



Oracle Cloud Infrastructure への Threat Defense Virtual の展開

Threat Defense Virtual は、Oracle Cloud Infrastructure (OCI) に展開できます。OCI は、オラクルが提供するパブリック クラウド コンピューティング サービスで、高可用性のホステッド環境でアプリケーションを実行できます。

次の手順では、OCI 環境を準備し、Threat Defense Virtual インスタンスを起動する方法について説明します。OCI ポータルにログインし、OCI Marketplace で Cisco Firepower NGFW virtual firewall (NGFWv) 製品を検索し、コンピューティングインスタンスを起動します。Threat Defense Virtual の起動後に、トラフィックの送信元と接続先に応じて、トラフィックをファイアウォールに転送するようにルートテーブルを設定する必要があります。

- [OCI への Threat Defense Virtual の展開について \(2 ページ\)](#)
- [エンドツーエンドの手順 \(3 ページ\)](#)
- [Threat Defense Virtual と OCI の前提条件 \(5 ページ\)](#)
- [Threat Defense Virtual および OCI のガイドラインと制限事項 \(5 ページ\)](#)
- [OCI 上の Threat Defense Virtual のネットワークトポロジーの例 \(7 ページ\)](#)
- [Secure Firewall Threat Defense Virtual デバイスの管理方法 \(8 ページ\)](#)
- [仮想クラウドネットワーク \(VCN\) の設定 \(9 ページ\)](#)
- [OCI への Threat Defense Virtual の展開 \(12 ページ\)](#)
- [インターフェイスの接続 \(14 ページ\)](#)
- [接続された VNIC のルートルールの追加 \(14 ページ\)](#)
- [OCI の Threat Defense Virtual 向け Auto Scale ソリューション \(15 ページ\)](#)
- [前提条件 \(16 ページ\)](#)
- [パスワードの暗号化 \(24 ページ\)](#)
- [Threat Defense Virtual の構成ファイルの準備 \(25 ページ\)](#)
- [OCI への Auto Scale の展開 \(31 ページ\)](#)
- [展開の検証 \(37 ページ\)](#)
- [Auto Scale のアップグレード \(37 ページ\)](#)
- [ロードバランサのバックエンドセット \(38 ページ\)](#)
- [OCI の Auto Scale 設定の削除 \(39 ページ\)](#)

- SSH を使用した Threat Defense Virtual インスタンスへの接続 (42 ページ)
- OpenSSH を使用した Threat Defense Virtual インスタンスへの接続 (42 ページ)
- PuTTY を使用した Threat Defense Virtual インスタンスへの接続 (43 ページ)

OCI への Threat Defense Virtual の展開について

Cisco Secure Firewall Threat Defense Virtual は、物理的な Cisco 脅威に対する防御 と同じソフトウェアを実行して、仮想フォームファクタにおいて実証済みのセキュリティ機能を提供します。Threat Defense Virtual は、パブリック OCI で展開できます。その後設定を行うことで、時間の経過とともにロケーションを展開、契約、またはシフトする仮想および物理データセンターのワークロードを保護できます。

OCI のコンピューティングシェイプ

シェイプは、インスタンスに割り当てられる CPU の数、メモリの量、およびその他のリソースを決定するテンプレートです。Threat Defense Virtual は、次の OCI のシェイプタイプをサポートします。

表 1: でサポートされるコンピューティングシェイプ *Threat Defense Virtual*

OCI シェイプ	属性		インターフェイス
	oCPU	RAM (GB)	
VM.DenseIO2.8	8	120	最小 4、最大 8
VM.StandardB1.4	4	3	最小 4、最大 4
VM.StandardB1.8	4	96	最小 4、最大 8
VM.Standard1.4	4	28	最小 4、最大 4
VM.Standard1.8	8	72	最小 4、最大 8
VM.Standard2.4	4	60 GB	最小 4、最大 4
VM.Standard2.8	8	120 GB	最小 4、最大 8
VM.Standard3.Flex	4	64	最小 4、最大 4
	6	96	最小 4、最大 6
	8	128	最小 4、最大 8

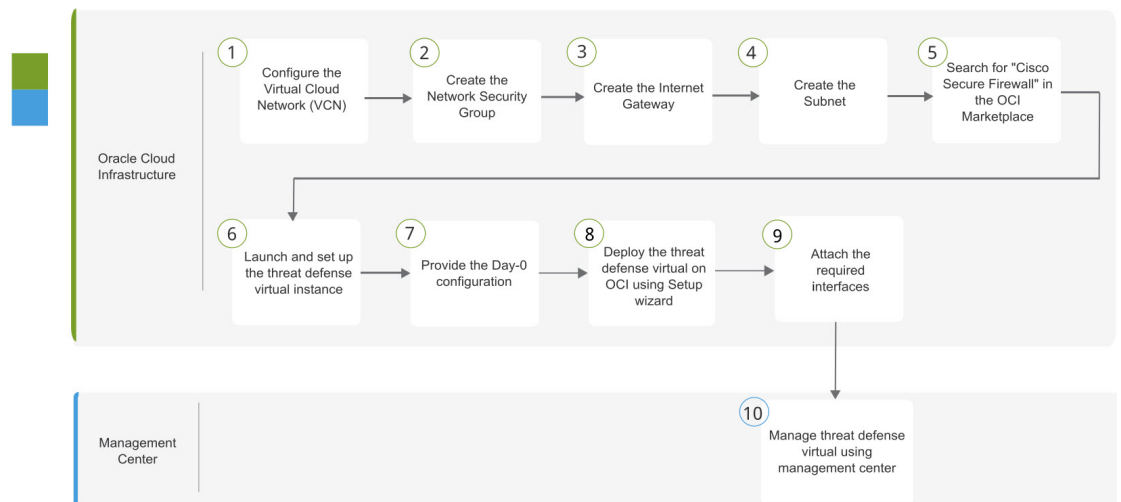
OCI シェイプ	属性		インターフェイス
	oCPU	RAM (GB)	
VM.Optimized3.Flex	4	72	最小 4、最大 8
	6	84	最小 4、最大 12
	8	112	最小 4、最大 16
VM.Standard.E4.Flex	4	32	最小 4、最大 4
	6	48	最小 4、最大 6
	8	64	最小 4、最大 8

- OCI では、1つの oCPU は2つの vCPU に相当します。
- Threat Defense Virtual には、少なくとも4つのインターフェイスが必要です。

ユーザーは、OCI でアカウントを作成し、Oracle Cloud Marketplace の Cisco Firepower NGFW 仮想ファイアウォール (NGFWv) 製品を使用してコンピューティング インスタンスを起動し、OCI のシェイプを選択します。

エンドツーエンドの手順

次のフローチャートは、Oracle Cloud Infrastructure 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	Oracle Cloud Infrastructure	仮想クラウドネットワーク (VCN) の設定 : Virtual Cloud Network (VCN) を設定します ([ネットワーキング (Networking)] > [仮想クラウドネットワーク (Virtual Cloud Networks)] > [CIDR ブロック (CIDR block)] > [VCNの作成 (Create VCN)]) 。
②	Oracle Cloud Infrastructure	ネットワークセキュリティグループの作成 : ネットワークセキュリティグループを作成します ([ネットワーキング (Networking)] > [仮想クラウドネットワーク (Virtual Cloud Networks)] > [仮想クラウドネットワークの詳細 (Virtual Cloud Network Details)] > [ネットワークセキュリティグループ (Network Security Groups)] > [ネットワークセキュリティグループの作成 (Create Network Security Group)]) 。
③	Oracle Cloud Infrastructure	インターネットゲートウェイの作成 : インターネットゲートウェイを作成します[ネットワーキング (Networking)] > [仮想クラウドネットワーク (Virtual Cloud Networks)] > [仮想クラウドネットワークの詳細 (Virtual Cloud Network Details)] > [インターネットゲートウェイ (Internet Gateways)] > [インターネットゲートウェイの作成 (Create Internet Gateway)] 。
④	Oracle Cloud Infrastructure	サブネットの作成 : サブネットを作成します ([ネットワーキング (Networking)] > [仮想クラウドネットワーク (Virtual Cloud Networks)] > [仮想クラウドネットワークの詳細 (Virtual Cloud Network Details)] > [サブネット (Subnets)] > [サブネットの作成 (Create Subnet)]) 。
⑤	Oracle Cloud Infrastructure	OCI への Threat Defense Virtual の展開 (12 ページ) : OCI Marketplace で「Cisco Secure Firewall」を検索します。
⑥	Oracle Cloud Infrastructure	OCI への Threat Defense Virtual の展開 (12 ページ) : Threat Defense Virtual インスタンスを起動して設定します。
⑦	Oracle Cloud Infrastructure	OCI への Threat Defense Virtual の展開 (12 ページ) : Day-0 構成ファイルを指定します。
⑧	Oracle Cloud Infrastructure	OCI への Threat Defense Virtual の展開 (12 ページ) : セットアップウィザードを使用して OCI に Threat Defense Virtual を展開します。
⑨	Oracle Cloud Infrastructure	インターフェイスの接続 : インターフェイスを接続します ([コンピューティング (Compute)] > [インスタンス (Instances)] > [インスタンスの詳細 (Instance Details)] > [接続されたVNIC (Attached VNICs)]) 。
⑩	Management Center	Management Center を使用した Threat Defense Virtual の管理

Threat Defense Virtual と OCI の前提条件

- <https://www.oracle.com/cloud/> で、OCI アカウントを作成します。
- Cisco スマートアカウント。Cisco Software Central (<https://software.cisco.com/>) で作成できます。
- Threat Defense Virtual へのライセンス付与。
 - 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、『*Cisco Secure Firewall Management Center Admin Guide*』の「Licensing」を参照してください。
- インターフェ이스の要件：
 - 管理インターフェース (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。
 - トラフィック インターフェース (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。
- Threat Defense Virtual のシステム要件については、『[Cisco Firepower Compatibility Guide](#)』を参照してください。

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

サポートされる機能

- OCI 仮想クラウドネットワーク (VCN) での展開
- ルーテッドモード (デフォルト)
- ライセンス : BYOL のみをサポート
- Management Center サポートのみ。
- Single Root I/O Virtualization (SR-IOV) をサポート。

FTDv スマートライセンスのパフォーマンス階層

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、100Mbps	4 コア/8 GB	100Mbps	50
FTDv10、1Gbps	4 コア/8 GB	1Gbps	250
FTDv20、3Gbps	4 コア/8 GB	3 Gbps	250
FTDv30、5Gbps	8 コア/16 GB	5 Gbps	250
FTDv50、10Gbps	12 コア/24 GB	10 Gbps	750
FTDv100、16 Gbps	16 コア/32 GB	16 Gbps	10,000

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、『Cisco Secure Firewall Management Center Admin Guide』の「Licensing」の章を参照してください。



(注) vCPU/メモリの値を変更するには、最初に Threat Defense Virtual デバイスの電源をオフにする必要があります。

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、「[OCI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。

Snort

- 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。

サポートされない機能

- Device Manager を介したローカル管理サポート。
- Threat Defense Virtual ネイティブ HA
- 自動スケール
- トランスペアレント/インライン/パッシブ モード
- DHCP を使用したデータインターフェイス設定
- IPv6

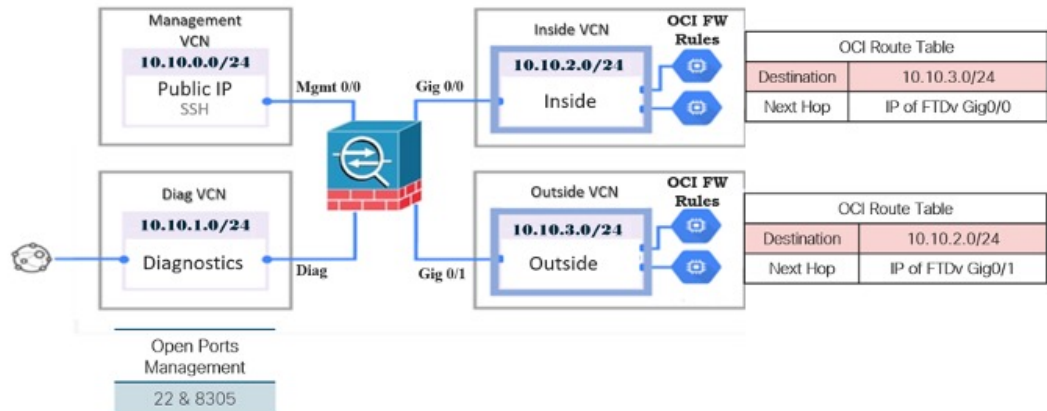
制限事項

- OCI に Threat Defense Virtual を展開する場合、Mellanox 5 は SR-IOV モードの vNIC としてサポートされません。
- 静的設定と DHCP 設定の両方で Firewall Threat Defense Virtual に必要な個別のルーティングルール。

OCI 上の Threat Defense Virtual のネットワークトポロジの例

次の図は、Threat Defense Virtual 用に 4 つのサブネット（管理、診断、内部、外部）が OCI 内に設定されたルーテッドファイアウォールモードの Threat Defense Virtual の推奨トポロジを示しています。

図 1: OCI 上の Threat Defense Virtual の展開例



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。ThreatDefense 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。

仮想クラウドネットワーク (VCN) の設定

Threat Defense Virtual 展開用の仮想クラウドネットワーク (VCN) を設定します。少なくとも、Threat Defense Virtual の各インターフェイスに 1 つずつ、合計 4 つの VCN が必要です。

次の手順に進み、管理 VCN を完了できます。次に、[ネットワークング (Networking)] に戻り、診断、内部、および外部の各インターフェイスの VCN を作成します。

ステップ 1 OCI にログインし、地域を選択します。

OCI 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 [ネットワークング (Networking)] > [仮想クラウドネットワーク (Virtual Cloud Networks)] を選択し、[VCN の作成 (Create VCN)] をクリックします。

ステップ 3 [名前 (Name)] に、VCN のわかりやすい名前を入力します (例: *FTDv-Management*) 。

ステップ 4 VCN の CIDR ブロックを入力します。

- a) 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ックスのコンパクトな表現です。たとえば、「10.0.0/24」と入力します。

(注) この VCN で DNS ホスト名を使用します。

ステップ 5 [VCN の作成 (Create VCN)] をクリックします。

次のタスク

次の手順に進み、管理 VCN を完了します。管理 VCN を完了したら、診断、内部、および外部の各インターフェイスの VCN を作成します。



- (注) ナビゲーションメニューからサービスを選択すると、左側のメニューにコンパートメントリストが表示されます。コンパートメントはリソースの整理に役立ち、リソースへのアクセスを制御しやすくなります。ルートコンパートメントは、テナントがプロビジョニングされるときに Oracle によって作成されます。管理者は、ルートコンパートメントにさらに多くのコンパートメントを作成し、アクセスルールを追加して、どのユーザーがそれらのコンパートメントを表示してアクションを実行できるかを制御できます。詳細については、Oracle のドキュメント『[Managing Compartments](#)』[英語]を参照してください。

ネットワーク セキュリティ グループの作成

ネットワーク セキュリティ グループは、一連の vNIC と、vNIC に適用される一連のセキュリティルールで構成されます。

ステップ 1 [ネットワーク (Networking)] > [仮想クラウドネットワーク (Virtual Cloud Networks)] > [仮想クラウドネットワークの詳細 (Virtual Cloud Network Details)] > [ネットワーク セキュリティ グループ (Network Security Groups)] を選択し、[ネットワーク セキュリティ グループの作成 (Create Network Security Group)] をクリックします。

ステップ 2 [名前 (Name)] に、ネットワーク セキュリティ グループのわかりやすい名前を入力します (例: *FTDv-Mgmt-Allow-22-8305*)。

ステップ 3 [Next] をクリックします。

ステップ 4 セキュリティルールを追加します。

- a) SSH アクセスに TCP ポート 22 を許可するルールを追加します。
- b) HTTPS アクセスに TCP ポート 443 を許可するルールを追加します。

Threat Defense Virtual は Management Center を介して管理できます。これには、HTTPS 接続用にポート 8305 を開く必要があります。

- (注) これらのセキュリティルールを管理インターフェイス/VCN に適用します。

ステップ 5 [作成 (Create)] をクリックします。

インターネットゲートウェイの作成

管理サブネットを公的にアクセス可能にするには、インターネットゲートウェイが必要です。

ステップ 1 [ネットワーク (Networking)] > [仮想クラウドネットワーク (Virtual Cloud Networks)] > [仮想クラウドネットワークの詳細 (Virtual Cloud Network Details)] > [インターネットゲートウェイ (Internet Gateways)] を選択し、[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2 [名前 (Name)]にインターネットゲートウェイのわかりやすい名前を入力します (例: *FTDv-IG*) 。

ステップ3 [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4 インターネットゲートウェイへのルートを追加します。

- a) [ネットワーキング (Networking)]>[仮想クラウドネットワーク (Virtual Cloud Networks)]>[仮想クラウドネットワークの詳細 (Virtual Cloud Network Details)]>[ルートテーブル (Route Tables)]を選択します。
- b) ルートルールを追加するには、デフォルトのルートテーブルのリンクをクリックします。
- c) [ルートルールの追加 (Add Route Rules)] をクリックします。
- d) [ターゲットタイプ (Target Type)] ドロップダウンから、[インターネットゲートウェイ (Internet Gateway)] を選択します。
- e) 宛先の IPv4 CIDR ブロックを入力します (例: 0.0.0.0/0) 。
- f) [ターゲットインターネットゲートウェイ (Target Internet Gateway)] ドロップダウンから、作成したゲートウェイを選択します。
- g) [ルートルールの追加 (Add Route Rules)] をクリックします。

サブネットの作成

各 VCN には、少なくとも1つのサブネットがあります。管理 VCN の管理サブネットを作成します。また、診断 VCN の診断サブネット、内部 VCN の内部サブネット、および外部 VCN の外部サブネットも必要です。

ステップ1 [ネットワーキング (Networking)]>[仮想クラウドネットワーク (Virtual Cloud Networks)]>[仮想クラウドネットワークの詳細 (Virtual Cloud Network Details)]>[サブネット (Subnets)]を選択し、[サブネットの作成 (Create Subnet)] をクリックします。

ステップ2 サブネットのわかりやすい名前を入力します (例: *Management*) 。

ステップ3 [サブネットタイプ (Subnet Type)] を選択します (推奨されるデフォルトの [地域 (Regional)] のままにします) 。

ステップ4 [CIDRブロック (CIDR Block)]に値を入力します (例: 10.10.0.0/24) 。サブネットの内部 (非公開) IP アドレスは、この CIDR ブロックから取得されます。

ステップ5 [ルートテーブル (Route Table)] ドロップダウンから、以前に作成したルートテーブルのいずれかを選択します。

ステップ6 サブネットの [サブネットアクセス (Subnet Access)] を選択します。

管理サブネットの場合、これはパブリックサブネットである必要があります。

ステップ7 [DHCP オプション (DHCP Option)] を選択します。

ステップ8 以前作成した [セキュリティリスト (Security List)] を選択します。

ステップ9 [サブネットの作成 (Create Subnet)] をクリックします。

次のタスク

VCN（管理、診断、内部、外部）を設定すると、Threat Defense Virtual を起動する準備が整います。Threat Defense Virtual VCN 構成の例については、次の図を参照してください。

図 2: Threat Defense Virtual 仮想クラウドネットワーク

Virtual Cloud Networks in ftdv Compartment

Virtual Cloud Networks are virtual, private networks that you set up in Oracle data centers. It closely resembles a traditional network, with firewall rules and specific types of communication gateways that you can choose to use.

Name	State	CIDR Block	Default Route Table	DNS Domain Name	Created
FTDy-Outside	● Available	10.10.3.0/24	Default Route Table for FTDy-Outside	ftdvoutside.oraclevcn.com	Mon, Jul 6, 2020, 14:32:07 UTC
FTDy-Inside	● Available	10.10.2.0/24	Default Route Table for FTDy-Inside	ftdvinside.oraclevcn.com	Mon, Jul 6, 2020, 14:31:38 UTC
FTDy-Diagnostic	● Available	10.10.1.0/24	Default Route Table for FTDy-Diagnostic	ftdvdiagnostic.oraclevcn.com	Mon, Jul 6, 2020, 14:30:46 UTC
FTDy-Management	● Available	10.10.0.0/24	Default Route Table for FTDy-Management	ftdvmanagement.oraclevcn.com	Mon, Jul 6, 2020, 14:29:16 UTC

Showing 4 items < 1 of 1 >

OCI への Threat Defense Virtual の展開

Oracle Cloud Marketplace の Cisco Firepower NGFW 仮想ファイアウォール (NGFWv) 製品を使用して、コンピューティング インスタンスを介して OCI に Threat Defense Virtual を展開します。CPU の数、メモリの量、ネットワークリソースなどの特性に基づいて、最適なマシンシェイプを選択します。

- ステップ 1 OCI ポータルにログインします。
地域は、画面の右上隅に表示されます。目的の地域内に存在していることを確認してください。
- ステップ 2 [マーケットプレイス (Marketplace)] > [アプリケーション (Applications)] を選択します。
- ステップ 3 マーケットプレイスで「Cisco Firepower NGFW virtual firewall (NGFWv)」を検索して、製品を選択します。
- ステップ 4 契約条件を確認し、[Oracle の利用規約とパートナーの契約条件を確認して同意します。 (I have reviewed and accept the Oracle Terms of Use and the Partner terms and conditions.)] チェックボックスをオンにします。
- ステップ 5 [インスタンスの起動 (Launch Instance)] をクリックします。
- ステップ 6 [名前 (Name)] に、インスタンスのわかりやすい名前を入力します (例: FTDv-6-7)。
- ステップ 7 [シェイプの変更 (Change Shape)] をクリックし、Threat Defense Virtual に必要な CPU の数、RAM の量、およびインターフェイスの数が指定されたシェイプ (VM.Standard2.4 など) を選択します (OCI への Threat Defense Virtual の展開について (2 ページ) を参照)。
- ステップ 8 [仮想クラウドネットワーク (Virtual Cloud Network)] ドロップダウンから、[管理 VCN (Management VCN)] を選択します。
- ステップ 9 自動入力されていない場合は、[サブネット (Subnet)] ドロップダウンから [管理サブネット (Management subnet)] を選択します。

ステップ 10 [ネットワーク セキュリティ グループを使用してトラフィックを制御する (Use Network Security Groups to Control Traffic)] にチェックを入れ、管理 VCN に設定したセキュリティグループを選択します。

ステップ 11 [パブリック IP アドレスの割り当て (Assign a Public Ip Address)] オプションボタンをクリックします。

ステップ 12 [SSH キーの追加 (Add SSH keys)] の下で、[公開キーの貼り付け (Paste Public Keys)] オプションボタンをクリックして、SSH キーを貼り付けます。

Linux ベースのインスタンスは、パスワードの代わりに SSH キーペアを使用してリモートユーザーを認証します。キーペアは、秘密キーと公開キーで構成されます。インスタンスを作成するときに、秘密キーをコンピュータに保持し、公開キーを提供します。ガイドラインについては、『Linux インスタンスでのキーペアの管理 (Managing Key Pairs on Linux Instances)』 <https://docs.cloud.oracle.com/en-us/iaas/Content/Compute/Tasks/managingkeypairs.htm> を参照してください。

ステップ 13 [詳細オプションの表示 (Show Advanced Options)] リンクをクリックして、オプションを展開します。

ステップ 14 [初期化スクリプト (Initialization Script)] の下で、[クラウド初期化スクリプトの貼り付け (Paste Cloud-Init Script)] オプションボタンをクリックして、Threat Defense Virtual の day0 構成を指定します。day0 構成は、Threat Defense Virtual の初回起動時に適用されます。

次に、[クラウド初期化スクリプト (Cloud-Init Script)] フィールドにコピーして貼り付けることができる day0 構成の例を示します。

```
{
  "Hostname": "ftdv-oci",
  "AdminPassword": "myPassword@123456",
  "FirewallMode": "routed",
  "IPv4Mode": "dhcp",
  "ManageLocally": "No",
  "FmcIp": "1.2.3.4",
  "FmcRegKey": "cisco123reg",
  "FmcNatId": "cisco123nat"
}
```

- **FmcRegKey** : これは、デバイスを Management Center に登録するために使用される 1 回限りの登録キーです。登録キーは、ユーザー定義の最大 37 文字の英数字値です。
- **FmcNatId** : これは 1 回限り使用される一意の文字列です (ユーザーが定義)。ただし、デバイスと Management Center が NAT デバイスにより分離されている場合は、この一意の登録キーと同時に一意の NAT ID を入力する必要があります。

ステップ 15 [作成 (Create)] をクリックします。

次のタスク

[作成 (Create)] ボタンをクリックした後、状態が [プロビジョニング (Provisioning)] として表示される Threat Defense Virtual インスタンスをモニターします。



重要 ステータスをモニターすることが重要です。Threat Defense Virtual インスタンスの状態が [プロビジョニング (Provisioning)] から [実行中 (Running)] に移行したら、Threat Defense Virtual の起動が完了する前に必要に応じて VNIC を接続する必要があります。

インターフェイスの接続

Threat Defense Virtual は、1 つの VNIC が接続された状態で実行状態になります（[コンピューティング（Compute）]>[インスタンス（Instances）]>[インスタンスの詳細（Instance Details）]>[接続された VNIC（Attached VNICs）]を参照）。これはプライマリ VNIC と呼ばれ、管理 VCN にマッピングされます。Threat Defense Virtual が最初のブートを完了する前に、VNIC が Threat Defense Virtual で正しく検出されるように、以前に作成した他の VCN サブネット（診断、内部、外部）の VNIC を接続する必要があります。

-
- ステップ 1 新しく起動した Threat Defense Virtual インスタンスを選択します。
 - ステップ 2 [接続された VNIC（Attached VNICs）]>[VNIC の作成（Create VNIC）]の順に選択します。
 - ステップ 3 [名前（Name）]に、VNIC のわかりやすい名前を入力します（例：*Inside*）。
 - ステップ 4 [仮想クラウドネットワーク（Virtual Cloud Network）] ドロップダウンから VCN を選択します。
 - ステップ 5 [サブネット（Subnet）] ドロップダウンからサブネットを選択します。
 - ステップ 6 [ネットワークセキュリティグループを使用してトラフィックを制御する（Use Network Security Groups to Control Traffic）]をオンにして、選択した VCN 用に設定したセキュリティグループを選択します。
 - ステップ 7 [送信元と宛先のチェックをスキップ（Skip Source Destination Check）]をオンにします。
 - ステップ 8 （オプション）[プライベート IP アドレス（Private IP Address）]を指定します。これは、VNIC に対して特定の IP を選択する場合にのみ必要です。

IP を指定しない場合、OCI はサブネットに割り当てられた CIDR ブロックから IP アドレスを割り当てません。
 - ステップ 9 [変更の保存 (Save Changes)] をクリックし、VNIC を作成します。
 - ステップ 10 展開で必要となる各 VNIC について、この手順を繰り返します。
-

接続された VNIC のルートルールの追加

診断、内部、および外部の各ルートテーブルにルートテーブルルールを追加します。

-
- ステップ 1 [ネットワーク（Networking）]>[仮想クラウドネットワーク（Virtual Cloud Networks）]を選択し、VCN に関連付けられているデフォルトルートテーブル（内部または外部）をクリックします。
 - ステップ 2 [ルートルールの追加（Add Route Rules）]をクリックします。
 - ステップ 3 [ターゲットタイプ（Target Type）] ドロップダウンから、[プライベート IP（Private IP）]を選択します。
 - ステップ 4 [宛先タイプ（Destination Type）] ドロップダウンから、[CIDR ブロック（CIDR Block）]を選択します。
 - ステップ 5 [宛先 CIDR ブロック（Destination CIDR Block）]を入力します（例：0.0.0.0/0）。
 - ステップ 6 [ターゲット選択（Target Selection）] フィールドに VNIC のプライベート IP アドレスを入力します。

VNICにIPアドレスを明示的に割り当てていない場合は、VNICの詳細 ([コンピューティング (Compute)] > [インスタンス (Instances)] > [インスタンスの詳細 (Instance Details)] > [接続された VNIC (Attached VNICs)]) で自動割り当てされた IP アドレスを確認できます。

ステップ7 [ルールールの追加 (Add Route Rules)] をクリックします。

ステップ8 展開で必要となる各 VNIC について、この手順を繰り返します。

OCI の Threat Defense Virtual 向け Auto Scale ソリューション

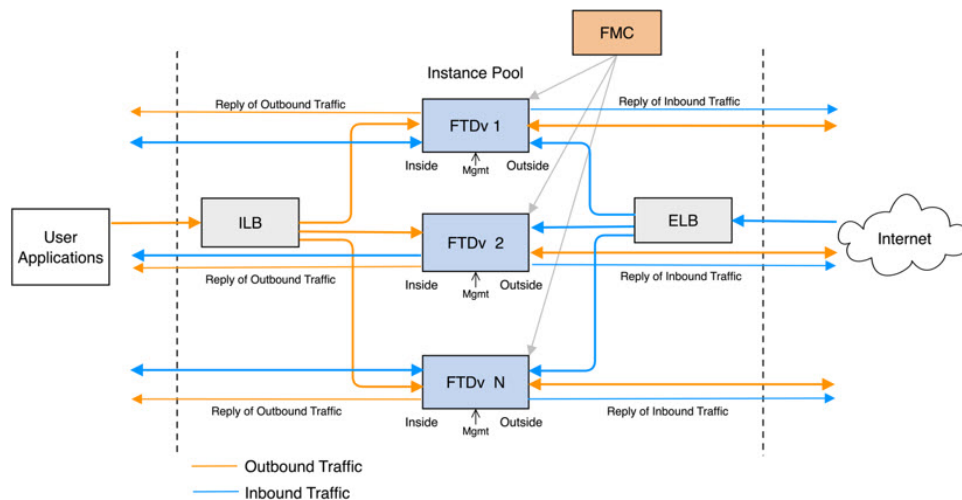
次の項では、Auto Scale ソリューションのコンポーネントが OCI の Threat Defense Virtual でのように機能するかについて説明します。

Auto Scale の導入例

OCI での Threat Defense Virtual Auto Scale ソリューションの導入例を次の図に示します。インターネット向けのロードバランサには、リスナーとターゲットグループの組み合わせを使用してポートが有効になっているパブリック IP アドレスがあります。

トラフィックに対してポートベースの分岐が可能であり、NATルールを介して実現できます。これについては次の項で説明します。

図 3 : Secure Firewall Threat Defense Virtual Auto Scale の導入例の図



スコープ

このドキュメントでは、Threat Defense Virtual Auto Scale for OCI ソリューションを導入する際の詳細な手順について説明します。

**重要**

- 導入を開始する前に、ドキュメント全体をお読みください。
- 導入を開始する前に、前提条件を満たしていることを確認します。
- ここに記載されている手順と実行順序に従っていることを確認します。

前提条件

権限およびポリシー

ソリューションを導入するために必要な OCI の権限とポリシーは次のとおりです。

1. ユーザーおよびグループ



(注) ユーザーとグループを作成するには、OCI ユーザーまたはテナンシー管理者である必要があります。

Oracle Cloud Infrastructure のユーザーアカウントと、そのユーザーアカウントが属するグループを作成します。ユーザーアカウントを持つ関連グループが存在する場合は、作成する必要はありません。ユーザーとグループの作成手順については、「[グループとユーザーの作成](#)」を参照してください。

2. グループ ポリシー

ポリシーを作成したら、それをグループにマッピングする必要があります。ポリシーを作成するには、[OCI] > [アイデンティティとセキュリティ (Identity & Security)] > [ポリシー (Policies)] > [ポリシーの作成 (Create Policy)] に移動します。次のポリシーを作成して、目的のグループに追加します。

- グループ <Group_Name> がコンパートメント <Compartment_Name> でメトリックを使用することを許可します。
- グループ <Group_Name> がコンパートメント <Compartment_Name> でアラームを管理することを許可します。
- グループ <Group_Name> がコンパートメント <Compartment_Name> で ONS トピックを管理することを許可します。
- グループ <Group_Name> がコンパートメント <Compartment_Name> でメトリックを検査することを許可します。
- グループ <Group_Name> がコンパートメント <Compartment_Name> でメトリックを読み取ることを許可します。

- グループ <Group_Name> がコンパートメント <Compartment_Name> でタグの名前空間を使用することを許可します。
- グループ <Group_Name> がコンパートメント <Compartment_Name> でロググループを読み取ることが許可されます。
- グループ <Group_Name> がインスタンスプールコンパートメント <Compartment_Name> を使用することを許可します。
- グループ <Group_Name> がテナントでクラウドシェルを使用することを許可します。
- グループ <Group_Name> がテナントのオブジェクトストレージ名前空間を読み取ることが許可されます。
- グループ <Group_Name> がテナント内のリポジトリを管理することを許可します。



(注) テナントレベルでポリシーを作成することもできます。ユーザーの責任と判断のもとで、すべての権限を自由に指定できます。

3. Oracle 関数の権限

Oracle 関数が別の Oracle Cloud Infrastructure リソースにアクセスできるようにするには、関数をダイナミックグループに含めてから、そのリソースへのダイナミックグループアクセスを許可するポリシーを作成します。

4. ダイナミックグループの作成

ダイナミックグループを作成するには、[OCI]>[アイデンティティとセキュリティ (Identity & Security)]>ダイナミックグループ (Dynamic Group)]>[ダイナミックグループの作成 (Create Dynamic Group)]に移動します。

ダイナミックグループの作成時に次のルールを指定します。

```
ALL {resource.type = 'fnfunc', resource.compartment.id = '<Your_Compartment_OCID>'}
```

ダイナミックグループの詳細については、次を参照してください。

- <https://docs.oracle.com/en-us/iaas/Content/Functions/Tasks/functionsaccessingociresources.htm>
- <https://docs.oracle.com/en-us/iaas/Content/Identity/Tasks/managingdynamicgroups.htm>

5. ダイナミックグループのポリシーの作成

ポリシーを追加するには、[OCI]>[アイデンティティとセキュリティ (Identity & Security)]>[ポリシー (Policies)]>[ポリシーの作成 (Create Policy)]に移動します。次のポリシーをグループに追加します。

```
Allow dynamic-group <Dynamic_Group_Name> to manage all-resources in compartment <Compartment_OCID>
```

GitHub からのファイルのダウンロード

FTDv : OCI Auto Scale ソリューションは、[GitHub](#) リポジトリ形式で配布されます。リポジトリからファイルをプルまたはダウンロードできます。

Python3 環境

`make.py` ファイルは、複製されたリポジトリ内にあります。このプログラムは、Oracle 関数とテンプレートファイルを ZIP ファイルに圧縮します。それらをターゲットフォルダーにコピーします。これらのタスクを実行するには、Python 3 環境が設定されている必要があります。



(注) この Python スクリプトは Linux 環境でのみ使用できます。

インフラストラクチャ設定

次を設定する必要があります。

1. VCN

FTDv アプリケーションの要件に応じて VCN を作成します。インターネットへのルートが割り当てられたサブネットが 1 つ以上あるインターネットゲートウェイを備えた VPC を作成します。

VCN の作成については、「<https://docs.oracle.com/en-us/iaas/Content/GSG/Tasks/creatingnetwork.htm>」を参照してください。

2. アプリケーションサブネット

FTDv アプリケーションの要件に応じてサブネットを作成します。このユースケースに従ってソリューションを導入するには、FTDv インスタンスの運用に 4 つのサブネットが必要です。

サブネットの作成については、https://docs.oracle.com/en-us/iaas/Content/Network/Tasks/managingVCNs_topic-Overview_of_VCNs_and_Subnets.htm# を参照してください。

3. 外部サブネット

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」のルートが必要です。このサブネットには、Cisco FTDv の外部インターフェイスとインターネット向けロードバランサが含まれています。アウトバウンドトラフィック用に NAT ゲートウェイが追加されていることを確認します。

詳細については、次のマニュアルを参照してください。

- <https://docs.oracle.com/en-us/iaas/Content/Network/Tasks/managingIGs.htm>
- https://docs.oracle.com/en-us/iaas/Content/Network/Tasks/NATgateway.htm#To_create_a_NAT_gateway

4. 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。



- (注) FTDv 正常性プローブの場合、ポート 80 を介してメタデータサーバー (169.254.169.254) に到達できます。

5. 管理サブネット

管理サブネットは、FTDv への SSH 接続をサポートするようにパブリックにする必要があります。

6. セキュリティグループ : FTDv インスタンスのネットワークセキュリティグループ

Oracle 関数 (同じ VCN 内) が FTDv の管理アドレスへに SSH 接続できるように、FTDv インスタンスのセキュリティグループを設定します。

7. オブジェクトストレージの名前空間

このオブジェクトストレージの名前空間は、`configuration.txt` ファイルを持つ静的 Web サイトをホストするために使用されます。`configuration.txt` ファイルの事前認証済みリクエストを作成する必要があります。この事前認証された URL は、テンプレートの展開時に使用されます。



- (注) アップロードされた次の設定に、HTTP URL を介して FTDv インスタンスからアクセスできることを確認します。

```
FTDv を起動すると、$ copy /noconfirm <configuration.txt file's pre-authenticated request URL > disk0:Connfiguration.txt コマンドが実行されます。
```

このコマンドにより、FTDv の起動を `configuration.txt` ファイルで設定できるようになります。

Secure Firewall Management Center の前提条件

Threat Defense Virtual デバイスを管理するには、フル機能のマルチデバイスマネージャである Secure Firewall 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 FMC を登録して通信します。

複数のデバイスにポリシーを展開して、更新をインストールするには、Threat Defense Virtual の設定と管理に必要なデバイスグループをはじめとするオブジェクトを作成します。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。

後続の項では、Management Center を準備するための基本的な手順の概要を説明します。手順の詳細については、『*Firepower Management Center Configuration Guide*』を参照してください。Management Center を準備する際は、次の情報を必ず記録してください。

- Secure Firewall Management Center のパブリック IP アドレス

- ユーザー名とパスワード（メモリベースのスケールリングが有効になっている場合は、2つのユーザーログイン情報を指定する必要があります）
- セキュリティゾーン名
- Secure Firewall Management Center のアクセスポリシー名
- Secure Firewall Management Center の NAT ポリシー名
- Device Group Name

Secure Firewall Management Center でのユーザーの作成

Auto Scale Manager だけが使用する管理者権限を持つ Secure Firewall Management Center で新規ユーザーを作成します。



(注) 他の FMC セッションとの競合を防ぐために、Threat Defense Virtual Auto Scale ソリューション専用の Secure Firewall Management Center ユーザーアカウントを持つ必要があります。

管理者権限を持つ Secure Firewall Management Center で新しいユーザーを作成します。[システム (System)] > [ユーザー (Users)] の順にクリックし、[ユーザーの作成 (Create User)] をクリックします。ユーザー名は、次のように Linux に対して有効である必要があります。

- 英数字、ハイフン (-)、およびアンダースコア (_) が使用可で、最大 32 文字
- すべて小文字
- ハイフン (-) から始めることはできず、英字は必須。ピリオド (.)、アットマーク (@)、スラッシュ (/) は使用不可

使用環境に必要なユーザーオプションを入力します。詳細については、『*Firepower Management Center Configuration Guide*』を参照してください。

デバイス グループの作成

デバイスグループにより、複数デバイスへのポリシーの割り当てとインストール更新が簡単にできます。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。

ステップ 1 [デバイス (Devices)] > [デバイス管理 (Device Management)] の順に選択します。

ステップ 2 [追加 (Add)] ドロップダウンメニューから、[グループの追加 (Add Group)] を選択します。

ステップ 3 デバイスグループ名を入力します。

ステップ4 [OK] をクリックしてデバイスグループを作成します。

ネットワークとホストオブジェクトの作成

Threat Defense Virtual の設定に使用する以下のオブジェクトを作成します。

- ステップ1 名前が *oci-metadata-server* で IP が *169.254.169.254* のホストを作成します。
- ステップ2 名前が *health-check-port* で値が *8080* のポートを作成します。必要に応じて他にもポートを作成します。
- ステップ3 内部インターフェイスを作成し、[インターフェイス (Interface)] > [セキュリティゾーン (Security Zone)] を選択します。[ルーテッド (Routed)] をタイプとして選択します。インターフェイス名 (*inside-sz* など) を指定します。
- ステップ4 外部インターフェイスを作成し、[インターフェイス (Interface)] > [セキュリティゾーン (Security Zone)] を選択します。[ルーテッド (Routed)] をタイプとして選択します。インターフェイス名 (*outside-sz* など) を指定します。

NAT ポリシーの作成

NAT ポリシーを作成し、外部インターフェイスからアプリケーションにトラフィックを転送するために必要な NAT ルールを作成します。次に、このポリシーを Auto Scale 用に作成したデバイスグループにアタッチします。

- ステップ1 [デバイス (Devices)] > [NAT] の順に選択します。
- ステップ2 [新しいポリシー (New Policy)] ドロップダウンリストで、[Threat Defense NAT] を選択します。
- ステップ3 [名前 (Name)] に一意の名前を入力します。
- ステップ4 必要に応じて、[説明 (Description)] を入力します。
- ステップ5 NAT ルールを設定します。NAT ルールの作成および NAT ポリシーの適用方法のガイドラインについては、『[Secure Firewall Management Center Device Configuration Guide](#)』の「[Configure NAT for Threat Defense](#)」を参照してください。[英語] 次の図に、ルールを設定する際の基本的なアプローチを示します。

図 4: NAT ルール

#	Direction	Type	Source Interface Objects	Destination Interface Objects	Original Sources	Original Destinations	Original Services	Translated Sources	Translated Destinations	Translated Services	Options
▼ NAT Rules Before											
1		Static	outside-zone	inside-zone	any-ipv4	Interface	Original oci-health-check	Interface	oci-metadata-server	Original HTTP	Dns:false
2		Static	inside-zone	outside-zone	any-ipv4	Interface	Original oci-health-check	Interface	oci-metadata-server	Original HTTP	Dns:false
3		Static	outside-zone	inside-zone	oci-marketplace-outside-sub	Interface	Interface	Interface	oci-inside-app-server	Interface	Dns:false
4		Static	inside-zone	outside-zone	oci-marketplace-inside-subn	Interface	Interface	Interface	external-server	Interface	Dns:false
▼ Auto NAT Rules											
▼ NAT Rules After											

ステップ6 [保存 (Save)] をクリックします。

NAT ルールの作成

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。詳細については、『[Secure Firewall Management Center Device Configuration Guide](#)』の「[Configure NAT for Threat Defense](#)」を参照してください。[英語]

NAT ポリシーに必要な次の 2 つの必須ルールを設定します。

ステップ 1 インバウンドヘルスチェックでは、次の NAT ルールを設定します。

- 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
- 元の宛先 (Original Destinations) : 送信元インターフェイスの IP
- 元の送信元ポート (Original source port) : デフォルト
- 元の宛先ポート (Original-destination-port) : health-check-port
- 変換済みの送信元 (Translated-sources) : 宛先インターフェイスの IP
- 変換済み宛先 (Translated-destination) : oci-metadata-server
- 変換済み送信元ポート (Translated source port) : デフォルト
- 変換済み宛先ポート (Translated-destination-port) : HTTP

次の図は、インバウンドヘルスチェックの NAT ルールを示しています。

図 5: インバウンドヘルス NAT ルール

Interface Objects	Translation	PAT Pool	Advanced
Original Packet			
Original Source:*	any-ipv4		
Original Destination:	Source Interface IP		
Original Source Port:			
Original Destination Port:	oci-health-check		
Translated Packet			
Translated Source:	Destination Interface IP		
Translated Destination:	oci-metadata-server		
Translated Source Port:			
Translated Destination Port:	HTTP		

ステップ 2 アウトバウンドヘルスチェックでは、次の NAT ルールを設定します。

- 送信元ゾーン (Source Zone) : 内部ゾーン
- 宛先ゾーン (Dest Zone) : 外部ゾーン

- 元の送信元 (Original-sources) : any-ipv4
- 元の宛先 (Original Destinations) : 送信元インターフェイスの IP
- 元の送信元ポート (Original source port) : デフォルト
- 元の宛先ポート (Original-destination-port) : health-check-port
- 変換済みの送信元 (Translated-sources) : 宛先インターフェイスの IP
- 変換済み宛先 (Translated-destination) : oci-metadata-server
- 変換済み送信元ポート (Translated source port) : デフォルト
- 変換済み宛先ポート (Translated-destination-port) : HTTP

次の図は、アウトバウンドヘルス チェックの NAT ルールを示しています。

図 6: アウトバウンドヘルス チェックの NAT ルール

Original Packet	Translated Packet
Original Source:* any-ipv4	Translated Source: Destination Interface IP
Original Destination: Source Interface IP	Translated Destination: oci-metadata-server
Original Source Port: (blank)	Translated Source Port: (blank)
Original Destination Port: oci-health-check	Translated Destination Port: 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、任意の NAT ルールをデータトラフィックに追加できます。

アクセスポリシーの作成

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デバイスに到達できるように、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 コントロール ポリシー内では、アクセスコントロールルールによって複数の管理対象デバイスでネットワークトラフィックを処理する詳細な方法が提供されます。効果的な導入を実現するには、ルールの適切な構成と順序付けが不可欠です。『[Secure Firewall Management Center Device Configuration Guide](#)』で「[Best Practices for Access Control Rules](#)」の項を参照してください。[英語]

[ポリシー割り当て (Policy Assignments)] を使用して、デバイスグループ (前提条件の一部として作成済み) をアクセスポリシーに割り当てます。

ステップ 1 [ポリシー (Policies)] > [アクセス制御 (Access Control)] を選択します。

ステップ2 [新しいポリシー (New Policy)] をクリックします。

ステップ3 [名前 (Name)] に一意の名前を入力し、オプションで [説明 (Description)] を入力します。

ステップ4 導入のセキュリティ設定とルールを設定します。詳細については、『[https://www.cisco.com/c/en/us/td/docs/security/secure-firewall/management-center/device-config/720/management-center-device-config-72/access-overview.html#ID-2176-00000005Secure Firewall Management Center Device Configuration Guide](https://www.cisco.com/c/en/us/td/docs/security/secure-firewall/management-center/device-config/720/management-center-device-config-72/access-overview.html#ID-2176-00000005Secure%20Firewall%20Management%20Center%20Device%20Configuration%20Guide)』の「Access Control」を参照してください。[英語]

パスワードの暗号化



(注) この手順の詳細については、「[Vault とシークレットの作成](#)」を参照してください。

FTDv のパスワードは、自動スケーリング中に使用されるすべての FTDv インスタンスを設定するために使用されます。また、いくつかの設定目的で Rest API 呼び出すための接続を作成するために使用されます。

したがって、パスワードを時々保存して処理する必要があります。頻繁な変更と脆弱性のため、プレーンテキスト形式での「パスワードの編集や保存はできません。パスワードには、暗号化された形式のみを使用する必要があります。

暗号化された形式のパスワードを取得するには、次の手順を実行します。

ステップ1 Vault を作成します。

OCI Vault は、マスター暗号化キーを安全に作成および保存するサービスと、それらを使用する際に暗号化および復号化する方法を提供します。したがって、Vault は、自動スケールソリューションの残りの部分と同じコンパートメントに作成する必要があります（まだ作成していない場合）。

[OCI] > [アイデンティティとセキュリティ (Identity & Security)] > [Vault] > [新規 Vault の選択または作成 (Choose or Create New Vault)] に移動します。

。

ステップ2 マスター暗号化キーを作成します。

プレーンテキストのパスワードを暗号化するには、マスター暗号化キーが 1 つ必要です。

[OCI] > [アイデンティティとセキュリティ (Identity & Security)] > [Vault] > [キーの選択または作成 (Choose or Create Key)] に移動します。

任意のビット長で、指定されたアルゴリズムのいずれかから任意のキーを選択します。

1. AES : 128、192、256
2. RSA : 2048、3072、4096
3. ECDSA : 256、384、521

図 7: キーの作成

ステップ 3 暗号化されたパスワードを作成します。

1. [OCI] > [CloudShell (OCI Cloud Terminal)] を開く (Open CloudShell (OCI Cloud Terminal)) に移動します。
2. <Password> をお使いのパスワードに置き換えて、次のコマンドを実行します。

```
echo -n '<Password>' | base64
```

3. 選択した Vault から、暗号化エンドポイントとマスター暗号化キーの OCID をコピーします。次のように値を置き換えてから、暗号化コマンドを実行します。
 - KEY_OCID : キーの OCID
 - Cryptographic_Endpoint_URL : Vault の暗号化エンドポイント URL
 - Password : パスワード

暗号化コマンド

```
oci kms crypto encrypt --key-id Key_OCID --endpoint  
Cryptographic_Endpoint_URL --plaintext <base64-value-of-password>
```

4. 上記のコマンドの出力から暗号文をコピーし、必要に応じて使用します。

Threat Defense Virtual の構成ファイルの準備

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。

ステップ 1 展開する前に、次の入力パラメータを収集します。

パラメータ	データタイプ	説明
tenancy_ocid	文字列	アカウントが属するテナントの OCID。テナントの OCID を見つける方法については、 こちら を参照してください。 テナントの OCID は ocid1.tenancy.oc1..<unique_ID> のようになります。
region	文字列	リソースを作成するリージョンの一意の識別子。 例：us-phoenix-1、us-ashburn-1
lb_size	文字列	事前にプロビジョニングする外部および内部ロードバランサの合計帯域幅（入力および出力）を決定するテンプレート。 サポートされる値：100 Mbps、10 Mbps、10 Mbps-Micro、400 Mbps、8000 Mbps 例：100 Mbps
availability_domain	文字列	例：Tpeb:PHX-AD-1、Tpeb:PHX-AD-2 (注) 可用性システムのドメイン名を取得するには、 こちら を参照してください。
min_and_max_instance_count	カンマ区切り値	インスタンスプールに保持するインスタンスの最小数と最大数。 例：1,5
autoscale_group_prefix	文字列	テンプレートを使用して作成したリソースの名前に付けるプレフィックス。たとえば、リソースプレフィックスとして「autoscale」を指定すると、すべてのリソースはautoscale_resource1、autoscale_resource2 のように名前が付けられます。
mgmt_subnet_ocid	文字列	使用する管理サブネットの OCID。
inside_subnet_ocid	文字列	使用する内部サブネットの OCID。
outside_subnet_ocid	文字列	使用する外部サブネットの OCID。
mgmt_nsg_ocid	文字列	使用する管理サブネットのネットワークセキュリティグループの OCID。
inside_nsg_ocid	文字列	使用する内部サブネットのネットワークセキュリティグループの OCID。

パラメータ	データタイプ	説明
outside_nsg_ocid	文字列	使用する外部サブネットのネットワークセキュリティグループの OCID。
elb_listener_port	カンマ区切り値	外部ロードバランサリスナーの通信ポートのリスト。 例：80
ilb_listener_port	カンマ区切り値	内部ロードバランサリスナーの通信ポートのリスト。 例：80
health_check_port	文字列	ヘルスチェックを実行するロードバランサーのバックエンドサーバーポート。 例：8080
instance_shape	文字列	作成するインスタンスのシェープ。シェイプにより、インスタンスに割り当てられる CPU の数、メモリの量、およびその他のリソースが決定されます。 サポートされているシェープ：「VM.Standard2.4」および「VM.Standard2.8」
lb_bs_policy	文字列	内部および外部ロードバランサのバックエンドセットに使用するロードバランサポリシー。ロードバランサポリシーの仕組みについては、 こちら を参照してください。 サポートされている値：「ROUND_ROBIN」、 「LEAST_CONNECTIONS」、 「IP_HASH」
image_name	文字列	インスタンスの構成に使用するマーケットプレイスのイメージ名。 デフォルト値：「Cisco Firepower NGFW 仮想ファイアウォール (NGFWv) (Cisco Firepower NGFW virtual firewall (NGFWv))」 (注) カスタムイメージを展開する場合は、 custom_image_ocid パラメータを設定する必要があります。

パラメータ	データタイプ	説明
scaling_thresholds	カンマ区切り値	スケールインとスケールアウトで使用する CPU 使用率のしきい値。スケールインとスケールアウトのしきい値をカンマで区切って入力します。 例：15,50 15 はスケールインのしきい値、50 はスケールアウトのしきい値です。
compartment_id	文字列	リソースを作成するコンパートメントの OCID。 例：ocid1.compartment.oc1..<unique_ID>
compartment_name	文字列	コンパートメント名
custom_image_ocid	文字列	マーケットプレイスイメージを使用しない場合に、インスタンス構成に使用するカスタムイメージの OCID。 (注) <code>custom_image_ocid</code> はオプションパラメータです
ftdv_password	文字列	Threat Defense Virtual を構成するために SSH 接続する際の、Threat Defense Virtual の暗号化形式のパスワード。パスワードを暗号化する方法については、 コンフィギュレーションガイド を使用するか、 こちら を参照してください。
ftdv_license_type	文字列	Threat Defense Virtual ライセンスタイプ (BYOL または PAYG)。現在、BYOL がサポートされています。
cryptographic_endpoint	文字列	暗号化エンドポイントは、パスワードの復号化に使用される URL です。Vault で検索できます。
master_encryption_key_id	文字列	パスワードの暗号化に使用されたキーの OCID。Vault で検索できます。 (注) <code>master_encryption_key_id</code> と <code>cryptographic_endpoint</code> の両方が同じ Vault に属している必要があります。

パラメータ	データタイプ	説明
fmc_ip	文字列	Secure Firewall Management Center の IP アドレス。 カスタマーが Threat Defense Virtual インスタンスを管理するために使用する Management Center の IP。 (注) <i>Management Center</i> の IP は、 <i>Threat Defense Virtual</i> と同じサブネットにある場合にのみプライベート IP を使用できます。それ以外の場合は、パブリック IP を使用する必要があります。
fmc_username	文字列	Management Center アカウントのユーザー名このユーザー名は、Management Center にログインして、新しい Threat Defense Virtual インスタンスの起動のたびに設定で使用されます。
fmc_password	文字列	暗号化された形式の Management Center のパスワード。パスワードを暗号化する手順については、 こちら を参照してください。
fmc_device_group_name	文字列	Management Center にデバイスグループがあり、この Auto Scale ソリューションのすべての Threat Defense Virtual 部分はそのグループに追加されている必要があります。これにより、同じポリシーと構成をそれらのすべてに適用できます。
enable_memory_based_scaling	Bool	Secure Firewall Management Center Virtual から Threat Defense Virtual メモリ使用量を公開します。このフラグを有効にすることで、メモリ使用率にも基づいてスケーリングを実行できます。デフォルトでは、CPU 使用率が使用されます。
fmc_metrics_username	文字列	enable_memory_based_scaling フラグを有効にしてメモリ使用率を選択した場合、実行中のすべての Threat Defense Virtual インスタンスからメモリ使用量をプルするために継続的に使用されるため、追加の Management Center ユーザー アカウントが必要です。
fmc_metrics_password	文字列	暗号化形式の追加の Management Center アカウントのパスワード。パスワードを暗号化する手順については、 こちら を参照してください。

パラメータ	データタイプ	説明
プロファイル名 (Profile Name)		OCI のユーザーのプロファイル名です。ユーザーのプロファイルセクションの下にあります。例： 「oracleidentitycloudservice/ <user>@<mail>.com」
オブジェクトストレージの名前空間		テナントの作成時に作成される一意の識別子です。 [OCI]>[管理 (Administration)]>[テナントの詳細 (Tenancy Details)] に移動します。
認証トークン (Authorization Token)		これは、OCI コンテナレジストリに Oracle 関数をプッシュすることを許可する Docker ログイン用のパスワードとして使用されます。 [OCI]>[アイデンティティ (Identity)]>[ユーザー (Users)]>[ユーザーの詳細 (User Details)]>[認証トークン (Auth Tokens)]>[トークンの生成 (Generate Token)] に移動します。

ステップ 2 次の内容の *Configuration.json* ファイルを作成します。

```
{
  "licenseCaps": ["BASE", "MALWARE", "THREAT"],
  "performanceTier": "FTDv30",
  "fmcIpforDeviceReg": "DONTRESOLVE",
  "RegistrationId": "cisco",
  "NatId": "cisco",
  "fmcAccessPolicyName": "<autoscale-access-policy-name>",
  "fmcNatPolicyName": "<autoscale-nat-policy-name>",
  "fmcInsideNicName": "inside",
  "fmcOutsideNicName": "outside",
  "fmcInsideNic": "GigabitEthernet0/0",
  "fmcOutsideNic": "GigabitEthernet0/1",
  "fmcOutsideZone": "<outside-zone-name>",
  "fmcInsideZone": "<inside-zone-name>",
  "MetadataServerObjectName": "oci-metadata-server",
  "interfaceConfig": [
    {
      "managementOnly": "false",
      "MTU": "1500",
      "securityZone": {
        "name": "inside-zone"
      },
      "mode": "NONE",
      "ifname": "inside",
      "name": "GigabitEthernet0/0"
    },
    {
      "managementOnly": "false",
      "MTU": "1500",
      "securityZone": {
        "name": "outside-zone"
      },
      "mode": "NONE",
      "ifname": "outside",
      "name": "GigabitEthernet0/1"
    }
  ],
  "trafficRoutes": [
```

```

    {
      "interface": "outside",
      "network": "any-ipv4",
      "gateway": "",
      "metric": "2"
    },
    {
      "interface": "inside",
      "network": "oci-metadata-server",
      "gateway": "",
      "metric": "1"
    }
  ]
}

```

ステップ 3 *Configuration.json* を構成設定で更新します。

ステップ 4 構成ファイルをオブジェクトストレージスペースにアップロードします。

configuration.txt ファイルは、ユーザーが作成したオブジェクトストレージスペースにアップロードする必要があり、アップロードしたファイルに対する事前認証リクエストを作成する必要があります。

(注) スタックの展開で、*configuration.txt* の事前認証済みリクエスト URL が使用されていることを確認します。

(注) OCI で事前認証済み URL を作成するときに有効期限を定義する必要があります。ソリューションの実行中に期限切れにならないように、この期間を十分に長くしてください。

ステップ 5 Zip ファイルを作成します。

make.py ファイルは、複製されたリポジトリ内にあります。python3 *make.py build* コマンドを実行して、zip ファイルを作成します。対象フォルダには以下のファイルがあります。

```

Wed Apr 21 09:35 AM [sumis@SUMIS-M-41KG target]$ tree -A
.
├── Oracle-Functions.zip
├── README.md
├── asav_configuration.txt
├── deploy_oracle_functions_cloudshell.py
├── template1.zip
└── template2.zip

```

OCI への Auto Scale の展開

展開の前提条件となる手順を完了したら、OCI スタックの作成を開始します。[手動展開](#)を実行するか、[クラウドシェルを使用した導入](#) () を実行できます。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。

手動展開

エンドツーエンドの Auto Scale ソリューションの展開は、次の 3 つの手順で構成されます。
[Terraform Template-1 スタックの展開](#)、[Oracle 関数の展開](#)、次いで [Terraform Template-2 の展開](#)

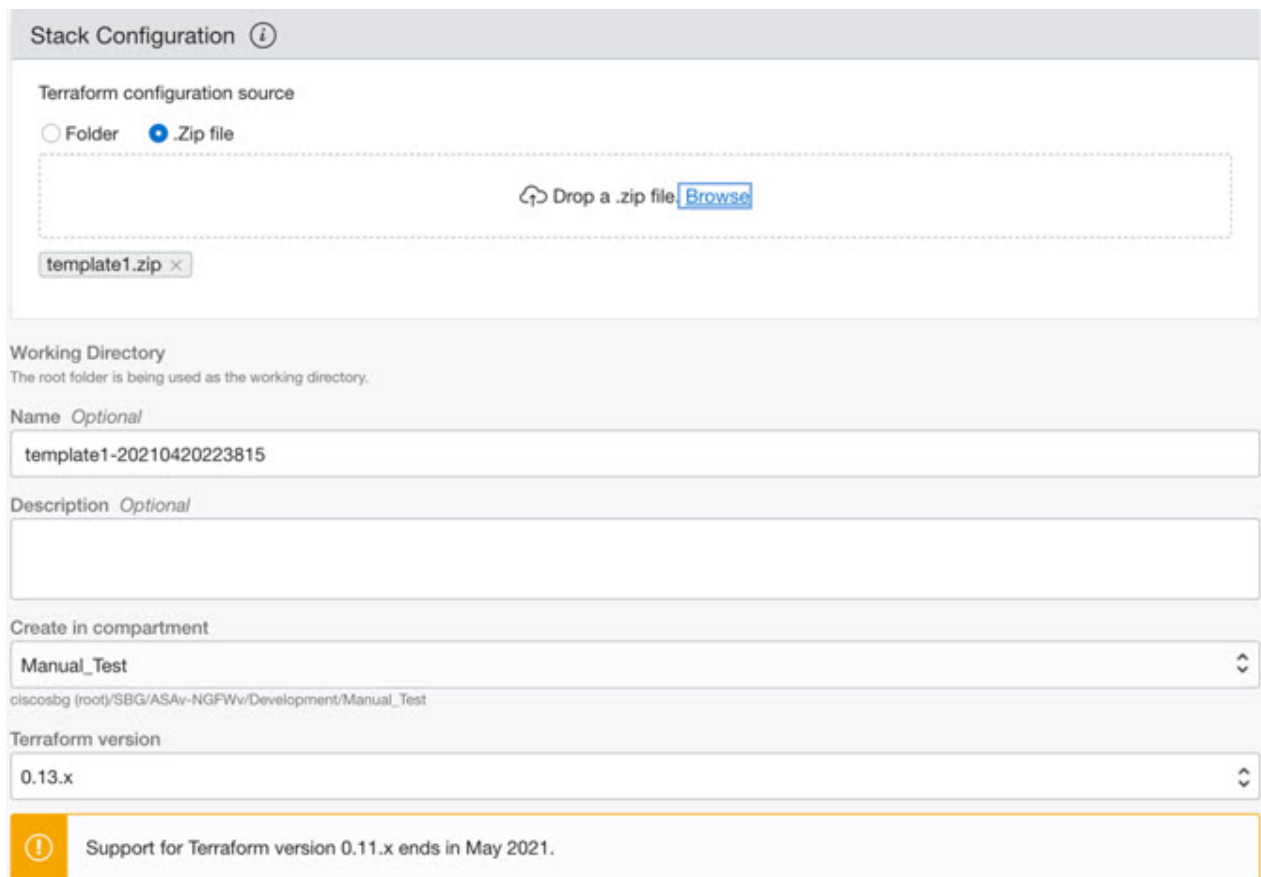
Terraform Template-1 スタックの展開

ステップ 1 OCI ポータルにログインします。

地域は、画面の右上隅に表示されます。目的の地域内に存在していることを確認してください。

ステップ 2 [デベロッパーサービス (Developer Service)] > [リソースマネージャ (Resource Manager)] > [スタック (Stack)] > [スタックの作成 (Create Stack)] の順に選択します。

[マイ設定 (My Configuration)] を選択し、次の図に示すように、対象フォルダ内にある *Terraform template1.zip* ファイルを Terraform の設定ソースとして選択します。



Stack Configuration ⓘ

Terraform configuration source

Folder .Zip file

Drop a .zip file [Browse](#)

template1.zip ×

Working Directory
The root folder is being used as the working directory.

Name *Optional*
template1-20210420223815

Description *Optional*

Create in compartment
Manual_Test
ciscosbg (root)/SBG/ASAv-NGFWv/Development/Manual_Test

Terraform version
0.13.x

ⓘ Support for Terraform version 0.11.x ends in May 2021.

ステップ 3 [トランスフォームバージョン (Transform version)] ドロップダウンリストで、0.13.x または 0.14.x を選択します。

ステップ 4 次の手順では、[ステップ 1](#) で収集した詳細情報をすべて入力します。

(注) 有効な入力パラメータを入力してください。そうしないと、以降の手順でスタックの展開に失敗する可能性があります。

ステップ 5 次の手順で[Terraformアクション (Terraform Actions)]>[適用 (Apply)]を選択します。

正常に展開されたら、Oracle 関数の展開に進みます。

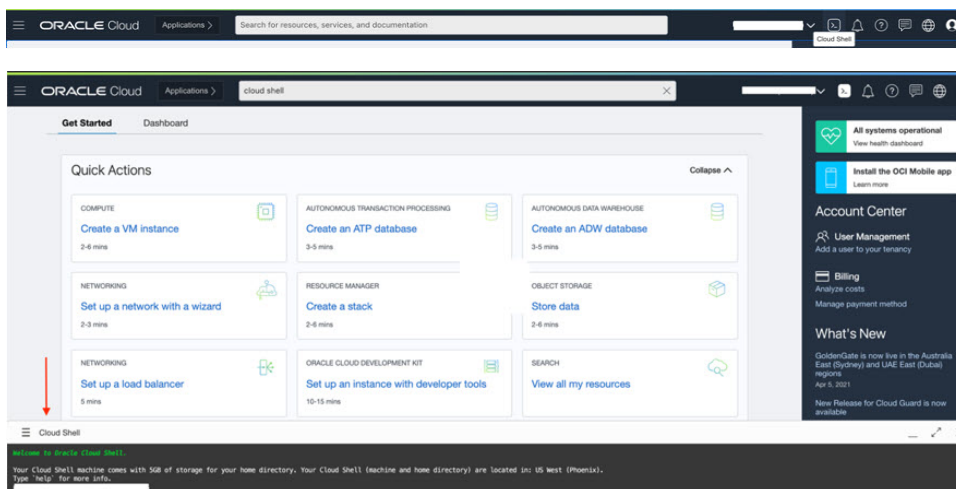
Oracle 関数の展開



(注) この手順は、*Terraform Template-1* の導入が成功した後にのみ実行する必要があります。

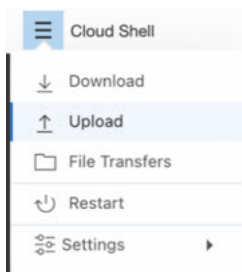
OCI では、Oracle 関数は Docker イメージとしてアップロードされ、OCI コンテナレジストリに保存されます。Oracle 関数は、導入時に OCI アプリケーション (Terraform Template-1 で作成) の 1 つにプッシュする必要があります。

ステップ 1 OCI のクラウドシェルを開きます。



ステップ 2 `deploy_oracle_functions_cloudshell.py` と `Oracle-Functions.zip` をアップロードします。

クラウドシェルのハンバーガーメニューから [アップロード (Upload)] を選択します。



ステップ3 ls コマンドを使用してファイルを確認します。

```
$ ls
Deploy_Oracle_Functions.py  Oracle-Functions.zip
```

ステップ4 python3 Deploy_Oracle_Functions.py -h を実行します。以下の図に示すように、deploy_oracle_functions_cloudshell.py スクリプトには、いくつかの入力パラメータが必要です。詳細は help 引数を使用して確認できます。

```
$ python3 Deploy_Oracle_Functions.py -h
usage: Deploy_Oracle_Functions.py [-h] -a -r -p -c -o -t

*** Script to deploy Oracle Function for OCI ASAv Autoscale Solution ***


Instruction to find values of required arguments:
Application Name: Name of Application created by first Terraform Template
Region Identifier: OCI -> Administration -> Region Management
Profile Name: OCI -> Profile
Compartment OCID: OCI -> Identity -> Compartment -> Compartment Details
Object Storage Namespace: OCI -> Administration -> Tenancy Details
Authorization Token: OCI -> Identity -> Users -> User Details -> Auth Tokens -> Generate Token

optional arguments:
  -h, --help  show this help message and exit
  -a          Name of Application in OCI to which functions will be deployed
  -r          Region Identifier
  -p          Profile Name of User
  -c          Compartment OCID
  -o          Object Storage Namespace
  -t          Authorization Token for Docker Login (*Please Put in Quotes)
```

スクリプトを実行するには、次の引数を渡します。

表 3: 引数と詳細

引数	特記事項
アプリケーション	Terraform Template-1 の導入で作成した OCI アプリケーションの名前です。この値は、Template-1 で付与された「autoscale_group_prefix」とサフィックス「_application」を組み合わせたものです。
リージョン識別子 (Region Identifier)	リージョン識別子は、さまざまな地域の OCI で固定された地域コードワードです。 例：フェニックスの場合は「us-phoenix-1」、メルボルンの場合は「ap-melbourne-1」。 すべてのリージョンとそのリージョン識別子のリストを取得するには、[OCI] > [管理 (Administration)] > [リージョン管理 (Region Management)] に移動します。

引数	特記事項
プロファイル名	OCI のシンプルなユーザープロファイル名です。 例 : <code>oracleidentitycloudservice/<user> @<mail> .com</code> 名前は、ユーザーのプロファイルセクションの下にあります。
コンパートメント OCID (Compartment OCID)	これは、コンパートメントの OCID (Oracle Cloud 識別子) です。ユーザーが OCI アプリケーションを格納しているコンパートメントの OCID。 [OCI] > [アイデンティティ (Identity)] > [コンパートメント (Compartment)] > [コンパートメントの詳細 (Compartment Details)] に移動します。
オブジェクトストレージの名前空間	テナントの作成時に作成される一意の識別子です。 [OCI] > [管理 (Administration)] > [テナントの詳細 (Tenancy Details)] に移動します。
認証トークン (Authorization Token)	これは、OCI コンテナレジストリに Oracle 関数をプッシュすることを許可する Docker ログイン用のパスワードとして使用されます。導入スクリプトでトークンを引用符で囲んで指定します。 [OCI] > [アイデンティティ (Identity)] > [ユーザー (Users)] > [ユーザの詳細 (User Details)] > [認証トークン (Auth Tokens)] > [トークンの生成 (Generate Token)] に移動します。 何らかの理由でユーザーの詳細が表示されない場合は、 [開発者サービス (Developer services)] > [機能 (Functions)] をクリックします。Terraform Template-1 で作成したアプリケーションに移動します。[利用を開始する (Getting Started)] をクリックし、[クラウドシェルの設定 (Cloud Shell Setup)] を選択すると、手順を進めていく中で、以下に示すように認証トークンを生成するためのリンクが表示されます。 

ステップ 5 有効な入力引数を渡して、`python3 Deploy_Oracle_Functions.py` コマンドを実行します。すべての機能を展開するには時間がかかります。その後、ファイルを削除してクラウドシェルを閉じることができます。

Terraform Template-2 の展開

Template-2 は、アラーム、関数を呼び出すための ONS トピックなど、アラーム作成に関連するリソースを展開します。Template-2 の展開は、Terraform Template-1 の展開に似ています。

ステップ 1 OCI ポータルにログインします。

地域は、画面の右上隅に表示されます。目的の地域内に存在していることを確認してください。

ステップ 2 [デベロッパーサービス (Developer Service)] > [リソースマネージャ (Resource Manager)] > [スタック (Stack)] > [スタックの作成 (Create Stack)] の順に選択します。

Terraform 設定のソースとして、ターゲットフォルダにある *Terraform template template2.zip* を選択します。

ステップ 3 次のステップで、**Terraform アクション (Terraform Actions)** > [適用 (Apply)] をクリックします。

クラウドシェルを使用した導入

導入のオーバーヘッドを回避するために、簡単なエンドツーエンドの導入スクリプトを呼び出して、自動スケールソリューション (terraform template1、template2、および Oracle 関数) を導入できます。

ステップ 1 対象フォルダ内にある *ftdv_autoscale_deploy.zip* ファイルをクラウドシェルにアップロードして、ファイルを抽出します。

```

☰ Cloud Shell

sumis@cloudshell:~ (us-phoenix-1)$ ls -ltrh
total 152K
-rw-r--r--. 1 sumis oci 151K Jun  9 07:25 ftdv_autoscale_deploy.zip
sumis@cloudshell:~ (us-phoenix-1)$ unzip ftdv_autoscale_deploy.zip
Archive:  ftdv_autoscale_deploy.zip
  extracting: template1.zip
  extracting: template2.zip
  extracting: Oracle-Functions.zip
    inflating: oci_ftdv_autoscale_deployment.py
    inflating: oci_ftdv_autoscale_tearardown.py
    inflating: deployment_parameters.json
    inflating: teardown_parameters.json
sumis@cloudshell:~ (us-phoenix-1)$ ls -ltrh
total 344K
-rw-r--r--. 1 sumis oci 2.7K Jun  9 07:19 template2.zip
-rw-r--r--. 1 sumis oci 5.0K Jun  9 07:19 template1.zip
-rw-r--r--. 1 sumis oci  70 Jun  9 07:19 teardown_parameters.json
-rw-r--r--. 1 sumis oci 133K Jun  9 07:19 Oracle-Functions.zip
-rw-r--r--. 1 sumis oci  7.1K Jun  9 07:19 oci_ftdv_autoscale_tearardown.py
-rw-r--r--. 1 sumis oci  25K Jun  9 07:19 oci_ftdv_autoscale_deployment.py
-rw-r--r--. 1 sumis oci  2.8K Jun  9 07:19 deployment_parameters.json
-rw-r--r--. 1 sumis oci 151K Jun  9 07:25 ftdv_autoscale_deploy.zip
sumis@cloudshell:~ (us-phoenix-1)$

```

ステップ 2 `python3 make.py build` コマンドを実行する前に、*deployment_parameters.json* の入力パラメータが更新されていることを確認してください。

- ステップ3** 自動スケールソリューションの導入を開始するには、クラウドシェルで `python3 oci_ftdv_autoscale_deployment.py` コマンドを実行します。
- ソリューションの展開が完了するまでに約 10 ~ 15 分かかります。

ソリューションの展開中にエラーが発生した場合、エラーログが保存されます。

展開の検証

すべてのリソースが展開され、Oracle 関数がアラームとイベントに接続されているかどうかを検証します。デフォルトでは、インスタンスプールのインスタンスの最小数と最大数はゼロです。OCI UI でインスタンスプールを編集して、必要な最小数と最大数に設定できます。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。この検証をポストすると、Threat Defense Virtual の実際の要件を展開できます。



-
- (注) OCI 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指定します。
-

Auto Scale のアップグレード

Auto Scale スタックのアップグレード

このリリースではアップグレードはサポートされていません。スタックを再導入する必要があります。

Threat Defense Virtual VM のアップグレード

このリリースでは、Threat Defense Virtual VM のアップグレードはサポートされていません。必要な Threat Defense Virtual イメージを使用してスタックを再導入する必要があります。

インスタンスプール

1. インスタンスプール内のインスタンスの最小数と最大数を変更するには、次の手順を実行します。

[デベロッパーサービス (Developer Services)] > [機能 (Function)] > [アプリケーション名 (Terraform template-1で作成済み) (Application Name(created by Terraform Template 1))] > [設定 (Configuration)] をクリックします。

`min_instance_count` と `max_instance_count` をそれぞれ変更します。

2. インスタンスの削除/終了は、スケールインと同等ではありません。インスタンスプール内のいずれかのインスタンスがスケールインアクションではなく外部アクションのために削除/終了された場合、インスタンスプールは自動的に新しいインスタンスを開始して回復します。
3. `Max_instance_count` では、スケールアウトアクションのしきい値制限を定義しますが、UI を介してインスタンスプールのインスタンス数を変更することでしきい値を上回ることができます。UI のインスタンス数が、OCI アプリケーションで設定された `max_instance_count` 未満であることを確認します。それ以外の場合は、適切なしきい値に増やします。
4. アプリケーションから直接インスタンスプール内のインスタンスの数を減らしても、プログラムで設定されたクリーンアップアクションは実行されません。両方のロードバランサからバックエンドがドレインおよび削除されないため、Threat Defense Virtual に供与されているライセンスは失われます。
5. 何らかの理由で、Threat Defense Virtual インスタンスに異常があり応答せず、一定期間 SSH 経由で到達できない場合、インスタンスがインスタンスプールから強制的に削除され、ライセンスが失われる可能性があります。

Oracle 関数

- Oracle 関数は、実際には Docker イメージです。Docker イメージは、OCI コンテナレジストリのルートディレクトリに保存されます。Docker イメージは削除しないでください。Auto Scale ソリューションで使用される関数も削除されます。
- Terraform Template-1 によって作成された OCI アプリケーションには、Oracle 関数が正しく動作するために必要な重要な環境変数が含まれています。必須でない限り、これらの環境変数の値もフォーマットも変更しないでください。加えられた変更は、新しいインスタンスにのみ反映されます。

ロードバランサのバックエンドセット

OCI でインスタンスプールにロードバランサを関連付ける場合、Threat Defense Virtual で管理インターフェースとして設定されたプライマリインターフェースを使用した方法のみサポートされています。したがって、内部インターフェイスは内部ロードバランサのバックエンドセットに紐づけられます。外部インターフェイスは、外部ロードバランサのバックエンドセットに紐づけられます。これらの IP はバックエンドセットに自動的に追加されたり、削除されたりしません。Auto Scale ソリューションでは、これら両方のタスクをプログラムで処理します。ただし、外部アクション、メンテナンス、トラブルシューティングの場合は、手動で実行する必要性が生じることがあります。

要件に応じて、リスナーとバックエンドセットを使用して、ロードバランサーで追加のポートを開くことができます。今後のインスタンス IP はバックエンドセットに自動的に追加されますが、既存のインスタンス IP は手動で追加する必要があります。

ロードバランサでのリスナーの追加

ロードバランサでポートをリスナーとして追加するには、**[OCI] > [ネットワーキング (Networking)] > [ロードバランサ (Load Balancer)] > [リスナー (Listener)] > [リスナーの作成 (Create Listener)]** に移動します。

バックエンドをバックエンドセットに登録

Threat Defense Virtual インスタンスをロードバランサに登録するには、Threat Defense Virtual インスタンスの外部インターフェイス IP を外部ロードバランサのバックエンドセットでバックエンドとして設定する必要があります。内部インターフェイス IP は、内部ロードバランサーのバックエンドセットでバックエンドとして設定する必要があります。使用しているポートがリスナーに追加されていることを確認してください。

OCI の Auto Scale 設定の削除

Terraform を使用して導入されたスタックは、OCI の Resource Manager を使用して、同じ方法で削除できます。スタックを削除すると、そのスタックによって作成されたすべてのリソースが削除され、これらのリソースに関連付けられているすべての情報が完全に削除されます。



- (注) スタックを削除する場合は、インスタンスプールのインスタンスの最小数を 0 にして、インスタンスが終了するまで待つことを推奨します。そうすることで、すべてのインスタンスの削除が容易になり、インスタンスが残りません。

手動による削除するか、クラウドシェルを使用した Auto Scale の削除を使用できます。

手動による削除

エンドツーエンドの Auto Scale ソリューションの削除は、次の 3 つの手順で構成されます。[Terraform Template-2 スタックの削除](#)、[Oracle 関数の削除](#)、次いで [Terraform Template-1 スタックの削除](#)

Terraform Template-2 スタックの削除

自動スケール設定を削除するには、最初に Terraform Template-2 スタックを削除する必要があります。

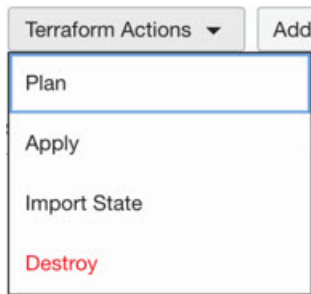
ステップ 1 OCI ポータルにログインします。

地域は、画面の右上隅に表示されます。目的の地域内に存在していることを確認してください。

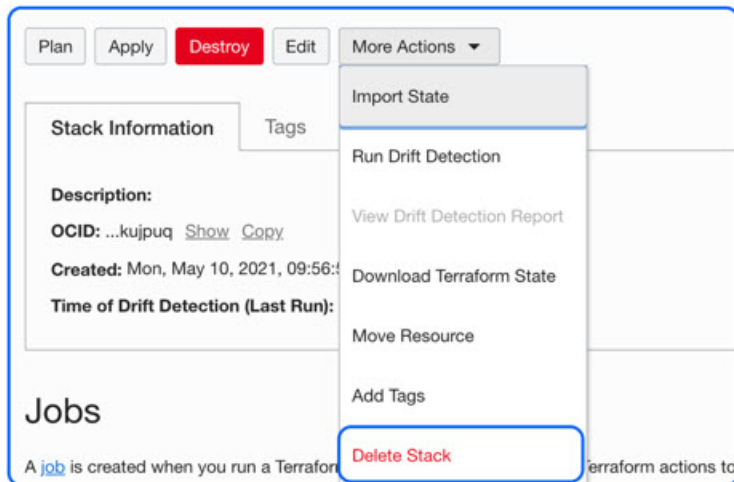
ステップ 2 **[デベロッパーサービス (Developer Service)] > [リソースマネージャ (Resource Manager)] > [スタック (Stack)]** の順に選択します。

ステップ 3 Terraform Template-2 によって作成されたスタックを選択し、次の図に示すように **[Terraform アクション (Terraform Actions)]** ドロップダウンメニューで **[破棄 (Destroy)]** を選択します。

Oracle 関数の削除



破棄ジョブが作成されます。リソースが順次削除されるまで時間がかかります。破棄ジョブが完了したら、下の図に示すようにスタックを削除できます。



ステップ 4 Oracle 関数の削除に進みます。

Oracle 関数の削除

Oracle 関数の展開は Terraform Template スタック展開の一部としてではなく、クラウドシェルを使用して個別にアップロードします。したがって、削除も Terraform スタックの削除ではサポートされていません。Terraform Template-1 によって作成された OCI アプリケーション内のすべての Oracle 関数を削除する必要があります。

ステップ 1 OCI ポータルにログインします。

地域は、画面の右上隅に表示されます。目的の地域内に存在していることを確認してください。

ステップ 2 [開発者サービス (Developer Services)] > [機能 (Functions)] の順に選択します。Template-1 スタックで作成されたアプリケーション名を選択します。

ステップ 3 このアプリケーション内で各機能にアクセスして削除します。

Terraform Template-1 スタックの削除



(注) Terraform Template-1 スタックの削除は、すべての Oracle 関数を削除した後のみ成功します。

Terraform Template-2 の削除と同じです。

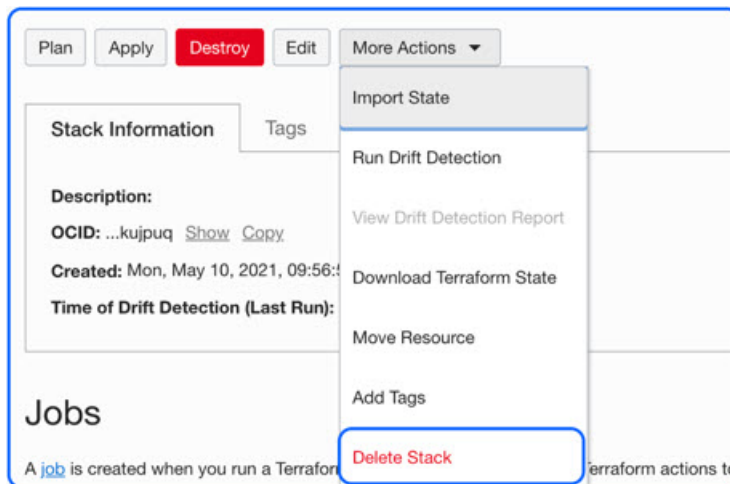
ステップ 1 OCI ポータルにログインします。

地域は、画面の右上隅に表示されます。目的の地域内に存在していることを確認してください。

ステップ 2 [デベロッパーサービス (Developer Service)] > [リソースマネージャ (Resource Manager)] > [スタック (Stack)] の順に選択します。

ステップ 3 Terraform Template-2 によって作成されたスタックを選択し、[Terraformアクション (Terraform Actions)] ドロップダウンメニューで[破棄 (Destroy)]を選択します。破棄ジョブが作成されます。リソースが順次削除されるまで時間がかかります。

ステップ 4 破棄ジョブが完了したら、下の図に示すように、[その他の操作 (More Actions)] ドロップダウンメニューからスタックを削除できます。



Terraform Template-1 スタックの削除が成功したら、すべてのリソースが削除され、残存しているリソースがないことを確認する必要があります。

クラウドシェルを使用した Auto Scale の削除

スクリプトを使用してスタックやオラクル関数を削除するには、コマンドシェルで `python3 oci_ftdv_autoscale_takedown.py` コマンドを実行します。スタックが手動で展開されている場合は、`stack1` と `stack2` のスタック ID を更新し、`takedown_parameters.json` ファイルのアプリケーション ID を更新します。

SSH を使用した Threat Defense Virtual インスタンスへの接続

UNIX スタイルのシステムから Threat Defense Virtual インスタンスに接続するには、SSH を使用してインスタンスにログインします。

ステップ 1 次のコマンドを使用して、ファイルの権限を設定し、自分だけがファイルを読み取れるようにします。

```
$ chmod 400 <private_key>
```

ここで、

<private_key> は、アクセスするインスタンスに関連付けられた秘密キーを含むファイルのフルパスと名前です。

ステップ 2 インスタンスにアクセスするには、次の SSH コマンドを使用します。

```
$ ssh -i <private_key> <username>@<public-ip-address>
```

<private_key> は、アクセスするインスタンスに関連付けられた秘密キーを含むファイルのフルパスと名前です。

<username> は、Threat Defense Virtual インスタンスのユーザー名です。

<public-ip-address> は、コンソールから取得したインスタンスの IP アドレスです。

OpenSSH を使用した Threat Defense Virtual インスタンスへの接続

Windows システムから Threat Defense Virtual インスタンスに接続するには、OpenSSH を使用してインスタンスにログインします。

ステップ 1 このキーペアを初めて使用する場合は、自分だけがファイルを読み取れるようにファイルの権限を設定する必要があります。

次の手順を実行します。

- Windows Explorer で、秘密キーファイルに移動し、ファイルを右クリックして[プロパティ (Properties)] をクリックします。
- [セキュリティ (Security)] タブで、[詳細設定 (Advanced)] をクリックします。
- [オーナー (Owner)] が自分のユーザーアカウントであることを確認します。

- d) [継承の無効化 (Disable Inheritance)] をクリックし、[継承された権限をこのオブジェクトの明示的な権限に変換する (Convert inherited permissions into explicit permissions on this object)] を選択します。
- e) 自分のユーザーアカウントではない各権限エントリを選択し、[削除 (Remove)] をクリックします。
- f) 自分のユーザーアカウントのアクセス権限が [フルコントロール (Full Control)] であることを確認します。
- g) 変更を保存します。

ステップ 2 インスタンスに接続するには、Windows PowerShell を開き、次のコマンドを実行します。

```
$ ssh -i <private_key> <username>@<public-ip-address>
```

ここで、

<private_key> は、アクセスするインスタンスに関連付けられた秘密キーを含むファイルのフルパスと名前です。

<username> は、Threat Defense Virtual インスタンスのユーザー名です。

<public-ip-address> は、コンソールから取得したインスタンスの IP アドレスです。

PuTTY を使用した Threat Defense Virtual インスタンスへの接続

PuTTY を使用して Windows システムから Threat Defense Virtual インスタンスに接続するには、次の手順を実行します。

ステップ 1 PuTTY を開きます。

ステップ 2 [カテゴリ (Category)] ペインで、[セッション (Session)] を選択し、次の内容を入力します。

- ホスト名または IP アドレス :

```
<username>@<public-ip-address>
```

ここで、

<username> は、Threat Defense Virtual インスタンスのユーザー名です。

<public-ip-address> は、コンソールから取得したインスタンスのパブリック IP アドレスです。

- ポート : 22
- 接続タイプ : SSH

ステップ 3 [カテゴリ (Category)] ペインで、[Window] を展開し、[変換 (Translation)] を選択します。

ステップ 4 [リモート文字セット (Remote character set)] ドロップダウンリストで、[UTF-8] を選択します。

Linux ベースのインスタンスでデフォルトのロケール設定は UTF-8 です。これにより、PuTTY は同じロケールを使用するように設定されます。

ステップ 5 [カテゴリ (Category)] ペインで、[接続 (Connection)]、[SSH] の順に展開し、[認証 (Auth)] をクリックします。

ステップ 6 [参照 (Browse)] をクリックして、秘密キーを選択します。

ステップ 7 [開く (Open)] をクリックして、セッションを開始します。

インスタンスに初めて接続する場合は、「サーバーのホストキーがレジストリにキャッシュされていない (the server's host key is not cached in the registry)」というメッセージが表示されることがあります。[はい (Yes)] をクリックして、接続を続行します。

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。