT ルール、アクセスポリシーの設定（38 ページ）を参照してください。たとえば、インターネットに面した LB DNS、ポート：80 のトラフィックは、アプリケーション 1 にルーティングでき、ポート：88 のトラフィックはアプリケーション 2 にルーティングできます。

AWS ゲートウェイロードバランサの自動スケールの導入例

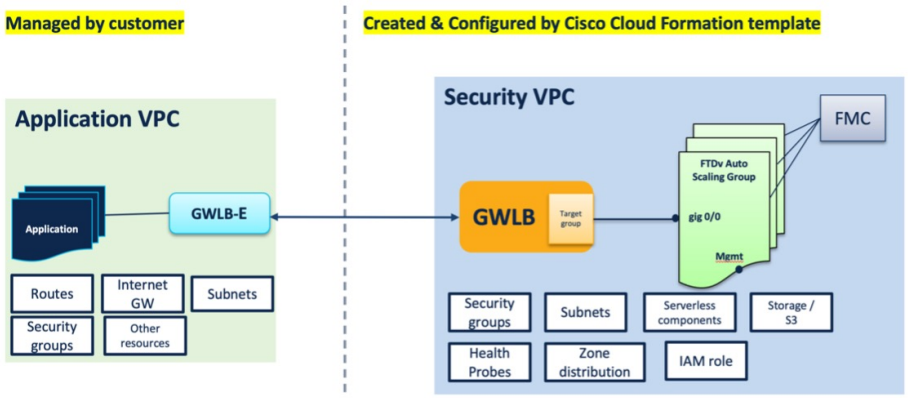
この Threat Defense Virtual AWS Gateway Load Balancer（GWLB）Auto Scale ソリューションの導入例は、導入例の図に示されています。AWS Gateway Load Balancer はインバウンド接続とアウトバウンド接続の両方を許可するため、内部と外部で生成されたトラフィックは共に Cisco Threat Defense Virtual ファイアウォール経由で内部を通過できます。

インターネットに接続するロードバランサは、AWS Gateway Load Balancer のエンドポイント（GWLBe）にすることができます。GWLBe はトラフィックを GWLB に送信し、次に検査の

ために Threat Defense Virtual に送信します。いずれの場合も、AWS のすべての要件と条件が適用されます。導入例の図に示されているように、点線の右側部分は Threat Defense Virtual テンプレートを介して展開された Threat Defense Virtual GWLB 自動スケールソリューションです。左側は完全にユーザー定義の部分です。

（注） アプリケーションが開始したアウトバウンドトラフィックは Threat Defense Virtual を通過しません。

図 *2 : AWS GWLB Auto Scale* の導入例の図



Auto Scale ソリューションの仕組み

Threat Defense Virtual インスタンスをスケールインおよびスケールアウトするには、Auto Scale Manager と呼ばれる外部エンティティがメトリックをモニターし、Auto Scale グループに Threat Defense Virtual インスタンスの追加または削除を指示し、Threat Defense Virtual デバイスを Management Center に登録および登録解除して、 Threat Defense Virtualインスタンスを設定します。

Auto Scale Manager は、AWS サーバーレスアーキテクチャを使用して実装され、AWS リソース、Threat Defense Virtual、Management Center と通信します。シスコでは、Auto Scale Manager コンポーネントの導入を自動化する CloudFormation テンプレートを提供しています。このテンプレートにより、包括的なソリューションが機能するために必要なその他のリソースも展開されます。

（注） サーバーレス Auto Scale スクリプトは CloudWatch イベントによってのみ呼び出されるため、インスタンスの起動時にのみ実行されます。

Auto Scale ソリューションのコンポーネント

Auto Scale ソリューションは、次のコンポーネントで構成されています。

CloudFormation テンプレート

CloudFormation テンプレートは、AWS の Auto Scale ソリューションに必要なリソースを展開するために使用されます。テンプレートの構成は次のとおりです。

- Auto Scale グループ、ロードバランサ、セキュリティグループ、およびその他のコンポーネント。
- 展開をカスタマイズするためのユーザー入力を取り込むテンプレート。

（注） テンプレートのユーザー入力の検証には限界があるため、展開時に入力を検証するのはユーザーの責任です。

Lambda 関数

Auto Scale ソリューションは、Python で開発された一連の Lambda 関数で、ライフサイクルフック、SNS、CloudWatch イベントやアラームイベントからトリガーされます。基本的な機能は次のとおりです。

- インスタンスに対して Diag、Gig0/0、および Gig 0/1 インターフェイスを追加/削除します。
- ロードバランサのターゲットグループに Gig0/1 インターフェイスを登録します。
- Management Center で Threat Defense Virtual を新規登録します。
- Management Center を介して新規の Threat Defense Virtual を展開します。
- スケールインした Threat Defense Virtual を Management Center から登録解除（削除）します。
- Management Center からメモリメトリックをパブリッシュします。

Lambda 関数は、Python パッケージの形式でお客様に提供されます。

ライフサイクルフック

- ライフサイクルフックは、インスタンスに関するライフサイクルの変更通知を取得するために使用されます。
- インスタンス起動の場合、ライフサイクルフックを使用して、Threat Defense Virtual インスタンスにインターフェイスを追加し、ターゲットグループに外部インターフェイス IP を登録できる Lambda 関数をトリガーします。
- インスタンス終了の場合、ライフサイクルフックを使用して Lambda 関数をトリガーし、ターゲットグループから Threat Defense Virtual インスタンスを登録解除します。

Simple Notification Service（SNS）

- AWS の Simple Notification Service（SNS）を使用してイベントが生成されます。
- AWS にはサーバーレス Lambda 関数に適した Orchestrator がないという制限があるため、ソリューションは、イベントに基づいて Lambda 関数をオーケストレーションするための一種の関数チェーンとして SNS を使用します。

Auto Scale ソリューションの前提条件

展開ファイルのダウンロード

Threat Defense Virtual Auto Scale for AWS ソリューションの起動に必要なファイルをダウンロードします。該当する バージョン用の展開スクリプトとテンプレートは、GitHub リポジトリから入手できます。

注目 Auto Scale 用のシスコ提供の導入スクリプトおよびテンプレートは、オープンソースの例として提供されており、通常の Cisco TAC サポートの範囲内ではカバーされないことに注意してください。更新と ReadMe の手順については、GitHub を定期的に確認してください。

インフラストラクチャ設定

複製/ダウンロードされた GitHub リポジトリでは、**infrastructure.yaml** ファイルおよび **infrastructure_gwlb.yaml** ファイルはテンプレートフォルダ内にあります。この CFT は、バケットポリシーを使用して VPC、サブネット、ルート、ACL、セキュリティグループ、VPC エンドポイント、および S3 バケットを展開するために使用できます。この CFT は、要件に合わせて変更できます。

次の項では、これらのリソースと Auto Scale での使用について詳しく説明します。これらのリソースを手動で展開し、Auto Scale で使用することもできます。

（注） **infrastructure.yaml** テンプレートは、VPC、サブネット、ACL、セキュリティグループ、S3 バケット、および VPC エンドポイントのみを展開します。SSL 証明書、Lambda レイヤ、または KMS キーリソースは作成されません。

Infrastructure_gwlb.yaml テンプレートは、AWS GWLB Auto Scale ソリューションを展開します。

VPC

アプリケーション要件に応じて VPC を作成する必要があります。VPC には、インターネットへのルートがある少なくとも 1 つのサブネットを持つインターネットゲートウェイがあること

が想定されます。セキュリティグループ、サブネットなどの要件については、該当するセクションを参照してください。

サブネット

サブネットは、アプリケーションの要件に応じて作成できます。導入例に示されているように、Threat Defense Virtual マシン の動作には 3 つのサブネットが必要です。

（注） 複数の可用性ゾーンのサポートが必要な場合、サブネットは AWS クラウド内のゾーンプロパティであるため、各ゾーンにサブネットが必要です。

外部サブネット

外部サブネットには、インターネットゲートウェイへの「0.0.0.0/0」のデフォルトルートが必要です。このサブネットには、Threat Defense Virtual の外部インターフェイスが含まれ、インターネットに面した NLB も含まれます。

内部サブネット

これは、NAT/インターネットゲートウェイの有無にかかわらず、アプリケーションサブネットに似ています。Threat Defense Virtual の正常性プローブでは、ポート 80 経由で AWS メタデータサーバー（169.254.169.254）に到達できる必要があることに注意してください。

（注） この AutoScale ソリューションでは、ロードバランサの正常性プローブが inside/Gig0/0 インターフェイスを介して AWS メタデータサーバーにリダイレクトされます。ただし、ロードバランサから Threat Defense Virtual に送信される正常性プローブ接続を提供する独自のアプリケーションでこれを変更できます。この場合、AWS メタデータ サーバー オブジェクトをそれぞれのアプリケーションの IP アドレスに置き換えて、正常性プローブ応答を提供する必要があります。

管理サブネット

このサブネットには、Threat Defense Virtual 管理インターフェイスが含まれます。 このサブネットで Management Center を使用している場合、Threat Defense Virtual への Elastic IP アドレス（EIP）の割り当ては任意です。診断インターフェイスもこのサブネット上にあります。

Lambda サブネット

AWS Lambda 関数では、デフォルトゲートウェイとして NAT ゲートウェイを持つ 2 つのサブネットが必要です。これにより、Lambda 関数が VPC に対してプライベートになります。Lambda サブネットは、他のサブネットと同じ幅である必要はありません。Lambda サブネットのベストプラクティスについては、AWS のドキュメントを参照してください。

アプリケーションサブネット

Auto Scale ソリューションからこのサブネットに課せられる制限はありませんが、アプリケーションに VPC 外部のアウトバウンド接続が必要な場合は、サブネット上にそれぞれのルートが設定されている必要があります。これは、アウトバウンドで開始されたトラフィックがロードバランサを通過しないためです。AWS Elastic Load Balancing ユーザーガイド [英語] を参照してください。

セキュリティ グループ

提供された Auto Scale グループテンプレートでは、すべての接続が許可されます。Auto Scale ソリューションを機能させるために必要なのは、次の接続だけです。

表 *3:* 必須のポート

ポート	使用方法	サブネット
8305	Management Center から Threat Defense Virtual へのセキュアなトンネル接続	管理サブネット
正常性プローブポート（デフォルト：8080）	インターネットに面したロードバランサの正常性プローブ	外部サブネット、内部サブネット
アプリケーションポート	アプリケーションデータ トラフィック	外部サブネット、内部サブネット

Management Center インスタンスのセキュリティグループまたは ACL

Lambda 関数と Management Center 間の HTTPS 接続を許可します。Lambda 関数は、NAT ゲートウェイをデフォルトルートとして持つ Lambda サブネットに保持されるため、Management Center は NAT ゲートウェイ IP アドレスからのインバウンド HTTPS 接続を持つことができます。

Amazon S3 バケット

Amazon Simple Storage Service（Amazon S3）は、業界をリードする拡張性、データ可用性、セキュリティ、およびパフォーマンスを提供するオブジェクト ストレージ サービスです。ファイアウォール テンプレートとアプリケーション テンプレートの両方に必要なすべてのファイルを S3 バケットに配置できます。

テンプレートが展開されると、S3 バケット内の Zip ファイルを参照して Lambda 関数が作成されます。したがって、S3 バケットはユーザーアカウントにアクセス可能である必要があります。

SSL サーバー証明書

インターネットに面したロードバランサが TLS/SSL をサポートしている必要がある場合、証明書 ARN が必要です。詳細については、次のリンクを参照してください。

- サーバー証明書の使用
- テスト用の秘密キーと自己署名証明書の作成
- 自己署名 SSL 証明書を使用した AWS ELB の作成（サードパーティリンク）

ARN の例：arn:aws:iam::[AWS Account]:server-certificate/[Certificate Name]

Lambda レイヤ

autoscale_layer.zip は、Python 3.9 がインストールされた Ubuntu 18.04 などの Linux 環境で作成できます。

```
#!/bin/bash
mkdir -p layer
virtualenv -p /usr/bin/python3.9 ./layer/
source ./layer/bin/activate
pip3 install cffi==1.15.1
pip3 install cryptography==2.9.1
pip3 install paramiko==2.7.1
pip3 install requests==2.23.0
pip3 install scp==0.13.2
pip3 install jsonschema==3.2.0
pip3 install pycryptodome==3.15.0
echo "Copy from ./layer directory to ./python\n"
cp -r ./layer/lib/python3.9/site-packages/* ./python/
zip -r autoscale_layer.zip ./python
```

作成された ***autoscale_layer.zip*** ファイルは、*lambda-python-files* フォルダにコピーする必要があります。

KMS マスターキー

これは、Management Center および Threat Defense Virtual パスワードが暗号化形式の場合に必要です。それ以外の場合、このコンポーネントは必要ありません。パスワードは、ここで提供される KMS のみを使用して暗号化する必要があります。KMS ARN が CFT で入力される場合、パスワードを暗号化する必要があります。それ以外の場合、パスワードはプレーンテキストである必要があります。

マスターキーと暗号化の詳細については、パスワードの暗号化と KMS に関する AWS のドキュメントのキーの作成 [英語] と AWS CLI コマンドリファレンス [英語] を参照してください。

例：

```
$ aws kms encrypt --key-id <KMS-ARN> --plaintext 'MyC0mplIc@tedProtect1oN'
{
    "KeyId": "KMS-ARN",
    "CiphertextBlob":
"AQICAHgcQFAGtz/hvaxMtJvY/x/rfHnKI3clFPpSXUU7HQRnCAFwfXhXHJAHL8tcVmDqurALAAAAajBoBgkqhki
G9w0BBwagWzBZAgEAMFQGCSqGSIb3DQEHATAeBglghkgBZQMEAS4wEQQM45AIkTqjSekX2mniAgEQgCcOav6Hhol
+wxpWKtXY4y1Z1d0z1P4fx0jTdosfCbPnUExmNJ4zdx8="
}
$
```

CiphertextBlob キーの値をパスワードとして使用する必要があります。

Python 3 環境

make.py ファイルは、複製されたリポジトリの最上位ディレクトリにあります。これにより、python ファイルが Zip ファイルに圧縮され、ターゲットフォルダにコピーされます。これらのタスクを実行するには、Python 3 環境が使用可能である必要があります。

Auto Scale の展開

準備

アプリケーションが展開されているか、アプリケーションの展開プランが利用可能である必要があります。

入力パラメータ

導入前に、次の入力パラメータを収集する必要があります。

（注） AWS Gateway Load Balancer（GWLB）の場合**LoadBalancerType**、**LoadBalancerSG**、**LoadBalancerPort**、および**SSLcertificate** パラメータは対象外です。

表 *4 : Auto Scale* 入力パラメータ

パラメータ	使用できる値/タイプ	説明
PodNumber	文字列 許可パターン：'^\d{1,3}$'	これはポッド番号です。Auto Scale グループ名（Threat Defense Virtual-Group-Name）の末尾に追加されます。たとえば、この値が「1」の場合、グループ名は *Threat Defense Virtual-Group-Name-1.* になります。 1 桁以上 3 桁以下の数字である必要があります。 デフォルト：1
AutoscaleGrpNamePrefix	文字列	これはAuto Scaleグループ名プレフィックスです。ポッド番号がサフィックスとして追加されます。 最大：18 文字 例：Cisco-Threat Defense Virtual-1

パラメータ	使用できる値/タイプ	説明
NotifyEmailID	文字列	Auto Scale イベントはこの電子メールアドレスに送信されます。サブスクリプション電子メール要求を受け入れる必要があります。 例：admin@company.com
VpcId	文字列	デバイスを展開する必要がある VPC ID。これは、AWS の要件に従って設定する必要があります。 タイプ：AWS::EC2::VPC::Id 「*infrastructure.yaml*」ファイルを使用してインフラストラクチャを展開すると、スタックの出力セクションにこの値が設定されます。その値を使用してください。
LambdaSubnets	リスト	Lambda 関数が展開されるサブネット。 タイプ：List<AWS::EC2::Subnet::Id> 「*infrastructure.yaml*」ファイルを使用してインフラストラクチャを展開すると、スタックの出力セクションにこの値が設定されます。その値を使用してください。
LambdaSG	リスト	Lambda 機能のセキュリティグループ。 タイプ：List<AWS::EC2::SecurityGroup::Id> 「*infrastructure.yaml*」ファイルを使用してインフラストラクチャを展開すると、スタックの出力セクションにこの値が設定されます。その値を使用してください。
S3BktName	文字列	ファイルの S3 バケット名。これは、AWS の要件に従ってアカウントに設定する必要があります。 「*infrastructure.yaml*」ファイルを使用してインフラストラクチャを展開すると、スタックの出力セクションにこの値が設定されます。その値を使用してください。
LoadBalancerType	文字列	インターネットに面したロードバランサのタイプ（「アプリケーション」または「ネットワーク」）。 例：アプリケーション

パラメータ	使用できる値/タイプ	説明
LoadBalancerSG	文字列	ロードバランサのセキュリティグループ。ネットワークロードバランサの場合は使用されません。ただし、セキュリティグループ ID を指定する必要があります。 タイプ：List<AWS::EC2::SecurityGroup::Id> 「*infrastructure.yaml*」ファイルを使用してインフラストラクチャを展開すると、スタックの出力セクションにこの値が設定されます。その値を使用してください。
LoadBalancerPort	整数	ロードバランサポート。このポートは、選択したロードバランサタイプに基づいて、プロトコルとして HTTP/HTTPS または TCP/TLS を使用して LB で開きます。 ポートが有効な TCP ポートであることを確認します。これはロードバランサリスナーの作成に使用されます。 デフォルト：80
SSL認証	文字列	セキュアポート接続の SSL 証明書の ARN。指定しない場合、ロードバランサで開かれるポートは TCP/HTTP になります。指定した場合、ロードバランサで開かれるポートは TLS/HTTPS になります。
TgHealthPort	整数	このポートは、正常性プローブのターゲットグループによって使用されます。Threat Defense Virtual のこのポートに到達する正常性プローブは、AWS メタデータサーバーにルーティングされるため、トラフィックには使用しないでください。このポートは有効な TCP ポートである必要があります。 アプリケーション自体が正常性プローブに応答するようにする場合は、それに応じて Threat Defense Virtual の NAT ルールを変更できます。このような場合、アプリケーションが応答しないと、Threat Defense Virtual は Unhealthy インスタンスのしきい値アラームにより、非正常としてマークされ、削除されます。 例：8080

パラメータ	使用できる値/タイプ	説明
AssignPublicIP	ブール値	「true」を選択すると、パブリック IP が割り当てられます。BYOL タイプの Threat Defense Virtual の場合、これは https://tools.cisco.com に接続するために必要です。 例：TRUE
InstanceType	文字列	Amazon マシンイメージ（AMI）は、さまざまなインスタンスタイプをサポートしています。インスタンスタイプによって、インスタンスのサイズと必要なメモリ容量が決まります。 Threat Defense Virtual をサポートする AMI インスタンスタイプのみを使用する必要があります。 例：c4.2xlarge
LicenseType	文字列	Threat Defense Virtual ライセンスタイプ（BYOL または PAYG）。関連する AMI ID が同じライセンスタイプであることを確認します。 例：BYOL
AmiId	文字列	Threat Defense Virtual AMI ID（有効な Cisco Threat Defense Virtual AMI ID）。 タイプ：AWS::EC2::Image::Id リージョンとイメージの目的のバージョンに応じて、正しい AMI ID を選択してください。Auto Scale 機能は、バージョン 6.4+、BYOL/PAYG イメージをサポートします。いずれの場合も、AWS マーケットプレイスでライセンスに同意する必要があります。 BYOL の場合、設定 JSON ファイルの「licenseCaps」キーを「BASE」、「MALWARE」、「THREAT」、「URLFilter」などの機能で更新してください。
NoOfAZs	整数	Threat Defense Virtual を展開する必要がある可用性ゾーンの数（1 ー 3）。ALB 導入の場合、AWS で必要な最小値は 2 です。 例：2。

パラメータ	使用できる値/タイプ	説明
ListOfAzs	カンマ区切り文字列	ゾーンの順序のカンマ区切りリスト。 （注） ゾーンのリスト順は重要です。サブネットリストは同じ順序で指定する必要があります。 「*infrastructure.yaml*」ファイルを使用してインフラストラクチャを展開すると、スタックの出力セクションにこの値が設定されます。その値を使用してください。 例：us-east-1a、us-east-1b、us-east-1c
MgmtInterfaceSG	文字列	Threat Defense Virtual 管理インターフェイスのセキュリティグループ。 タイプ：List<AWS::EC2::SecurityGroup::Id> 「*infrastructure.yaml*」ファイルを使用してインフラストラクチャを展開すると、スタックの出力セクションにこの値が設定されます。その値を使用してください。
InsideInterfaceSG	文字列	Threat Defense Virtual 内部インターフェイスのセキュリティグループ。 タイプ：AWS::EC2::SecurityGroup::Id 「*infrastructure.yaml*」ファイルを使用してインフラストラクチャを展開すると、スタックの出力セクションにこの値が設定されます。その値を使用してください。
OutsideInterfaceSG	文字列	Threat Defense Virtual 外部インターフェイスのセキュリティグループ。 タイプ：AWS::EC2::SecurityGroup::Id 「*infrastructure.yaml*」ファイルを使用してインフラストラクチャを展開すると、スタックの出力セクションにこの値が設定されます。その値を使用してください。 例：sg-0c190a824b22d52bb

パラメータ	使用できる値/タイプ	説明
MgmtSubnetId	カンマ区切りリスト	管理サブネットIDのカンマ区切りリスト。リストは、対応する可用性ゾーンと同じ順序にする必要があります。 タイプ：List<AWS::EC2::SecurityGroup::Id> 「*infrastructure.yaml*」ファイルを使用してインフラストラクチャを展開すると、スタックの出力セクションにこの値が設定されます。その値を使用してください。
InsideSubnetId	カンマ区切りリスト	内部/Gig0/0サブネットIDのカンマ区切りリスト。リストは、対応する可用性ゾーンと同じ順序にする必要があります。 タイプ：List<AWS::EC2::SecurityGroup::Id> 「*infrastructure.yaml*」ファイルを使用してインフラストラクチャを展開すると、スタックの出力セクションにこの値が設定されます。その値を使用してください。
OutsideSubnetId	カンマ区切りリスト	外部/Gig0/1サブネットIDのカンマ区切りリスト。リストは、対応する可用性ゾーンと同じ順序にする必要があります。 タイプ：List<AWS::EC2::SecurityGroup::Id> 「*infrastructure.yaml*」ファイルを使用してインフラストラクチャを展開すると、スタックの出力セクションにこの値が設定されます。その値を使用してください。
KmsArn	文字列	既存の KMS の ARN（保存時に暗号化するための AWS KMS キー）。 指定した場合、Management Center と Threat Defense Virtual のパスワードを暗号化する必要があります。 パスワードの暗号化は、指定されたARNのみを使用して実行する必要があります。 暗号化パスワードの生成例： " aws kms encrypt --key-id <KMS ARN> --plaintext <password> " 次のような生成されたパスワードを使用してください。 例：arn:aws:kms:us-east-1:[AWS Account]:key/7d586a25-5875-43b1-bb68-a452e2f6468e

パラメータ	使用できる値/タイプ	説明
ngfwPassword	文字列	すべての Threat Defense Virtual インスタンスには、起動テンプレート（自動スケールグループ）の [ユーザーデータ（Userdata）] フィールドに入力されたデフォルトのパスワードが設定されています。 この入力により、Threat Defense Virtual にアクセスできるようになると、パスワードが新しく提供されたパスワードに変更されます。 KMS ARN が使用されていない場合は、プレーンテキストのパスワードを使用してください。KMS ARN が使用されている場合は、暗号化されたパスワードを使用する必要があります。 例：Cisco123789! または AQIAgcQFAGtz/hvaxMtJvY/x/rfHnI3lPpSXU
fmcServer	数値文字列	Lambda 関数と Threat Defense Virtual 管理インターフェイスの両方に到達可能な Management Center 管理用の IP アドレス。 例：10.10.17.21
fmcOperationsUsername	文字列	Management Center を管理する際に作成された Network-Admin 以上の特権ユーザー。ユーザとロールの作成の詳細については、『Cisco Secure Firewall Management Center デバイス構成ガイド』を参照してください。 例：apiuser-1
fmcOperationsPassword	文字列	KMS ARN が記載されていない場合は、プレーンテキストのパスワードを使用してください。記載されている場合は、暗号化されたパスワードを使用する必要があります。 例：Cisco123@ または AQICAHgcQAtz/hvaxMtJvY/x/mKI3clFPpSXUHQRnCAajB
fmcDeviceGrpName	文字列	Management Center のデバイスグループ名。 例：AWS-Cisco-NGFW-VMs-1

パラメータ	使用できる値/タイプ	説明
fmcPublishMetrics	ブール値	「TRUE」に設定すると、指定されたデバイスグループ内の登録済み Threat Defense Virtual センサーのメモリ消費量を取得するために、2 分に 1 回実行される Lambda 関数が作成されます。 使用可能な値：TRUE、FALSE 例：TRUE
fmcMetricsUsername	文字列	AWS CloudWatch にメトリックを公開するための一意の Management Center ユーザー名。ユーザとロールの作成の詳細については、『Cisco Secure Firewall Management Center デバイス構成ガイド』を参照してください。 「fmcPublishMetrics」が「FALSE」に設定されている場合は、この入力を行う必要はありません。 例：publisher-1
fmcMetricsPassword	文字列	AWS CloudWatch にメトリックを公開するための Management Center パスワード。KMS ARN が記載されていない場合は、プレーンテキストのパスワードを使用してください。記載されている場合は、暗号化されたパスワードを使用する必要があります。 「fmcPublishMetrics」が「FALSE」に設定されている場合は、この入力を行う必要はありません。 例：Cisco123789!
CpuThresholds	カンマ区切り整数	CPU しきい値の下限と CPU しきい値の上限。最小値は 0 で、最大値は 99 です。 デフォルト：10, 70 しきい値の下限はしきい値の上限よりも小さくする必要があります。 例：30,70

パラメータ	使用できる値/タイプ	説明
MemoryThresholds	カンマ区切り整数	MEM しきい値の下限と MEM しきい値の上限。最小値は 0 で、最大値は 99 です。 デフォルト：40, 70 しきい値の下限はしきい値の上限よりも小さくする必要があります。「fmcPublishMetrics」パラメータが「FALSE」の場合、影響はありません。 例：40,50

Management Center でのオブジェクト、デバイスグループ、NAT ルール、アクセスポリシーの設定

Threat Defense Virtual を管理するには、別のサーバー上で実行されるフル機能のマルチデバイスマネージャである Management Center を使用します。Threat Defense Virtual は、Threat Defense Virtual 仮想マシンに割り当てた管理インターフェイス上の Management Center を登録して通信します。詳細については、「Secure Firewall Management Center を備えた Secure Firewall Threat Defense Virtual について」を参照してください。

Threat Defense Virtual の設定に使用されるオブジェクトはすべて、ユーザーが作成する必要があります。

重要 デバイスグループを作成し、ルールを適用する必要があります。デバイスグループに適用されたすべての設定が Threat Defense Virtual インスタンスにプッシュされます。

オブジェクト

次のオブジェクトを作成します。

表 *5 : Threat Defense Virtual* 管理用の *Management Center* の設定オブジェクト

オブジェクト タイプ	名前	値
ホスト	aws-metadata-server	169.254.169.254
ポート	health-check-port	必要に応じて、8080 またはその他のポート
ゾーン	内部またはその他の名前	—
ゾーン	外部またはその他の名前	—

NAT ポリシー

一般的な NAT ルールでは、内部アドレスを外部インターフェイスの IP アドレスのポートに変換します。このタイプの NAT ルールのことをインターフェイス ポート アドレス変換（PAT）と呼びます。NAT ポリシーの詳細については、Secure Firewall Management Center を使用した Secure Firewall Threat Defense Virtual の管理のNAT の設定を参照してください。

NAT ポリシーには 1 つの必須ルールが必要です。

- 送信元ゾーン（Source Zone）：外部ゾーン
- 宛先ゾーン（Dest Zone）：内部ゾーン
- 元の送信元（Original-sources）：any-ipv4
- 元の送信元ポート（Original source port）：元/デフォルト
- 元の宛先（Original Destinations）：インターフェイス（Interface）
- 元の宛先ポート（Original-destination-port）：8080 またはユーザーが設定する正常性ポート
- 変換済み送信元（Translated-sources）：any-ipv4
- 変換済み送信元ポート（Translated source port）：元/デフォルト
- 変換済み宛先（Translated-destination）：aws-metadata-server
- 変換済み宛先ポート（Translated-destination-port）：80/HTTP

同様に、この設定が Threat Defense Virtual デバイスにプッシュされるように、データトラフィックの NAT ルールを追加できます。

重要 作成された NAT ポリシーは、デバイスグループに適用する必要があります。Lambda 関数からの Management Center 検証により、これが検証されます。

アクセス ポリシー

内部から外部へのトラフィックを許可するアクセス制御を設定します。必要なすべてのポリシーを含むアクセスポリシーを作成できます。このポートのトラフィックが到達できるように、正常性ポートオブジェクトを許可する必要があります。アクセスポリシーの詳細については、Secure Firewall Management Center を使用した Secure Firewall Threat Defense Virtual の管理のアクセス制御の設定を参照してください。

設定 JSON ファイルの更新

Configuration.json ファイルは、GitHub リポジトリから取得したアーカイブ Zip の一部である ***lambda_python_files*** フォルダにあります。JSON キーは変更しないでください。Threat Defense Virtual VM のスタティックルートは、JSON ファイルで設定する必要があります。

スタティックルートの設定例については、次を参照してください。

```
{
  "interface": "inside",
  "network": "any-ipv4",
  "gateway": "",
  "metric": "1"
}
```

JSON ファイルのすべての値は、デフォルトの Threat Defense Virtual パスワードを除き、要件に応じて変更できます。

Amazon Simple Storage Service（S3）へのファイルのアップロード

target ディレクトリ内のすべてのファイルを Amazon S3 バケットにアップロードする必要があります。必要に応じて、CLI を使用して、*target* ディレクトリ内のすべてのファイルを Amazon S3 バケットにアップロードできます。

```
$ cd ./target
$ aws s3 cp . s3://<bucket-name> --recursive
```

スタックの展開

展開のすべての前提条件が完了すると、AWS CloudFormation スタックを作成できます。

target ディレクトリ内の *deploy_ngfw_autoscale.yaml* ファイルを使用します。

target ディレクトリ内のファイルを使用します。

（注） *deploy_ngfw_autoscale_with_gwlb.yaml* ファイルを展開する前に、AWS GWLB 自動スケール ソリューション用に **infrastructure_gwlb.yaml** ファイルを展開する必要があります。

deploy_autoscale_with_gwlb.yaml テンプレートの展開時に作成される GWLB を選択して、ゲートウェイ ロードバランサー エンドポイント（GWLB-E）を作成する必要があります。GWLBe を作成したら、アプリケーションサブネットとデフォルトルートテーブルで GWLBe を使用するようにデフォルトルートを更新する必要があります。

詳細については、「https://docs.amazonaws.cn/en_us/vpc/latest/privatelink/create-endpoint-service-gwlbe.html」を参照してください。

入力パラメータ （30 ページ）で収集されたパラメータを入力します。

展開の検証

テンプレートの展開が成功したら、Lambda 関数と CloudWatch イベントが作成されていることを検証する必要があります。デフォルトでは、Auto Scale グループのインスタンスの最小数と最大数はゼロです。AWS EC2 コンソールで必要な数のインスタンスを使用して、Auto Scale グループを編集する必要があります。これにより、新しい Threat Defense Virtual インスタンスがトリガーされます。

1 つのインスタンスのみを起動してワークフローを確認し、そのインスタンスが期待どおりに動作しているかどうかを検証することを推奨します。その後に Threat Defense Virtual の実際の

要件を展開でき、動作を確認することもできます。AWS スケーリングポリシーによる削除を回避するために、最小数の Threat Defense Virtual インスタンスをスケールイン保護としてマークできます。

Auto Scale メンテナンスタスク

スケーリングプロセス

このトピックでは、Auto Scale グループの 1 つ以上のスケーリングプロセスを一時停止してから再開する方法について説明します。

スケールアクションの開始と停止

スケールアクションを開始および停止するには、次の手順を実行します。

- AWS 動的スケーリングの場合：スケールアウトアクションを有効化または無効化する方法については、次のリンクを参照してください。

  スケーリングプロセスの一時停止と再開

ヘルスモニター

60 分ごとに、CloudWatch Cron ジョブは、Health Doctor モジュールの Auto Scale Manager Lambda をトリガーします。

- 有効な Threat Defense Virtual VM に属する異常な IP がある場合、Threat Defense Virtual の展開時間が 1 時間を超えると、そのインスタンスは削除されます。
- それらの IP が有効な Threat Defense Virtual マシン の IP ではない場合、IP だけがターゲットグループから削除されます。

ヘルスモニターは、デバイスグループ、アクセスポリシー、および NAT ルールの Management Center 構成も検証します。IP やインスタンスが正常でない場合、または Management Center の検証が失敗した場合、ヘルスモニターはユーザーに電子メールを送信します。

ヘルスモニターの無効化

ヘルスモニターを無効にするには、***constant.py*** で constant を「True」に設定します。

ヘルスモニターの有効化

ヘルスモニターを有効にするには、***constant.py*** で固定値を「False」に設定します。

ライフサイクルフックの無効化

まれに、ライフサイクルフックを無効にする必要があります。無効にすると、インスタンスに追加のインターフェイスが追加されません。また、Threat Defense Virtual インスタンスの展開に連続して失敗することがあります。

Auto Scale Manager の無効化

Auto Scale Manager を無効化するには、それぞれの CloudWatch イベント「notify-instance-launch」と「notify-instance-terminate」を無効化する必要があります。これらのイベントを無効にしても、新しいイベントの Lambda はトリガーされません。ただし、すでに実行されている Lambda アクションは続行されます。Auto Scale Manager が突然停止することはありません。スタックの削除またはリソースの削除による突然の停止を試みると、不定状態になる可能性があります。

ロードバランサのターゲット

AWS ロードバランサでは、複数のネットワーク インターフェイスを持つインスタンスに対してインスタンスタイプのターゲットが許可されないため、Gigabit0/1 インターフェイス IP はターゲットグループのターゲットとして設定されます。ただし、現在のところ、AWS Auto Scale のヘルスチェックは、IP ではなく、インスタンスタイプのターゲットに対してのみ機能します。また、これらの IP はターゲットグループから自動的に追加されたり、削除されたりしません。したがって、Auto Scale ソリューションは、これら両方のタスクをプログラムで処理します。ただし、メンテナンスやトラブルシューティングの場合は、手動で実行する必要があることがあります。

ターゲットグループへのターゲットの登録

Threat Defense Virtual インスタンスをロードバランサに登録するには、Gigabit0/1 インスタンス IP（外部サブネット）をターゲットとしてターゲットグループに追加する必要があります。「IP アドレスによるターゲットの登録または登録解除」を参照してください。

ターゲットグループからのターゲットの登録解除

ロードバランサに対する Threat Defense Virtual インスタンスの登録を解除するには、Gigabit0/1 インスタンス IP（外部サブネット）をターゲットグループのターゲットとして削除する必要があります。「IP アドレスによるターゲットの登録または登録解除」を参照してください。

インスタンスのスタンバイ

AWS では、Auto Scale グループでのインスタンスの再起動は許可されませんが、ユーザーはインスタンスをスタンバイ状態にして再起動アクションを実行できます。これは、ロードバランサのターゲットがインスタンスタイプの場合に最も機能しますが、Threat Defense Virtual マシンは、複数のネットワーク インターフェイスがあるため、インスタンスタイプのターゲットとして設定できません。

インスタンスをスタンバイ状態にする

インスタンスがスタンバイ状態になると、正常性プローブが失敗するまで、ターゲットグループ内のそのインスタンスの IP は同じ状態のままになります。このため、インスタンスをスタンバイ状態にする前に、ターゲットグループからそれぞれの IP を登録解除することをお勧めします。詳細については、ターゲットグループからのターゲットの登録解除 （42 ページ）を参照してください。

IP が削除されたら、「Auto Scaling グループからのインスタンスの一時的な削除」を参照してください。

スタンバイ状態からのインスタンスの削除

同様に、インスタンスをスタンバイ状態から実行状態に移行できます。スタンバイ状態から削除すると、インスタンスの IP がターゲットグループのターゲットに登録されます。「ターゲットグループへのターゲットの登録 （42 ページ）」を参照してください。

トラブルシューティングやメンテナンスのためにインスタンスをスタンバイ状態にする方法の詳細については、AWS News Blog を参照してください。

Auto Scale グループからのインスタンスの削除または分離

Auto Scale グループからインスタンスを削除するには、まずインスタンスをスタンバイ状態に移行する必要があります。「インスタンスをスタンバイ状態にする」を参照してください。スタンバイ状態になったインスタンスは、削除または分離できます。「Auto Scaling グループから EC2 インスタンスをデタッチする」を参照してください。

Management Center 側に変更はありません。必要な変更は手動で実行する必要があります。

インスタンスで終了

インスタンスを終了するには、スタンバイ状態にする必要があります。インスタンスのスタンバイ （42 ページ）を参照してください。インスタンスがスタンバイ状態になったら、終了できます。

インスタンスのスケールイン保護

Auto Scale グループから特定のインスタンスが誤って削除されないようにするために、そのインスタンスをスケールイン保護として作成できます。インスタンスがスケールイン保護されている場合、スケールインイベントが原因で終了することはありません。

インスタンスをスケールイン保護状態にするには、次のリンクを参照してください。

https://docs.aws.amazon.com/autoscaling/ec2/userguide/as-instance-termination.html

重要 正常（EC2インスタンスだけでなく、ターゲット IP が正常）なインスタンスの最小数をスケールイン保護として設定することをお勧めします。

ログイン情報と登録 ID の変更

設定の変更は、すでに実行中のインスタンスには自動的に反映されません。変更は新しいデバイスにのみ反映されます。このような変更は、既存のデバイスに手動でプッシュする必要があります。

Management Center のユーザー名とパスワードの変更

Management Center の IP、ユーザー名、またはパスワードを変更する場合は、Auto Scale Manager Lambda 関数とカスタム指標パブリッシャ Lambda 関数の環境変数でそれぞれの変更を実行する必要があります。「AWS Lambda 環境変数の使用」を参照してください。

Lambda の次回実行時に、変更された環境変数が参照されます。

（注） 環境変数は Lambda 関数に直接渡されます。パスワードの複雑さはチェックされません。

Threat Defense Virtual の管理者パスワードを変更します。

Threat Defense Virtual パスワードを変更すると、インスタンスを実行するために各デバイスでパスワードを手動で変更する必要があります。新しい Threat Defense Virtual デバイスをオンボードする場合、Threat Defense Virtual パスワードは Lambda 環境変数から取得されます。「AWS Lambda 環境変数の使用」を参照してください。

登録 ID と NAT ID の変更

新しい Threat Defense Virtual デバイスを異なる登録 ID と NAT ID でオンボードする場合、Management Center 登録のために、Configuration.json ファイルでこの情報を変更する必要があります。Configuration.json ファイルは、[Lambda] リソースページにあります。

アクセスポリシーと NAT ポリシーの変更

アクセスポリシーまたは NAT ポリシーへの変更は、デバイスグループの割り当てにより、今後のインスタンスに自動的に適用されます。ただし、既存の Threat Defense Virtual インスタンスを更新するには、設定変更を手動でプッシュして、Management Center から展開する必要があります。

AWS リソースに対する変更

AWS の導入後、Auto Scale グループ、起動設定、CloudWatch イベント、スケーリングポリシーなど、多くの項目を変更できます。CloudFormation スタックにリソースをインポートするか、既存のリソースから新しいスタックを作成できます。

AWS リソースで実行される変更を管理する方法の詳細については、「既存リソースの CloudFormation 管理への取り込み」を参照してください。

CloudWatch ログの収集および分析

CloudWatch ログをエクスポートするには、「AWS CLI を使用した Amazon S3 へのログデータのエクスポート」を参照してください。

Auto Scale のトラブルシューティングとデバッグ

AWS CloudFormation コンソール

AWS CloudFormation コンソールで CloudFormation スタックへの入力パラメータを確認できます。これにより、Web ブラウザからスタックを直接作成、監視、更新、削除できます。

目的のスタックに移動し、[パラメータ（parameter）] タブを確認します。[Lambda関数環境変数（Lambda Functions environment variables）] タブで Lambda 関数への入力を確認することもできます。*configuration.json* ファイルは、Auto Scale Manager Lambda 関数自体でも表示できます。

AWS CloudFormation コンソールの詳細については、『AWS CloudFormation ユーザーガイド（AWS CloudFormation User Guide）』を参照してください。

Amazon CloudWatch ログ

個々の Lambda 関数のログを表示できます。AWS Lambda はお客様の代わりに Lambda 関数を自動的に監視し、Amazon CloudWatch を通じてメトリックを報告します。関数の障害のトラブルシューティングに役立つように、Lambda は関数によって処理されたすべての要求をログに記録し、Amazon CloudWatch ログを通じてコードによって生成されたログも自動的に保存します。

Lambda コンソール、CloudWatch コンソール、AWS CLI、または CloudWatch API を使用して、Lambda のログを表示できます。ロググループと CloudWatch コンソールを介したロググループへのアクセスの詳細については、『*Amazon CloudWatch* ユーザーガイド（*Amazon CloudWatch User Guide*）』でモニターリングシステム、アプリケーション、およびカスタムログファイルについて参照してください。

ロードバランサのヘルスチェックの失敗

ロードバランサのヘルスチェックには、プロトコル、ping ポート、ping パス、応答タイムアウト、ヘルスチェック間隔などの情報が含まれます。ヘルスチェック間隔内に 200 応答コードを返す場合、インスタンスは正常と見なされます。

一部またはすべてのインスタンスの現在の状態が `OutOfService` であり、説明フィールドに「インスタンスがヘルスチェックの異常しきい値の数以上連続して失敗しました（Instance has failed at least the Unhealthy Threshold number of health checks consecutively）」というメッセージが表示された場合、インスタンスはロードバランサのヘルスチェックに失敗しています。

Management Center 構成の正常性プローブ NAT ルールを確認する必要があります。詳細については、『Troubleshoot a Classic Load Balancer: Health checks』を参照してください。

トラフィックの問題

Threat Defense Virtual インスタンスのトラフィックの問題をトラブルシューティングするには、ロードバランサルール、NAT ルール、および Threat Defense Virtual インスタンスで設定されているスタティックルートを確認する必要があります。

セキュリティグループのルールなど、展開テンプレートで提供される AWS 仮想ネットワーク/サブネット/ゲートウェイの詳細も確認する必要があります。たとえば、「EC2 インスタンス

のトラブルシューティング（Troubleshooting EC2 instances）」https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-instance-troubleshoot.htmlなど、AWS のドキュメントを参照することもできます。

Management Center への接続に失敗

管理接続が中断された場合は、設定とログイン情報を確認する必要があります。『*Firepower Management Center Configuration Guide*』の「Requirements and Prerequisites for Device Management」を参照してください。

デバイスが FMC への登録に失敗 Management Center

デバイスが Management Center に登録できない場合は、Management Center 構成に障害があるか到達不能であるか、または Management Center に新しいデバイスを収容するキャパシティがあるかどうかを判断する必要があります。『*Firepower Management Center Configuration Guide*』の「Add a Device to the FMC」を参照してください。

Threat Defense Virtual に SSH 接続できない

Threat Defense Virtual に SSH 接続できない場合は、テンプレートを介して複雑なパスワードが Threat Defense Virtual に渡されたかどうかを確認します。

Threat Defense Virtualイメージスナップショット

AWS ポータルで Amazon Machine Image（AMI）スナップショットを使用して Threat Defense Virtual を作成および展開できます。イメージスナップショットは、状態データのない、複製された Threat Defense Virtual イメージインスタンスです。

Threat Defense Virtualスナップショットの概要

Threat Defense Virtual インスタンスのスナップショットイメージを作成するプロセスは、Threat Defense Virtual および FSIC に対して実行される最初のブート手順をスキップすることにより、初期システムの初期化時間を最小限に抑えるのに役立ちます。スナップショットイメージは、事前に入力されたデータベースと Threat Defense Virtual 初期ブートプロセスで構成されます。これにより、イメージは Management Center またはその他の管理センターのシステム ID に関連する一意の ID（UUID、シリアル番号）を再生成できます。このプロセスは、自動スケール展開に不可欠な Threat Defense Virtual の起動時間を短縮するのに役立ちます。

Threat Defense Virtual スナップショット AMI の作成

Threat Defense Virtual のイメージスナップショットの作成は、既存の Threat Defense Virtual イメージを複製して、Azure ポータルで Threat Defense Virtual のプレーンインスタンスを作成するプロセスです。

始める前に

- Threat Defense Virtual バージョン 7.2 以降を展開している必要があります。Threat Defense Virtual の展開については、「AWS での Threat Defense Virtual の展開 （1 ページ）」を参照してください。
- イメージスナップショットの準備をしている Threat Defense Virtual インスタンスを Management Center Virtual や Device Manager などのマネージャに登録しないでください。

ステップ 1 Threat Defense Virtual インスタンスを展開した AWS コンソールに移動します。

（注） イメージスナップショットとして複製する予定の Threat Defense Virtual インスタンスが Management Center に登録されていないこと、または他のローカルマネージャに設定されたり設定が適用されたりしていないことを確認します。

ステップ 2 次のスクリプトを使用して、エキスパートシェルからプレスナップショット プロセスを実行します。

```
> expert
admin@FTDvbaseimg:~$ Sudo su
root@firepower:/ngfw/var/common# prepare_snapshot
Do you want to continue [Y/N]:
```

スクリプトで `prepare_snapshot` コマンドを使用すると、スクリプトの実行の確認を求める中間メッセージが表示されます。スクリプトを実行するには、[Y] を押します。

または、`root@firepower:/ngfw/var/common# prepare_snapshot -f` のように、このコマンドに `-f` を追加して、ユーザーの確認メッセージをスキップしてスクリプトを直接実行することもできます。

このスクリプトは、Threat Defense Virtual インスタンスに関連付けられたすべての回線設定、展開されたポリシー、設定されたマネージャ、UUID を削除します。処理が完了すると、Threat Defense Virtual インスタンスはシャットダウンされます。Threat Defense Virtual インスタンスは、AWS ポータルの [インスタンス（Instances）] ページに一覧表示されます。

ステップ 3 http://aws.amazon.com/ にログインし、地域を選択します。

AWS は互いに分かれた複数の地域に分割されています。地域は、ウィンドウの右上隅に表示されます。ある地域内のリソースは、別の地域には表示されません。目的の地域に属していることを定期的に確認してください。

次のタスク

スナップショット AMI を使用して Threat Defense Virtual インスタンスを展開します。参照 スナップショット AMI を使用した Threat Defense Virtual インスタンスの展開 （48 ページ）

（注） Threat Defense Virtual コンソールから CLI コマンド **show version** および **show snapshot detail** を実行すると、作成した Threat Defense Virtual のイメージスナップショットのバージョンと詳細を確認できます。

スナップショット AMI を使用した Threat Defense Virtual インスタンスの展開

始める前に

次のことを推奨します。

- AWS 環境の設定 （13 ページ）の説明に従って、AWS VPC および EC2 のエレメントを設定します。
- AMI が Threat Defense Virtual インスタンスで使用できることを確認します。

ステップ 1 https://aws.amazon.com/marketplace（Amazon マーケットプレイス）に移動してサインインします。

ステップ 2 **[EC2ダッシュボード（EC2 Dashboard）]** > **[インスタンス（Instances）]**の順にクリックします。イメージのスナップショットを作成するために展開した Threat Defense Virtual インスタンスが [インスタンス（Instances）] ページに表示されます。

（注） イメージのスナップショットを作成するには、操作ステータス（[インスタンス状態（Instance Status）]）が [停止（Stopped）] の Threat Defense Virtual インスタンスを常に選択する必要があります。

ステップ 3 [インスタンス（Instances）]ページで、対応する[インスタンス状態（Instance Status）]が[停止（Stopped）]と示されている Threat Defense Virtual インスタンスを特定して選択します。

ステップ 4 [アクション（Actions）] ドロップダウンメニューから、[イメージとテンプレート（Image and templates）]をポイントし、[イメージの作成（Create Image）] をクリックします。

ステップ 5 [イメージの作成（Create Image）] ページで、イメージのスナップショットの名前と説明を入力します。

ステップ 6 [再起動なし（No reboot）]セクションの下にある[有効化（Enable）]チェックボックスをオンにします。

ステップ 7 [Create Image] をクリックします。Threat Defense Virtual のイメージスナップショット AMI が作成されます。

ステップ 8 **[イメージ（Images）]**>**[AMI（AMIs）]**の順にクリックします。このページでは、新しく作成したイメージのスナップショット AMI を表示できます。

ステップ 9 イメージスナップショット AMI を選択します。

ステップ 10 [起動（Launch）] をクリックして、イメージスナップショット AMI を使用して新しい Threat Defense Virtual インスタンスを展開します。

ステップ 11 Threat Defense Virtual インスタンスの展開を続行します。Threat Defense Virtual インスタンスの展開 （18 ページ）またはAuto Scale の展開 （30 ページ）を参照してください。

Amazon GuardDuty サービスについて

Amazon GuardDuty は AWS 環境において、VPC ログ、CloudTrail 管理イベントログ、CloudTrail S3 データイベントログ、DNS ログといったさまざまなソースからのデータを処理して、不正の可能性がある悪意のあるアクティビティを特定する監視サービスです。

Secure Firewall Threat Defense Virtual と GuardDuty の統合について

シスコでは、管理センターとデバイスマネージャを介して Amazon GuardDuty サービスと Secure Firewall Threat Defense Virtual を統合するソリューションを提供しています。

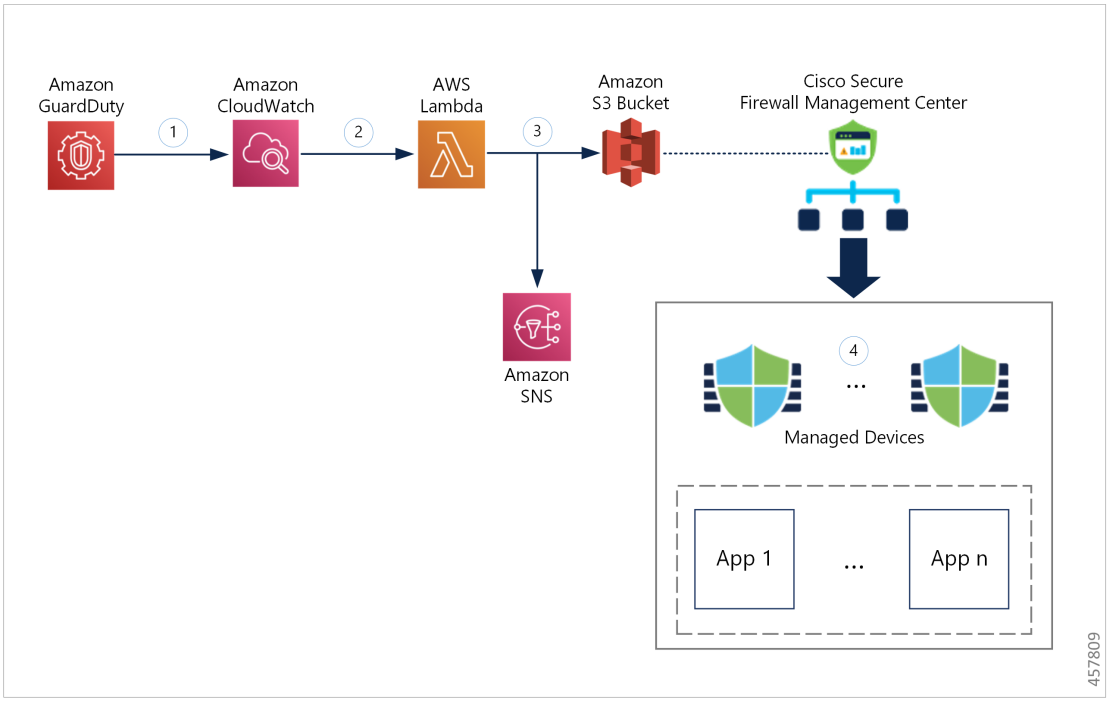
このソリューションでは、Amazon GuardDuty から受け取った脅威分析データや検出結果（脅威、攻撃などを生成する悪意のある IP）を使用して、その情報（悪意のある IP）をマネージャ（Secure Firewall Management Center Virtual および Secure Firewall デバイスマネージャ）経由で Secure Firewall Threat Defense Virtual にフィードし、これらのソース（悪意のある IP）が発生源となる将来の脅威から基盤となるネットワークやアプリケーションを保護します。

動作の仕組み

次の統合ソリューションとワークフローの図は、Amazon GuardDuty の Secure Firewall Threat Defense Virtual との統合を理解するのに役立ちます。

セキュリティ インテリジェンス ネットワーク フィードを使用した Secure Firewall Management Center Virtual との統合ソリューション

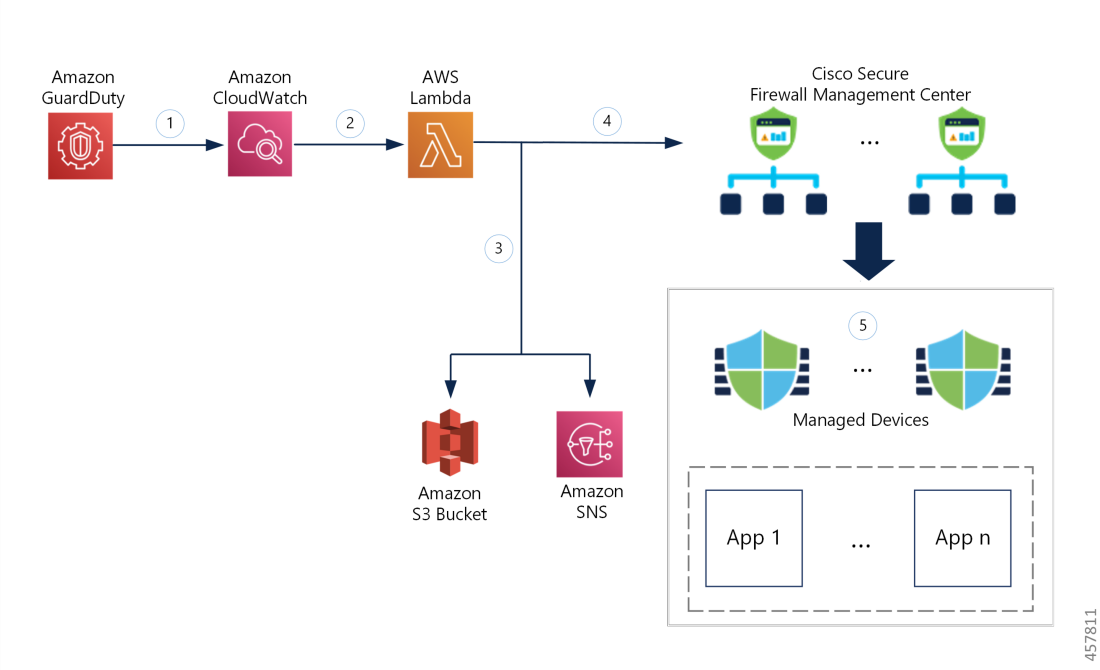
次のワークフロー図は、セキュリティインテリジェンス ネットワーク フィード URL を使用した Secure Firewall Management Center Virtual と Amazon GuardDuty の統合ソリューションを示しています。



1	GuardDuty サービスは、悪意のあるアクティビティを検出すると、脅威の検出結果を CloudWatch に送信します。
2	CloudWatch イベントにより、AWS Lambda 関数がアクティブ化されます。
3	Lambda 関数は、S3 バケットのレポートファイル内に記載された悪意のあるホストを更新し、SNS 経由で通知を送信します。
4	Secure Firewall Management Center のアクセス コントロール ポリシーは、設定されたアクションに基づいてトラフィックを処理するように対象デバイスに指示します。たとえば、GuardDuty によって報告された悪意のあるホストからのトラフィックをブロックします。 このアクセスポリシーでは、セキュリティ インテリジェンス ネットワーク フィードが、Lambda 関数によって提供された悪意のある IP アドレスレポートファイルの S3 オブジェクト URL と共に使用されます。

ネットワーク オブジェクト グループを使用した Secure Firewall Management Center Virtual との統合ソリューション

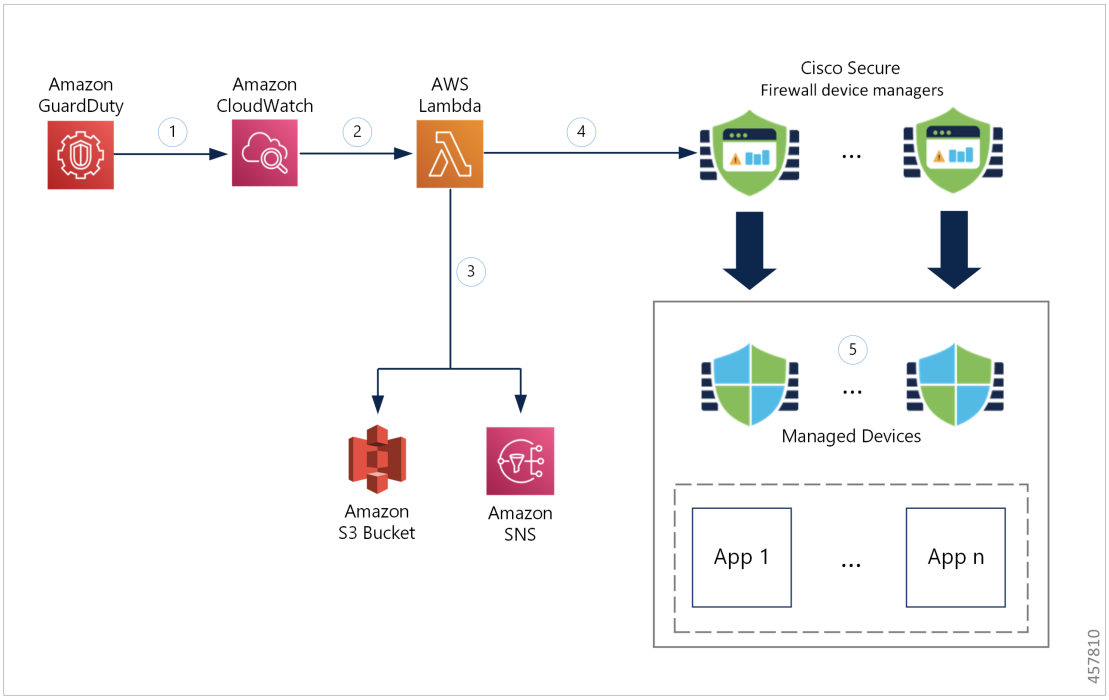
次のワークフロー図は、ネットワーク オブジェクト グループを使用した Secure Firewall Management Center Virtual と Amazon GuardDuty の統合ソリューションを示しています。



1	GuardDuty サービスは、悪意のあるアクティビティを検出すると、脅威の検出結果を CloudWatch に送信します。
2	CloudWatch イベントにより、AWS Lambda 関数がアクティブ化されます。
3	Lambda 関数は、S3 バケットのレポートファイル内に記載された悪意のあるホストを更新し、SNS 経由で通知を送信します。
4	Lambda 関数は、悪意のあるホスト IP アドレスを追加して Secure Firewall Management Center Virtual のネットワーク オブジェクト グループを設定または更新します。
5	Secure Firewall Management Center のアクセス コントロール ポリシーは、設定されたアクションに基づいてトラフィックを処理するように対象デバイスに指示します。たとえば、GuardDuty によって報告された悪意のあるホストからのトラフィックをブロックします。 このアクセス コントロール ポリシーは、Lambda 関数によって検出された悪意のある IP アドレスが追加されたネットワーク オブジェクト グループを使用します。

ネットワーク オブジェクト グループを使用した Secure Firewall Device Manager との統合ソリューション

次のワークフロー図は、ネットワーク オブジェクト グループを使用した Secure Firewall Device Manager と Amazon GuardDuty の統合ソリューションを示しています。



1	GuardDuty サービスは、悪意のあるアクティビティを検出すると、脅威の検出結果を CloudWatch に送信します。
2	CloudWatch イベントにより、AWS Lambda 関数がアクティブ化されます。
3	Lambda 関数は、S3 バケットのレポートファイル内に記載された悪意のあるホストを更新し、SNS 経由で通知を送信します。
4	Lambda 関数は、悪意のあるホスト IP アドレスを追加して Secure Firewall Device Manager のネットワーク オブジェクト グループを設定または更新します。
5	Secure Firewall Device Manager のアクセス コントロール ポリシーは、設定されたアクションに基づいてトラフィックを処理するように管理対象デバイスに指示します。たとえば、GuardDuty によって報告された悪意のあるホストからのトラフィックをブロックします。 このアクセス コントロール ポリシーは、Lambda 関数によって検出された悪意のある IP アドレスが追加されたネットワーク オブジェクト グループを使用します。

この統合の主要コンポーネント

コンポーネント	説明
Amazon GuardDuty	特定のリージョン（EC2、S3、IAM など）のさまざまな AWS リソースについて、脅威検出結果の生成を行う Amazon サービス。

Amazon Simple Storage Service（S3）	ソリューションに関連するさまざまなアーティファクトを保存するために使用される Amazon サービスは以下のとおりです。 • Lambda 関数の zip ファイル • Lambda レイヤの zip ファイル • Cisco Secure Firewall Management Center Secure Firewall と Device Manager 構成の入力ファイル（.ini） • Lambda 関数によって報告された悪意のある IP アドレスのリストが保存された出力レポートファイル（.txt）
Amazon CloudWatch	Amazon サービスは次の目的で使用されます。 • GuardDuty サービスで報告された検出結果についてモニタリングし、Lambda 関数をトリガーして検出結果を処理します。 • CloudWatch ロググループで Lambda 関数に関連するアクティビティをロギングします。
Amazon Simple Notification Service（SNS）	電子メール通知をプッシュするために使用される Amazon サービスです。この電子メール通知には、次の内容が含まれます。 • Lambda 関数によって正常に処理された GuardDuty 検出結果の詳細。 • Lambda 関数によって Cisco Secure Firewall Manager で実行された更新の詳細。 • Lambda 関数によって発生した重大なエラー。
AWS Lambda 関数	AWS サーバーレス コンピューティング サービスはイベントに応じてコードを実行し、基盤となるコンピューティングリソースを自動的に管理します。CloudWatch イベントルールが GuardDuty の検出結果に基づいて Lambda 関数をトリガーします。Lambda 関数はこの連携で以下を実行します。 • GuardDuty の検出結果を処理して、重大度、接続方向、悪意のある IP アドレスの存在など、必要なすべての基準が満たされていることを確認します。 • （設定に応じて）悪意のある IP アドレスを追加して、Cisco Secure Firewall Manager のネットワーク オブジェクト グループを更新します。 • S3 バケットのレポートファイルで悪意のある IP アドレスを更新します。 • Cisco Secure Firewall の管理者に対して、さまざまなマネージャの更新やエラーについて通知します。

CloudFormation テンプレート	AWS での連携に必要なさまざまなリソースを展開するために使用されます。 CloudFormation テンプレートには、次のリソースが含まれています。 • **AWS::SNS::Topic**：電子メール通知をプッシュするための SNS トピック。 • **AWS::Lambda::Function, AWS::Lambda::LayerVersion**：Lambda 関数とレイヤファイル。 • **AWS::Events::Rule**：GuardDuty の検出結果イベントに基づいて Lambda 関数をトリガーする CloudWatch イベントルール。 • **AWS::Lambda::Permission**：Lambda 関数をトリガーする CloudWatch イベントルールのアクセス許可。 • **AWS::IAM::Role、AWS::IAM::Policy**：各種 AWS リソースの Lambda 関数へのさまざまなアクセス許可を付与する IAM ロールとポリシーリソース。 このテンプレートは、展開をカスタマイズするためのユーザー入力を取り込みます。

サポートされるソフトウェア プラットフォーム

- GuardDuty 統合ソリューションは、Secure Firewall Management Center Virtual または Secure Firewall Device Manager によって管理される Secure Firewall Threat Defense Virtual に適用できます。
- Lambda 関数は、管理センターのネットワーク オブジェクト グループと、任意の仮想プラットフォームに展開されたデバイスマネージャを更新できます。Lambda 関数がパブリック IP アドレスを介してこれらのマネージャに接続できることを確認してください。

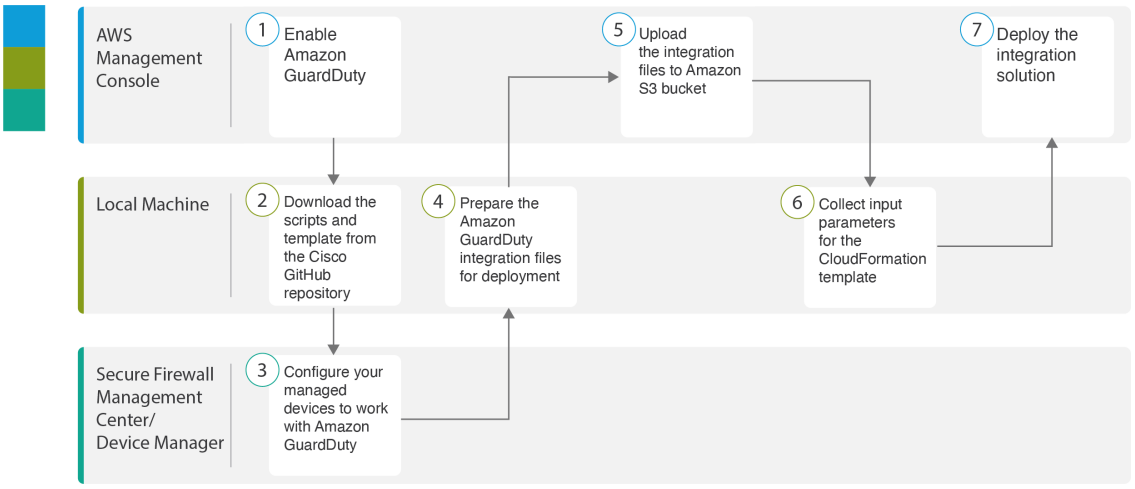
注意事項と制約事項

- Lambda 関数は、悪意のある IP アドレスを追加した Cisco Secure Firewall マネージャのネットワーク オブジェクト グループの更新のみを実行します。したがって、これらの更新または変更を管理対象デバイスに展開する必要があります。
- この統合で使用される AWS のサービスはリージョン固有です。したがって、異なるリージョンの GuardDuty 検出結果を使用する場合は、リージョン固有のインスタンスを展開する必要があります。
- Lambda 関数は、REST API を介して Cisco Secure Firewall マネージャを更新します。したがって、他の方法やマネージャ（Cisco Defense Orchestrator など）を使用することはできません。

- パスワードベースのログインのみを使用できます。他の認証方式はサポートされていません。
- 入力ファイルで暗号化されたパスワードを使用している場合は、次の点に注意してください。
  - 対称 KMS キーを使用した暗号化のみがサポートされます。
  - すべてのパスワードは、Lambda 関数にアクセス可能な単一の KMS キーを使用して暗号化する必要があります。

Amazon GuardDuty と Secure Firewall Threat Defense の統合方法

次のタスクを実行して、Amazon GuardDuty と Secure Firewall Threat Defense を統合します。



	ワークスペース	手順
1	AWS 管理コンソール	AWS での Amazon GuardDuty サービスの有効化 （56 ページ）
2	Local Machine	Secure Firewall Threat Defense Virtual および Amazon GuardDuty 統合ソリューションリポジトリのダウンロード （56 ページ）
3	Secure Firewall Management CenterまたはSecure Firewall Device Manager	Amazon GuardDuty と連携するための管理対象デバイスの設定 （57 ページ）
4	Local Machine	展開に向けた Amazon GuardDuty リソースファイルの準備 （61 ページ）

	ワークスペース	手順
5	AWS 管理コンソール	Amazon Simple Storage Service へのファイルのアップロード （64 ページ）
6	Local Machine	CloudFormation テンプレートの入力パラメータの収集 （65 ページ）
7	AWS 管理コンソール	スタックの展開 （67 ページ）

AWS での Amazon GuardDuty サービスの有効化

ここでは、AWS で Amazon GuardDuty サービスを有効にする方法について説明します。

始める前に

すべての AWS リソースが同じリージョンにあることを確認します。

ステップ 1 https://aws.amazon.com/marketplace（Amazon マーケットプレイス）に移動してサインインします。

ステップ 2 [サービス（Services）] > [GuardDuty] を選択します。

ステップ 3 [GuardDuty] ページで [利用を開始する（Get Started）] をクリックします。

ステップ 4 [GuardDutyの有効化（Enable GuardDuty）] をクリックして、Amazon GuardDuty サービスを有効にします。

GuardDuty の有効化の詳細については、AWS ドキュメントの『Getting started with GuardDuty』[英語] を参照してください。

次のタスク

Cisco GitHub リポジトリから Amazon GuardDuty ソリューションファイル（テンプレートとスクリプト）をダウンロードします。Secure Firewall Threat Defense Virtual および Amazon GuardDuty 統合ソリューションリポジトリのダウンロード （56 ページ） を参照してください。

Secure Firewall Threat Defense Virtual および Amazon GuardDuty 統合ソリューションリポジトリのダウンロード

Amazon GuardDuty ソリューションに必要なファイルをダウンロードします。Secure Firewall Threat Defense Virtual の該当するバージョン用の導入スクリプトとテンプレートは、次の Cisco GitHub リポジトリから入手できます。

https://github.com/CiscoDevNet/cisco-ftdv

以下は、Cisco GitHub リポジトリリソースのリストです。

ファイル	説明
READ.MD	ReadMe ファイル
configuration/	Secure Firewall Threat Defense Virtual マネージャの構成ファイルテンプレート。
images/	Secure Firewall Threat Defense Virtual および Amazon GuardDuty 統合ソリューションの図が格納されています。
lambda/	Lambda 関数の Python ファイル。
templates/	導入用の CloudFormation テンプレート

Amazon GuardDuty と連携するための管理対象デバイスの設定

Lambda 関数は Amazon GuardDuty の検出結果を処理し、CloudWatch イベントをトリガーした悪意のある IP アドレスを特定します。Secure Firewall Threat Defense Virtual は 次のいずれかの方法で Secure Firewall Management Center Virtual および Secure Firewall Device Manager を介してこの脅威データを受信します。

- **ネットワーク オブジェクト グループの更新**：Lambda 関数は、悪意のある IP アドレスを追加してマネージャのネットワーク オブジェクト グループを更新します。次に、このネットワーク オブジェクト グループを使用してトラフィックを処理するアクセス コントロール ポリシーを設定できます。この方法は Secure Firewall Management Center Virtual と Secure Firewall Device Manager が対象です。
- **セキュリティ インテリジェンス ネットワーク フィード**：Lambda 関数は、悪意のある IP アドレスを追加して Amazon S3 バケット内のレポートファイルを作成または更新します。レポートファイルの URL を使用してセキュリティ インテリジェンス フィードを設定し、このフィードを使用してトラフィックを処理するアクセス コントロール ポリシーを設定できます。この方法は Secure Firewall Management Center Virtual のみが対象です。

レポートファイルの URL を使用したセキュリティ インテリジェンス ネットワーク フィードの設定

ここでは、Secure Firewall Management Center Virtual でセキュリティ インテリジェンス ネットワーク フィードを設定する方法について説明します。

始める前に

- Secure Firewall Management Center Virtual で脅威ライセンスが有効になっていることを確認します。「脅威ライセンス」を参照してください。
- Amazon S3 バケットで使用可能なレポートファイルの URL を作成して書き留めておきます。

- Secure Firewall Management Center Virtual から Amazon S3 バケット内のレポートファイルにアクセスできることを確認します。

ステップ 1 Secure Firewall Management Center Virtual にログインします。

ステップ 2 Amazon S3 バケットのレポートファイル URL を使用して、セキュリティ インテリジェンス ネットワーク フィードを作成します。セキュリティ インテリジェンス ネットワーク フィードを手動で作成する方法については、「カスタム セキュリティ インテリジェンス フィード」を参照してください。

ステップ 3 トラフィックを処理するセキュリティ インテリジェンス ネットワーク フィード URL を使用して、アクセス コントロール ポリシーやアクセス制御ルールを作成または更新します。「手動 URL フィルタリングオプション」および「アクセス コントロール ルールの作成と編集」を参照してください。

（注） 展開の前または後に、セキュリティ インテリジェンス ネットワーク フィードを作成し、アクセス コントロール ポリシーの URL を更新できます。Amazon S3 バケットに出力レポートファイルを作成している場合は、展開前にセキュリティ インテリジェンス ネットワーク フィードを作成できます。展開後にセキュリティ インテリジェンス ネットワーク フィードを作成している場合は、Amazon GuardDuty から最初の検出結果の電子メール通知を受信するまで待ち、その電子メール通知で指定された URL を使用してセキュリティ インテリジェンス ネットワーク フィードを設定します。

ステップ 4 Secure Firewall Management Center Virtual に設定の変更を展開します。「設定変更の展開」を参照してください。

次のタスク

展開に向けて Amazon GuardDuty ソースファイルを準備します。展開に向けた Amazon GuardDuty リソースファイルの準備 （61 ページ） を参照してください。

ネットワーク オブジェクト グループの作成

Secure Firewall Management Center Virtual および Secure Firewall デバイスマネージャ で Lambda 関数のネットワーク オブジェクト グループを設定または作成して、Amazon GuardDuty によって検出された悪意のある IP アドレスを更新する必要があります。

Lambda 関数でネットワーク オブジェクト グループを設定しない場合、デフォルト名 **aws-gd-suspicious-hosts** のネットワーク オブジェクト グループが Lambda 関数によって作成され、悪意のある IP アドレスが更新されます。

Secure Firewall Management Center Virtual でのネットワーク オブジェクト グループの作成

ここでは、Secure Firewall Management Center Virtual でネットワーク オブジェクト グループを作成する方法について説明します。

ステップ 1 Secure Firewall Management Center Virtual にログインします。

ステップ 2 ダミーの IP アドレスを使用してネットワーク オブジェクト グループを作成します。「ネットワークオブジェクト」を参照してください。

ステップ 3 ネットワーク オブジェクト グループを使用してトラフィックを処理するためのアクセス コントロール ポリシーやアクセス制御ルールを作成または更新します。「アクセス コントロール ポリシーの管理」および「アクセス コントロール ルールの作成および編集」を参照してください。

ヒント Lambda 関数が悪意のある IP アドレスを追加してネットワーク オブジェクト グループを更新していることを確認した後に、アクセスコントロールポリシーやアクセス制御ルールを作成または更新することもできます。

ステップ 4 設定変更を管理対象デバイスに展開します。「設定変更の展開」を参照してください。

次のタスク

展開に向けて Amazon GuardDuty ソースファイルを準備します。展開に向けた Amazon GuardDuty リソースファイルの準備 （61 ページ） を参照してください。

Secure Firewall Device Manager のネットワーク オブジェクト グループの作成

ここでは、Secure Firewall デバイスマネージャ でネットワーク オブジェクト グループを作成する方法について説明します。

ステップ 1 Secure Firewall Device Manager にログインします。

ステップ 2 ダミーの IP アドレスを使用してネットワーク オブジェクト グループを作成します。「ネットワークオブジェクトとグループの設定」を参照してください。

ステップ 3 ネットワーク オブジェクト グループを使用してトラフィックを処理するためのアクセス コントロール ポリシーやアクセス制御ルールを作成または更新します。「アクセス コントロール ポリシーの設定」および「アクセス制御ルールの設定」を参照してください。

ヒント Lambda 関数が悪意のある IP アドレスを追加してネットワーク オブジェクト グループを更新していることを確認した後に、アクセスコントロールポリシーやアクセス制御ルールを作成または更新することもできます。

ステップ 4 設定変更を管理対象デバイスに展開します。「変更の展開」を参照してください。

次のタスク

展開に向けて Amazon GuardDuty ソースファイルを準備します。展開に向けた Amazon GuardDuty リソースファイルの準備 （61 ページ） を参照してください。

Secure Firewall Management Center Virtual で Lambda 関数を利用するためのユーザーアカウントの作成

Lambda 関数には、管理センターとデバイスマネージャでネットワーク オブジェクト グループを更新するための管理者権限を持つユーザーアカウントが必要です。したがって、管理センターとデバイスマネージャで管理者権限を持つ排他的なユーザーアカウントを作成する必要があります。ユーザーアカウントの作成は、ネットワーク オブジェクト グループの更新メソッドを使用する場合にのみ必要です。

ユーザーアカウントの作成の詳細については、以下を参照してください。

- FDM および FTD ユーザ アクセスの管理
- FMC のユーザーアカウント

（任意）パスワードの暗号化

必要に応じて、入力構成ファイルに暗号化されたパスワードを指定できます。プレーンテキスト形式でパスワードを指定することもできます。

Lambda 関数にアクセスできる単一の KMS キーを使用して、すべてのパスワードを暗号化します。**aws kms encrypt --key-id** <*KMS-ARN*> **--plaintext** <*password*> コマンドを使用して暗号化されたパスワードを生成します。このコマンドを実行するには、AWS CLI をインストールして設定する必要があります。

（注） パスワードが対称 KMS キーを使用して暗号化されていることを確認します。

AWS CLI については、AWS のコマンド ライン インタフェース [英語] を参照してください。マスターキーと暗号化の詳細については、パスワードの暗号化と KMS に関する AWS ドキュメントのキーの作成 [英語] と AWS CLI コマンドリファレンス [英語] を参照してください。

例：

```
$ aws kms encrypt --key-id <KMS-ARN> --plaintext <password>
{
    "KeyId": "KMS-ARN",
    "CiphertextBlob":
"AQICAHgcQFAGtz/hvaxMtJvY/x/rfHnKI3clFPpSXUU7HQRnCAFwfXhXHJAHL8tcVmDqurALAAAAajBoBgkqhki
G9w0BBwagWzBZAgEAMFQGCSqGSIb3DQEHATAeBglghkgBZQMEAS4wEQQM45AIkTqjSekX2mniAgEQgCcOav6Hhol
+wxpWKtXY4y1Z1d0z1P4fx0jTdosfCbPnUExmNJ4zdx8="
}
$
```

CiphertextBlob キーの値をパスワードとして使用する必要があります。

展開に向けた Amazon GuardDuty リソースファイルの準備

Amazon GuardDuty ソリューションの展開リソースファイルは、Cisco GitHub リポジトリで入手できます。

AWS に Amazon GuardDuty ソリューションを展開する前に、次のファイルを準備する必要があります。

- Secure Firewall Threat Defense Virtual マネージャの構成入力ファイル
- Lambda 関数の zip ファイル
- Lambda レイヤの zip ファイル

構成入力ファイルの準備

構成テンプレートでは、Amazon GuardDuty ソリューションと連携する管理センターまたはデバイスマネージャの詳細を定義する必要があります。ネットワーク オブジェクト グループの更新メソッドで管理センターやデバイスマネージャと Amazon GuardDuty の統合を計画している場合にのみ、構成ファイルを更新することを推奨します。

始める前に

- 構成ファイルにユーザーアカウントの詳細を指定する前に、デバイスマネージャのユーザーアカウントを認証および検証します。
- 構成ファイルで複数の管理センターやデバイスマネージャを設定している場合は、各管理センターやデバイスマネージャのパラメータが構成ファイルに 1 つだけ入力され、重複するエントリがないことを確認します。
- 管理センターとデバイスマネージャの IP アドレスと名前を書き留めておく必要があります。
- 管理センターとデバイスマネージャでこれらのネットワーク オブジェクト グループにアクセスして更新するには、Lambda 関数の管理者権限を持つユーザーアカウントを作成しておく必要があります。

ステップ 1 Amazon GuardDuty リソースファイルをダウンロードしたローカルマシンにログインします。

ステップ 2 **ngfwv-template > configuration** フォルダを参照します。

ステップ 3 テキストエディタツールで `ngfwv-manager-config-input.ini` ファイルを開きます。

このファイルには、Amazon GuardDuty ソリューションの統合と展開を計画している管理センターまたはデバイスマネージャの詳細を入力する必要があります。

ステップ 4 各パラメータに対応する管理センターまたはデバイスマネージャに関する以下の詳細を入力します。

パラメータ	説明
[ngfwv-1]	セクション名：管理センターまたはデバイスマネージャの一意の識別子。
public-ip	管理センターまたはデバイスマネージャの IP アドレス。
device-type	管理センターまたはデバイスマネージャを介して Amazon GuardDuty ソリューションを展開する管理対象デバイスのタイプ。使用できる値は FMC または FDM です。
ユーザー名	管理センターまたはデバイスマネージャにログインするためのユーザー名。
パスワード	管理センターまたはデバイスマネージャにログインするためのパスワード。パスワードには、プレーンテキスト形式または KMS を使用して暗号化された文字列を使用できます。
object-group-name	Lambda 関数が悪意のあるホスト IP を追加して更新するネットワーク オブジェクト グループの名前。複数のネットワーク オブジェクト グループ名を入力する場合は、カンマ区切り値になっていることを確認してください。

ステップ 5 `ngfwv-manager-config-input.ini` ファイルを保存して閉じます。

次のタスク

Lambda 関数のアーカイブファイルを作成します。Lambda 関数のアーカイブファイルの準備（62 ページ） を参照してください。

Lambda 関数のアーカイブファイルの準備

ここでは、Linux 環境で Lambda 関数ファイルをアーカイブする方法について説明します。

（注） アーカイブプロセスは、ファイルのアーカイブを実行するローカルマシンのオペレーティングシステムによって異なる場合があります。

ステップ 1 Amazon GuardDuty リソースをダウンロードしたローカルマシンで CLI コンソールを開きます。

ステップ 2 `/lambda` フォルダに移動し、ファイルをアーカイブします。

以下は、Linux ホストからのサンプルトランスクリプトです。

```
$ cd lambda
$ zip ngfwv-gd-lambda.zip *.py
adding: aws.py (deflated 71%) adding: fdm.py (deflated 79%)
adding: fmcv.py (deflated 79%)
adding: main.py (deflated 73%)
adding: utils.py (deflated 65%)
$
```

zip ファイル `ngfwv-gd-lambda.zip` が作成されます。

ステップ 3 終了して CLI コンソールを閉じます。

次のタスク

zip ファイル `ngfwv-gd-lambda.zip` を使用して、Lambda レイヤの zip ファイルを作成します。Lambda レイヤファイルの準備 （63 ページ）を参照してください

Lambda レイヤファイルの準備

ここでは、Linux 環境で Lambda レイヤファイルをアーカイブする方法について説明します。

（注） アーカイブプロセスは、ファイルのアーカイブを実行するローカルマシンのオペレーティングシステムによって異なる場合があります。

ステップ 1 Amazon GuardDuty リソースをダウンロードしたローカルマシンで CLI コンソールを開きます。

ステップ 2 CLI コンソールで次のアクションを実行します。

以下は、Python 3.9 がインストールされている Ubuntu 22.04 などの Linux ホストでのサンプルトランスクリプトです。

```
$ mkdir -p layer
$ virtualenv -p /usr/bin/python3.9 ./layer/
$ source ./layer/bin/activate
$ pip3.9 install cffi==1.15.0
$ pip3.9 install cryptography==37.0.2
$ pip3.9 install paramiko==2.7.1
$ pip3.9 install requests==2.23.0
$ mkdir -p ./python/.libs_cffi_backend/
$ cp -r ./layer/lib/python3.9/site-packages/* ./python/
$ zip -r ngfwv-gd-lambda-layer.zip ./python
```

zip ファイル `ngfwv-gd-lambda-layer.zip` が作成されます。

Lambda レイヤを作成するには、Python 3.9 とその依存関係をインストールする必要があることに注意してください。

以下は、Ubuntu 22.04 などの Linux ホストに Python 3.9 をインストールするためのサンプルトランスクリプトです。

```
$ sudo apt update
$ sudo apt install software-properties-common
$ sudo add-apt-repository ppa:deadsnakes/ppa
$ sudo apt install python3.9
$ sudo apt install python3-virtualenv
$ sudo apt install zip
$ sudo apt-get install python3.9-distutils
$ sudo apt-get install python3.9-dev
$ sudo apt-get install libffi-dev
```

ステップ 3 終了して CLI コンソールを閉じます。

次のタスク

Amazon S3 バケットでは、Secure Firewall Threat Defense Virtual の構成ファイル、Lambda 関数の zip ファイル、および Lambda レイヤの zip ファイルをアップロードする必要があります。Amazon Simple Storage Service へのファイルのアップロード （64 ページ）を参照してください

Amazon Simple Storage Service へのファイルのアップロード

すべての Amazon GuardDuty ソリューション アーティファクトを準備したら、AWS ポータルの Amazon Simple Storage Service（S3）バケットフォルダにファイルをアップロードする必要があります。

ステップ 1 https://aws.amazon.com/marketplace（Amazon マーケットプレイス）に移動してサインインします。

ステップ 2 Amazon S3 コンソールを開きます。

ステップ 3 Amazon GuardDuty アーティファクトをアップロードするための Amazon S3 バケットを作成します。Amazon S3 の作成 [英語] を参照してください。

ステップ 4 次の Amazon GuardDuty アーティファクトを Amazon S3 バケットにアップロードします。

- Secure Firewall Threat Defense Virtual構成ファイル：`ngfwv-config-input.ini`

  （注） 管理センターでセキュリティ インテリジェンスのネットワーク フィード メソッドを使用して Amazon GuardDuty ソリューションを展開する場合、このファイルをアップロードする必要はありません。

- Lambda レイヤ zip ファイル：`ngfwv-gd-lambda-layer.zip`
- Lambda 関数 zip ファイル：`ngfwv-gd-lambda.zip`

次のタスク

Amazon GuardDuty リソースの展開に使用する CloudFormation テンプレートを準備します。CloudFormation テンプレートの入力パラメータの収集 （65 ページ） を参照してください。

CloudFormation テンプレートの入力パラメータの収集

シスコでは、AWS の Amazon GuardDuty ソリューションに必要なリソースを展開する際に使用する CloudFormation テンプレートを提供しています。展開する前に、次のテンプレートパラメータの値を収集します。

Template Parameters

パラメータ	説明	例
展開名*	このパラメータに入力する名前は、Cloud Formation テンプレートによって作成されるすべてのリソースのプレフィックスとして使用されます。	**cisco-ngfwv-gd**
GD 検出結果の最小の重大度レベル*	Amazon GuardDuty の検出結果で処理の対象となる最小重大度レベルは、**1.0** から **8.9** の範囲にする必要があります。報告された検出結果の重大度が最小範囲よりも低い場合は無視されます。 重大度の分類は次のとおりです。 • 低：1.0 ～ 3.9 中：4.0 ～ 6.9 高：7.0 ～ 8.9	**4.0%**
管理者の電子メール ID*	管理センターまたはデバイスマネージャ の Lambda 関数によって実行された更新に関する通知を受信する Secure Firewall Threat Defense Virtual マネージャ の管理者の電子メールアドレス。	**abc@xyz.com**
S3 バケット名*	Amazon GuardDuty アーティファクトファイル （Lambda 関数の zip ファイル、Lambda レイヤの zip ファイル、および Secure Firewall Threat Defense Virtual 設定マネージャファイル）が格納された Amazon S3 バケットの名前。	例：**ngfwv-gd-bucket**
S3 バケットフォルダ/パスプレフィックス	構成ファイルが保存されている Amazon S3 バケットのパスまたは	例：「」または「**cisco/ngfwv-gd/**」

パラメータ	説明	例
	フォルダ名。フォルダがない場合は、このフィールドを空白のままにします。	
Lambda レイヤの zip ファイル名*	Lambda レイヤの zip ファイル名。	例：`ngfwv-gd-lambda-layer.zip`
Lambda 関数の zip ファイル名*	Lambda 関数の zip ファイル名。	例：`ngfwv-gd-lambda.zip`
Cisco Secure Firewall Management Center Secure Firewall と Device Manager マネージャの構成ファイル名	Cisco Firewall Threat Defense Virtual のマネージャ設定の詳細が保存された ***.ini** ファイル（パブリック IP、ユーザー名、パスワード、デバイスタイプ、ネットワーク オブジェクト グループ名など）。 （注） このファイルは、Amazon GuardDuty との統合でネットワーク オブジェクト グループの更新メソッドを使用している場合にのみ必要です。 セキュリティ インテリジェンス フィード メソッドを使用している場合は、この入力をスキップできます。	例：`ngfwv-config-input.ini`
パスワードの暗号化に使用される KMS キーの ARN	既存の KMS（パスワードの暗号化に使用される AWS KMS キー）の ARN。Secure Firewall Threat Defense Virtual の構成入力ファイルでプレーンテキストパスワードが指定されている場合は、このパラメータを空のままにしておくことができます。指定する場合、Secure Firewall Threat Defense Virtual の構成入力ファイルに記載されているすべてのパスワードを暗号化する必要があります。パスワードの暗号化には、指定された ARN のみを使用する必要があります。暗号化パスワードの生成：aws kms	例： **arn:aws:kms:<region>:<awsaccountid>:key/<keyid>**

パラメータ	説明	例
	encrypt --key-id <KMS ARN> --plaintext <password>	
デバッグログの有効化/無効化*	CloudWatch で Lambda 関数のデバッグログを有効または無効にします。	例：**enable** または **disable**

*：必須フィールド

次のタスク

CloudFormation テンプレートを使用してスタックを展開します。スタックの展開 （67 ページ）を参照してください

スタックの展開

Amazon GuardDuty ソリューションを導入するためのすべての前提条件プロセスを完了した後に、AWS CloudFormation スタックを作成します。対象ディレクトリのテンプレートファイル（`templates/cisco-ngfwv-gd-integration.yaml`）を使用し、「CloudFormation テンプレートの入力パラメータの収集」で収集したパラメータを指定します。

ステップ 1 AWS コンソールにログインします。

ステップ 2 [サービス（Services）] > [CloudFormation] > [スタック（Stacks）] > [スタックの作成（Create stack）]（新しいリソースを使用）> [テンプレートの準備（Prepare template）]（テンプレートはフォルダ内にあります）> [テンプレートの指定（Specify template）] > [テンプレートソース（Template source）]（ターゲットディレクトリ `templates/cisco-ngfwv-gd-integration.yaml` からテンプレートファイルをアップロード）> [スタックの作成（Create Stack）] の順に操作を行います。

AWS でスタックを展開する方法の詳細については、AWS ドキュメント [英語] を参照してください。

次のタスク

展開を検証します。展開の検証 （68 ページ） を参照してください。

また、Amazon GuardDuty によって報告された脅威検出の更新に関する電子メール通知を受信するように登録します。電子メール通知の登録 （67 ページ） を参照してください。

電子メール通知の登録

CloudFormation テンプレートでは、GuardDutyの検出結果の更新に関する通知を受信するように、電子メール ID が設定されています。これは Lambda 関数によって実行されます。AWS に

CloudFormation テンプレートを展開すると、Amazon Simple Notification Service（SNS）サービスを介してこの電子メール ID に電子メール通知が送信され、通知の更新を登録するように要求されます。

ステップ 1 電子メール通知を開きます。

ステップ 2 電子メール通知で利用可能な**サブスクリプション**リンクをクリックします。

次のタスク

展開を検証します。展開の検証 （68 ページ） を参照してください。

展開の検証

この項で説明されているように、AWS には Amazon GuardDuty ソリューションを検証するオプションがあります。CloudFormation の展開が完了したら、以下に示す展開の検証手順を実行できます。

始める前に

展開を検証するためのコマンドを実行するには、AWS コマンド ライン インターフェイス（CLI）がインストールおよび設定されていることを確認します。AWS CLI のドキュメントについては、AWS のコマンド ライン インターフェイス [英語] を参照してください。

ステップ 1 AWS 管理コンソールにログインします。

ステップ 2 [サービス（Services）]>[GuardDuty]>[設定（Settings）]>[GuardDutyのが概要（About GuardDuty）]>[ディテクタID（Detector ID）] に移動して、ディテクタ ID を書き留めます。

このディテクタ ID は、Amazon GuardDuty のサンプル検出結果を生成するために必要です。

ステップ 3 AWS CLI コンソールを開き、次のコマンドを実行して Amazon GuardDuty のサンプル検出結果を生成します。

```
aws guardduty create-sample-findings --detector-id <detector-id> --finding-types
UnauthorizedAccess:EC2/MaliciousIPCaller.Custom
```

```
aws guardduty create-sample-findings --detector-id <detector-id> --finding-types
UnauthorizedAccess:EC2/MaliciousIPCaller.Custom
```

ステップ 4 Amazon GuardDuty コンソールの結果リストでサンプルの検出結果を確認します。

サンプル検出結果には、プレフィックス **[sample]** が含まれています。接続方向、リモート IP アドレスなどの属性を参照して、サンプル検出結果の詳細を確認できます。

ステップ 5 Lambda 関数が実行されるのを待ちます。

Lambda 関数がトリガーされたら、以下を確認します。

- 受信した Amazon GuardDuty の検出結果と、Lambda 関数によって実行された Secure Firewall Threat Defense Virtual マネージャ の更新に関する詳細が記載された電子メール通知。
- レポートファイルが Amazon S3 バケットに生成されているかどうかを確認します。レポートファイルには、サンプルの Amazon GuardDuty の検出結果によって報告された悪意のある IP アドレスが含まれています。レポートファイル名は、`<deployment-name>-report.txt` の形式になっています。
- ネットワーク オブジェクト グループの更新メソッドの場合：設定されたマネージャ（Secure Firewall Management Center Virtual または Secure Firewall デバイスマネージャ）で、サンプルの検出結果から更新された悪意のある IP アドレスを追加してネットワーク オブジェクト グループが更新されていることを確認します。
- セキュリティ インテリジェンス フィード メソッドの場合：レポートファイルの URL が管理センターの設定で既に更新されているかどうかを確認します。レポートファイルURL の最終更新タイムスタンプは、管理センターの次のパスで表示できます。
  - [オブジェクト（Objects）] > [オブジェクト管理（Object Management）] > [セキュリティ インテリジェンス（Security Intelligence ）] > [ネットワークリストとフィード（Network Lists and Feeds）] > 設定したフィードを選択
  - または、フィードを手動で更新してから、[最終更新（Last Updated）] のタイムスタンプを確認することもできます。次のパスでフィードを選択して更新できます。

    [オブジェクト（Objects）] > [オブジェクト管理（Object Management）] > [セキュリティ インテリジェンス（Security Intelligence ）] > [ネットワークリストとフィード（Network Lists and Feeds）] > [フィードの更新（Update Feeds）]

ステップ 6 [AWSコンソール（AWS Console）] > [サービス（Services）] > [CloudWatch] > [ログ（Logs）] > [ロググループ（Log groups）] に移動し、ロググループを選択して、CloudWatch コンソールで Lambda ログを確認します。CloudWatch のロググループ名は、`<deployment-name>-lambda` の形式になっています。

ステップ 7 展開を検証した後、次のようにサンプル検出結果によって生成されたデータをクリーンアップすることを推奨します。

a) AWS コンソールから [サービス（Services）] > [GuardDuty] > [結果（Findings）] > [結果を選択（Select the finding）] > [アクション（Actions）] > [アーカイブ（Archive）] に移動して、サンプルの検出結果データを表示します。

b) ネットワーク オブジェクト グループに追加された悪意のある IP アドレスを削除して、キャッシュされたデータを Secure Firewall Management Center Virtual から消去します。

c) Amazon S3 バケットのレポートファイルをクリーンアップします。サンプルの検出結果で報告された悪意のある IP アドレスを削除することで、ファイルを更新できます。

既存のソリューション展開構成の更新

展開後にS3バケットやS3バケットフォルダとパスプレフィックス値を更新しないことを推奨します。ただし、展開したソリューションの構成を更新する必要がある場合は、AWSコンソールの [CloudFormation] ページで [スタックの更新（Update Stack）] オプションを使用します。

以下のパラメータを更新できます。

パラメータ	説明
Secure Firewall Threat Defense Virtual マネージャの構成ファイル名	Amazon S3 バケットの構成ファイルを追加または更新します。以前のファイルと同じ名前でファイルを更新できます。構成ファイル名が変更された場合は、AWS コンソールの [スタックの更新（Update stack）] オプションを使用して、このパラメータを更新できます。
GD 検出結果の最小の重大度レベル*	AWS コンソールの [スタックの更新（Update stack）] オプションを使用して、パラメータ値を更新します。
管理者の電子メール ID*	AWS コンソールの [スタックの更新（Update stack）] オプションを使用して、電子メール ID のパラメータ値を更新します。SNS サービスコンソールを介して電子メールのサブスクリプションを追加または更新することもできます。
S3 バケット名*	Amazon S3 バケット内の zip ファイルを新しい名前で更新してから、AWS コンソールの [スタックの更新（Update Stack）] オプションを使用してパラメータを更新します。
Lambda レイヤの zip ファイル名*	Amazon S3 バケット内の Lambda レイヤ zip ファイル名を新しい名前で更新してから、AWS コンソールの [スタックの更新（Update stack）] オプションを使用して、このパラメータ値を更新します。
Lambda 関数の zip ファイル名*	Amazon S3 バケット内の Lambda 関数 zip ファイルを新しい名前で更新してから、AWS コンソールの [スタックの更新（Update stack）] オプションを使用して、このパラメータ値を更新します。

パラメータ	説明
パスワードの暗号化に使用される KMS キーの ARN	AWS コンソールの [スタックの更新（Update stack）] オプションを使用して、パラメータ値を更新します。
デバッグログの有効化/無効化*	AWS コンソールの [スタックの更新（Update stack）] オプションを使用して、パラメータ値を更新します。

ステップ 1 AWS 管理コンソールに進みます。

ステップ 2 必要に応じて、新しいバケットとフォルダを作成します。

ステップ 3 以下に示すアーティファクトが古いバケットから新しいバケットにコピーされていることを確認します。

- Secure Firewall Threat Defense Virtual構成ファイル：`ngfwv-config-input.ini`
- Lambda レイヤ zip ファイル：`ngfwv-gd-lambda-layer.zip`
- Lambda 関数 zip ファイル：`ngfwv-gd-lambda.zip`
- Output レポートファイル：`<deployment-name>-report.txt`

ステップ 4 パラメータ値を更新するには、**Services > CloudFormation > Stacks > > Update (Update Stack) > Prepare template > Use current template > Next >** <*update parameters*>**> Update Stack** に移動します。

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。