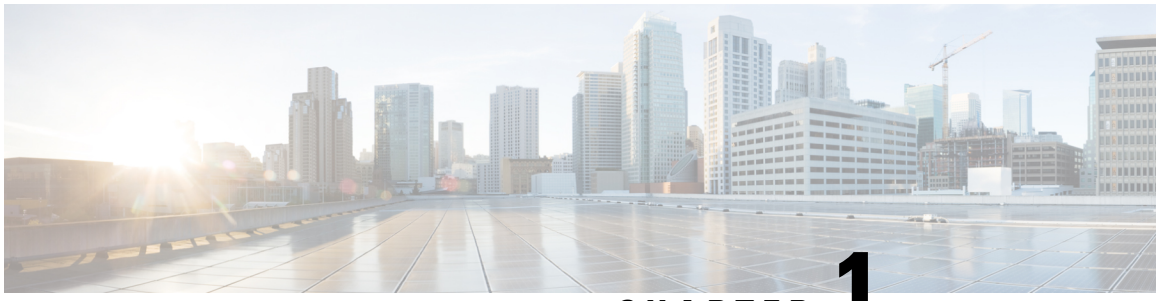


# Cisco Secure Firewall Threat Defense Virtual Getting Started Guide, Version 7.6

**First Published:** 2024-09-16

**Americas Headquarters**
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883



CHAPTER 1

# Introduction to the Cisco Secure Firewall Threat Defense Virtual

The Cisco Secure Firewall Threat Defense Virtual (Firewall Threat Defense Virtual) brings Cisco's Next Generation Firewall functionality to virtualized environments, enabling consistent security policies to follow workloads across your physical, virtual, and cloud environments, and between clouds.

Today, organizations rely on a mixture of physical and virtual control points to meet their network security needs. They need the flexibility to deploy different physical and virtual firewalls across a wide range of environments while still maintaining consistent policy across branch offices, corporate datacenters, and all points between. From data center consolidation to office relocations, mergers and acquisitions, as well as seasonal peaks in demand on your applications, Cisco’s virtual firewall portfolio helps you simplify security management with the convenience of unified policy and the flexibility to deploy everywhere.

The Cisco Secure Firewall Threat Defense Virtual combines Cisco’s proven network firewall with Snort IPS, URL filtering, and malware defense. It simplifies threat protection with consistent security policies across physical, private, and public cloud environments. Get deep visibility into your network and quickly detect threat origin and activity. Then, stop attacks before they impact your operations.

Secure Firewall Threat Defense Virtual is the popular virtualized solution. Prioritize threats with automated risk rankings and impact flags to focus your resources on events requiring immediate action. License portability provides flexibility to move from your on-premises private cloud to public cloud while maintaining consistent policy and unified management across all of your appliances. Cisco Smart Software Licensing makes it easy to deploy, manage, and track virtual firewall instances.


- How to Manage Secure Firewall Threat Defense Virtual Device, on page 1


# How to Manage Secure Firewall Threat Defense Virtual Device

You have two options to manage your Secure Firewall Threat Defense Virtual.

## Secure Firewall Management Center

If you are managing large numbers of devices, or if you want to use the more complex features and configurations that the Firewall Threat Defense allows, use the Firewall Management Center to configure your devices instead of the integrated Firewall Device Manager.

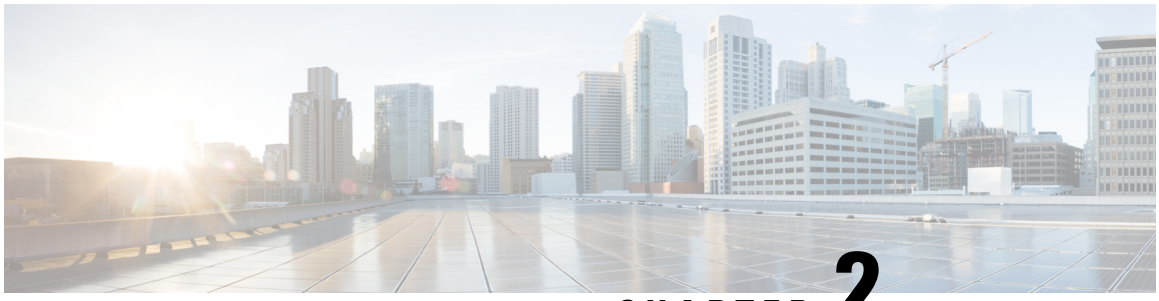
**Important** You cannot use both the Firewall Device Manager and the Firewall Management Center to manage the Firewall Threat Defense device. Once the Firewall Device Manager integrated management is enabled, it won't be possible to use the Firewall Management Center to manage the Firewall Threat Defense device, unless you disable the local management and re-configure the management to use the Firewall Management Center. On the other hand, when you register the Firewall Threat Defense device to the Firewall Management Center, the Firewall Device Manager onboard management service is disabled.

**Caution** Currently, Cisco does not have an option to migrate your Firewall Device Manager configuration to the Firewall Management Center and vice-versa. Take this into consideration when you choose what type of management you configure for the Firewall Threat Defense device.

# Secure Firewall Device Manager

The Firewall Device Manager is a web interface included on most Firewall Threat Defense devices. It lets you configure the basic features of the software that are most commonly used for small networks. It is especially designed for networks that include a single device or just a few, where you do not want to use a high-powered multiple-device manager to control a large network with many devices.

**Note** See the Cisco Secure Firewall Threat Defense Compatibility Guide for list of devices that support the Firewall Device Manager.



CHAPTER 2

# Deploy the Firewall Threat Defense Virtual on VMware

This chapter describes the procedures to deploy the Firewall Threat Defense Virtual to a VMware vSphere environment, either to a vSphere vCenter or to a stand-alone ESXi host.


- Overview, on page 3
- VMware Feature Support for the Firewall Threat Defense Virtual, on page 3
- System Requirements, on page 4
- Guidelines and Limitations, on page 8
- Plan the Interfaces, on page 14
- About VMware Deployment, on page 18
- End-to-End Procedure, on page 18
- Deploy the Firewall Threat Defense Virtual to vSphere vCenter, on page 20
- Prepare the Day 0 Configuration File for Cluster Deployment, on page 28
- Deploy the Firewall Threat Defense Virtual to a vSphere ESXi Host, on page 30
- Complete the Firewall Threat Defense Virtual Setup Using the CLI, on page 33
- Increasing Performance on ESXi Configurations, on page 34
- NUMA Guidelines, on page 35
- SR-IOV Interface Provisioning, on page 35


## Overview

Cisco packages 64-bit Firewall Threat Defense Virtual devices for VMware vSphere vCenter and ESXi hosting environments. The Firewall Threat Defense Virtual is distributed in an Open Virtualization Format (OVF) package available from Cisco.com. OVF is an open-source standard for packaging and distributing software applications for virtual machines (VM). An OVF package contains multiple files in a single directory.

You can deploy the Firewall Threat Defense Virtual to any x86 device that is capable of running VMware ESXi. In order to deploy the Firewall Threat Defense Virtual you should be familiar with VMware and vSphere, including vSphere networking, ESXi host setup and configuration, and virtual machine guest deployment.

## VMware Feature Support for the Firewall Threat Defense Virtual

The following table lists the VMware feature support for the Firewall Threat Defense Virtual.

*Table 1: VMware Feature Support for the Firewall Threat Defense Virtual*

| Feature | Description | Support (Yes/No) | Comment |
|---|---|---|---|
| Cold Clone | The VM is powered off during cloning. | No | – |
| vMotion | Used for live migration of VMs. | Yes | Use shared storage. See vMotion Support. |
| Distributed Resource Scheduler (DRS) | Monitors VM workload and resolves imbalance, identifies VMs for live migration with vMotion. | No | Not qualified |
| Hot add | The VM is running during an addition. | No | – |
| Hot clone | The VM is running during cloning. | No | – |
| Hot removal | The VM is running during removal. | No | – |
| Snapshot | The VM freezes for a few seconds. | No | Risk of out-of-sync situations between the Firewall Management Center and managed devices. |
| Suspend and resume | The VM is suspended, then resumed. | Yes | – |
| vCloud Director | Allows automatic deployment of VMs. | No | – |
| VMware FT | Used for HA on VMs. | No | Use the failover feature for Firewall Threat Defense Virtual VM failovers. |
| VMware HA with VM heartbeats | Used for VM failures. | No | Use the failover feature for Firewall Threat Defense Virtual VM failovers. |
| VMware vSphere Standalone Windows Client | Used to deploy VMs. | Yes | – |
| VMware vSphere Web Client | Used to deploy VMs. | Yes | – |

# System Requirements

See the Cisco Secure Firewall Threat Defense Compatibility Guide for the most current information about hypervisor support for the Firewall Threat Defense Virtual.

The specific hardware used for Firewall Threat Defense Virtual deployments can vary, depending on the number of instances deployed and usage requirements. Each instance of the Firewall Threat Defense Virtual requires a minimum resource allocation—number of memory, CPUs, and disk space—on the server.

Systems running VMware vCenter Server and ESXi instances must meet specific hardware and operating system requirements. For a list of supported platforms, see the VMware online Compatibility Guide.

***Table 2: Firewall Threat Defense Virtual Appliance Resource Requirements***

| Settings | Value |
|---|---|
| Performance Tiers | The Firewall Threat Defense Virtual supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.<br>• FTDv5 4vCPU/8GB (100Mbps)<br>• FTDv10 4vCPU/8GB (1Gbps)<br>• FTDv20 4vCPU/8GB (3Gbps)<br>• FTDv30 8vCPU/16GB (5Gbps)<br>• FTDv50 12vCPU/24GB (10Gbps)<br>• FTDv100 16vCPU/32GB (16Gbps)<br>See the "Licensing" chapter in the Cisco Secure Firewall Management Center Administration Guide for guidelines when licensing your Firewall Threat Defense Virtual device.<br>**Note**<br>To change the vCPU/memory values, you must first power off the Firewall Threat Defense Virtual device. |
| Storage | Based on Disk Format selection.<br>• Thin Provision disk size is 48.24GB. |

| Settings | Value |
|---|---|
| vNICs | The Firewall Threat Defense Virtual supports the following virtual network adapters:<br>• **VMXNET3**—Firewall Threat Defense Virtual on VMware now defaults to vmxnet3 interfaces when you create a virtual device. Previously, the default was e1000. The vmxnet3 driver uses two management interfaces. The first two Ethernet adapters must be configured as management interfaces; one for device management/registration, one for diagnostics.<br>• **IXGBE**—The ixgbe driver uses two management interfaces. The first two PCI devices must be configured as management interfaces; one for device management/registration, one reserved for internal use. The ixgbe driver does not support failover (HA) deployments of Firewall Threat Defense Virtual.<br>• **E1000**—<br>• **IXGBE-VF**—The ixgbe-vf (10 Gbit/s) driver supports virtual function devices that can only be activated on kernels that support SR-IOV. SR-IOV requires the correct platform and OS support; see Support for SR-IOV section for more information. |

### Support for Virtualization Technology

- Virtualization Technology (VT) is a set of enhancements to newer processors that improves performance for running virtual machines. Your system should have CPUs that support either Intel VT or AMD-V extensions for hardware virtualization. Both Intel and AMD provide online processor identification utilities to help you identify CPUs and determine their capabilities.
- Many servers that include CPUs with VT support might have VT disabled by default, so you must enable VT manually. You should consult your manufacturer's documentation for instructions on how to enable VT support on your system.



**Note** If your CPUs support VT, but you do not see this option in the BIOS, contact your vendor to request a BIOS version that lets you enable VT support.

### Disable Hyperthreading

We recommend that you disable hyperthreading for your systems that run the Firewall Threat Defense Virtual; see Hyperthreading Not Recommended, on page 10. The following processors support hyperthreading and have two threads per core:

- Processors based on the Intel Xeon 5500 processor microarchitecture.
- Intel Pentium 4 (HT-enabled)
- Intel Pentium EE 840 (HT-enabled)

To disable hyperthreading, you must first disable it in your system's BIOS settings and then turn it off in the vSphere Client (note that hyperthreading is enabled by default for vSphere). Consult your system documentation to determine whether your CPU supports hyperthreading.

### Support for SR-IOV

SR-IOV Virtual Functions require specific system resources. A server that supports SR-IOV is required in addition to an SR-IOV capable PCIe adapter. You must be aware of the following hardware considerations:

- The capabilities of SR-IOV NICs, including the number of VFs available, differ across vendors and devices. The following NICs are supported:
  - Intel Ethernet Server Adapter X520 - DA2
  - Intel Ethernet Server Adapter X540
- Not all PCIe slots support SR-IOV.
- SR-IOV-capable PCIe slots may have different capabilities.
- x86_64 multicore CPU — Intel Sandy Bridge or later (Recommended).

**Note** We tested the Firewall Threat Defense Virtual on Intel's Broadwell CPU (E5-2699-v4) at 2.3GHz.

- Cores
  - Minimum of 8 physical cores per CPU socket.

**Note** Firewall Threat Defense Virtual does not support multi-Non-uniform memory access (NUMA) nodes and multiple CPU sockets for physical cores.

  - Ensure that you assign all the allocated physical cores to a single socket.

**Note** CPU pinning is recommended to achieve full throughput.

You should consult your manufacturer's documentation for SR-IOV support on your system. You can search the VMware online Compatibility Guide for system recommendations that include SR-IOV support.

### Support for SSSE3

- Firewall Threat Defense Virtual requires support for Supplemental Streaming SIMD Extensions 3 (SSSE3 or SSE3S), a single instruction, multiple data (SIMD) instruction set created by Intel.
- Your system should have CPUs that support SSSE3, such as Intel Core 2 Duo, Intel Core i7/i5/i3, Intel Atom, AMD Bulldozer, AMD Bobcat, and later processors.

- See this reference page for more information about the SSSE3 instruction set and CPUs that support SSSE3.

### Verify CPU Support

You can use the Linux command line to get information about the CPU hardware. For example, the **/proc/cpuinfo** file contains details about individual CPU cores. Output its contents with **less** or **cat**.

You can look at the flags section for the following values:

- vmx—Intel VT extensions
- svm—AMD-V extensions
- ssse3—SSSE3 extensions

Use **grep** to see if any of these values exist in the file by running the following command:

```
egrep "vmx|svm|ssse3" /proc/cpuinfo
```

If your system supports VT or SSSE3, then you should see vmx, svm, or ssse3 in the list of flags. The following example shows output from a system with two CPUs:

```
flags    : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat
pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm syscall nx lm constant_tsc pni monitor
ds_cpl vmx est tm2 ssse3 cx16 xtpr lahf_lm

flags    : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat
pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm syscall nx lm constant_tsc pni monitor
ds_cpl vmx est tm2 ssse3 cx16 xtpr lahf_lm
```

# Guidelines and Limitations

### Performance Tiers for Firewall Threat Defense Virtual Smart Licensing

The Firewall Threat Defense Virtual supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.

***Table 3: Firewall Threat Defense Virtual Licensed Feature Limits Based on Entitlement***

| Performance Tier | Device Specifications (Core/RAM) | Rate Limit | RA VPN Session Limit |
|---|---|---|---|
| FTDv5, 100Mbps | 4 core/8 GB | 100Mbps | 50 |
| FTDv10, 1Gbps | 4 core/8 GB | 1Gbps | 250 |
| FTDv20, 3Gbps | 4 core/8 GB | 3Gbps | 250 |
| FTDv30, 5Gbps | 8 core/16 GB | 5Gbps | 250 |
| FTDv50, 10Gbps | 12 core/24 GB | 10Gbps | 750 |

| Performance Tier | Device Specifications (Core/RAM) | Rate Limit | RA VPN Session Limit |
|---|---|---|---|
| FTDv100, 16Gbps | 16 core/32 GB | 16Gbps | 10,000 |

See the "Licensing" chapter in the Cisco Secure Firewall Management Center Administration Guide for guidelines when licensing your Firewall Threat Defense Virtual device.

### Performance Optimizations

To achieve the best performance out of the Firewall Threat Defense Virtual, you can make adjustments to the both VM and the host. See Increasing Performance on ESXi Configurations, on page 34, NUMA Guidelines, on page 35, and SR-IOV Interface Provisioning, on page 35, for more information.

**Receive Side Scaling**—The Firewall Threat Defense Virtual supports Receive Side Scaling (RSS), which is a technology utilized by network adapters to distribute network receive traffic to multiple processor cores. RSS is supported on Version 7.0 and later. See Multiple RX Queues for Receive Side Scaling (RSS) for more information.

### Clustering

Starting from version 7.2, clustering is supported on threat defense virtual instances deployed on VMware. See Clustering for Threat Defense Virtual in a Private Cloud for more information.

### Management Mode

- You have two options to manage your Secure Firewall Threat Defense (formerly Firepower Threat Defense) device:
  - The Firewall Device Manager onboard integrated manager.



**Note** The Firewall Threat Defense Virtual on VMware supports Firewall Device Manager starting with Cisco software version 6.2.2 and later. Any Firewall Threat Defense Virtual on VMware running software earlier than version 6.2.2 can only be managed using the Firewall Management Center; see How to Manage Secure Firewall Threat Defense Virtual Device, on page 1

  - The Firewall Management Center.

- You must install a new image (version 6.2.2 or greater) to get Firewall Device Manager support. You cannot upgrade an existing Firewall Threat Defense Virtual machine from an older version (earlier than 6.2.2) and then switch to the Firewall Device Manager.
- Firewall Device Manager (local manager) is enabled by default.



**Note** When you choose **Yes** for **Enable Local Manager**, the Firewall Mode is changed to routed. This is the only supported mode when using the Firewall Device Manager.

### OVF File Guidelines

You have the following installation options for installing a Firewall Threat Defense Virtual appliance:

```
Cisco_Secure_Firewall_Threat_Defense_Virtual-VI-X.X.X-xxx.ovf
Cisco_Secure_Firewall_Threat_Defense_Virtual-ESXi-X.X.X-xxx.ovf
```

where X.X.X-xxx is the version and build number of the file you want to use.

- If you deploy with a VI OVF template, the installation process allows you to perform the entire initial setup for the Firewall Threat Defense Virtual appliance. You can specify:
  - A new password for the admin account.
  - Network settings that allow the appliance to communicate on your management network.
  - Management, either local management using the Firewall Device Manager (default) or remote management using the Firewall Management Center.
  - Firewall Mode—hen you choose Yes for Enable Local Manager, the Firewall Mode is changed to routed. This is the only supported mode when using the Firewall Device Manager.



**Note** You must manage this virtual appliance using VMware vCenter.

- If you deploy using an ESXi OVF template, you must configure System-required settings after installation. You manage this Firewall Threat Defense Virtual as a standalone appliance on ESXi; see Deploy the Firewall Threat Defense Virtual to a vSphere ESXi Host, on page 30 for more information.

### Unable to Save Virtual Machine (VM) Configuration in vSphere 7.0.2

If you are using vSphere 7.0.2, you may not be allowed to save the VM configuration.



**Note** You can resolve this issue by following the instructions in VMware knowledge base article: https://kb.vmware.com/s/article/83898.

### vMotion Support

We recommend that you only use shared storage if you plan to use vMotion. During deployment, if you have a host cluster, you can either provision storage locally (on a specific host) or on a shared host. However, if you try to vMotion the Secure Firewall Management Center Virtual (formerly Firepower Management Center Virtual) to another host, using local storage will produce an error.

### Hyperthreading Not Recommended

Hyperthreading technology allows a single physical processor core to behave like two logical processors. We recommend that you disable hyperthreading for your systems that run the Firewall Threat Defense Virtual. The Snort process already maximizes the processing resources in a CPU core. When you attempt to push two CPU utilization threads through each processor, you do not receive any improvement in performance. You may actually see a decrease in performance because of the overhead required for the hyperthreading process.

### INIT Respawning Error Messages Symptom

You may see the following error message on the Firewall Threat Defense Virtual console running on ESXi 6 and ESXi 6.5:

```
"INIT: Id "ftdv" respawning too fast: disabled for 5 minutes"
```

**Workaround**—Edit the virtual machine settings in vSphere to add a serial port while the device is powered off.

1. Right-click the virtual machine and select **Edit Settings**.
2. On the Virtual Hardware tab, select **Serial port** from the **New device** drop-down menu, and click **Add**.

   The serial port appears at the bottom of the virtual device list.
3. On the **Virtual Hardware** tab, expand **Serial port**, and select connection type **Use physical serial port**.
4. Uncheck the **Connect at power on** checkbox.

   Click **OK** to save settings.

### Exclude Virtual Machines from Firewall Protection

In a vSphere enviroment where the vCenter Server is integrated with VMware NSX Manager, a Distributed Firewall (DFW) runs in the kernel as a VIB package on all the ESXi host clusters that are prepared for NSX. Host preparation automatically activates DFW on the ESXi host clusters.

The Firewall Threat Defense Virtual uses promiscuous mode to operate, and the performance of virtual machines that require promiscuous mode may be adversely affected if these virtual machines are protected by a distributed firewall. VMware recommends that you exclude virtual machines that require promiscuous mode from distributed firewall protection.

1. Navigate to Exclusion List settings.
   - In NSX 6.4.1 and later, navigate to **Networking & Security** > **Security** > **Firewall Settings** > **Exclusion List**.
   - In NSX 6.4.0, navigate to **Networking & Security** > **Security** > **Firewall** > **Exclusion List**.
2. Click **Add**.
3. Move the VMs that you want to exclude to **Selected Objects**.
4. Click **OK**.

If a virtual machine has multiple vNICs, all of them are excluded from protection. If you add vNICs to a virtual machine after it has been added to the Exclusion List, Firewall is automatically deployed on the newly added vNICs. To exclude the new vNICs from firewall protection, you must remove the virtual machine from the Exclusion List and then add it back to the Exclusion List. An alternative workaround is to power cycle (power off and then power on) the virtual machine, but the first option is less disruptive.

### Modify the Security Policy Settings for a vSphere Standard Switch

For a vSphere standard switch, the three elements of the Layer 2 Security policy are promiscuous mode, MAC address changes, and forged transmits. Firewall Threat Defense Virtual uses promiscuous mode to operate, and Firewall Threat Defense Virtualrtual high availability depends on switching the MAC address between the active and the standby to operate correctly.

The default settings will block correct operation of the Firewall Threat Defense Virtual. See the following required settings:

***Table 4: vSphere Standard Switch Security Policy Options***

| Option | Required Setting | Action |
|---|---|---|
| Promiscuous Mode | Accept | You must edit the security policy for a vSphere standard switch in the vSphere Web Client and set the Promiscuous mode option to Accept.<br>Firewalls, port scanners, intrusion detection systems and so on, need to run in promiscuous mode. |
| MAC Address Changes | Accept | You should verify the security policy for a vSphere standard switch in the vSphere Web Client and confirm the MAC address changes option is set to Accept. |
| Forged Transmits | Accept | You should verify the security policy for a vSphere standard switch in the vSphere Web Client and confirm the Forged transmits option is set to Accept. |



**Note** We do not have any recommendations for NSX-T configuration of security policy settings for a vSphere standard switch as the VMware with NSX-T is not qualified.

### Snort

- If you are observing abnormal behavior such as Snort taking a long time to shut down, or the VM being slow in general or when a certain process is executed, collect logs from the Firewall Threat Defense Virtual and the VM host. Collection of overall CPU usage, memory, I/O usage, and read/write speed logs will help troubleshoot the issues.
- High CPU and I/O usage is observed when Snort is shutting down. If a number of Firewall Threat Defense Virtual instances have been created on a single host with insufficient memory and no dedicated CPU, Snort will take a long time to shut down which will result in the creation of Snort cores.

# Modify the Security Policy Settings for a vSphere Standard Switch

The default settings will block correct operation of the Firewall Threat Defense Virtual.

### Procedure

**Step 1** In the vSphere Web Client, navigate to the host.

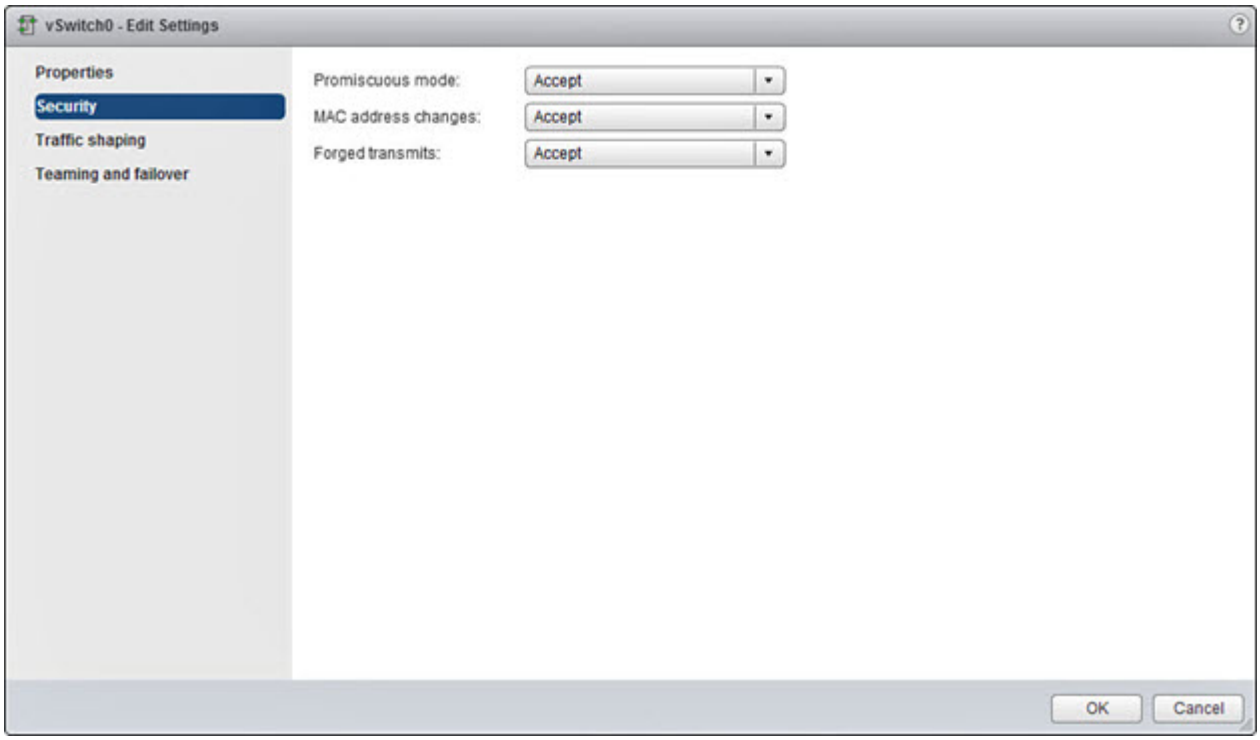
**Step 2** On the **Manage** tab, click **Networking**, and select **Virtual switches**.

**Step 3** Select a standard switch from the list and click **Edit settings**.

**Step 4** Select **Security** and view the current settings.

**Step 5** **Accept** promiscuous mode activation, MAC address changes, and forged transmits in the guest operating system of the virtual machines attached to the standard switch.

***Figure 1: vSwitch Edit Settings***




**Step 6** Click **OK**.

### What to do next

- Ensure these settings are the same on all networks that are configured for management and failover (HA) interfaces on the Firewall Threat Defense Virtual devices.

# Plan the Interfaces

You can avoid reboots and configuration issues by planning the Firewall Threat Defense Virtual vNIC and interface mapping in advance of deployment. The Firewall Threat Defense Virtual deploys with 10 interfaces, and must be powered up at firstboot with at least 4 interfaces.

The Firewall Threat Defense Virtual supports the vmxnet3 (default), ixgbe, and e1000 virtual network adapters. In addition, with a properly configured system, Firewall Threat Defense Virtual also supports the ixgbe-vf driver for SR-IOV; see System Requirements, on page 4 for more information.



**Important** Firewall Threat Defense Virtual on VMware now defaults to vmxnet3 interfaces when you create a virtual device. Previously, the default was e1000. If you are using e1000 interfaces, we **strongly recommend** you switch. The vmxnet3 device drivers and network processing are integrated with the ESXi hypervisor, so they use fewer resources and offer better network performance.

# Interface Guidelines and Limitations

The following sections provide guidelines and limitations for the supported virtual network adapters used with Firewall Threat Defense Virtual on VMware. It's important to keep these guidelines in mind when planning your deployment.

### General Guidelines

- As previously stated, the Firewall Threat Defense Virtual deploys with 10 interfaces, and must be powered up at firstboot with at least 4 interfaces. You need to assign a network to **AT LEAST FOUR INTERFACES**.
- We recommend that you avoid using the HOLDING port group for the Firewall Threat Defense Virtual interface. The HOLDING port group from vSphere causes inconsistent interface connectivity. A holding port is a generic port group which is assigned to a VLAN ID. This may lead to issues during HA formation with the secondary Firewall Threat Defense Virtual device.
- You do not need to use all 10 Firewall Threat Defense Virtual interfaces; for interfaces you do not intend to use, you can simply leave the interface disabled within the Firewall Threat Defense Virtual configuration.
- Keep in mind that you cannot add more virtual interfaces to the virtual machine after deployment. If you delete some interfaces and then decide you want more, you'll have to delete the virtual machine and start over.
- You can optionally configure a data interface for the Firewall Management Center instead of the Management interface. The Management interface is a prerequisite for data interface management, so you still need to configure it in your initial setup. Note that the Firewall Management Center access from a data interface is not supported in High Availability deployments. For more information about configuring a data interface for the Firewall Management Center access, see the **configure network management-data-interface** command in Cisco Secure Firewall Threat Defense Command Reference.
- The order of failover having two virtual NICs for the ESX port group, which is used in Firewall Threat Defense Virtual inside interface or the failover high availability link, must be configured in a manner

where one virtual NIC acts as an active uplink and the other as the standby uplink. This is necessary for the two VMs to ping each other or for the threat defense virtual high availability (HA) link to be up.

### Default VMXNET3 Interfaces



**Important** Firewall Threat Defense Virtual on VMware now defaults to vmxnet3 interfaces when you create a virtual device. Previously, the default was e1000. If you are using e1000 interfaces, we **strongly recommend** you switch. The vmxnet3 device drivers and network processing are integrated with the ESXi hypervisor, so they use fewer resources and offer better network performance.

- The vmxnet3 driver uses two management interfaces. The first two Ethernet adapters must be configured as management interfaces; one for device management/registration, one that is reserved for internal use.
- For vmxnet3, Cisco recommends using a host managed by VMware vCenter when using more than four vmxnet3 network interfaces. When deployed on standalone ESXi, additional network interfaces are not added to the virtual machine with sequential PCI bus addresses. When the host is managed with a VMware vCenter, the correct order can be obtained from the XML in the configuration CDROM. When the host is running standalone ESXi, the only way to determine the order of the network interfaces is to manually compare the MAC addresses seen on the Firewall Threat Defense Virtual to the MAC addresses seen from the VMware configuration tool.

The following table describes the concordance of Network Adapter, Source Networks and Destination Networks for Firewall Threat Defense Virtual for vmxnet3 and ixgbe interfaces.

***Table 5: Source to Destination Network Mapping—VMXNET3 and IXGBE***

| Network Adapter | Source Networks | Destination Networks | Function |
|---|---|---|---|
| Network adapter 1 | Management0-0 | Management0/0 | Management |
| Network adapter 2 | Reserved for internal use. | Reserved for internal use. | Reserved for internal use. |
| Network adapter 3 | GigabitEthernet0-0 | GigabitEthernet0/0 | Outside data |
| Network adapter 4 | GigabitEthernet0-1 | GigabitEthernet0/1 | Inside data |
| Network adapter 5 | GigabitEthernet0-2 | GigabitEthernet0/2 | Data traffic (Optional) |
| Network adapter 6 | GigabitEthernet0-3 | GigabitEthernet0/3 | Data traffic (Optional) |
| Network adapter 7 | GigabitEthernet0-4 | GigabitEthernet0/4 | Data traffic (Optional) |
| Network adapter 8 | GigabitEthernet0-5 | GigabitEthernet0/5 | Data traffic (Optional) |
| Network adapter 9 | GigabitEthernet0-6 | GigabitEthernet0/6 | Data traffic (Optional) |
| Network adapter 10 | GigabitEthernet0-7 | GigabitEthernet0/7 | Data traffic (Optional) |

### IXGBE Interfaces

- The ixgbe driver uses two management interfaces. The first two PCI devices must be configured as management interfaces; one for device management/registration, one reserved for internal use.
- For ixgbe, the ESXi platform requires the ixgbe NIC to support the ixgbe PCI device. In addition, the ESXi platform has specific BIOS and configuration requirements that are needed to support ixgbe PCI devices. Refer to the Intel Technical Brief for more information.
- The only ixgbe traffic interface types supported are routed and ERSPAN passive. This is due to VMware limitations with respect to MAC address filtering.
- The ixgbe driver does not support failover (HA) deployments of Firewall Threat Defense Virtual.

### E1000 Interfaces



**Important** Firewall Threat Defense Virtual on VMware now defaults to vmxnet3 interfaces when you create a virtual device. Previously, the default was e1000. If you are using e1000 interfaces, we **strongly recommend** you switch. The vmxnet3 device drivers and network processing are integrated with the ESXi hypervisor, so they use fewer resources and offer better network performance.

- If you are upgrading your Firewall Threat Defense Virtual to 6.4 and are using e1000 interfaces, you should replace the e1000 interfaces with either vmxnet3 or ixgbe interfaces for greater network throughput.

The following table describes the concordance of Network Adapter, Source Networks and Destination Networks for Firewall Threat Defense Virtual for the default e1000 interfaces.

***Table 6: Source to Destination Network Mapping—E1000 Interfaces***

| Network Adapter | Source Networks | Destination Networks | Function |
|---|---|---|---|
| Network adapter 1 | Management0-0 | Management0/0 | Management |
| Network adapter 2 | GigabitEthernet0-0 | GigabitEthernet0/0 | Outside data |
| Network adapter 3 | GigabitEthernet0-1 | GigabitEthernet0/1 | Inside data |
| Network adapter 4 | GigabitEthernet0-2 | GigabitEthernet0/2 | Data traffic (Required) |
| Network adapter 5 | GigabitEthernet0-3 | GigabitEthernet0/3 | Data traffic (Optional) |
| Network adapter 6 | GigabitEthernet0-4 | GigabitEthernet0/4 | Data traffic (Optional) |
| Network adapter 7 | GigabitEthernet0-5 | GigabitEthernet0/5 | Data traffic (Optional) |
| Network adapter 8 | GigabitEthernet0-6 | GigabitEthernet0/6 | Data traffic (Optional) |
| Network adapter 9 | GigabitEthernet0-7 | GigabitEthernet0/7 | Data traffic (Optional) |
| Network adapter 10 | GigabitEthernet0-8 | GigabitEthernet0/8 | Data traffic (Optional) |

# Configure VMXNET3 Interfaces



**Important** Starting with the 6.4 release, the Firewall Threat Defense Virtual and the Firewall Management Center Virtual on VMware default to vmxnet3 interfaces when you create a virtual device. Previously, the default was e1000. If you are using e1000 interfaces, we **strongly recommend** you switch. The vmxnet3 device drivers and network processing are integrated with the ESXi hypervisor, so they use fewer resources and offer better network performance.

To change e1000 interfaces to vmxnet3, you must delete ALL interfaces and reinstall them with the vmxnet3 driver.

Although you can mix interfaces in your deployment (such as, e1000 interfaces on the Firewall Management Center and vmxnet3 interfaces on its managed virtual device), you cannot mix interfaces on the same virtual appliance. All sensing and management interfaces on the virtual appliance must be of the same type.

### Procedure

**Step 1** Power off the Firewall Threat Defense Virtual or the Firewall Management Center Virtual Machine.

To change the interfaces, you must power down the appliance.

**Step 2** Right-click the Firewall Threat Defense Virtual or the Firewall Management Center Virtual Machine in the inventory and select **Edit Settings**.

**Step 3** Select the applicable network adapters and then select **Remove**.

**Step 4** Click **Add** to open the **Add Hardware Wizard**.

**Step 5** Select **Ethernet adapter** and click **Next**.

**Step 6** Select the vmxnet3 adapter and then choose network label.

**Step 7** Repeat for all interfaces on the Firewall Threat Defense Virtual.

### What to do next

- Power on the Firewall Threat Defense Virtual or the Firewall Management Center Virtual from the VMware console.

# Adding Interfaces

You can have a total of 10 interfaces (1 management, 1 reserved for internal use, 8 data interfaces) when you deploy a Firewall Threat Defense Virtual device. For data interfaces, make sure that the **Source Networks** map to the correct **Destination Networks**, and that each data interface maps to a unique subnet or VLAN.

**Caution** You cannot add more virtual interfaces to the virtual machine and then have the Firewall Threat Defense Virtual automatically recognize them. Adding interfaces to a virtual machine requires that you completely wipe out the Firewall Threat Defense Virtual configuration. The only part of the configuration that remains intact is the management address and gateway settings.

If you need more physical-interface equivalents for a Firewall Threat Defense Virtual device, you basically have to start over. You can either deploy a new virtual machine, or you can use the "Scan for Interface Changes, and Migrate an Interface" procedure in the Cisco Secure Firewall Device Manager Configuration Guide.

# About VMware Deployment

You can deploy the Firewall Threat Defense Virtual to a standalone ESXi server or, if you have vSphere vCenter, you can deploy using the vSphere Client or the vSphere Web Client. To successfully deploy the Firewall Threat Defense Virtual you should be familiar with VMware and vSphere including vSphere networking, ESXi host setup and configuration, and virtual machine guest deployment.

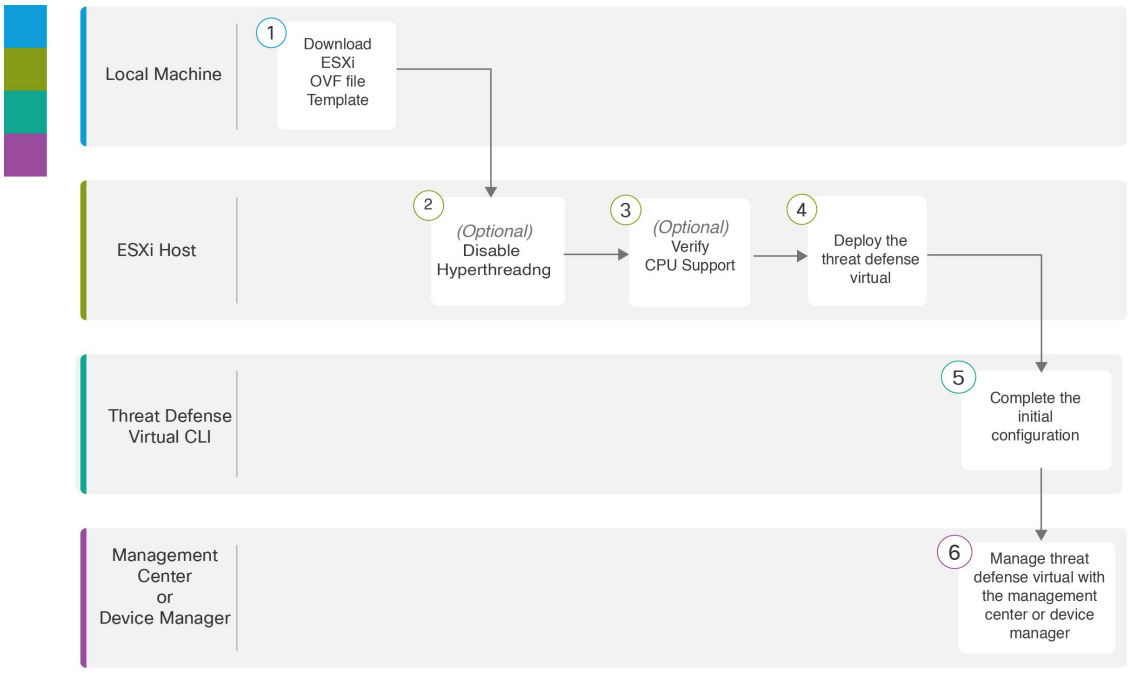
Firewall Threat Defense Virtual for VMware is distributed using the Open Virtualization Format (OVF), which is a standard method of packaging and deploying virtual machines. VMware provides several methods to provision vSphere virtual machines. The optimal method for your environment depends on factors such as the size and type of your infrastructure and the goals that you want to achieve.

The VMware vSphere Web Client and the vSphere Client are interfaces to vCenter Server, ESXi hosts, and virtual machines. With the vSphere Web Client and the vSphere Client, you can connect remotely to vCenter Server. With the vSphere Client you can also connect directly to ESXi from any Windows system. The vSphere Web Client and the vSphere Client are the primary interfaces for managing all aspects of the vSphere environment. They also provide console access to virtual machines.

All administrative functions are available through the vSphere Web Client. A subset of those functions is available through the vSphere Client.

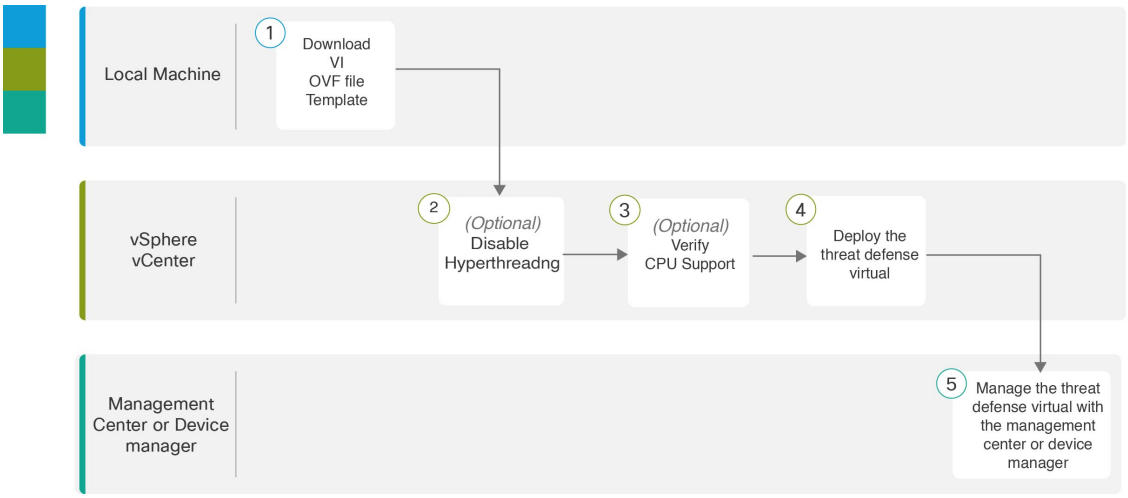
# End-to-End Procedure

The following flowchart illustrates the workflow for deploying the Firewall Threat Defense Virtual on ESXi host.




| | Workspace | Steps |
|---|---|---|
| 1 | Local Machine | Download ESXi OVF Template: Download Open Virtualization Format (OVF) package available from Cisco.com. |
| 2 | ESXi Host | (Optional) Disable Hyperthreading: Disable hyperthreading for your systems that run the Firewall Threat Defense Virtual. |
| 3 | ESXi Host | (Optional) Verify CPU Support: Use the Linux command line to get information about the CPU hardware. |
| 4 | ESXi Host | Deploy the Threat Defense Virtual: Deploy the Firewall Threat Defense Virtual appliance on a single ESXi host. |
| 5 | Firewall Threat Defense Virtual CLI | Complete the Initial Configuration: If you deployed with an ESXi OVF template, you must set up the Firewall Threat Defense Virtual using the CLI. |
| 6 | Firewall Management Center or Firewall Device Manager | Manage the Firewall Threat Defense Virtual:<br>• Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409<br>• Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Device Manager, on page 427 |

The following flowchart illustrates the workflow for deploying the Firewall Threat Defense Virtual on vSphere vCenter.




| | Workspace | Steps |
|---|---|---|
| 1 | Local Machine | Download VI OVF template: Download Open Virtualization Format (OVF) package available from Cisco.com. |
| 2 | vSphere vCenter | (Optional) Disable Hyperthreading: Disable hyperthreading for your systems that run the Firewall Threat Defense Virtual. |
| 3 | vSphere vCenter | (Optional) Verify CPU Support: Use the Linux command line to get information about the CPU hardware. |
| 4 | vSphere vCenter | Deploy the Threat Defense Virtual: Deploy the Firewall Threat Defense Virtual appliance on a single ESXi host. |
| 5 | Firewall Management Center or Firewall Device Manager | Manage the Firewall Threat Defense Virtual:<br>• Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409<br>• Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Device Manager, on page 427 |

# Deploy the Firewall Threat Defense Virtual to vSphere vCenter

Use this procedure to deploy the Firewall Threat Defense Virtual appliance to VMware vSphere vCenter. You can use the VMware Web Client (or vSphere Client) to deploy and configure the Firewall Threat Defense Virtual machines.

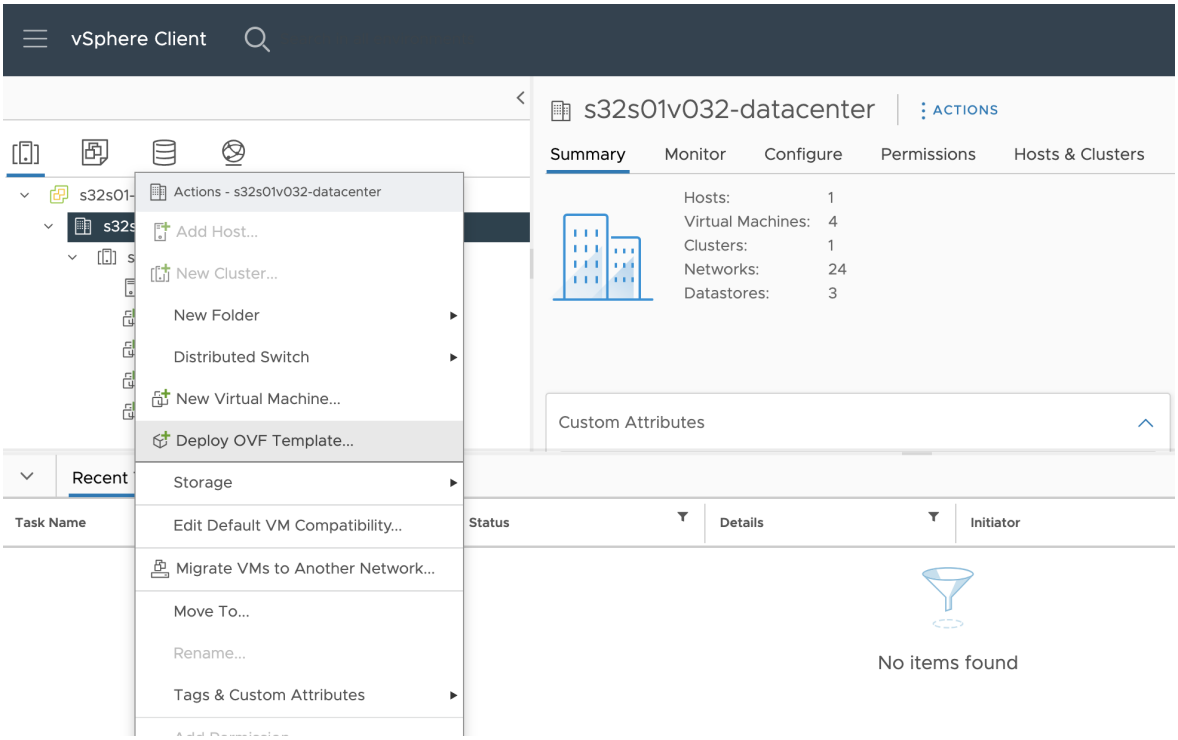
### Before you begin

- You must have at least one network configured in vSphere (for management) before you deploy the Firewall Threat Defense Virtual.

## Procedure

**Step 1** Log in to the vSphere Web Client (or the vSphere Client).

**Step 2** Using the vSphere Web Client (or the vSphere Client), deploy the OVF template file you downloaded earlier by clicking **File** > **Deploy OVF Template**.




The Deploy OVF Template wizard appears.

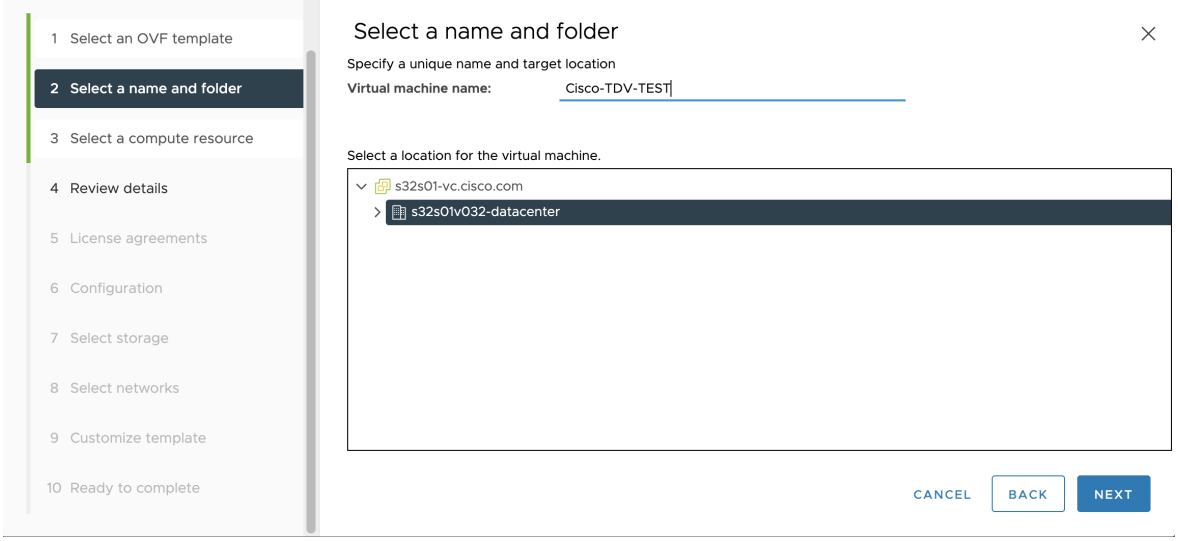
**Step 3** Browse your file system for the OVF template source location and click **Next**.

Select the Firewall Threat Defense Virtual VI OVF template:

*Cisco_Secure_Firewall_Threat_Defense_Virtual-VI-X.X.X-xxx.ovf*

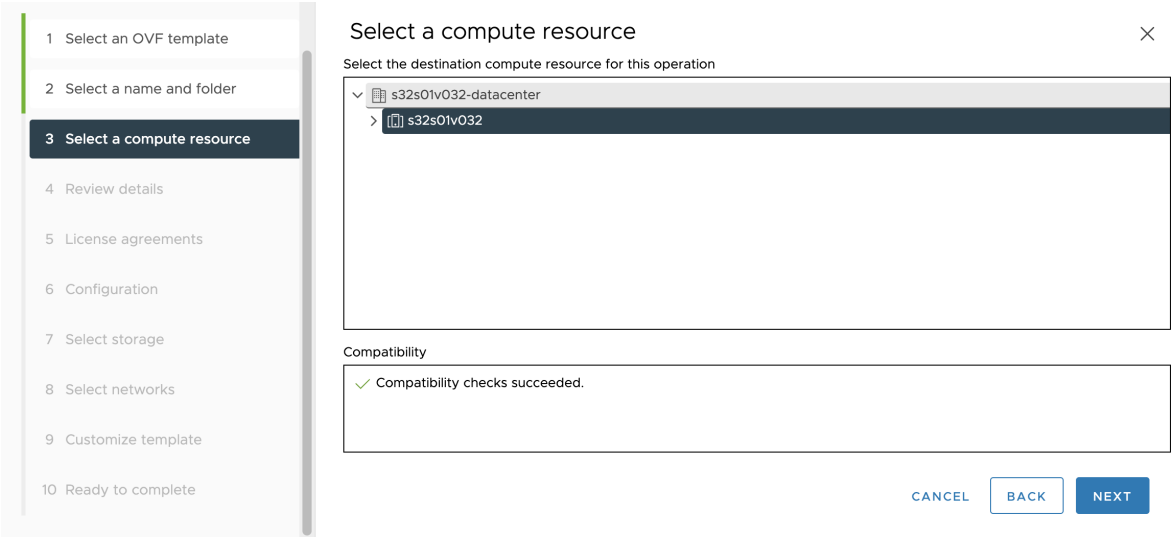
where *X.X.X-xxx* is the version and build number of the archive file you downloaded.

**Step 4** On the **Name and Location** page, enter a name for this deployment and select the location in the inventory (host or cluster) on which you want to deploy the Firewall Threat Defense Virtual, then click **Next**. The name must be unique within the inventory folder and can contain up to 80 characters.



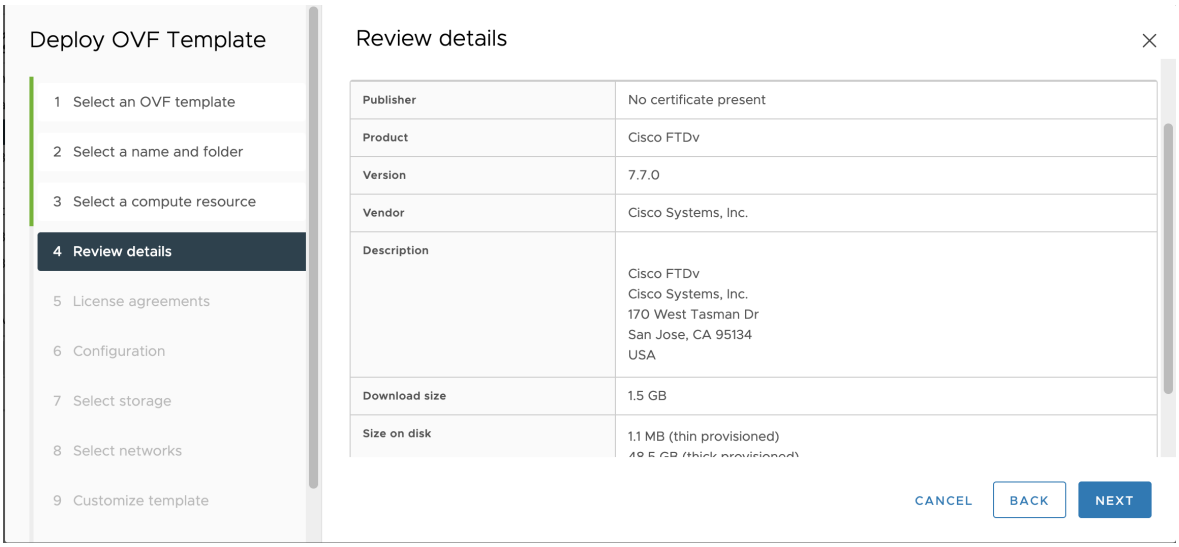

The vSphere Web Client presents the organizational hierarchy of managed objects in inventory views. Inventories are the hierarchical structure used by vCenter Server or the host to organize managed objects. This hierarchy includes all of the monitored objects in vCenter Server.

**Step 5** Navigate to, and select the resource pool where you want to run the Firewall Threat Defense Virtual and click **Next**.



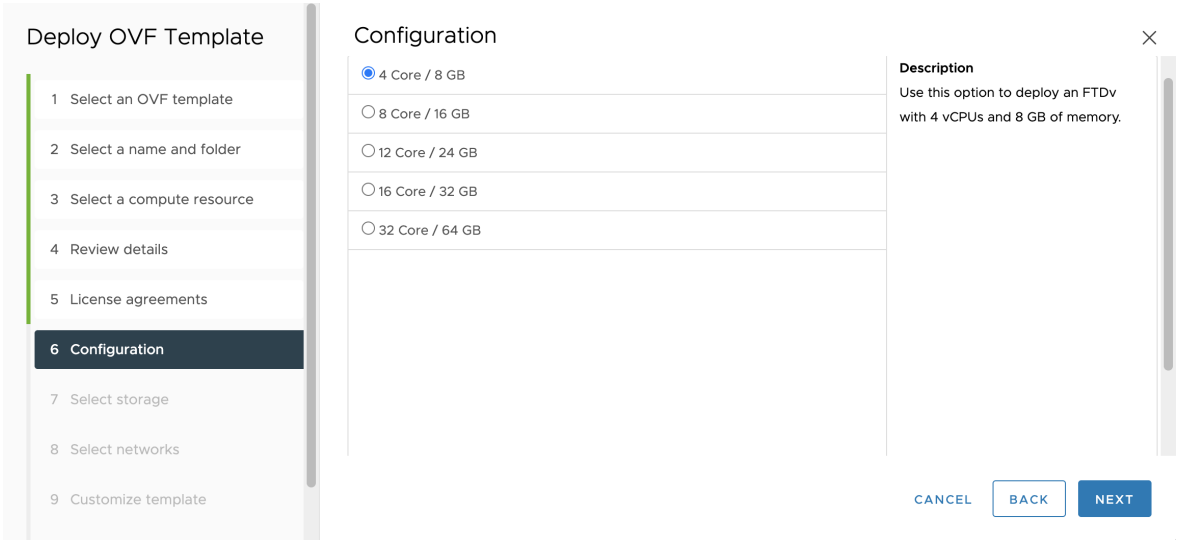

**Note**
This page appears only if the cluster contains a resource pool.

**Step 6** Review the **OVF Template Details** page and verify the OVF template information (product name, version, vendor, download size, size on disk, and description) and click **Next**.



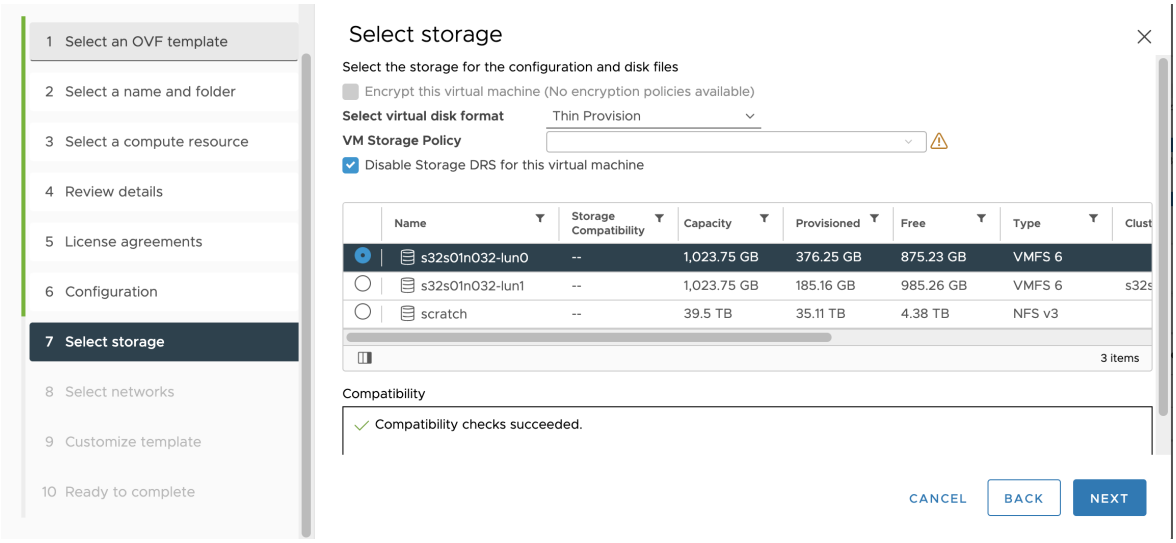

**Step 7** The **End User License Agreement** page appears. Review the license agreement packaged with the OVF template (VI templates only), click **Accept** to agree to the terms of the licenses and click **Next**.

**Step 8** On the **Configuration** page, choose one of the supported vCPU/memory values from the **Configuration** options, and click **Next**.




**Important**
The Firewall Threat Defense Virtual deploys with adjustable vCPU and memory resources.

**Step 9** Select a **Storage** location to store the virtual machine files.

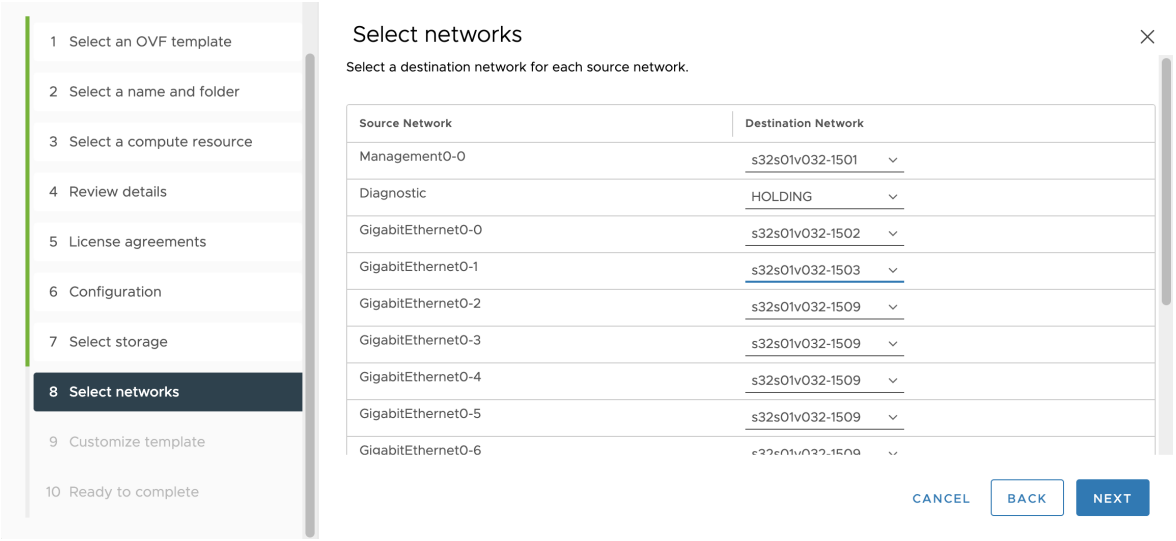



On this page, you select from datastores already configured on the destination cluster or host. The virtual machine configuration file and virtual disk files are stored on the datastore. Select a datastore large enough to accommodate the virtual machine and all of its virtual disk files.

**Step 10** From the **Select virtual disk format** drop-down list, choose the **disk format** to store the virtual machine virtual disks, and click **Next**.

When you select **Thick Provisioned**, all storage is immediately allocated. When you select **Thin Provisioned**, storage is allocated on demand as data is written to the virtual disks. Thin provisioning can also reduce the amount of time it takes to deploy the virtual appliance.

**Step 11** On the **Select networks** page, map the networks specified in the OVF template to networks in your inventory, and then click **Next**.




Ensure the Management0-0 interface is associated with a VM Network that is reachable from the Internet. Non-management interfaces are configurable from either the Firewall Management Center or from the Firewall Device Manager depending on your management mode.

**Important**

Firewall Threat Defense Virtual on VMware now defaults to vmxnet3 interfaces when you create a virtual device. Previously, the default was e1000. If you are using e1000 interfaces, we **strongly recommend** you switch. The vmxnet3 device drivers and network processing are integrated with the ESXi hypervisor, so they use fewer resources and offer better network performance.

The networks may not be in alphabetical order. If it is too difficult to find your networks, you can change the networks later from the **Edit Settings** dialog box. After you deploy, right-click the Firewall Threat Defense Virtual instance, and choose **Edit Settings**. However, that screen does not show the Firewall Threat Defense Virtual IDs (only Network Adapter IDs).

See the following concordance of Network Adapter, Source Networks and Destination Networks for the Firewall Threat Defense Virtual interfaces (note these are the default vmxnet3 interfaces):

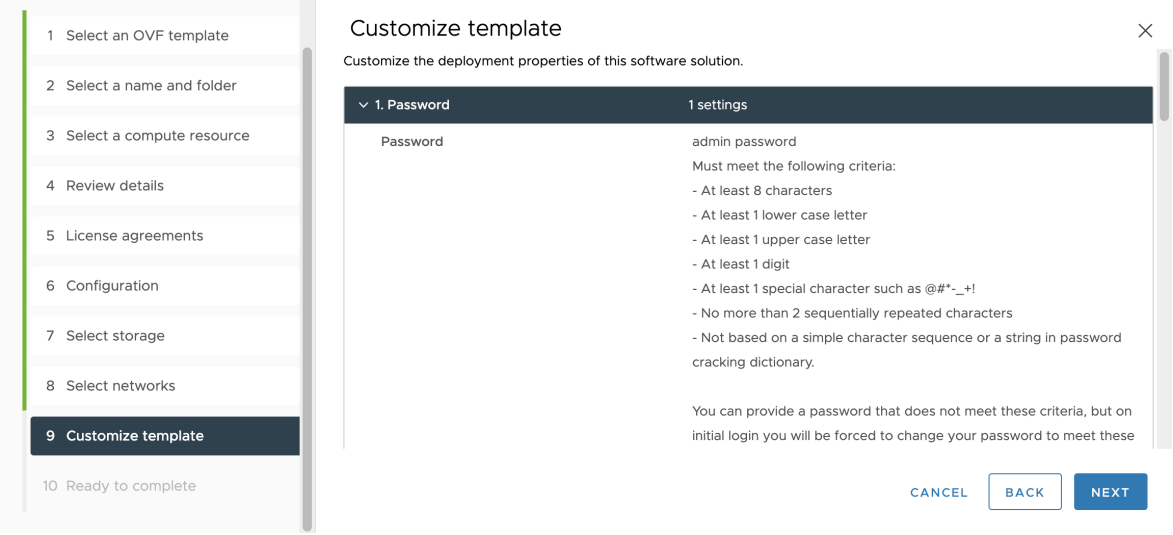
***Table 7: Source to Destination Network Mapping—VMXNET3***

| Network Adapter | Source Networks | Destination Networks | Function |
|---|---|---|---|
| Network adapter 1 | Management0-0 | Management0/0 | Management |
| Network adapter 2 | Reserved for internal use. | Reserved for internal use. | Reserved for internal use. |
| Network adapter 3 | GigabitEthernet0-0 | GigabitEthernet0/0 | Outside data |
| Network adapter 4 | GigabitEthernet0-1 | GigabitEthernet0/1 | Inside data |
| Network adapter 5 | GigabitEthernet0-2 | GigabitEthernet0/2 | Data traffic (Optional) |
| Network adapter 6 | GigabitEthernet0-3 | GigabitEthernet0/3 | Data traffic (Optional) |
| Network adapter 7 | GigabitEthernet0-4 | GigabitEthernet0/4 | Data traffic (Optional) |
| Network adapter 8 | GigabitEthernet0-5 | GigabitEthernet0/5 | Data traffic (Optional) |
| Network adapter 9 | GigabitEthernet0-6 | GigabitEthernet0/6 | Data traffic (Optional) |
| Network adapter 10 | GigabitEthernet0-7 | GigabitEthernet0/7 | Data traffic (Optional) |

You can have a total of 10 interfaces when you deploy the Firewall Threat Defense Virtual. For data interfaces, make sure that the Source Networks map to the correct Destination Networks, and that each data interface maps to a unique subnet or VLAN. You do not need to use all Firewall Threat Defense Virtual interfaces; for interfaces you do not intend to use, you can simply leave the interface disabled within the Firewall Threat Defense Virtual configuration

**Step 12** On the **Customize templates** properties page, set the user-configurable properties packaged with the OVF template (VI templates only):

a) **Password**

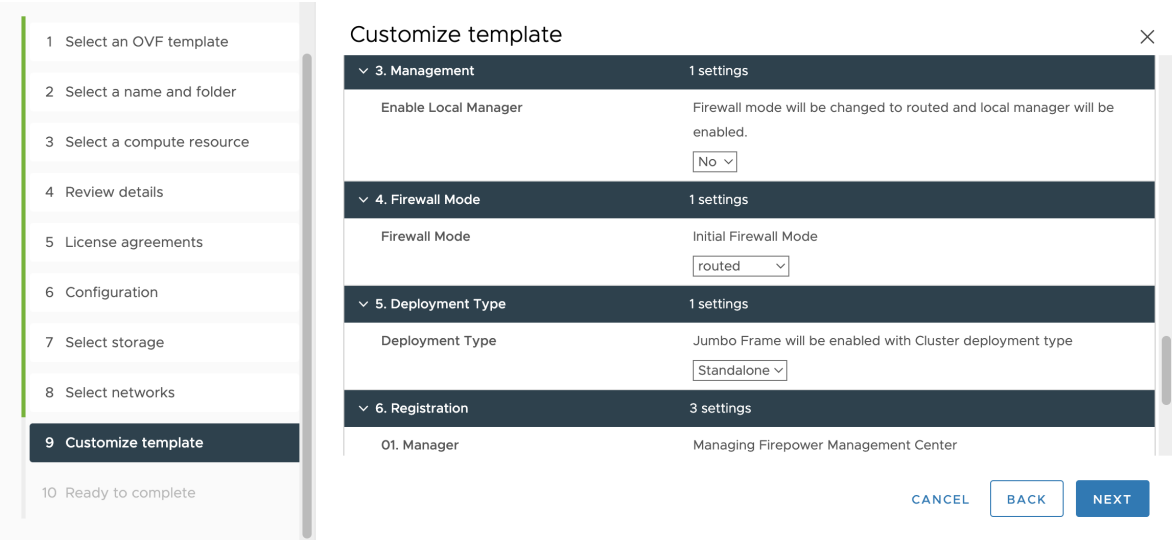



Set the password for Firewall Threat Defense Virtual admin access.

b) **Network**

Set the network information, including the Fully Qualified Domain Name (FQDN), DNS, search domain, and network protocol (IPv4 or IPv6).

c) **Management**




Set the management mode. Click the drop-down arrow for **Enable Local Manager** and select **Yes** to use the integrated Firewall Device Manager web-based configuration tool. Select **No** to use a Firewall Management Center to manage this device. See How to Manage Secure Firewall Threat Defense Virtual Device, on page 1 for an overview of how to choose your management option.

d) **Firewall Mode**

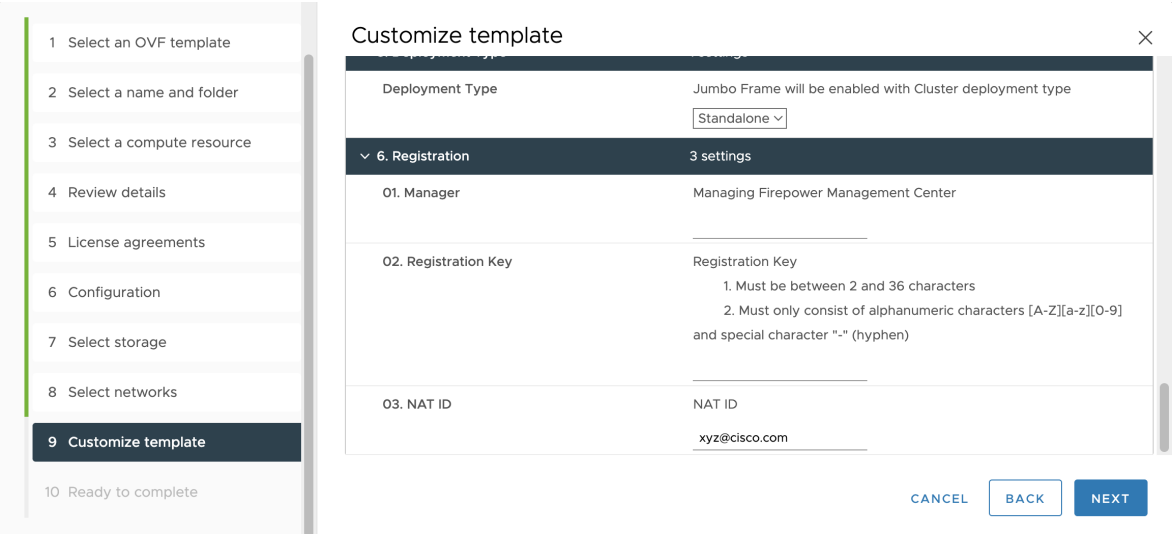
Set the initial firewall mode. Click the drop-down arrow for **Firewall Mode** and choose one of the two supported modes, either **Routed** or **Transparent**.

If you chose **Yes** for **Enable Local Manager**, you can only select **Routed** firewall mode. You cannot configure transparent firewall mode interfaces using the local Firewall Device Manager.

e) **Deployment Type**

Set the deployment type to **Standalone** or **Cluster**. Choose **Cluster** to enable jumbo-frame reservation, which is required for the cluster control link. Choose **Standalone** for a standalone or High Availability deployment. Note that if you deploy as a Standalone device, you can still use it in a cluster; however, enabling jumbo frames for clustering after deployment means you will have to restart.

f) **Registration**

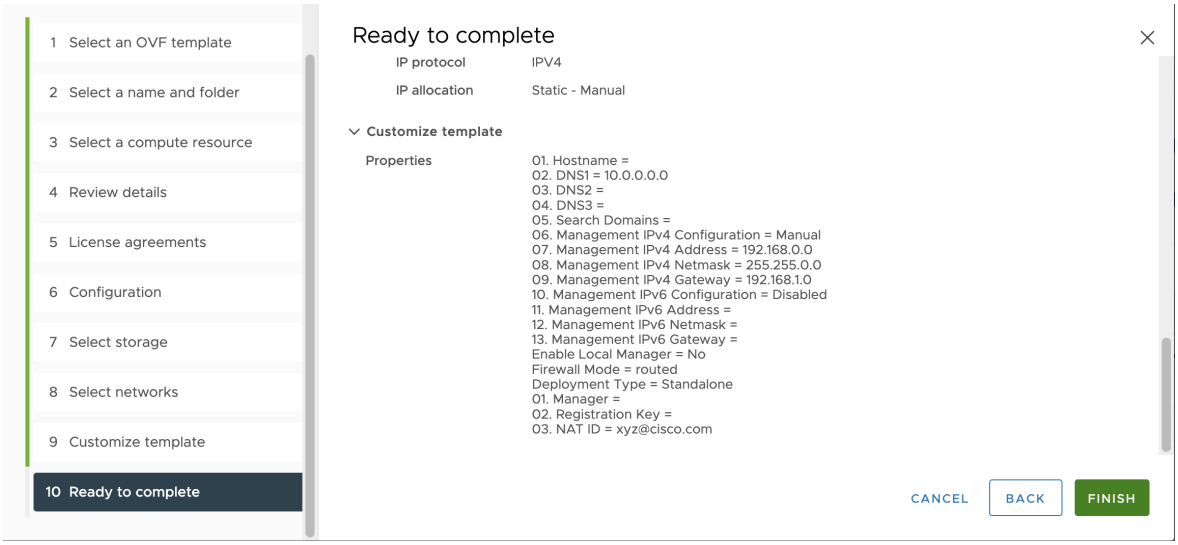



If you chose **No** for **Enable Local Manager**, you need to provide the required credentials to register this device to the managing Secure Firewall Management Center. Provide the following:

- **Managing Defense Center**—Enter the host name or IP address of the Firewall Management Center.
- **Registration Key**—The registration key is a user-generated one-time use key that must not exceed 37 characters. Valid characters include alphanumerical characters (A–Z, a–z, 0–9) and the hyphen (-). You will need to remember this registration key when you add the device to the Firewall Management Center.
- **NAT ID**—If the Firewall Threat Defense Virtual and the Firewall Management Center are separated by a Network Address Translation (NAT) device, and the Firewall Management Center is behind a NAT device, enter a unique NAT ID. This is a user-generated one-time use key that must not exceed 37 characters. Valid characters include alphanumerical characters (A–Z, a–z, 0–9) and the hyphen (-).

g) Click **Next**.

**Step 13** On the **Ready to complete** page, review and verify the displayed information. To begin the deployment with these settings, click **Finish**. To make any changes, click **Back** to navigate back through the screens.




Optionally, check the **Power on after deployment** option to power on the Firewall Threat Defense Virtual, then click **Finish**.

After you complete the wizard, the vSphere Web Client processes the virtual machine; you can see the "Initialize OVF deployment" status in the **Global Information** area **Recent Tasks** pane.

When it is finished, you see the Deploy OVF Template completion status.

The Firewall Threat Defense Virtual instance appears under the specified data center in the Inventory. Booting up the new VM could take up to 30 minutes.

**Note**
To successfully register the Firewall Threat Defense Virtual with the Cisco Licensing Authority, the Firewall Threat Defense Virtual requires Internet access. You might need to perform additional configuration after deployment to achieve Internet access and successful license registration.

### What to do next

Your next steps depend on what management mode you chose.

- If you chose **No** for **Enable Local Manager**, you'll use the Firewall Management Center to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409.

See How to Manage Secure Firewall Threat Defense Virtual Device, on page 1 for an overview of how to choose your management option.

# Prepare the Day 0 Configuration File for Cluster Deployment

You can prepare a Day 0 configuration file before you launch the Firewall Threat Defense Virtual. This file is a text file that contains the initial configuration data that gets applied at the time a virtual machine is deployed.

This initial configuration is placed into a text file named “day0-config” in a working directory you choose, and is manipulated into a day0.iso file that is mounted and read on first boot.

**Important** The day0.iso file must be available during first boot.

If you deploy with a Day 0 configuration file, the process allows you to perform the entire initial setup for the Firewall Threat Defense Virtual appliance. You can specify:

- The End User License Agreement (EULA) acceptance.
- A host name for the system.
- A new administrator password for the admin account.
- The management mode; see How to Manage Secure Firewall Threat Defense Virtual Device, on page 1.

  Enter information for the Firewall Management Center fields (**FmcIp**, **FmcRegKey**, and **FmcNatId**). Leave fields empty for the management mode you are not using.
- Network settings that allow the appliance to communicate on your management network.
- The deployment type where you can specify whether you are deploying Firewall Threat Defense Virtual as a cluster or standalone deployment.

**Note** Linux machine is used in this example, but there are similar utilities for Windows.

## Procedure

**Step 1** Log in to the Linux host where you want to deploy Firewall Threat Defense Virtual.

**Step 2** Create a text file called “day0-config” for the Firewall Threat Defense Virtual. In this text file, you must add Cluster deployment settings, network settings and information about managing the Firewall Management Center.

**Example:**

```
#Firepower Threat Defense
{

     "DeploymentType": "Cluster"

}
```

Enter the Firewall Management Center fields (**FmcIp**, **FmcRegKey**, and **FmcNatId**). For the management option you aren’t using, leave those fields blank.

**Step 3** Generate the virtual CD-ROM by converting the text file to an ISO file:

**Example:**

```
/usr/bin/genisoimage -r -o day0.iso day0-config
```

**Step 4** Log in to your target ESXi host.

**Step 5** Open the virtual machine instance where you want to deploy the Firewall Threat Defense Virtual in cluster mode.

**Step 6** Browse and attach the day0 ISO image file that you have created to the **CD/DVD drive 1** field under **Hardware Configuration** settings before you power on the virtual machine.

**Step 7** Power on the virtual machine to deploy the Firewall Threat Defense Virtual in cluster mode.

# Deploy the Firewall Threat Defense Virtual to a vSphere ESXi Host

Use this procedure to deploy the Firewall Threat Defense Virtual appliance on a single ESXi host. You can use the VMware Host Client (or vSphere Client) to manage single ESXi hosts and to perform administrative tasks such as basic virtualization operations, such as deploying and configuring Firewall Threat Defense Virtual machines.

**Note** It is important to know that the VMware Host Client is different from the vSphere Web Client, regardless of their similar user interfaces. You use the vSphere Web Client to connect to vCenter Server and manage multiple ESXi hosts, whereas you use the VMware Host Client to manage a single ESXi host.

For instructions on how to deploy the Firewall Threat Defense Virtual appliance to a vCenter environment, see Deploy the Firewall Threat Defense Virtual to vSphere vCenter, on page 20.

### Before you begin

- You must have at least one network configured in vSphere (for management) before you deploy the Firewall Threat Defense Virtual.

### Procedure

**Step 1** Download the Firewall Threat Defense Virtual install package for VMware ESXi from Cisco.com, and save it to your local management computer:

https://www.cisco.com/go/ftd-software

A Cisco.com login and Cisco service contract is required.

**Step 2** Unpack the tar file into a working directory. Do not remove any files from the directory. The following files are included:

- Cisco_Secure_Firewall_Threat_Defense_Virtual-VI-*X.X.X-xxx*.ovf—For vCenter deployments
- Cisco_Secure_Firewall_Threat_Defense_Virtual-ESXi-*X.X.X-xxx*.ovf—For ESXi deployments.
- Cisco_Secure_Firewall_Threat_Defense_Virtual-*X.X.X-xxx*.vmdk—VMware virtual disk file.
- Cisco_Secure_Firewall_Threat_Defense_Virtual-VI-*X.X.X-xxx*.mf—Manifest file for vCenter deployments.
- Cisco_Secure_Firewall_Threat_Defense_Virtual-ESXi-*X.X.X-xxx*.mf—Manifest file for ESXi deployments.

where *X.X.X-xx* is the version and build number of the archive file you downloaded.

**Step 3** In a browser, enter the ESXi target host name or IP address using the format *http://host-name/ui* or *http://host-IP-address/ui*.

A log in screen appears.

**Step 4** Enter the administrator user name and password.

**Step 5** Click **Login** to continue.

You are now logged in to your target ESXi host.

**Step 6** Right-click on **Host** in the VMware Host Client inventory and select **Create/Register VM**.

The New Virtual Machine wizard opens.

**Step 7** On the **Select creation type** page of the wizard, select **Deploy a virtual machine from an OVF or OVA file** and click **Next**.

**Step 8** On the **Select OVF and VMDK files** page of the wizard:

a) Enter a name for your Firewall Threat Defense Virtual machine.

Virtual machine names can contain up to 80 characters and must be unique within each ESXi instance.

b) Click the blue pane, browse to the directory where you unpacked the Firewall Threat Defense Virtual tar file, and choose the ESXi OVF template and the accompanying VMDK file:

Cisco_Secure_Firewall_Threat_Defense_Virtual-ESXi-*X.X.X-xxx*.ovf

Cisco_Secure_Firewall_Threat_Defense_Virtual-*X.X.X-xxx*.vmdk

where *X.X.X-xx* is the version and build number of the archive file you downloaded.

**Attention**
Make sure you select the ESXi OVF.

**Step 9** Click **Next**.

Your local system storage opens.

**Step 10** Choose a datastore from the list of accessible datastores on the **Select storage** page of the wizard.

The datastore stores the virtual machine configuration files and all of the virtual disks. Each datastore might have a different size, speed, availability, and other properties.

**Step 11** Click **Next**.

**Step 12** Configure the **Deployment options** that come packaged with the ESXi OVF for the Firewall Threat Defense Virtual:

a) **Network Mapping**—Map the networks specified in the OVF template to networks in your inventory, and then select **Next**.

Ensure the Management0-0 interface is associated with a VM Network that is reachable from the Internet. Non-management interfaces are configurable from either the Firewall Management Center or from the Firewall Device Manager depending on your management mode.

**Important**
Firewall Threat Defense Virtual on VMware now defaults to vmxnet3 interfaces when you create a virtual device. Previously, the default was e1000. If you are using e1000 interfaces, we **strongly recommend** you switch. The

vmxnet3 device drivers and network processing are integrated with the ESXi hypervisor, so they use fewer resources and offer better network performance.

The networks may not be in alphabetical order. If it is too difficult to find your networks, you can change the networks later from the **Edit Settings** dialog box. After you deploy, right-click the Firewall Threat Defense Virtual instance, and choose **Edit Settings**. However, that screen does not show the Firewall Threat Defense Virtual IDs (only Network Adapter IDs).

See the following concordance of Network Adapter, Source Networks and Destination Networks for Firewall Threat Defense Virtual interfaces (note these are the default vmxnet3 interfaces):

***Table 8: Source to Destination Network Mapping—VMXNET3***

| Network Adapter | Source Networks | Destination Networks | Function |
|---|---|---|---|
| Network adapter 1 | Management0-0 | Management0/0 | Management |
| Network adapter 2 | Reserved for internal use. | Reserved for internal use. | Reserved for internal use. |
| Network adapter 3 | GigabitEthernet0-0 | GigabitEthernet0/0 | Outside data |
| Network adapter 4 | GigabitEthernet0-1 | GigabitEthernet0/1 | Inside data |
| Network adapter 5 | GigabitEthernet0-2 | GigabitEthernet0/2 | Data traffic (Optional) |
| Network adapter 6 | GigabitEthernet0-3 | GigabitEthernet0/3 | Data traffic (Optional) |
| Network adapter 7 | GigabitEthernet0-4 | GigabitEthernet0/4 | Data traffic (Optional) |
| Network adapter 8 | GigabitEthernet0-5 | GigabitEthernet0/5 | Data traffic (Optional) |
| Network adapter 9 | GigabitEthernet0-6 | GigabitEthernet0/6 | Data traffic (Optional) |
| Network adapter 10 | GigabitEthernet0-7 | GigabitEthernet0/7 | Data traffic (Optional) |

You can have a total of 10 interfaces when you deploy the Firewall Threat Defense Virtual. For data interfaces, make sure that the Source Networks map to the correct Destination Networks, and that each data interface maps to a unique subnet or VLAN. You do not need to use all Firewall Threat Defense Virtual interfaces; for interfaces you do not intend to use, you can simply leave the interface disabled within the Firewall Threat Defense Virtual configuration

b) **Disk provisioning**—Select the disk format to store the virtual machine virtual disks.

When you select **Thick** provisioned, all storage is immediately allocated. When you select **Thin** provisioned, storage is allocated on demand as data is written to the virtual disks. Thin provisioning can also reduce the amount of time it takes to deploy the virtual appliance.

**Step 13** On the **Ready to complete** page of the New virtual machine wizard, review the configuration settings for the virtual machine.

a) (Optional) Click **Back** to go back and review or modify the wizard settings.
b) (Optional) Click **Cancel** to discard the creation task and close the wizard.
c) Click **Finish** to complete the creation task and close the wizard.

After you complete the wizard, the ESXi host processes the VM; you can see the deployment status in the **Recent Tasks** pane. A successful deployment shows *Completed successfully* under the **Results** column.

The new Firewall Threat Defense Virtual virtual machine instance then appears under the Virtual Machines inventory of the ESXi host. Booting up the new virtual machine could take up to 30 minutes.

**Note**
To successfully register the Firewall Threat Defense Virtual with the Cisco Licensing Authority, the Firewall Threat Defense Virtual requires Internet access. You might need to perform additional configuration after deployment to achieve Internet access and successful license registration.

### What to do next

- Complete the set up of your virtual device using the CLI. This is the next step when you deploy the Firewall Threat Defense Virtual using the ESXi OVF template; see Complete the Firewall Threat Defense Virtual Setup Using the CLI, on page 33.

# Complete the Firewall Threat Defense Virtual Setup Using the CLI

If you deployed with an ESXi OVF template, you must set up the Firewall Threat Defense Virtual using the CLI. Firewall Threat Defense Virtual appliances do not have web interfaces. You can also use the CLI to configure System-required settings if you deployed with a VI OVF template and did not use the setup wizard during deployment.

**Note** If you deployed with a VI OVF template and used the setup wizard, your virtual device is configured and no further device configuration is required. Your next steps depend on which management mode you choose.

When you first log into a newly configured device, you must read and accept the EULA. Then, follow the setup prompts to change the administrator password, and configure the device's network settings and firewall mode.

When following the setup prompts, for multiple-choice questions, your options are listed in parentheses, such as (y/n). Defaults are listed in square brackets, such as [y]. Press Enter to confirm a choice.

### Procedure

**Step 1** Open the VMware console.

**Step 2** At the **firepower login** prompt, log in with the default credentials of username **admin** and the password **Admin123**.

**Step 3** When the Firewall Threat Defense Virtual system boots, a setup wizard prompts you for the following information required to configure the system:

- Accept EULA
- New admin password
- IPv4 or IPv6 configuration

- IPv4 or IPv6 DHCP settings
- Management port IPv4 address and subnet mask, or IPv6 address and prefix
- System name
- Default gateway
- DNS setup
- HTTP proxy
- Management mode (local management uses the Firewall Device Manager).

**Step 4** Review the Setup wizard settings. Defaults or previously entered values appear in brackets. To accept previously entered values, press **Enter**.

The VMware console may display messages as your settings are implemented.

**Step 5** Complete the system configuration as prompted.

**Step 6** Verify the setup was successful when the console returns to the firepower # prompt.

**Note**
To successfully register the Firewall Threat Defense Virtual with the Cisco Licensing Authority, the Firewall Threat Defense Virtual requires Internet access. You might need to perform additional configuration after deployment to achieve Internet access and successful license registration.

### What to do next

Your next steps depend on what management mode you chose.

- If you chose **No** for **Enable Local Manager**, you'll use the Firewall Management Center to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409.

See How to Manage Secure Firewall Threat Defense Virtual Device, on page 1 for an overview of how to choose your management option.

# Increasing Performance on ESXi Configurations

You can increase the performance for the Firewall Threat Defense Virtual in the ESXi environment by tuning the ESXi host CPU configuration settings. The Scheduling Affinity option gives you control over how virtual machine CPUs are distributed across the host's physical cores (and hyperthreads if hyperthreading is enabled). By using this feature, you can assign each virtual machine to processors in the specified affinity set.

See the following VMware documents for more information:

- The *Administering CPU Resources* chapter of vSphere Resource Management.
- Performance Best Practices for VMware vSphere.
- The vSphere Client online help.

# NUMA Guidelines

Non-Uniform Memory Access (NUMA) is a shared memory architecture that describes the placement of main memory modules with respect to processors in a multiprocessor system. When a processor accesses memory that does not lie within its own node (remote memory), data must be transferred over the NUMA connection at a rate that is slower than it would be when accessing local memory.

The x86 server architecture consists of multiple sockets and multiple cores within a socket. Each CPU socket along with its memory and I/O is referred to as a NUMA node. To efficiently read packets from memory, guest applications and associated peripherals (such as the NIC) should reside within the same node.

For optimum Firewall Threat Defense Virtual performance:

- The Firewall Threat Defense Virtual VM must run on a single numa node. If a single Firewall Threat Defense Virtual is deployed so that is runs across 2 sockets, the performance will be significantly degraded.
- An 8-core Firewall Threat Defense Virtual requires that each socket on the host CPU have a minimum of 8 cores per socket. Consideration must be given to other VMs running on the server.
- A 16-core Firewall Threat Defense Virtual requires that each socket on the host CPU have a minimum of 16 cores per socket. Consideration must be given to other VMs running on the server.
- The NIC should be on same NUMA node as Firewall Threat Defense Virtual VM.

More information about using NUMA systems with ESXi can be found in the VMware document *vSphere Resource Management* for your VMware ESXi version. To check for more recent editions of this and other relevant documents, see http://www.vmware.com/support/pubs....

# SR-IOV Interface Provisioning

Single Root I/O Virtualization (SR-IOV) allows multiple VMs running a variety of guest operating systems to share a single PCIe network adapter within a host server. SR-IOV allows a VM to move data directly to and from the network adapter, bypassing the hypervisor for increased network throughput and lower server CPU burden. Recent x86 server processors include chipset enhancements, such as Intel VT-d technology, that facilitate direct memory transfers and other operations required by SR-IOV.

The SR-IOV specification defines two device types:

- Physical Function (PF)—Essentially a static NIC, a PF is a full PCIe device that includes SR-IOV capabilities. PFs are discovered, managed, and configured as normal PCIe devices. A single PF can provide management and configuration for a set of virtual functions (VFs).
- Virtual Function (VF)—Similar to a dynamic vNIC, a VF is a full or lightweight virtual PCIe device that provides at least the necessary resources for data movements. A VF is not managed directly but is derived from and managed through a PF. One or more VFs can be assigned to a VM.

VFs are capable of providing up to 10 Gbps connectivity to Firewall Threat Defense Virtual machines within a virtualized operating system framework. This section explains how to configure VFs in a VMware environment.

# Best Practices for SR-IOV Interfaces

### Guidelines for SR-IOV Interfaces

VMware vSphere 5.1 and later releases support SR-IOV in an environment with specific configurations only. Some features of vSphere are not functional when SR-IOV is enabled.

In addition to the System Requirements for the Firewall Threat Defense Virtual and SR-IOV, you should review the Supported Configurations for Using SR-IOV in the VMware documentation for more information about requirements, supported NICs, availability of features, and upgrade requirements for VMware and SR-IOV.

Firewall Threat Defense Virtual on VMware using the SR-IOV interface supports mixing of interface types. You can use SR-IOV or VMXNET3 for the management interface and SR-IOV for the data interface.

This section shows various setup and configuration steps for provisioning SR-IOV interfaces on a VMware system. The information in this section was created from devices in a specific lab environment, using VMware ESXi 6.0 and vSphere Web Client, a Cisco UCS C Series server, and an Intel Ethernet Server Adapter X520 - DA2.

### Limitations for SR-IOV Interfaces

When the Firewall Threat Defense Virtual is booted, be aware that SR-IOV interfaces can show up in reverse order when compared to the order presented in ESXi. This could cause interface configuration errors that result in a lack of network connectivity for a particular Firewall Threat Defense Virtual machine.

**Caution** It is important that you verify the interface mapping before you begin configuring the SR-IOV network interfaces on the Firewall Threat Defense Virtual. This ensures that the network interface configuration will apply to the correct physical MAC address interface on the VM host.

After the Firewall Threat Defense Virtual boots, you can confirm which MAC address maps to which interface. Use the **show interface** command to see detailed interface information, including the MAC address for an interface. Compare the MAC address to the results of the **show kernel ifconfig** command to confirm the correct interface assignment.

NOTE:

### Limitations of using ixgbe-vf Interfaces

Be aware of the following limitations when using ixgbe-vf interfaces:

- The guest VM is not allowed to set the VF to promiscuous mode. Because of this, transparent mode is not supported when using ixgbe-vf.
- The guest VM is not allowed to set the MAC address on the VF. Because of this, the MAC address is not transferred during HA like it is done on other Firewall Threat Defense Virtual platforms and with other interface types. HA failover works by transferring the IP address from active to standby.

**Note** This limitation is applicable to the i40e-vf interfaces too.

- The Cisco UCS-B server does not support the ixgbe-vf vNIC.

- In a failover setup, when a paired Firewall Threat Defense Virtual (primary unit) fails, the standby Firewall Threat Defense Virtual unit takes over as the primary unit role and its interface IP address is updated with a new MAC address of the standby Firewall Threat Defense Virtual unit. Thereafter, the Firewall Threat Defense Virtual sends a gratuitous Address Resolution Protocol (ARP) update to announce the change in MAC address of the interface IP address to other devices on the same network. However, due to incompatibility with these types of interfaces, the gratuitous ARP update is not sent to the global IP address that is defined in the NAT or PAT statements for translating the interface IP address to global IP addresses.

# Check the ESXi Host BIOS

### Before you begin

To deploy the Firewall Threat Defense Virtual with SR-IOV interfaces on VMware, virtualization needs to be supported and enabled. VMware provides several methods of verifying virtualization support, including their online Compatibility Guide for SR-IOV support as well as a downloadable CPU identification utility that detects whether virtualization is enabled or disabled.

You can also determine if virtualization is enabled in the BIOS by logging into the ESXi host.

### Procedure

**Step 1** Log in to the ESXi Shell using one of the following methods:

- If you have direct access to the host, press Alt+F2 to open the login page on the machine's physical console.
- If you are connecting to the host remotely, use SSH or another remote console connection to start a session on the host.

**Step 2** Enter a user name and password recognized by the host.

**Step 3** Run the following commands:

```
esxcfg-info|grep "\----\HV Support"
```

- The output of the HV Support command indicates the type of hypervisor support available. These are the descriptions for the possible values:
- 0 - VT/AMD-V indicates that support is not available for this hardware.
- 1 - VT/AMD-V indicates that VT or AMD-V might be available but it is not supported for this hardware.
- 2 - VT/AMD-V indicates that VT or AMD-V is available but is currently not enabled in the BIOS.
- 3 - VT/AMD-V indicates that VT or AMD-V is enabled in the BIOS and can be used.

```
~ # esxcfg-info|grep "\----\HV Support"
         |----HV Support.........................3
```

The value 3 indicates that virtualization is supported and enabled.

### What to do next

Enable SR-IOV on the host physical adapter.

# Enable SR-IOV on the Host Physical Adapter

Before you can connect virtual machines to virtual functions, use the vSphere Web Client to enable SR-IOV and set the number of virtual functions on your host.

### Before you begin

- Make sure you have an SR-IOV-compatible network interface card (NIC) installed; see System Requirements, on page 4.

### Procedure

**Step 1** In the vSphere Web Client, navigate to the ESXi host where you want to enable SR-IOV.

**Step 2** On the **Manage** tab, click **Networking** and choose **Physical adapters**.

You can look at the SR-IOV property to see whether a physical adapter supports SR-IOV.

**Step 3** Select the physical adapter and click **Edit adapter settings**.

**Step 4** Under SR-IOV, select **Enabled** from the **Status** drop-down menu.

**Step 5** In the **Number of virtual functions** text box, type the number of virtual functions that you want to configure for the adapter.

**Note**
We recommend that you **DO NOT** use more than 1 VF per interface. Performance degradation is likely to occur if you share the physical interface with multiple virtual functions.

**Step 6** Click **OK**.

**Step 7** Restart the ESXi host.

The virtual functions become active on the NIC port represented by the physical adapter entry. They appear in the PCI Devices list in the **Settings** tab for the host.

### What to do next

- Create a standard vSwitch to manage the SR-IOV functions and configurations.

# Create a vSphere Switch

Create a vSphere switch to manage the SR-IOV interfaces.

## Procedure

**Step 1** In the vSphere Web Client, navigate to the ESXi host.

**Step 2** Under **Manage** select **Networking**, and then select **Virtual switches**.

**Step 3** Click the **Add host networking** icon, which is the green globe icon with the plus (+) sign.

**Step 4** Select a **Virtual Machine Port Group for a Standard Switch** connection type and click **Next**.

**Step 5** Choose **New standard switch** and click **Next**.

**Step 6** Add physical network adapters to the new standard switch.

a) Under Assigned adapters, click the green plus (+) sign to **Add adapters**.

b) Select the corresponding network interface for SR-IOV from the list. For example, Intel(R) 82599 10 Gigabit Dual Port Network Connection.

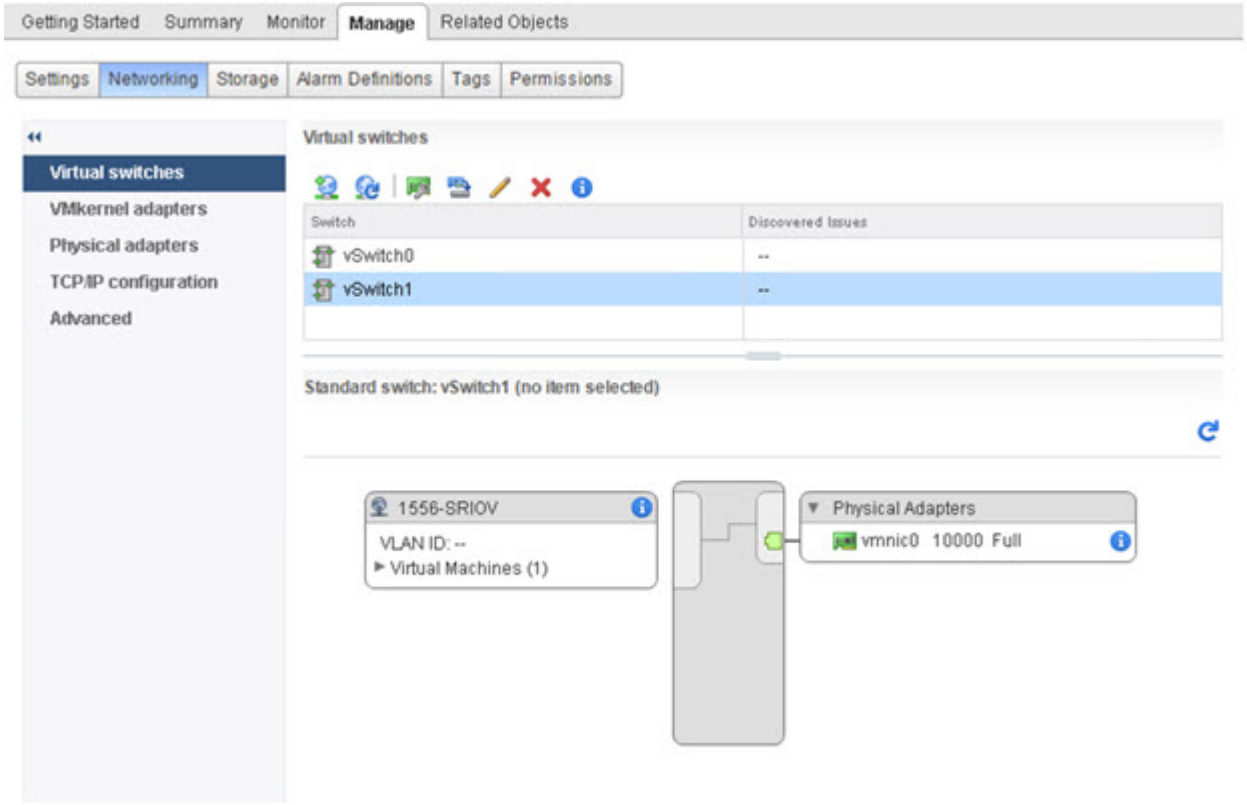
c) From the **Failover order group** drop-down menu, select from the **Active adapters**.

d) Click **OK**.

**Step 7** Enter a **Network label** for the SR-IOV vSwitch and click **Next**.

**Step 8** Review your selections on the **Ready to complete** page, then click **Finish**.

*Figure 2: New vSwitch with an SR-IOV Interface attached*



### What to do next

- Review the compatibility level of your virtual machine.

# Upgrade the Compatibility Level for Virtual Machines

The compatibility level determines the virtual hardware available to the virtual machine, which corresponds to the physical hardware available on the host machine. The Firewall Threat Defense Virtual VM needs to be at Hardware Level 10 or higher. This will expose the SR-IOV passthough feature to the Firewall Threat Defense Virtual. This procedure upgrades the Firewall Threat Defense Virtual to the latest supported virtual hardware version immediately.

For information about virtual machine hardware versions and compatibility, see the vSphere Virtual Machine Administration documentation.

### Procedure

**Step 1** Log in to the vCenter Server from the vSphere Web Client.

**Step 2** Locate the Firewall Threat Defense Virtual machine that you want to modify.

a) Select a datacenter, folder, cluster, resource pool, or host and click the **Related Objects** tab.
b) Click **Virtual Machines** and select the Firewall Threat Defense Virtual machine from the list.

**Step 3** Power off the selected virtual machine.

**Step 4** Right-click the Firewall Threat Defense Virtual and select **Actions** > **All vCenter Actions** > **Compatibility** > **Upgrade VM Compatibility**.

**Step 5** Click **Yes** to confirm the upgrade.

**Step 6** Choose the **ESXi 5.5 and later** option for the virtual machines compatiblity.

**Step 7** (Optional) Select **Only upgrade after normal guest OS shutdown**.

The selected virtual machine is upgraded to the corresponding hardware version for the Compatibility setting that you chose, and the new hardware version is updated in the **Summary** tab of the virtual machine.

### What to do next

- Associate the Firewall Threat Defense Virtual with a virtual function through an SR-IOV passthrough network adapter.

# Assign the SR-IOV NIC to the Firewall Threat Defense Virtual

To ensure that the Firewall Threat Defense Virtual machine and the physical NIC can exchange data, you must associate the Firewall Threat Defense Virtual with one or more virtual functions as SR-IOV passthrough network adapters. The following procedure explains how to assign the SR-IOV NIC to the Firewall Threat Defense Virtual machine using the vSphere Web Client.

### Procedure

**Step 1** Log in to the vCenter Server from the vSphere Web Client.

**Step 2** Locate the Firewall Threat Defense Virtual machine you wish to modify.

a) Select a datacenter, folder, cluster, resource pool, or host and click the **Related Objects** tab.

b) Click **Virtual Machines** and select the Firewall Threat Defense Virtual machine from the list.

**Step 3** On the **Manage** tab of the virtual machine, select **Settings** > **VM Hardware**.

**Step 4** Click **Edit** and choose the **Virtual Hardware** tab.

**Step 5** From the **New device** drop-down menu, select **Network** and click **Add**.

A **New Network** interface appears.

**Step 6** Expand the **New Network** section and select an available SRIOV option.

**Step 7** From the **Adapter Type** drop-down menu, select **SR-IOV passthrough**.

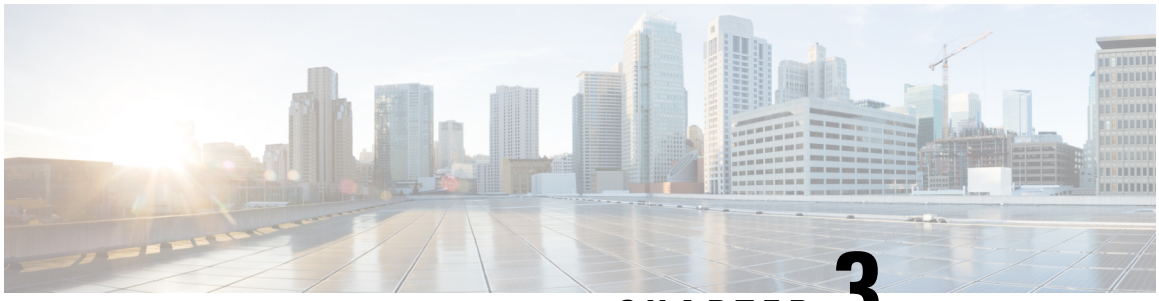
**Step 8** From the **Physical function** drop-down menu, select the physical adapter that corresponds to the passthrough virtual machine adapter.

**Step 9** Power on the virtual machine.

When you power on the virtual machine, the ESXi host selects a free virtual function from the physical adapter and maps it to the SR-IOV passthrough adapter. The host validates all properties of the virtual machine adapter and the underlying virtual function.



**Note** Using SR-IOV interfaces as passive interfaces on the Firewall Threat Defense Virtual is not supported on some Intel network adapters (such as Intel X710 or 82599) using SR-IOV drivers due to a promiscuous mode restriction. In such cases, use a network adapter that supports this functionality. See Intel Ethernet Products for more information on Intel network adapters.



CHAPTER 3

# Deploy the Firewall Threat Defense Virtual on KVM

This chapter describes the procedures to deploy the Firewall Threat Defense Virtual to a KVM environment.


- Overview, on page 43
- System Requirements, on page 44
- Guidelines and Limitations, on page 45
- How to Manage Secure Firewall Threat Defense Virtual Device, on page 49
- Prerequisites, on page 50
- End-to-End Procedure, on page 51
- Prepare the Day 0 Configuration File, on page 53
- Launch the Firewall Threat Defense Virtual, on page 54
- Troubleshooting, on page 60


## Overview

KVM is a full virtualization solution for Linux on x86 hardware containing virtualization extensions (such as Intel VT). It consists of a loadable kernel module, kvm.ko, that provides the core virtualization infrastructure and a processor specific module, such as kvm-intel.ko.

You can run multiple virtual machines running unmodified OS images using KVM. Each virtual machine has private virtualized hardware: a network card, disk, graphics adapter, and so forth.

### Performance Tiers for Firewall Threat Defense Virtual Smart Licensing

The Firewall Threat Defense Virtual supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.

***Table 9: Firewall Threat Defense Virtual Licensed Feature Limits Based on Entitlement***

| Performance Tier | Device Specifications (Core/RAM) | Rate Limit | RA VPN Session Limit |
|---|---|---|---|
| FTDv5, 100Mbps | 4 core/8 GB | 100Mbps | 50 |
| FTDv10, 1Gbps | 4 core/8 GB | 1Gbps | 250 |

| Performance Tier | Device Specifications (Core/RAM) | Rate Limit | RA VPN Session Limit |
|---|---|---|---|
| FTDv20, 3Gbps | 4 core/8 GB | 3Gbps | 250 |
| FTDv30, 5Gbps | 8 core/16 GB | 5Gbps | 250 |
| FTDv50, 10Gbps | 12 core/24 GB | 10Gbps | 750 |
| FTDv100, 16Gbps | 16 core/32 GB | 16Gbps | 10,000 |

See the "Licensing the System" chapter in the *Secure Firewall Management Center* for guidelines when licensing your Firewall Threat Defense Virtual device.

# System Requirements

See the Cisco Secure Firewall Threat Defense Compatibility Guide for the most current information about hypervisor support for the Firewall Threat Defense Virtual.

The specific hardware used for the Firewall Threat Defense Virtual deployments can vary, depending on the number of instances deployed and usage requirements. Each instance of the Firewall Threat Defense Virtual requires a minimum resource allocation—amount of memory, number of CPUs, and disk space—on the server.

***Table 10: Firewall Threat Defense Virtual Appliance Resource Requirements***

| Settings | Value |
|---|---|
| Performance Tiers | **Version 7.0 and later**<br>The Firewall Threat Defense Virtual supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.<br>• FTDv5 4vCPU/8GB (100Mbps)<br>• FTDv10 4vCPU/8GB (1Gbps)<br>• FTDv20 4vCPU/8GB (3Gbps)<br>• FTDv30 8vCPU/16GB (5Gbps)<br>• FTDv50 12vCPU/24GB (10Gbps)<br>• FTDv100 16vCPU/32GB (16Gbps)<br>See the "Licensing the System" chapter in the *Secure Firewall Management Center Configuration* for guidelines when licensing your Firewall Threat Defense Virtual device.<br>**Note**<br>To change the vCPU/memory values, you must first power off the Firewall Threat Defense Virtual device. |

| Settings | Value |
|---|---|
| Number of cores and memory | **Version 6.4 to Version 6.7**<br>The Firewall Threat Defense Virtual deploys with adjustable vCPU and memory resources. There are three supported vCPU/memory pair values:<br>• 4vCPU/8GB (default)<br>• 8vCPU/16GB<br>• 12vCPU/24GB<br>**Note**<br>To change the vCPU/memory values, you must first power off the Firewall Threat Defense Virtual device. Only the above three combinations are supported. |
| | **Version 6.3 and earlier**<br>The Firewall Threat Defense Virtual deploys with fixed vCPU and memory resources. There is only one supported vCPU/memory pair value:<br>• 4vCPU/8GB<br>**Note**<br>Adjustments to vCPUs and memory are not supported. |
| Hard disk provisioned size | • 50 GB<br>• Adjustable setting. Supports virtio block devices |
| vNICs | The Firewall Threat Defense Virtual on KVM supports the following virtual network adapters:<br>• **VIRTIO**—Virtio is the main platform for IO virtualization in KVM and provides a common framework for hypervisors for IO virtualization. The host implementation is in userspace - qemu, so no driver is needed in the host.<br>• **IXGBE-VF**—The ixgbe-vf (10 Gbit/s) driver supports virtual function devices that can only be activated on kernels that support SR-IOV. SR-IOV requires the correct platform and OS support; see Support for SR-IOV for more information. |

# Guidelines and Limitations

- Requires two management interfaces and two data interfaces to boot.

**Note** The Firewall Threat Defense Virtual default configuration puts the management interface and inside interface on the same subnet.

- Supports virtIO drivers.
- Supports ixgbe-vf drivers for SR-IOV.
- Supports a total of 11 interfaces.

- The default configuration for the Firewall Threat Defense Virtual assumes that you put both the management and inside interfaces on the **same subnet**, and the management address uses the inside address as its gateway to the Internet (going through the outside interface).
- The Firewall Threat Defense Virtual must be powered up on firstboot with at least four interfaces. Your system will not deploy without four interfaces
- The Firewall Threat Defense Virtual supports a total of 11 interfaces—1 management interface, 1 interface reserved for internal use, and a maximum of 9 network interfaces for data traffic. The interface-to-network assignments must be ordered as follows:
  - Management interface (1) (required)

**Note** You can optionally configure a data interface for the Firewall Management Center management instead of the Management interface. The Management interface is a pre-requisite for data interface management, so you still need to configure it in your initial setup. Note that the Firewall Management Center access from a data interface is not supported in High Availability deployments. For more information about configuring a data interface for the Firewall Management Center access, see the **configure network management-data-interface** command in the FTD command reference.

  - Reserved for internal use (2) (required)
  - Outside interface (3) (required)
  - Inside interface (4) (required)
  - Data interfaces (5-11) (optional)

See the following concordance of Network Adapter, Source Networks and Destination Networks for the Firewall Threat Defense Virtual interfaces:

***Table 11: Source to Destination Network Mapping***

| Network Adapter | Source Network | Destination Network | Function |
|---|---|---|---|
| vnic0* | Management0-0 | Management0/0 | Management |
| vnic1 | Reserved for internal use. | Reserved for internal use. | Reserved for internal use. |

| Network Adapter | Source Network | Destination Network | Function |
|---|---|---|---|
| vnic2 | GigabitEthernet0-0 | GigabitEthernet0/0 | Outside |
| vnic3* | GigabitEthernet0-1 | GigabitEthernet0/1 | Inside |
| *Important. Attach to the same subnet. | | | |

- Cloning a virtual machine is not supported.
- For console access, supports terminal server via telnet.
- For creating vNICs with IPv6 support configuration on KVM, you must create an XML file for each interface that consists of IPv6 configuration parameters. You can install vNICs with the IPV6 network protocol configurations by running these XML files using the command **virsh net-create** <<*interface configuration XML file name*>>.

  For each interface, you can create the following XML file:

  - Management interface - *mgmt-vnic.xml*
  - Inside interface - *inside-vnic.xml*
  - Outside interface - *outside-vnic.xml*

  **Example:**

  To create an XML file for Management interface with IPv6 configuration.

```
<network>
            <name>mgmt-vnic</name>
            <bridge name='mgmt-vnic' stp='on' delay='0' />
            <ip family='ipv6' address='2001:db8::a111:b220:0:abcd' prefix='96'/>
       </network>
```

  Similarly, you must create XML file for other interfaces.

  You can verify the virtual network adapters installed on KVM by running the following command.

```
virsh net-list
        brctl show
```

### CPU Mode

KVM can emulate a number different of CPU types. For your VM, you typically should select a processor type which closely matches the CPU of the host system, as it means that the host CPU features (also called CPU flags) will be available in your VMs. You should set the CPU type to **host** in which case the VM will have exactly the same CPU flags as your host system.

### Clustering

Clustering is supported on threat defense virtual instances deployed on KVM. See Clustering for Threat Defense Virtual in a Private Cloud for more information.

### Performance Optimizations

To achieve the best performance out of the Firewall Threat Defense Virtual, you can make adjustments to the both the VM and the host. See Virtualization Tuning and Optimization on KVM for more information.

**Receive Side Scaling**—The Firewall Threat Defense Virtual supports Receive Side Scaling (RSS), which is a technology utilized by network adapters to distribute network receive traffic to multiple processor cores. Supported on Version 7.0 and later. See Multiple RX Queues for Receive Side Scaling (RSS) for more information.

### Support for SR-IOV

SR-IOV Virtual Functions require specific system resources. A server that supports SR-IOV is required in addition to an SR-IOV capable PCIe adapter.

Firewall Threat Defense Virtual on KVM using the SR-IOV interfaces supports mixing interface types. You can use SR-IOV or VMXNET3 for the management interface and SR-IOV for the data interface.

You must be aware of the following hardware considerations:

- The capabilities of SR-IOV NICs, including the number of VFs available, differ across vendors and devices. The following NICs are supported:
  - Intel Ethernet Server Adapter X710
  - Intel Ethernet Server Adapter X520 - DA2
  - Intel Ethernet Network Adapter E810-CQDA2
    - The firmware (NVM image) and network driver is updated on Intel® Network Adapter E810 using an NVM utility tool. The Non-Volatile Memory (NVM) image and network driver are a compatible set of components that you update as a combination on the Intel® Network Adapter E810. For information on NVM and Software compatibility matrix, refer to the Intel® Ethernet Controller E810 Datasheet to update the correct firmware drivers on Intel® Network Adapter E810.
- Not all PCIe slots support SR-IOV.
- SR-IOV-capable PCIe slots may have different capabilities.
- x86_64 multicore CPU — Intel Sandy Bridge or later (Recommended).

**Note** We tested the Firewall Threat Defense Virtual on Intel's Broadwell CPU (E5-2699-v4) at 2.3GHz.

- Cores
  - Minimum of 8 physical cores per CPU socket.
  - The 8 cores must be on a single socket.

**Note** CPU pinning is recommended to achieve full throughput.

- You should consult your manufacturer's documentation for SR-IOV support on your system. For KVM, you can verify CPU compatibility for SR-IOV support. Note that for the Firewall Threat Defense Virtual on KVM we only support x86 hardware.

### Limitations of using ixgbe-vf Interfaces

Be aware of the following limitations when using ixgbe-vf interfaces:

- The guest VM is not allowed to set the VF to promiscuous mode. Because of this, transparent mode is not supported when using ixgbe-vf.
- The guest VM is not allowed to set the MAC address on the VF. Because of this, the MAC address is not transferred during HA like it is done on other Firewall Threat Defense Virtual platforms and with other interface types. HA failover works by transferring the IP address from active to standby.



**Note** This limitation is applicable to the i40e-vf interfaces too.

- The Cisco UCS-B server does not support the ixgbe-vf vNIC.
- In a failover setup, when a paired Firewall Threat Defense Virtual (primary unit) fails, the standby Firewall Threat Defense Virtual unit takes over as the primary unit role and its interface IP address is updated with a new MAC address of the standby Firewall Threat Defense Virtual unit. Thereafter, the Firewall Threat Defense Virtual sends a gratuitous Address Resolution Protocol (ARP) update to announce the change in MAC address of the interface IP address to other devices on the same network. However, due to incompatibility with these types of interfaces, the gratuitous ARP update is not sent to the global IP address that is defined in the NAT or PAT statements for translating the interface IP address to global IP addresses.

### Snort

- If you are observing abnormal behavior such as Snort taking a long time to shut down, or the VM being slow in general or when a certain process is executed, collect logs from the Firewall Threat Defense Virtual and the VM host. Collection of overall CPU usage, memory, I/O usage, and read/write speed logs will help troubleshoot the issues.
- High CPU and I/O usage is observed when Snort is shutting down. If a number of Firewall Threat Defense Virtual instances have been created on a single host with insufficient memory and no dedicated CPU, Snort will take a long time to shut down which will result in the creation of Snort cores.

# How to Manage Secure Firewall Threat Defense Virtual Device

You have two options to manage your Secure Firewall Threat Defense Virtual.

## Secure Firewall Management Center

If you are managing large numbers of devices, or if you want to use the more complex features and configurations that the Firewall Threat Defense allows, use the Firewall Management Center to configure your devices instead of the integrated Firewall Device Manager.

**Important** You cannot use both the Firewall Device Manager and the Firewall Management Center to manage the Firewall Threat Defense device. Once the Firewall Device Manager integrated management is enabled, it won't be possible to use the Firewall Management Center to manage the Firewall Threat Defense device, unless you disable the local management and re-configure the management to use the Firewall Management Center. On the other hand, when you register the Firewall Threat Defense device to the Firewall Management Center, the Firewall Device Manager onboard management service is disabled.

**Caution** Currently, Cisco does not have an option to migrate your Firewall Device Manager configuration to the Firewall Management Center and vice-versa. Take this into consideration when you choose what type of management you configure for the Firewall Threat Defense device.

## Secure Firewall Device Manager

The Firewall Device Manager is a web interface included on most Firewall Threat Defense devices. It lets you configure the basic features of the software that are most commonly used for small networks. It is especially designed for networks that include a single device or just a few, where you do not want to use a high-powered multiple-device manager to control a large network with many devices.

**Note** See the Cisco Secure Firewall Threat Defense Compatibility Guide for list of devices that support the Firewall Device Manager.

# Prerequisites

- Download the Firewall Threat Defense Virtual qcow2 file from Cisco.com and put it on your Linux host:

  https://software.cisco.com/download/navigator.html

**Note** A Cisco.com login and Cisco service contract are required.

- For the purpose of the sample deployment in this document, we are assuming you are using Ubuntu 18.04 LTS. Install the following packages on top of the Ubuntu 18.04 LTS host:
  - qemu-kvm
  - libvirt-bin
  - bridge-utils
  - virt-manager
  - virtinst
  - virsh tools

- genisoimage

- Performance is affected by the host and its configuration. You can maximize the throughput of the Firewall Threat Defense Virtual on KVM by tuning your host. For generic host-tuning concepts, see Network Function Virtualization: Quality of Service in Broadband Remote Access Servers with Linux and Intel Architecture.
- Useful optimizations for Ubuntu 18.04 LTS include the following:
  - macvtap—High performance Linux bridge; you can use macvtap instead of a Linux bridge. You must configure specific settings to use macvtap instead of the Linux bridge.
  - Transparent Huge Pages—Increases memory page size and is on by default in Ubuntu 18.04.
  - Hyperthread disabled—Reduces two vCPUs to one single core.
  - txqueuelength—Increases the default txqueuelength to 4000 packets and reduces drop rate.
  - pinning—Pins qemu and vhost processes to specific CPU cores; under certain conditions, pinning is a significant boost to performance.
- For information on optimizing a RHEL-based distribution, see Red Hat Enterprise Linux6 Virtualization Tuning and Optimization Guide.
- For KVM and System compatibility, see Cisco Secure Firewall Threat Defense Compatibility Guide.
- You can use the following methods to verify whether your virtual machine is running the KVM:
  - Run the **lsmod** to list the modules in the Linux Kernel. If the KVM is running, it is indicated by displaying the following output:

    **root@kvm-host:~$ lsmod | grep kvm**

    **kvm_intel 123675 0**

    **kvm 257361 1 kvm_intel**
- If **ls -l /dev/kvm** does not exist on the target VM then you are probably running **qemu**, and not taking advantage of the KVM hardware assist features.

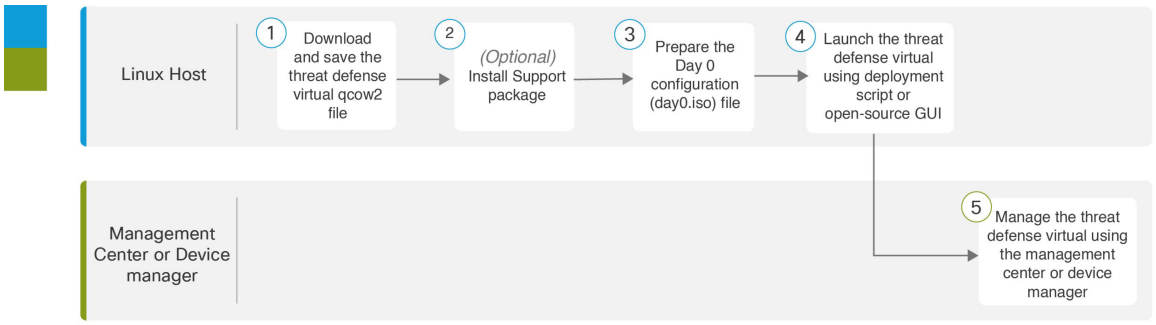
  **root@kvm-host:~$ ls -l /dev/kvm**

  **crw------- 1 root root 10, 232 Mar 23 13:53 /dev/kvm**
- Run the following command to also check whether the host machine supports KVM:

  **root@kvm-host:~$ sudo kvm-ok**
- You can also use KVM acceleration.

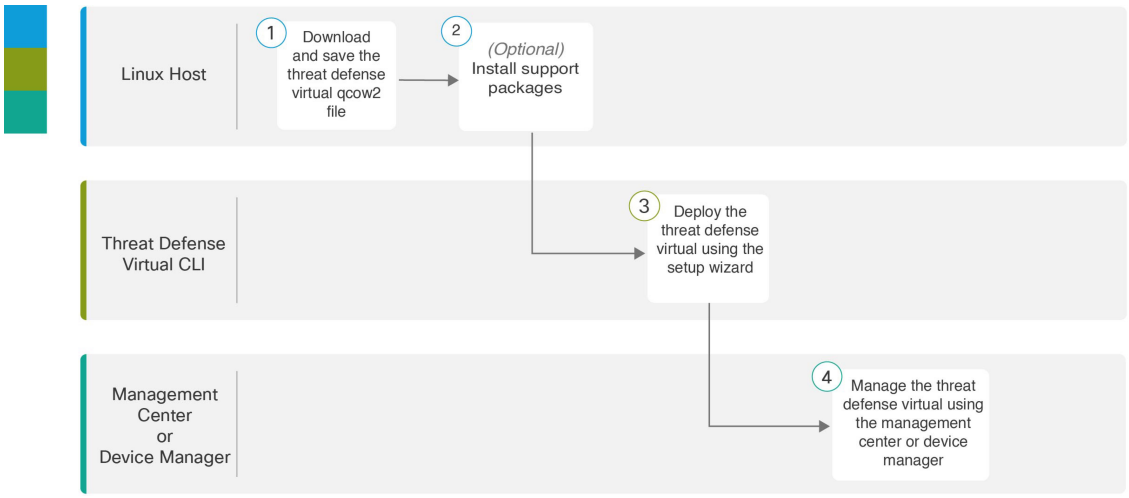
# End-to-End Procedure

The following flowchart illustrates the workflow for deploying Firewall Threat Defense Virtual on a KVM instance using a Day 0 configuration file.




| | Workspace | Steps |
|---|---|---|
| 1 | Linux Host | Prerequisites, on page 50: Download and save the Firewall Threat Defense Virtual qcow2 file on the Linux host. |
| 2 | Linux Host | Prerequisites, on page 50: Install support packages. |
| 3 | Linux Host | Prepare the Day 0 Configuration File |
| 4 | Linux Host | Launch the Firewall Threat Defense Virtual:<br>• Using a Deployment script<br>• Using a GUI |
| 5 | Firewall Management Center | Manage the Firewall Threat Defense Virtual by using the Management Center. |

The following flowchart illustrates the workflow for deploying the Firewall Threat Defense Virtual on a KVM instance without using a Day 0 configuration file.



| | Workspace | Steps |
|---|---|---|
| 1 | Linux Host | Prerequisites, on page 50: Download and save the Firewall Threat Defense Virtual qcow2 file on the Linux host. |
| 2 | Linux Host | Prerequisites, on page 50: Install support packages. |
| 3 | Firewall Threat Defense Virtual CLI | Launch Without the Day 0 Configuration File: Deploy the Firewall Threat Defense Virtual using the setup wizard. |
| 4 | Firewall Management Center | Manage the Firewall Threat Defense Virtual by using the Management Center |

# Prepare the Day 0 Configuration File

You can prepare a Day 0 configuration file before you deploy Firewall Threat Defense. This file is a text file that contains the initial configuration data that gets applied at the time a virtual machine is deployed. This initial configuration is placed into a text file named “day0-config” in a working directory you choose, and is manipulated into a day0.iso file that is mounted and read on first boot.

**Important** The day0.iso file must be available during first boot.

If you deploy with a Day 0 configuration file, the process allows you to perform the entire initial setup. You can specify:

- The End User License Agreement (EULA) acceptance.
- A host name for the system.
- A new administrator password for the admin account.
- The management mode; see How to Manage Secure Firewall Threat Defense Virtual Device, on page 1.

  You can either set **ManageLocally** to **Yes**, or enter information for the Firewall Management Center fields (**FmcIp**, **FmcRegKey**, and **FmcNatId**). Leave fields empty for the management mode you are not using.
- The initial firewall mode; sets the initial firewall mode, either **routed** or **transparent**.

  If you plan to manage your deployment using the local Firewall Device Manager, you can only enter **routed** for firewall mode. You cannot configure transparent firewall mode interfaces using the Firewall Device Manager.
- Network settings that allow the device to communicate on your management network.
- The deployment type where you can specify whether you are deploying in cluster or standalone mode.

If you deploy without a Day 0 configuration file, you must configure system-required settings after launch; see Launch Without the Day 0 Configuration File, on page 59 for more information.

**Note** We are using Linux in this example, but there are similar utilities for Windows.

### Procedure

**Step 1** Enter the CLI configuration for the device in a text file called "day0-config". Add network settings and information about managing the device.

**Example:**

Enter **Yes** for **ManageLocally** in your Day 0 configuration file to use the local Firewall Device Manager; or enter the Firewall Management Center fields (**FmcIp**, **FmcRegKey**, and **FmcNatId**). For the management option you aren't using, leave those fields blank.

**Step 2** Generate the virtual CD-ROM by converting the text file to an ISO file:

**Example:**

```
/usr/bin/genisoimage -r -o day0.iso day0-config
```

or

**Example:**

```
/usr/bin/mkisofs -r -o day0.iso day0-config
```

**Step 3** Repeat to create unique default configuration files for each of the Firewall Device Manager you want to deploy.

#### What to do next

- If using virt-install, add the following line to the virt-install command:

  ```
  --disk path=/home/user/day0.iso,format=iso,device=cdrom \
  ```

- If using virt-manager, you can create a virtual CD-ROM using the virt-manager GUI; see Launch Using a Graphical User Interface (GUI), on page 56.

# Launch the Firewall Threat Defense Virtual

## Launch Using a Deployment Script

Use a virt-install based deployment script to launch the Firewall Threat Defense Virtual.

Be aware that you can optimize performance by selecting the best guest caching mode for your environment. The cache mode in use will affect whether data loss occurs, and the cache mode can also affect disk performance.

Each KVM guest disk interface can have one of the following cache modes specified: *writethrough*, *writeback*, *none*, *directsync*, or *unsafe*. *writethrough* provides read caching. *writeback* provides read and write caching.

*directsync* bypasses the host page cache. *unsafe* may cache all content and ignore flush requests from the guest.

- A *cache=writethrough* will help reduce file corruption on KVM guest machines when the host experiences abrupt losses of power. We recommend that you use *writethrough* mode.
- However, *cache=writethrough* can also affect disk performance due to more disk I/O writes than *cache=none*.
- If you remove the cache parameter on the *--disk* option, the default is *writethrough*.
- Not specifying a cache option may also significantly reduce the time required for the VM creation. This is due to the fact that some older RAID controllers have poor disk caching capability. Hence, disabling disk caching (*cache=none*) and thus defaulting to *writethrough*, helps ensure data integrity.
- Starting with version 6.4, the Firewall Threat Defense Virtual deploys with adjustable vCPU and memory resources. Prior to version 6.4, the Firewall Threat Defense Virtual deployed as a fixed configuration 4vCPU/8GB device. See the following table for supported values for the --vcpus and --ram parameters for each Firewall Threat Defense Virtual platform size.

***Table 12: Supported vCPU and Memory Parameters for virt-install***

| --vcpus | --ram | Firewall Threat Defense Virtual Platform Size |
|---|---|---|
| 4 | 8192 | 4vCPU/8GB (default) |
| 8 | 16384 | 8vCPU/16GB |
| 12 | 24576 | 12vCPU/24GB |

## Procedure

**Step 1** Create a virt-install script called “virt_install_ftdv.sh”.

The name of the Firewall Threat Defense Virtual VM must be unique across all other virtual machines (VMs) on this KVM host. The Firewall Threat Defense Virtual can support up to 10 network interfaces. This example uses four interfaces. The virtual NIC must be VirtIO.

**Note**
The default configuration for the Firewall Threat Defense Virtual assumes that you put the management interface, diagnostic interface, and inside interface on the **same subnet**. The system requires at least 4 interfaces to successfully boot up. The interface-to-network assignments must be ordered as follows:

- (1) Management interface (required)
- (2) Reserved for internal use (required)
- (3) Outside interface (required)
- (4) Inside interface (required)
- (5) (Optional) Data interfaces—up to 6

**Example:**

```
virt-install \
   --connect=qemu:///system \
   --network network=default,model=virtio \
   --network network=default,model=virtio \
   --network network=default,model=virtio \
   --network network=default,model=virtio \
   --name=ftdv \
   --arch=x86_64 \
   --cpu host \
   --vcpus=8 \
   --ram=16384 \
   --os-type=linux \
   --os-variant=generic26 \
   --virt-type=kvm \
   --import \
   --watchdog i6300esb,action=reset \
   --disk path=<ftd_filename>.qcow2,format=qcow2,device=disk,bus=virtio,cache=none \
   --disk path=<day0_filename>.iso,format=iso,device=cdrom \
   --console pty,target_type=serial \
   --serial tcp,host=127.0.0.1:<port>,mode=bind,protocol=telnet \
   --force
```

**Step 2** Run the virt_install script:

**Example:**

```
/usr/bin/virt_install_ftdv.sh

Starting install...
Creating domain...
```

A window appears displaying the console of the VM. You can see that the VM is booting. It takes a few minutes for the VM to boot. Once the VM stops booting, you can issue CLI commands from the console screen.

### What to do next

Your next steps depend on what management mode you chose.

- If you chose **No** for **Manage Locally**, you'll use the Firewall Management Center to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409.

See How to Manage Secure Firewall Threat Defense Virtual Device, on page 1 for an overview of how to choose your management option.

# Launch Using a Graphical User Interface (GUI)

There are a several open-source options available to manage KVM virtual machines using a GUI. The following procedure uses virt-manager, also known as Virtual Machine Manager, to launch the Firewall Threat Defense Virtual. virt-manager is a graphical tool for creating and managing guest virtual machines.

**Note** KVM can emulate a number different of CPU types. For your VM, you typically should select a processor type which closely matches the CPU of the host system, as it means that the host CPU features (also called CPU flags) will be available in your VMs. You should set the CPU type to **host** in which case the VM will have exactly the same CPU flags as your host system.

### Procedure

**Step 1** Start virt-manager **(Applications** > **System Tools** > **Virtual Machine Manager**).

You may be asked to select the hypervisor and/or enter your root password.

**Step 2** Click the button in the top left corner to open the **New VM** wizard.

**Step 3** Enter the virtual machine details:

a) For the operating system, select **Import existing disk image**.

This method allows you to import a disk image (containing a pre-installed, bootable operating system) to it.

b) Click **Forward** to continue.

**Step 4** Load the disk image:

a) Click **Browse...** to select the image file.
b) Choose *Generic* for the **OS type**.
c) Click **Forward** to continue.

**Step 5** Configure the memory and CPU options:

**Important**
The Firewall Threat Defense Virtual supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.

See the following table for supported performance tiers and values for the --vcpus and --ram parameters for each the Firewall Threat Defense Virtual platform.

***Table 13: Supported vCPU and Memory Parameters for Virtual Machine Manager***

| CPUs | Memory | Firewall Threat Defense Virtual Platform Size |
|---|---|---|
| 4 | 8192 | 4vCPU/8GB (default) |
| 8 | 16384 | 8vCPU/16GB |
| 12 | 24576 | 12vCPU/24GB |

a) Set the **Memory (RAM)** parameter for your Firewall Threat Defense Virtual platform size.
b) Set the corresponding **CPUs** parameter for the Firewall Threat Defense Virtual platform size.
c) Click **Forward** to continue.

**Step 6** Check the **Customize configuration before install** box, specify a **Name**, then click **Finish**.

Doing so opens another wizard that allows you to add, remove, and configure the virtual machine's hardware settings.

**Step 7** Modify the CPU configuration:

From the left panel, select **Processor**, then select **Configuration** > **Copy host CPU configuration**.

This applies the physical host's CPU model and configuration to your VM.

**Step 8** Configure the Virtual Disk:

a) From the left panel, select **Disk 1**.
b) Select **Advanced options**.
c) Set the **Disk bus** to *Virtio*.
d) Set the **Storage format** to *qcow2*.

**Step 9** Configure a serial console:

a) From the left panel, select **Console**.
b) Select **Remove** to remove the default console.
c) Click **Add Hardware** to add a serial device.
d) For **Device Type**, select *TCP net console (tcp)*.
e) For **Mode**, select *Server mode (bind)*.
f) For **Host**, enter **0.0.0.0** for the IP address, then enter a unique **Port** number.
g) Check the **Use Telnet** box.
h) Configure device parameters.

**Step 10** Configure a watchdog device to automatically trigger some action when the KVM guest hangs or crashes:

a) Click **Add Hardware** to add a watchdog device.
b) For **Model**, select *default*.
c) For **Action**, select *Forcefully reset the guest*.

**Step 11** Configure at least 4 virtual network interfaces.

Click **Add Hardware** to add an interface, then choose **macvtap** or specify a shared device name (use a bridge name).

**Note**
The Firewall Threat Defense Virtual on KVM supports a total of 11 interfaces—1 management interface, 1 interface reserved for internal use, and a maximum of 9 network interfaces for data traffic. The interface-to-network assignments must be ordered as follows:

vnic0—Management interface (required)

vnic1—Reserved for internal use (required)

vnic2— Outside interface (required)

vnic3—Inside interface (required)

vnic4-10—Data interfaces (optional)

**Important**
Make sure vnic0 and vnic3 are mapped to the same subnet.

**Step 12** If deploying using a Day 0 configuration file, create a virtual CD-ROM for the ISO:

a) Click **Add Hardware**.
b) Select **Storage**.
c) Click **Select managed or other existing storage** and browse to the location of the ISO file.
d) For **Device type**, select *IDE CDROM*.

**Step 13** After configuring the virtual machine's hardware, click **Apply**.

**Step 14** Click **Begin installation** for virt-manager to create the virtual machine with your specified hardware settings.

**Note**
When launching the Threat Defense Virtual in virt-manager, a graphical (SPICE) console opens by default. On some systems, this console may appear frozen or show only partial output during boot. However, the device continues to boot normally in the background.

To view the full console output, go to:

```
View → Consoles → Console or Serial
```

If a TCP serial console is configured, use telnet to access the console instead — output will not appear in virt-manager.

### What to do next

Your next steps depend on what management mode you chose.

- If you chose **No** for **ManageLocally**, you'll use the Firewall Management Center to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409.

See How to Manage Secure Firewall Threat Defense Virtual Device, on page 1for an overview of how to choose your management option.

# Launch Without the Day 0 Configuration File

Because the Firewall Threat Defense Virtual appliances do not have web interfaces, you must set up a virtual device using the CLI if you deployed without a Day 0 configuration file.

When you first log into a newly deployed device, you must read and accept the EULA. Then, follow the setup prompts to change the administrator password, and configure the device's network settings and firewall mode.

When following the setup prompts, for multiple-choice questions, your options are listed in parentheses, such as (*y/n*). Defaults are listed in square brackets, such as [*y*]. Press Enter to confirm a choice.

**Note** To change any of these settings for a virtual device after you complete the initial setup, you must use the CLI.

### Procedure

**Step 1** Open a console to the Firewall Threat Defense Virtual.

**Step 2** At the **firepower login** prompt, log in with the default credentials of **username** *admin* and the **password** *Admin123*.

**Step 3** When the Firewall Threat Defense Virtual system boots, a setup wizard prompts you for the following information required to configure the system:

- Accept EULA
- New admin password

- IPv4 or IPv6 configuration
- IPv4 or IPv6 DHCP settings
- Management port IPv4 address and subnet mask, or IPv6 address and prefix
- System name
- Default gateway
- DNS setup
- HTTP proxy
- Management mode (local management required)

**Step 4** Review the Setup wizard settings. Defaults or previously entered values appear in brackets. To accept previously entered values, press **Enter**.

**Step 5** Complete the system configuration as prompted.

**Step 6** Verify the setup was successful when the console returns to the # prompt.

**Step 7** Close the CLI.

### What to do next

Your next steps depend on what management mode you chose.

- If you chose **No** for **Enable Local Manager**, you'll use the Firewall Management Center to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409.

See How to Manage Secure Firewall Threat Defense Virtual Device, on page 1 for an overview of how to choose your management option.

# Troubleshooting

This section provides you with some basic troubleshooting steps related to your KVM deployment on your virtual machine.

**Verify whether your virtual machine is running the KVM**

You can use the following methods to verify whether your virtual machine is running the KVM:

- Run the **lsmod** command to list the modules in the Linux Kernel. If the KVM is running, it is indicated by displaying the following output:

  **root@kvm-host:~$ lsmod | grep kvm**

  **kvm_intel 123675 0**

  **kvm 257361 1 kvm_intel**

- If **ls -l /dev/kvm** command does not exist on the target VM then you are probably running **QEMU**, and not taking advantage of the KVM hardware assist features.

**root@kvm-host:~$ ls -l /dev/kvm**

**crw------- 1 root root 10, 232 Mar 23 13:53 /dev/kvm**

- Run the following command to also check whether the host machine supports KVM:

  **root@kvm-host:~$ sudo kvm-ok**

- You can also use KVM acceleration.

### Experiencing boot loops while deploying the Firewall Threat Defense Virtual

You must ensure the following if your virtual machine experiences boot loops:

- Ensure to deploy the VM with at least 8 GB of memory.
- Ensure to deploy the VM with a minimum of 4 interfaces.
- Ensure to deploy the VM with a minimum of 4 vCPUs.
- Verify that the QEMU process is using a server class CPU, for example, SandyBridge, IvyBridge, Haswell, and so forth. Use the following command, **ps -edaf | grep qemu** to inspect the process params.

### Experiencing boot loops while deploying the Firewall Management Center Virtual

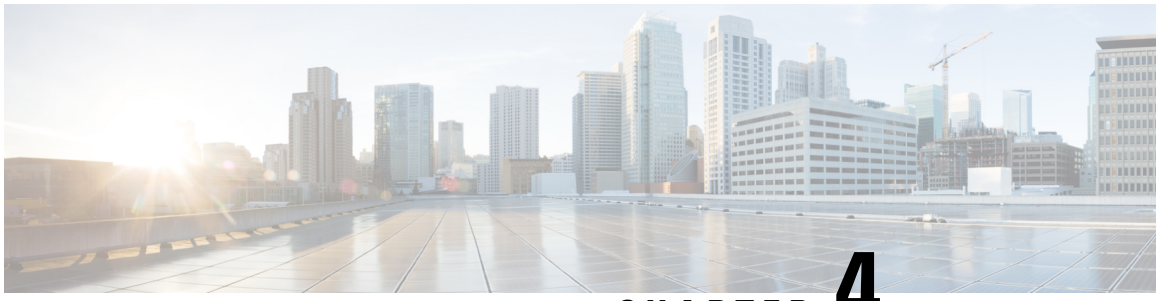
You must ensure the following if your virtual machine experiences boot loops:

- Ensure that you deploy the VM with at least 28 GB of memory.
- Ensure to deploy the VM with a minimum of 4 interfaces.
- Ensure to deploy the VM with a minimum of 4 vCPUs.
- Verify that the QEMU process is using a server class CPU, for example, SandyBridge, IvyBridge, Haswell, and so forth. Use the following command, **ps -edaf | grep qemu** to inspect the process params.

### Post deployment troubleshooting

You can run the following command on the Firewall Threat Defense Virtual to check the issues to capture logs for debugging, **system generate-troubleshoot <space> ALL**

Alternatively, use **system generate-troubleshoot <space>**, followed by a question mark **(?)** or a **Tab** button to view the possible option or command.



CHAPTER 4

# Deploy the Firewall Threat Defense Virtual on AWS

This chapter explains how to deploy the Firewall Threat Defense Virtual from the AWS portal.


- Overview, on page 63
- End-to-End Procedure, on page 64
- How to Manage Secure Firewall Threat Defense Virtual Device, on page 66
- AWS Solution Overview, on page 67
- Prerequisites, on page 67
- Guidelines and Limitations, on page 68
- Configuring AWS Environment, on page 71
- Instance Metadata Service (IMDS) for Firewall Threat Defense Virtual in AWS, on page 77
- Deploy the Firewall Threat Defense Virtual, on page 78
- Firewall Threat Defense Virtual using Image Snapshot, on page 81
- Integrating Amazon GuardDuty Service and Firewall Threat Defense Virtual, on page 84
- Overview, on page 84
- Integrate Amazon GuardDuty with Secure Firewall Threat Defense, on page 90
- Update Existing Solution Deployment Configuration, on page 103


## Overview

AWS is a public cloud environment. The Firewall Threat Defense Virtual runs as a guest in the AWS environment on the following instance types.

***Table 14: Supported VM Sizes***

| VM Size | vCPUs | Memory (RAM) in GB | vNICs | Version |
|---|---|---|---|---|
| c3.xlarge | 4 | 7.5 | 4 | Any |
| c4.xlarge | 4 | 7.5 | 4 | Any |
| c5.xlarge | 4 | 8 | 4 | 6.6 or above |
| c5.2xlarge | 8 | 16 | 4 | 6.6 or above |

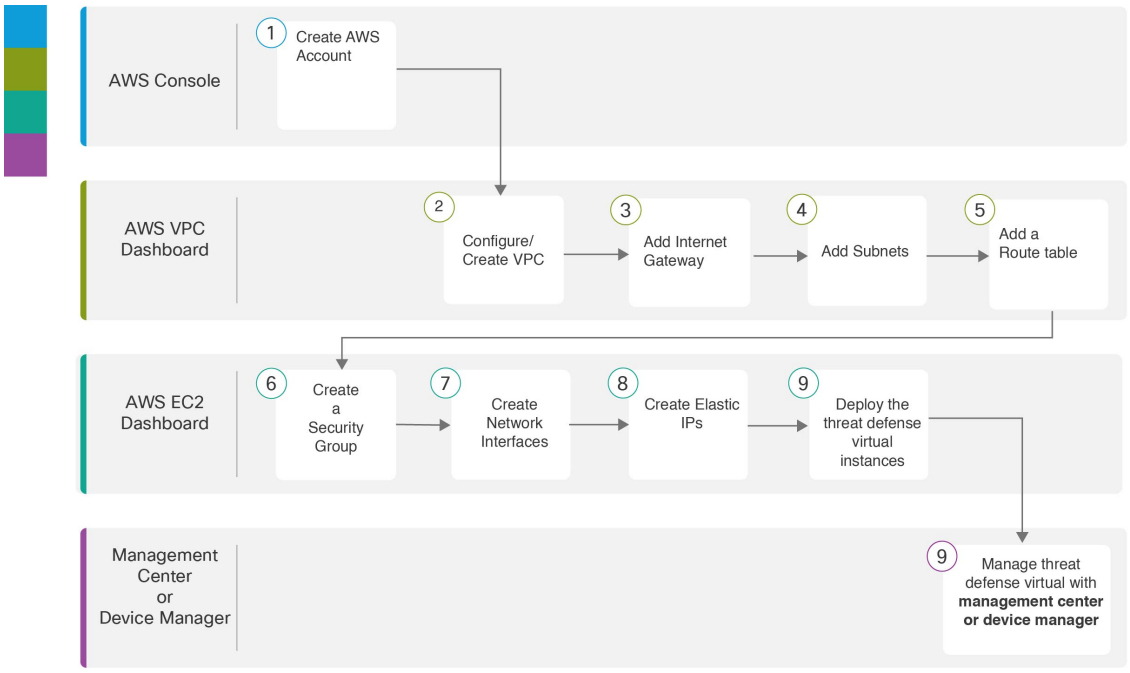
| VM Size | vCPUs | Memory (RAM) in GB | vNICs | Version |
|---|---|---|---|---|
| c5.4xlarge | 16 | 32 | 8 | 6.6 or above |
| c5n.xlarge | 4 | 10.5 | 4 | 7.1 or above |
| c5n.2xlarge | 8 | 21 | 4 | 7.1 or above |
| c5n.4xlarge | 16 | 42 | 8 | 7.1 or above |
| c5a.xlarge | 4 | 8 | 4 | 7.1 or above |
| c5a.2xlarge | 8 | 16 | 4 | 7.1 or above |
| c5a.4xlarge | 16 | 32 | 8 | 7.1 or above |
| c5d.xlarge | 4 | 8 | 4 | 7.1 or above |
| c5d.2xlarge | 8 | 16 | 4 | 7.1 or above |
| c5d.4xlarge | 16 | 32 | 8 | 7.1 or above |
| c5ad.xlarge | 4 | 8 | 4 | 7.1 or above |
| c5ad.2xlarge | 8 | 16 | 4 | 7.1 or above |
| c5ad.4xlarge | 16 | 32 | 8 | 7.1 or above |
| m5n.xlarge | 4 | 16 | 4 | 7.1 or above |
| m5n.2xlarge | 8 | 32 | 4 | 7.1 or above |
| m5n.4xlarge | 16 | 64 | 8 | 7.1 or above |
| m5zn.xlarge | 4 | 16 | 4 | 7.1 or above |
| m5zn.2xlarge | 8 | 32 | 4 | 7.1 or above |

**Note** Firewall Threat Defense Virtual does not support changing the instance type by resizing the instance size. You can deploy a Firewall Threat Defense Virtual with a different instance size only with a fresh deployment.

For information about the NGFWv supported EC2 Instance Type listed on the aws marketplace, see https://aws.amazon.com/marketplace/pp/prodview-p2336sqyya34e#pdp-overview.

# End-to-End Procedure

The following flowchart illustrates the workflow for deploying the Firewall Threat Defense Virtual on Amazon Web Services (AWS).




| | Workspace | Steps |
|---|---|---|
| 1 | AWS Console | www.amazon.com: Create a user account in AWS console. |
| 2 | AWS VPC Dashboard | Configure/Create VPC: Create and configure a VPC that is dedicated to your AWS account. |
| 3 | AWS VPC Dashboard | Add Internet Gateway: Add an Internet gateway to connect your VPC to the Internet. |
| 4 | AWS VPC Dashboard | Add Subnets: Add subnets to your VPC. |
| 5 | AWS VPC Dashboard | Add a Route Table: Attach a route table to the gateway you configured for your VPC. |
| 6 | AWS EC2 Dashboard | Create a Security Group: Create a security group with rules specifying allowed protocols, ports and source IP ranges. |
| 7 | AWS EC2 Dashboard | Create Network Interface: Create network interfaces for the Firewall Threat Defense Virtual using static IP addresses. |
| 8 | AWS EC2 Dashboard | Create Elastic IPs: Elastic IPs are reserved public IPs that are used for remote access to the Firewall Threat Defense Virtual as well as other instances. |

| | Workspace | Steps |
|---|---|---|
| 9 | AWS EC2 Dashboard | Deploy the Firewall Threat Defense Virtual Instance: Deploy the Firewall Threat Defense Virtual from the AWS portal. |
| 10 | Firewall Management Center or Firewall Device Manager | Manage Firewall Threat Defense Virtual:<br>• Managing the Firepower Threat Defense Virtual with the Firepower Management Center<br>• Managing the Firepower Threat Defense Virtual with the Firepower Device Manager |

# How to Manage Secure Firewall Threat Defense Virtual Device

You have two options to manage your Secure Firewall Threat Defense Virtual.

## Secure Firewall Management Center

If you are managing large numbers of devices, or if you want to use the more complex features and configurations that the Firewall Threat Defense allows, use the Firewall Management Center to configure your devices instead of the integrated Firewall Device Manager.

**Important** You cannot use both the Firewall Device Manager and the Firewall Management Center to manage the Firewall Threat Defense device. Once the Firewall Device Manager integrated management is enabled, it won't be possible to use the Firewall Management Center to manage the Firewall Threat Defense device, unless you disable the local management and re-configure the management to use the Firewall Management Center. On the other hand, when you register the Firewall Threat Defense device to the Firewall Management Center, the Firewall Device Manager onboard management service is disabled.

**Caution** Currently, Cisco does not have an option to migrate your Firewall Device Manager configuration to the Firewall Management Center and vice-versa. Take this into consideration when you choose what type of management you configure for the Firewall Threat Defense device.

## Secure Firewall Device Manager

The Firewall Device Manager is a web interface included on most Firewall Threat Defense devices. It lets you configure the basic features of the software that are most commonly used for small networks. It is especially designed for networks that include a single device or just a few, where you do not want to use a high-powered multiple-device manager to control a large network with many devices.

**Note** See the Cisco Secure Firewall Threat Defense Compatibility Guide for list of devices that support the Firewall Device Manager.

# AWS Solution Overview

AWS is a collection of remote computing services offered by Amazon.com, also called web services, that make up a cloud-computing platform. These services operate from 11 geographical regions across the world. In general, you should become familiar with the following AWS services when deploying theSecure Firewall Management Center Virtual (formerly Firepower Management Center Virtual ) and the Firewall Threat Defense Virtual:

- Amazon Elastic Compute Cloud (EC2)—a web service that enables you to rent virtual computers to launch and manage your own applications and service, such as a firewall, in Amazon's data centers.
- Amazon Virtual Private Cloud (VPC)—a web service that enables you to configure an isolated private network that exists within the Amazon public cloud. You run you EC2 instances within a VPC.
- Amazon Simple Storage Service (S3)—a web service that provides you with a data storage infrastructure.

You create an account on AWS, set up the VPC and EC2 components (using either the AWS Wizards or manual configuration), and choose an Amazon Machine Image (AMI) instance. The AMI is a template that contains the software configuration needed to launch your instance.

**Note** The AMI images are not available for download outside of the AWS environment.

# Prerequisites

- An AWS account. You can create one at http://aws.amazon.com/.
- An SSH client (for example, PuTTY on Windows or Terminal on macOS) is required to access the Firewall Threat Defense Virtual console.
- A Cisco Smart Account. You can create one at Cisco Software Central https://software.cisco.com/
- License the Firewall Threat Defense Virtual.

  Secure Firewall Management Center

  - Configure all license entitlements for the security services from the Firewall Management Center.
  - See “Licensing the System” in the Secure Firewall Management Center Configuration Guide for more information about how to manage licenses.

  Secure Firewall Device Manager

  - Configure the performance-tiered license entitlements for the security services from the Secure Firewall Device Manager.

- See Threat Defense Virtual Licensing for more information about how to manage licenses.

- Firewall Threat Defense Virtual interface requirements:
  - Management interfaces (2)— One used to connect the Firewall Threat Defense Virtual to the Firewall Management Center, second reserved for internal use; cannot be used for through traffic.

    You can optionally configure a data interface for the Firewall Management Center management instead of the Management interface. The Management interface is a pre-requisite for data interface management, so you still need to configure it in your initial setup. Note that Firewall Management Center access from a data interface is not supported in High Availability deployments. For more information about configuring a data interface for the Firewall Management Center access, see the **configure network management-data-interface** command in the FTD command reference.
  - Traffic interfaces (2) — Used to connect the Firewall Threat Defense Virtual to inside hosts and to the public network.

- Communication Paths:
  - Public/elastic IPs for access to the Firewall Threat Defense Virtual.

### Supported Software Platforms

The Firewall Threat Defense Virtual Auto Scale solution is applicable to the Firewall Threat Defense Virtual managed by the Firewall Management Center, and is software version agnostic. The Cisco Secure Firewall Threat Defense Compatibility Guide provides Cisco software and hardware compatibility, including operating system and hosting environment requirements.

- The Firewall Management Center Virtual Compatibility Guide table lists the compatibility and virtual hosting environment requirements for the Firewall Management Center Virtual on AWS.
- The Cisco Secure Firewall Threat Defense Virtual Compatibility Guide table lists the compatibility and virtual hosting environment requirements for the Firewall Threat Defense Virtual on AWS.

**Note** For purposes of deploying the AWS Auto Scale solution, the minimum supported version for Firewall Threat Defense Virtual on AWS is Version 6.4. The Firewall Management Center must be running Version 6.6+ at a minimum to use memory-based scaling.

# Guidelines and Limitations

### Supported Features

- Deployment in the Virtual Private Cloud (VPC).
- Enhanced networking (SR-IOV).
- Deployment from Amazon Marketplace.
- Deployment of L3 networks.

- Routed mode (default).
- Passive mode via ERSPAN.
- Clustering (version 7.2 and later). For more information, see Clustering for Threat Defense Virtual in a Public Cloud.
- Health monitoring metrics recorded by Amazon CloudWatch
- Jumbo Frames
- Snapshot (version 7.2 and later)
- IPv6

### Unsupported Features

- Cloning
- Transparent, Inline, and Passive modes
- Transport Layer Security (TLS) Server Identity Discovery is not supported with Geneve single-arm setup on AWS.

### Licensing

- BYOL (Bring Your Own License) using a Cisco Smart License Account is supported.
- PAYG (Pay As You Go) licensing, a usage-based billing model that allows customer to run the Firewall Threat Defense Virtual without having to purchase Cisco Smart Licensing. All licensed features (Malware/Threat/URL Filtering/VPN, etc.) are enabled for a registered PAYG Firewall Threat Defense Virtual device. These licensed features are automatically flagged as active on the registered Firewall Management Center. Licensed features cannot be edited or modified from the Firewall Management Center. (Version 6.5+)



**Note** PAYG licensing is not supported on the Firewall Threat Defense Virtual devices deployed in the Firewall Device Manager mode.

See the "Licenses" chapter in the Firewall Management Center Administration Guide for guidelines when licensing your Firewall Threat Defense Virtual device.

### Performance Tiers for Firewall Threat Defense Virtual Smart Licensing

Starting from Firewall Threat Defense Virtual version 7.0.0 release, the Firewall Threat Defense Virtual supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.

**Note** Any threat defense virtual tier license can be used with any supported threat defense virtual vCPU or memory configuration.

This allows threat defense virtual customers to use a license on a wide variety of VM resources.

This can also increase the number of AWS instance types that are supported. When configuring a threat defense virtual VM, the maximum number of cores (vCPUs) supported is 16, and the maximum memory supported is 32 GB RAM.

*Table 15: Firewall Threat Defense Virtual Licensed Feature Limits Based on Entitlement*

| Performance Tier | Suggested vCPU/Memory Configuration | Rate Limit | RA VPN Session Limit |
|---|---|---|---|
| FTDv5 | 4 core/8 GB | 100 Mbps | 50 |
| FTDv10 | 4 core/8 GB | 1 Gbps | 250 |
| FTDv20 | 4 core/8 GB | 3 Gbps | 250 |
| FTDv30 | 8 core/16 GB | 5 Gbps | 250 |
| FTDv50 | 12 core/24 GB | 10 Gbps | 750 |
| FTDv100 | 16 core/34 GB | 16 Gbps | 10,000 |

### Performance Optimizations

To achieve the best performance out of the Firewall Threat Defense Virtual, you can make adjustments to the both the VM and the host. See Virtualization Tuning and Optimization on AWS for more information.

**Receive Side Scaling**—The Firewall Threat Defense Virtual supports Receive Side Scaling (RSS), which is a technology utilized by network adapters to distribute network receive traffic to multiple processor cores. Supported on Version 7.0 and later. See Multiple RX Queues for Receive Side Scaling (RSS) for more information.

### Firewall Threat Defense Virtual Limitations

- The c5.xlarge is the recommended instance; the c3.xlarge instance has limited availability across AWS regions.
- You must have two management interfaces configured during launch.
- You must have two traffic interfaces and two management interfaces to launch, for a total of four interfaces.

**Note** The Firewall Threat Defense Virtual will not launch without four interfaces.

- When configuring traffic interfaces in AWS, you must disable the “Change Source/Dest. Check” option.

- Any IP address (IPv4 and IPv6) configuration (either from CLI or Firewall Management Center) must match what is created in the AWS console; you should note your configurations during deployment.
- After you register the Firewall Threat Defense Virtual, you must edit the interfaces and enable them on the Firewall Management Center; please note that the IP address must match the AWS configured interfaces.
- Transparent/inline/passive modes are not currently supported.
- To modify interfaces, you need to make changes from the AWS console. On the AWS console, deregister the interfaces from the Firewall Management Center and stop the instance that is using the AWS AMI user interface. Then, detach the interfaces you want to change and attach the new interfaces (note that you need two traffic interfaces and two management interfaces to launch). Now, start the instance and re-register to the Firewall Management Center.

  From the Firewall Management Center, edit the Device interface and modify the IP address (IPv4 and IPv6) and other parameters to match the changes you made through the AWS console.



**Note** IPv6 can be used in Dual Stack (IPv4 + IPv6) mode only.

- You cannot add interfaces after boot.
- If you are observing abnormal behavior such as Snort taking a long time to shut down, or the VM being slow in general or when a certain process is executed, collect logs from the Firewall Threat Defense Virtual and the VM host. Collection of overall CPU usage, memory, I/O usage, and read/write speed logs will help troubleshoot the issues.
- High CPU and I/O usage is observed when Snort is shutting down. If a number of Firewall Threat Defense Virtual instances have been created on a single host with insufficient memory and no dedicated CPU, Snort will take a long time to shut down which will result in the creation of Snort cores.

### Firewall Threat Defense Virtual Limitations for Instance Metadata Data Service (IMDS) Service

- Deployments occur in IMDSv2 Optional mode by default.
- IMDS mode for instance can be changed at any point in time.
- Before switching to IMDSv2 Required mode, ensure that the product version supports it otherwise some services, which depend on IMDS, might fail.
- For older versions(without IMDSv2 support), deployment will be possible only with IMDSv2 Optional mode.
- For newer versions(with IMDSv2 support), deployment is possible in both IMDSv2 Optional and IMDSv2 Required mode. But IMDSv2 Required mode is recommended.

# Configuring AWS Environment

To deploy the Firewall Threat Defense Virtual on AWS you need to configure an Amazon VPC with your deployment-specific requirements and settings. In most situations a setup wizard can guide you through your setup. AWS provides online documentation where you can find useful information about the services ranging

from introductions to advanced features. See https://aws.amazon.com/documentation/gettingstarted/ for more information.

For greater control over your AWS setup, the following sections offer a guide to your VPC and EC2 configurations prior to launching the Firewall Threat Defense Virtual instances:


- Creating the VPC, on page 72
- Adding the Internet Gateway, on page 73
- Adding Subnets, on page 74
- Adding a Route Table, on page 74
- Creating a Security Group, on page 75
- Creating Network Interfaces, on page 76
- Creating Elastic IPs, on page 76


### Before You Begin

- Create your AWS account.
- Confirm that AMIs are available for your Firewall Threat Defense Virtual instances.

# Creating the VPC

A virtual private cloud (VPC) is a virtual network dedicated to your AWS account. It is logically isolated from other virtual networks in the AWS cloud. You can launch your AWS resources, such as the Firewall Management Center Virtual and the Firewall Threat Defense Virtual instances, into your VPC. You can configure your VPC; you can select its IP address range, create subnets, and configure route tables, network gateways, and security settings.

For information on enabling IPv6 CIDR block with your VPC and subnets, see AWS documentation Enable IPv6 in a VPC with a public and private subnet.

### Procedure

**Step 1** Log into http://aws.amazon.com/ and choose your region.

AWS is divided into multiple regions that are isolated from each other. The region is displayed in the upper right corner of your screen. Resources in one region do not appear in another region. Check periodically to make sure you are in the intended region.

**Step 2** Click **Services** > **VPC**.

**Step 3** Click **VPC Dashboard** > **Your VPCs**.

**Step 4** Click **Create VPC**.

**Step 5** Enter the following in the **Create VPC** dialog box:

a) A user-defined **Name tag** to identify the VPC.

b) An **IPv4 CIDR block** of IP addresses. CIDR (Classless Inter-Domain Routing) notation is a compact representation of an IP address and its associated routing prefix. For example, 10.0.0.0/24.

c) An **IPv6 CIDR block** of IP addresses. CIDR (Classless Inter-Domain Routing) notation is a compact representation of an IP address and its associated routing prefix. For example, [::/0].
d) Select the **IPv6 CIDR block** as **Amazon-provided IPv6 CIDR block** to enable IPv6 in Virtual Private Cloud.
e) A **Tenancy** setting of Default to ensure that instances launched in this VPC use the tenancy attribute specified at launch.

**Step 6** Click **Yes, Create** to create your VPC.

### What to do next

**Note** Virtual Networks, Subnets, Interface, etc., cannot be created by using IPv6 alone. The IPv4 is used by default, and IPv6 can be enabled along with it.

Add an Internet gateway to your VPC as described in the next section.

# Adding the Internet Gateway

You can add an Internet gateway to connect your VPC to the Internet. You can route traffic for IP addresses outside your VPC to the Internet gateway.

### Before You Begin

- Create a VPC for your Firewall Threat Defense Virtual instances.

### Procedure

**Step 1** Click **Services > VPC**.
**Step 2** Click **VPC Dashboard > Internet Gateways**, and then click **Create Internet Gateway**.
**Step 3** Enter a user-defined **Name tag** to identify the gateway and click **Yes, Create** to create the gateway.
**Step 4** Select the gateway created in the previous step.
**Step 5** Click **Attach to VPC** and select the VPC you created previously.
**Step 6** Click **Yes, Attach** to attach the gateway to your VPC.

By default, the instances launched on the VPC cannot communicate with the Internet until a gateway is created and attached to the VPC.

### What to do next

Add subnets to your VPC as described in the next section.

# Adding Subnets

You can segment the IP address range of your VPC that the Firewall Threat Defense Virtual instances can be attached to. You can create subnets to group instances according to security and operational needs. For the Firewall Threat Defense Virtual you need to create a subnet for management as well as subnets for traffic.

**Before You Begin**

- Create a VPC for your Firewall Threat Defense Virtual instances.

**Procedure**

**Step 1** Click **Services** > **VPC**.

**Step 2** Click **VPC Dashboard** > **Subnets**, and then click **Create Subnet**.

**Step 3** Enter the following in the **Create Subnet** dialog box:

a) A user-defined **Name tag** to identify the subnet.
b) A **VPC** to use for this subnet.
c) The **Availability Zone** where this subnet will reside. Select **No Preference** to let Amazon select the zone.
d) A **CIDR block** of IP addresses (IPv4 and IPv6). The range of IP addresses in the subnet must be a subset of the range of IP addresses in the VPC. Block sizes must be between a /16 network mask and a /28 network mask. The size of the subnet can equal the size of the VPC.

**Step 4** Click **Yes, Create** to create your subnet.

**Step 5** Repeat for as many subnets required. Create a separate subnet for management traffic and create as many subnets as needed for data traffic.

**What to do next**

Add a route table to your VPC as described in the next section.

# Adding a Route Table

You can attach a route table to the gateway you configured for your VPC. You can also associate multiple subnets with a single route table, but a subnet can be associated with only one route table at a time.

**Procedure**

**Step 1** Click **Services** > **VPC**.

**Step 2** Click **VPC Dashboard** > **Route Tables**, and then click **Create Route Table**.

**Step 3** Enter a user-defined **Name tag** to identify the route table.

**Step 4** Select the **VPC** from the drop-down list that will use this route table.

**Step 5** Click **Yes, Create** to create your route table.

**Step 6** Select the route table that you just created.

**Step 7** Click the **Routes** tab to display the route information in the details pane.

**Step 8** Click **Edit**, then click **Add another route**.

a) In the **Destination** column, enter **0.0.0.0/0 or [::/0]** for all IPv6 traffic.

b) In the **Target** column, select your gateway.

**Step 9** Click **Save**.

### What to do next

Create a security group as described in the next section.

# Creating a Security Group

You can create a security group with rules specifying allowed protocols, ports and source IP ranges. Multiple security groups can be created with different rules which you can assign to each instance.

### Procedure

**Step 1** Click **Services** > **EC2**.

**Step 2** Click **EC2 Dashboard** > **Security Groups**.

**Step 3** Click **Create Security Group**.

**Step 4** Enter the following in the Create **Security Group** dialog box:

a) A user-defined **Security group name** to identify the security group.

b) A **Description** for this security group.

c) The **VPC** associated with this security group.

**Step 5** Configure **Security group rules**:

a) Click the **Inbound** tab, then click **Add Rule**.

**Note**
HTTPS and SSH access is required to manage the Firewall Management Center Virtual from outside AWS. You should specify the Source IP addresses accordingly. Also, if you are configuring both the Firewall Management Center Virtual and Firewall Threat Defense Virtual within the AWS VPC, you should allow the private IP management subnet access.

b) Click the **Outbound** tab, then click **Add Rule** to add a rule for outbound traffic, or leave the defaults of **All traffic** (for **Type**) and **Anywhere** (for **Destination**).

**Step 6** Click **Create** to create your security group.

### What to do next

Create network interfaces as described in the next section.

# Creating Network Interfaces

You can create network interfaces for the Firewall Threat Defense Virtual using static IP addresses (IPv4 and IPv6) or DHCP. Create network interfaces (external and internal) as needed for your particular deployment.

### Procedure

**Step 1** Click **Services > EC2**.

**Step 2** Click **EC2 Dashboard > Network Interfaces**.

**Step 3** Click **Create Network Interface**.

**Step 4** Enter the following in the **Create Network Interface** dialog box:

a) A optional user-defined **Description** for the network interface.

b) Select a **Subnet** from the drop-down list. Make sure to select the subnet of the VPC where you want to create the Firewall Threat Defense Virtual instance.

c) Enter a **Private IP** address. You can use a static IP address (IPv4 and IPv6) or Auto-generate (DHCP).

d) Select one or more **Security groups**. Make sure the security group has all the required ports open.

**Step 5** Click **Create network interface** to create your network interface.

**Step 6** Select the network interface that you just created.

**Step 7** Right-click and select **Change Source/Dest. Check**.

**Step 8** Uncheck the **Enable** checkbox under **Source/destination check** and click **Save**.

### What to do next

Create elastic IP addresses as described in the next section.

# Creating Elastic IPs

When an instance is created, a public IP address is associated with the instance. That public IP address (IPv4 and IPv6) changes automatically when you STOP and START the instance. To resolve this issue, assign a persistent public IP address to the instance using Elastic IP addressing. Elastic IPs are reserved public IPs that are used for remote access to the Firewall Threat Defense Virtual as well as other instances.



**Note** At a minimum, you want to create elastic IP addresses for the Firewall Threat Defense Virtual management interface.

### Procedure

**Step 1** Click **Services > EC2**.

**Step 2** Click **EC2 Dashboard > Elastic IPs**.

**Step 3** Click **Allocate New Address**.

**Step 4** Repeat this step for as many elastic/public IPs that you require.

**Step 5** Click **Yes, Allocate** to create your elastic IP.

**Step 6** Repeat for as many elastic IPs required for your deployment.

### What to do next

Deploy the Firewall Threat Defense Virtual as described in the next section.

# Instance Metadata Service (IMDS) for Firewall Threat Defense Virtual in AWS

Instance Metadata Data Service (IMDS) provides information about the Firewall Threat Defense Virtual instances data, which are deployed on AWS. The information includes details about the virtual instance's network, storage and other data. This metadata can be used to automate configuration decisions (Day0 config) and display instance information such as instance type, region and so on.

IMDS APIs collect metadata of the Firewall Threat Defense Virtual instance from AWS during device bootup and later configure the instance. Currently, Firewall Threat Defense Virtual instances use the IMDSv1 API to fetch and validate the instance's metadata. From version 7.6 and later, the IMDSv2 metadata service, a more secure and robust service is supported.

### Configure IMDS in AWS for Firewall Threat Defense Virtual instance

AWS supports the following two IMDS modes for Firewall Threat Defense Virtual instance:

- **IMDSv2 Optional**: You can deploy a Firewall Threat Defense Virtual instance enabling either IMDSv1 or IMDSv2 or a combination of both IMDSv1 and IMDSv2 API calls.
- **IMDSv2 Required**: You must specifically configure only this mode during the Firewall Threat Defense Virtual instance deployment.



**Note** IMDSv2 Required is the recommended mode, where only IMDSv2 API calls are supported.

You can configure IMDS in AWS for the instances in the following deployments scenarios:

**New Deployments**: You can configure IMDSv2 Required mode when you are newly deploying Firewall Threat Defense Virtual instances. For new deployment, you can use one of the following methods to enable the IMDSv2.

- AWS EC2 console – You can enable the **V2 only (token required)** for a standalone instance deployment in the Advance Details section of the AWS EC2 console.
- CloudFormation template – You can use HttpEndpoint: enabled and HttpTokens: required properties under **MetadataOptions** in the template to enable **V2 only (token required)** - IMDSv2 Required mode. This is applicable for Autoscale and Clustering deployment.

# Deploy the Firewall Threat Defense Virtual

### Before you begin

Cisco recommends the following:

- Configure AWS VPC and EC2 elements as described in Configuring AWS Environment, on page 71.
- Confirm that an AMI is available for the Firewall Threat Defense Virtual instances.

### Procedure

**Step 1** Go to https://aws.amazon.com/marketplace (Amazon Marketplace) and sign in.

**Step 2** After you log in to the Amazon Marketplace, click the link provided for the Firewall Threat Defense Virtual (Cisco Firepower NGFW Virtual (NGFWv) - BYOL).

**Note**
If you were previously in AWS, you may need to sign out and then sign back in for the link to work.

**Step 3** Click **Continue**, then click the **Manual Launch** tab.

**Step 4** Click **Accept Terms**.

**Step 5** Click **Launch with EC2 Console** in your desired region

**Step 6** Choose the **Instance Type** supported by the Firewall Threat Defense Virtual, c4.xlarge recommended.

**Step 7** Click the **Next: Configure Instance Details** button at the bottom of the screen:

- Change the **Network** to match your previously created VPC.
- Change the **Subnet** to match your previously created management subnet. You can specify an IP address or use auto-generate.
- You can enable **Auto-generate** the **Public IP** (IPv4 and IPv6).
- Virtual Networks, Subnets, Interface, etc., cannot be created by using IPv6 alone. The IPv4 is used by default, and IPv6 can be enabled along with it. For more information on IPv6 Migration, see AWS IPv6 Overview and AWS VPC.
- Click the **Add Device** button under Network Interfaces to add the eth1 network interface.
- Change the **Subnet** to match your previously created management subnet that is used for eth0.

**Note**
The Firewall Threat Defense Virtual requires two management interfaces.

**CAUTION:** Use only plain text when entering data in the **Advanced Details** field. If you copy this information from a text editor, make sure you copy only as plain text. If you copy any Unicode data into the **Advanced Details** field, including white space, the instance may be corrupted and you will have to terminate the instance and re-create it.

Sample login configuration to manage the Firewall Threat Defense Virtual using the Firewall Management Center:

```
#Sensor
```

```
{
            "AdminPassword": "<your_password>",
            "Hostname": "<your_hostname>",
            "IPv6Mode": "dhcp",
            "ManageLocally": "No",
            "FmcIp":  "<IP address of FMC>",
            "FmcRegKey":"<registration_passkey>",
            "FmcNatId":"<NAT_ID_if_required>"
}
```

Sample login configuration to manage the Firewall Threat Defense Virtual using the Firewall Device Manager:

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

- Under **Advanced Details**, add the default login information. Modify the example below to match your requirements for device name and password.
- Under **Advanced Details**, enable the IMDSv2 metadata:
  **a.** Choose **Enabled** from the **Metadata accessible** drop-down list.
  **b.** Choose **V2 only (token required)** from the **Metadata version** drop-down list.

  You can also enable the IMDSv2 from the AWS CLI by perform the following:
- Open the AWS CLI console and add the following arguments to enable IMDSv2 Required mode **--metadata-options "HttpEndpoint=enabled,HttpTokens=required”**

  Sample IMDSv2 configuration:

```
            aws ec2 run-instances \,
            --image-id ami-0abcdef1234567890 \
            --instance-type c5x.large \
...
--metadata-options "HttpEndpoint=enabled,HttpTokens=required"
```

**Step 8** Click **Next: Add Storage**.

You can proceed with the default value.

**Step 9** Click **Next: Tag Instance**.

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with **Key** = Name and **Value** = Firewall.

**Step 10** Select **Next: Configure Security Group**.

**Step 11** Click **Select an existing Security Group** and choose the previously configured Security Group, or create a new Security Group; see AWS documentation for more information on creating Security Groups.

**Step 12** Click **Review and Launch**.

**Step 13** Click **Launch**.

**Step 14** Select an existing key pair or create a new key pair.

**Note**
You can select an existing key pair, or create a new key pair. The key pair consists of a public key that AWS stores and a private key file that the user stores. Together they allow you to connect to your instance securely. Be sure to save the key pair to a known location, as it will may be required to connect to the instance.

**Step 15** Click **Launch Instances**.

**Step 16** Click **View Launch** and follow the prompts.

**Step 17** Click **EC2 Dashboard > Network Interfaces.**

**Step 18** Find the traffic interfaces previously created in Configuring AWS Environment, on page 71, then click **Attach**. This will become the **eth2** interface on your Firewall Threat Defense Virtual instance.

**Step 19** Find the traffic interfaces previously created in Configuring AWS Environment, on page 71, then click **Attach**. This will become the **eth3** interface on your Firewall Threat Defense Virtual instance.

**Note**
You must have four interfaces configured or the Firewall Threat Defense Virtual will not complete the boot process.

**Step 20** Click **EC2 Dashboard > Instances**.

**Step 21** Right-click the instance, then select **Instance Settings > Get System Log** to view the status.

**Note**
There will possibly be a warning of a connectivity issue. This is expected, since the eth0 interface will not be active until the EULA is completed.

**Step 22** After 20 minutes, register your Firewall Threat Defense Virtual to the Firewall Management Center.

### What to do next

Your next steps depend on what management mode you chose.

- If you chose **No** for **Enable Local Manager**, you'll use the Firewall Management Center to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409.
- If you chose **Yes** for **Enable Local Manager**, you'll use the integrated Firewall Device Manager to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Device Manager, on page 427.

See How to Manage Secure Firewall Threat Defense Virtual Device, on page 1 for an overview of how to choose your management option.

# Configure IMDSv2 Required Mode for Existing Firewall Threat Defense Virtual Instances

You can configure the IMDSv2 Required mode for the Firewall Threat Defense Virtual instances that are already deployed on the AWS.

### Before you begin

IMDSv2 Required mode is only supported by Firewall Threat Defense Virtual version 7.6 and later. You must ensure that your existing instances version is compatible (upgraded to version 7.6) with IMDSv2 mode before configuring the IMDSv2 mode for your deployment.

### Procedure

**Step 1** Log into http://aws.amazon.com/ and choose your region.

**Step 2** Click **EC2** > **Instances**.

**Step 3** Right-click the instance, then select **Instance Settings** > **Modify instance metadata options**. The **Modify instance metadata options** dialog box is displayed.

**Step 4** Under **Instance metadata service** section, click **Enable**.

**Step 5** Under **IMDSv2** options, click **Required**.

This enables the IMDSv2 Required mode for the selected instance.

**Step 6** Click **Save**.

# Firewall Threat Defense Virtual using Image Snapshot

You can create and deploy the Firewall Threat Defense Virtual using an Amazon Machine Image (AMI) snapshot in the AWS portal. The image snapshot is a replicated Firewall Threat Defense Virtual image instance with no state data.

## Firewall Threat Defense Virtual Snapshot Overview

The process of creating a snapshot image of the Firewall Threat Defense Virtual instance helps to minimize the initial system *init* time by skipping the first boot procedures done for the Firewall Threat Defense Virtual and FSIC. The snapshot image consists of prepopulated database and the Firewall Threat Defense Virtual initial boot process, which enables the image to regenerate unique IDs (UUIDs, Serial number) that is related to the system identity in the Firewall Management Center or any other management center. This process helps in faster boot time of Firewall Threat Defense Virtual, which is essential in auto scale deployment.

**Note** Snapshot image creation is used to minimize the initial system init time on a non-registered Secure Firewall Threat Defense Virtual. It should not be used for the purpose of *backup/restore*.

**Note** Currently, Pay-As-You-Go (PAYG) licensing is not supported for instances deployed using a snapshot image of the Firewall Threat Defense Virtual. PAYG licensing is only available for instances that you deploy directly from the Marketplace. You can use Smart Licensing for such new Firewall Threat Defense Virtual deployments with PAYG licensing.

# Create Firewall Threat Defense Virtual Snapshot AMI

The Firewall Threat Defense Virtual image snapshot creation is a process of replicating an existing Firewall Threat Defense Virtual instance to create a plain Firewall Threat Defense Virtual instance in the AWS portal.

### Before you begin

- You must have deployed the Firewall Threat Defense Virtual version 7.2 or later. For information on deploying the Firewall Threat Defense Virtual, see Deploy the Firewall Threat Defense Virtual on AWS, on page 63.
- You must not register the Firewall Threat Defense Virtual instance you are preparing for image snapshot to any manager such as Firewall Management Center Virtual or Firewall Device Manager.

### Procedure

**Step 1** Go to the AWS console where you have deployed the Firewall Threat Defense Virtual instance.

**Note**
Ensure that the Firewall Threat Defense Virtual instance which you are planning to replicate as image snapshot is not registered to Firewall Management Center or configured to any other local manager or applied with any configuration.

**Step 2** Use the following scripts to run the pre-snapshot process from the expert shell:

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

When you use `prepare_snapshot` command in the script, an intermediate message appears prompting for confirmation to execute the script. Press **Y** to run the script.

Alternatively, you can append `-f` to this command, such as `root@firepower:/ngfw/var/common# prepare_snapshot -f` to skip the user confirmation message and directly execute the script.

This script removes all the line configurations, deployed policies, configured manager, UUIDs associated with the Firewall Threat Defense Virtual instance. After the processing is done, the Firewall Threat Defense Virtual instance is shut down. The Firewall Threat Defense Virtual instance is listed in the **Instances** page in the AWS portal.

**Step 3** Log into http://aws.amazon.com/ and choose your region.

AWS is divided into multiple regions that are isolated from each other. The region is displayed in the upper right corner of your window. Resources in one region do not appear in another region. You should periodically check the region to ensure you are in the intended region.

### What to do next

Deploy the Firewall Threat Defense Virtual Instance using Snapshot AMI. See, Deploy the Firewall Threat Defense Virtual Instance using Snapshot AMI, on page 83

**Note** You can run the CLI commands **show version** and **show snapshot detail** from the Firewall Threat Defense Virtual console to know about the version and details of the Firewall Threat Defense Virtual image snapshot instance you have created.

# Deploy the Firewall Threat Defense Virtual Instance using Snapshot AMI

### Before you begin

Cisco recommends the following:

- Configure AWS VPC and EC2 elements as described in Configuring AWS Environment, on page 71.
- Confirm that an AMI is available for Firewall Threat Defense Virtual instances.

### Procedure

**Step 1** Go to https://aws.amazon.com/marketplace (Amazon Marketplace) and sign in.

**Step 2** Click **EC2 Dashboard** > **Instances**. The Firewall Threat Defense Virtual instance you have deployed to create an image snapshot is displayed in the **Instances** page.

**Note**
For creating an image snapshot, you must always choose the Firewall Threat Defense Virtual instance whose operational status (**Instance Status**) is **Stopped**.

**Step 3** On the **Instances** page, identify and choose the Firewall Threat Defense Virtual instance whose corresponding **Instance Status** is indicated as **Stopped**.

**Step 4** From the **Actions** drop-down menu, point to **Image and templates** and then click **Create Image**.

**Step 5** In the **Create Image** page, provide the name and description for the image snapshot.

**Step 6** Check the **Enable** check box under the **No reboot** section.

**Step 7** Click **Create Image**. The Firewall Threat Defense Virtual image snapshot AMI is created.

**Step 8** Click **Images** > **AMIs**. You can view the newly created image snapshot AMI on this page.

**Step 9** Select the image snapshot AMI.

**Step 10** Click **Launch** to deploy a new Firewall Threat Defense Virtual instance using the image snapshot AMI.

**Step 11** Continue to deploy the Firewall Threat Defense Virtual instance. See Deploy the Firewall Threat Defense Virtual, on page 78 or About the Firewall Threat Defense Virtual Auto Scale Solution on AWS, on page 106.

# Integrating Amazon GuardDuty Service and Firewall Threat Defense Virtual

Amazon GuardDuty is a monitoring service that processes data from various sources such as VPC logs, CloudTrail management event logs, CloudTrail S3 data event logs, DNS logs, and so on to identify potentially unauthorized and malicious activity in the AWS environment.

# Overview

Cisco offers a solution to integrate the Amazon GuardDuty service with Secure Firewall Threat Defense Virtual via the management centers and device managers.

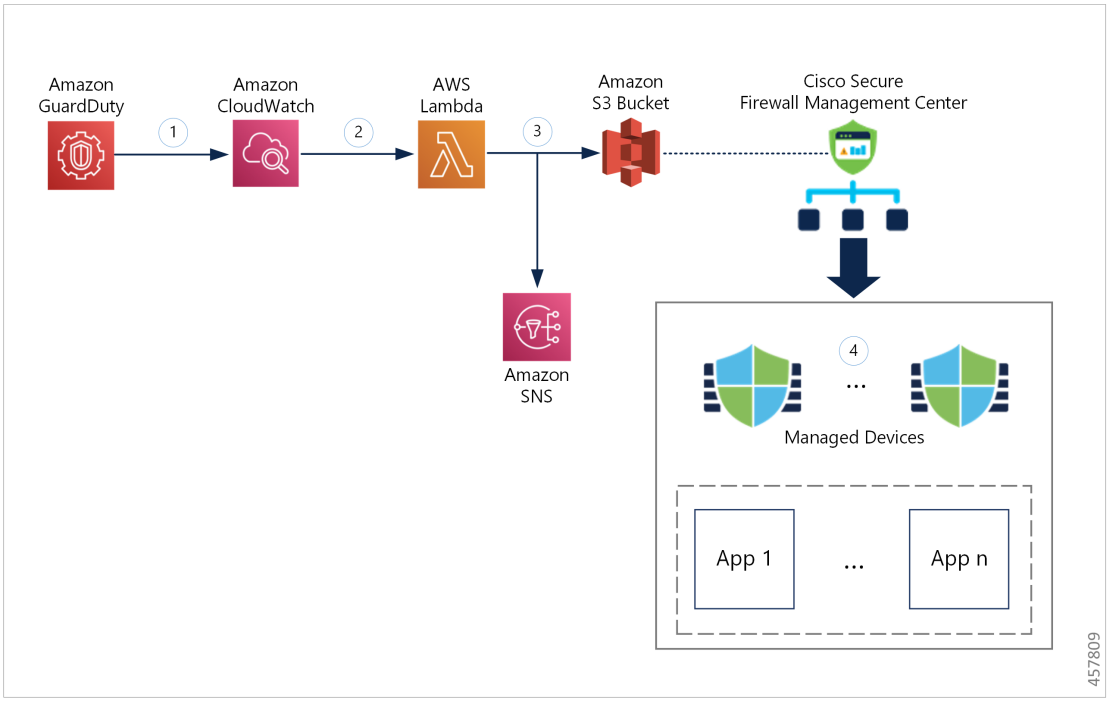
This solution use threat analysis data or results from the Amazon GuardDuty (malicious IPs generating threats, attacks and so on) and feeds that information (malicious IP) to the Secure Firewall Threat Defense Virtual via the managers: Secure Firewall Management Center Virtual and Secure Firewall Device Manager to protect the underlying network and applications against future threats originating from these sources (malicious IP).

## End-to-End Procedure

The following integration solutions with workflow illustrations help you understand the integration of Amazon GuardDuty with Secure Firewall Threat Defense Virtual.

### Integration with Secure Firewall Management Center Virtual using Security Intelligence Network Feed

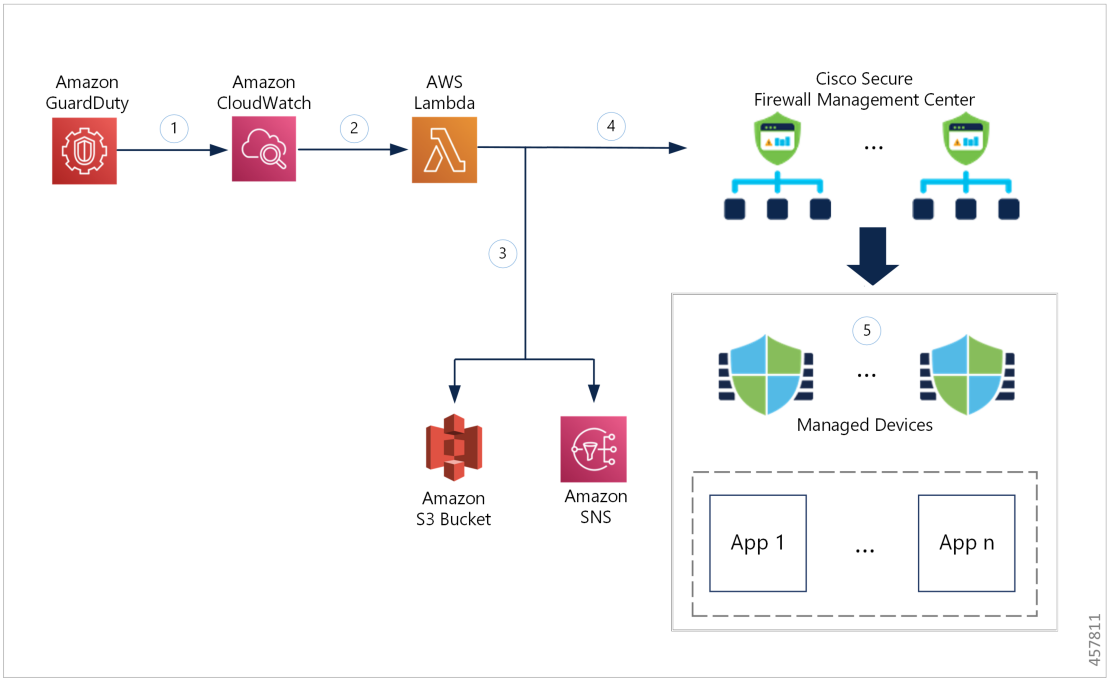
The following workflow diagram shows the Amazon GuardDuty integration solution with Secure Firewall Management Center Virtual using the Security Intelligence network feed URL.




| | |
|---|---|
| 1 | The GuardDuty service sends threat findings to CloudWatch when it detects a malicious activity. |
| 2 | The CloudWatch event activates the AWS Lambda function. |
| 3 | The Lambda function updates the malicious host in the report file in the S3 bucket and sends a notification via SNS. |
| 4 | The Secure Firewall Management Center access control policy directs its target devices to handle the traffic based on the configured actions, for example, block traffic from the malicious hosts reported by GuardDuty.<br>This access policy uses the Security Intelligence network feed with the S3 object URL of the malicious IP address report file provided by the Lambda function. |

## Integration with Secure Firewall Management Center Virtual using Network Object Group

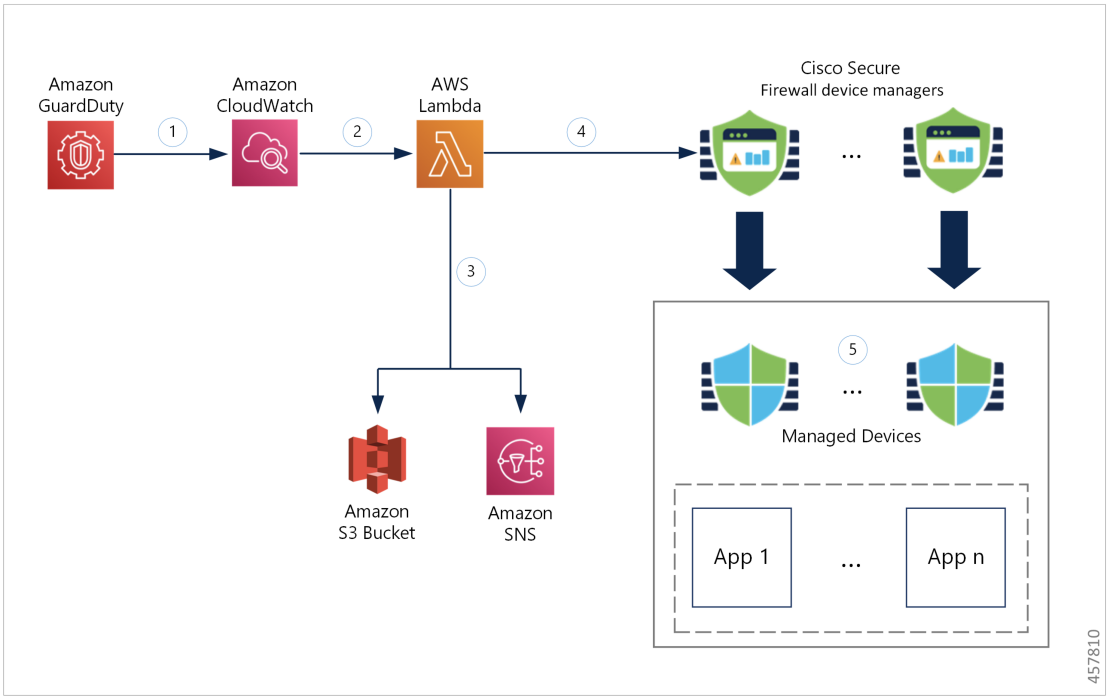
The following workflow diagram shows the Amazon GuardDuty integration solution with Secure Firewall Management Center Virtual using the network object group.




| | |
|---|---|
| 1 | The GuardDuty service sends threat findings to CloudWatch when it detects a malicious activity. |
| 2 | The CloudWatch event activates the AWS Lambda function. |
| 3 | The Lambda function updates the malicious host in the report file in the S3 bucket and sends a notification via SNS. |
| 4 | The Lambda function configures or updates the network object group with the malicious host IP address in Secure Firewall Management Center Virtual. |
| 5 | The Secure Firewall Management Center access control policy directs its target devices to handle the traffic based on the configured actions, for example, block traffic from the malicious hosts reported by GuardDuty.<br>This access control policy uses the network object group with the malicious IP address provided by the Lambda function. |

## Integration with Secure Firewall Device Manager using Network Object Group

The following workflow diagram shows the Amazon GuardDuty integration solution with Secure Firewall Device Manager using the network object group.




| | |
|---|---|
| 1 | The GuardDuty service sends threat findings to CloudWatch when it detects a malicious activity. |
| 2 | The CloudWatch event activates the AWS Lambda function. |
| 3 | The Lambda function updates the malicious host in the report file in the S3 bucket and sends a notification via SNS. |
| 4 | The Lambda function configures or updates the network object group with the malicious host IP address in Secure Firewall Device Manager. |
| 5 | The Secure Firewall Device Manager access control policy directs the managed device to handle the traffic based on the configured actions, for example, block traffic from the malicious hosts reported by GuardDuty.<br>This access control policy uses the network object group with the malicious IP address provided by the Lambda function. |

# Key Components of This Integration

| Component | Description |
|---|---|
| **Amazon GuardDuty** | An Amazon service responsible for generating threat findings for the various AWS resources in a specific region, such as EC2, S3, IAM, and so on. |

| | |
|---|---|
| **Amazon Simple Storage Service (S3)** | An Amazon service used for storing various artifacts associated with the solution:<br>• Lambda function zip file<br>• Lambda layer zip file<br>• Secure Firewall management center and device manager configuration input file(.ini)<br>• Output report file (.txt) containing a list of malicious IP addresses reported by the Lambda function |
| **Amazon CloudWatch** | An Amazon service used for:<br>• Monitoring the GuardDuty service for any reported findings and triggering the Lambda function to process the finding.<br>• Logging the Lambda function-related activities in the CloudWatch log group. |
| **Amazon Simple Notification Service (SNS)** | An Amazon service used to push email notifications. These email notifications contain:<br>• The details of the GuardDuty finding that was successfully processed by the Lambda function.<br>• The details of the updates performed on the Secure Firewall managers by the Lambda function.<br>• Any significant errors encountered by the Lambda function. |
| **AWS Lambda Function** | An AWS serverless compute service that runs your code in response to events and automatically manages the underlying compute resources for you. The Lambda function is triggered by the CloudWatch event rule based on GuardDuty findings. In this integration, the Lambda function is responsible for:<br>• Processing the GuardDuty findings to verify that all the required criteria are met, such as severity, connection direction, presence of malicious IP address, and so on.<br>• (Depending on the configuration) Updating the network object group on the Secure Firewall managers with the malicious IP address.<br>• Updating the malicious IP address in the report file on the S3 bucket.<br>• Notifying the Secure Firewall administrator about various manager updates and any errors. |

| | |
|---|---|
| **CloudFormation Template** | Used to deploy various resources required for the integration in AWS.<br>The CloudFormation template contains the following resources:<br>• **AWS::SNS::Topic**: The SNS Topic for pushing email notifications.<br>• **AWS::Lambda::Function, AWS::Lambda::LayerVersion** : The Lambda function and layer files<br>• **AWS::Events::Rule**: The CloudWatch event rule to trigger the Lambda function based on the GuardDuty findings event.<br>• **AWS::Lambda::Permission**: Permission for the CloudWatch event rule to trigger the Lambda function.<br>• **AWS::IAM::Role, AWS::IAM::Policy**: The IAM role and policy resources to allow various access permissions to the Lambda function for various AWS resources.<br>This template accepts user input parameters to customize the deployment. |

# Supported Software Platforms

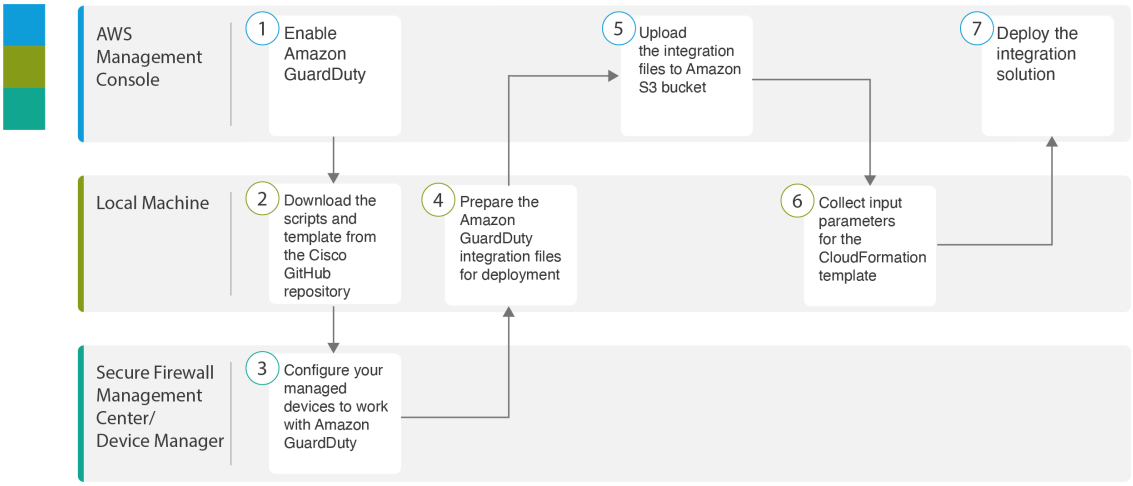
- The GuardDuty integration solution is applicable to Secure Firewall Threat Defense Virtual managed by Secure Firewall Management Center Virtual or Secure Firewall Device Manager.
- The Lambda function can update the network object groups in the management center and device manager deployed on any virtual platform. Ensure that the Lambda function can connect to these managers via public IP addresses.

# Guidelines and Limitations

- The Lambda function is responsible only for updating the network objects groups on the Secure Firewall managers with the malicious IP addresses. Therefore, ensure that you deploy these updates or changes to the managed devices.
- The AWS services used in this integration are region-specific. Therefore, if you want to use the GuardDuty findings from different regions, you must deploy region-specific instances.
- The Lambda function updates the Secure Firewall managers via REST APIs. Therefore, you cannot use any other methods or managers, for example, Cisco Security Cloud Control.
- You can use only password-based login. No other authentication methods are supported.
- If you are using encrypted passwords in the input file, keep in mind that:
  - Only encryption using the symmetric KMS keys is supported.
  - All the passwords must be encrypted using a single KMS key accessible to the Lambda function.

# Integrate Amazon GuardDuty with Secure Firewall Threat Defense

Perform the following tasks to integrate Amazon GuardDuty with Secure Firewall Threat Defense




| | Workspace | Steps |
|---|---|---|
| 1 | AWS Management Console | Enable Amazon GuardDuty Service on AWS, on page 90 |
| 2 | Local Machine | Download the Secure Firewall Threat Defense Virtual and Amazon GuardDuty Integration Solution Repository, on page 91 |
| 3 | Secure Firewall Management Center or Secure Firewall Device Manager | Configure Your Managed Devices to Work with Amazon GuardDuty, on page 91 |
| 4 | Local Machine | Prepare Amazon GuardDuty Resource Files for Deployment, on page 95 |
| 5 | AWS Management Console | Upload Files to Amazon Simple Storage Service, on page 98 |
| 6 | Local Machine | Collect Input Parameters for CloudFormation Template, on page 98 |
| 7 | AWS Management Console | Deploy the Stack, on page 100 |

## Enable Amazon GuardDuty Service on AWS

This section describes how to enable Amazon GuardDuty service on AWS.

### Before you begin

Ensure that all the AWS resources are in the same region.

### Procedure

**Step 1** Go to https://aws.amazon.com/marketplace (Amazon Marketplace) and sign in.

**Step 2** Choose **Services** > **GuardDuty**.

**Step 3** Click **Get Started** in the **GuardDuty** page.

**Step 4** Click **Enable GuardDuty** to enable the Amazon GuardDuty service.

For more information on enabling GuardDuty, see Getting started with GuardDuty in AWS Documentation.

### What to do next

Download the Amazon GuardDuty solution files (templates and scripts) from the Cisco GitHub repository. See Download the Secure Firewall Threat Defense Virtual and Amazon GuardDuty Integration Solution Repository, on page 91.

# Download the Secure Firewall Threat Defense Virtual and Amazon GuardDuty Integration Solution Repository

Download the files required for the Amazon GuardDuty solution. The deployment scripts and templates for your Secure Firewall Threat Defense Virtual version are available from the Cisco GitHub repository at:

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

The following is a list of resources in the Cisco GitHub repository:

| Files | Description |
|---|---|
| READ.MD | ReadMe file |
| configuration/ | Secure Firewall Threat Defense Virtual manager configuration file template. |
| images/ | It contains the Secure Firewall Threat Defense Virtual and Amazon GuardDuty integration solution illustrations. |
| lambda/ | Lambda function Python files. |
| templates/ | CloudFormation template for deployment. |

# Configure Your Managed Devices to Work with Amazon GuardDuty

The Lambda function processes the Amazon GuardDuty findings and identifies the malicious IP address that triggered the CloudWatch Event. The Secure Firewall Threat Defense Virtual receives this threat data via the

Secure Firewall Management Center Virtual and Secure Firewall Device Manager in one of the following methods:

- **Network object group update**—The Lambda function updates the network object group in the managers with the malicious IP address. You can then configure an access control policy that uses this network object group to handle the traffic. This method applies to Secure Firewall Management Center Virtual and Secure Firewall Device Manager.
- **Security Intelligence Network feed**—The Lambda function creates or updates a report file in the Amazon S3 bucket with the malicious IP address. You can set up a Security Intelligence feed using the report file URL and then configure an access control policy that uses this feed to handle the traffic. This method applies only to Secure Firewall Management Center Virtual.

## Configure Security Intelligence Network Feed with the Report File URL

This section describes how to configure Security Intelligence network feed in Secure Firewall Management Center Virtual.

### Before you begin

- Ensure that you have enabled Threat license on Secure Firewall Management Center Virtual. See Threat License.
- Ensure that you have created and noted the report file URL that is available in the Amazon S3 bucket.
- Ensure that the report file in the Amazon S3 bucket is reachable from the Secure Firewall Management Center Virtual.

### Procedure

**Step 1** Log in to Secure Firewall Management Center Virtual.

**Step 2** Create a Security Intelligence network feed using the report file URL of the Amazon S3 bucket. For information about manually creating the Security Intelligence network feed, see Custom Security Intelligence Feeds.

**Step 3** Create or update the access control policy or access control rule with the Security Intelligence network feed URL to handle the traffic. See Manual URL Filtering Options and Create and Edit Access Control Rules.

**Note**
You can create the Security Intelligence network feed and update the URL in the access control policy before or after deployment. If you are creating the output report file in the Amazon S3 bucket, the Security Intelligence network feed can be created before deployment. If you are creating the Security Intelligence network feed after deployment, wait until you receive the email notification of the first finding from Amazon GuardDuty and configure the Security Intelligence network feed using the URL given in that email notification.

**Step 4** Deploy the configuration changes on Secure Firewall Management Center Virtual. See Deploy Configuration Changes.

### What to do next

Prepare the Amazon GuardDuty source files for deployment. See Prepare Amazon GuardDuty Resource Files for Deployment, on page 95.

# Create Network Object Group

In the Secure Firewall Management Center Virtual and Secure Firewall Device Manager, you must configure or create a network object group for the Lambda function to update the malicious IP address detected by the Amazon GuardDuty.

If you do not configure a network object group with the Lambda function, then a network object group with the default name **aws-gd-suspicious-hosts** is created by the Lambda function to update the malicious IP address.

## Create Network Object Groups Secure Firewall Management Center Virtual

This section describes how to create network object group in Secure Firewall Management Center Virtual.

### Procedure

**Step 1** Log in to Secure Firewall Management Center Virtual.

**Step 2** Create a network object group with a dummy IP address. See Network Objects.

**Step 3** Create or update the access control policy or access control rule to handle the traffic using the network object group. See Managing Access Control Policies and Create and Edit Access Control Rules.

**Tip**
You can also create or update the access control policy or access control rule after verifying that the Lambda function is updating the network object group with the malicious IP address.

**Step 4** Deploy your configuration changes to the managed devices. See Deploy Configuration Changes.

### What to do next

Prepare the Amazon GuardDuty source files for deployment. See Prepare Amazon GuardDuty Resource Files for Deployment, on page 95.

## Create Network Object Group in Secure Firewall Device Manager

This section describes how to create network object group in Secure Firewall Device Manager.

### Procedure

**Step 1** Log in to Secure Firewall Device Manager.

**Step 2** Create a network object group with a dummy IP address. See Configuring Network Objects and Groups.

**Step 3** Create or update the access control policy or access control rule to handle the traffic using the network object group. See Configuring the Access Control Policy and Configuring Access Control Rules.

**Tip**
You can also create or update the access control policy or access control rule after verifying that the Lambda function is updating the network object group with the malicious IP address.

**Step 4** Deploy your configuration changes to the managed devices. See Deploying Your Changes.

### What to do next

Prepare the Amazon GuardDuty source files for deployment. See Prepare Amazon GuardDuty Resource Files for Deployment, on page 95.

## Create User Account in Secure Firewall Management Center Virtual for Lambda Function Access

The Lambda function requires a user account with admin privileges to update the network object group in the management center and device manager. Therefore, you must create an exclusive user account with admin privileges in the management center and device manager. The user account creation is necessary only when you are using the network object group update method.

For more information to create a new user account, see:

- Managing FDM and FTD User Access
- User Accounts for FMC

## (Optional) Encrypt Passwords

If required, you can provide encrypted passwords in the input configuration file. You can also provide passwords in plain text format.

Encrypt all the passwords using a single KMS key that is accessible to the Lambda function. Use the **aws kms encrypt --key-id** <*KMS-ARN*> **--plaintext** <*password*> command to generate the encrypted password. You have to install and configure AWS CLI to run this command.

**Note** Ensure that passwords are encrypted using symmetric KMS keys.

For more information on AWS CLI, see AWS Command Line Interface. For more information on master keys and encryption, see the AWS document Creating keys and the AWS CLI Command Reference about password encryption and KMS.

Example:

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

The value of *CiphertextBlob* key should be used as a password.

# Prepare Amazon GuardDuty Resource Files for Deployment

The Amazon GuardDuty solution deployment resource files are available on the Cisco GitHub repository.

Before deploying the Amazon GuardDuty solution on AWS, you must prepare the following files:

- Secure Firewall Threat Defense Virtual manager configuration input file
- Lambda function zip file
- Lambda layer zip file

## Prepare Configuration Input File

In the configuration template, you must define the details of the management center or device manager you are integrating with the Amazon GuardDuty solution. We recommend that you update the configuration file only when you are planning to implement the network object group update method for Amazon GuardDuty integration with the management center and device manager.

### Before you begin

- Ensure to authenticate and verify the user account of the device manager before you provide the user account details in the configuration file.
- If you are configuring multiple management centers or device managers in the configuration file, ensure that the parameters for each management center or device manager is entered only once in the configuration file and there are no duplicate entries.
- You must have noted the IP address and name of the management center and device manager.
- You must have created a user account having admin privileges for the Lambda function to access and update these network object group in the management center and device manager.

### Procedure

**Step 1** Log in to the local machine where you have downloaded the Amazon GuardDuty resource files.

**Step 2** Browse to the **ngfwv-template** > **configuration** folder.

**Step 3** Open the `ngfwv-manager-config-input.ini` file a text editor tool.

In this file, you must enter the details of the management center or device manager where you are planning to integrate and deploy the Amazon GuardDuty solution.

**Step 4** Enter the following details of the management center or device manager corresponding to each parameter:

| Parameters | Description |
|---|---|
| [ngfwv-1] | Section name: Unique identifier of the management center or device manager. |
| public-ip | IP address of the management center or device manager. |

| Parameters | Description |
|---|---|
| device-type | The type of managed device where you are deploying the Amazon GuardDuty solution through management center or device manager. Allowed values are FMC or FDM. |
| user name | Username to log in to management center or device manager. |
| password | Password to log in to management center or device manager. The password can be a plain text format or encrypted string that is created using KMS. |
| object-group-name | Name of the network object groups name that is updated with malicious host IP by the Lambda function. If you are entering multiple network object groups name, then ensure that they are comma separated values. |

**Step 5** Save and close the `ngfwv-manager-config-input.ini` file.

### What to do next

Create the Lambda function archive file. See Prepare Lambda Function Archive File, on page 96.

## Prepare Lambda Function Archive File

This section describes how to archive the Lambda function files in a Linux environment.

**Note** The archiving process may differ depending on the operating system of the local machine where you are archiving the files.

### Procedure

**Step 1** Open the CLI console on the local machine where you have downloaded the Amazon GuardDuty resources.

**Step 2** Navigate to the `/lambda` folder and archive the files.

The following is a sample transcript from a Linux host.

```
$ 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%)
$
```

The zip file `ngfwv-gd-lambda.zip` is created.

**Step 3** Exit and close the CLI console.

### What to do next

Create the Lambda layer zip file using the zip file `ngfwv-gd-lambda.zip`. See Prepare Lambda Layer File, on page 97

## Prepare Lambda Layer File

This section describes how to archive the Lambda layer file in a Linux environment.

**Note** The archiving process may differ depending on the operating system of the local machine where you are archiving the file.

### Procedure

**Step 1** Open the CLI console on the local machine where you have downloaded the Amazon GuardDuty resources.

**Step 2** Perform the following actions in your CLI console.

The following is a sample transcript from a Linux host such as Ubuntu 22.04 with Python 3.9 installed.

```
$ 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
```

The zip file `ngfwv-gd-lambda-layer.zip` is created.

Note that you must install Python 3.9 and its dependencies for creating the Lambda layer.

The following is the sample transcript for installing Python 3.9 on a Linux host such as Ubuntu 22.04.

```
$ 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
```

**Step 3** Exit and close the CLI console.

### What to do next

In Amazon S3 bucket, you must upload the Secure Firewall Threat Defense Virtual configuration file, the Lambda function zip file, and the Lambda layer zip file. See Upload Files to Amazon Simple Storage Service, on page 98

# Upload Files to Amazon Simple Storage Service

After you prepare all the Amazon GuardDuty solution artifacts, you must upload the files to an Amazon Simple Storage Service (S3) bucket folder in the AWS portal.

### Procedure

**Step 1** Go to https://aws.amazon.com/marketplace (Amazon Marketplace) and sign in.

**Step 2** Open the Amazon S3 console.

**Step 3** Create an Amazon S3 Bucket for uploading the Amazon GuardDuty artifacts. See Creating Amazon S3.

**Step 4** Upload the following Amazon GuardDuty artifacts to the Amazon S3 bucket.

- Secure Firewall Threat Defense Virtual configuration file: `ngfwv-config-input.ini`

  **Note**
  This file is not required to be uploaded when you are using Security Intelligence Network Feed method for deploying the Amazon GuardDuty solution in management centers.

- Lambda layer zip file: `ngfwv-gd-lambda-layer.zip`
- Lambda function zip file: `ngfwv-gd-lambda.zip`

### What to do next

Prepare the CloudFormation template that is used for deploying Amazon GuardDuty resources. See Collect Input Parameters for CloudFormation Template, on page 98.

# Collect Input Parameters for CloudFormation Template

Cisco provides the CloudFormation template that is used to deploy resources required by Amazon GuardDuty solution in AWS. Collect values for the following template parameters before deployment.

### Procedure

**Template Parameters**

| Parameter | Description | Example |
|---|---|---|
| Deployment name* | The name you enter in this parameter is used as prefix for all the resources created by the Cloud Formation template. | **cisco-ngfwv-gd** |
| Minimum severity level of GD finding* | Minimum severity level Amazon GuardDuty findings to be considered for processing must be in the range between **1.0** to **8.9**. Any finding reported with a lesser severity than the minimum range is ignored.<br>Severity classification is as follows:<br>• Low: 1.0 to 3.9<br>Medium: 4.0 to 6.9<br>High: 7.0 to 8.9. | **4.0** |
| Administrator email ID* | Administrator email address to receive notifications on Secure Firewall Threat Defense Virtual manager about the updates done by Lambda function in the management center or device manager. | **abc@xyz.com** |
| S3 Bucket name* | Name of the Amazon S3 bucket containing Amazon GuardDuty artifacts files (Lambda function zip, Lambda layer zip, and Secure Firewall Threat Defense Virtual configuration manager files). | For example: **ngfwv-gd-bucket** |
| S3 Bucket folder/path prefix | Amazon S3 bucket path or folder name where the configuration files are stored. If there is no folder, leave this field empty. | For example: "" or "**cisco/ngfwv-gd/**" |
| Lambda layer zip file name* | Lambda layer zip file name. | For example: `ngfwv-gd-lambda-layer.zip` |
| Lambda function zip file name* | Lambda function zip file name. | For example:`ngfwv-gd-lambda.zip` |
| Secure Firewall management center and device manager manager configuration file name | The ***.ini** file containing the manager configuration details of the Cisco Firewall Threat Defense Virtual. (Public IP, username, password, device-type, network object group names and so on.)<br>**Note** | For example: `ngfwv-config-input.ini` |

| Parameter | Description | Example |
|---|---|---|
| | This file is required only when you are using the Network Object Group update method for Amazon GuardDuty integration.<br>If you are using the Security Intelligence Feed method, then you can skip providing this input. | |
| ARN of KMS key used for password encryption | ARN of an existing KMS (AWS KMS key used for password encryption). You can leave this parameter empty in case plain text passwords are provided in the Secure Firewall Threat Defense Virtual configuration input file. If you specify, all the passwords mentioned in the Secure Firewall Threat Defense Virtual configuration input file must be encrypted. The passwords must be encrypted using only the specified ARN. Generating encrypted passwords: aws kms encrypt --key-id <KMS ARN> --plaintext <password> | For example: arn:aws:kms:<region>:<aws account id>:key/<key id> |
| Enable/Disable debug logs* | Enable or Disable the Lambda function debug logs in the CloudWatch. | For example: **enable** or **disable** |

*: Mandatory field

### What to do next

Deploy the stack using the CloudFormation template. See Deploy the Stack, on page 100

# Deploy the Stack

After all the pre-requisite processes for Amazon GuardDuty solution deployment are completed, create the AWS CloudFormation stack. Use the template file in the target directory: `templates/cisco-ngfwv-gd-integration.yaml`, and provide the parameters collected in Collect Input Parameters for CloudFormation Template.

### Procedure

**Step 1** Log in to AWS console.

**Step 2** Go to Services > CloudFormation > Stacks > Create stack (with new resources) > Prepare template (The template is provided in the folder) > Specify template > Template source (Upload the template file from the target directory: `templates/cisco-ngfwv-gd-integration.yaml`) > Create Stack

For more information on deploying a stack on AWS, see AWS Documentation.

### What to do next

Validate your deployment. See Validate Your Deployment, on page 101.

Also, subscribe to receive an email notifications on threat detection updates reported by Amazon GuardDuty. See Subscribe to the Email Notifications, on page 101.

# Subscribe to the Email Notifications

In the CloudFormation template, an email ID is configured to receive notification about GuardDuty finding updates done by the Lambda function. After deploying the CloudFormation template on AWS, an email notification is sent to this email ID via Amazon Simple Notification Service (SNS) service requesting you to subcribe for notification updates.

### Procedure

**Step 1** Open the email notification.

**Step 2** Click the **Subscription** link available in the email notification.

### What to do next

Validate your deployment. See Validate Your Deployment, on page 101.

# Validate Your Deployment

In AWS, you have options to verify the Amazon GuardDuty solution as described in this section. You can follow these deployment validation instructions after the CloudFormation deployment is complete.

### Before you begin

Ensure that you have installed and configured AWS Command Line Interface (CLI) to run commands for validating the deployment. For information on AWS CLI documentation, see AWS Command Line Interface.

### Procedure

**Step 1** Log in to AWS Management console.

**Step 2** Go to **Services > GuardDuty > Settings > About GuardDuty > Detector ID** to note the detector ID.

This detector ID is required for generating sample Amazon GuardDuty findings.

**Step 3** Open the AWS CLI console to generate the sample Amazon GuardDuty finding by running the following commands:

```
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
```

**Step 4** Check for the sample finding in the findings list on Amazon GuardDuty console.

The sample findings contains the prefix **[sample]**. You can check the sample finding details viewing the attributes such as connection direction, remote IP address and so on.

**Step 5** Wait for the Lambda function to run.

After the Lambda function is triggered, verify the following:

- An email notification with the details regarding Amazon GuardDuty finding received and Secure Firewall Threat Defense Virtual manager updates done by the Lambda function.
- Verify whether the report file is generated in the Amazon S3 bucket. It contains the malicious IP address reported by the sample Amazon GuardDuty finding. You can identify the report file name in the format: `<deployment-name>-report.txt`.
- For the Network Object Group update method - Verify that the network object groups are updated on the configured managers (Secure Firewall Management Center Virtual or Secure Firewall Device Manager) with the malicious IP address updated from the sample finding.
- For Security Intelligence Feed method - Verify whether the report file URL is already updated in the management center configuration. You can view the last updated timestamp of the report file URL in the following path of management center.
  - **Objects > Object Management > Security Intelligence > Network Lists and Feeds > select the configured feed**
  - Alternatively, you can manually update the feeds and then check for the **Last Updated** timestamps. You can select and update the feed in the following path:

    **Objects > Object Management > Security Intelligence > Network Lists and Feeds > Update Feeds**

**Step 6** Go to **AWS Console > Services > CloudWatch > Logs > Log groups >** *select the log group* to verify the Lambda logs in the CloudWatch console. You can identify the CloudWatch log group name in the format: `<deployment-name>-lambda`.

**Step 7** After validating the deployment, we recommend that you can clean the data generated by the sample finding as follows:

a) Go to **AWS Console > Services > GuardDuty > Findings > Select the finding > Actions > Archive** to view the sample finding data.

b) Delete the malicious IP addresses added in the network object group to clear cached data from the Secure Firewall Management Center Virtual.

c) Clean up the report file in Amazon S3 bucket. You may update the file by removing the malicious IP addresses reported by the sample finding.

# Update Existing Solution Deployment Configuration

We recommend that you do not update the S3 bucket or the S3 bucket folder and path prefix values after deployment. However, if there is a requirement to update the configuration for a solution that has been deployed, use the **`Update Stack`** option on the CloudFormation page in the AWS console.

You can update any of the parameters given below.

| Parameter | Description |
|---|---|
| Secure Firewall Threat Defense Virtual manager configuration file name | Add or update the configuration file in Amazon S3 bucket. You are allowed to update the file with same name as previous one. If the configuration file name is modified, then you can update this parameter by using **Update stack** option in the AWS console. |
| Minimum severity level of GD finding* | Use the **Update stack** option in AWS console to update the parameter value. |
| Administrator email ID* | Update the email ID parameter value using the **Update Stack** option in AWS console. You can also add or update email subscriptions via SNS service console. |
| S3 Bucket name* | Update the zip file in the Amazon S3 bucket with a new name and then update the parameter by using the **Update Stack** option in AWS console. |
| Lambda layer zip file name* | Update the Lambda layer zip file name in the Amazon S3 bucket with a new name and then update this parameter value by using the **Update stack** option in AWS console. |
| Lambda function zip file name* | Update the Lambda function zip file in the Amazon S3 bucket with a new name and then update this parameter value by using the **Update stack** option in AWS console. |
| ARN of KMS key used for password encryption | Use the **Update stack** option in AWS console to update the parameter value. |
| Enable/Disable debug logs* | Use the **Update stack** option in AWS console to update the parameter value. |

### Procedure

**Step 1** Go to the AWS management console.

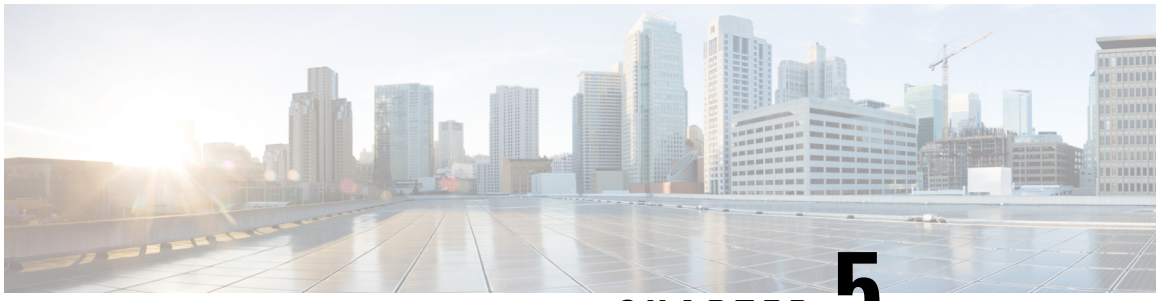
**Step 2** If required, create the new bucket and folder.

**Step 3** Ensure that the artifacts given below are copied from the old bucket to the new bucket.

- Secure Firewall Threat Defense Virtual configuration file: `ngfwv-config-input.ini`

- Lambda layer zip file: `ngfwv-gd-lambda-layer.zip`
- Lambda function zip file: `ngfwv-gd-lambda.zip`
- Output report file: `<deployment-name>-report.txt`

**Step 4** To update the parameter values, go to **Services > CloudFormation > Stacks > > Update (Update Stack) > Prepare template > Use current template > Next >** <*update parameters*>**> Update Stack**.



CHAPTER 5

# Deploy the Firewall Threat Defense Virtual Auto Scale Solution on AWS

This document explains how to deploy the Firewall Threat Defense Virtual auto scale solution on AWS.


- About the Firewall Threat Defense Virtual Auto Scale Solution on AWS, on page 106
- Auto Scale Solution with NLB, on page 107
- End-to-End Process for Deploying Auto Scale Solution with NLB, on page 108
- Auto Scale Solution with Gateway Load Balancer, on page 109
- End-to-End Process for Deploying Auto Scale Solution with GWLB, on page 110
- Guidelines and Limitations for the Firewall Threat Defense Virtual and AWS, on page 111
- Components Required to Set Up the Auto Scale Solution with GWLB or NLB, on page 112
- CloudFormation Templates on GitHub, on page 115
- Download the Required Files and CFTs from GitHub to your Local Host, on page 130
- Auto Scale Solution with NLB - Customize and Deploy the NLB Infrastructure Template on the Amazon CloudFormation Console, on page 131
- Auto Scale Solution with GWLB - Customize and Deploy the GWLB Infrastructure Template on the Amazon CloudFormation Console, on page 131
- Configure Network Infrastructure in the Management Center, on page 132
- Update the configuration.json and configuration-schema.json Files, on page 138
- Configure Infrastructure Components using AWS CLI, on page 139
- Create Target Folder, on page 140
- Upload Files to Amazon S3 Bucket, on page 141
- Auto Scale Solution with NLB - Deploy the Auto Scale Solution with NLB, on page 141
- Auto Scale Solution with GWLB - Deploy the Auto Scale Solution with GWLB, on page 141
- Configure Routing for VPC, on page 143
- Edit the Auto Scale Group, on page 143
- Validate Deployment, on page 144
- Maintenance Tasks, on page 145
- Configure IMDSv2 Required Mode for Existing Autoscale Group Instances, on page 148
- Troubleshooting, on page 149

# About the Firewall Threat Defense Virtual Auto Scale Solution on AWS

The Firewall Threat Defense Virtual instances deployed in public cloud environments such as AWS support applications that experience occasional spikes and dips in network traffic. A spike in traffic could lead to a scenario in which the number of deployed threat defense virtual instances is not enough to inspect the network traffic. A dip in traffic could lead to idle threat defense virtual instances leading to unnecessary operational costs.

The auto scale solution helps organisations to automatically scale up the number of threat defense virtual instances if there is a spike in traffic and also scale down the number of instances during a lull in traffic. This leads to efficient handling of network resources and reduces operational costs.

The Firewall Threat Defense Virtual auto scale in AWS is a completely serverless implementation (no helper VMs involved in the automation of this feature) that adds auto scaling capability to Firewall Threat Defense Virtual instances in the AWS environment.

Starting from version 6.4, Network Load Balancer (NLB)-based auto scale solution is supported on Firewall Threat Defense Virtual managed by management center. Starting from version 7.2, Gateway Load Balancer (GWLB)-based auto scale solution is also supported.

Cisco provides CloudFormation templates and scripts for deploying an auto-scaling group of Firewall Threat Defense Virtual firewalls using several AWS services, including Lambda, auto scaling groups, Elastic Load Balancing (ELB), Amazon S3 Buckets, SNS, and CloudWatch.

The Firewall Threat Defense Virtual auto scale solution is a CloudFormation template-based deployment that provides:

- Completely automated Firewall Threat Defense Virtual instance registration and de-registration with the Firewall Management Center.
- NAT policy, access control policy, and routes automatically applied to the scaled-out Firewall Threat Defense Virtual instances.
- Support for load balancers and multi-availability zones.
- Works only with the Firewall Management Center; the Firewall Device Manager is not supported.

### Enhancements to Auto Scale (Version 6.7)

- Custom metric publisher—A new lambda function polls the Firewall Management Center every second minute for memory consumption of all the Firewall Threat Defense Virtual instances in the auto scale group, then publishes the value to the CloudWatch metric.
- A new scaling policy based on memory consumption is available.
- Firewall Threat Defense Virtual private IP connectivity for SSH and Secure Tunnel to the Firewall Management Center.
- Firewall Management Center configuration validation.
- Support for opening more listening ports on the ELB.

- Modified to single stack deployment. All lambda functions and AWS resources are deployed from a single stack for a streamlined deployment.

# Auto Scale Solution with NLB

As the AWS Load Balancer allows only inbound-initiated connections, only externally generated traffic is allowed to pass inside via the Cisco threat defense virtual firewall.

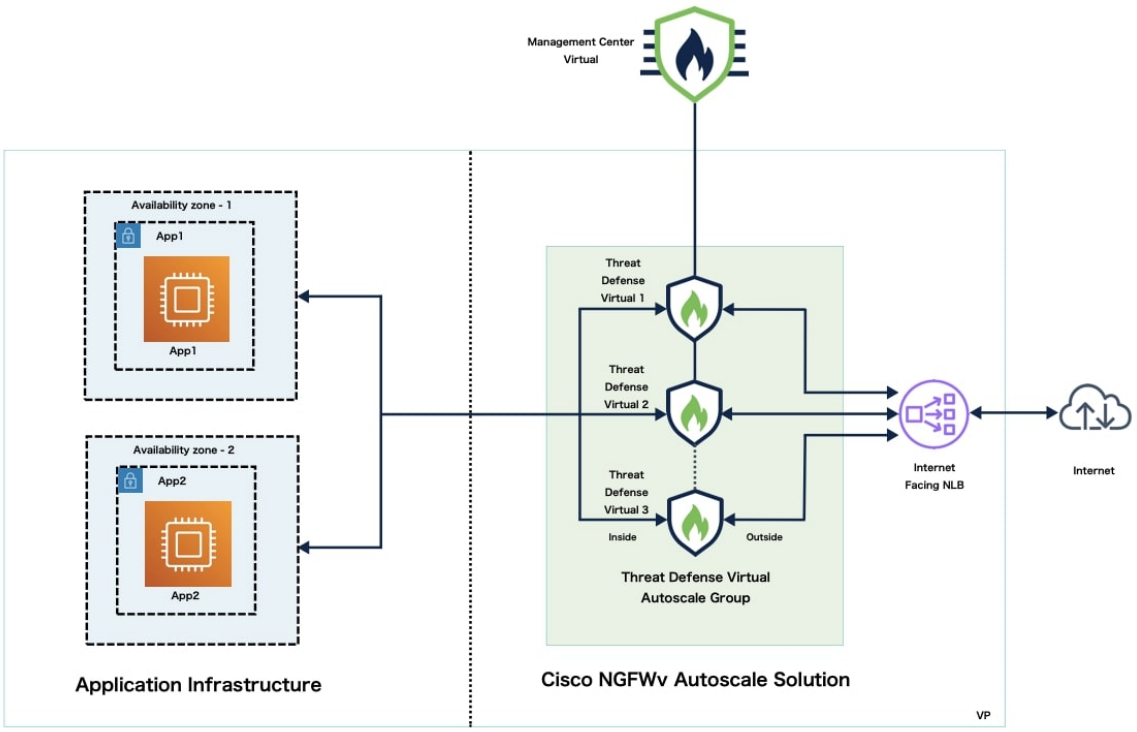
The internet-facing load balancer can be a Network Load Balancer or an Application Load Balancer. All the AWS requirements and conditions hold true for either case. As indicated in the sample topology given below, the right side of the dotted line is deployed via the threat defense virtual templates. The left side is user-defined.

**Note** Application-initiated outbound traffic will not go through the threat defense virtual.

Port-based bifurcation for traffic is possible. This can be achieved via NAT rules; see Create a Host object, Add a Device Group, Configure and Deploy NAT Policy, Create a Basic Access Control Policy, on page 138, Create a Basic Access Control Policy in Management Center. For example, traffic on Internet-facing LB DNS, Port: 80 can be routed to Application-1; Port: 88 traffic can be routed to Application-2.

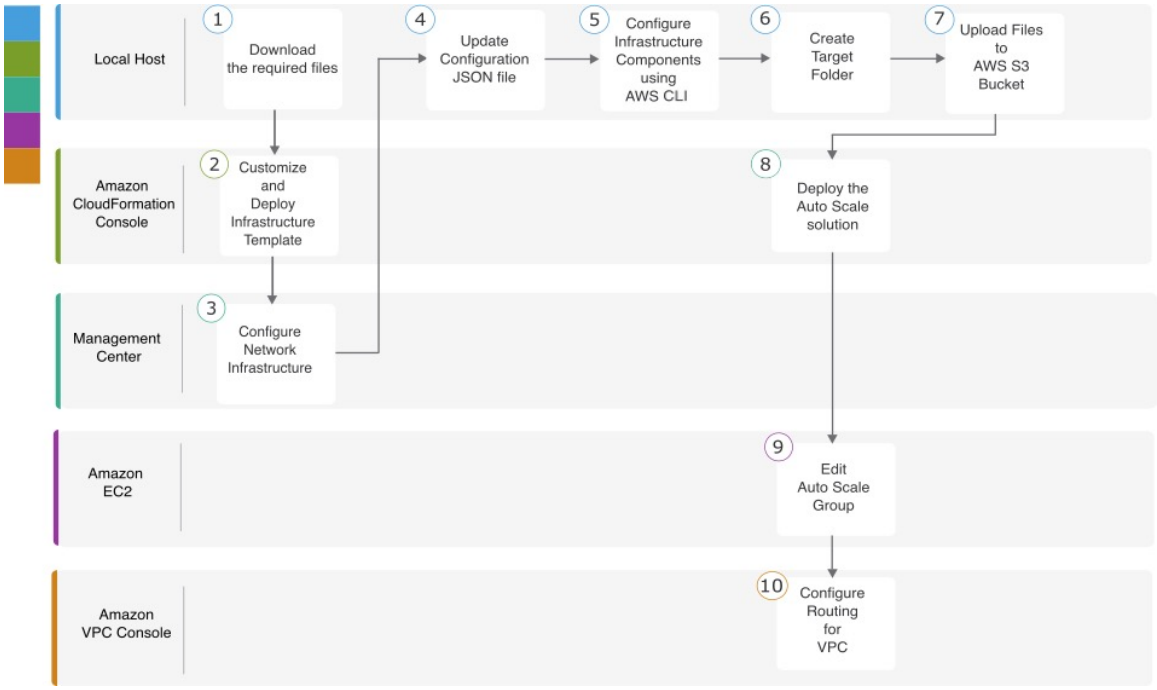
## Sample Topology

***Figure 3: Firewall Threat Defense Virtual Auto Scale Solution with NLB***



# End-to-End Process for Deploying Auto Scale Solution with NLB

The following flowchart illustrates the workflow for deploying Firewall Threat Defense Virtual auto scale solution with NLB on Amazon Web Services (AWS).




| | Workspace | Steps |
|---|---|---|
| 1 | Local Host | Download the required files from GitHub |
| 2 | Amazon CloudFormation Console | Auto Scale Solution with NLB - Customize and Deploy the NLB Infrastructure Template on the Amazon CloudFormation Console, on page 131 |
| 3 | Management Center | Configure Network Infrastructure in the Management Center, on page 132 |
| 4 | Local Host | Update the configuration.json and configuration-schema.json Files, on page 138 |
| 5 | Local Host | Configure Infrastructure Components using AWS CLI, on page 139 |
| 6 | Local Host | Create Target Folder, on page 140 |
| 7 | Local Host | Upload Files to Amazon S3 Bucket, on page 141 |

| | Workspace | Steps |
|---|---|---|
| 8 | Amazon CloudFormation Console | Auto Scale Solution with NLB - Deploy the Auto Scale Solution with NLB, on page 141 |
| 9 | Amazon EC2 Console | Edit the Auto Scale Group, on page 143 |
| 10 | Amazon VPC Console | Configure Routing for VPC, on page 143 |

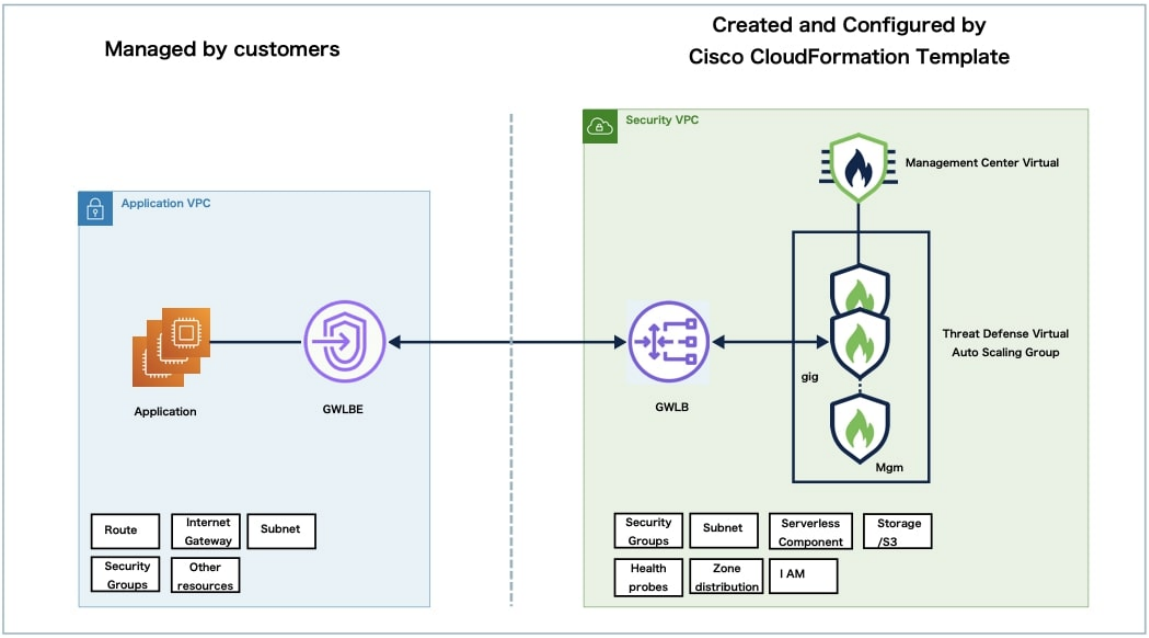
# Auto Scale Solution with Gateway Load Balancer

The AWS Gateway Load Balancer (GWLB) allows both inbound and outbound connections. Therefore, both internally and externally generated traffic is allowed to pass through the Firewall Threat Defense Virtual firewall.

The GWLB endpoint sends traffic to the GWLB, and then to the threat defense virtual for inspection. All of the AWS requirements and conditions hold true for either case. As indicated in the Use Case diagram, the right side of the dotted line is threat defense virtual GWLB Autoscale solution deployed via the threat defense virtual templates. The left side is completely user-defined.

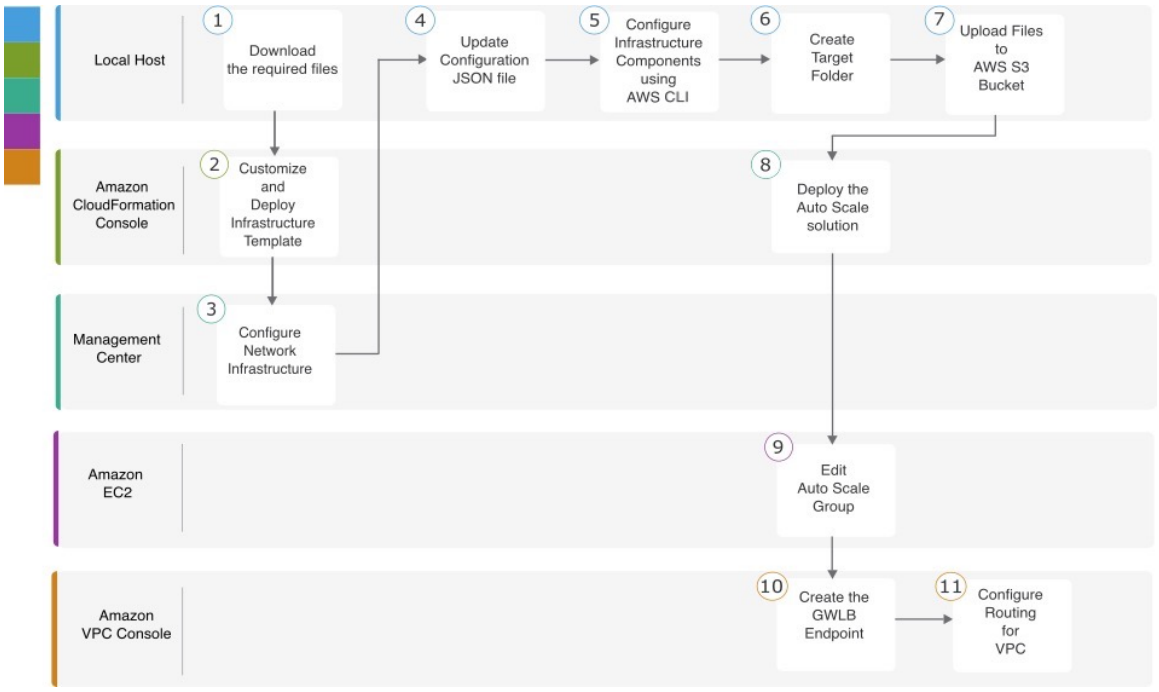
## Sample Topology

***Figure 4: Firewall Threat Defense Virtual Auto Scale Solution with GWLB***



# End-to-End Process for Deploying Auto Scale Solution with GWLB

The following flowchart illustrates the workflow for deploying Firewall Threat Defense Virtual auto scale solution with GWLB on Amazon Web Services (AWS).




| | Workspace | Steps |
|---|---|---|
| 1 | Local Host | Download the required files from GitHub |
| 2 | Amazon CloudFormation Console | Auto Scale Solution with GWLB - Customize and Deploy the GWLB Infrastructure Template on the Amazon CloudFormation Console, on page 131 |
| 3 | Management Center | Configure Network Infrastructure in the Management Center, on page 132 |
| 4 | Local Host | Update the configuration.json and configuration-schema.json Files, on page 138 |
| 5 | Local Host | Configure Infrastructure Components using AWS CLI, on page 139 |
| 6 | Local Host | Create Target Folder, on page 140 |
| 7 | Local Host | Upload Files to Amazon S3 Bucket, on page 141 |

| | Workspace | Steps |
|---|---|---|
| 8 | Amazon CloudFormation Console | Auto Scale Solution with GWLB - Deploy the Auto Scale Solution with GWLB, on page 141 |
| 9 | Amazon EC2 Console | Edit the Auto Scale Group, on page 143 |
| 10 | Amazon VPC Console | Auto Scale with GWLB solution - Create the GWLB Endpoint, on page 142 |
| 11 | Amazon VPC Console | Configure Routing for VPC, on page 143 |

# Guidelines and Limitations for the Firewall Threat Defense Virtual and AWS

### Licensing

- BYOL (Bring Your Own License) using a Cisco Smart License Account is supported.
- PAYG (Pay As You Go) licensing, a usage-based billing model that allows customer to run Firewall Threat Defense Virtual without having to purchase Cisco Smart Licensing. All licensed features (Malware/Threat/URL Filtering/VPN, etc.) are enabled for a registered PAYG device. Licensed features cannot be edited or modified from the Firewall Management Center. (Version 6.5+)

**Note** PAYG licensing is not supported with Firewall Device Manager.

See Cisco Secure Firewall Management Center Administration Guide for licensing guidelines.

### Performance Tiers for Firewall Threat Defense Virtual Smart Licensing

In Version 7.0+, Firewall Threat Defense Virtual supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.

*Table 16: Firewall Threat Defense Virtual Licensed Feature Limits Based on Entitlement*

| Performance Tier | Device Specifications (Core/RAM) | Rate Limit | RA VPN Session Limit |
|---|---|---|---|
| FTDv5 | 4 core/8 GB | 100 Mbps | 50 |
| FTDv10 | 4 core/8 GB | 1 Gbps | 250 |
| FTDv20 | 4 core/8 GB | 3 Gbps | 250 |
| FTDv30 | 8 core/16 GB | 5 Gbps | 250 |
| FTDv50 | 12 core/24 GB | 10 Gbps | 750 |

| Performance Tier | Device Specifications (Core/RAM) | Rate Limit | RA VPN Session Limit |
|---|---|---|---|
| FTDv100 | 16 core/34 GB | 16 Gbps | 10,000 |

### Best Practices

- Ensure that you have configured the required components in the Firewall Management Center. See Configure Network Infrastructure on the Management Center.
- Ensure that you enter the required values for the parameters in the CloudFormation templates. See CloudFormation Templates on GitHub.

### Prerequisites

- An AWS account. You can create one at http://aws.amazon.com/.
- An SSH client (for example, PuTTY on Windows or Terminal on macOS) is required to access the device console.
- A Cisco Smart Account. You can create one at Cisco Software Central https://software.cisco.com/.
- A GitHub account to download the configuration files and templates.
- Firewall Threat Defense Virtual interface requirements:
  - Management interfaces (2) — One used to connect the device to the Firewall Management Center, second used for diagnostics; cannot be used for through traffic.
  - You can optionally configure a data interface for Firewall Management Center management instead of the Management interface. The Management interface is a pre-requisite for data interface management, so you still need to configure it in your initial setup. For more information about configuring a data interface for Firewall Management Center access, see **configure network management-data-interface** in the Cisco Secure Firewall Threat Defense Command Reference.
  - Traffic interfaces (2) — Used to connect the device to inside hosts and to the public network.
- Communication Paths — Public/elastic IP addresess for access to the device

# Components Required to Set Up the Auto Scale Solution with GWLB or NLB

The following components make up the auto scale solution.

### CloudFormation Template

The CloudFormation template is used to deploy resources required to set up the auto scale solution in AWS. The template consists of:

- Auto Scale Group, Load Balancer, Security Groups, and other miscellaneous components.
- The template takes user input to customize the deployment.



**Note** The template has limitations in validating user input, hence it is the user's responsibility to validate input during deployment.

### Lambda Functions

The auto scale solution is a set of Lambda functions developed in Python, which gets triggered from Lifecycle hooks, SNS, CloudWatch event/alarm events. The basic functionality includes:

- Add/Remove Diag, Gig0/0, and Gig 0/1 interfaces to instance.
- Register Gig0/1 interface to Load Balancer's Target Groups.
- Register a new Firewall Threat Defense Virtual with the Firewall Management Center.
- Configure and deploy a new Firewall Threat Defense Virtual via Firewall Management Center.
- Unregister (remove) a scaled-in Firewall Threat Defense Virtual from the Firewall Management Center.
- Publish the memory metric from the Firewall Management Center.

Lambda Functions are delivered to customer in the form of a Python package.

### Lifecycle Hooks

- Lifecycle hooks are used to get lifecycle change notification about an instance.
- In the case of instance launch, a Lifecycle hook is used to trigger a Lambda function which can add interfaces to the Firewall Threat Defense Virtual instance, and register outside interface IPs to target groups.
- In the case of instance termination, a Lifecycle hook is used to trigger a Lambda function to deregister the Firewall Threat Defense Virtual instance from the target group.

### Simple Notification Service (SNS)

- Simple Notification Service (SNS) from AWS is used to generate events.
- Due to the limitation that there is no suitable orchestrator for Serverless Lambda functions in AWS, the solution uses SNS as a kind of function chaining to orchestrate Lambda functions based on events.

### VPC

You should create the VPC as required for your application requirements. It is expected that the VPC have an internet gateway with at least one subnet attached with a route to the internet. Refer to the appropriate sections for the requirements for Security Groups, Subnets, etc.

### Security Groups

All connections are allowed in the provided Auto Scale Group template. You only need the following connections for the auto scale solution to work.

| Port | Usage | Subnet |
|---|---|---|
| 8305 | Management Center to Threat Defense Virtual Secured tunnel connection | Management subnets |
| Health Probe port (default: 8080) | Internet-facing Load Balancer health probes | Outside, Inside Subnets |
| Application ports | Application data traffic | Outside, Inside Subnets |

#### Security Groups or ACLs for the Management Center Instance

These are needed to allow HTTPS connections between lambda functions and the management center. As lambda functions are to be kept in lambda subnets having a NAT gateway as the default route, the management center should be allowed to have inbound HTTPS connections from the NAT gateway IP address.

### Subnets

Subnets can be created as needed for the requirements of the application. The threat defense virtual requires 3 subnets for operation.

**Note** If multiple availability zone support is needed, then subnets are needed for each zone as subnets are zonal properties within the AWS Cloud.

#### Outside Subnet

The Outside subnet should have a default route with '0.0.0.0/0' to the Internet gateway. This will contain the Outside interface of the threat defense virtual, and also the Internet-facing NLB will be in this subnet.

#### Inside Subnet

This can be similar to the application subnets, with or without NAT/Internet gateway. Please note that for the threat defense virtual health probes, it should be possible to reach the AWS Metadata Server (169.254.169.254) via port 80.

**Note** In this Auto Scale solution, Load Balancer health probes are redirected to the AWS Metadata Server via inside/Gig0/0 interface. However, you can change this with your own application serving the health probe connections sent to the threat defense virtual from the Load Balancer. In that case, you need to replace the AWS Metadata Server object with the application IP address to provide the health probe response.

#### Management Subnet

This subnet includes the threat defense virtual management interface. If you are using the management center on this subnet, then assigning an elastic IP address (EIP) to the threat defense virtual is optional. The diagnostic interface is also on this subnet.

#### Lambda Subnets

The AWS Lambda function requires two subnets having the NAT gateway as the default gateway. This makes the Lambda function private to the VPC. Lambda subnets do not need to be as wide as other subnets.

#### Application Subnets

There is no restriction imposed on this subnet from the auto scale solution, but if the application needs outbound connections outside the VPC, the respective routes should be configured on the subnet. This is because outbound-initiated traffic does not pass through load balancers.For more information, see AWS Elastic Load Balancing User Guide.

### Serverless Components

#### S3 Bucket

Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance. You can place all the required files in the S3 bucket.

When templates are deployed, Lambda functions get created referencing zip files in the S3 bucket. Hence, the S3 bucket should be accessible to the user account.

# CloudFormation Templates on GitHub

There are two sets of templates are provided for the supported auto scale solutions-one set for setting up the Auto Scale solution using an Network Load Balancer (NLB) and another set for setting up the Auto Scale solution using a Gateway Load Balancer (GWLB).

# Auto Scale Solution with Network Load Balancer

The following templates are available on GitHub:

- infrastructure.yaml
- deploy_ngfw_autoscale.yaml

***Table 17: List of Template Parameters***

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| PodNumber | String Allowed Pattern: '^\d{1,3}$' | This is the pod number. This will be suffixed to the Auto Scale Group name (threat defense virtual-Group-Name). For example, if this value is '1', then the group name will be *threat defense virtual-Group-Name-1*.<br>It should be at least 1 numerical digit but not more than 3 digits. Default: 1. |
| AutoscaleGrpNamePrefix | String | This is the Auto Scale Group Name Prefix. The pod number will be added as a suffix.<br>Maximum: 18 characters<br>Example: Cisco-threat defense virtual-1. |
| NotifyEmailID | String | Auto Scale events will be sent to this email address. You need to accept a subscription email request.<br>Example: admin@company.com. |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| VpcId | String | The VPC ID in which the device needs to be deployed. This should be configured as per AWS requirements.<br>Type: AWS::EC2::VPC::Id<br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| LambdaSubnets | List | The subnets where Lambda functions will be deployed.<br>Type: List<AWS::EC2::Subnet::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| LambdaSG | List | The Security Groups for Lambda functions.<br>Type: List<AWS::EC2::SecurityGroup::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| S3BktName | String | The S3 bucket name for files. This should be configured in your account as per AWS requirements.<br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| LoadBalancerType | String | The type of Internet-facing Load Balancer, either “application” or “network”.<br>Example: application |
| LoadBalancerSG | String | The Security Groups for the Load Balancer. In the case of a network load balancer, it won't be used. But you should provide a Security Group ID.<br>Type: List<AWS::EC2::SecurityGroup::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| LoadBalancerPort | Integer | The Load Balancer port. This port will be opened on LB with either HTTP/HTTPS or TCP/TLS as the protocol, based on the chosen Load Balancer type.<br>Make sure the port is a valid TCP port, it will be used to create the Load Balancer listener.<br>Default: 80 |
| SSLcertificate | String | The ARN for the SSL certificate for secured port connections. If not specified, a port opened on the Load Balancer will be TCP/HTTP. If specified, a port opened on the Load Balancer will be TLS/HTTPS. |
| TgHealthPort | Integer | This port is used by the Target group for health probes. Health probes arriving at this port on the threat defense virtual will be routed to the AWS Metadata server and should not be used for traffic. It should be a valid TCP port.<br>If you want your application itself to reply to health probes, then accordingly NAT rules can be changed for the threat defense virtual. In such a case, if the application does not respond, the threat defense virtual will be marked as unhealthy and deleted due to the Unhealthy instance threshold alarm.<br>Example: 8080 |
| AssignPublicIP | Boolean | If selected as "true" then a public IP will be assigned. In case of a BYOL-type threat defense virtual, this is required to connect to https://tools.cisco.com.<br>Example: TRUE |
| InstanceType | String | The Amazon Machine Image (AMI) supports different instance types, which determine the size of the instance and the required amount of memory.<br>Only AMI instance types that support the threat defense virtual should be used.<br>Example: c4.2xlarge |
| LicenseType | String | The threat defense virtual license type, either BYOL or PAYG. Make sure the related AMI ID is of the same licensing type.<br>Example: BYOL |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| AmiId | String | The threat defense virtual AMI ID (a valid Cisco threat defense virtual AMI ID).<br>Type: AWS::EC2::Image::Id<br>Please choose the correct AMI ID as per the region and desired version of the image. The Auto Scale feature supports version 6.4+, BYOL/PAYG images. In either case you should have accepted a License in the AWS marketplace.<br>In the case of BYOL, please update 'licenseCaps' key in Configuration JSON with features such as 'BASE', 'MALWARE', 'THREAT', 'URLFilter' etc. |
| NoOfAZs | Integer | The number of availability zones that the threat defense virtual should span across, between 1 and 3. In the case of an ALB deployment, the minimum value is 2, as required by AWS.<br>Example: 2 |
| ListOfAzs | Comma separated string | A comma-separated list of zones in order.<br>**Note**<br>The order in which these are listed matters. Subnet lists should be given in the same order.<br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value.<br>Example: us-east-1a, us-east-1b, us-east-1c |
| MgmtInterfaceSG | String | The Security Group for the threat defense virtual Management interface.<br>Type: List<AWS::EC2::SecurityGroup::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| InsideInterfaceSG | String | The Security Group for the threat defense virtual inside interface.<br>Type: AWS::EC2::SecurityGroup::Id<br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| OutsideInterfaceSG | String | The Security Group for the threat defense virtual outside interface.<br>Type: AWS::EC2::SecurityGroup::Id<br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value.<br>Example: sg-0c190a824b22d52bb |
| MgmtSubnetId | Comma separated list | A comma-separated list of management subnet-ids. The list should be in the same order as the corresponding availability zones.<br>Type: List<AWS::EC2::SecurityGroup::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| InsideSubnetId | Comma separated list | A comma-separated list of inside/Gig0/0 subnet-ids. The list should be in the same order as the corresponding availability zones.<br>Type: List<AWS::EC2::SecurityGroup::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| OutsideSubnetId | Comma separated list | A comma-separated list of outside/Gig0/1 subnet-ids. The list should be in the same order as the corresponding availability zones.<br>Type: List<AWS::EC2::SecurityGroup::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| KmsArn | String | The ARN of an existing KMS (AWS KMS key to encrypt at rest). If specified, the management center and threat defense virtual passwords should be encrypted. The password encryption should be done using only the specified ARN.<br>Generating Encrypted Password Example: " aws kms encrypt --key-id <KMS ARN> --plaintext <password> ". Please used such generated passwords as shown.<br>Example: arn:aws:kms:us-east-1:[AWS Account]:key/7d586a25-5875-43b1-bb68-a452e2f6468e |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| ngfwPassword | String | All the threat defense virtual instances come up with a default password, which is entered in the *Userdata* field of the Launch Template (Autoscale Group).<br>This input will change the password to new provided password once the threat defense virtual is accessible.<br>Please use a plain text password if KMS ARN is not used. If KMS ARN is used, then an encrypted password should be used.<br>Example: Cisco123789! or AQIAgcQFAGtz/hvaxMtJvY/x/rfHnI3lPpSXU |
| fmcServer | Numeric string | The IP address of managing the management center, which is reachable to both Lambda functions and the threat defense virtual management interface.<br>Example: 10.10.17.21 |
| fmcOperationsUsername | String | The Network-Admin or higher privileged user created in managing the management center. See the information about creating users and roles in the Cisco Secure Firewall Management Center Device Configuration Guide.<br>Example: apiuser-1 |
| fmcOperationsPassword | String | Please use a plain text password if KMS ARN is not mentioned. If mentioned, then an encrypted password should be used.<br>Example: Cisco123@ or AQICAHgcQAtz/hvaxMtJvY/x/mKI3clFPpSXUHQRnCAajB |
| fmcDeviceGrpName | String | The management center device group name.<br>Example: AWS-Cisco-NGFW-VMs-1 |
| fmcPerformanceLicenseTier | String | The performance tier license used while registering the Firewall Threat Defense Virtual device on the Firewall Management Center Virtual.<br>Allowed values: FTDv/FTDv5/FTDv10/FTDv20/FTDv30/FTDv50/FTDv100 |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| fmcPublishMetrics | Boolean | If set to "TRUE", then a Lambda function will be created which runs once in every 2 minutes to fetch the memory consumption of registered threat defense virtual sensors in the provided device group.<br>Allowed values: TRUE, FALSE<br>Example: TRUE |
| fmcMetricsUsername | String | The unique management center user name for metric publication to AWS CloudWatch. See the information about creating users and roles in the Cisco Secure Firewall Management Center Device Configuration Guide.<br>If the "fmcPublishMetrics' is set to "FALSE" then there is no need to provide this input.<br>Example: publisher-1 |
| fmcMetricsPassword | String | The management center password for metric publication to AWS CloudWatch. Please use a plain text password if KMS ARN is not mentioned. If mentioned, then an encrypted password should be used.<br>If the "fmcPublishMetrics' is set to "FALSE" then there is no need to provide this input.<br>Example: Cisco123789! |
| CpuThresholds | Comma separated integers | The lower CPU threshold and the upper CPU threshold. The minimum value is 0 and maximum value is 99.<br>Defaults: 10, 70<br>Please note that the lower threshold should be less than the upper threshold.<br>Example: 30,70 |
| MemoryThresholds | Comma separated integers | The lower MEM threshold and the upper MEM threshold. The minimum value is 0 and maximum value is 99.<br>Defaults: 40, 70<br>Please note that the lower threshold should be less than the upper threshold. If the "fmcPublishMetrics" parameter is "FALSE" then this has no effect.<br>Example: 40,50 |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| Instance Metadata Service Version | Boolean | The Instance Metadata Data Service (IMDS) version you want enable for Firewall Threat Defense Virtual instances.<br>• V1 and V2 (token optional) : To enables either IMDSv1 or IMDSv2 or a combination of both IMDSv1 and IMDSv2 API calls.<br>• V2 only (token required) : To enable only IMDSv2 mode.<br>**Note**<br>Firewall Threat Defense Virtual version 7.6 and later supports only IMDSv2 service.<br>If you are enabling IMDSv2 service for versions earlier than 7.6, then you must select combination of both IMDSv1 and IMDSv2 **V1 and V2 (token optional)** parameter.<br>**Note**<br>If you are using a custom template (that is not provided by Cisco), then note that you must include the **HttpEndpoint: enabled** and **HttpTokens: required** properties under **MetadataOptions** in your template to enable IMDSv2 Required mode. |

# Auto Scale Solution with Gateway Load Balancer

**The following templates are available on GitHub:**

- infrastructure_gwlb.yaml
- deploy_ngfw_autoscale_with_gwlb.yaml

*Table 18: List of Template Parameters*

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| DeploymentType | String | Deployment type that helps to process the traffic from Firewall Threat Defense Virtual to GWLB or the internet.<br>• Single-arm: This deployment type enables the Firewall Threat Defense Virtual to return the traffic to AWS GWLB (U-turn) after inspection. By default, the proxy type is set to single-arm when not specified.<br>• Dual-arm: This deployment type enables the Firewall Threat Defense Virtual to perform NAT and forward outbound traffic from the outside interface to the internet through the NAT gateway. |
| PodNumber | String<br>Allowed Pattern: '^\d{1,3}$' | This is the pod number. This will be suffixed to the Auto Scale Group name (Firewall Threat Defense Virtual-Group-Name). For example, if this value is '1', then the group name will be *Firewall Threat Defense Virtual-Group-Name-1*.<br>It should be at least 1 numerical digit but not more than 3 digits. Default: 1 |
| AutoscaleGrpNamePrefix | String | This is the Auto Scale Group Name Prefix. The pod number will be added as a suffix.<br>Maximum: 18 characters<br>Example: Cisco-Firewall Threat Defense Virtual-1 |
| NotifyEmailID | String | Auto Scale events will be sent to this email address. You need to accept a subscription email request.<br>Example: admin@company.com |
| VpcId | String | The VPC ID in which the device needs to be deployed. This should be configured as per AWS requirements.<br>Type: AWS::EC2::VPC::Id<br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| LambdaSubnets | List | The subnets where Lambda functions will be deployed.<br>Type: List<AWS::EC2::Subnet::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| LambdaSG | List | The Security Groups for Lambda functions.<br>Type: List<AWS::EC2::SecurityGroup::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| S3BktName | String | The S3 bucket name for files. This should be configured in your account as per AWS requirements.<br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| LoadBalancerType | String | The type of Internet-facing Load Balancer, either “application” or “network”.<br>Example: application |
| LoadBalancerSG | String | The Security Groups for the Load Balancer. In the case of a network load balancer, it won't be used. But you should provide a Security Group ID.<br>Type: List<AWS::EC2::SecurityGroup::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| LoadBalancerPort | Integer | The Load Balancer port. This port will be opened on LB with either HTTP/HTTPS or TCP/TLS as the protocol, based on the chosen Load Balancer type.<br>Make sure the port is a valid TCP port, it will be used to create the Load Balancer listener.<br>Default: 80 |
| SSLcertificate | String | The ARN for the SSL certificate for secured port connections. If not specified, a port opened on the Load Balancer will be TCP/HTTP. If specified, a port opened on the Load Balancer will be TLS/HTTPS. |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| TgHealthPort | Integer | This port is used by the Target group for health probes. Health probes arriving at this port on the Firewall Threat Defense Virtual will be routed to the AWS Metadata server and should not be used for traffic. It should be a valid TCP port.<br>If you want your application itself to reply to health probes, then accordingly NAT rules can be changed for the Firewall Threat Defense Virtual. In such a case, if the application does not respond, the Firewall Threat Defense Virtual will be marked as unhealthy and deleted due to the Unhealthy instance threshold alarm.<br>Example: 8080 |
| AssignPublicIP | Boolean | If selected as "true" then a public IP will be assigned. In case of a BYOL-type Firewall Threat Defense Virtual, this is required to connect to https://tools.cisco.com.<br>Example: TRUE |
| InstanceType | String | The Amazon Machine Image (AMI) supports different instance types, which determine the size of the instance and the required amount of memory.<br>Only AMI instance types that support the Firewall Threat Defense Virtual should be used.<br>Example: c4.2xlarge |
| LicenseType | String | The Firewall Threat Defense Virtual license type, either BYOL or PAYG. Make sure the related AMI ID is of the same licensing type.<br>Example: BYOL |
| AmiId | String | The Firewall Threat Defense Virtual AMI ID (a valid Cisco Firewall Threat Defense Virtual AMI ID).<br>Type: AWS::EC2::Image::Id<br>Please choose the correct AMI ID as per the region and desired version of the image. The Auto Scale feature supports version 6.4+, BYOL/PAYG images. In either case you should have accepted a License in the AWS marketplace.<br>In the case of BYOL, please update 'licenseCaps' key in Configuration JSON with features such as 'BASE', 'MALWARE', 'THREAT', 'URLFilter' etc. |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| NoOfAZs | Integer | The number of availability zones that the Firewall Threat Defense Virtual should span across, between 1 and 3. In the case of an ALB deployment, the minimum value is 2, as required by AWS.<br>Example: 2 |
| ListOfAzs | Comma separated string | A comma-separated list of zones in order.<br>**Note** The order in which these are listed matters. Subnet lists should be given in the same order.<br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value.<br>Example: us-east-1a, us-east-1b, us-east-1c |
| MgmtInterfaceSG | String | The Security Group for the Firewall Threat Defense Virtual Management interface.<br>Type: List<AWS::EC2::SecurityGroup::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| InsideInterfaceSG | String | The Security Group for the Firewall Threat Defense Virtual inside interface.<br>Type: AWS::EC2::SecurityGroup::Id<br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| OutsideInterfaceSG | String | The Security Group for the Firewall Threat Defense Virtual outside interface.<br>Type: AWS::EC2::SecurityGroup::Id<br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value.<br>Example: sg-0c190a824b22d52bb |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| MgmtSubnetId | Comma separated list | A comma-separated list of management subnet-ids. The list should be in the same order as the corresponding availability zones.<br>Type: List<AWS::EC2::SecurityGroup::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| InsideSubnetId | Comma separated list | A comma-separated list of inside/Gig0/0 subnet-ids. The list should be in the same order as the corresponding availability zones.<br>Type: List<AWS::EC2::SecurityGroup::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| OutsideSubnetId | Comma separated list | A comma-separated list of outside/Gig0/1 subnet-ids. The list should be in the same order as the corresponding availability zones.<br>Type: List<AWS::EC2::SecurityGroup::Id><br>If the "*infrastructure.yaml*" file is used to deploy the infrastructure, the output section of the stack will have this value. Please use that value. |
| KmsArn | String | The ARN of an existing KMS (AWS KMS key to encrypt at rest). If specified, the Firewall Management Center and Firewall Threat Defense Virtual passwords should be encrypted. The password encryption should be done using only the specified ARN.<br>Generating Encrypted Password Example: " aws kms encrypt --key-id <KMS ARN> --plaintext <password> ". Please used such generated passwords as shown.<br>Example: arn:aws:kms:us-east-1:[AWS Account]:key/7d586a25-5875-43b1-bb68-a452e2f6468e |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| ngfwPassword | String | All the Firewall Threat Defense Virtual instances come up with a default password, which is entered in the *Userdata* field of the Launch Template (Autoscale Group).<br>This input will change the password to new provided password once the Firewall Threat Defense Virtual is accessible.<br>Please use a plain text password if KMS ARN is not used. If KMS ARN is used, then an encrypted password should be used.<br>Example: Cisco123789! or AQIAgcQFAGtz/hvaxMtJvY/x/rfHnI3lPpSXU |
| fmcServer | Numeric string | The IP address of managing the Firewall Management Center, which is reachable to both Lambda functions and the Firewall Threat Defense Virtual management interface.<br>Example: 10.10.17.21 |
| fmcOperationsUsername | String | The Network-Admin or higher privileged user created in managing the Firewall Management Center. See the information about creating users and roles in the Cisco Secure Firewall Management Center Device Configuration Guide.<br>Example: apiuser-1 |
| fmcOperationsPassword | String | Please use a plain text password if KMS ARN is not mentioned. If mentioned, then an encrypted password should be used.<br>Example: Cisco123@ or AQICAHgcQAtz/hvaxMtJvY/x/rnKI3clFPpSXUHQRnCAajB |
| fmcDeviceGrpName | String | The Firewall Management Center device group name.<br>Example: AWS-Cisco-NGFW-VMs-1 |
| fmcPerformanceLicenseTier | String | The performance tier license used while registering the Firewall Threat Defense Virtual device on the Firewall Management Center Virtual.<br>Allowed values: FTDv/FTDv20/FTDv30/FTDv50/FTDv100<br>**Note**<br>FTDv5 and FTDv10 performance tier licenses are not supported with AWS Gateway Load Balancer. |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| fmcPublishMetrics | Boolean | If set to "TRUE", then a Lambda function will be created which runs once in every 2 minutes to fetch the memory consumption of registered Firewall Threat Defense Virtual sensors in the provided device group.<br>Allowed values: TRUE, FALSE<br>Example: TRUE |
| fmcMetricsUsername | String | The unique Firewall Management Center user name for metric publication to AWS CloudWatch. See the information about creating users and roles in the Cisco Secure Firewall Management Center Device Configuration Guide.<br>If the "fmcPublishMetrics' is set to "FALSE" then there is no need to provide this input.<br>Example: publisher-1 |
| fmcMetricsPassword | String | The Firewall Management Center password for metric publication to AWS CloudWatch. Please use a plain text password if KMS ARN is not mentioned. If mentioned, then an encrypted password should be used.<br>If the "fmcPublishMetrics' is set to "FALSE" then there is no need to provide this input.<br>Example: Cisco123789! |
| CpuThresholds | Comma separated integers | The lower CPU threshold and the upper CPU threshold. The minimum value is 0 and maximum value is 99.<br>Defaults: 10, 70<br>Please note that the lower threshold should be less than the upper threshold.<br>Example: 30,70 |
| MemoryThresholds | Comma separated integers | The lower MEM threshold and the upper MEM threshold. The minimum value is 0 and maximum value is 99.<br>Defaults: 40, 70<br>Please note that the lower threshold should be less than the upper threshold. If the "fmcPublishMetrics" parameter is "FALSE" then this has no effect.<br>Example: 40,50 |

| Parameter | Allowed Values/Type | Description |
|---|---|---|
| Instance Metadata Service Version | Boolean | The Instance Metadata Data Service (IMDS) version you want enable for Firewall Threat Defense Virtual instances:<br>• V1 and V2 (token optional) : Enables either IMDSv1, IMDSv2, or a combination of both IMDSv1 and IMDSv2 API calls.<br>• V2 only (token required) : Enables only the IMDSv2 mode.<br>**Note**<br>Firewall Threat Defense Virtual Version 7.6 and later support only IMDSv2.<br>If you are enabling IMDSv2 service for versions earlier than 7.6, you must select combination of both IMDSv1 and IMDSv2 **V1 and V2 (token optional)** parameters.<br>**Note**<br>If you are using a custom template (that is not provided by Cisco) note that you must include the **HttpEndpoint: enabled** and **HttpTokens: required** properties under **MetadataOptions** in your template to enable the IMDSv2 Required mode. |

# Download the Required Files and CFTs from GitHub to your Local Host

Download the **lambda-python-files** folder from GitHub. This folder contains the following files:

- Python (.py) files that are used to create the lambda layer.
- A configuration.json file that is used to add static routes and customize any network parameters, as required.

Download the following CloudFormation templates from GitHub:

- Templates for the Auto Scale solution with NLB-
  - **infrastructure.yaml –** Used to customize the components in the AWS environment.
  - **deploy_ngfw_autoscale.yaml –** Used to deploy the AWS Auto Scale with NLB solution.
- Templates for the Auto Scale solution with GWLB-
  - **infrastructure_gwlb.yaml –** Used to customize the components in the AWS environment.
  - **deploy_ngfw_autoscale_with_gwlb.yaml –** Used to deploy the AWS Auto Scale with GWLB solution.

**Note** Collect values for the template parameters, wherever possible. This will make it easier to enter the values quickly while deploying the templates on the AWS Management console.

# Auto Scale Solution with NLB - Customize and Deploy the NLB Infrastructure Template on the Amazon CloudFormation Console

Perform the steps given in this section if you are deploying the auto scale solution with NLB.

**Procedure**

**Step 1** On the AWS Management console, go to **Services** > **Management and Governance** > **CloudFormation**, and click **Create stack** > **With new resources(standard)**.

**Step 2** Choose **Upload a template** file, click **Choose** file, and select **infrastructure.yaml** from the folder in which you downloaded the files.

**Step 3** Click **Next**

**Step 4** On the **Specify stack details** page, enter a name for the stack.

**Step 5** Provide values for the input parameters in the ***infrastructure.yaml*** template.

**Step 6** Click **Next**.

**Step 7** Click **Next** on the **Configure Stack Options** window.

**Step 8** On the **Review** page, review and confirm the settings.

**Step 9** Click **Create Stack** to deploy the **infrastructure.yaml** template and create the stack.

**Step 10** After the deployment is complete, go to **Outputs** and note the **S3 Bucket Name**.

# Auto Scale Solution with GWLB - Customize and Deploy the GWLB Infrastructure Template on the Amazon CloudFormation Console

Perform the steps given in this section if you are deploying the auto scale solution using GWLB.

**Procedure**

**Step 1** On the AWS Management console, go to **Services** > **Management and Governance** > **CloudFormation**, and click **Create stack** > **With new resources(standard)**.

**Step 2** Choose **Upload a template** file, click **Choose** file, and select **infrastructure_gwlb.yaml** from the folder in which you downloaded the files.

**Step 3** Click **Next**

**Step 4** On the **Specify stack details** page, enter a name for the stack.

**Step 5** Provide values for the input parameters in the ***infrastructure_gwlb.yaml*** template.

**Step 6** Click **Next**.

**Step 7** Click **Next** on the **Configure Stack Options** window.

**Step 8** On the **Review** page, review and confirm the settings.

**Step 9** Click **Create Stack** to deploy the **infrastructure_gwlb.yaml** template and create the stack.

**Step 10** After the deployment is complete, go to **Outputs** and note the **S3 Bucket Name**.

# Configure Network Infrastructure in the Management Center

Create and configure device droups, objects, health check port, NAT policy, and access policies, in the Management centre for the registered Threat Defense Virtual.

You can manage the threat defense virtual using the management center, a full-featured, multidevice manager on a separate server. The threat defense virtual registers and communicates with the management center on the Management interface that you allocated to the threat defense virtual virtual machine.

See About Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center for more information.

All the objects used for the threat defense virtual configuration should be created by user.

## Add Users

### Procedure

**Step 1** Log in to the Management Center. You must create two user accounts in the management center: one for Autoscale automation and the other for user publishing metrics to CloudWatch.

**Step 2** Choose **Settings** > **Users**.

**Step 3** To create a new user:

a) Click **Create User**.

b) Enter a **User Name**.

The username must comply with the following restrictions:

- Maximum 32 alphanumeric characters, plus hyphen (-) and underscore (_).
- Letters may be upper case or lower case.
- Cannot include any punctuation or special characters other than period (.), hyphen (-), and underscore (_).

**Step 4** Enter values in the **Password** and **Confirm Password** fields.

The values must conform to the password options you set for this user.

**Step 5** In the **User Role Configuration** area, check the **Administrator** check box.

Note down these user credentials to use when deploying the Autoscale Solution Template. You need to mention these user credentials in the FMC Automation Configuration and FMC Device Group Metrics Publish Configuration.

**Step 6** Click **Save**.

# Add Device Group

The management centre allows you to group devices so you can easily deploy policies and install updates on multiple devices. You can expand and collapse the list of devices in the group.



**Important** A device group should be created and rules should be applied on it. All the configurations applied on the device group will be pushed to the Threat Defense Virtual instances.

### Procedure

**Step 1** Choose **Devices** > **Device Management**.

**Step 2** From the **Add** drop-down menu, choose **Add Group**.

**Step 3** To edit an existing group, click Edit (edit icon) for the group you want to edit.

**Step 4** Enter a **Name**.

**Step 5** Click **OK** to add the device group.

**Note**
Enter this Device Group Name under **fmcDeviceGrp** when creating Autoscale Stack.

# Create a Host object

### Before you begin

You must create a host object when using the AWS Network Load Balancer (NLB).



**Note** AWS Gateway Load Balancer (GWLB) does not require a host object.

### Procedure

**Step 1** Log in to the Management Center.
**Step 2** Choose **Objects** > **Object Management**.
**Step 3** Choose **Network** from the list of object types.
**Step 4** Choose **Add Object** from the **Add Network** drop-down menu.
**Step 5** Enter a **Name**.
**Step 6** Enter a Description.
**Step 7** In the **Network** field, select the **Host** option and enter the following values.
a) Name of the object type as **aws-metadata-server**.
b) Depending on the type of host protocol, enter the following IP address for IPv4 - **169.254.169.254**.
**Step 8** Click **Save**.

# Create a Port object

#### Before you begin

You must create a port object when using the AWS Network Load Balancer (NLB).

**Note** AWS Gateway Load Balancer (GWLB) does not require a port object.

### Procedure

**Step 1** Log in to the Management Center.
**Step 2** Choose **Objects** > **Object Management**.
**Step 3** Choose **Port** from the list of object types.
**Step 4** Choose **Add Object** from the **Add Port** drop-down menu.
**Step 5** Enter a **Name**.
**Step 6** Choose a **Protocol**. You must choose the protocol that you have entered for the Host object type. Depending on the protocol you chose, constrain by **Port**, or choose an ICMP **Type** and **Code**.
**Step 7** Enter **8080**. Note that you can customise the port number that you enter here as per your requirement.

**Note**
You must constrain the object by port if you chose to match All protocols, using the **Other** drop-down list.

**Step 8** Click **Save**.

# Create Security Zone and Interface Group Objects

**Procedure**

**Step 1** Choose **Objects** > **Object Management**.
**Step 2** Choose **Interface** from the list of object types.
**Step 3** Click **Add** > **Security Zone**.
**Step 4** Enter a **Name** - *Inside-sz*.
**Step 5** Choose **Routed** from the **Interface Type** drop-down list.
**Step 6** Click **Save**.
**Step 7** Similarly, you must create *Outside-sz* by repeating Step 1 through Step 6.
**Step 8** Create **VNI Security Zone** if deploying GWLB Dual-arm.

# Enable Port for Health Check Probe

You can enable port 22 (SSH) or port 443 (HTTP) for the health check probe.

**Note** AWS Gateway Load Balancer (GWLB) does not require the health check probe.

## Enable Port 22 (SSH) for Health Check Probe

If you are using port 22 (SSH) for the health check probe, perform the following procedure to enable the port for the health check probe.

**Procedure**

**Step 1** Choose **Devices** > **Platform Settings** > **SSH Access**.
**Step 2** Click **+ Add**.
**Step 3** Select any **IP Address** from the drop-down list.
**Step 4** From the **Available Zones/Interfaces** window, select the outside interface that is connected to the GWLB or the outside subnet.
**Step 5** Click **Add** to add that interface to the **Selected Zones/Interfaces** window.
**Step 6** Click **OK**.
**Step 7** Click **Save**.

## Enable Port 443 (HTTP) for Health Check Probe

If you are using port 443 (HTTP) for the health check probe, perform the following procedure to enable the port for the health check probe.

### Procedure

**Step 1** Choose **Devices** > **Platform Settings** > **HTTP Access**.

**Step 2** Select the **Enable HTTP Server** checkbox.

**Step 3** Enter **443** in the **Port** field.

**Step 4** Click + **Add**.

**Step 5** Select the relevant **IP Address** from the drop-down list.

**Step 6** From the **Available Zones/Interfaces** window, select the outside interface that is connected to the GWLB or the outside subnet.

**Step 7** Click **Add** to add that interface to the **Selected Zones/Interfaces** window.

**Step 8** Click **OK**.

**Step 9** Click **Save**.

# Auto Scale Solution with NLB - Configure and Deploy Network Address Translation (NAT) Policy

A typical NAT rule converts internal addresses to a port on the outside interface IP address. This type of NAT rule is called interface Port Address Translation (PAT). See Configure NAT in Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center for information about the NAT policy.

One mandatory rule is required in your NAT policy. An example of a NAT rule is given below:

- Source Zone: Outside Zone
- Dest Zone: Inside Zone
- Original-sources: any-ipv4
- Original source port: Original/default
- Original Destinations: Interface
- Original-destination-port: 8080/or any health port that user configures
- Translated-sources: any-ipv4
- Translated source port: Original/default
- Translated-destination: aws-metadata-server
- Translated-destination-port: 80/HTTP

Similarly, any data-traffic NAT rules can be added, so that this configuration will be pushed to the threat defense virtual devices.

**Important** NAT Policy created should be applied on the device group. The management center validation from the Lambda function verifies this.

### Procedure

**Step 1** Log in to Secure Firewall Management Center.

**Step 2** On the **Devices** menu, click **NAT**.

**Step 3** Click **New Policy** > **Threat Defense NAT** to create a new policy.

**Step 4** Enter the Name and Description for the NAT policy.

**Step 5** Click **Save**.

You can see a new policy is added and listed on the NAT page.

**Step 6** Click **Add Rule**.

**Step 7** Select the **Manual NAT Rule** from the **NAT Rule** drop-down list.

**Step 8** Select **In Category** and **NAT Rule** Before from the Insert drop-down list.

**Step 9** Select **Static** from the **Type** drop-down menu.

**Step 10** Enter the description.

**Step 11** In the **Interface Objects** menu, add the source and destination objects.

**Step 12** In the **Translations** menu, add the following values for each parameter.

| Parameter | Values |
|---|---|
| Original Source | any-ipv4 |
| Original Destination | Address |
| Original Source Port | HTTP |
| Original Destination Port | 8080 |
| Translated Source | any-ipv4 |
| Translated Source Port | Original/default |
| Translated Destination | aws-metadata-server |
| Translated Destination Port | 80/HTTP |

**Step 13** Click **Save** to save and add the Rule.

**Step 14** Select the new rule you have created to deploy on the threat defense virtual.

**Step 15** Click **Deploy** > **Deployment** to deploy the policy to assigned devices. The changes are not active until you deploy them.

# Create a Basic Access Control Policy

When you create a new access control policy, it contains default actions and settings. After creating the policy, you are immediately placed in an edit session so that you can adjust the policy to suit your requirements.

**Procedure**

**Step 1** Choose **Policies** > **Access Control**.

**Step 2** Click **New Policy**.

**Step 3** Enter a unique Name - *aws-asg-policy* and Description.

**Step 4** Specify the initial **Default Action - Block all traffic**.

**Step 5** Click **Save**.

**Step 6** Click the **Targeted Devices** on the upper right side of the page to assign the policy.

**Step 7** Select the Device Group created previously.

**Step 8** Click **Add to Policy** and then click **OK** to add to policy.

**Step 9** Click the **Edit** icon for the new policy that you created.

**Step 10** Click **Add Rule**.

**Step 11** Set the following parameters:

- Name: *inside-to-outside*
- Insert: *into Mandatory*
- Action: *Allow*
- Add a source zone and destination zone.

**Step 12** Click **Apply**.

# Update the configuration.json and configuration-schema.json Files

The `configuration.json` and `configuration-schema.json` files are in the **lambda_python_files** folder that you downloaded from GitHub. Update the parameters in the `configuration.json` file with the parameters set up by you in the management center. Note that the JSON key should not be changed.

When editing the `configuration.json` file, ensure that the names of interfaces, security zone objects, device groups, and policies exactly match the corresponding object names in the management center.

The sample scripts for the `configuration.json` and `configuration-schema.json` files are available in the **sample-az-configuration-jsons** folder at https://github.com/CiscoDevNet/cisco-ftdv/tree/master/autoscale/aws/sample-configuration-jsons.

Based on your deployment topology, you must replace the script in the `configuration.json` and `configuration-schema.json` files with the sample script from the files with the same name as follows.

- For deploying **GWLB single-arm** topology - Use `gwlb-single-arm-Configuration.json` and `gwlb-single-arm-Configuration-schema.json`.
- For deploying **GWLB dual-arm** topology - Use 'gwlb-dual-arm-configuration.json and `gwlb-dual-arm-configuration-schema.json`.
- For deploying **NLB** topology - Use `nlb-Configuration.json` and `nlb-Configuration-schema.json`.

# Configure Infrastructure Components using AWS CLI

The templates do not create the Lambda layer and encrypted passwords for the threat defense virtual and management center. Configure these components using the procedures given below. See AWS Command Line Interface for more information on the AWS CLI.

## Create the Lambda Layer Zip File to Manage Compute Resources

Create Lambda Layer *autoscale_layer.zip* needed for Lambda Functions.

The *autoscale_layer.zip* can be created on an Amazon Linux VM, with Python 3.11 installed. It is recommended to use AWS Cloudshell, which runs the latest version of Amazon Linux.

For creating the *autoscale_layer.zip* file, you need to first create *requirements.txt* file that consists of the python library package details and then run the shell script.

### Procedure

**Step 1** Create the *requirements.txt* file by specifying the python package details.

```
$ cat requirements.txt
pycryptodome
paramiko
requests
scp
jsonschema
cffi
zipp
importlib-metadata
```

**Step 2** Run the following commands to create autoscale_layer.zip file.

```
$ pip3 install --platform manylinux2014_x86_64
--target=./python/lib/python3.11/site-packages
--implementation cp --python-version 3.11 --only-binary=:all:
--upgrade -r requirements.txt
$ zip -r autoscale_layer.zip ./python
```

**Note**
If you encounter a dependency conflict during installation, for example, for packages urllib3 or cryptography, include the conflicting packages along with their recommended versions in the requirements.txt file. Run the installation command again to resolve the conflict.

**Step 3** Place the resultant *autoscale_layer.zip* file in the directory `cisco-ftdv/autoscale/aws/lambda-python-files`.

**Step 4** Create **autoscale_manager.zip**, **lifecycle_ftdv.zip** and **custom_metric_fmc.zip** files.

a) Run the following `make.py` file that is available in the cloned repository of the top-level directory.

```
python3 make.py build
```

It creates compressed python files into *four* separate zip files and then copies them to the **target** folder.

b) Upload these *four* zip files to the S3 bucket created by the infrastructure template. Ensure that the Python 3 environment is available.

# (Optional) Create Encrypted Passwords for the Threat Defense Virtual and Management Center

If a KMS ARN value has been entered in the infrastructure_gwlb.yaml template file, the passwords that you set up in the threat defense virtual and management centre have to be encrypted. See Finding the key ID and key ARN to identify the key ARN using the AWS KMS console. On your local host, encrypt the password by running the following AWS CLI command.

```
$ aws kms encrypt --key-id <KMS-ARN> --plaintext
'MyC0mplIc@tedProtect1oN'
{
    "KeyId": "KMS-ARN",
    "CiphertextBlob":
"AQICAHgcQFAGtz/hvaxMtJvY/x/rfHnKI3clFPpSXUU7HQRnCAFwfXhXH
JAHL8tcVmDqurALAAAAajBoBgkqhki
G9w0BBwagWzBZAgEAMFQGCSqGSIb3DQEHATAeBglghkgBZQMEAS4wEQQM45
AIkTqjSekX2mniAgEQgCcOav6Hhol
+wxpWKtXY4y1Z1d0z1P4fx0jTdosfCbPnUExmNJ4zdx8="
}
$
```

The value of “CiphertextBlob” is the encrypted password. Use this password as the value of the **NGFWv Password** (threat defense virtual password) or the **FMC Password for AutoScale Automation** (management center password) parameter in the **infrastructure_gwlb.yaml** file. You can also use this password as the value of the **FMC Password for Publishing Metrics to CloudWatch**.

# Create Target Folder

On the local host, use the command given below to create a target folder containing the files that have to be uploaded to the Amazon S3 bucket.

**python3 make.py build**

This creates a folder named ‘target’ on your local host. The target folder contains the *zip* files and *yaml* files required for deployment of the auto scale solution.

# Upload Files to Amazon S3 Bucket

On the local host, use the commands given below to upload all the files in the target directory to the Amazon S3 bucket.

**$ cd ./target**

**$ aws s3 cp . s3://**<*bucket-name*> **--recursive**

# Auto Scale Solution with NLB - Deploy the Auto Scale Solution with NLB

Perform the steps given in this section if you are deploying the auto scale solution with NLB.

**Procedure**

**Step 1** On the AWS Management console, go to **Services** > **Management and Governance** > **CloudFormation** > **Stacks,** and click the stack that was created by the template.

**Step 2** Click **Create stack** > **With new resources(standard)**.

**Step 3** Select **Upload a template** file, click **Choose** File, and select ***deploy_ngfw_autoscale.yaml*** from the target folder.

**Step 4** Click **Next**.

**Step 5** On the **Specify stack details** page, enter a name for the stack.

**Step 6** Provide values for the input parameters in the ***deploy_ngfw_autoscale.yaml*** template.

**Step 7** Click **Next** on the **Configure Stack Options** window.

**Step 8** On the **Review** page, review and confirm the settings.

**Step 9** Click **Create Stack** to deploy the ***deploy_ngfw_autoscale.yaml*** template and create the stack.

This completes deployment of both the templates that are required to set up the auto scale solution for threat defense virtual with NLB.

# Auto Scale Solution with GWLB - Deploy the Auto Scale Solution with GWLB

Perform the steps given in this section if you are deploying the auto scale solution with GWLB.

**Procedure**

**Step 1** On the AWS Management console, go to **Services** > **Management and Governance** > **CloudFormation** > **Stacks,** and click the stack that was created by the template.

**Step 2** Click **Create stack** > **With new resources(standard)**.

**Step 3** Select **Upload a template** file, click **Choose** File, and select ***deploy_ngfw_autoscale_with_gwlb.yaml*** from the target folder.

**Step 4** Click **Next**.

**Step 5** On the **Specify stack details** page, enter a name for the stack.

**Step 6** Provide values for the input parameters in the ***deploy_ngfw_autoscale_with_gwlb.yaml*** template.

**Step 7** Click **Next** on the **Configure Stack Options** window.

**Step 8** On the **Review** page, review and confirm the settings.

**Step 9** Click **Create Stack** to deploy the ***deploy_ngfw_autoscale_with_gwlb.yaml*** template and create the stack.

This completes deployment of both the templates that are required to set up the auto scale solution for threat defense virtual using GWLB.

# AWS Security Group Changes

1. Allow eipNat created by Infrastructure Stack to FMC Security Group (If using FMC Public IP address).
2. Allow FMC IP to FTD Instance Security Group (If using public FTD IP address).

# Auto Scale with GWLB solution - Create the GWLB Endpoint

Perform the steps given in this section if you are deploying the auto scale solution using GWLB.

**Procedure**

**Step 1** On the AWS Management console, go to **Services** > **Networking & Content Delivery** > **VPC** > **Endpoint Services**.

**Step 2** Click **Create Endpoint Service**

**Step 3** Under Load balancer type, choose **Gateway**.

**Step 4** Under **Available load balancers**, choose the Gateway Load balancer that was created as part of the Auto scale deployment.

**Step 5** Under **Require acceptance for endpoint,** choose **Acceptance required**. This ensures that you have to manually accept any endpoint service connection requests.

**Step 6** Under **Supported IP address types**, choose **IPv4**.

**Step 7** Click **Create**.

**Step 8** Copy the Service name of the newly created endpoint service.

**Step 9** Go to **Services** > **Networking & Content Delivery** > **VPC** > **Endpoints**.

**Step 10** Click **Create endpoint**.

**Step 11** Under **Service category**, choose **Other endpoint services**.

**Step 12** For **Service name**, enter the name of the service, and then choose **Verify service**.

**Step 13** In the **VPC** field, select the VPC in which to create the endpoint.

**Step 14** Under **Subnets**, select the subnet in which to create the endpoint.

**Step 15** For IP address type, choose the IPv4 option to assign IPv4 addresses to the endpoint network interfaces.

**Step 16** Click **Create endpoint**.

# Configure Routing for VPC

**Procedure**

**Step 1** On the AWS Management console, go to **Services** > **Networking & Content** > **Virtual Private Cloud** > **Route tables**.

**Step 2** Select the route table for the internet gateway and perform the following steps:

**a.** Click **Actions** > **Edit routes**.

**b.** For IPv4, click **Add route**. For **Destination**, enter the IPv4 CIDR block of the subnet for the application servers. For **Target**, select the VPC endpoint.

**c.** Click **Save changes**.

**Step 3** Select the route table for the subnet with the application servers and perform the following steps:

**a.** Click **Actions** > **Edit routes**.

**b.** For IPv4, click **Add route**. For **Destination**, enter **0.0.0.0/0**. For **Target**, select the VPC endpoint.

**c.** Click **Save changes**.

**Step 4** Select the route table for the subnet with the Gateway Load Balancer endpoint, and perform the following steps:

**a.** Click **Actions** > **Edit routes**.

**b.** For IPv4, click **Add route**. For **Destination**, enter **0.0.0.0/0**. For **Target**, select the internet gateway.

**c.** Click **Save changes**.

# Edit the Auto Scale Group

By default, the Auto Scale group has the minimum and maximum number of threat defense virtual instances set to 0 and 2 respectively. Change these values as per your requirement.

**Procedure**

**Step 1** On the AWS Management console, go to **Services** > **Compute** > **EC2**, and click **Auto Scaling Groups**.

**Step 2** Select the auto scaling group created by you and click **Edit** to modify the values in the **Desired capacity**, **Minimum capacity**, **Maximum capacity** fields as per your requirement. These values correspond to the number of threat defense

virtual instances that you want to bring up for the auto scaling functionality. Set the **Desired capacity** to a value that is within the minimum and maximum capacity values.

**Step 3** Click **Update**.

**Note** We recommend that you launch only one threat defense virtual instance and verify that the behaviour of this instance is as expected. You can then launch more instances as per your requirement.

# Validate Deployment

Once the template deployment is successful, go to the Amazon CloudWatch console to ensure that logs are being collected and the required alarms have been created.

## Logs

Check the log files to troubleshoot any issues with management center connectivity.

**Procedure**

**Step 1** On the **AWS Management** console, go to **Services** > **Management & Governance** > **CloudWatch**.

**Step 2** Click **Log groups** and click any log group displayed here to view the logs.

## Alarms

Ensure that the required alarms have been created on the Amazon CloudWatch console.

**Procedure**

**Step 1** On the **AWS Management** console, go to **Services** > **Management & Governance** > **CloudWatch**.

**Step 2** Click **Alarms** > **All Alarms** to display the list of alarms along with the conditions which will trigger the scale-out and scale-in functions.

# Maintenance Tasks

## Scaling Processes

This topic explains how to suspend and then resume one or more of the scaling processes for your Auto Scale group.

### Start and Stop Scale Actions

To start and stop scale out/in actions, follow these steps.

- For AWS Dynamic Scaling—Refer to the following link for information to enable or disable scale out actions:

  Suspending and Resuming Scaling Processes

## Health Monitor

Every 60 minutes, a CloudWatch Cron job triggers the Auto Scale Manager Lambda for the Health Doctor module:

- If there are unhealthy IPs which belong to a valid Firewall Threat Defense Virtual VM, that instance gets deleted if the Firewall Threat Defense Virtual is more than an hour old.
- If those IPs are not from a valid Firewall Threat Defense Virtual machine, then only IPs are removed from the Target Group.

The health monitor also validates the Firewall Management Center configuration for device group, access policy, and NAT rules. In case of an unhealthy IP/instance, or if the Firewall Management Center validation fails, the health monitor sends an email to the user.

### Disable Health Monitor

To disable a health monitor, in ***constant.py*** make the constant as “True”.

### Enable Health Monitor

To enable a health monitor, in ***constant.py*** make the constant as “False”.

## Disable Lifecycle Hooks

In the unlikely event that Lifecycle hook needs to be disabled, if disabled it won’t add additional interfaces to Instances. It can also cause a series of failed deployment of the Firewall Threat Defense Virtual instances.

## Disable Auto Scale Manager

To disable Auto Scale Manager, respective CloudWatch Events “notify-instance-launch” and “notify-instance-terminate” should be disabled. Disabling this won’t trigger Lambda for any new events. But

already executing Lambda actions will continue. There is no abrupt stop of Auto Scale Manager. Trying abrupt stopping by stack deletion or deleting resources can cause an indefinite state.

# Load Balancer Targets

Because the AWS Load Balancer does not allow instance-type targets for instances having more than one network interface, the Gigabit0/1 interface IP is configured as a target on Target Groups. As of now however, the AWS Auto Scale health checks work only for instance-type targets, not IPs. Also, these IPs are not automatically added or removed from target groups. Hence our Auto Scale solution programmatically handles both of these tasks. But in the case of maintenance or troubleshooting, there could be a situation demanding manual effort to do so.

### Register a Target to a Target Group

To register the Firewall Threat Defense Virtual instance to the Load Balancer, its Gigabit0/1 instance IP (outside subnet) should be added as a target in Target Group(s). See Register or Deregister Targets by IP Address.

### Deregister a Target from a Target Group

To deregister the Firewall Threat Defense Virtual instance to the Load Balancer, its Gigabit0/1 instance IP (outside subnet) should be deleted as a target in Target Group(s). See Register or Deregister Targets by IP Address.

# Instance Stand-by

AWS does not allow instance reboot in the Auto Scale group, but it does allow a user to put an instance in Stand-by and perform such actions. However, this works best when the Load Balancer targets are instance-type. However, the Firewall Threat Defense Virtual machines cannot be configured as instance-type targets, because of multiple network interfaces.

### Put an Instance in Stand-by

If an instance is put into stand-by, its IP in Target Groups will still continue to be in the same state until the health probes fail. Because of this, it is recommended to deregister respective IPs from the Target Group before putting the instance into stand-by state; see Load Balancer Targets, on page 146 for more information.

Once the IPs are removed, see Temporarily Removing Instances from Your Auto Scaling Group.

### Remove an Instance from Stand-by

Similarly you can move an instance from stand-by to running state. After removal from stand-by state, the instance's IP should be registered to Target Group targets. See Load Balancer Targets, on page 146.

For more information about how to put instances into stand-by state for troubleshooting or maintenance, see the AWS News Blog.

### Remove/Detach Instance from Auto Scale Group

To remove an instance from the Auto Scale group, first it should be moved to stand-by state. See "Put Instances on Stand-by". Once the instance is in the stand-by state it can be removed or detached. See Detach EC2 Instances from Your Auto Scaling Group.

There won't be any changes on the Firewall Management Center side. Any changes required should be performed manually.

# Terminate an Instance

To terminate an instance it should be put into stand-by state; see Instance Stand-by, on page 146. Once the instance is in stand-by, you can proceed to terminate.

# Instance Scale-In Protection

To avoid an accidental removal of any particular instance from the Auto Scale group, it can be made as Scale-In protected. If an instance is Scale-In protected, it won't be terminated due to a Scale-In event.

Please refer to the following link to put an instance into Scale-In protected state.

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



**Important** It is recommended to make the minimum number of instances which are healthy (the target IP should be healthy, not just the EC2 instance) as Scale-In protected.

# Changes to Configuration

Any changes in configuration won't be automatically reflected on already running instances. Changes will be reflected on upcoming devices only. Any such changes should be manually pushed to already existing devices.

If you are facing issues while manually updating the configuration on existing instances, we recommend removing these instances from the Scaling Group and replacing them with new instances.

### Change the Firewall Management Center User Name and Password

In the case of changes to the Firewall Management Center IP, username, or password—the respective changes should be performed on Auto Scale Manager Lambda function and custom metric publisher Lambda function environment variables. See Using AWS Lambda Environment Variables.

When Lambda runs next time, it will reference the changed environment variables.



**Note** Environment variables are directly fed to Lambda functions. There is no password complexity check here.

### Change the Firewall Threat Defense Virtual Admin Password

A change to the Firewall Threat Defense Virtual password requires the user to change it on each device manually for running instances. For new Firewall Threat Defense Virtual devices to be onboarded, the Firewall Threat Defense Virtual password will be taken from the Lambda environment variables. See Using AWS Lambda Environment Variables.

### Change Registration and NAT IDs

For new Firewall Threat Defense Virtual devices to be onboarded with different registration and NAT IDs, for the Firewall Management Center registration this information should be changed in Configuration.json file. The Configuration.json file can be located in Lambda resource page.

## Changes to Access Policy and NAT Policy

Any changes to Access policies or NAT policies are automatically applied to upcoming instances with the help of the Device Group assignment. However, to update existing Firewall Threat Defense Virtual instances you need to manually push configuration changes and deploy them from the Firewall Management Center.

## Changes to AWS Resources

You can change many things in AWS post deployment, such as the Auto Scale Group, Launch Configuration, CloudWatch events, Scaling Policies etc. You can import your resources into a CloudFormation stack or create a new stack from your existing resources.

See Bringing Existing Resources Into CloudFormation Management for more information about how to manage changes performed on AWS resources.

## Collect and Analyze CloudWatch Logs

For information about exporting CloudWatch logs, see Export Log Data to Amazon S3 Using the AWS CLI.

# Configure IMDSv2 Required Mode for Existing Autoscale Group Instances

You can configure the IMDSv2 Required mode for the Firewall Threat Defense Virtual autoscale group instances that are already deployed on the AWS.

### Before you begin

The IMDSv2 Required mode is only supported in Firewall Threat Defense Virtual Version 7.6 and later. Ensure that your existing instances' version is compatible (upgraded to Version 7.6) with the IMDSv2 mode before configuring the IMDSv2 mode for your deployment.

### Procedure

**Step 1** Log in to http://aws.amazon.com/.

**Step 2** Click **EC2** and select **Auto Scaling** > **Auto Scaling Groups**.

**Step 3** Select the Auto Scaling group from the list to configure the IMDSv2 Required mode for that group's associated instances.

**Step 4** Click the **Launch Template**.

**Step 5** On the **Launch templates** page, choose **Modify template (Create new version)** from the **Actions** drop-down list.

**Step 6** Update the AMI ID with the IMDSv2-supported image.

**Step 7** Under **Advanced Details**, enable the IMDSv2 metadata:

a) Choose **Enabled** from the **Metadata accessible** drop-down list.

b) Choose **V2 only (token required)** from the **Metadata version** drop-down list.

**Step 8** Use this Metadata version of the launch template to deploy the auto scaling group instances with the IMDSv2 Required mode enabled.

# Troubleshooting

### AWS CloudFormation Console

You can verify the input parameters to your CloudFormation stack in the AWS CloudFormation Console, which allows you to create, monitor, update and delete stacks directly from your web browser.

Navigate to the required stack and check the parameter tab. You can also check inputs to Lambda Functions on the Lambda Functions environment variables tab. The *configuration.json* file can also be viewed on the Auto Scale Manager Lambda function itself.

To learn more about the AWS CloudFormation console, see the *AWS CloudFormation User Guide*.

### Amazon CloudWatch Logs

You can view logs of individual Lambda functions. AWS Lambda automatically monitors Lambda functions on your behalf, reporting metrics through Amazon CloudWatch. To help you troubleshoot failures in a function, Lambda logs all requests handled by your function and also automatically stores logs generated by your code through Amazon CloudWatch Logs.

You can view logs for Lambda by using the Lambda console, the CloudWatch console, the AWS CLI, or the CloudWatch API. To learn more about log groups and accessing them through the CloudWatch console, see the Monitoring system, application, and custom log files in the *Amazon CloudWatch User Guide*.

### Load Balancer Health Check Failure

The load balancer health check contains information such as the protocol, ping port, ping path, response timeout, and health check interval. An instance is considered healthy if it returns a 200 response code within the health check interval.

If the current state of some or all your instances is `OutOfService` and the description field displays the message that the `Instance has failed at least the Unhealthy Threshold number of health checks consecutively`, the instances have failed the load balancer health check.

You should check the health probe NAT rule in the Firewall Management Center configuration. For more information, see Troubleshoot a Classic Load Balancer: Health checks.

### Traffic Issues

To troubleshoot traffic issues with your Firewall Threat Defense Virtual instances, you should check the Load Balancer rules, the NAT rules, and the static routes configured in the Firewall Threat Defense Virtual instances.

You should also check the AWS virtual network/subnets/gateway details provided in the deployment template, including security group rules, etc. You can also refer to AWS documentation, for example, Troubleshooting EC2 instances.

### Connection to the Firewall Management Center Failed

If the management connection is disrupted, you should check the configuration and credentials. See "Requirements and Prerequisites for Device Management" in *Secure Firewall Management Center Configuration Guide*.

### Device Failed to Register with the Firewall Management Center

If the device fails to register with the Firewall Management Center fails, you need to determine if the Firewall Management Center configuration is faulty/unreachable, or if the Firewall Management Center has the capacity to accommodate a new device. See "Add a Device to the Management Center " in *Secure Firewall Management Center Configuration Guide*.

### Unable to SSH into the Firewall Threat Defense Virtual

If you are unable to SSH into the Firewall Threat Defense Virtual, check to see if the complex password was passed to the Firewall Threat Defense Virtual via the template.

### Troubleshooting Dual-arm Proxy Configuration

**Firewall Threat Defense Virtual EC2 instances fail to launch or go into a continuous launch-terminate loop.**

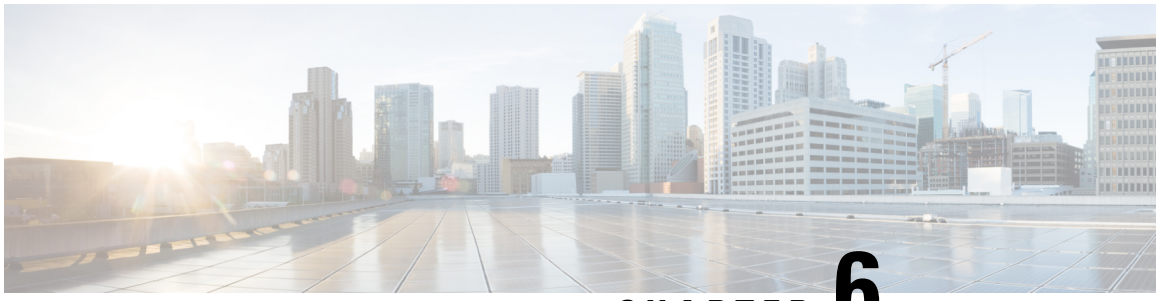
You must verify the activity history: go to **Autoscale Group** > **Autoscale Group** > **Activity** > **Activity History** to start debugging.

A possible cause may include missing AMI and error in the Lambda function execution.

- Ensure the AMI is present in AWS Account.
- Check Connection, Configuration, and Licensing.
- Check the Tags associated with the Firewall Threat Defense Virtual EC2 instances to get the device's connection status, configuration status, licensing status and other statuses.
- Check CloudWatch Logs
  - Logs of individual Lambda functions can be viewed in CloudWatch Logs.
  - If there are issues connecting or configuring the device, proceed to CloudWatch Logs of Lambda functions to understand the root cause.
  - Verify the log groups for Lambda functions.

### AWS Cloud-Side Debugging: Traffic Not Passing

- Check routing tables of all AWS subnets are as specified in the topology diagram. Ensure Egress (East-West traffic routes are present on devices.
- Ensure AWS Security Groups associated with all device interfaces allow the expected IP address ranges.



CHAPTER 6

# Deploy the Firewall Threat Defense Virtual on Azure

This chapter explains how to deploy the Secure Firewall Threat Defense Virtual from the Azure portal.


- Overview, on page 152
- Prerequisites, on page 152
- Guidelines and Limitations, on page 153
- How to Manage Secure Firewall Threat Defense Virtual Device, on page 157
- Sample Network Topology for the Firewall Threat Defense Virtual on Azure, on page 157
- Resources Created During Deployment , on page 158
- Accelerated Networking (AN), on page 159
- Azure Routing , on page 160
- Routing Configuration for VMs in the Virtual Network, on page 160
- IP Addresses, on page 161
- Deploy the Firewall Threat Defense Virtual, on page 161
- Deploy from the Azure Marketplace Using the Solution Template, on page 162
- Deploy from Azure Using a VHD and Resource Template, on page 165
- About Deployment of Threat Defense Virtual without Diagnostic Interface on Azure, on page 168
- Guidelines and Limitations for Deployment of Threat Defense Virtual without Diagnostic Interface on Azure, on page 169
- NIC Mapping to Data Interfaces for Deployment of Threat Defense Virtual without Diagnostic Interface on Azure, on page 169
- Deploy Threat Defense Virtual without Diagnostic Interface on Azure, on page 170
- Deploy the Azure Marketplace offers in the restricted Azure Private Marketplace environment, on page 172
- Upgrade Scenarios, on page 173
- Deployment of Threat Defense Virtual Cluster or Auto Scale Solution without Diagnostic Interface, on page 174
- Troubleshooting, on page 174
- Auto Scale Solution for the Threat Defense Virtual on Azure, on page 174
- Deploy the IPv6 Supported Secure Firewall Threat Defense Virtual on Azure, on page 218
- About IPv6 Supported Deployment on Azure, on page 218
- Deploy from Azure Using Custom IPv6 Template with Marketplace Image Reference, on page 220
- Deploy from Azure Using a VHD and Custom IPv6 Template, on page 225

- Deploy the Azure Marketplace offers in the restricted Azure Private Marketplace environment, on page 229
- Firewall Threat Defense Virtual Image Snapshot, on page 231


# Overview

The Secure Firewall Threat Defense Virtual is integrated into the Microsoft Azure marketplace and supports the following VM sizes:

***Table 19: Supported VM Sizes***

| VM Size | vCPUs | Memory (RAM) in GB | vNICs | Version |
|---|---|---|---|---|
| Standard_D3 | 4 | 14 | 4 | Any |
| Standard_D3_v2 | 4 | 14 | 4 | Any |
| Standard_D4_v2 | 8 | 28 | 8 | 6.5 or above |
| Standard_D5_v2 | 16 | 56 | 8 | 6.5 or above |
| Standard_D8s_v3 | 8 | 32 | 4 | 7.1 or above |
| Standard_D16s_v3 | 16 | 64 | 8 | 7.1 or above |
| Standard_F8s_v2 | 8 | 16 | 4 | 7.1 or above |
| Standard_F16s_v2 | 16 | 32 | 4 | 7.1 or above |

# Prerequisites

- A Microsoft Azure account. You can create one at https://azure.microsoft.com/en-us/ .

  After you create an account on Azure, you can log in, search the marketplace for Cisco Firepower Threat Defense, and choose the “Cisco Firepower NGFW Virtual (NGFWv)” offering.

- A Cisco Smart Account. You can create one at Cisco Software Central.

  License the Firewall Threat Defense Virtual; see Cisco Secure Firewall Management Center Feature Licenses for an overview of feature licenses for the firewall System, including helpful links.

- For the Firewall Threat Defense Virtual and system compatibility, see Firewall Threat Defense Virtual Compatibility.

### Communication Paths

- Management interface—Used to connect the Firewall Threat Defense Virtual to the Secure Firewall Management Center.

**Note** In 6.7 and later, you can optionally configure a data interface for Firewall Management Center management instead of the Management interface. The Management interface is a pre-requisite for data interface management, so you still need to configure it in your initial setup. For more information about configuring a data interface for Firewall Management Center access, see the **configure network management-data-interface** command in Cisco Secure Firewall Threat Defense Command Reference.

- Diagnostic interface—Used for diagnostics and reporting; cannot be used for through traffic.
- Inside interface (required)—Used to connect the Firewall Threat Defense Virtual to inside hosts.
- Outside interface (required)—Used to connect the Firewall Threat Defense Virtual to the public network.
- From Secure Firewall version 7.4.1, you can remove the diagnostic interface and deploy the Threat Defense Virtual on Azure with a minimum of 3 interfaces – 1 management, and 2 data interfaces. We recommend that you deploy the Threat Defense Virtual on Azure without the diagnostic interface from Secure Firewall version 7.4.1. For more information, see About Deployment of Threat Defense Virtual without Diagnostic Interface on Azure, on page 168.

# Guidelines and Limitations

### Supported Features

- Routed firewall mode only
- Azure Accelerated Networking (AN)
- Management mode, one of two choices:
  - You can use the Secure Firewall Management Center to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409.
  - You can use the integrated Secure Firewall Device Manager to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Device Manager, on page 427.
- Clustering (version 7.3 and later). For more information, see Clustering for Threat Defense Virtual in a Public Cloud.
- Public IP addressing—Assign public IP addresses to Management 0/0 and GigabitEthernet0/0.

  You can assign a public IP address to other interfaces as needed; see Public IP addresses for Azure's guidelines regarding public IPs, including how to create, change, or delete a public IP address.
- IPv6

  The following are the guidelines and limitations that must be considered while deploying IPv6 supported Firewall Threat Defense Virtual:

- For enabling the programmatic deployment option through the Azure CLI method for IPv6 support, pre-deployment of Firewall Threat Defense Virtual instance is not required.
- You cannot add a Firewall Threat Defense Virtual from the Azure Marketplace to the same Vnet that you have manually upgraded from IPV4 to IPV6 addressing.

- Interfaces:
  - Firewall Threat Defense Virtual deploys with 4 vNICs by default.
  - With larger instance support, you have the ability to deploy the Firewall Threat Defense Virtual with a maximum of 8 vNICs.
  - To add additional vNICs to your Firewall Threat Defense Virtual deployment, refer to the information given in Add network interfaces to or remove network interfaces from virtual machines.
  - To change the configuration of the vNICs, or if IP forwarding is required, refer to the information given in Create, change, or delete a network interface.
  - You configure your Firewall Threat Defense Virtual interfaces using your manager. See the configuration guide for your management platform, either Firewall Management Center or Firewall Device Manager, for complete information about interface support and configuration.

### Licensing

- BYOL (Bring Your Own License) using a Cisco Smart License Account.
- PAYG (Pay As You Go) licensing, a usage-based billing model that allows customer to run Firewall Threat Defense Virtual without having to purchase Cisco Smart Licensing. All licensed features (Malware/Threat/URL Filtering/VPN, etc.) are enabled for a registered PAYG Firewall Threat Defense Virtual device. Licensed features cannot be edited or modified from the Firewall Management Center. (Version 6.5+)

**Note** PAYG licensing is not supported on the Firewall Threat Defense Virtual devices deployed in the Firewall Device Manager mode.

See the "Licensing" chapter in the Secure Firewall Management Center Administration Guide for guidelines when licensing your Firewall Threat Defense Virtual device.

### Performance Tiers for Firewall Threat Defense Virtual Smart Licensing

The Firewall Threat Defense Virtual supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.

**Note** Any threat defense virtual tier license can be used with any supported threat defense virtual vCPU or memory configuration.

This allows threat defense virtual customers to use a license on a wide variety of VM resources.

This can also increase the number of Azure VM sizes that are supported. When configuring a threat defense virtual VM, the maximum number of cores (vCPUs) supported is 16, and the maximum memory supported is 32 GB RAM.

*Table 20: Firewall Threat Defense Virtual Licensed Feature Limits Based on Entitlement*

| Performance Tier | Suggested vCPU/Memory Configuration | Rate Limit | RA VPN Session Limit |
|---|---|---|---|
| FTDv5, 100Mbps | 4 core/8 GB | 100Mbps | 50 |
| FTDv10, 1Gbps | 4 core/8 GB | 1Gbps | 250 |
| FTDv20, 3Gbps | 4 core/8 GB | 3Gbps | 250 |
| FTDv30, 5Gbps | 8 core/16 GB | 5Gbps | 250 |
| FTDv50, 10Gbps | 12 core/24 GB | 10Gbps | 750 |
| FTDv100, 16Gbps | 16 core/34 GB | 16Gbps | 10,000 |

### Performance Optimizations

To achieve the best performance out of the Firewall Threat Defense Virtual, you can make adjustments to the both the VM and the host. See Virtualization Tuning and Optimization on Azure for more information.

**Receive Side Scaling**—The Firewall Threat Defense Virtual supports Receive Side Scaling (RSS), which is a technology utilized by network adapters to distribute network receive traffic to multiple processor cores. Supported on Version 7.0 and later. See Multiple RX Queues for Receive Side Scaling (RSS) for more information.

### Unsupported Features

- Licensing:
  - PLR (Permanent License Reservation)
  - PAYG (Pay As You Go) (Versions 6.4 and earlier)
- Networking (many of these limitations are Microsoft Azure restrictions):
  - Jumbo frames
  - 802.1Q VLANs
  - Transparent Mode and other Layer 2 features; no broadcast, no multicast.

- Proxy ARP for an IP address that the device does not own from an Azure perspective (impacts some NAT capabilities).
- Promiscuous mode (no capture of subnet traffic).
- Inline-set modes, passive mode.



**Note** Azure policy prevents the Firewall Threat Defense Virtual from operating in transparent firewall or inline mode because it does not allow interfaces to operate in promiscuous mode.

- ERSPAN (uses GRE, which is not forwarded in Azure).

- Management:
  - Azure portal “reset password” function
  - Console-based password recovery; because the user does not have real-time access to the console, password recovery is not possible. It is not possible to boot the password recovery image. The only recourse is to deploy a new Firewall Threat Defense Virtual VM.
- High Availability (active/standby)
- VM import/export
- Gen 2 VM generation on Azure
- Re-sizing the VM after deployment
- Migration or update of the Azure Storage SKU for the OS Disk of the VM from premium to standard SKU and vice versa
- Firewall Device Manager user interface (Versions 6.4 and earlier)

### Azure DDoS Protection Feature

Azure DDoS Protection in Microsoft Azure is an additional feature implemented at the forefront of the Firewall Threat Defense Virtual. In a virtual network, when this feature is enabled it helps to defend applications against common network layer attacks depending on the packet per second of a network’s expected traffic. You can customize this feature based on the network traffic pattern.

For more information about the Azure DDoS Protection feature, see Azure DDoS Protection Standard overview.

### Snort

- If you are observing abnormal behavior such as Snort taking a long time to shut down, or the VM being slow in general or when a certain process is executed, collect logs from the Firewall Threat Defense Virtual and the VM host. Collection of overall CPU usage, memory, I/O usage, and read/write speed logs will help troubleshoot the issues.
- High CPU and I/O usage is observed when Snort is shutting down. If a number of Firewall Threat Defense Virtual instances have been created on a single host with insufficient memory and no dedicated CPU, Snort will take a long time to shut down which will result in the creation of Snort cores.

# How to Manage Secure Firewall Threat Defense Virtual Device

You have two options to manage your Secure Firewall Threat Defense Virtual.

## Secure Firewall Management Center

If you are managing large numbers of devices, or if you want to use the more complex features and configurations that the Firewall Threat Defense allows, use the Firewall Management Center to configure your devices instead of the integrated Firewall Device Manager.

**Important** You cannot use both the Firewall Device Manager and the Firewall Management Center to manage the Firewall Threat Defense device. Once the Firewall Device Manager integrated management is enabled, it won't be possible to use the Firewall Management Center to manage the Firewall Threat Defense device, unless you disable the local management and re-configure the management to use the Firewall Management Center. On the other hand, when you register the Firewall Threat Defense device to the Firewall Management Center, the Firewall Device Manager onboard management service is disabled.

**Caution** Currently, Cisco does not have an option to migrate your Firewall Device Manager configuration to the Firewall Management Center and vice-versa. Take this into consideration when you choose what type of management you configure for the Firewall Threat Defense device.

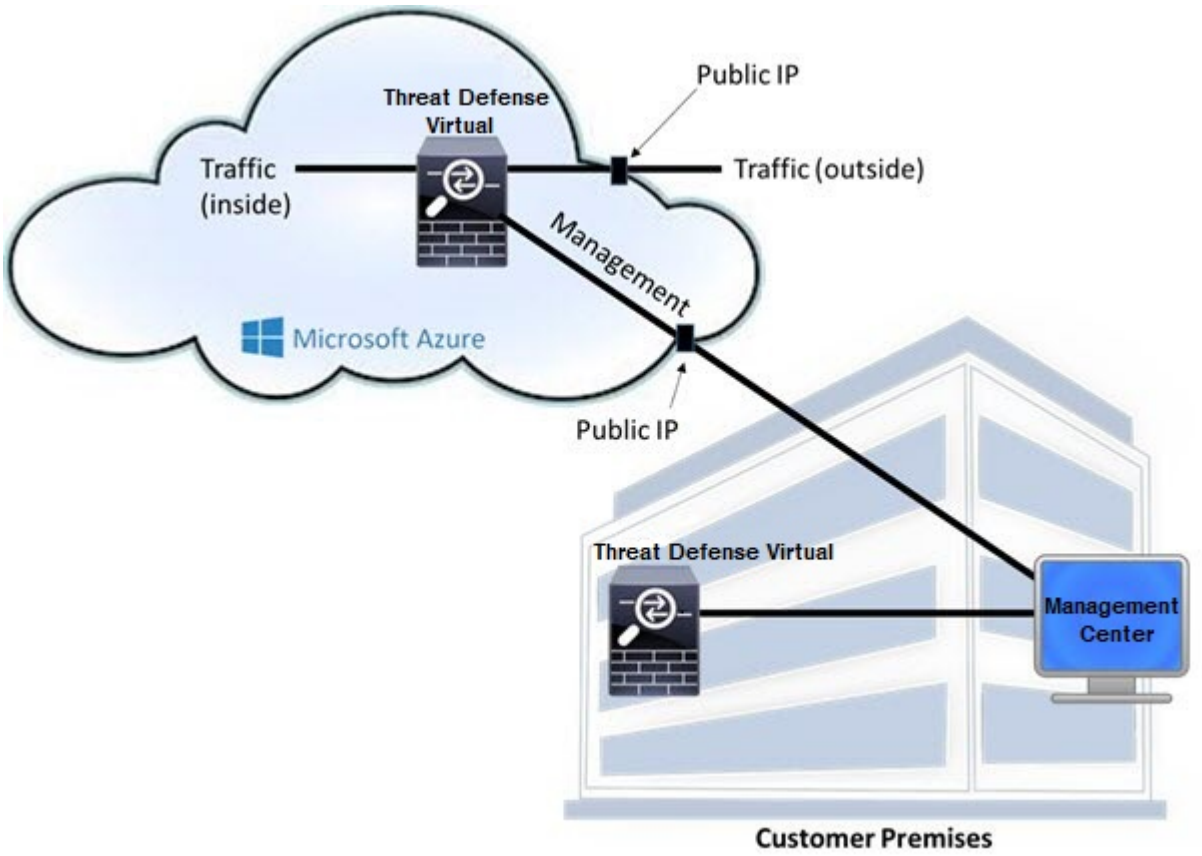
## Secure Firewall Device Manager

The Firewall Device Manager is a web interface included on most Firewall Threat Defense devices. It lets you configure the basic features of the software that are most commonly used for small networks. It is especially designed for networks that include a single device or just a few, where you do not want to use a high-powered multiple-device manager to control a large network with many devices.

**Note** See the Cisco Secure Firewall Threat Defense Compatibility Guide for list of devices that support the Firewall Device Manager.

# Sample Network Topology for the Firewall Threat Defense Virtual on Azure

The following figure shows a typical topology for the Firewall Threat Defense Virtual in Routed Firewall Mode within Azure. The first defined interface is always the Management interface, and only the Management 0/0 and GigabitEthernet0/0 are assigned public IP addresses.




# Resources Created During Deployment

When you deploy the Secure Firewall Threat Defense Virtual in Azure the following resources are created:

- The Firewall Threat Defense Virtual Machine (VM)
- A Resource Group
  - The Firewall Threat Defense Virtual is always deployed into a new Resource Group. However, you can attach it to an existing Virtual Network in another Resource Group.
- Four NICS named *vm name* -Nic0, *vm name* -Nic1, *vm name* -Nic2, *vm name* -Nic3

**Note** Based on the requirement, you can create VNet with IPv4 only or Dual Stack (IPv4 and IPv6 enabled).

These NICs map to the Firewall Threat Defense Virtual interfaces Management, Diagnostic 0/0, GigabitEthernet 0/0, and GigabitEthernet 0/1 respectively.

- A security group named *vm name* -mgmt-SecurityGroup

The security group will be attached to the VM's Nic0, which maps to the Firewall Threat Defense Virtual management interface.

The security group includes rules to allow SSH (TCP port 22) and the management traffic for the Firewall Management Center interface (TCP port 8305). You can modify these values after deployment.

- Public IP addresses (named according to the value you chose during deployment).

  You can assign a public IP address to any interface; see Public IP addresses for Azure's guidelines regarding public IPs, including how to create, change, or delete a public IP address..

- A Virtual Network with four subnets will be created if you choose the New Network option.
- A Routing Table for each subnet (updated if it already exists)

  The tables are named *"subnet name "*-FTDv-RouteTable.

  Each routing table includes routes to the other three subnets with the Firewall Threat Defense Virtual IP address as the next hop. You may chose to add a default route if traffic needs to reach other subnets or the Internet.

- A boot diagnostics file in the selected storage account

  The boot diagnostics file will be in Blobs (binary large objects).

- Two files in the selected storage account under Blobs and container VHDs named *vm name* -disk.vhd and *vm name* -<uuid>.status
- A Storage account (unless you chose an existing storage account)

**Note** When you delete a VM, you must delete each of these resources individually, except for any resources you want to keep.

# Accelerated Networking (AN)

Azure's Accelerated Networking (AN) feature enables single root I/O virtualization (SR-IOV) to a VM, which accelerates networking by allowing VM NICs to bypass the hypervisor and go directly to the PCIe card underneath. AN significantly enhances the throughput performance of the VM and also scales with additional cores (i.e. larger VMs).

AN is disabled by default. Azure supports enabling AN on pre-provisioned virtual machines. You simply have to stop VM in Azure and update the network card property to set the *enableAcceleratedNetworking* parameter to true. See the Microsoft documentation Enable accelerated networking on existing VMs. Then restart the VM.

### Limitations of using ixgbe-vf Interfaces

Be aware of the following limitations when using ixgbe-vf interfaces:

- The guest VM is not allowed to set the VF to promiscuous mode. Because of this, transparent mode is not supported when using ixgbe-vf.

- The guest VM is not allowed to set the MAC address on the VF. Because of this, the MAC address is not transferred during HA like it is done on other Firewall Threat Defense Virtual platforms and with other interface types. HA failover works by transferring the IP address from active to standby.

**Note** This limitation is applicable to the i40e-vf interfaces too.

- The Cisco UCS-B server does not support the ixgbe-vf vNIC.
- In a failover setup, when a paired Firewall Threat Defense Virtual (primary unit) fails, the standby unit takes over as the primary unit role and its interface IP address is updated with a new MAC address of the standby Firewall Threat Defense Virtual unit. Thereafter, the Firewall Threat Defense Virtual sends a gratuitous Address Resolution Protocol (ARP) update to announce the change in MAC address of the interface IP address to other devices on the same network. However, due to incompatibility with these types of interfaces, the gratuitous ARP update is not sent to the global IP address that is defined in the NAT or PAT statements for translating the interface IP address to global IP addresses.

# Azure Routing

Routing in an Azure Virtual Network Subnet is determined by the Subnet's Effective Routing Table. The Effective Routing Table is a combination of built-in system routes and the routes in the User Defined Route (UDR) Table.

**Note** You can view the Effective Routing Table under VM NIC properties.

You can view and edit the User Defined Routing table. When the system routes and the user defined routes are combined to form the Effective Routing Table, the most specific route wins and ties go to the User Defined Routing table. The System Routing Table includes a default route (0.0.0.0/0) IPv4 or [::/0] IPv6 pointing to Azure's Virtual Network Internet Gateway. The System Routing Table also includes specific routes to the other defined subnets with the next-hop pointing to Azure's Virtual Network infrastructure gateway.

To route traffic through the Azure Routing Firewall Threat Defense Virtual, routes must be added/updated in the User Defined Routing table associated with each data subnet. Traffic of interest should be routed by using the Firewall Threat Defense Virtual IP address on that subnet as the next-hop. Also, a default route for 0.0.0.0/0 IPv4 or [::/0] IPv6 can be added with a next hop of the Firewall Threat Defense Virtual IP if needed.

Because of the existing specific routes in the System Routing Table, you must add specific routes to the User Defined Routing table to point to the Firewall Threat Defense Virtual as the next-hop. Otherwise, a default route in the User Defined table would lose to the more specific route in the System Routing Table and traffic would bypass the Firewall Threat Defense Virtual.

# Routing Configuration for VMs in the Virtual Network

Routing in the Azure Virtual Network depends on the Effective Routing Table and not the particular gateway settings on the clients. Clients running in a Virtual Network may be given routes by DHCP that are the .1 address on their respective subnets. This is a place holder and serves only to get the packet to the Virtual

Network's infrastructure virtual gateway. Once a packet leaves the VM it is routed according to the Effective Routing Table (as modified by the User Defined Table). The Effective Routing Table determines the next hop regardless of whether a client has a gateway configured as .1 or as the Firewall Threat Defense Virtual address.

Azure VM ARP tables will show the same MAC address (1234.5678.9abc) for all known hosts. This ensures that all packets leaving an Azure VM will reach the Azure gateway where the Effective Routing Table will be used to determine the path of the packet.

# IP Addresses

The following information applies to IP addresses in Azure:

- The first NIC on the Firewall Threat Defense Virtual (which maps to Management) is given a private IP address in the subnet to which it is attached.

  A public IP address may be associated with this private IP address and the Azure Internet gateway handles the NAT translations.

  You can associate a public IP address with a data interface (GigabitEthernet0/0, for example) after the Firewall Threat Defense Virtual has been deployed; see Public IP addresses for Azure's guidelines regarding public IPs, including how to create, change, or delete a public IP address.

- You can enable **IP Forwarding** in the network interface attached to a Firewall Threat Defense Virtual appliance in a Virtual Machine Scale Set (VMSS). If network traffic is not destined to any of the configured IP addresses in the network interface, then enabling this option forwards such network traffic to other IP addresses other than the IP addresses configured in the virtual machine. See Azure documentation on how to enable IP Forwarding in the network interface - Enable or disable IP forwarding.

- Public IP addresses (IPv4 and IPv6) are dynamic and may change during an Azure stop/start cycle. However, they are persistent during Azure restart and during Firewall Threat Defense Virtual reload. See IPv6 Public IP Address Standards.

- Public IP addresses that are static do not change until you change them in Azure.

- Firewall Threat Defense Virtual interfaces may use DHCP to set their IP addresses. The Azure infrastructure ensures that the Firewall Threat Defense Virtual interfaces are assigned the IP addresses set in Azure.

# Deploy the Firewall Threat Defense Virtual

You can deploy the Firewall Threat Defense Virtual in Azure using templates. Cisco provides two kinds of templates:

- **Solution Template in the Azure Marketplace**—Use the solution template available in the Azure Marketplace to deploy the Firewall Threat Defense Virtual using the Azure portal. You can use an existing resource group and storage account (or create them new) to deploy the virtual appliance. To use the solution template, see Deploy from the Azure Marketplace Using the Solution Template, on page 162.

- **Custom Template using a Managed Image from a VHD (available from** https://software.cisco.com/download/home )—In addition to the Marketplace-based deployment, Cisco provides a compressed virtual hard disk (VHD) that you can upload to Azure to simplify the process of deploying the Firewall

Threat Defense Virtual in Azure. Using a Managed Image and two JSON files (a Template file and a Parameters File), you can deploy and provision all the resources for the Firewall Threat Defense Virtual in a single, coordinated operation. To use the custom template, see Deploy from Azure Using a VHD and Resource Template, on page 165.



**Note** While searching for Cisco offers in Marketplace, you may find two different offers with similar names, but different offer types, Application Offer and Virtual Machine Offer.

For marketplace deployments, use ONLY the Application Offers.

Virtual Machine offer (may be visible) with VMSR (Virtual Machine Software Reservations) plan in marketplace. These are specific Multiparty Private Offer plans specifically for channel/resale and should be ignored for regular deployments.

**Application Offers available in Marketplace:**

- Cisco Secure Firewall Threat Defence Virtual - BYOL and PAYG
- Cisco Secure Firewall Threat Defence for Azure Virtual WAN

# Deploy from the Azure Marketplace Using the Solution Template

The following instructions show you how to deploy the solution template for the Firewall Threat Defense Virtual that is available in the Azure Marketplace. This is a top-level list of steps to set up the Firewall Threat Defense Virtual in the Microsoft Azure environment. For detailed steps about the Azure setup, see Getting Started with Azure.

When you deploy the Firewall Threat Defense Virtual in Azure, it automatically generates various configurations, such as resources, public IP addresses (IPv4 and IPv6), and route tables. You can further manage these configurations after deployment. For example, you may want to change the Idle Timeout value from the default, which is a low timeout.



**Note** To use the customizable ARM templates available in the GitHub repository, see Deploy from Azure Using a VHD and Resource Template, on page 165.

## Procedure

**Step 1** Log into the Azure Resource Manager (ARM) portal.

The Azure portal shows virtual elements associated with the current account and subscription regardless of data center location.

**Step 2** Choose **Azure Marketplace** > **Virtual Machines**.

**Step 3** Search Marketplace for “Cisco Firepower NGFW Virtual (Firewall Threat Defense Virtual)”, choose the offering, and click **Create**.

**Step 4** Configure the basic settings.

a) Enter a name for the virtual machine. This name should be unique within your Azure subscription.

**Important**
If you use an existing name the deployment will fail.

b) Choose your licensing method, either **BYOL** or **PAYG**.

Choose **BYOL** (Bring Your Own License) to use a Cisco Smart License Account.

Choose **PAYG** (Pay As You Go) licensing to use a usage-based billing model without having to purchase Cisco Smart Licensing.

**Important**
You can only use **PAYG** when you manage the Firewall Threat Defense Virtual using the Firewall Management Center.

c) Enter a username for the Firewall Threat Defense Virtual administrator.

**Note**
The name “admin” is reserved in Azure and cannot be used.

d) Choose an authentication type, either password or SSH key.

If you choose password, enter a password and confirm.

If you choose SSH key, specify the RSA public key of the remote peer.

e) Create a password to use with the **Admin** user account when you log in to configure the Firewall Threat Defense Virtual.

f) Select the management center you want to register the Firewall Threat Defense Virtual from the **FTDv Management** drop-down list.

If you are choosing **FMC: Firepower Management Center** as the management center for your device, using the following option you can configure the management center for your device.

- Click **Yes** to enter the **FMC registration information**.
  1. Enter the **FMC IP** address.
  2. Enter the **FMC Registration Key** for registering the Threat Defense Virtual instances.
  3. [Optional] Enter the management center NAT ID that is used during instance registration.

g) If you are using the virtual machine you are deploying as a cluster, then click **Yes (provide day0 cluster configuration)** to create and enter the basic day0 configuration details.

- Enter the day0 configuration details in the **Day0 cluster configuration** field.

  For information on creating day0 configuration for Azure, see Create the Day0 Configuration for Azure in the Deploy a Threat Defense Virtual Cluster on Azure guide.

  **Note**
  You can only configure the partial day0 config (cluster config): "Cluster": {...} OR "run_config": [...] details.

h) Choose your subscription.

i) Create a new Resource Group.

The Firewall Threat Defense Virtual should be deployed into a new Resource Group. The option to deploy into an existing Resource Group only works if that existing Resource Group is empty.

However, you can attach the Firewall Threat Defense Virtual to an existing Virtual Network in another Resource Group when configuring the network options in later steps.

j) Select geographical location. This should be the same for all resources used in this deployment (for example: Firewall Threat Defense Virtual, Network, storage accounts).

k) Click **OK**.

**Step 5** Configure the Firewall Threat Defense Virtual settings.

a) Choose the virtual machine size.

b) Choose a storage account.

**Note**
You can use an existing storage account or create a new one. The storage account name can only contain lowercase letters and numbers.

c) Choose a public IP address.

You can choose a public IP address available for the selected subscription and location, or click **Create new**.

When you create a new public IP address, you get one from the block of IP addresses that Microsoft owns, so you can't choose a specific one. The maximum number of public IP addresses you can assign to an interface is based on your Azure subscription.

**Important**
Azure creates a dynamic public IP address by default. The public IP may change when the VM is stopped and restarted. If you prefer a fixed IP address, you should create a static address. You can also modify the public IP address after deployment and change it from a dynamic to a static address.

In case the VM needs to assign the public IPv6 address, refer to the IPv6 standards IPv6 Public IP Address Standards.

d) Add the DNS label.

**Note**
The fully qualified domain name will be your DNS label plus the Azure URL: <dnslabel>.<location>.cloudapp.azure.com

e) Choose a virtual network.

You can choose an existing Azure Virtual Network (VNet) or create a new one and enter the IP address space for the VNet. By default, the Classless Inter-Domain Routing (CIDR) IP address is 10.0.0.0/16.

If the Virtual Machine is required for the IPv6 addressing, you need to enable it in the virtual network. Example: By default, the CIDR IPv6 address is [ace:cab:deca::/48].

**Note**
Virtual Networks, Subnets, Interface, etc., cannot be created by using IPv6 alone. The IPv4 is used by default, and IPv6 can be enabled along with it. For more information on IPv6, see Azure IPv6 Overview

f) Configure four subnets for the Firewall Threat Defense Virtual network interfaces:

- **FTDv Management** interface, attached to Nic0 in Azure, the "First subnet"
- **FTDv Diagnostic** interface, attached to Nic1 in Azure, the "Second subnet"
- **FTDv Outside** interface, attached to Nic2 in Azure, the "Third subnet"

- **FTDv Inside** interface, attached to Nic3 in Azure, the “Fourth subnet”

**Note**
For the above subnets, if we require IPv6 configuration while creating the subnets, select the IPv6 option and configure IPv6 subnets for the interface.

g) Provide **Public inbound ports (mgmt.interface)** input to indicate whether any ports are to be opened for public or not. By default, **None** is selected.

- Click **None** to create and attach a network security group with Azure's default security rule to the management interface. Selecting this option allows traffic from sources in the same virtual network and from the Azure load balancer.
- Click **Allow selected ports** to view and choose the inbound ports to be opened for acess by the internet. Choose any of the following ports from the **Select Inbound Ports** drop-down list. By default, **SSH (22)** is selected.
  - SSH (22)
  - SFTunnel (8305)
  - HTTPs (443)

h) Click **OK**.

**Step 6** View the configuration summary, and then click **OK**.

**Step 7** View the terms of use and then click **Purchase**.

Deployment times vary in Azure. Wait until Azure reports that the Firewall Threat Defense Virtual VM is running.

### What to do next

Your next steps depend on what management mode you chose.

- If you chose **No** for **Enable Local Manager**, you'll use the Secure Firewall Management Center to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409.
- If you chose **Yes** for **Enable Local Manager**, you'll use the integrated Secure Firewall Device Manager to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Device Manager, on page 427.

See How to Manage Secure Firewall Threat Defense Virtual Device, on page 1 for an overview of how to choose your management option.

# Deploy from Azure Using a VHD and Resource Template

You can create your own custom Firewall Threat Defense Virtual images using a compressed VHD image available from Cisco. To deploy using a VHD image, you must upload the VHD image to your Azure storage account. Then, you can create a managed image using the uploaded disk image and an Azure Resource Manager template. Azure templates are JSON files that contain resource descriptions and parameter definitions.

### Before you begin

- You need the JSON template and corresponding JSON parameter file for your Firewall Threat Defense Virtual template deployment. You can download these files from the Github repository.
- This procedure requires an existing Linux VM in Azure. We recommend that you use a temporary Linux VM (such as Ubuntu 16.04) to upload the compressed VHD image to Azure. This image will require about 50GB of storage when unzipped. Also, your upload time to Azure storage is faster from a Linux VM in Azure.

  If you need to create a VM, use one of the following methods:

  - Create a Linux virtual machine with the Azure CLI
  - Create a Linux virtual machine with the Azure portal

- In your Azure subscription, you should have a storage account available in the location in which you want to deploy the Firewall Threat Defense Virtual.

## Procedure

**Step 1** Download the Firewall Threat Defense Virtual compressed VHD image from the Cisco Download Software page:

a) Navigate to **Products** > **Security** > **Firewalls** > **Next-Generation Firewalls (NGFW)** > **Secure Firewall Threat Defense Virtual**.

b) Click **Firepower Threat Defense Software**.

Follow the instructions for downloading the image.

For example, Cisco_Secure_Firewall_Threat_Defense_Virtual-X.X.X-xxx.vhd.bz2

**Step 2** Copy the compressed VHD image to your Linux VM in Azure.

There are many options that you can use to move files up to Azure and down from Azure. This example shows SCP or secure copy:

```
# scp /username@remotehost.com/dir/Cisco_Secure_Firewall_Threat_Defense_Virtual-7.1.0-92.vhd.bz2
<linux-ip>
```

**Step 3** Log in to the Linux VM in Azure and navigate to the directory where you copied the compressed VHD image.

**Step 4** Unzip the Firewall Threat Defense Virtual VHD image.

There are many options that you can use to unzip or decompress files. This example shows the Bzip2 utility, but there are also Windows-based utilities that would work.

```
# bunzip2 Cisco_Firepower_Threat_Defense_Virtual-7.1.0-92.vhd.bz2
```

**Step 5** Upload the VHD to a container in your Azure storage account. You can use an existing storage account or create a new one. The storage account name can only contain lowercase letters and numbers.

There are many options that you can use to upload a VHD to your storage account, including AzCopy, Azure Storage Copy Blob API, Azure Storage Explorer, Azure CLI, or the Azure Portal. We do not recommend using the Azure Portal for a file as large as the Firewall Threat Defense Virtual VHD.

The following example shows the syntax using Azure CLI:

```
azure storage blob upload \
       --file <unzipped vhd> \
       --account-name <azure storage account> \
       --account-key yX7txxxxxxxx1dnQ== \
       --container <container> \
       --blob <desired vhd name in azure> \
       --blobtype page
```

**Step 6** Create a Managed Image from the VHD:

a) In the Azure Portal, select **Images**.
b) Click **Add** to create a new image.
c) Provide the following information:

- **Subscription**—Choose a subscription from the drop-down list.
- **Resource group**—Choose an existing resource group or create a new one.
- **Name**—Enter a user-defined name for the managed image.
- **Region**—Choose the region in which the VM Is deployed.
- **OS type**—Choose **Linux** as the OS type.
- **VM generation**—Choose **Gen 1**.

  **Note**
  **Gen 2** is not supported.

- **Storage blob**—Browse to the storage account to select the uploaded VHD.
- **Account type**—As per your requirement, choose Standard HDD, Standard SSD, or Premium SSD, from the drop-down list.

  When you select the VM size planned for deployment of this image, ensure that the VM size supports the selected account type.
- **Host caching**—Choose Read/write from the drop-down list.
- **Data disks**—Leave at default; don't add a data disk.

d) Click **Create**.

Wait for the **Successfully created image** message under the **Notifications** tab.

**Note**
Once the Managed Image has been created, the uploaded VHD and upload Storage Account can be removed.

**Step 7** Acquire the Resource ID of the newly created Managed Image.

Internally, Azure associates every resource with a Resource ID. You'll need the Resource ID when you deploy new Firewall Threat Defense Virtual firewalls from this managed image.

a) In the Azure Portal, select **Images**.
b) Select the managed image created in the previous step.
c) Click **Overview** to view the image properties.
d) Copy the **Resource ID** to the clipboard.

The **Resource ID** takes the form of:

**/subscriptions/<subscription-id>/resourceGroups/<resourceGroup>/providers/Microsoft.Compute/<container>/<vhdname>**

**Step 8** Build a Firewall Threat Defense Virtual firewall using the managed image and a resource template:

a) Select **New**, and search for **Template Deployment** until you can select it from the options.
b) Select **Create**.
c) Select **Build your own template in the editor**.

You have a blank template that is available for customizing. See Github for the template files.

d) Paste your customized JSON template code into the window, and then click **Save**.
e) Choose a **Subscription** from the drop-down list.
f) Choose an existing **Resource group** or create a new one.
g) Choose a **Location** from the drop-down list.
h) Paste the Managed Image **Resource ID** from the previous step into the **Vm Managed Image Id** field.

**Step 9** Click **Edit parameters** at the top of the **Custom deployment** page. You have a parameters template that is available for customizing.

a) Click **Load file** and browse to the customized Firewall Threat Defense Virtual parameter file. See Github for the template parameters.
b) Paste your customized JSON parameters code into the window, and then click **Save**.

**Step 10** Review the Custom deployment details. Make sure that the information in **Basics** and **Settings** matches your expected deployment configuration, including the **Resource ID**.

**Step 11** Review the Terms and Conditions, and check the **I agree to the terms and conditions stated above** check box.

**Step 12** Click **Purchase** to deploy a Firewall Threat Defense Virtual firewall using the managed image and a custom template.

If there are no conflicts in your template and parameter files, you should have a successful deployment.

The Managed Image is available for multiple deployments within the same subscription and region.

### What to do next

- Update the Firewall Threat Defense Virtual's IP configuration in Azure.

# About Deployment of Threat Defense Virtual without Diagnostic Interface on Azure

On Secure Firewall version 7.3 and earlier, the Threat Defense Virtual is deployed with a minimum of 4 interfaces –1 management, 1 diagnostic, and 2 data interfaces.

From Secure Firewall version 7.4.1, you can remove the diagnostic interface and deploy the Threat Defense Virtual with a minimum of 3 interfaces – 1 management, and 2 data interfaces. This feature enables deployment of Threat Defense Virtual with an additional data interface on the same instance type. For example, on a Standard D4_v2 VM instance, instead of deploying Threat Defense Virtual with 1 management, 1 diagnostic, and 6 data interfaces, you can now deploy Threat Defense Virtual with 1 management, and 7 data interfaces.

From Secure Firewall version 7.4.1, we recommend that you deploy the Threat Defense Virtual on Azure without the diagnostic interface.

This feature is supported only on new deployments of Threat Defense Virtual instances on Azure.



**Note** As the maximum number of supported interfaces is 8, you can add up to 5 more interfaces after deploying the Threat Defense Virtual to have a maximum of 8 interfaces.

# Guidelines and Limitations for Deployment of Threat Defense Virtual without Diagnostic Interface on Azure

- When the diagnostic interface is removed, syslog and SNMP is supported using either the Firewall Threat Defense Virtual management or the data interface instead of the diagnostic interface.
- Clustering and auto scale is supported with this deployment.
- Grouping of Firewall Threat Defense Virtual instances with diagnostic interface port and Firewall Threat Defense Virtual instances without diagnostic interface port is not supported.

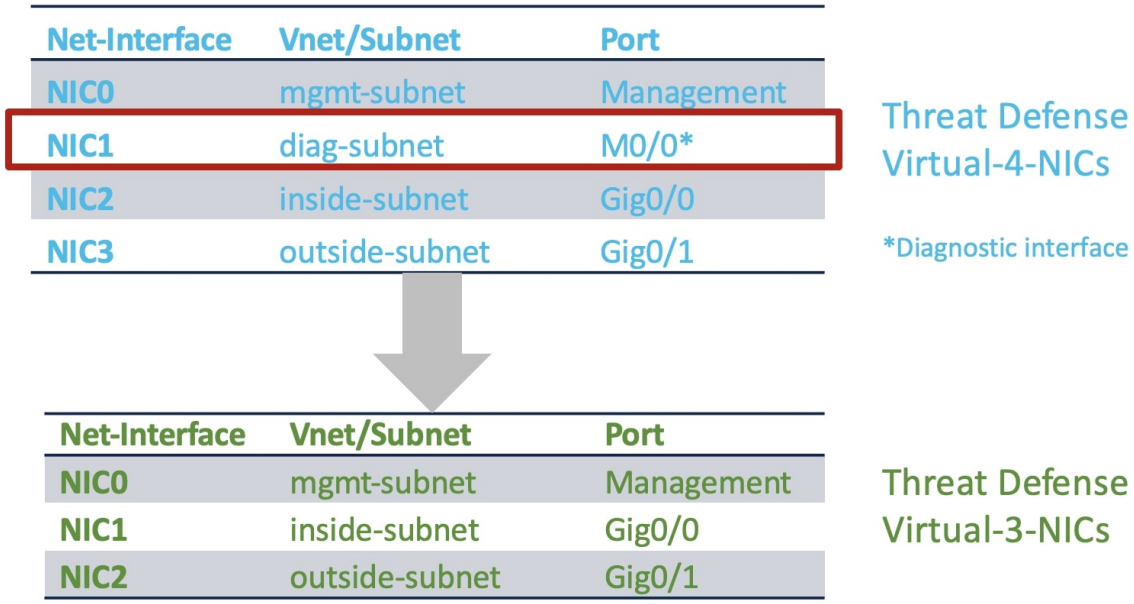


**Note** The grouping of Firewall Threat Defense Virtual instances here refers to the grouping of the instances in the Virtual Machine Scale Set (VMSS) on Azure. This does not pertain to the grouping of Firewall Threat Defense Virtual instances on the Management Center Virtual.

- CMI is not supported.

# NIC Mapping to Data Interfaces for Deployment of Threat Defense Virtual without Diagnostic Interface on Azure

The NIC mapping to data interfaces for deployment of Threat Defense Virtual without the diagnostic interface on Azure is given below.




# Deploy Threat Defense Virtual without Diagnostic Interface on Azure

Perform the steps given below to deploy Firewall Threat Defense Virtual without diagnostic interface.

**Procedure**

**Step 1** Depending on your deployment option, you can enable this feature by using one of the methods given below.

- **Solution template in the Azure Marketplace** – On the Azure console, search for **`Cisco Secure Firewall Threat Defense Virtual - BYOL and PAYG`** and click **Create**. In the **Basics** window, enter the required information and choose **7.4.x** from the **Software version** drop-down list. Choose the **No** button next to **Attach diagnostic interface**. By default, **No** is selected.

  For the complete procedure to deploy Threat defense virtual on Azure using the solution template in the Azure marketplace, see Deploy from the Azure Marketplace Using the Solution Template.

- **Custom Template using a Managed Image from a VHD** – Go to **Virtual machines** > **+ Create** > **Azure Virtual Machine** > **Advanced** window, and enter a day-0 configuration script that includes the key-value pair **`Diagnostic: OFF`** in the **Custom data** field. A sample day-0 configuration script that you can enter in the **Custom data** field is given below.

```
{
"AdminPassword": "E28@2OiUrhx!",
"Hostname": "ciscothreatdefensevirtual",
"FirewallMode": "routed",
"ManageLocally": "No",
"Diagnostic": "OFF"
}
```

**Note**
The key value pair, "Diagnostic": "ON/OFF", is case-sensitive.

You can also modify the script in the **Customdata** field in the ARM template that is used for fresh deployment. By default, the key-value pair is set to **`Diagnostic: ON`** and the diagnostic interface comes up. When the key-value pair is set to **`Diagnostic: OFF`**, the deployment comes up without the diagnostic interface.

For the complete procedure to deploy Threat defense virtual on Azure with a custom template using a managed image from VHD, see Deploy from Azure Using a VHD and Resource Template.

**Step 2** Attach the required minimum number of NICs - 3. For more information on attaching interfaces on Azure, see Add network interfaces to or remove network interfaces from virtual machines on Azure.

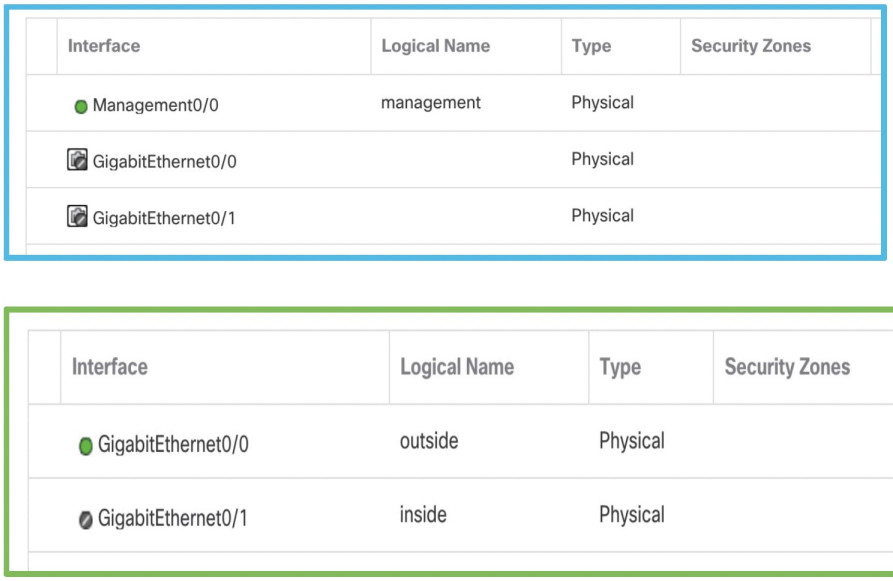
***Figure 5: Network Interface Attachment on Azure***




For more information on interfaces, see Interface Overview.

**Step 3** (Optional) Use the **show interface ip brief** command on the console to display interface details. You can also view interface details on the Management Center Virtual as given below

The interfaces are displayed on the Management Center Virtual as given below.




# Deploy the Azure Marketplace offers in the restricted Azure Private Marketplace environment

This applies only for the Azure Private Marketplace users. If you are using Azure Private Marketplace, then ensure that both Application Offers and required Virtual Machine Offers (hidden) are enabled for the user in respective private marketplace.

**Virtual Machine Offers and Plans (hidden):**

- Publisher ID: **cisco**
- Cisco Secure Firewall Threat Defense Virtual VM Offers (used for both the Cisco Secure Firewall Threat Defense Virtual Application offers)
  - Offer ID: **cisco-ftdv**
  - BYOL Plan ID: **ftdv-azure-byol**
  - PAYG Plan ID: **ftdv-azure-payg**

When user deploys the visible application offer from Marketplace, based on user selection of PAYG or BYOL licensing corresponding image from the VM offer plan is referenced and deployed.

Therefore, for the deployment to work, both Application and VM offers needs to be enabled/available on the Private Marketplace for the Azure tenant/subscription.

Refer the Azure documentation for enabling these application and VM offers in private marketplaces.

- Govern and control using private Azure Marketplace
- Add an offer to a private marketplace

- Set-AzMarketplacePrivateStoreOffer

Application offers are easily enabled via Azure UI as they are visible in the marketplace.

In order to enable hidden virtual machine offers in private marketplace, you might have to rely on CLI commands (at the time of this doc creation only CLI way is possible).

**Sample command:**

Cisco Secure Firewall Threat Defense Virtual BYOL plan can be enabled using similar sample command given below:

```
$Params = @{
  privateStoreId = '<private-store-id>'
  offerId = '<publisher-id>.<vm-offer-id>'
  SpecificPlanIdsLimitation =@('<plan-id-under-vm-offer>')
}
Set-AzMarketplacePrivateStoreOffer @Params

$Params = @{
  privateStoreId = '<private-store-id>'
  offerId = 'cisco.cisco-ftdv'
  SpecificPlanIdsLimitation =@('ftdv-azure-byol')
}
Set-AzMarketplacePrivateStoreOffer @Params
```

**Note** The sample command is only for reference, check Azure documentation for more details.

**Reference Error message**

```
{
  "code": "MarketplacePurchaseEligibilityFailed",
  "details": [
    {
      "code": "BadRequest",
     "message": "Offer with PublisherId: 'cisco', OfferId: 'cisco-XXXX' cannot be purchased
 due to validation errors. For more information see details.
Correlation Id: 'XXXXX`
This plan is not available for purchase because it needs to be added to your tenant's Private
 Marketplace. Contact your admin to request adding the plan.
Link to plan: <URL>.
Plan: '<PLAN NAME>'(planId=<VM-OFFER-PLAN-ID>),
Offer: <OFFER_NAME>, Publisher: 'Cisco Systems, Inc.'(publisherId='cisco').
…
…
    }
  ],
  "message": "Marketplace purchase eligibilty check returned errors. See inner errors for
details. "
}
```

User may run into the above error while deploying the Marketplace offer. To resolve this, both Application and VM offers need to be enabled/available on the Azure tenant/subscription.

# Upgrade Scenarios

You can upgrade a Firewall Threat Defense Virtual instance as per the scenarios given below.

- All Secure firewall versions – You can upgrade a Firewall Threat Defense Virtual instance deployed with a diagnostic interface to a Firewall Threat Defense Virtual instance with a diagnostic interface.
- Secure Firewall version 7.4 and later – You can upgrade a Firewall Threat Defense Virtual instance deployed without a diagnostic interface to a Firewall Threat Defense Virtual instance without a diagnostic interface.

The upgrade scenarios given below are not supported.

- All Secure firewall versions – You cannot upgrade a Firewall Threat Defense Virtual instance deployed with a diagnostic interface to a Firewall Threat Defense Virtual instance without a diagnostic interface.
- Secure Firewall version 7.4.1 and later – You cannot upgrade a Firewall Threat Defense Virtual instance deployed without a diagnostic interface to a Firewall Threat Defense Virtual instance with a diagnostic interface.



**Note** The number and order of the NICs is maintained after upgrading.

# Deployment of Threat Defense Virtual Cluster or Auto Scale Solution without Diagnostic Interface

To perform a new deployment of a Firewall Threat Defense Virtual cluster or an auto scale solution consisting of Firewall Threat Defense Virtual instances without the diagnostic interface, ensure that the key-value pair, **Diagnostic: OFF/ON**, is set to **OFF** in the day-0 configuration script.

# Troubleshooting

If the diagnostic interface is not removed when the Firewall Threat Defense Virtual is deployed, check if the key-value pair, **Diagnostic: OFF/ON**, has been set to **OFF** in the day-0 configuration script.

# Auto Scale Solution for the Threat Defense Virtual on Azure

## Overview

The auto scale solution enables allocation of resources to match performance requirements and reduce costs. If the demand for resources increases, the system ensures that resources are allocated as required. If the demand for resources decreases, resources are deallocated to reduce costs.

The Firewall Threat Defense Virtual auto scale for Azure is a complete serverless implementation which makes use of serverless infrastructure provided by Azure (Logic App, Azure Functions, Load Balancers, Security Groups, Virtual Machine Scale Set, etc.).

Some of the key features of the Firewall Threat Defense Virtual auto scale for Azure implementation include:

- Azure Resource Manager (ARM) template-based deployment.

- Support for scaling metrics based on CPU and memory (RAM).

**Note** See Auto Scale Logic, on page 213 for more information.

- Support for Firewall Threat Defense Virtual deployment and multi-availability zones.
- Completely automated Firewall Threat Defense Virtual instance registration and de-registration with the Firewall Management Center.
- NAT policy, Access Policy, and Routes automatically applied to scaled-out Firewall Threat Defense Virtual instances.
- Support for Load Balancers and multi-availability zones.
- Support for enabling and disabling the auto scale feature.
- Works only with the Firewall Management Center; the Firewall Device Manager is not supported.
- Support to deploy the Firewall Threat Defense Virtual with PAYG or BYOL licensing mode. PAYG is applicable only for Firewall Threat Defense Virtual software version 6.5 and onwards. See Supported Software Platforms, on page 175.
- Cisco provides an auto scale for Azure deployment package to facilitate the deployment.

The Firewall Threat Defense Virtual auto scale solution on Azure supports two types of use cases configured using different topologies:

- Auto scale using Sandwich Topology – The Firewall Threat Defense Virtual scale set is sandwiched between an Azure Internal load balancer (ILB) and an Azure External load balancer (ELB).
- Auto scale with Azure Gateway load balancer (GWLB) – The Azure GWLB is integrated with Secure Firewall, public load balancer, and internal servers - to simplify deployment, management, and scaling of firewalls.

### Supported Software Platforms

The Firewall Threat Defense Virtual auto scale solution is applicable to the Firewall Threat Defense Virtual managed by the Firewall Management Center, and is software version agnostic. The Cisco Secure Firewall Threat Defense Compatibility Guide provides software and hardware compatibility, including operating system and hosting environment requirements.

- The Firewall Management Centers: Virtual table lists compatibility and virtual hosting environment requirements for the Firewall Management Center Virtual.
- The Firewall Threat Defense Virtual Compatibility table lists compatibility and virtual hosting environment requirements for the Firewall Threat Defense Virtual on Azure.

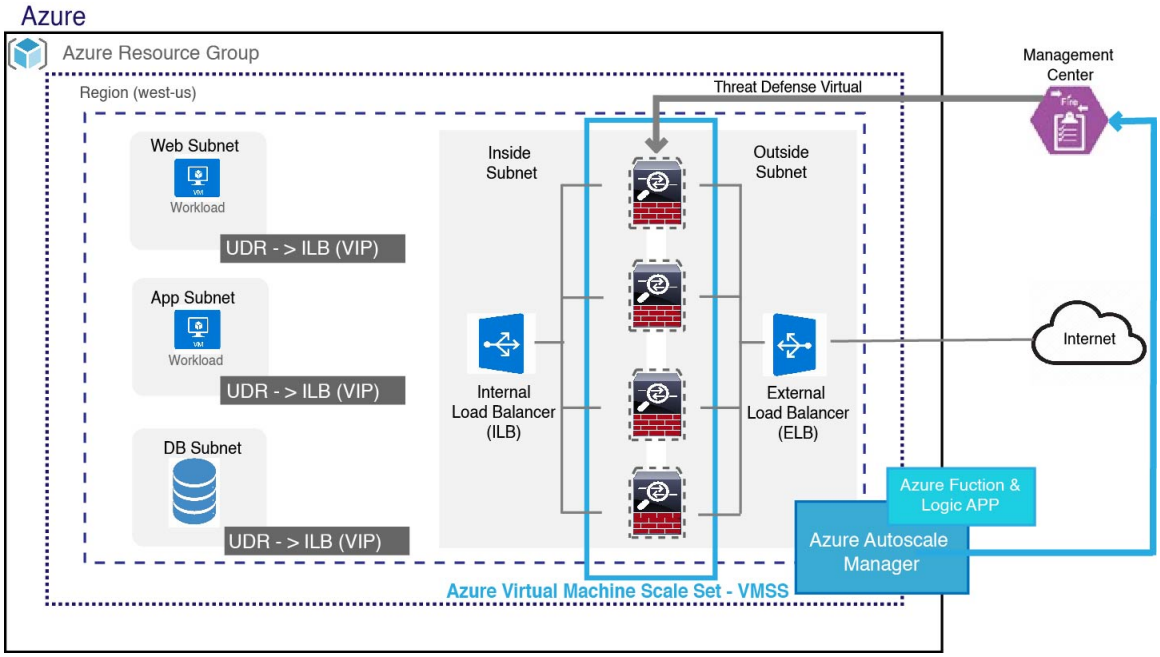
**Note** For purposes of deploying the Azure auto scale solution, the minimum supported version for the Firewall Threat Defense Virtual on Azure is Version 6.4.

# Auto Scale using Sandwich Topology Use Case

The Firewall Threat Defense Virtual auto scale for Azure is an automated horizontal scaling solution that positions the Firewall Threat Defense Virtual scale set sandwiched between an Azure Internal load balancer (ILB) and an Azure External load balancer (ELB).

- The ELB distributes traffic from the Internet to Firewall Threat Defense Virtual instances in the scale set; the firewall then forwards traffic to application.
- The ILB distributes outbound Internet traffic from an application to Firewall Threat Defense Virtual instances in the scale set; the firewall then forwards traffic to Internet.
- A network packet will never pass through both (internal & external) load balancers in a single connection.
- The number of Firewall Threat Defense Virtual instances in the scale set will be scaled and configured automatically based on load conditions.

***Figure 6: Firewall Threat Defense Virtual Auto Scale using Sandwich Topology Use Case Diagram***




# Auto Scale with Azure Gateway Load Balancer Use Case

The Azure Gateway Load Balancer (GWLB) ensures that internet traffic to and from an Azure VM, such as an application server, is inspected by Secure Firewall without requiring any routing changes. This integration of the Azure GWLB with Secure Firewall simplifies deployment, management, and scaling of firewalls. This integration also reduces operational complexity and provides a single entry and exit point for traffic at the firewall. The applications and infrastructure can maintain visibility of source IP address, which is critical in some environments.

In the Azure GWLB Auto Scale use case, the Firewall Threat Defense Virtual uses only two interfaces: Management and one data interface.

**Note**

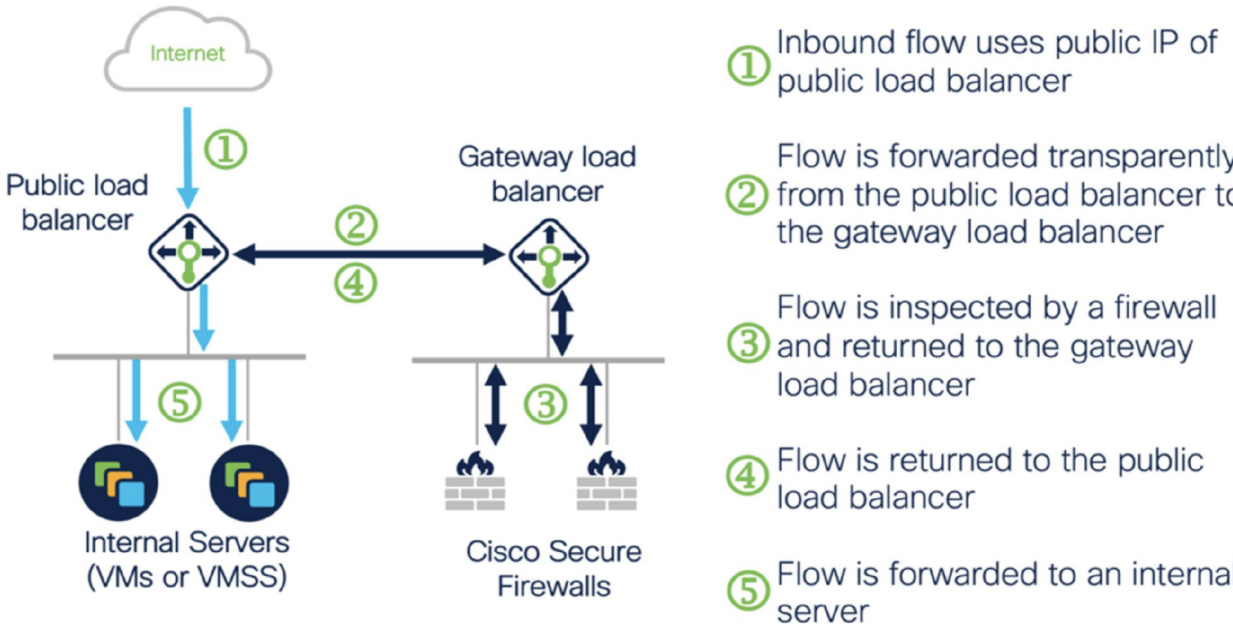
- Network Address Translation (NAT) is not required if you are deploying the Azure GWLB.
- Only IPv4 is supported.

### Licensing

Both PAYG and BYOL are supported.

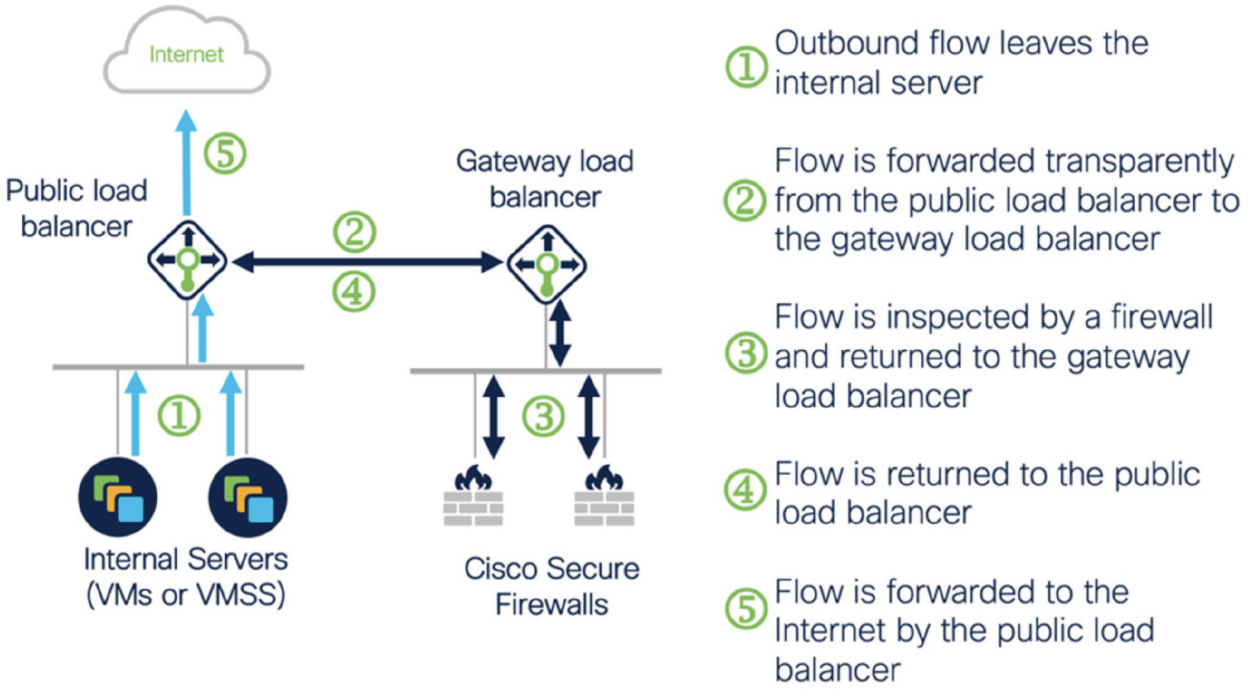
### Inbound Traffic Use Case and Topology

The following diagram displays the traffic flow for inbound traffic.




### Outbound Traffic Use Case and Topology

The following diagram displays the traffic flow for outbound traffic.

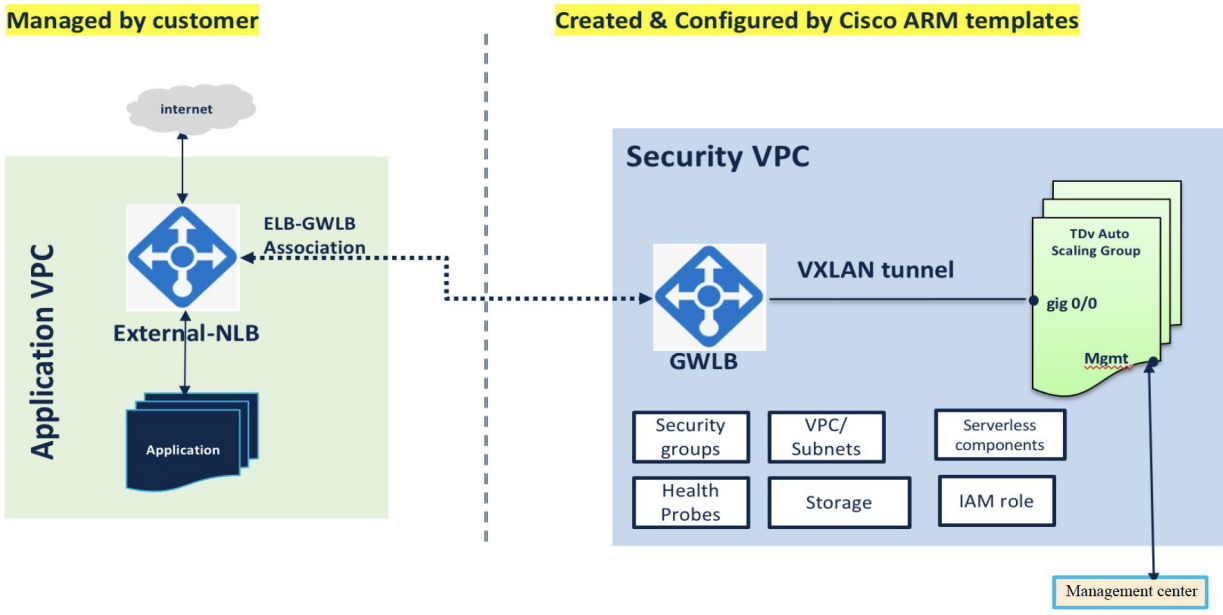



**Note** To deploy and configure the management center, see the procedures in the Cisco Secure Firewall Management Center Device Configuration Guide. Use the deployed management center to manage the threat defense virtual instances.

### Traffic Flow between the Application VNet and Security VNet

In the diagram shown below, traffic is redirected from the existing topology to the firewalls for inspection by the external load balancer. The traffic is then routed to the newly created GWLB. Any traffic that is routed to the ELB is forwarded to the GWLB.

The GWLB then forwards the VXLAN-encapsulated traffic to a Firewall Threat Defense Virtual instance. You have to create two Firewall Threat Defense Virtual associations as the GWLB uses two separate VXLAN tunnels for ingress and egress traffic. The Firewall Threat Defense Virtual decapsulates the VXLAN-encapsulated traffic, inspects it, and routes the traffic to the GWLB. The GWLB then forwards the traffic to the ELB.




# Scope

This document covers the detailed procedures to deploy the serverless components for the Firewall Threat Defense Virtual auto scale for Azure solution and the auto scale with Azure GWLB solution.

**Important**

- Read the entire document before you begin your deployment.
- Make sure the prerequisites are met before you start deployment.
- Make sure you follow the steps and order of execution as described herein.

# Download the Deployment Package

The Firewall Threat Defense Virtual auto scale for Azure solution using sandwich topology is an Azure Resource Manager (ARM) template-based deployment which makes use of the serverless infrastructure provided by Azure (Logic App, Azure Functions, Load Balancers, Virtual Machine Scale Set, and so on).

The Firewall Threat Defense Virtual auto scale with Azure GWLB solution is an ARM template-based deployment that creates the GWLB, networking infrastructure, threat defense virtual auto scaling group, serverless components, and other required resources.

The deployment procedure for both the solutions are similar.

Download the files required to launch the Firewall Threat Defense Virtual auto scale for Azure solution. Deployment scripts and templates for your version are available in the GitHub repository.

**Attention** Note that Cisco-provided deployment scripts and templates for auto scale are provided as open source examples, and are not covered within the regular Cisco TAC support scope. Check GitHub regularly for updates and ReadMe instructions.

See Build Azure Functions from Source Code, on page 217 for instructions on how to build the *ASM_Function.zip* package.

# Auto Scale Solution Components

The following components make up the Firewall Threat Defense Virtual auto scale for Azure solution.

### Azure Functions (Function App)

The Function App is a set of Azure functions. The basic functionality includes:

- Communicate/Probe Azure metrics periodically.
- Monitor the Firewall Threat Defense Virtual load and trigger Scale In/Scale Out operations.
- Register a new Firewall Threat Defense Virtual with the Firewall Management Center.
- Configure a new Firewall Threat Defense Virtual via Firewall Management Center.
- Unregister (remove) a scaled-in Firewall Threat Defense Virtual from the Firewall Management Center.

These functions are delivered in the form of compressed Zip package (see Build the Azure Function App Package, on page 183). The functions are as discrete as possible to carry out specific tasks, and can be upgraded as needed for enhancements and new release support.

### Orchestrator (Logic App)

The Auto Scale Logic App is a workflow, i.e. a collection of steps in a sequence. Azure functions are independent entities and cannot communicate with each other. This orchestrator sequences the execution of these functions and exchanges information between them.

- The Logic App is used to orchestrate and pass information between the auto scale Azure functions.
- Each step represents an auto scale Azure function or built-in standard logic.
- The Logic App is delivered as a JSON file.
- The Logic App can be customized via the GUI or JSON file.

### Virtual Machine Scale Set (VMSS)

The VMSS is a collection of homogeneous virtual machines, such as Firewall Threat Defense Virtual devices.

- The VMSS is capable of adding new identical VMs to the set.
- New VMs added to the VMSS are automatically attached with Load Balancers, Security Groups, and network interfaces.
- The VMSS has a built–in auto scale feature which is disabled for Firewall Threat Defense Virtual for Azure.

- You should not add or delete Firewall Threat Defense Virtual instances in the VMSS manually.

### Azure Resource Manager (ARM) Template

ARM templates are used to deploy the resources required by the Firewall Threat Defense Virtual auto scale for Azure solution.

Threat defense virtual auto scale for Azure - The ARM template **azure_ftdv_autoscale.json** provides input for the Auto Scale Manager components including:

- Azure Function App
- Azure Logic App
- The Virtual Machine Scale Set (VMSS)
- Internal/External load balancers.
- Security Groups and other miscellaneous components needed for deployment.

Threat defense virtual auto scale with Azure GWLB - The ARM template **azure_ftdv_autoscale_with_GWLB.json** provides input for the Auto Scale Manager components including:

- Azure Function App
- Azure Logic App
- Virtual Machine (VM) or Virtual Machine Scale Set (VMSS)
- Networking Infrastructure
- Gateway load balancer
- Security Groups and other miscellaneous components needed for deployment



**Important** The ARM template has limitations with respect to validating user input, hence it is your responsibility to validate input during deployment.

# Prerequisites

- Ensure that you have Owner role in the Azure subscription.
- Create the Azure Resource Group. Ensure that the Azure Virtual Network along with the necessary subnets are created.
  - Interfaces for NLB-based cluster : Management, Diagnostic, Inside, Outside, CCL and the function app.
  - Interfaces for GWLB-based cluster : Management, Diagnostic, Data, CCL and the function app.
- On the Management Center:
  - Ensure that Management Center Virtual is licensed correctly.
  - Create the access control policy.

- Create the Security Zone (SZ) object for the interfaces. For NLB based cluster, create the SZ for inside and outside interfaces. For GWLB-based cluster, create the SZ for the data interface.
- Create a separate user name and password for the azure function to add the Threat Defense Virtual instances to the Management Center Virtual and configure the instances.

- Install the Azure CLI on your local system.
- Download the Azure Clustering Autoscale repository from GitHub to your local computer and run the command **python3 make.py build** to create the Azure functions zip file.

## Azure Resources

### Resource Group

An existing or newly created Resource Group is required to deploy all the components of this solution.



**Note** Record the Resource Group name, the Region in which it is created, and the Azure Subscription ID for later use.

### Networking

Make sure a virtual network is available or created. An auto scale deployment with sandwich topology does not create, alter, or manage any networking resources. However, note that auto scale deployment with the Azure GWLB creates networking infrastructure.

The Firewall Threat Defense Virtual requires four network interfaces, thus your virtual network requires four subnets for:

1. Management traffic
2. Diagnostic traffic
3. Inside traffic
4. Outside traffic

The following ports should be open in the Network Security Group to which the subnets are connected:

- SSH(TCP/22)

  Required for the Health probe between the Load Balancer and Firewall Threat Defense Virtual.

  Required for communication between the Serverless functions and Firewall Threat Defense Virtual.

- TCP/8305

  Required for communication between Firewall Threat Defense Virtual and the Firewall Management Center.

- HTTPS(TCP/443)

  Required for communication between the Serverless components and the Firewall Management Center.

- Application-specific protocol/ports

Required for any user applications (for example, TCP/80, etc.).

**Note** Record the virtual network name, the virtual network CIDR, the names of the 4 subnets, and the Gateway IP addresses of the outside and inside subnets.

## Build the Azure Function App Package

The Firewall Threat Defense Virtual auto scale solution requires that you build an archive file: *ASM_Function.zip*. which delivers a set of discrete Azure functions in the form of a compressed ZIP package.

See Build Azure Functions from Source Code, on page 217 for instructions on how to build the *ASM_Function.zip* package.

These functions are as discrete as possible to carry out specific tasks, and can be upgraded as needed for enhancements and new release support.

## Prepare the Firewall Management Center

You manage the Firewall Threat Defense Virtual using the Firewall Management Center, a full-featured, multidevice manager. The Firewall Threat Defense Virtual registers and communicates with the Firewall Management Center on the Management interface that you allocated to the Firewall Threat Defense Virtual machine.

Create all the objects needed for the Firewall Threat Defense Virtual configuration and management, including a device group, so you can easily deploy policies and install updates on multiple devices. All the configurations applied on the device group will be pushed to the Firewall Threat Defense Virtual instances.

The following sections provide a brief overview of basic steps to prepare the Firewall Management Center. You should consult the full Secure Firewall Management Center Configuration Guide for complete information. When you prepare the Firewall Management Center, make sure you record the following information:

- The Firewall Management Center public IP address.
- The Firewall Management Center username/password.
- The security policy name.
- The inside and outside security zone object names.
- The device group name.

### Create a New Firewall Management Center User

Create a new user in the Firewall Management Center with Admin privileges to be used only by AutoScale Manager.

**Important** It's important to have the Firewall Management Center user account dedicated to the Firewall Threat Defense Virtual auto scale solution to prevent conflicts with other Firewall Management Center sessions.

### Procedure

**Step 1** Create new user in the Firewall Management Center with Admin privileges. Choose **System** > **Users** and click **Create User**.

The username must be Linux-valid:

- Maximum 32 alphanumeric characters, plus hyphen (-) and underscore (_)
- All lowercase
- Cannot start with hyphen (-); cannot be all numbers; cannot include a period (.), at sign (@), or slash (/)

**Step 2** Complete user options as required for your environment. See the Cisco Secure Firewall Management Center Administration Guide for complete information.

## Configure Access Control

Configure access control to allow traffic from inside to outside. Within an access control policy, access control rules provide a granular method of handling network traffic across multiple managed devices. Properly configuring and ordering rules is essential to building an effective deployment. See "Best Practices for Access Control" in the Cisco Secure Firewall Management Center Device Configuration Guide.

### Procedure

**Step 1** Choose **Policies** > **Access Control**.

**Step 2** Click **New Policy**.

**Step 3** Enter a unique **Name** and, optionally, a **Description**.

**Step 4** See the Cisco Secure Firewall Management Center Device Configuration Guide to configure security settings and rules for your deployment.

## Configure Licensing

All licenses are supplied to the Firewall Threat Defense by the Firewall Management Center. You can optionally purchase the following feature licenses:

- **Secure Firewall Threat Defense IPS**—Security Intelligence and Cisco Secure IPS
- **Secure Firewall Threat Defense Malware Defense**—Malware Defense
- **Secure Firewall Threat Defense URL Filtering**—URL Filtering
- **RA VPN**—AnyConnect Plus, AnyConnect Apex, or AnyConnect VPN Only.

**Note** When you buy a IPS , malware defense, or URL filtering license, you also need a matching subscription license to access updates for 1, 3, or 5 years.

### Before you begin

- Have a master account on the Cisco Smart Software Manager.

  If you do not yet have an account, click the link to set up a new account. The Smart Software Manager lets you create a master account for your organization.

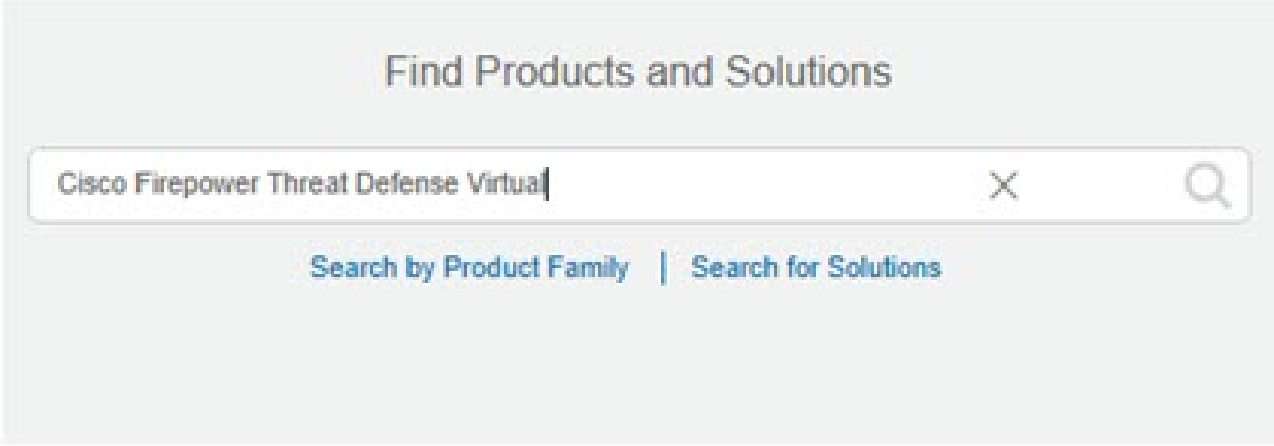
- Your Cisco Smart Software Licensing account must qualify for the Strong Encryption (3DES/AES) license to use some features (enabled using the export-compliance flag).

## Procedure

**Step 1** Make sure your Smart Licensing account contains the available licenses you need.

When you bought your device from Cisco or a reseller, your licenses should have been linked to your Smart Software License account. However, if you need to add licenses yourself, use the **Find Products and Solutions** search field on the Cisco Commerce Workspace. Search for the following license PIDs:

***Figure 7: License Search***




**Note**
If a PID is not found, you can add the PID manually to your order.

**Step 2** If you have not already done so, register the Firewall Management Center with the Smart Licensing server.

Registering requires you to generate a registration token in the Smart Software Manager. See the Cisco Secure Firewall Management Center Administration Guide for detailed instructions.

## Create Security Zone Objects

Create inside and outside security zone objects for your deployment.

### Procedure

**Step 1** Choose **Objects** > **Object Management**.

**Step 2** Choose **Interface** from the list of object types.

**Step 3** Click **Add** > **Security Zone**.

**Step 4** Enter a **Name** (for example *inside*, *outside*).

**Step 5** Choose **Routed** as the **Interface Type**.

**Step 6** Click **Save**.

## Create a Device Group

Device groups enable you to easily assign policies and install updates on multiple devices.

### Procedure

**Step 1** Choose **Devices** > **Device Management**.

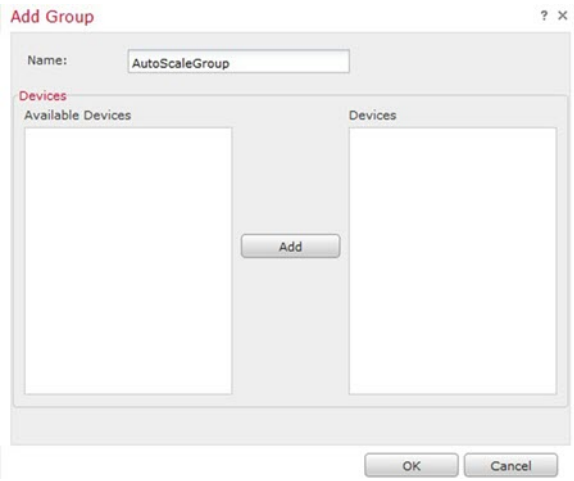
***Figure 8: Device Management***




**Step 2** From the **Add** drop-down menu, choose **Add Group**.

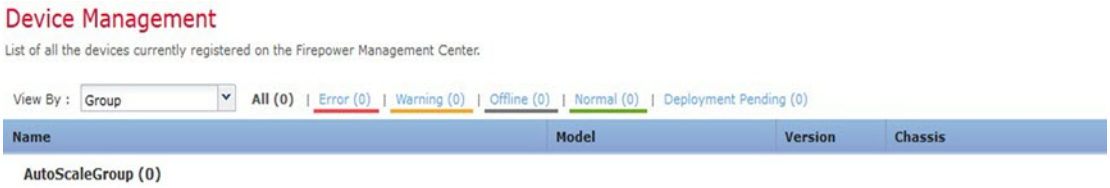
**Step 3** Enter a **Name**. For example, *AutoScaleGroup*.

*Figure 9: Add Device Group*




**Step 4** Click **OK** to add the device group.

*Figure 10: Device Group Added*




## Configure Secure Shell Access

Platform settings for Firewall Threat Defense devices configure a range of unrelated features whose values you might want to share among several devices. Firewall Threat Defense Virtual Auto Scale for Azure requires a Firewall Threat Defense Platform Settings Policy to allow SSH on the Inside/Outside zones and the device group created for the auto scale Group. This is required so that the Firewall Threat Defense Virtual's data interfaces can respond to Health Probes from Load Balancers.

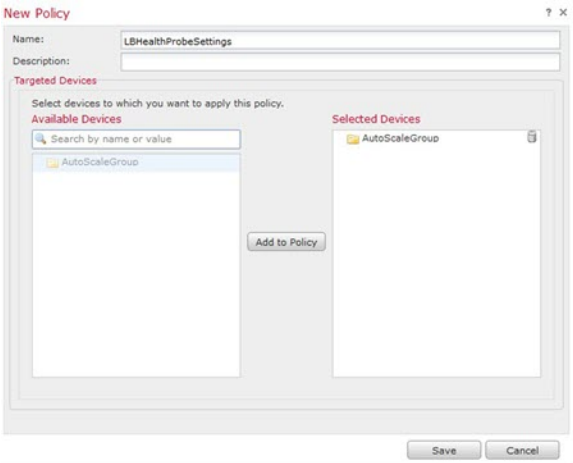
### Before you begin

You need network objects that define the hosts or networks you will allow to make SSH connections to the device. You can add objects as part of the procedure, but if you want to use object groups to identify a group of IP addresses, ensure that the groups needed in the rules already exist. Select **Objects** > **Object Management** to configure objects. For example, see the *azure-utility-ip (168.63.129.16)* object in the following procedure.

## Procedure

**Step 1** Select **Devices** > **Platform Settings** and create or edit a Firewall Threat Defense policy, for example *LBHealthProbeSettings*.

***Figure 11: Firewall Threat Defense Platform Settings Policy***




**Step 2** Select **Secure Shell**.

**Step 3** Identify the interfaces and IP addresses that allow SSH connections.

a) Click **Add** to add a new rule, or click **Edit** to edit an existing rule.

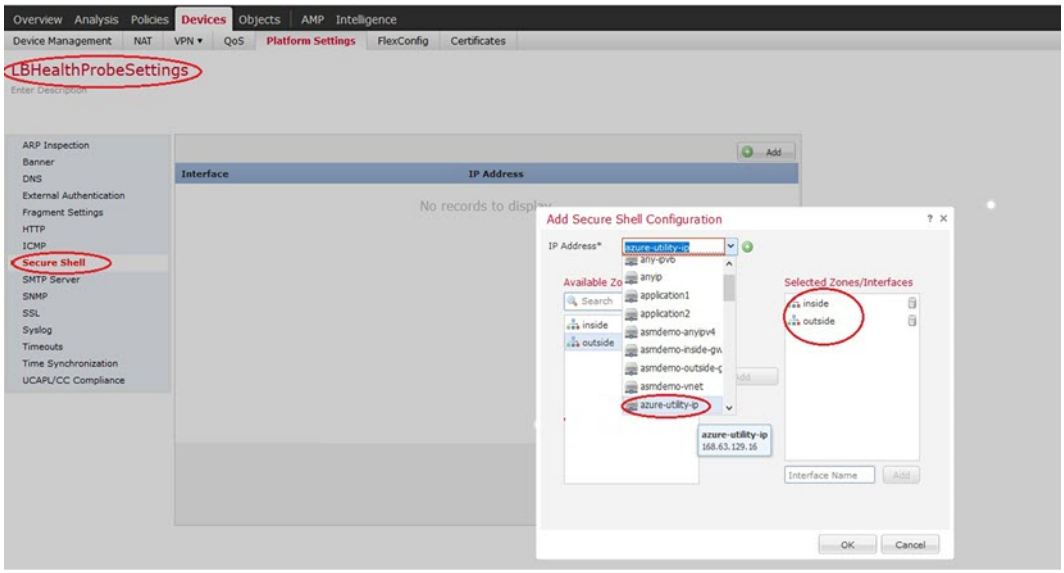
b) Configure the rule properties:

- **IP Address**—The network object that identifies the hosts or networks you are allowing to make SSH connections (for example, *azure-utility-ip (168.63.129.16)*). Choose an object from the drop-down menu, or add a new network object by clicking +.
- **Security Zones**—Add the zones that contain the interfaces to which you will allow SSH connections. For example, you can assign the inside interface to the **inside** zone; and the outside interface to the **outside** zone. You can create security zones from the Firewall Management Center's **Objects** page. See the Cisco Secure Firewall Management Center Device Configuration Guide for complete information about security zones.

  **Note**
  Inside interfaces are not used in the Auto Scale with Azure Gateway Load Balancer use case.

- Click **OK**.

*Figure 12: SSH Access for the Firewall Threat Defense Virtual Auto Scale*




**Step 4** Click **Save**.

You can now go to **Deploy** > **Deployment** and deploy the policy to assigned devices. The changes are not active until you deploy them.

**Note**
You can also configure TCP port 443 for the health probe instead of using **SSH Access**. To do this, go to **Devices > Platform settings > HTTP Access**, select the **Enable HTTP Server** checkbox, and enter **443** in the **Port** field. Associate this setting with the inside and outside interfaces. You have to also change the health probe port in the ARM template to 443. For more information on configuring HTTP Access, see Configuring HTTP.

## Configure NAT

Create a NAT policy and create the necessary NAT rules to forward traffic from the outside interface to your application, and attach this policy to the device group you created for auto scale.

**Note** You have to configure NAT only if you are configuring auto scale using a sandwich topology.

### Procedure

**Step 1** Choose **Devices** > **NAT**.

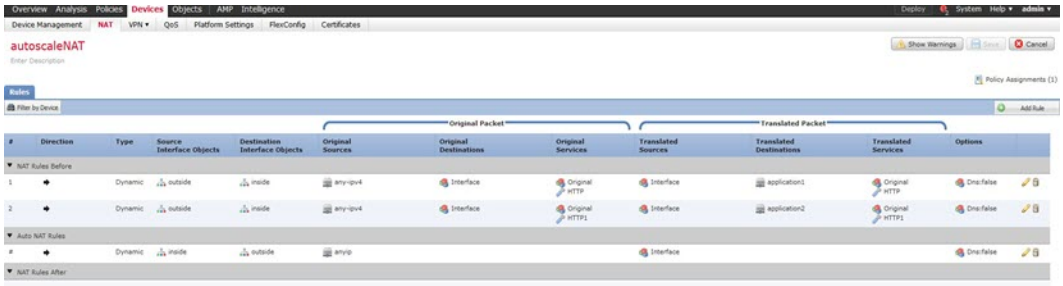
**Step 2** From the **New Policy** drop-down list, choose **Threat Defense NAT**.

**Step 3** Enter a unique **Name**.

**Step 4** Optionally, enter a **Description**.

**Step 5** Configure your NAT rules. See the procedure "Configure NAT for Threat Defense" in the Cisco Secure Firewall Management Center Device Configuration Guide for guidelines on how to create NAT rules and apply NAT policies. The following figure shows a basic approach.

***Figure 13: NAT Policy Example***




**Note**
We recommend that you keep your rules as simple as possible to avoid translation problems and difficult troubleshooting situations. Careful planning before you implement NAT is critical.

**Step 6** Click **Save**.

## Input Parameters

The following table defines the template parameters and provides an example. Once you decide on these values, you can use these parameters to create the Firewall Threat Defense Virtual device when you deploy the ARM template into your Azure subscription. See Deploy the Auto Scale ARM Template, on page 197.In the Auto scale with Azure GWLB solution, networking infrastructure is also created due to which additional input parameters have to be configured in the template. The parameter descriptions are self-explanatory.

***Table 21: Template Parameters***

| Parameter Name | Allowed Values/Type | Description | Resource Creation Type |
|---|---|---|---|
| resourceNamePrefix | String* (3-10 characters) | All the resources are created with name containing this prefix.<br>Note: Use only lowercase letters.<br>Example: ftdv | New |
| virtualNetworkRg | String | The virtual network resource group name.<br>Example: cisco-virtualnet-rg | Existing |
| virtualNetworkName | String | The virtual network name (already created).<br>Example: cisco-virtualnet | Existing |

| Parameter Name | Allowed Values/Type | Description | Resource Creation Type |
|---|---|---|---|
| virtualNetworkCidr | CIDR format<br>x.x.x.x/y | CIDR of Virtual Network (already created) | Existing |
| mgmtSubnet | String | The management subnet name (already created).<br>Example: cisco-mgmt-subnet | Existing |
| diagSubnet | String | The diagnostic subnet name (already created).<br>Example: cisco-diag-subnet | Existing |
| insideSubnet | String | The inside Subnet name (already created).<br>Example: cisco-inside-subnet | Existing |
| internalLbIp | String | The internal load balancer IP address for the inside subnet (already created).<br>Example: 1.2.3.4 | Existing |
| insideNetworkGatewayIp | String | The inside subnet gateway IP address (already created). | Existing |
| outsideSubnet | String | The outside subnet name (already created).<br>Example: cisco-outside-subnet | Existing |
| outsideNetworkGatewayIp | String | The outside subnet gateway IP (already created). | Existing |
| deviceGroupName | String | Device group in Firewall Management Center (already created) | Existing |
| insideZoneName | String | Inside Zone name in the Firewall Management Center (already created) | Existing |
| outsideZoneName | String | Outside Zone name in the Firewall Management Center (already created) | Existing |
| softwareVersion | String | The Firewall Threat Defense Virtual Version (selected from drop-down during deployment). | Existing |

| Parameter Name | Allowed Values/Type | Description | Resource Creation Type |
|---|---|---|---|
| vmSize | String | Size of Firewall Threat Defense Virtual instance (selected from drop-down during deployment). | N/A |
| ftdLicensingSku | String | Firewall Threat Defense Virtual Licensing Mode (PAYG/BYOL)<br>Note: PAYG is supported in Version 6.5+. | N/A |
| licenseCapability | Comma-separated string | BASE, MALWARE, URLFilter, THREAT | N/A |
| ftdVmManagementUserName | String* | The Firewall Threat Defense Virtual VM management administrator user name.<br>This **cannot** be 'admin'. See Azure for VM administrator user name guidelines. | New |
| ftdVmManagementUserPassword | String* | Password for the Firewall Threat Defense Virtual VM management administrator user.<br>Passwords must be 12 to 72 characters long, and must have: lowercase, uppercase, numbers, and special characters; and must have no more than 2 repeating characters.<br>**Note**<br>There is no compliance check for this in the template. | New |
| fmcIpAddress | String<br>x.x.x.x | The public IP address of the Firewall Management Center (already created) | Existing |
| fmcUserName | String | Firewall Management Center user name, with administrative privileges (already created) | Existing |
| fmcPassword | String | Firewall Management Center password for above Firewall Management Center user name (already created) | Existing |

| Parameter Name | Allowed Values/Type | Description | Resource Creation Type |
|---|---|---|---|
| policyName | String | Security Policy created in the Firewall Management Center (already created) | Existing |
| scalingPolicy | POLICY-1 / POLICY-2 | **POLICY-1**: Scale-Out will be triggered when the average load of any Firewall Threat Defense Virtual goes beyond the Scale Out threshold for the configured duration.<br>**POLICY-2**: Scale-Out will be triggered when average load of all the Firewall Threat Defense Virtual devices in the auto scale group goes beyond the Scale Out threshold for the configured duration.<br>In both cases Scale-In logic remains the same: Scale-In will be triggered when average load of all the Firewall Threat Defense Virtual devices comes below the Scale In threshold for the configured duration. | N/A |
| scalingMetricsList | String | Metrics used in making the scaling decision.<br>Allowed: CPU<br>CPU, MEMORY<br>Default: CPU | N/A |
| cpuScaleInThreshold | String | The Scale-In threshold in percent for CPU metrics.<br>Default: 10<br>When the Firewall Threat Defense Virtual metric goes below this value the Scale-In will be triggered.<br>See Auto Scale Logic, on page 213. | N/A |

| Parameter Name | Allowed Values/Type | Description | Resource Creation Type |
|---|---|---|---|
| cpuScaleOutThreshold | String | The Scale-Out threshold in percent for CPU metrics.<br>Default: 80<br>When the Firewall Threat Defense Virtual metric goes above this value, the Scale-Out will be triggered.<br>The 'cpuScaleOutThreshold' should always be **greater** than the 'cpuScaleInThreshold'.<br>See Auto Scale Logic, on page 213. | N/A |
| memoryScaleInThreshold | String | The Scale-In threshold in percent for memory metrics.<br>Default: 0<br>When the Firewall Threat Defense Virtual metric goes below this value the Scale-In will be triggered.<br>See Auto Scale Logic, on page 213. | N/A |
| memoryScaleOutThreshold | String | The Scale-Out threshold in percent for memory metrics.<br>Default: 0<br>When the Firewall Threat Defense Virtual metric goes above this value, the Scale-Out will be triggered.<br>The 'memoryScaleOutThreshold' should always be **greater** than the 'memoryScaleInThreshold'.<br>See Auto Scale Logic, on page 213. | N/A |
| minFtdCount | Integer | The minimum Firewall Threat Defense Virtual instances available in the scale set at any given time.<br>Example: 2 | N/A |

| Parameter Name | Allowed Values/Type | Description | Resource Creation Type |
|---|---|---|---|
| maxFtdCount | Integer | The maximum Firewall Threat Defense Virtual instances allowed in the Scale set.<br>Example: 10<br>**Note** This number is restricted by the Firewall Management Center capacity.<br>The Auto Scale logic will not check the range of this variable, hence fill this carefully. | N/A |
| metricsAverageDuration | Integer | Select from the drop-down.<br>This number represents the time (in minutes) over which the metrics are averaged out.<br>If the value of this variable is 5 (i.e. 5min), when the Auto Scale Manager is scheduled it will check the past 5 minutes average of metrics and based on this it will make a scaling decision.<br>**Note** Only numbers 1, 5, 15, and 30 are valid due to Azure limitations. | N/A |

| Parameter Name | Allowed Values/Type | Description | Resource Creation Type |
|---|---|---|---|
| initDeploymentMode | BULK / STEP | Primarily applicable for the first deployment, or when the Scale Set does not contain any Firewall Threat Defense Virtual instances.<br><br>BULK: The Auto Scale Manager will try to deploy 'minFtdCount' number of Firewall Threat Defense Virtual instances in parallel at one time.<br><br>**Note**<br>The launch is in parallel, but registering with the Firewall Management Center is sequential due to Firewall Management Center limitations.<br><br>STEP: The Auto Scale Manager will deploy the 'minFtdCount' number of Firewall Threat Defense Virtual devices one by one at each scheduled interval.<br><br>**Note**<br>The STEP option will take a long time for the ‘minFtdCount’ number of instances to be launched and configured with the Firewall Management Center and become operational, but useful in debugging.<br><br>The BULK option takes same amount of time to launch all ‘minFtdCount’ number of Firewall Threat Defense Virtual as one Firewall Threat Defense Virtual launch takes (because it runs in parallel), but the Firewall Management Center registration is sequential.<br><br>The total time to deploy ‘minFtdCount’ number of Firewall Threat Defense Virtual = (time to launch One Firewall Threat Defense Virtual + time to register/configure one Firewall Threat Defense Virtual * minFtdCount ). | |

| Parameter Name | Allowed Values/Type | Description | Resource Creation Type |
|---|---|---|---|
| *Azure has restrictions on the naming convention for new resources. Review the limitations or simply use all lowercase. **Do not use spaces or any other special characters**. | | | |

# Deploy the Auto Scale Solution

## Download the Deployment Package

The Firewall Threat Defense Virtual auto scale for Azure solution using sandwich topology is an Azure Resource Manager (ARM) template-based deployment which makes use of the serverless infrastructure provided by Azure (Logic App, Azure Functions, Load Balancers, Virtual Machine Scale Set, and so on).

The Firewall Threat Defense Virtual auto scale with Azure GWLB solution is an ARM template-based deployment that creates the GWLB, networking infrastructure, threat defense virtual auto scaling group, serverless components, and other required resources.

The deployment procedure for both the solutions are similar.

Download the files required to launch the Firewall Threat Defense Virtual auto scale for Azure solution. Deployment scripts and templates for your version are available in the GitHub repository.

**Attention** Note that Cisco-provided deployment scripts and templates for auto scale are provided as open source examples, and are not covered within the regular Cisco TAC support scope. Check GitHub regularly for updates and ReadMe instructions.

See Build Azure Functions from Source Code, on page 217 for instructions on how to build the *ASM_Function.zip* package.

## Deploy the Auto Scale ARM Template

**Threat defense virtual auto scale for Azure using Sandwich Topology -** Use the ARM template **azure_ftdv_autoscale.json** to deploy the resources required by the Firewall Threat Defense Virtual auto scale for Azure. Within a given resource group, the ARM template deployment creates the following:

- Virtual Machine Scale Set (VMSS)
- External Load Balancer
- Internal Load Balancer
- Azure Function App
- Logic App
- Security groups (For Data and Management interfaces)

**Threat defense virtual auto scale with Azure GWLB -** Use the ARM template **azure_ftdv_autoscale_with_GWLB.json** to deploy the resources required by the Firewall Threat Defense Virtual auto scale with Azure GWLB solution. Within a given resource group, the ARM template deployment creates the following:

- Virtual Machine (VM) or Virtual Machine Scale Set (VMSS)
- Gateway Load Balancer
- Azure Function App
- Logic App
- Networking Infrastructure
- Security Groups and other miscellaneous components needed for deployment

### Before you begin

- Download the ARM templates from the GitHub repository (https://github.com/CiscoDevNet/cisco-ftdv/tree/master/autoscale/azure).

## Procedure

**Step 1** If you need to deploy the Firewall Threat Defense Virtual instances in multiple Azure zones, edit the ARM template based on the zones available in the Deployment region.

**Example:**

```
"zones": [
    "1",
    "2",
    "3"
],
```

This example shows the “Central US” region which has 3 zones.

**Step 2** Edit the traffic rules required in External Load Balancer. You can add any number of rules by extending this ‘json’ array. This is valid for the Auto Scale using Sandwich Topology use case.

**Example:**

```
{
 "type": "Microsoft.Network/loadBalancers",
 "name": "[variables('elbName')]",
 "location": "[resourceGroup().location]",
 "apiVersion": "2018-06-01",
 "sku": {
   "name": "Standard"
 },
 "dependsOn": [
   "[concat('Microsoft.Network/publicIPAddresses/', variables('elbPublicIpName'))]"

 ],
 "properties": {
   "frontendIPConfigurations": [
     {
       "name": "LoadBalancerFrontEnd",
         "properties": {
           "publicIPAddress": {
             "id": "[resourceId('Microsoft.Network/publicIPAddresses/',
```

```
variables('elbPublicIpName'))]"
                    }
                  }
                }
              ],
              "backendAddressPools": [
                {
                  "name": "backendPool"
                }
              ],
              "loadBalancingRules": [
                {
                  "properties": {
                    "frontendIPConfiguration": {
                     "Id": "[concat(resourceId('Microsoft.Network/loadBalancers', variables('elbName')),
 '/frontendIpConfigurations/LoadBalancerFrontend')]"
                    },
                    "backendAddressPool": {
                    "Id": "[concat(resourceId('Microsoft.Network/loadBalancers', variables('elbName')),
 '/backendAddressPools/BackendPool')]"
                    },
                    "probe": {
                    "Id": "[concat(resourceId('Microsoft.Network/loadBalancers', variables('elbName')),
 '/probes/lbprobe')]"
                    },
                    "protocol": "TCP",
                    "frontendPort": "80",
                    "backendPort": "80",
                    "idleTimeoutInMinutes": "[variables('idleTimeoutInMinutes')]"
                  },
                  "Name": "lbrule"
                }
              ],
```

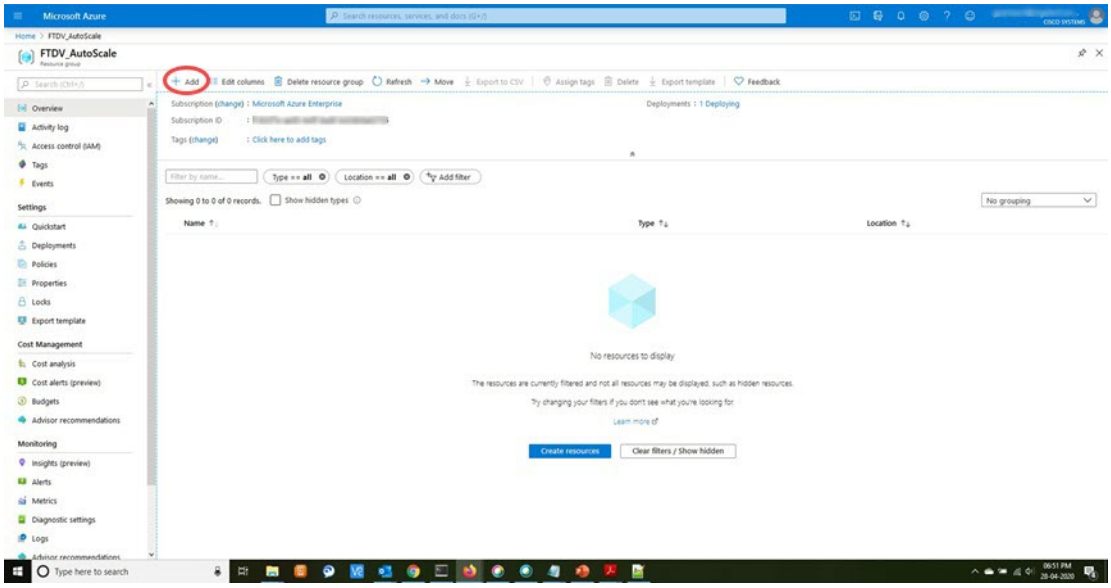
**Note**
You can also edit this from the Azure portal post-deployment if you prefer not to edit this file.

**Step 3** Log in to the Microsoft Azure portal using your Microsoft account username and password.

**Step 4** Click **Resource groups** from the menu of services to access the Resource Groups blade. You will see all the resource groups in your subscription listed in the blade.

Create a new resource group or select an existing, empty resource group; for example, *Firewall Threat Defense Virtual_AutoScale*.

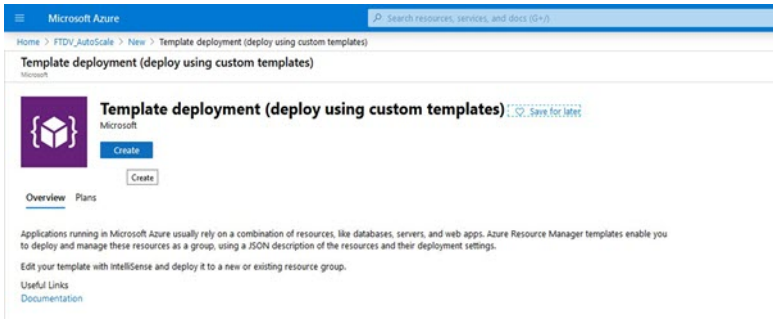
*Figure 14: Azure Portal*




**Step 5** Click **Create a resource (+)** to create a new resource for template deployment. The Create Resource Group blade appears.

**Step 6** In **Search the Marketplace**, type **Template deployment (deploy using custom templates)**, and then press **Enter**.

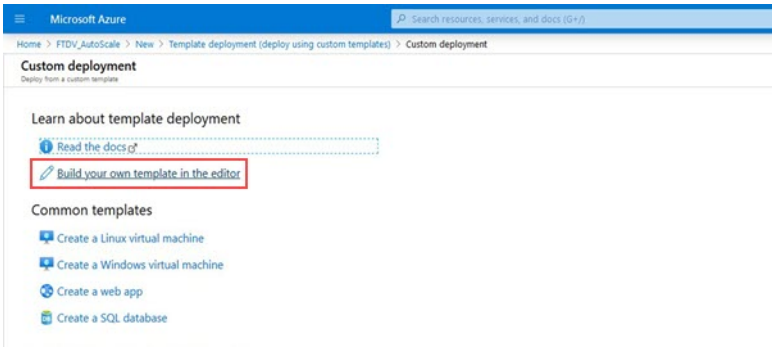
*Figure 15: Custom Template Deployment*




**Step 7** Click **Create**.

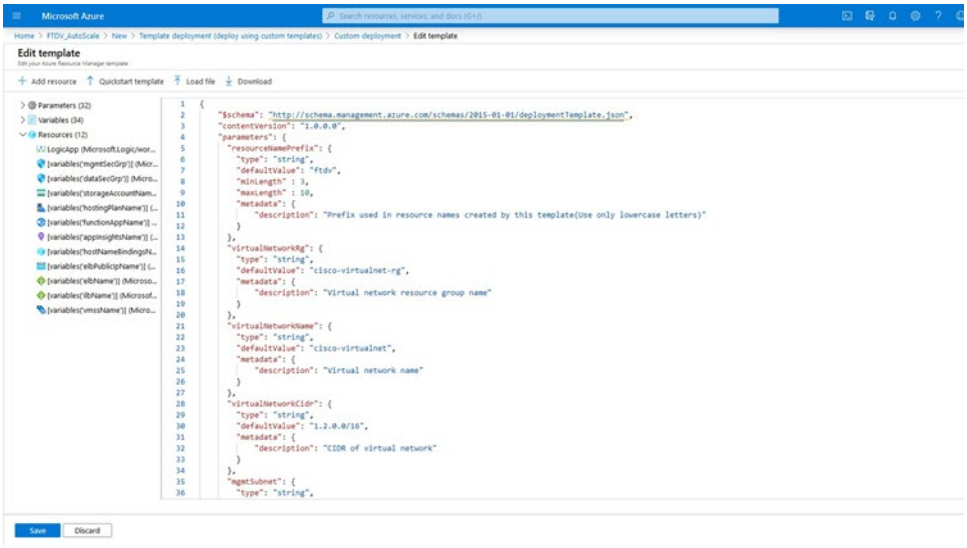
**Step 8** There are several options for creating a template. Choose **Build your own template in editor**.

*Figure 16: Build Your Own Template*



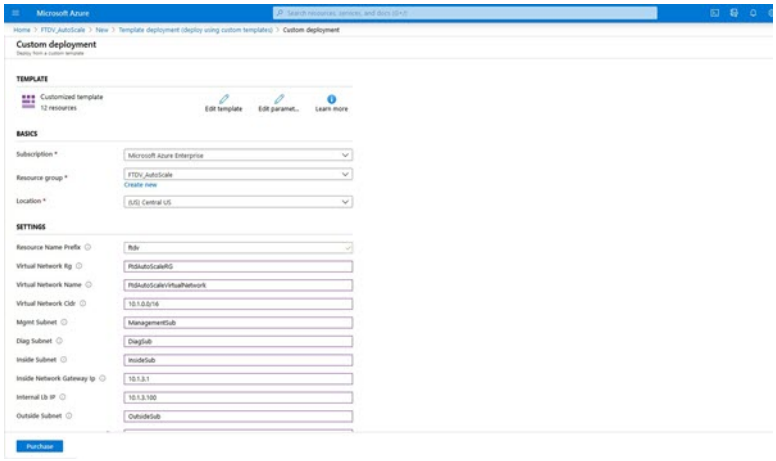

**Step 9** In the **Edit template** window, delete all the default content and copy the contents from the updated *azure_ftdv_autoscale.json* and click **Save**.

*Figure 17: Edit Template*




**Step 10** In next section, fill all the parameters. Refer to Input Parameters, on page 190 for details about each parameter, then click **Purchase**.

*Figure 18: ARM Template Parameters*



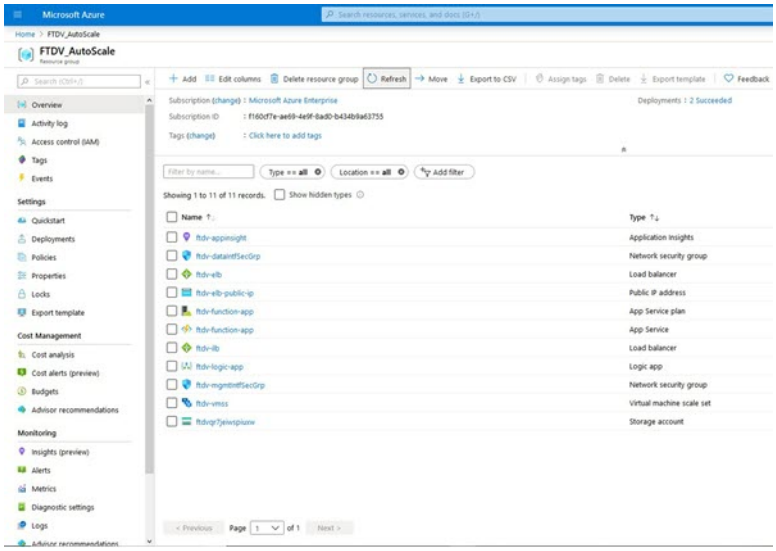

**Note**

You can also click **Edit Parameters** and edit the JSON file or upload pre-filled contents.

The ARM template has limited input validation capabilities, hence it is your responsibility to validate the input.

**Step 11** When a template deployment is successful, it creates all the required resources for the Firewall Threat Defense Virtual auto scale for Azure solution. See the resouses in the following figure. The Type column describes each resource, including the Logic App, VMSS, Load Balancers, Public IP address, etc.

*Figure 19: Threat Defense Virtual Auto Scale Template Deployment*




## Deploy the Azure Function App

When you deploy the ARM template, Azure creates a skeleton Function App, which you then need to update and configure manually with the functions required for the Auto Scale Manager logic.

### Before you begin

- Build the *ASM_Function.zip* package. See Build Azure Functions from Source Code, on page 217.

## Procedure

**Step 1** Go to the Function App you created when you deployed the ARM template, and verify that no functions are present. In a browser go to this URL:

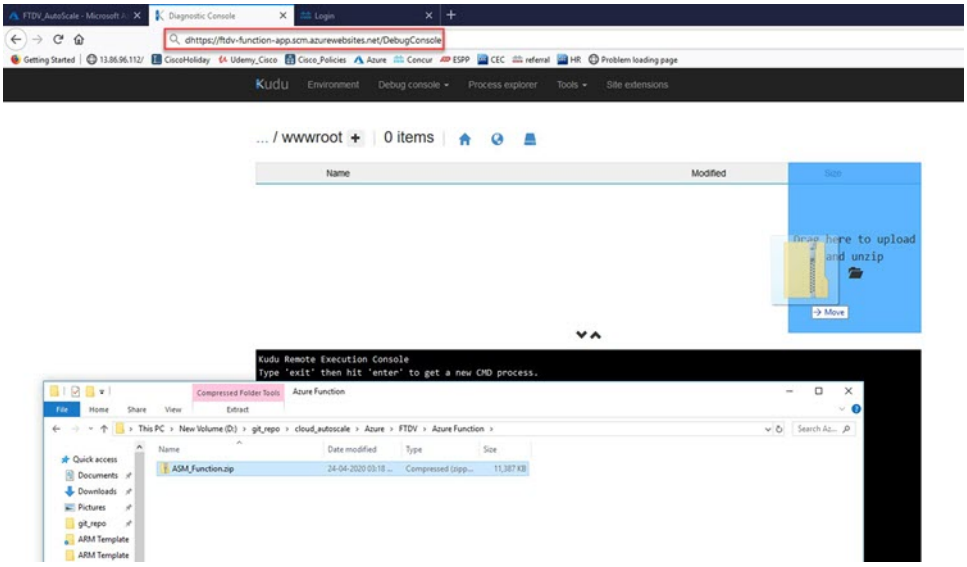
https://<Function App Name>.scm.azurewebsites.net/DebugConsole

For the example in Deploy the Auto Scale ARM Template, on page 197:

**Step 2** In the file explorer navigate to **site/wwwroot**.

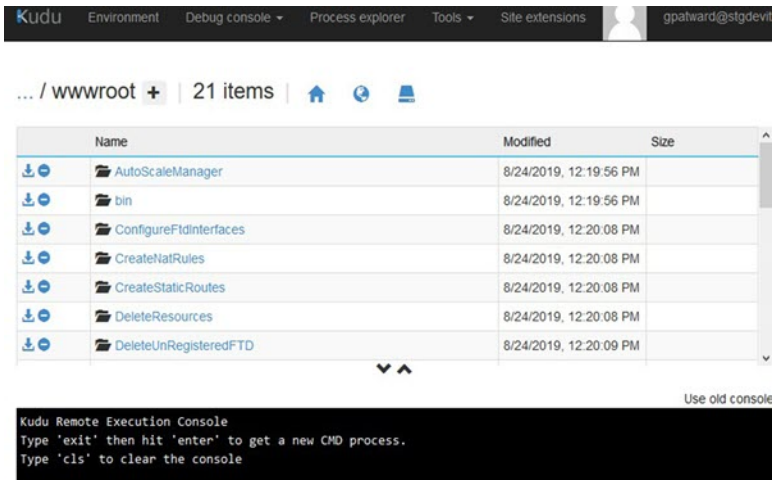
**Step 3** Drag-and-drop the **ASM_Function.zip** to the right side corner of the file explorer.

***Figure 20: Upload the Threat Defense Virtual Auto Scale Functions***




**Step 4** Once the upload is successful, all of the serverless functions should appear.

*Figure 21: Threat Defense Virtual Serverless Functions*

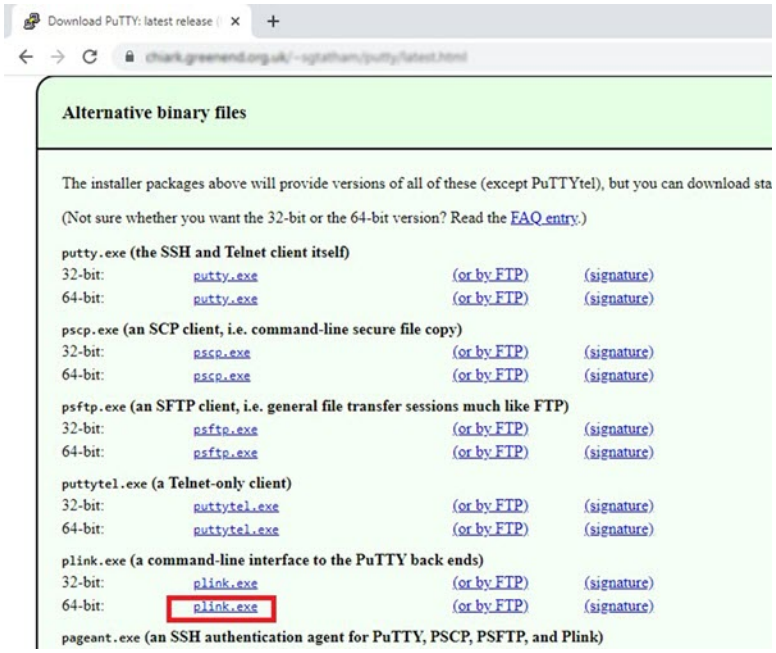



**Step 5** Download the PuTTY SSH client.

Azure functions need to access the Firewall Threat Defense Virtual via an SSH connection. However, the opensource libraries used in the serverless code do not support the SSH key exchange algorithms used by the Firewall Threat Defense Virtual. Hence you need to download a pre-built SSH client.

Download the PuTTY command-line interface to the PuTTY back end (*plink.exe*) from www.putty.org.

*Figure 22: Download PuTTY*




**Step 6** Rename the SSH client executable file **plink.exe** to **ftdssh.exe**.

**Step 7** Drag-and-drop the **ftdssh.exe** to the right side corner of the file explorer, to the location where **ASM_Function.zip** was uploaded in the previous step.

**Step 8** Verify the SSH client is present with the function application. Refresh the page if necessary.

## Fine Tune the Configuration

There are a few configurations available to fine tune the Auto Scale Manager or to use in debugging. These options are not exposed in the ARM template, but you can edit them under the Function App.

### Before you begin

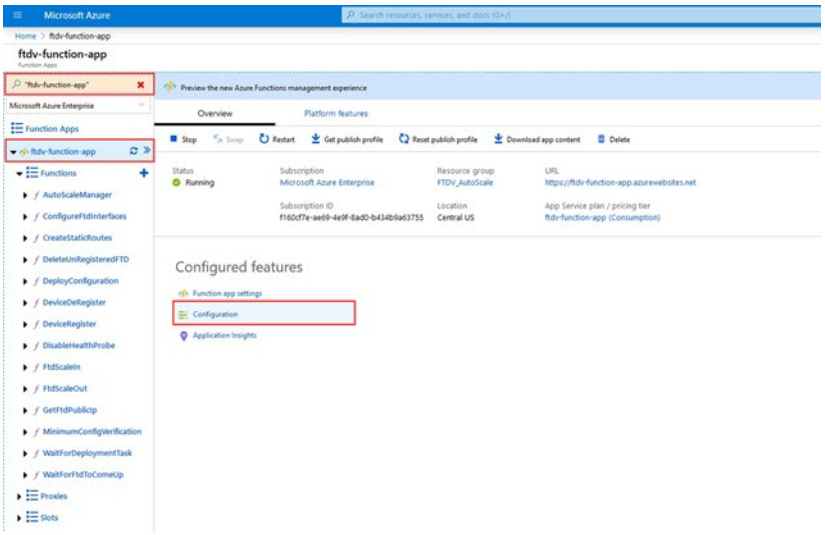
**Note** This can be edited at any time. Follow this sequence to edit the configurations.

- Disable the Function App.
- Wait for existing scheduled task to finish.
- Edit and save the configuration.
- Enable the Function App.

### Procedure

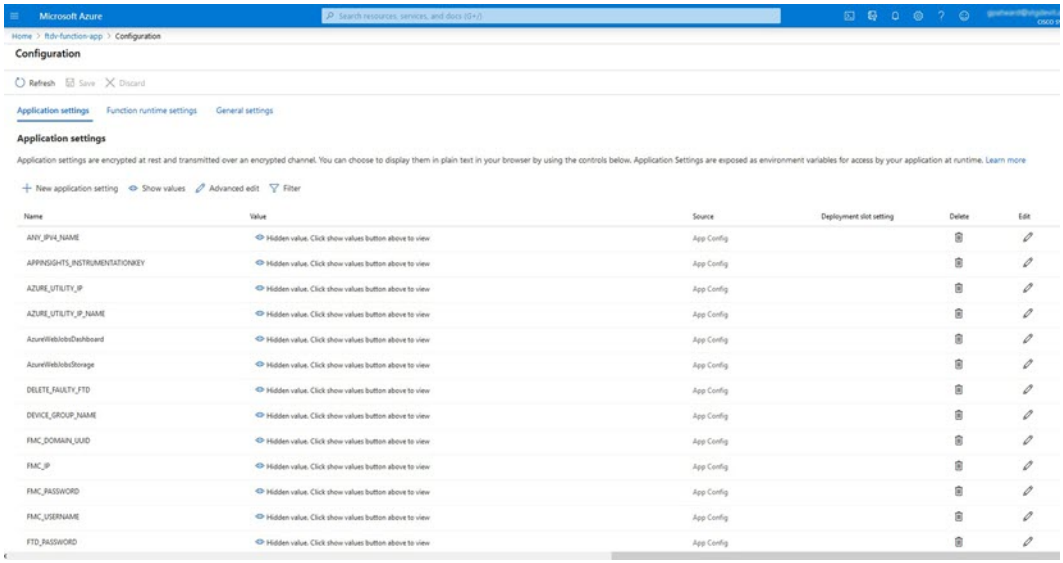
**Step 1** In the Azure portal, search for and select the Firewall Threat Defense Virtual function application.

***Figure 23: Threat Defense Virtual Function Application***




**Step 2** Configurations passed via the ARM template can also be edited here. Variable names may appear different from the ARM template, but you can easily identify the purpose of these variables from their name.

*Figure 24: Application Settings*




Most of the options are self-explanatory from the name. For example:

- Configuration Name: “DELETE_FAULTY_FTD” (Default value : YES )

  During Scale-Out, a new Firewall Threat Defense Virtual instance is launched and registered with the Firewall Management Center. In case the registration fails, based on this option, Auto Scale Manager will decide to keep that Firewall Threat Defense Virtual instance or delete it. (YES : Delete faulty Firewall Threat Defense Virtual / NO : Keep the Firewall Threat Defense Virtual instance even if it fails to register with the Firewall Management Center).

- In the Function App settings, all the variables (including variables containing a secure string like ‘password’) can be seen in clear text format by users that have access to the Azure subscription.

  If users have any security concerns with this (for example, if an Azure subscription is shared among users with lower privilages within the organization), a user can make use of Azure’s *Key Vault* service to protect passwords. Once this is configured, instead of providing a clear text ‘password’ in function settings, a user has to provide a secure identifier generated by the key vault where the password is stored.

  **Note**
  Search the Azure documentation to find the best practices to secure your application data.

## Configure the IAM Role in the Virtual Machine Scale Set

Azure Identity and Access Management (IAM) is used as a part of Azure Security and Access Control to manage and control a user’s identity. Managed identities for Azure resources provides Azure services with an automatically managed identity in Azure Active Directory.

This allows the Function App to control the Virtual Machine Scale Sets (VMSS) without explicit authentication credentials.

### Procedure

**Step 1** In the Azure portal, go to the VMSS.

**Step 2** Click **Access control (IAM)**.

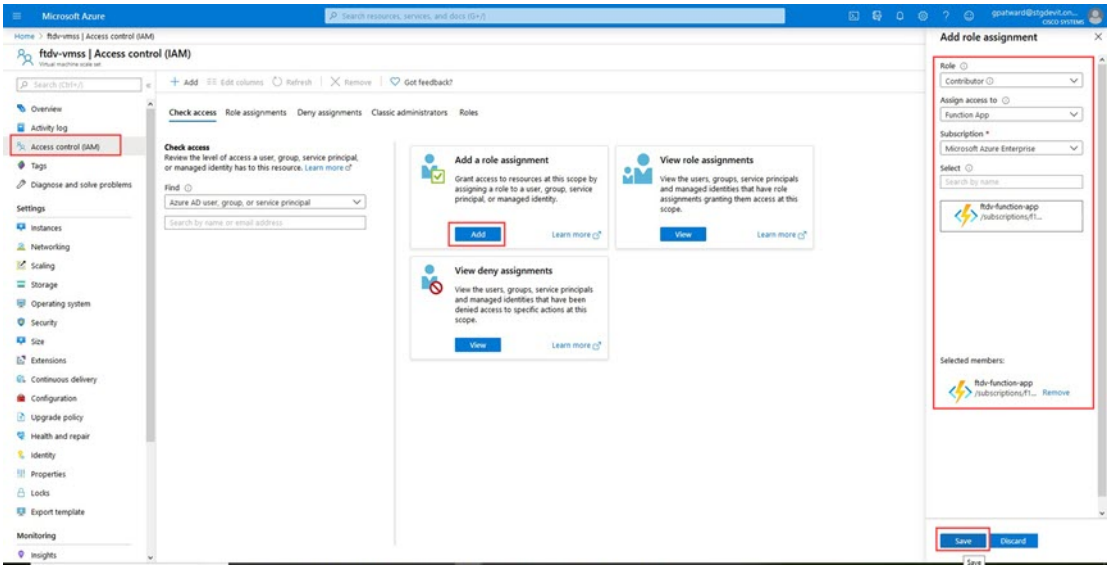
**Step 3** Click **Add** to add a role assignment

**Step 4** From the **Add role assignment** drop-down, choose **Contributor**.

**Step 5** From the **Assign access to** drop-down, choose **Function App**.

**Step 6** Select the Firewall Threat Defense Virtual function application.

***Figure 25: AIM Role Assignment***




**Step 7** Click **Save**.

**Note**
You should also verify that there are no Firewall Threat Defense Virtual instances launched yet.

## Update Security Groups

The ARM template creates two security groups, one for the Management interface, and one for data interfaces. The Management security group will allow only traffic required for Firewall Threat Defense Virtual management activities. However, the data interface security group will allow all traffic.

### Procedure

Fine tune the security group rules based on the topology and application needs of your deployments.

**Note**

The data interface security group should allow, at a minimum, SSH traffic from the load balancers.

## Update the Azure Logic App

The Logic App acts as the orchestrator for the Autoscale functionality. The ARM template creates a skeleton Logic App, which you then need to update manually to provide the information necessary to function as the auto scale orchestrator.

### Procedure

**Step 1** From the repository, retrieve the file *LogicApp.txt* to the local system and edit as shown below.

**Important**
Read and understand all of these steps before proceeding.

These manual steps are not automated in the ARM template so that only the Logic App can be upgraded independently later in time.

a) Required: Find and replace all the occurrences of "SUBSCRIPTION_ID" with your subscription ID information.
b) Required: Find and replace all the occurrences of "RG_NAME" with your resource group name.
c) Required: Find and replace all of the occurrences of "FUNCTIONAPPNAME" to your function app name.

The following example shows a few of these lines in the *LogicApp.txt* file:

```
  "AutoScaleManager": {
      "inputs": {
          "function": {
              "id":
"/subscriptions/SUBSCRIPTION_ID/resourceGroups/RG_NAME/providers/Microsoft.Web/sites/FUNCTIONAPPNAME/functions/AutoScaleManager"

          }
.
.
                      },
                      "Deploy_Changes_to_FTD": {
                          "inputs": {
                              "body": "@body('AutoScaleManager')",
                              "function": {
                                  "id":
"/subscriptions/SUBSCRIPTION_ID/resourceGroups/RG_NAME/providers/Microsoft.Web/sites/FUNCTIONAPPNAME/functions/DeployConfiguration"

                              }
.
.
                      "DeviceDeRegister": {
                          "inputs": {
                              "body": "@body('AutoScaleManager')",
                              "function": {
                                  "id":
"/subscriptions/SUBSCRIPTION_ID/resourceGroups/RG_NAME/providers/Microsoft.Web/sites/FUNCTIONAPPNAME/functions/DeviceDeRegister"

                              }
                          },
                          "runAfter": {
```

```
"Delay_For_connection_Draining": [
```

d) (Optional) Edit the trigger interval, or leave the default value (5). This is the time interval at which the Autoscale functionality is periodically triggered. The following example shows these lines in the *LogicApp.txt* file:

```
"triggers": {
    "Recurrence": {
        "conditions": [],
        "inputs": {},
        "recurrence": {
            "frequency": "Minute",
            "interval": 5
        },
```

e) (Optional) Edit the time to drain, or leave the default value (5). This is the time interval to drain existing connections from the Firewall Threat Defense Virtual before deleting the device during the Scale-In operation. The following example shows these lines in the *LogicApp.txt* file:

```
"actions": {
     "Branch_based_on_Scale-In_or_Scale-Out_condition": {
         "actions": {
             "Delay_For_connection_Draining": {
                 "inputs": {
                     "interval": {
                         "count": 5,
                         "unit": "Minute"
                     }
```

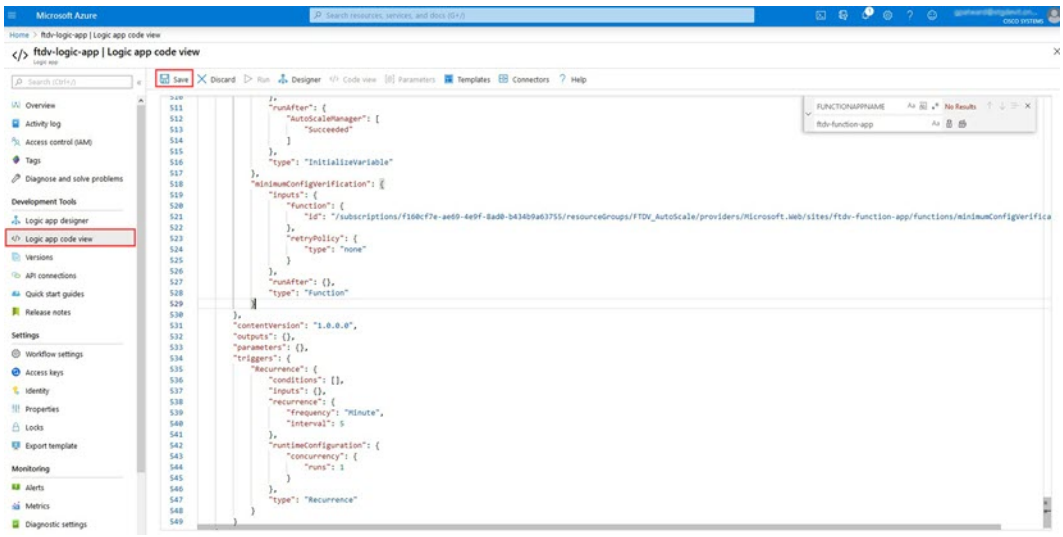
f) (Optional) Edit the cool down time, or leave the default value (10). This is the time to perform NO ACTION after the Scale-Out is complete. The following example shows these lines in the *LogicApp.txt* file:

```
"actions": {
    "Branch_based_on_Scale-Out_or_Invalid_condition": {
        "actions": {
            "Cooldown_time": {
                "inputs": {
                    "interval": {
                        "count": 10,
                        "unit": "Second"
                    }
```

**Note**
These steps can also be done from the Azure portal. Consult the Azure documentation for more information.

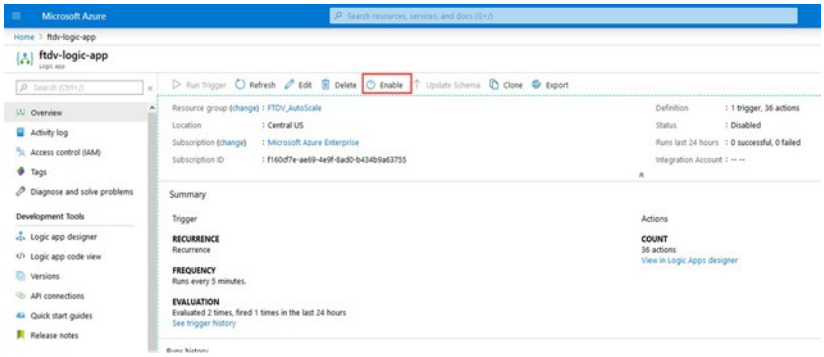
**Step 2** Go to the **Logic App code view**, delete the default contents and paste the contents from the edited *LogicApp.txt* file, and click **Save**.

*Figure 26: Logic App Code View*



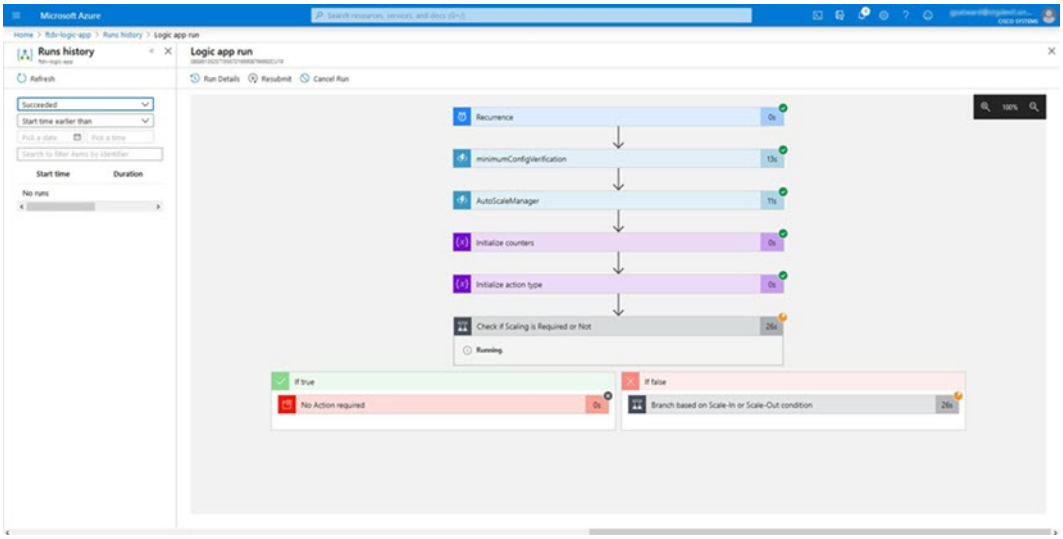

**Step 3** When you save the Logic App, it is in a 'Disabled' state. Click **Enable** when you want to start the Auto Scale Manager.

*Figure 27: Enable Logic App*




**Step 4** Once enabled, the tasks start running. Click the 'Running' status to see the activity.

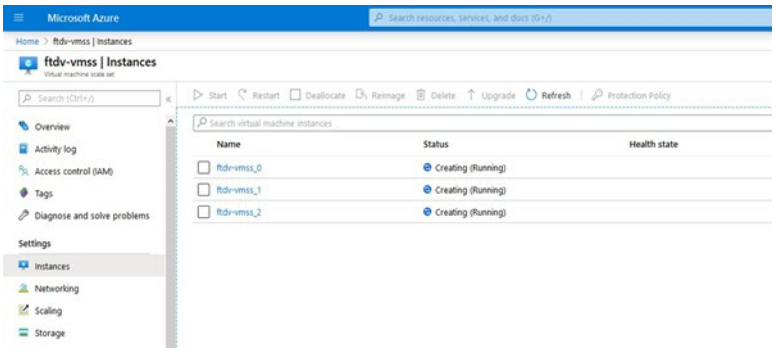
*Figure 28: Logic App Running Status*




**Step 5** Once the Logic App starts, all the deployment-related steps are complete.

**Step 6** Verify in the VMSS that Firewall Threat Defense Virtual instances are being created.

*Figure 29: Threat Defense Virtual Instances Running*




In this example, three Firewall Threat Defense Virtual instances are launched because 'minFtdCount' was set to '3' and 'initDeploymentMode' was set to 'BULK' in the ARM template deployment.

## Upgrade the Firewall Threat Defense Virtual

The Firewall Threat Defense Virtual upgrade is supported only in the form of an image upgrade of virtual machine scale set (VMSS). Hence, you upgrade the Firewall Threat Defense Virtual through the Azure REST API interface.

**Note** You can use any REST client to upgrade the Firewall Threat Defense Virtual.

### Before you begin

- Obtain the new Firewall Threat Defense Virtual image version available in market place (example: 650.32.0).
- Obtain the SKU used to deploy original scale set (example: ftdv-azure-byol ).
- Obtain the Resource Group and the virtual machine scale set name.

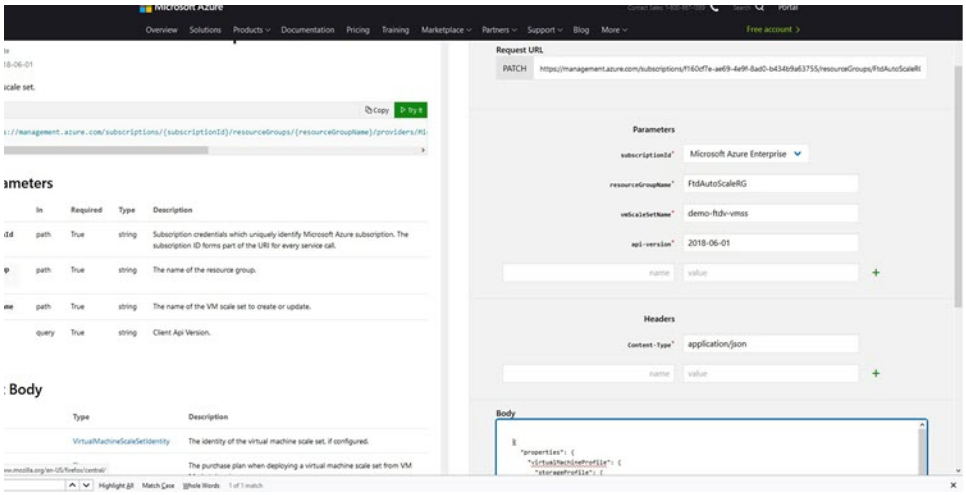
## Procedure

**Step 1** In a browser go to the following URL:

https://docs.microsoft.com/en-us/rest/api/compute/virtualmachinescalesets/update#code-try-0

**Step 2** Enter the details in the Parameters section.

***Figure 30: Upgrade the Firewall Threat Defense Virtual***




**Step 3** Enter the JSON input containing the new Firewall Threat Defense Virtual image version, SKU, and trigger RUN in the **Body** section.

```
{
 "properties": {
        "virtualMachineProfile": {
              "storageProfile": {
                      "imageReference": {
                           "publisher": "cisco",
                           "offer": "cisco-ftdv",
                           "sku": "ftdv-azure-byol",
                            "version": "650.32.0"
                      }
                  },
            }
      }
}
```

**Step 4** A successful response from Azure means that the VMSS has accepted the change.

The new image will be used in the new Firewall Threat Defense Virtual instances which will get launched as part of Scale-Out operation.

- Existing Firewall Threat Defense Virtual instances will continue to use the old software image while they exist in a scale set.
- You can override the above behavior and upgrade the existing Firewall Threat Defense Virtual instances manually. To do this, click the **Upgrade** button in the VMSS. It will reboot and upgrade the selected Firewall Threat Defense Virtual instances. You must reregister and reconfigure these upgraded Firewall Threat Defense Virtual instances manually. **Note that this method is NOT recommended.**

# Auto Scale Logic

### Scaling Metrics

You use the ARM template to deploy the resources required by the Firewall Threat Defense Virtualauto scale solution. During ARM template deployment, you have the following options for scaling metrics:

- CPU
- CPU, Memory (Version 6.7+).

**Note** CPU metrics are collected from Azure; memory metrics are collected from the Firewall Management Center.

### Scale-Out Logic

- **POLICY-1**: Scale-Out will be triggered when the average load of **any** Firewall Threat Defense Virtual goes beyond the Scale-Out threshold for the configured duration. When using the 'CPU,MEMORY' scaling metric, the Scale-Out threshold is the average CPU **or** memory utilization of **any** Firewall Threat Defense Virtual in the scale set.
- **POLICY-2**: Scale-Out will be triggered when average load of **all** of the Firewall Threat Defense Virtual devices go beyond Scale-Out threshold for the configured duration. When using the 'CPU,MEMORY' scaling metric, the Scale-Out threshold is the average CPU **or** Memory utilization of **all** Firewall Threat Defense Virtual devices in the scale set.

### Scale-In Logic

- If the CPU utilization of **all** of the Firewall Threat Defense Virtual devices goes below the configured Scale-In threshold for the configured duration. When using the 'CPU,MEMORY' scaling metric, if the CPU **and** memory utilization of all Firewall Threat Defense Virtual devices in the scale set goes below the configured Scale-In threshold for the configured duration, the Firewall Threat Defense Virtual with the least loaded CPU will be selected for termination

### Notes

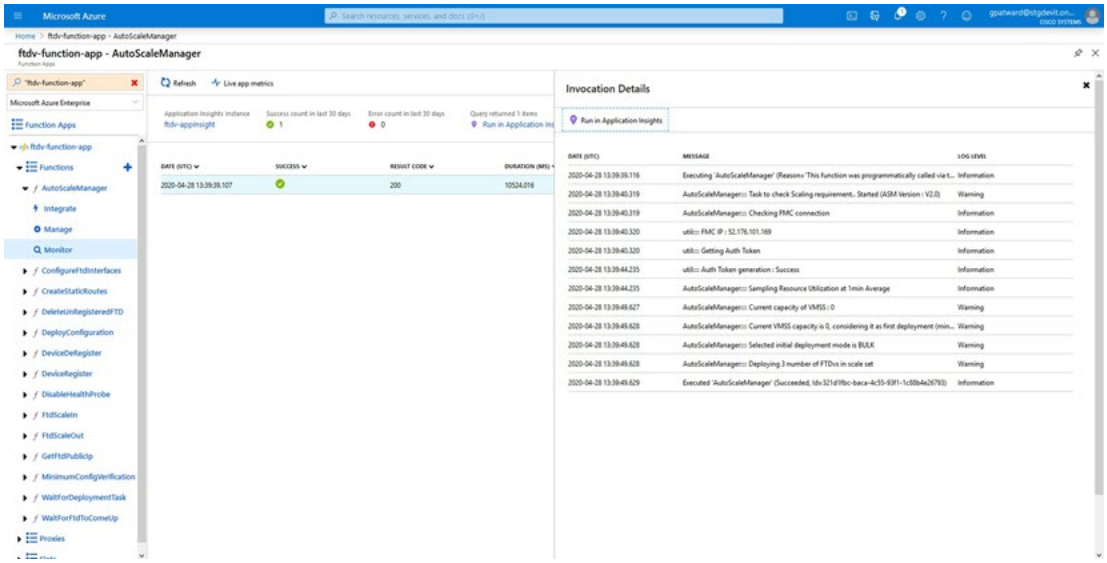
- Scale-In/Scale-Out occurs in steps of 1 (i.e. only 1 Firewall Threat Defense Virtual will be scaled in/out at a time).
- The memory consumption metric received from the Firewall Management Center is not an average value calculated over time, but rather an instantaneous snapshot/sample value. Therefore, the memory metric alone cannot be considered in making scaling decisions. You do not have the option to use a memory-only metric during deployment.

# Auto Scale Logging and Debugging

Each component of the serverless code has its own logging mechanism. In addition, logs are published to application insight.

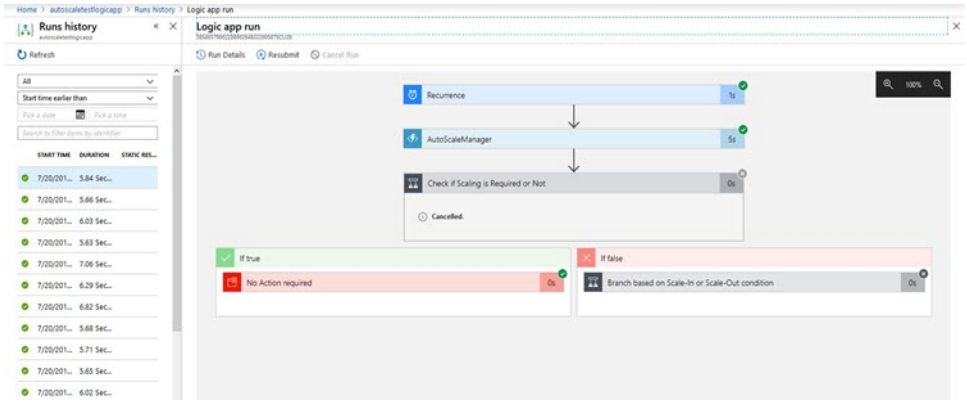
- Logs of individual Azure functions can be viewed.

***Figure 31: Azure Function Logs***




- Similar logs for each run of the Logic App and its individual components can be viewed.

*Figure 32: Logic App Run Logs*




- If needed, any running task in the Logic App can be stopped/terminated at any time. However, currently running Firewall Threat Defense Virtual devices getting launched/terminated will be in an inconsistent state.
- The time taken for each run/individual task can be seen in the Logic App.
- The Function App can be upgraded at any time by uploading a new zip. Stop the Logic App and wait for all tasks to complete before upgrading the Function App.

# Auto Scale Guidelines and Limitations

Be aware of the following guidelines and limitations when deploying the Firewall Threat Defense Virtual auto scale for Azure:

- (Version 6.6 and earlier) Scaling decisions are based on CPU utilization.
- (Version 6.7+) Scaling decisions can use either CPU-only utilization, or CPU and memory utilization.
- Firewall Management Center management is required. Firewall Device Manager is not supported.
- The Firewall Management Center should have a public IP address.
- The Firewall Threat Defense Virtual Management interface is configured to have public IP address.
- Only IPv4 is supported.
- Firewall Threat Defense Virtual auto scale for Azure only supports configurations such as Access policies, NAT policies, Platform Settings, etc. which are applied the Device Group and propagated to scaled-out Firewall Threat Defense Virtual instances. You can only modify Device Group configurations using the Firewall Management Center. Device-specific configurations are not supported.
- The ARM template has limited input validation capabilities, hence it is your responsibility to provide the correct input validation.
- The Azure administrator can see sensitive data (such as admin login credentials and passwords) in plain text format inside Function App environment. You can use the *Azure Key Vault* service to secure sensitive data.

- Any changes in configuration won't be automatically reflected on already running instances. Changes will be reflected on upcoming devices only. Any such changes should be manually pushed to already existing devices.
- If you are facing issues while manually updating the configuration on existing instances, we recommend removing these instances from the Scaling Group and replacing them with new instances.

# Troubleshooting

The following are common error scenarios and debugging tips for the Firewall Threat Defense Virtual auto scale for Azure:

- Connection to the Firewall Management Center failed: Check the Firewall Management Center IP / Credentials; check if the Firewall Management Center is faulty / unreachable.
- Unable to SSH into the Firewall Threat Defense Virtual: Check if a complex password is passed to the Firewall Threat Defense Virtual via the template; check if Security Groups allow SSH connections.
- Load Balancer Health check failure: Check if the Firewall Threat Defense Virtual responds to SSH on data interfaces; check Security Group settings.
- Traffic issues: Check Load Balancer rules, NAT rules / Static routes configured in Firewall Threat Defense Virtual; check Azure virtual network / subnets / gateway details provided in the template and Security Group rules.
- The Firewall Threat Defense Virtual failed to register with the Firewall Management Center: Check the Firewall Management Center capacity to accommodate new Firewall Threat Defense Virtual devices; check Licensing; check the Firewall Threat Defense Virtual version compatibility.
- Logic App failed to access VMSS: Check if the IAM role configuration in VMSS is correct.
- Logic App runs for very long time: Check SSH access on scaled-out Firewall Threat Defense Virtual devices; check any device registration issues in Firewall Management Center; check the state of the Firewall Threat Defense Virtual devices in Azure VMSS.
- Azure Function throwing error related to subscription ID : Verify that you have a default subscription selected in your account.
- Failure of Scale-In operation: Sometimes, Azure takes a considerably long time to delete an instance in such situations, Scale-in operation may time out and report an error; but eventually the instance, will get deleted.
- Before doing any configuration change, make sure to disable the logic application and wait for all the running tasks to complete.

The following are troubleshooting tips if you encounter any issues during Firewall Threat Defense Virtual auto scale with Azure GWLB deployment:

- Check the ELB-GWLB association.
- Check the health probe status in the GWLB.
- Check VXLAN configuration by verifying the traffic flow at the physical and logical interfaces of the Firewall Threat Defense Virtual.
- Check security group rules.

# Build Azure Functions from Source Code

### System Requirements

- Microsoft Windows desktop/laptop.
- Visual Studio (tested with Visual studio 2019 version 16.1.3)

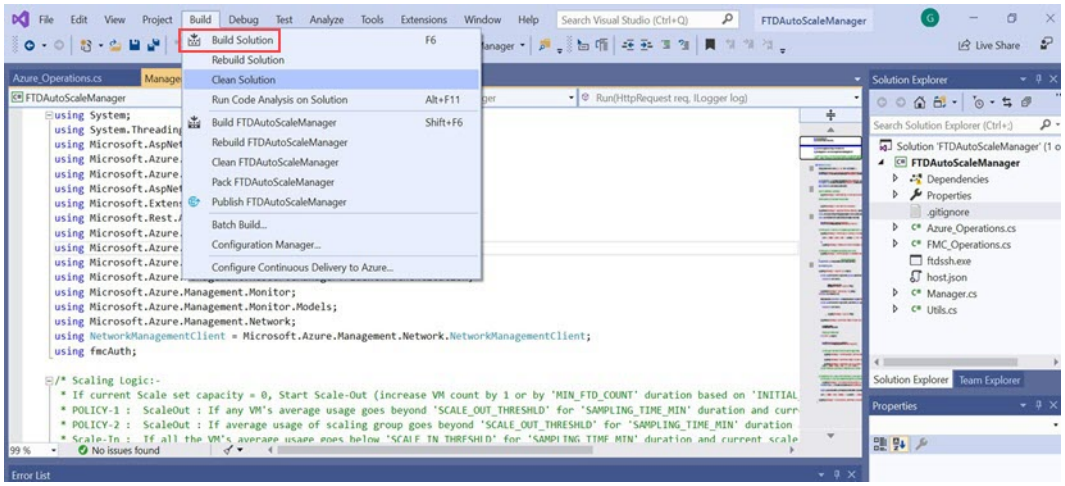
**Note** Azure functions are written using C#.

- The "Azure Development" workload needs to be installed in Visual Studio.

### Build with Visual Studio

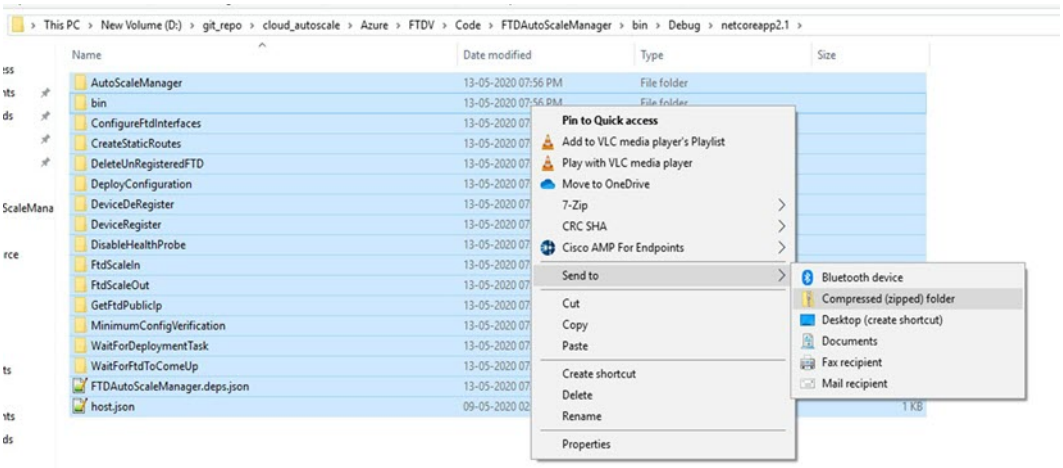
1. Download the 'code' folder to the local machine.
2. Navigate to the folder 'FTDAutoScaleManager'.
3. Open the project file 'FTDAutoScaleManager.csproj' in Visual Studio.
4. Use Visual Studio standard procedure to Clean and Build.

***Figure 33: Visual Studio Build***




5. Once the build is compiled successfully, navigate to the **\bin\Release\netcoreapp2.1** folder.
6. Select all the contents, click **Send to** > **Compressed (zipped) folder**, and save the ZIP file as *ASM_Function.zip*.

*Figure 34: Build ASM_Function.zip*




# Deploy the IPv6 Supported Secure Firewall Threat Defense Virtual on Azure

This chapter explains how to deploy the IPv6 Supported Firewall Threat Defense Virtual from the Azure portal.

# About IPv6 Supported Deployment on Azure

Firewall Threat Defense Virtual offerings support both IPV4 and IPv6 from 7.3 and later. In Azure, you can deploy Firewall Threat Defense Virtual directly from the Marketplace offering, which creates or uses a virtual network, but currently, a limitation in Azure restricts the Marketplace application offer to use or create only IPv4-based VNet/subnets. Although, you can manually configure the IPv6 addresses to the existing VNet, a new Firewall Threat Defense Virtual instance cannot be added to the VNet configured with the IPv6 subnets. Azure imposes certain restrictions to deploy any third-party resources using an alternative approach other than deploying resources through Marketplace.

Cisco is currently offering two methods to deploy Firewall Threat Defense Virtual to support IPv6 addressing.

The following two distinct custom IPv6 templates are offered, where:

- **Custom IPv6 template (ARM template)** — It is offered to deploy Firewall Threat Defense Virtual with IPv6 configuration using an Azure Resource Manager (ARM) template that internally refers to a marketplace image on Azure. This template contains JSON files with resources and parameter definitions that you can configure to deploy IPv6-supported Firewall Threat Defense Virtual . To use this template, see Deploy from Azure Using Custom IPv6 Template with Marketplace Image Reference, on page 220.

  Programmatic deployment is a process of granting access to the VM images on Azure Marketplace to deploy custom templates through PowerShell, Azure CLI, ARM template, or API. You are restricted to deploy these custom templates on VM without providing access to VMs. If you attempt to deploy such custom templates on VM, then the following error message is displayed:

*Legal terms have not been accepted for this item on this subscription. To accept legal terms ….and configure programmatic deployment for the Marketplace item …..*

You can use one of the following methods to enable Programmatic deployment in Azure to deploy the custom IPv6 (ARM) template refering to the marketplace image:

- **Azure Portal** – Enable programmatic deployment option corresponding to the Firewall Threat Defense Virtual offering available on Azure Marketplace for deploying the custom IPv6 template (ARM template).
- **Azure CLI** – Run the CLI command to enable programmatic deployment for deploying the custom IPv6 (ARM template).

- **Custom VHD image and IPv6 template (ARM template)** — Create a managed image using the VHD image and ARM template on Azure. This process is similar to deploying Firewall Threat Defense Virtual by using a VHD and resource template. This template refers to a managed image during deployment and uses an ARM template which you can upload and configure on Azure to deploy IPv6-supported Firewall Threat Defense Virtual . See, Deploy from Azure Using a VHD and Custom IPv6 Template, on page 225.

The process involved in deploying Firewall Threat Defense Virtual using custom IPv6 template (ARM template) in reference to marketplace image or VHD image with custom IPv6 template.

The steps involved in deploying the Firewall Threat Defense Virtual is as follows:

***Table 22:***

| Step | Process |
|---|---|
| 1 | Create a Linux VM in Azure where you are planning to deploy the IPv6-supported Firewall Threat Defense Virtual |
| 2 | Enable Programmatic deployment option on Azure portal or Azure CLI **only** when you are deploying Firewall Threat Defense Virtual using the custom IPv6 template with Marketplace image reference. |
| 3 | Depending on the type of deployment download the following custom templates:<br>• Custom IPv6 Template with Azure Marketplace reference image.<br>VHD image with custom IPv6 (ARM) template. |
| 4 | Update the IPv6 parameters in the custom IPv6 (ARM) template.<br>**Note**<br>The equivalent Software image version parameter value of the marketplace image version is required only when you are deploying Firewall Threat Defense Virtual using the custom IPv6 template with Marketplace image reference. You must run a command to retrieve the Software version details. |
| 5 | Deploy the ARM template through Azure portal or Azure CLI. |

# Deploy from Azure Using Custom IPv6 Template with Marketplace Image Reference

The process involved in deploying Firewall Threat Defense Virtual using custom IPv6 template (ARM template) in reference to marketplace image.

### Procedure

**Step 1** Log into the Azure portal.

The Azure portal shows virtual elements associated with the current account and subscription regardless of data center location.

**Step 2** Enable Programmatic deployment through Azure portal or Azure CLI as follows:

To enable this option on Azure Portal.

a) Under **Azure Services**, click **Subscriptions** to view the subscription blade page.
b) On the left pane, click **Programmatic Deployment** under the **Settings** option.

All the types of resources deployed on the VM are displayed along with the associated subscription offerings.

c) Click **Enable** under the **Status** column and corresponding to the Firewall Threat Defense Virtual offering to obtain for programmatic deployment of the custom IPv6 template.

OR

To enable this option through Azure CLI.

a) Go to the Linux VM.
b) Run the following CLI command to enable programmatic deployment for deploying custom IPv6 (ARM) template.

During the command execution, you must only accept the terms once per subscription of the image.

**# Accept terms**

**az vm image terms accept -p <publisher> -f <offer> --plan <SKU/plan>**

**# Review that terms were accepted (i.e., accepted=true)**

**az vm image terms show -p <publisher> -f <offer> --plan <SKU/plan>**

Where,

- **<publisher>** - 'cisco'.
- **<offer>** - 'cisco-ftdv'
- **<sku/plan>** - 'ftdv-azure-byol'

The following is a command script example to enable programmatic deployment for deploying Firewall Threat Defense Virtual with BYOL subscription plan.

- **az vm image terms show -p cisco -f cisco-ftdv --plan ftdv-azure-byol**

**Step 3** Run the following command to retrieve the Software version details equivalent to the marketplace image version.

**az vm image list --all -p <publisher> -f <offer> -s <sku>**

Where,

- **<publisher>** - 'cisco'.
- **<offer>** - 'cisco-ftdv'
- **<sku>** - 'ftdv-azure-byol'

The following is a command script example to retrieve the Software version details equivalent to the marketplace image version for Firewall Threat Defense Virtual .

**az vm image list --all -p cisco -f cisco-ftdv -s ftdv-azure-byol**

**Step 4** Select one of the Firewall Threat Defense Virtual version from the list of available marketplace image versions that are displayed.

For IPv6 support deployment of Firewall Threat Defense Virtual , you must select the Firewall Threat Defense Virtual version as 73* or higher.

**Step 5** Download the marketplace custom IPv6 template (ARM templates) from the Cisco GitHub repository.

**Step 6** Prepare the parameters file by providing the deployment values in the parameters template file (JSON).

The following table describes the deployment values you need to enter in the custom IPv6 template parameters for Firewall Threat Defense Virtual custom deployment:

| Parameter Name | Examples of allowed Values/Type | Description |
|---|---|---|
| vmName | csf-tdv | Name the Firewall Threat Defense Virtual VM in Azure. |
| softwareVersion | 730.33.0 | The software version of the marketplace image version. |
| billingType | BYOL | The licensing method is BYOL or PAYG.<br>BYOL license is more cost effective compared to PAYG, hence it is recommended to opt for BYOL subscribed deployment. |
| adminUsername | hjohn | The username to log into Firewall Threat Defense Virtual .<br>You cannot use the reserved name ‘admin’, which is assigned to administrator. |
| adminPassword | E28@4OiUrhx! | The admin password.<br>Password combination must be an alphanumeric characters with 12 to 72 characters long. The |

| Parameter Name | Examples of allowed Values/Type | Description |
|---|---|---|
| | | password combination must comprise of lowercase and uppercase letters, numbers and special characters. |
| vmStorageAccount | hjohnvmsa | Your Azure storage account. You can use an existing storage account or create a new one. The storage account characters must be between three and 24 characters long. The password combination must contain only lowercase letters and numbers. |
| availabilityZone | 0 | Specify the availability zone for deployment, public IP and the virtual machine will be created in the specified availability zone.<br>Set it to '**0**' if you do not need availability zone configuration. Ensure that selected region supports availability zones and value provided is correct. (This must be an integer between 0-3). |
| customData | `{\"AdminPassword\": \"E28@4OiUrhx!\",\"Hostname\":\"cisco-tdv\", \"ManageLocally\":\"No\", \"IPv6Mode\": \"DHCP\"}` | The field to provide in the Day 0 configuration to the Firewall Threat Defense Virtual . By default it has the following three key-value pairs to configure:<br>• 'admin' user password<br>• Firewall Management Center Virtual hostname<br>• the Firewall Management Center Virtual hostname or CSF-DM for management.<br>'ManageLocally : yes' - This configures the CSF-DM to be used as Firewall Threat Defense Virtual manager.<br>You can configure the Firewall Management Center Virtual as Firewall Threat Defense Virtual |

| Parameter Name | Examples of allowed Values/Type | Description |
|---|---|---|
| | | manager and also give the inputs for fields required to configure the same on Firewall Management Center Virtual. |
| virtualNetworkResourceGroup | cisco-tdv-rg | Name of the resource group containing the virtual network. In case virtualNetworkNewOrExisting is new, this value should be same as resource group selected for template deployment. |
| virtualNetworkName | cisco-tdv-vent | The name of the virtual network. |
| virtualNetworkNewOrExisting | new | This parameter determines whether a new virtual network should be created or an existing virtual network is to be used. |
| virtualNetworkAddressPrefixes | 10.151.0.0/16 | IPv4 address prefix for the virtual network, this is required only if 'virtualNetworkNewOrExisting' is set to 'new'. |
| virtualNetworkv6AddressPrefixes | ace:cab:deca::/48 | IPv6 address prefix for the virtual network, this is required only if 'virtualNetworkNewOrExisting' is set to 'new'. |
| Subnet1Name | mgmt | Management subnet name. |
| Subnet1Prefix | 10.151.1.0/24 | Management subnet IPv4 Prefix, this is required only if 'virtualNetworkNewOrExisting' is set to 'new'. |
| Subnet1IPv6Prefix | ace:cab:deca:1111::/64 | Management subnet IPv6 Prefix, this is required only if 'virtualNetworkNewOrExisting' is set to 'new'. |
| subnet1StartAddress | 10.151.1.4 | Management interface IPv4 address. |
| subnet1v6StartAddress | ace:cab:deca:1111::6 | Management interface IPv6 address. |
| Subnet2Name | diag | Data interface 1 subnet name. |

| Parameter Name | Examples of allowed Values/Type | Description |
|---|---|---|
| Subnet2Prefix | 10.151.2.0/24 | Data interface 1 Subnet IPv4 prefix, this is required only if 'virtualNetworkNewOrExisting' is set to 'new'. |
| Subnet2IPv6Prefix | ace:cab:deca:2222::/64 | Data interface 1 Subnet IPv6 Prefix, this is required only if 'virtualNetworkNewOrExisting' is set to 'new'. |
| subnet2StartAddress | 10.151.2.4 | Data interface 1 IPv4 address. |
| subnet2v6StartAddress | ace:cab:deca:2222::6 | Data interface 1 IPv6 address. |
| Subnet3Name | inside | Data interface 2 subnet name. |
| Subnet3Prefix | 10.151.3.0/24 | Data interface 2 Subnet IPv4 Prefix, this is required only if 'virtualNetworkNewOrExisting' is set to 'new'. |
| Subnet3IPv6Prefix | ace:cab:deca:3333::/64 | Data interface 2 Subnet IPv6 Prefix, this is required only if 'virtualNetworkNewOrExisting' is set to 'new'. |
| subnet3StartAddress | 10.151.3.4 | Data interface 2 IPv4 address. |
| subnet3v6StartAddress | ace:cab:deca:3333::6 | Data interface 2 IPv6 address. |
| Subnet4Name | outside | Data interface 3 subnet name. |
| Subnet4Prefix | 10.151.4.0/24 | Data interface 3 subnet IPv4 Prefix, this is required only if 'virtualNetworkNewOrExisting' is set to 'new' |
| Subnet4IPv6Prefix | ace:cab:deca:4444::/64 | Data interface 3 Subnet IPv6 Prefix, this is required only if 'virtualNetworkNewOrExisting' is set to 'new'. |
| subnet4StartAddress | 10.151.4.4 | Data interface 3 IPv4 Address. |
| subnet4v6StartAddress | ace:cab:deca:4444::6 | Data interface 3 IPv6 Address. |
| vmSize | Standard_D4_v2 | Size of the Firewall Threat Defense Virtual VM. Standard_D3_v2 is the default. |

**Step 7** Use the ARM template to deploy Firewall Threat Defense Virtual firewall through the Azure portal or Azure CLI. For information about deploying the ARM template on Azure, refer to the following Azure documentation:

- Create and deploy ARM templates by using the Azure portal
- Deploy a local ARM template through CLI

### What to do next

Your next steps depend on what management mode you chose.

- If you chose **No** for **Enable Local Manager**, you'll use the Secure Firewall Management Center to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center.
- If you chose **Yes** for **Enable Local Manager**, you'll use the integrated Secure Firewall Device Manager to manage your Firewall Threat Defense VirtualFirewall Threat Defense VirtualFirewall Threat Defense VirtualFirewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall device manager.

  See How to Manage Your Secure Firewall Threat Defense Virtual Device for an overview of how to choose your management option.

# Deploy from Azure Using a VHD and Custom IPv6 Template

You can create your own custom Firewall Threat Defense Virtual images using a compressed VHD image available from Cisco. This process is similar to deploying Firewall Threat Defense Virtual by using a VHD and resource template.

### Before you begin

- You need the JSON template and corresponding JSON parameter file for your Firewall Threat Defense Virtual deployment using VHD and ARM updated template on Github, where you'll find instructions on how to build a template and parameter file.
- This procedure requires an existing Linux VM in Azure. We recommended you use a temporary Linux VM (such as Ubuntu 16.04) to upload the compressed VHD image to Azure. This image will require about 50GB of storage when unzipped. Also, your upload times to Azure storage will be faster from a Linux VM in Azure.

  If you need to create a VM, use one of the following methods:

  - Create a Linux virtual machine with the Azure CLI
  - Create a Linux virtual machine with the Azure portal

- In your Azure subscription, you should have a storage account available in the Location in which you want to deploy the Firewall Threat Defense Virtual .

## Procedure

**Step 1** Download the Firewall Threat Defense Virtual compressed VHD image (*.bz2) from the Cisco Download Software page:

a) Navigate to **Products** > **Security** > **Firewalls** > **Next-Generation Firewalls (NGFW)** > **Secure Firewall Threat Defense Virtual**.

b) Click **Firepower Threat Defense Software**.

Follow the instructions for downloading the image.

Cisco_Secure_Firewall_Threat_Defense_Virtual-X.X.X-xxx.vhd.bz2

**Step 2** Perform **Step 2** through **Step 8** in Deploy from Azure Using a VHD and Resource Template .

**Step 3** Click **Edit parameters** at the top of the **Custom deployment** page. You have a parameters template that is available for customizing.

a) Click **Load** file and browse to the customized Firewall Threat Defense Virtual parameter file. See the sample for the Azure Firewall Threat Defense Virtual deployment using VHD and custom IPv6 (ARM) template on Github, where you'll find instructions on how to build a template and parameter file.

b) Paste your customized JSON parameters code into the window, and then click **Save**.

The following table describes the deployment values you need to enter in the custom IPv6 template parameters for Firewall Threat Defense Virtual deployment:

| Parameter Name | Examples of allowed values/types | Description |
|---|---|---|
| vmName | csf-tdv | Name the Firewall Threat Defense Virtual VM in Azure. |
| vmImageId | /subscriptions/{subscription-id}/resourceGroups/{resource-group}/providers/Microsoft.Compute/images/{image-name} | The ID of the image used for deployment. Internally, Azure associates every resource with a Resource ID. |
| adminUsername | hjohn | The username to log into Firewall Threat Defense Virtual .<br>You cannot use the reserved name 'admin', which is assigned to administrator. |
| adminPassword | E28@4OiUrhx! | The admin password.<br>Password combination must be an alphanumeric characters with 12 to 72 characters long. The password combination must comprise of lowercase and uppercase letters, numbers and special characters. |
| vmStorageAccount | hjohnvmsa | Your Azure storage account. You can use an existing storage account or create a new one. The storage account characters must be between three and |

| Parameter Name | Examples of allowed values/types | Description |
|---|---|---|
| | | 24 characters long. The password combination must contain only lowercase letters and numbers. |
| availabilityZone | 0 | Specify the availability zone for deployment, public IP and the virtual machine will be created in the specified availability zone.<br>Set it to '**0**' if you do not need availability zone configuration. Ensure that selected region supports availability zones and value provided is correct. (This must be an integer between 0-3). |
| customData | `{\"AdminPassword\": \"E28@4OiUrhx!\",\"Hostname\" :\"cisco-tdv\", \"ManageLocally\":\"No\", \"IPv6Mode\": \"DHCP\"}` | The field to provide in the Day 0 configuration to the Firewall Threat Defense Virtual . By default it has the following three key-value pairs to configure:<br>• 'admin' user password<br>• CSF-MCv hostname<br>• the CSF-MCv hostname or CSF-DM for management.<br>'ManageLocally : yes' - This configures the CSF-DM to be used as Firewall Threat Defense Virtual manager.<br>You can configure the CSF-MCv as Firewall Threat Defense Virtual manager and also give the inputs for fields required to configure the same on CSF-MCv. |
| virtualNetworkResourceGroup | csf-tdv | Name of the resource group containing the virtual network. In case virtualNetworkNewOr Existing is new, this value should be same as resource group selected for template deployment. |
| virtualNetworkName | hjohn-vm-vn | The name of the virtual network. |
| virtualNetworkNewOrExisting | new | This parameter determines whether a new virtual network should be created or an existing virtual network is to be used. |
| virtualNetworkAddressPrefixes | 10.151.0.0/16 | IPv4 address prefix for the virtual network, this is required only if |

| Parameter Name | Examples of allowed values/types | Description |
|---|---|---|
| | | 'virtualNetworkNewOr Existing' is set to 'new'. |
| virtualNetworkv6AddressPrefixes | ace:cab:deca::/48 | IPv6 address prefix for the virtual network, this is required only if 'virtualNetworkNewOr Existing' is set to 'new'. |
| Subnet1Name | mgmt-ipv6 | Management subnet name. |
| Subnet1Prefix | 10.151.1.0/24 | Management subnet IPv4 Prefix, this is required only if 'virtualNetworkNewOr Existing' is set to 'new'. |
| Subnet1IPv6Prefix | ace:cab:deca:1111::/64 | Management subnet IPv6 Prefix, this is required only if 'virtualNetworkNewOr Existing' is set to 'new'. |
| subnet1StartAddress | 10.151.1.4 | Management interface IPv4 address. |
| subnet1v6StartAddress | ace:cab:deca:1111::6 | Management interface IPv6 address. |
| Subnet2Name | diag | Data interface 1 subnet name. |
| Subnet2Prefix | 10.151.2.0/24 | Data interface 1 Subnet IPv4 prefix, this is required only if 'virtualNetworkNewOr Existing' is set to 'new'. |
| Subnet2IPv6Prefix | ace:cab:deca:2222::/64 | Data interface 1 Subnet IPv6 Prefix, this is required only if 'virtualNetworkNewOr Existing' is set to 'new'. |
| subnet2StartAddress | 10.151.2.4 | Data interface 1 IPv4 address. |
| subnet2v6StartAddress | ace:cab:deca:2222::6 | Data interface 1 IPv6 address. |
| Subnet3Name | inside | Data interface 2 subnet name. |
| Subnet3Prefix | 10.151.3.0/24 | Data interface 2 Subnet IPv4 Prefix, this is required only if 'virtualNetworkNewOr Existing' is set to 'new'. |
| Subnet3IPv6Prefix | ace:cab:deca:3333::/64 | Data interface 2 Subnet IPv6 Prefix, this is required only if 'virtualNetworkNewOr Existing' is set to 'new'. |
| subnet3StartAddress | 10.151.3.4 | Data interface 2 IPv4 address. |

| Parameter Name | Examples of allowed values/types | Description |
|---|---|---|
| subnet3v6StartAddress | ace:cab:deca:3333::6 | Data interface 2 IPv6 address. |
| Subnet4Name | outside | Data interface 3 subnet name. |
| Subnet4Prefix | 10.151.4.0/24 | Data interface 3 subnet IPv4 Prefix, this is required only if 'virtualNetworkNewOr Existing' is set to 'new' |
| Subnet4IPv6Prefix | ace:cab:deca:4444::/64 | Data interface 3 Subnet IPv6 Prefix, this is required only if 'virtualNetworkNewOr Existing' is set to 'new'. |
| subnet4StartAddress | 10.151.4.4 | Data interface 3 IPv4 Address. |
| subnet4v6StartAddress | ace:cab:deca:4444::6 | Data interface 3 IPv6 Address. |
| vmSize | Standard_D4_v2 | Size of the Firewall Threat Defense Virtual VM. Standard_D3_v2 is the default. |

**Step 4** Use the ARM template to deploy Firewall Threat Defense Virtual firewall through the Azure portal or Azure CLI. For information about deploying the ARM template on Azure, refer to the following Azure documentation:

- Create and deploy ARM templates by using the Azure portal
- Deploy a local ARM template through CLI

### What to do next

- Update the Firewall Threat Defense Virtual's IP configuration in Azure.

# Deploy the Azure Marketplace offers in the restricted Azure Private Marketplace environment

This applies only for the Azure Private Marketplace users. If you are using Azure Private Marketplace, then ensure that both Application Offers and required Virtual Machine Offers (hidden) are enabled for the user in respective private marketplace.

**Virtual Machine Offers and Plans (hidden):**

- Publisher ID: **cisco**
- Cisco Secure Firewall Threat Defense Virtual VM Offers (used for both the Cisco Secure Firewall Threat Defense Virtual Application offers)

- Offer ID: **cisco-ftdv**
- BYOL Plan ID: **ftdv-azure-byol**
- PAYG Plan ID: **ftdv-azure-payg**

When user deploys the visible application offer from Marketplace, based on user selection of PAYG or BYOL licensing corresponding image from the VM offer plan is referenced and deployed.

Therefore, for the deployment to work, both Application and VM offers needs to be enabled/available on the Private Marketplace for the Azure tenant/subscription.

Refer the Azure documentation for enabling these application and VM offers in private marketplaces.

- Govern and control using private Azure Marketplace
- Add an offer to a private marketplace
- Set-AzMarketplacePrivateStoreOffer

Application offers are easily enabled via Azure UI as they are visible in the marketplace.

In order to enable hidden virtual machine offers in private marketplace, you might have to rely on CLI commands (at the time of this doc creation only CLI way is possible).

**Sample command:**

Cisco Secure Firewall Threat Defense Virtual BYOL plan can be enabled using similar sample command given below:

```
$Params = @{
  privateStoreId = '<private-store-id>'
  offerId = '<publisher-id>.<vm-offer-id>'
  SpecificPlanIdsLimitation =@('<plan-id-under-vm-offer>')
}
Set-AzMarketplacePrivateStoreOffer @Params

$Params = @{
  privateStoreId = '<private-store-id>'
  offerId = 'cisco.cisco-ftdv'
  SpecificPlanIdsLimitation =@('ftdv-azure-byol')
}
Set-AzMarketplacePrivateStoreOffer @Params
```

**Note** The sample command is only for reference, check Azure documentation for more details.

**Reference Error message**

```
{
  "code": "MarketplacePurchaseEligibilityFailed",
  "details": [
    {
      "code": "BadRequest",
     "message": "Offer with PublisherId: 'cisco', OfferId: 'cisco-XXXX' cannot be purchased
 due to validation errors. For more information see details.
Correlation Id: 'XXXXX`
This plan is not available for purchase because it needs to be added to your tenant's Private
 Marketplace. Contact your admin to request adding the plan.
Link to plan: <URL>.
```

```
Plan: '<PLAN NAME>'(planId=<VM-OFFER-PLAN-ID>),
Offer: <OFFER_NAME>, Publisher: 'Cisco Systems, Inc.'(publisherId='cisco').
…
…
    }
  ],
  "message": "Marketplace purchase eligibilty check returned errors. See inner errors for
details. "
}
```

User may run into the above error while deploying the Marketplace offer. To resolve this, both Application and VM offers need to be enabled/available on the Azure tenant/subscription.

# Firewall Threat Defense Virtual Image Snapshot

You can create and deploy the Firewall Threat Defense Virtual using a snapshot image in the Azure portal. The image snapshot is a replicated Firewall Threat Defense Virtual image instance with no state data.

## Firewall Threat Defense Virtual Snapshot Overview

The process of creating a snapshot image of the Firewall Threat Defense Virtual instance helps to minimize the initial system *init* time by skipping the first boot procedures done for the Firewall Threat Defense Virtual and FSIC. The snapshot image consists of prepopulated database and the Firewall Threat Defense Virtual initial boot process, which enables the image to regenerate unique IDs (UUIDs, Serial number) that is related to the system identity in the Firewall Management Center or any other management center. This process helps in faster boot time of Firewall Threat Defense Virtual, which is essential in auto scale deployment.

**Note** Snapshot image creation is used to minimize the initial system init time on a non-registered Secure Firewall Threat Defense Virtual. It should not be used for the purpose of *backup/restore*.

**Note** Currently, Pay-As-You-Go (PAYG) licensing is not supported for instances deployed using a snapshot image of the Firewall Threat Defense Virtual. PAYG licensing is only available for instances that you deploy directly from the Marketplace. You can use Smart Licensing for such new Firewall Threat Defense Virtual deployments with PAYG licensing.

## Create the Firewall Threat Defense Virtual Snapshot Image from Managed Image

Firewall Threat Defense Virtual image snapshot creation is a process of replicating an existing managed image of the Firewall Threat Defense Virtual instance in the Azure portal.

### Before you begin

You must have created a managed image of the Firewall Threat Defense Virtual version 7.2 or later by uploading the resized VHD image to a container in your Azure storage account of a Linux VM in the Azure

portal. For information on creating resized VHD image, see Deploy from Azure Using a VHD and Resource Template, on page 165.

You must not register the Firewall Threat Defense Virtual instance you are preparing for image snapshot to any manager such as the Firewall Management Center or the Firewall Device Manager.

### Procedure

**Step 1** Go to Azure portal where you have created the managed image of the Firewall Threat Defense Virtual instance.

**Note**
Ensure that the Firewall Threat Defense Virtual instance which you are planning to replicate is not registered to the Firewall Management Center or configured to any other local manager or applied with any configuration.

**Step 2** Go to **Resource Group** and select the Firewall Threat Defense Virtual instance.

**Step 3** Click the **Serial Console** on the navigation page of the Firewall Threat Defense Virtual instance.

**Step 4** Use the following scripts to run the pre-snapshot process from the expert shell:

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

When you use `prepare_snapshot` command in the script, an intermediate message appears prompting for confirmation to execute the script. Press **Y** to run the script.

Alternatively, you can append `-f` to this command, such as `root@firepower:/ngfw/var/common# prepare_snapshot -f` to skip the user confirmation message and directly execute the script.

This script removes all the line configurations, deployed policies, configured manager, UUIDs associated with the Firewall Threat Defense Virtual instance. After the processing is done, the Firewall Threat Defense Virtual instance is shut down.

**Step 5** Click **Capture**.

**Step 6** In the **Create an image** page, choose an existing resource group or create a new one from the **Resource Group** drop-down list.

**Step 7** Click **No, capture only a managed image** in the **Instance Details** section to create only a managed image.

**Step 8** Provide name for the snapshot image you are creating using the managed image of the Firewall Threat Defense Virtual instance.

**Step 9** Click **Review+Create** to create a new snapshot image of the Firewall Threat Defense Virtual instance.

### What to do next

Deploy the Firewall Threat Defense Virtual instance using snapshot image. See Deploy Secure Firewall Threat Defenece Virtual using snapshot image.

# Deploy the Firewall Threat Defense Virtual Instance using Image Snapshot

### Before you begin

Cisco recommends the following:

- Confirm that a snapshot image is available for the Firewall Threat Defense Virtual instance.

## Procedure

**Step 1** Log in to Azure portal.

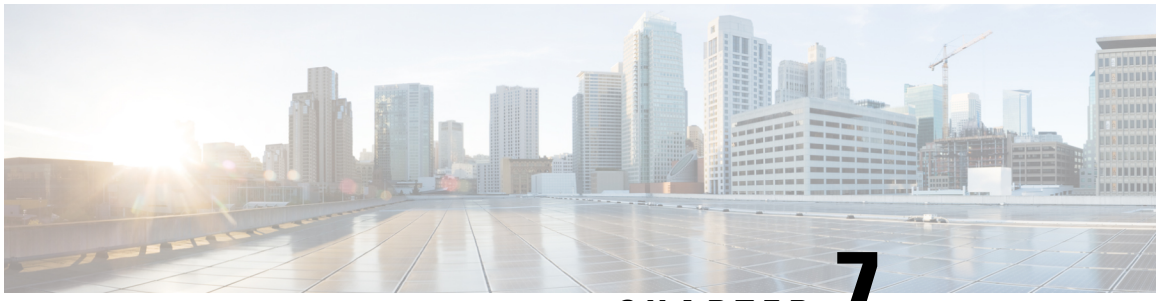
**Step 2** Copy the Resource ID of the newly created snapshot image.

**Note**
Azure associates every resource (snapshot image) with a Resource ID. The Resource ID of the snapshot image is required for deploying the new Firewall Threat Defense Virtual instance.

a) In the Azure Portal, select **Images**.
b) Select the snapshot image you have created by using a managed image.
c) Click **Overview** to view the image properties.
d) Copy the **Resource ID** to the clipboard. The **Resource ID** syntax is represented as: **/subscriptions/<subscription-id>/resourceGroups/<resourceGroup>/providers/Microsoft.Compute/<container>/<vhdname>**

**Step 3** Continue deploying the Firewall Threat Defense Virtual instance using the snapshot image. See Deploy from Azure Using a VHD and Resource Template, on page 165.

**Note**
You can run the CLI commands **show version** and **show snapshot detail** from the Firewall Threat Defense Virtual console to know about the version and details of the newly deployed Firewall Threat Defense Virtual instance.



CHAPTER 7

# Deploy the Firewall Threat Defense Virtual on Azure Virtual WAN

This chapter explains how to deploy Firewall Threat Defense Virtual instances using a solution template on Azure Virtual WAN.


- Introduction to Threat Defense Virtual in Azure Virtual WAN, on page 235
- System Requirements, on page 236
- Limitations, on page 237
- Network Topology, on page 238
- Deploy Threat Defense Virtual on Azure Virtual WAN, on page 241
- Configure Threat Defense Virtual in Management Center, on page 248
- Register Threat Defense Virtual Instances in the Management Center, on page 249
- Configure Interfaces, on page 249
- Configure Route for Interfaces, on page 250
- Configure Traffic Routing, on page 253
- Configure Health Probe, on page 255
- Troubleshooting, on page 256


## Introduction to Threat Defense Virtual in Azure Virtual WAN

Microsoft Azure Virtual WAN employs a 'hub-and-spoke' architecture to manage traffic across various virtual networks and branch locations. Within the Azure Virtual WAN, integrating Threat Defense Virtual with the Azure Virtual hub facilitates the efficient management and inspection of traffic originating from your organization's on-premises (spoke) networks (like headquarters, branches, and remote users) as it passes through the hub to access Vnets on your Azure network. This integration facilitates the management, inspection, filtering, and routing of network traffic through dedicated connectivity channels using Threat Defense Virtual functioning as firewall.

**Note** Threat Defense Virtual deployment model with only three interfaces is supported by Azure Virtual WAN.

Deploying Threat Defense Virtual on the Azure Virtual WAN hub offers several advantages, including:

- Eliminating the need to implement a firewall solution in each spoke connected to the hub.

- Leveraging Azure's inbuilt capabilities of Internal Load Balancer (ILB).
- Scaling of instances with predefined configuration during deployment.

### Traffic Routing Through Threat Defense Virtual on Azure Virtual WAN

#### Routing Traffic Methods in Azure Virtual WAN

Azure Virtual WAN offers Border Gateway Protocol (BGP), a dynamic routing protocol that helps determine the best route to send traffic between different Azure networks while constantly updating and sharing the routing table. The virtual WAN hub provides a set of BGP endpoints (for High Availability) and Autonomous System Number (ASN), which you must configure as BGP neighbors for Threat Defense Virtual in the management center.

You can also use the static routing method to manually configure routes in the Threat Defense Virtual.

For more information on routing in Azure, see About BGP and VPN Gateway in the Azure documentation.

#### Routing Intent

Routing Intent is a routing ability in the Azure Virtual WAN hub that simplifies the process of forwarding Internet-bound and Private traffic to the Threat Defense Virtual firewall deployed in the hub for inspection.

For more information, see Routing Intent in the Azure documentation.

# System Requirements

### Scaling Units

The scaling required to achieve maximum throughput depends on the instance size and number of Firewall Threat Defense Virtual instances (NVA) you select or configure during deployment in the Azure Virtual WAN hub.

**For example**: If two Firewall Threat Defense Virtual instances with **D3_V2** size can support 3.2 Gbps, then the NVA throughput is defined as **Scale-Unit-4: 3.2 Gbps**.

***Table 23: Firewall Threat Defense Virtual Throughput Level Based on Instance Type***

| Scale Unit | Threat Defense Virtual instances | Instance Type | Throughput Support Level |
|---|---|---|---|
| 4 | 2 | Standard_D3_v2 | 3.2 Gbps |
| 10 | 2 | Standard_D4_v2 | 4.8 Gbps |
| 20 | 2 | Standard_D5_v2 | 12 Gbps |
| 40 | 3 | Standard_D5_v2 | 18 Gbps |
| 60 | 4 | Standard_D5_v2 | 24 Gbps |
| 80 | 5 | Standard_D5_v2 | 30 Gbps |

# Limitations

### Interfaces

Firewall Threat Defense Virtual in an Azure Virtual WAN hub supports *Three* interfaces for deployment due to the restriction by Azure that an NVA can only support a maximum of three network interfaces.

**Note** Firewall Threat Defense Virtual version 7.4.1 and later that supports the three interface models is compatible for deployment on Azure Virtual WAN.

Three subnets for the Firewall Threat Defense Virtual network interfaces are as follows:

- **Management interfaces** – It is the **first interface** that connects the Firewall Threat Defense Virtual to the management center using a public IP address.
- **Outside interface (required)** - It is the **second interface** that connects the Firewall Threat Defense Virtual to an untrusted public IP address.
- **Inside interface (required)** - It is the **third interface** that connects the Firewall Threat Defense Virtual to the Virtual WAN hub and inside the host network on a trusted private IP address.

### Firewall Threat Defense Virtual as Network Virtual Appliance (NVA)

The following are key features that are related to the network configuration of Firewall Threat Defense Virtual as NVA in Azure Virtual WAN.

- Azure internally creates the VNet and subnets during the deployment of Firewall Threat Defense Virtual on Azure Virtual WAN. As a result, you cannot modify or create them after the deployment is complete. However, you can view all the IP addresses attached to the instance after the deployment.
- You cannot choose the ports in network security groups for each interface, however, these ports are predefined during deployment. Only TCP ports 443, 8305, and 22 are allowed on the Management interface to connect to the internet.
- The Inside interface only allows communication within the Azure Virtual WAN hub and internal networks that are connected to it.

### Access Restriction to Firewall Threat Defense Virtual on the Azure Virtual WAN Hub

You require authorization to access the Firewall Threat Defense Virtual instances that are deployed on the hub as a managed application into a managed resource group. The administrator can grant limited or restricted access to this managed resource group.

Azure managed applications offers a just-in-time (JIT) access feature that allows you to define access to managed applications. For information on the JIT, see Azure Managed Applications overview and just-in-time in the Azure documentation.

### IP Support

- Only IPv4 is supported.

### Unsupported Features

- Bootstrapping via Day 0 / Custom data is not supported.
- Firewall Threat Defense Virtual does not support streaming metrics to Azure.
- Virtual Machine upgrade by replacing the operating system disk is not supported.
- SSH key-based login to Firewall Threat Defense Virtual is not supported.
- PAYG is not supported.

### Licensing

BYOL using a Cisco Smart License Account.

# Network Topology

Firewall Threat Defense Virtual, as an NVA in the Azure Virtual WAN hub, inspects network traffic routing through the hub from different on-premises networks (spoke) such as Internet, Branch (Sites), or as VNETs.

These traffic routes through which the network traffic is traversing is categorized into the following topologies:

- East-West: Branch to Branch
- East-West: VNet to VNet
- North-South: Branch to Internet
- North-South: VNet to Internet

**Note** Traffic routing from **Internet** to **VNet or Branch** through Firewall Threat Defense Virtual is not supported.

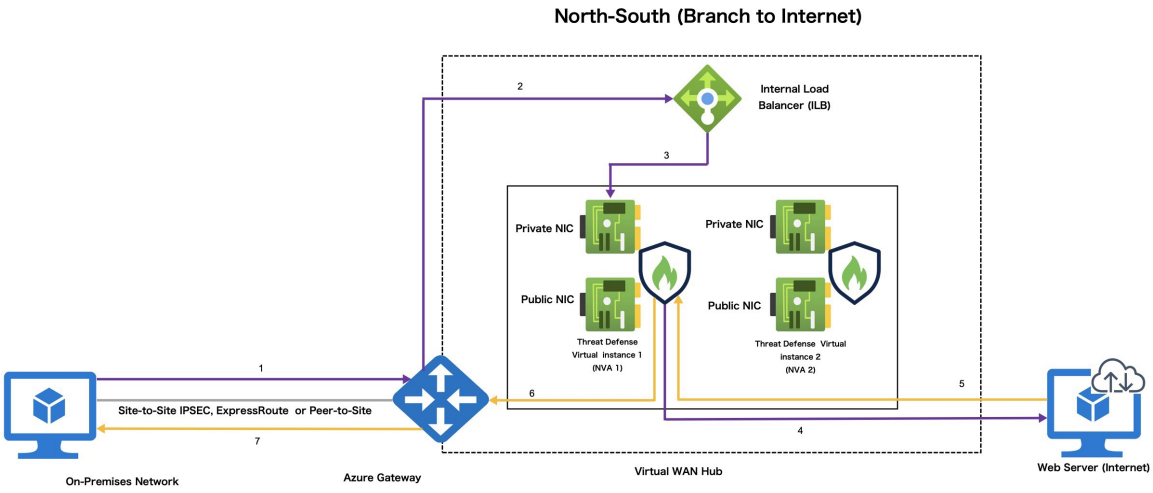
**Note** You can deploy multiple hubs across the Azure regions and connect to a Virtual WAN. Also, you can configure each hub to have its own Firewall Threat Defense Virtual for East-West and North-South traffic inspection.

### North-South Traffic Inspection Topology by Firewall Threat Defense Virtual on a Single Virtual WAN Hub

This topology refers to Firewall Threat Defense Virtual inspecting the network traffic navigating between:

- Branches and VNETs, and vice versa are connected to the Virtual WAN hub.

*Figure 35: Firewall Threat Defense Virtual North-South Traffic Inspection Topology in Azure Virtual WAN Hub*




The following steps explain the traffic flow process in the North-South traffic inspection.

1. On-premises network sends traffic to Azure Gateway.
2. Gateway forwards to ILB.
3. ILB sends to Firewall Threat Defense Virtual (NVA)
4. NVA SNATS to instances PIP and sends to the Internet.
5. Web server replies to instance PIP Firewall Threat Defense Virtual (NVA) undoes SNAT and forwards to gateway.
6. Gateway forwards to on-premises network.

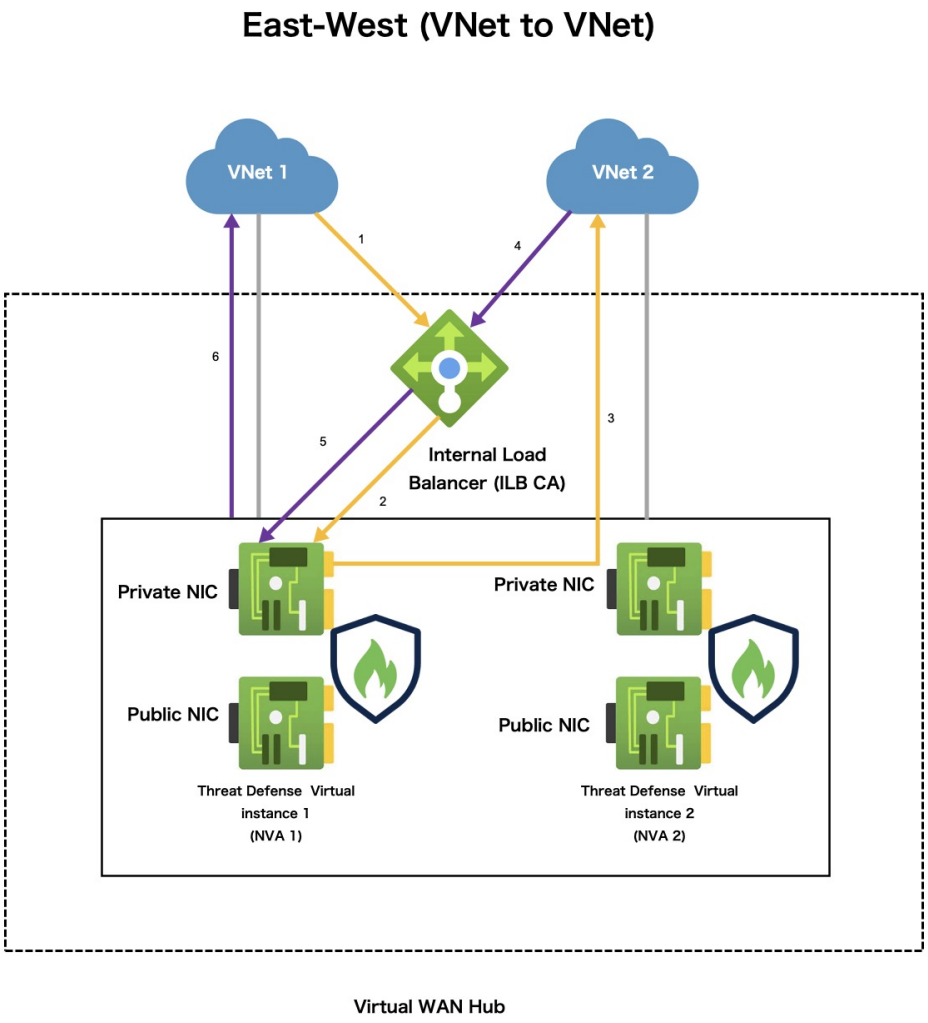
### East-West Traffic Inspection Topology by Firewall Threat Defense Virtual on a Single Virtual WAN Hub

This topology refers to Firewall Threat Defense Virtual inspecting the network traffic navigating between:

- Branches and VNETs, and vice versa are connected to the Virtual WAN hub.
- Internet to Branch or VNETs connected to the Virtual WAN hub.

*Figure 36: Firewall Threat Defense Virtual East-West Traffic Inspection Topology in Azure Virtual WAN Hub*

This topology refers to Firewall Threat Defense Virtual inspecting the network traffic navigating between Site-to-Site (Branch and Branch) and VNET-to-VNET that are connected to the Virtual WAN hub.




The following steps explains the traffic flow process in the East-West traffic inspection.

1. VNet1 sends traffic to ILB.
2. ILB chooses one of the active instances.
3. Firewall Threat Defense Virtual (NVA) sends directly to the destination (VNet 2).
4. VNet sends traffic to ILB.
5. ILB forwards traffic to the appropriate Firewall Threat Defense Virtual (NVA) state fully.
6. Firewall Threat Defense Virtual (NVA) sends traffic back to VNet 1.

# Deploy Threat Defense Virtual on Azure Virtual WAN

You can use the Cisco Secure Firewall Threat Defense Virtual for Azure Virtual WAN offering that is available on Azure Marketplace to deploy Threat Defense Virtual on the Azure Virtual WAN hub.

## Prerequisites

- A Microsoft Azure account. You can create one at https://azure.microsoft.com/en-us/.
- Create a hub on your Virtual WAN. For information on creating a virtual hub in Azure, see Create a hub in the Azure documentation.
- Ensure that the Virtual WAN hub address space is less than or equal to /23.

**Note** Microsoft Azure allows Virtual WAN hubs with /24 address spaces. However, Microsoft does not recommend the deployment of such hubs due to future enhancements. We do not support deploying Firewall Threat Defense Virtual in a Virtual WAN hub with a /24 address space.

- A Cisco Smart Account. You can create one at Cisco Software Central.

**Note** Post deployment of Threat Defense Virtual instances, you can view all the public and private IPs attached to the instance.

### Communication Paths

- Management interface—Used to connect the Threat Defense Virtual to the Management Center.
- Inside interface (required)—Used to connect the Threat Defense Virtual to inside hosts.
- Outside interface (required)—Used to connect the Threat Defense Virtual to the public network.

## Role-Based Access Control

A managed identity is required for deploying a Firewall Threat Defense Virtual as a Network Virtual Appliance (NVA) on Azure Virtual WAN. It provides a secure and efficient way for the NVA to access the necessary Azure resources without requiring explicit credentials by leveraging a role-based access control and management. This setup is a one-time operation per subscription, and the same managed identity can be reused for any Threat Defense Virtual (NVA) deployments within the same subscription.

### Create Managed Identity

You need to create a unique ID to a resource group of a subscription. Ensure to have a resource group created before creating the managed identity.

### Procedure

**Step 1** Log in to the Microsoft Azure portal using your Microsoft account username and password.

**Step 2** Search for **Managed Identities** and click the service. The **Managed Identities** page is displayed.

**Step 3** Click **Create**.

**Step 4** Enter the values in the following fields:

- Select the **Subscription** and **Resource Group** where you are planning to deploy the Firewall Threat Defense Virtual (NVA). This ID will be assigned to the Threat Defense Virtual (NVA) during deployment to provide access to the resource group.
- Enter the **Instance Details**.

**Step 5** Click **Next**.

**Step 6** Enter the necessary **Name** and **Values** in the **Tags** menu.

**Step 7** Click **Next** to review the managed identity details.

**Step 8** Click **Review + create** to create the managed identity. The newly created managed identity is displayed in the **Managed Identities** page.

## Create Custom Role

You must setup a role within an Azure resource to manage access to the resource groups or subscription. This involves creating a role configured with a necessary permission for accessing these Azure resources.

### Procedure

**Step 1** Go to **Subscriptions** service in the Azure portal.

**Step 2** Click the **Subscription** where you want to create a custom role.

**Step 3** Click **Access control (IAM)**.

**Step 4** Click **Add custom role** from the **Add** drop-down list.

**Step 5** Enter the values in the following fields:

- Custom role name and description: Enter a unique and description for the role.
- Baseline permissions: Click **Start from JSON**.

**Step 6** Click **Next** to proceed to the **Assignable scopes** page.

**Step 7** Review and ensure the scope is assigned to the desired subscription.

**Step 8** Click **Next**.

**Step 9** Edit the JSON and update the **actions** field with the following permissions.

```
"actions": [
"Microsoft.Network/virtualHubs/read",
"Microsoft.Network/publicIPAddresses/join/action",
"Microsoft.Network/networkVirtualAppliances/*",
```

```
"Microsoft.Network/networkVirtualAppliances/inboundSecuri
tyRules/*"
]
```

**Step 10** Click **Save**.

**Step 11** Click **Next** to review the custom role details.

**Step 12** Click **Review + create**. The newly created role is displayed in **Roles** menu of **Access control (IAM)** page.

## Assign Custom Role

After creating the custom role, you can assign the role associated to the subscription or resource group to the Managed identity.

### Procedure

**Step 1** Go to the **Managed Identities** page.

**Step 2** Click the identity you want to assign the custom role. The identity page is displayed.

**Step 3** Click **Azure role assignment** from menu on the left pane.

**Step 4** Click **Add role assignment**. The **Add role assignment** dialog box is displayed.

**Step 5** Click **Subscription** from the **Scope** drop-down list. Note that the managed identity you have created is already associated to the subscription and the resource group.

**Step 6** Select the custom role from the **Role** drop-down list to assign to the managed identity. Note that you must assign the custom role that you had newly created for this subscription.

**Step 7** Click **Save** to save the role assignment.
You can see the role assigned to the identity in the Assigned role column next to the identity in the Managed Identities page.

### What to do next

Deploy Threat Defense Virtual on Azure Virtual WAN Using Solution Template, on page 243.

# Deploy Threat Defense Virtual on Azure Virtual WAN Using Solution Template

The following instructions show how to deploy the Threat Defense Virtual on Azure Virtual WAN using the solution template that is available in the Azure Marketplace. This is a top-level list of steps to set up the Threat Defense Virtual in the Microsoft Azure Virtual WAN environment.

For more information about the Azure setup, see Getting Started with Azure.

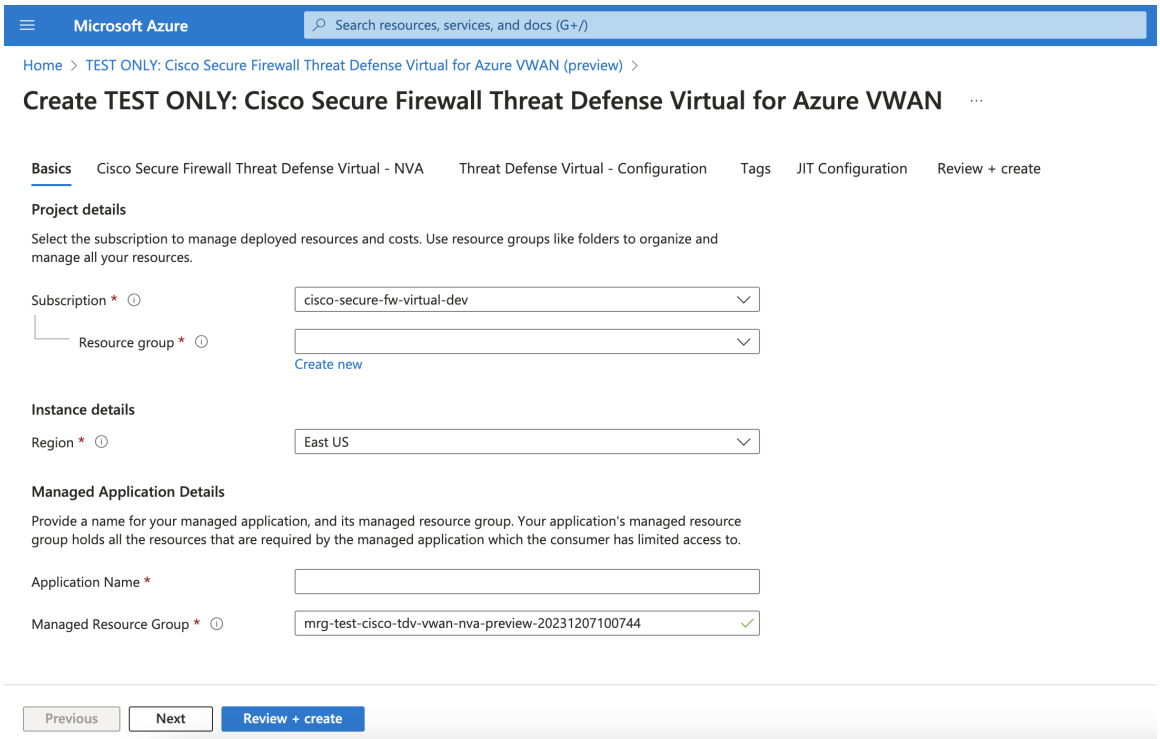
### Procedure

**Step 1** Log in to the Azure Resource Manager (ARM) portal.

The Azure portal shows virtual elements that are associated with the current account and subscription regardless of data center location.

**Step 2** Choose **Azure Marketplace** > **Virtual Machines**.

**Step 3** Search the Marketplace for **Cisco Secure Firewall Threat Defense Virtual for Azure VWAN**, choose the offering, and click **Create** to display the **Basics** page.




**Step 4** Configure the **Basics** settings.

a) Choose your subscription.

b) Select the geographical location or region of the Virtual WAN hub. Each deployment will have multiple resoorces such as virtual WAN hub, Threat Defense Virtual, Network, and storage accounts. Ensure that you select the same geographical location for all the resources.

**Step 5** Configure **Managed Application Details** settings.

a) Enter a name for managed application.

b) Select the managed resource group where you deploy the Threat Defense Virtual instance.

**Step 6** Configure the user assigned managed identity.

a) Click **Next** to display the **NVA Application Settings** page.

b) Click **Add** to assign a managed identity.

c) Choose the **Subscription** from the drop-down list and the managed identity in the **User assigned managed identities** field.

d) Click **Add** to assign.

**Step 7** Click **Next** to display the **Cisco Secure Firewall Threat Defense Virtual - NVA** page.

**Step 8** Configure the Virtual hub and the NVA details:

a) Select the Virtual WAN hub from the **vWAN Hub** drop-down list to deploy a Threat Defense Virtual instance.

b) Enter an appropriate name for the Threat Defense Virtual instance you are deploying.

c) Select the scale units that define the number of Threat Defense Virtual instances you want to deploy.

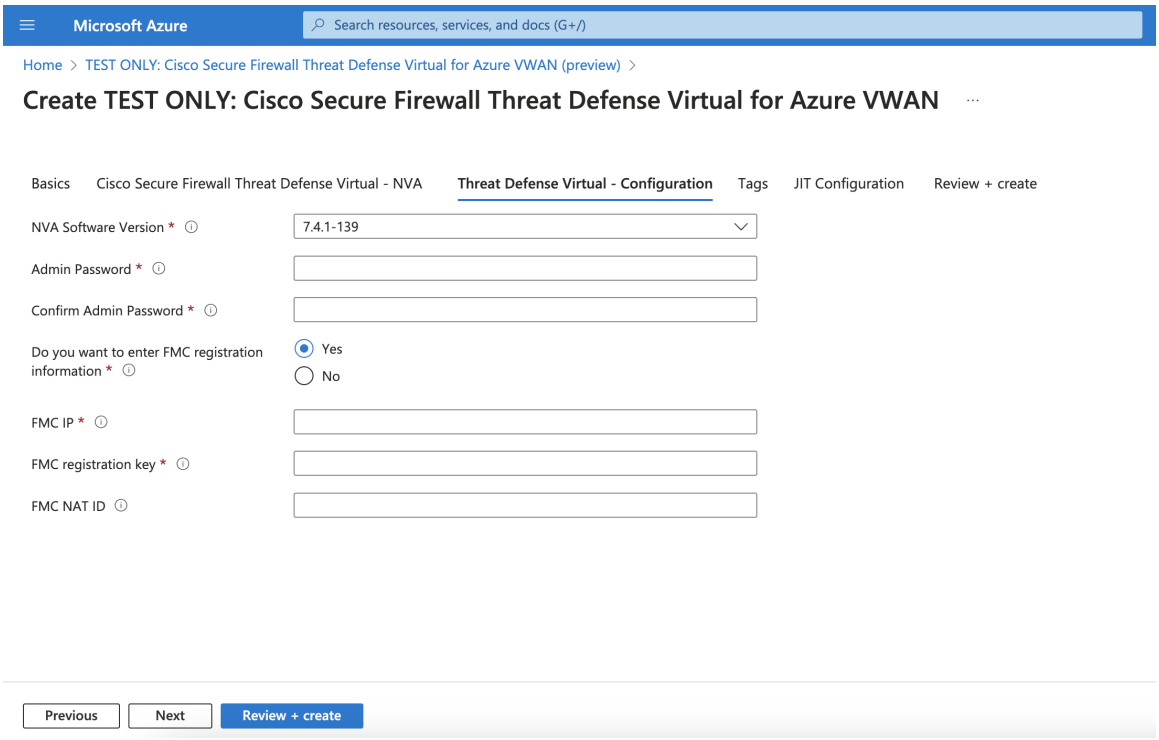
You can select the required scale unit to achieve the needed NVA throughput level. For example, selecting **4 Scale Units – 3.2 Gbps (2 x Standard_D3_v2_instances)** implies "**Number scale unit – Throughput level (2 Threat Defense Virtual with instance type**).

**Note**
Scale unit defines the number of Threat Defense Virtual instances and its associated instance type that you are deploying in the hub.

d) Enter the **Virtual Appliance ASN**.

**Note**
The ASN value that you enter must be within the range 64512 – 65534.

**Step 9** Click **Next** to display the **Threat Defense Virtual - Configuration** page.




**Step 10** Select the appropriate **NVA Software version** compatible version from the drop-down list.

**Note**
This field provides a list of NVA software versions compatible with the corresponding Threat Defense Virtual version you are deploying. Ensure to select the appropriate version from the list.

**Step 11** Create and confirm the admin password that is required to access the managed resource group containing Threat Defense Virtual instances. This is the password that is used for accessing Threat Defense Virtual instances.

**Step 12** Click **Yes** to enter the **FMC registration information**.

a) Enter the **FMC IP** address.

b) Enter the **FMC Registration Key** for registering the Threat Defense Virtual instances.

**Note**

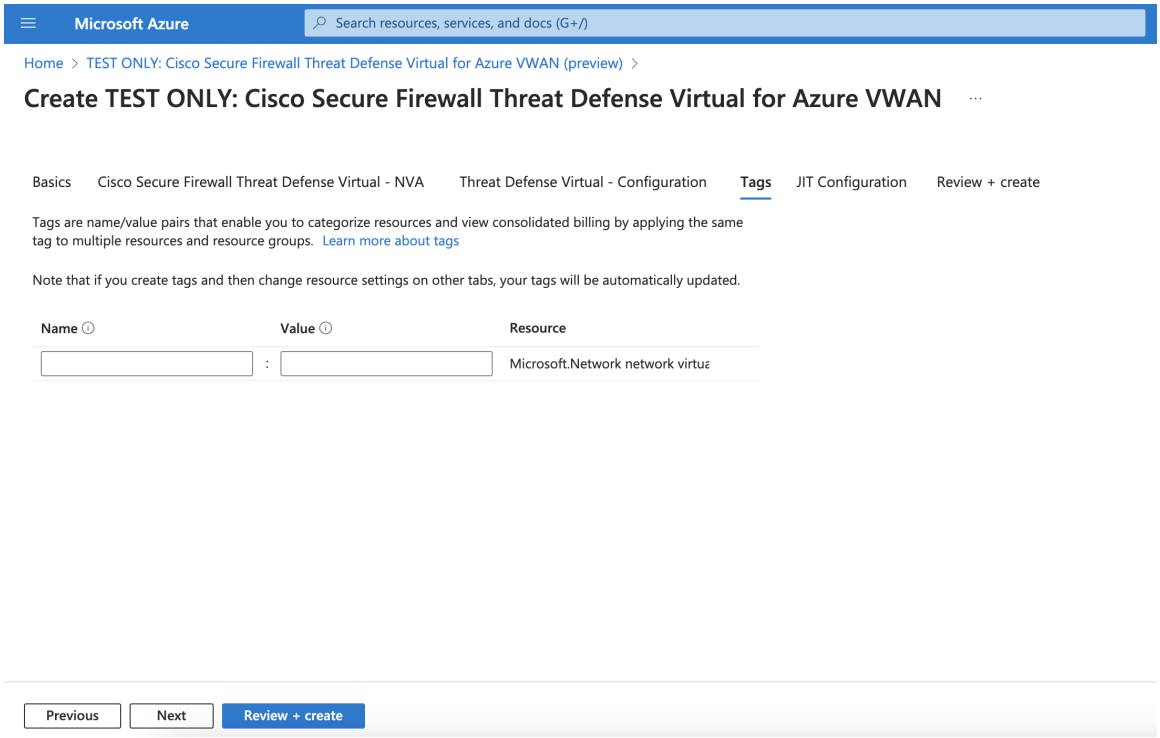
- The FMC Registration key must be an alphanumeric string of 1 – 37 characters in length. You will enter this key on the Management Center when adding Threat Defense Virtual.

c) [Optional] Enter the management center NAT ID that is used during instance registration.

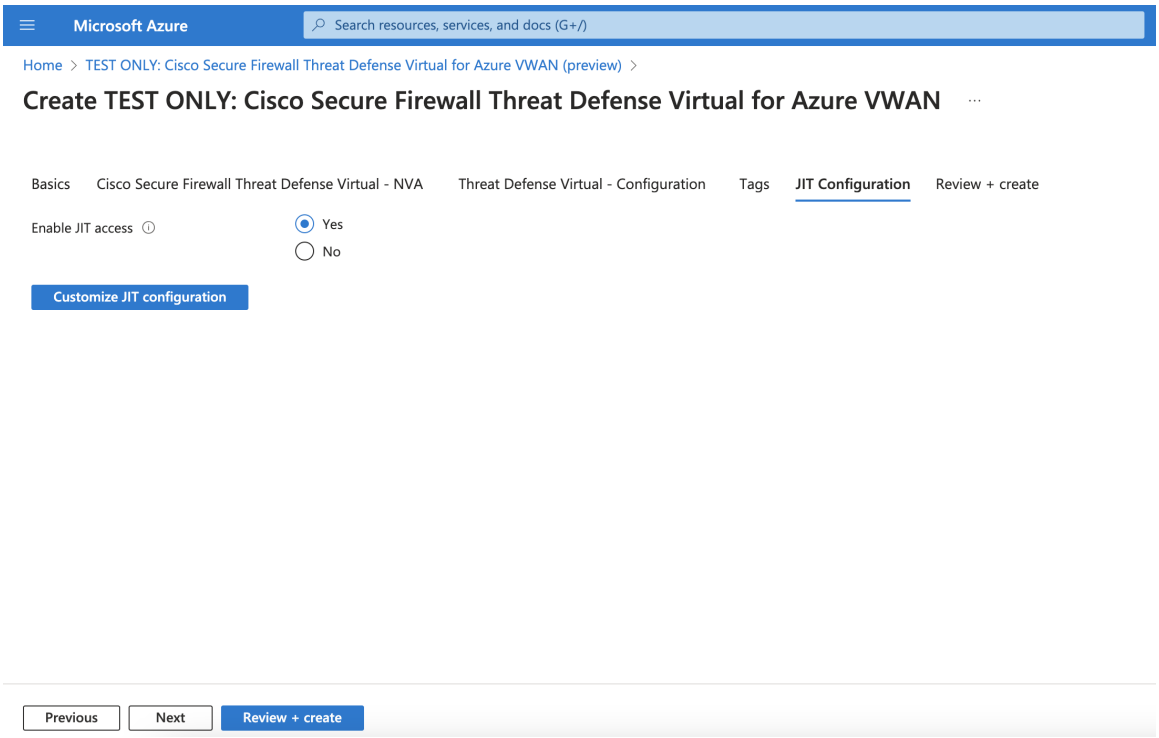
**Note**

- The NAT ID must be an alphanumeric string between 1 – 37 characters in length and is used only during the registration process between the Management Center and the device when one side does not specify an IP address. The NAT ID is essentially a one-time password, so it must be unique and not used by any other devices awaiting registration. To ensure successful registration, be sure to specify the same NAT ID on the FMC when adding the Threat Defense Virtual.

**Step 13** Click **Next** to configure the **Tags**.

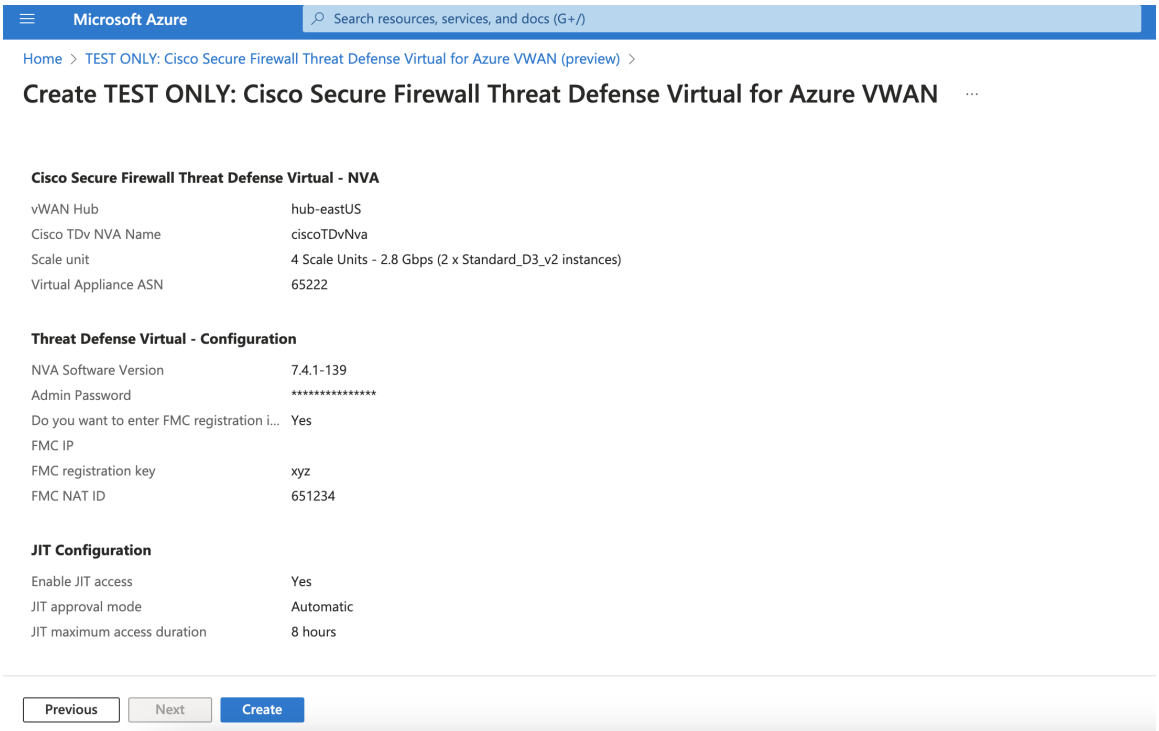



**Step 14** Click **Next** to display the **JIT configuration** page.



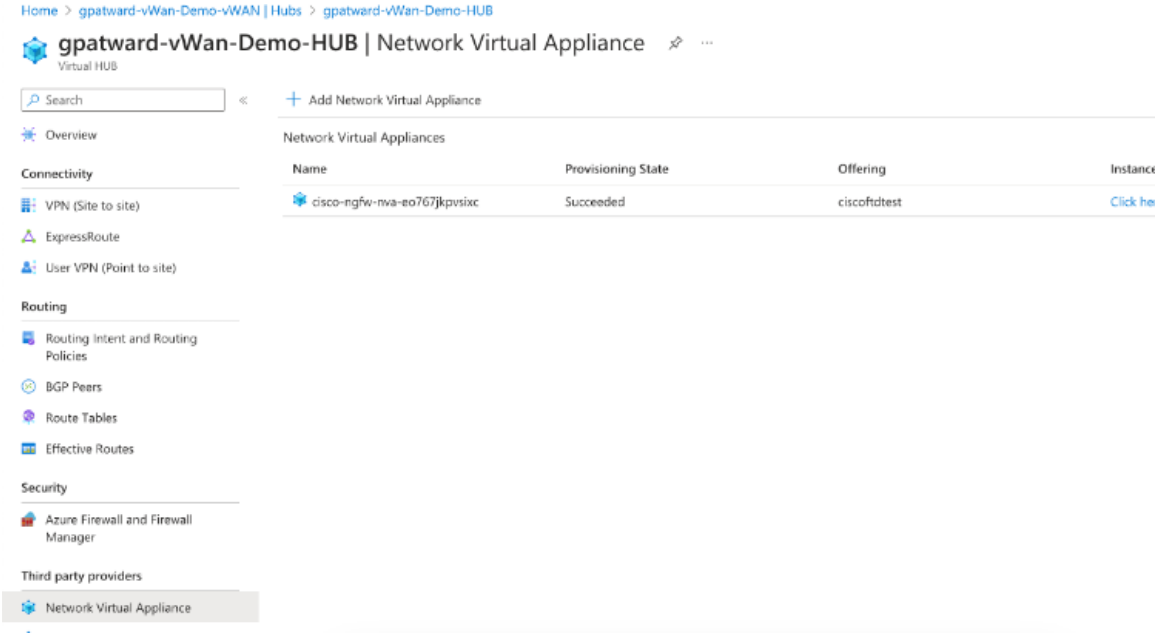

By default, the **Enable JIT access** option is set to **Yes**, which enables JIT for provisioning access to manage and troubleshoot the Threat Defense Virtual instances.

**Step 15** Click **Next** to display the **Review+Create** page.



**Step 16** Before deploying, you must review the subscription, NVA, Threat Defense Virtual and JIT configuration details, accept the Terms and conditions and then click **Create** to deploy the Threat Defense Virtual (NVA) on the Virtual WAN hub.

**Step 17** Go to **Home** > **Security** > **Third-party providers**, and click **Network Virtual Appliance** to view the NVA created on the hub.




**Step 18** Click the **NVA** to view all the Threat Defense Virtual instances deployed.

You can access the Threat Defense Virtual using the management public IP address of the instance and login using the SSH.

**Note**
The public IP addresses of each Threat Defense Virtual instance that you deploy on the hub is used for registering the instances in the management center.

### What to do next

Register and configure the Threat Defense Virtual instances that you deployed on the hub in the management center.

# Configure Threat Defense Virtual in Management Center

You configure each Threat Defense Virtual instance deployed on the hub through the management center.

Create all the objects needed for the Threat Defense Virtual configuration and management, including a device group, so you can easily deploy policies and install updates on multiple devices. All the configurations applied on the device group will be pushed to the Threat Defense Virtual instances.

This section provides a brief overview of the basic steps to configure the Threat Defense Virtual instances in the management center.

For more information, see Cisco Secure Firewall Management Center Device Configuration Guide.

# Register Threat Defense Virtual Instances in the Management Center

You must register all the Threat Defense Virtual instances that are deployed in the virtual WAN hub under a common Device group in the management center. It helps you to quickly deploy policies and configurations to those instances.

### Before you begin

- Require the management public IP address of each Threat Defense Virtual instance deployed in the Azure Virtual WAN hub. It is used to set up and register instances in the management center.
- Create a Device Group in the management center. See Add a Device Group.
- Create an Access Control Policy. See Creating a Basic Access Control Policy.
- FMC Registration Key created during Threat Defense Virtual deployment in the hub.

### Procedure

**Step 1** Log in to the Management Center.

**Step 2** Choose **Devices** > **Device Management**.

**Step 3** Click **Add** > **Device**

**Step 4** Enter the public IP address of the Threat Defense Virtual instance deployed in the hub.

**Step 5** Provide the display name for the Threat Defense Virtual instance.

**Step 6** Enter the **Registration Key** of the management center that you have created during the Threat Defense Virtual deployment in the hub.

**Step 7** From the **Group** drop-down list, choose the device group to which you want to add the Threat Defense Virtual instance.

**Step 8** From the **Access Control Policy** drop-down list, select the policy that you want to apply to the Threat Defense Virtual instance.

**Step 9** Enter other details as required.

**Step 10** Click **Register**.

**Step 11** Repeat Step 1 through Step 10 to register other Threat Defense Virtual instances.

### What to do next

Configure interfaces of Threat Defense Virtual instances.

# Configure Interfaces

After registering the Threat Defense Virtual instance, you must configure its interfaces in the management center.

The Azure Virtual WAN supports only **three** interfaces, which is configured as follows:

- Management interface with the public IP as the first interface.
- Outside interface with the public IP as the second interface.
- Inside interface with the private IP as the third interface (which has only private IP).

**Procedure**

**Step 1** Log in to the Management Center.
**Step 2** Go to the **Devices** page.
**Step 3** Click the **Edit** icon corresponding to the Threat Defense Virtual you have registered.
**Step 4** Click the **Edit** icon corresponding to an interface. For example, **GigbitEthernet0/0**.
**Step 5** Enter the name of the first interface as **`outside`**.
**Step 6** Check the **Enabled** check box to enable the interface.
**Step 7** From the **Security Zone** drop-down list, select **outside**.
**Step 8** Click the **IPv4** menu to assign the type of IP to the interface.
**Step 9** From the **IP Type** drop-down list, select **Use DHCP** to configure your interface to obtain an IP address from DHCP.
**Step 10** Check the **Obtain default route using DHCP** check box.
**Step 11** Enter the **Default route** metric as **`1`**.
**Step 12** Click **OK** to save the configuration.
**Step 13** Repeat Step 1 through Step 10 to configure the Inside interface.

**What to do next**

Configure routes for interfaces.

# Configure Route for Interfaces

Configure the static routes for Outside and Inside interfaces by creating network objects and assigning the gateway IP address.

- The Outside interface route configuration uses the gateway IP address as the default route for all the packets.
- The Inside interface route configuration uses the gateway IP address as the default route for the health probe packets and the packets that are destined for the hub network range.

The gateway IP address is computed using each interface's IP address and subnet mask address.

## Compute Gateway IP Address for Outside and Inside Interface

This section explains the process of computing the gateway IP address for the Outside and Inside interfaces with an example.

**Procedure**

**Step 1** Log in to the Management Center.

**Step 2** Go to the **Devices** > **Device Management**.

**Step 3** Access the Firewall Threat Defense Virtual instance you have deployed on the hub.

**Step 4** In the **>_Command** field, enter **`show interface GigbitEthernet 0/0`** to get the Outside interface configuration or **`show interface GigbitEthernet 0/1`** to get the Inside interface configuration details.

**Step 5** Repeat Step 1 through Step 4 to get the IP address and subnet mask addresses for the Inside interface or Outside interface.

**Step 6** Note the IP address and subnet mask addresses from the command result.

**Step 7** Compute the gateway IP addresses for Inside and Outside by following the example:

- To compute gateway IP address for Outside interface:

  **For Example**: For GigabitEthernet0/0 (Outside interface)

  IP address - **15.0.112.136**

  Subnet mask - **255.255.255.128**

  Hence, compute the gateway IP address as (that is the first IP address in this subnet) **15.0.112.129**.

- To compute gateway IP address for Inside interface:

  **For Example**: For GigabitEthernet0/1 (Inside interface)

  IP address - **15.0.112.10**

  Subnet mask - **255.255.255.128**

  Hence, compute the gateway IP as (that is the first IP address in this subnet) **15.0.112.1**.

### What to do next

Configure default route for Inside and Outside interfaces.

# Configure Default Route for Outside Interface

**Procedure**

**Step 1** Log in to the Management Center.

**Step 2** Go to the **Devices** > **Device Management**.

**Step 3** Click the Firewall Threat Defense Virtual instance.

**Step 4** Click **Routing** > **Static Route**.

**Step 5** Click **Add Route**.

**Step 6** From the **Interface** drop-down list, select **Outside**.

**Step 7** Select **any-ipv4** for the Outside interface under **Available Network** and click **Add**.

**Step 8** Enter the gateway IP address:

a) Click the + icon to add a network object.
b) Enter the name and description of the network object.
c) Click the **Host** network.
d) Enter the gateway IP address of the Outside interface that you have computed.
e) Click **Save**.

# Configure Default Route for Inside Interface

### Before you begin

You must have the CIDR IP address of the Firewall Threat Defense Virtual deployed on the hub. You require this to configure the Inside interface.

### Procedure

**Step 1** Log in to the Management Center.

**Step 2** Go to the **Devices** > **Device Management**.

**Step 3** Click the Firewall Threat Defense Virtual instance.

**Step 4** Click **Routing** > **Static Route**.

**Step 5** Click **Add Route**.

**Step 6** From the **Interface** drop-down list, select **Inside**.

**Step 7** Add the network object to configure the Inside interface with the CIDR IP address of the hub.

a) Click the + icon to add a network object.
b) Enter the name and description of the network object.
c) Click the **Host** network.
d) Enter the CIDR IP address (Private address space) of the hub.
e) Click **Save**.

**Step 8** Add the network object to configure the Inside interface with the load balancer health probe IP address.

a) Click the + icon to add a network object.
b) Enter the name and description of the network object.
c) Click the **Host** network.
d) Enter the IP address of the load balancer health probe. For example:`168.63.129.16`.

This IP address is a standard or fixed address.

**Step 9** Enter the gateway IP address:

a) Click the + icon to add a network object.
b) Enter the name and description of the object.
c) Click the **Host** network.
d) Enter the gateway IP address of the Inside interface that you have computed.

e) Click **Save**.

# Configure Traffic Routing

Note that you can configure either static routing or Border Gateway Protocol (BGP) for data exchange between the Threat Defense Virtual instances and the hub. These are two different routing methods that you can configure for network traffic in a Virtual WAN hub.

BGP is a dynamic routing protocol that factors the route based on the real-time traffic exchange between the hub and your threat defense virtual appliance. Whereas the static routing uses a preconfigured routing protocol to exchange traffic.

For more information about Azure Virtual WAN, refer to the Microsoft Azure Virtual WAN documentation.

## Configure Static Routing

### Procedure

**Step 1** Log in to the Management Center.

**Step 2** Go to the **Devices** > **Device Management**.

**Step 3** Click the Threat Defense Virtual instance.

**Step 4** Click **Routing** > **Static Route**.

**Step 5** Click **Add Route**.

**Step 6** From the **Interface** drop-down list, select **Outside**.

If you are configuring the Inside interface, select **Inside**.

**Step 7** Add the network object IP address:

a) Click the + icon to add a network object.
b) Enter the name and description of the object.
c) Click the **Host** network.
d) Enter the IP address.
e) Click **Save**.

## Enable BGP Routing

### Procedure

**Step 1** Log in to the Management Center.

**Step 2** Choose **Devices** > **Device Management**.

**Step 3** Click the Threat Defense Virtual instance.
**Step 4** Click the **Routing** menu.
**Step 5** Click **BGP** under **General Settings**.
**Step 6** Check the **Enable BGP** check box.
**Step 7** Enter the AS number of your virtual hub.
**Step 8** Click **Save**.

### What to do next

Configure BGP neighbors.

# Configure BGP Neighbors

### Procedure

**Step 1** Log in to the Management Center.
**Step 2** Choose **BGP** > **IPv4** > **Neighbor**.
**Step 3** Check the **Enable IPv4** check box.
**Step 4** Enter the Autonomous System (AS) number of your virtual hub.
**Step 5** Click **Add** in the Neighbor.
**Step 6** Enter the first IP address of the BGP endpoint that you have noted.
**Step 7** Check the **Enabled address** check box.
**Step 8** Enter the AS number in the **Remote AS** field.
**Step 9** Check the **Disable Connection Verification** check box on the **Advanced** menu.
**Step 10** Click **Save**.
**Step 11** Repeat Step 1 through Step 8 to add the second IP address of the BGP endpoint.

### What to do next

Verify BGP route configuration.

# Verify BGP Route Configuration

### Before you begin

After configuring the BGP endpoints, you must verify whether a connection through the BGP endpoints is established between Threat Defense Virtual and the virtual WAN hub.

### Procedure

**Step 1** Log in to the Management Center.

**Step 2** Choose **Devices** > **Device Management**.

**Step 3** Click the Threat Defense Virtual instance.

**Step 4** Click **CLI** in the **Device** > **General** widget.

**Step 5** In the **>_Command** field, enter **show route** to view and verify the connection status.

**Note**
Code **B** indicates the BGP endpoint connection status with the Threat Defense Virtual.

# Configure Health Probe

To ensure the Threat Defense Virtual status is stable, you must configure the Inside interface (Trusted) that connects to the Internal Load Balancer (ILB). The ILB performs periodic health check probes through the TCP port 443 to verify the response from Threat Defense Virtual.

### Procedure

**Step 1** Log in to the Management Center.

**Step 2** Choose **Devices** > **Platform Settings** > **New Policy** > **Threat Defense Settings**.

**Step 3** Add a **New policy** for the Threat Defense Virtual to connect to the load balancer.

**Step 4** Edit the new policy that you have added.

**Step 5** Check the **Enable HTTP Server** check box, and enter **`443`** in the **Port** field.

**Step 6** Click **+ Add** to configure the HTTP address.

**Step 7** Select the health probe IP address name.

**Step 8** Select the required IP address from the **Available Zone/Interface** and click **Add** to add it to **Selected Zones/Interfaces**.

**Step 9** Click **OK**.

**Step 10** Choose **Devices** > **Device Management**.

**Step 11** Click the edit icon in the **Applied Policies** widget.

**Step 12** Select this policy from the **Platform Settings** drop-down list.

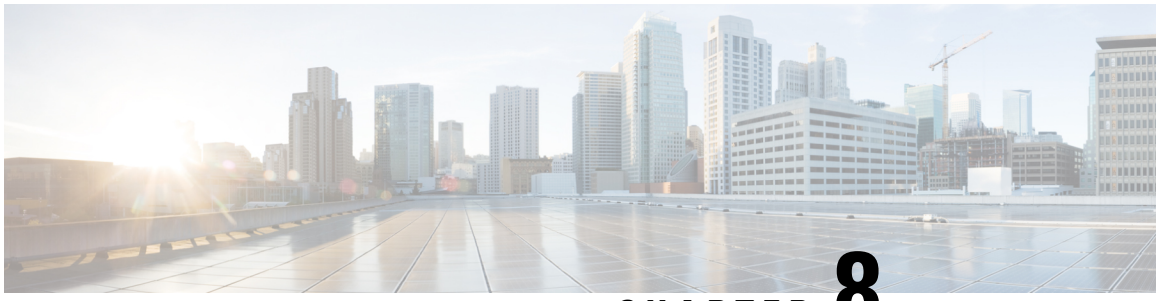
**Step 13** Update and apply the security policies as required.

For more information about configuring HTTP Access, see Configuring HTTP.

# Troubleshooting

The following are common error scenarios and debugging tips for the Threat Defense Virtual in Virtual WAN:

- Traffic is not routed to Threat Defense Virtual.
  - Verify the Threat Defense Virtual response to health probe checks in the management center.
  - Verify whether the derived gateway IP addresses of the Inside and Outside interfaces are correct.
  - Check the static route.
- Non-RFC RFC 1918 not reaching Threat Defense Virtual: Ensure Non-RFC 1918 ranges that are explicitly specified as Private addresses in the Routing Intent.
- Threat Defense deployment error: If you encounter the Error: **Hub Prefix Length should be less or equal to 23** during Threat Defense Virtual deployment, then ensure that the CIDR of the HUB address space is less than or equal to **/23**.



CHAPTER 8

# Deploy the Firewall Threat Defense Virtual on OCI

You can deploy the Firewall Threat Defense Virtual on the Oracle Cloud Infrastructure (OCI), a public cloud computing service that enables you to run your applications in a highly-available, hosted environment offered by Oracle.

The following procedures describe how to prepare your OCI environment and launch the Firewall Threat Defense Virtual instance. You log into the OCI portal, search the OCI Marketplace for the Cisco Firepower NGFW virtual firewall (NGFWv) offering, and launch the compute instance. After launching the Firewall Threat Defense Virtual, you must configure route tables to direct traffic to the firewall depending on the traffic's source and destination.


- Overview, on page 258
- End-to-End Procedure, on page 259
- Prerequisites, on page 261
- Guidelines and Limitations, on page 261
- Sample Network Topology, on page 263
- How to Manage Secure Firewall Threat Defense Virtual Device, on page 264
- Configure OCI Environment, on page 265
- Deploy the Threat Defense Virtual on OCI, on page 269
- Attach the Interfaces, on page 270
- Add Route Rules for the Attached VNICs, on page 271
- Deploy Auto Scale Solution, on page 272
- Prerequisites, on page 273
- Encrypt Password, on page 280
- Preparation of Firewall Threat Defense Virtual configuration File, on page 282
- Deploy the Auto Scale Solution, on page 287
- Validate Deployment, on page 292
- Upgrade, on page 293
- Load Balancer Backend Sets, on page 294
- Delete Autoscale Configuration from OCI, on page 294
- Connect to the Firewall Threat Defense Virtual Instance Using SSH, on page 297
- Connect to the Firewall Threat Defense Virtual Instance Using OpenSSH, on page 297
- Connect to the Firewall Threat Defense Virtual Instance Using PuTTY, on page 298
- IPv6 Troubleshooting, on page 299

# Overview

The Cisco Secure Firewall Threat Defense Virtual runs the same software as physical Cisco Firewall Threat Defense to deliver proven security functionality in a virtual form factor. The Firewall Threat Defense Virtual can be deployed in the public OCI. It can then be configured to protect virtual and physical data center workloads that expand, contract, or shift their location over time.

## OCI Compute Shapes

A shape is a template that determines the number of CPUs, amount of memory, and other resources that are allocated to an instance. The Firewall Threat Defense Virtual support the following OCI shape types:

***Table 24: Supported Compute Shapes for Firewall Threat Defense Virtual***

| OCI Shape | Supported Threat Defense Virtual version | Attributes | | Interfaces |
|---|---|---|---|---|
| | | oCPUs | RAM (GB) | |
| Intel VM.DenseIO2.8 | 7.3.x and later | 8 | 120 | Minimum 4, Maximum 8 |
| Intel VM.StandardB1.4 | 7.3.x and later | 4 | 48 | Minimum 4, Maximum 4 |
| Intel VM.StandardB1.8 | 7.3.x and later | 4 | 96 | Minimum 4, Maximum 8 |
| Intel VM.Standard1.4 | 7.3.x and later | 4 | 28 | Minimum 4, Maximum 4 |
| Intel VM.Standard1.8 | 7.3.x and later | 8 | 56 | Minimum 4, Maximum 8 |
| Intel VM.Standard2.4 | 7.1, 7.2.x, and 7.3.x | 4 | 60 | Minimum 4, Maximum 4 |
| Intel VM.Standard2.8 | 7.1, 7.2.x, and 7.3.x | 8 | 120 | Minimum 4, Maximum 8 |
| Intel VM.Standard3.Flex* | 7.3.x and later | 4 | 16 | Minimum 4, Maximum 4 |
| | 7.3.x and later | 6 | 24 | Minimum 4, Maximum 6 |
| | 7.3.x and later | 8 | 32 | Minimum 4, Maximum 8 |

| OCI Shape | Supported Threat Defense Virtual version | Attributes | | Interfaces |
|---|---|---|---|---|
| | | oCPUs | RAM (GB) | |
| Intel VM.Optimized3.Flex* | 7.3.x and later | 4 | 16 | Minimum 4, Maximum 8 |
| | 7.3.x and later | 6 | 24 | Minimum 4, Maximum 10 |
| | 7.3.x and later | 8 | 32 | Minimum 4, Maximum 10 |
| AMD VM.Standard.E4.Flex* | 7.3.x and later | 4 | 16 | Minimum 4, Maximum 4 |
| | 7.3.x and later | 6 | 24 | Minimum 4, Maximum 6 |
| | 7.3.x and later | 8 | 32 | Minimum 4, Maximum 8 |

- *SR-IOV mode is supported with Flex shapes from Version 7.4.x and later.
- In OCI, 1 oCPU is equal to 2 vCPU.
- The Firewall Threat Defense Virtual requires a minimum of 4 interfaces.

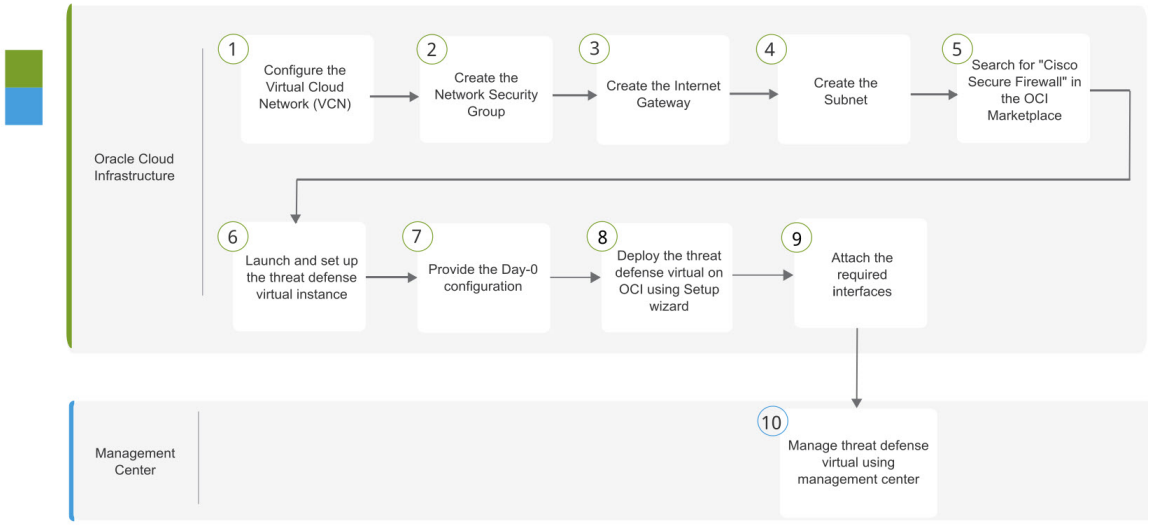
Recommendations for using the OCI Compute shapes supported by version Firewall Threat Defense Virtual 7.3 and later.

- OCI marketplace image version **7.3.0-69-v3** and later are compatible only with the OCI compute shapes of Firewall Threat Defense Virtual 7.3 and later.
- You can use the OCI compute shapes supported by Firewall Threat Defense Virtual 7.3 and later only for new deployments.
- OCI compute shapes version **7.3.0-69-v3** and later are not compatible with upgrading VMs that are deployed with Firewall Threat Defense Virtual using the OCI compute shape versions earlier to Firewall Threat Defense Virtual 7.3.
- The billing will continue for the **VM.DenseIO2.8** compute shape subscription, even after you shut down the instance. For more information, see OCI Documentation.

You create an account on OCI, launch a compute instance using the Cisco Firepower NGFW virtual firewall (NGFWv) offering on the Oracle Cloud Marketplace, and choose an OCI shape.

# End-to-End Procedure

The following flowchart illustrates the workflow for deploying Threat Defense Virtual on Oracle Cloud Infrastructure.




| | Workspace | Steps |
|---|---|---|
| 1 | Oracle Cloud Infrastructure | Deploy the Threat Defense virtual on OCI: Configure the Virtual Cloud Network (VCN) (**Networking** > **Virtual Cloud Networks** > **CIDR block** > **Create VCN**. |
| 2 | Oracle Cloud Infrastructure | Deploy the Threat Defense virtual on OCI: Create the Network Security Group. (**Networking** > **Virtual Cloud Networks** > **Virtual Cloud Network Details** > **Network Security Groups** > **Create Network Security Group**. |
| 3 | Oracle Cloud Infrastructure | Deploy the Threat Defense Virtual on OCI: Create the Internet Gateway. **Networking** > **Virtual Cloud Networks** > **Virtual Cloud Network Details** > **Internet Gateways** > **Create Internet Gateway**. |
| 4 | Oracle Cloud Infrastructure | Deploy the Threat Defense Virtual on OCI: Create the Subnet. **Networking** > **Virtual Cloud Networks** > **Virtual Cloud Network Details** > **Subnets** > **Create Subnet**. |
| 5 | Oracle Cloud Infrastructure | Deploy the Threat Defense Virtual on OCI, on page 269: Search for “Cisco Secure Firewall” in the OCI Marketplace. |
| 6 | Oracle Cloud Infrastructure | Deploy the Threat Defense Virtual on OCI, on page 269: Launch and set up the threat defense virtual instance. |
| 7 | Oracle Cloud Infrastructure | Deploy the Threat Defense Virtual on OCI, on page 269: Provide the Day-0 configuration. |
| 8 | Oracle Cloud Infrastructure | Deploy the Threat Defense Virtual on OCI, on page 269: Deploy the threat defense virtual on OCI using setup wizard. |
| 9 | Oracle Cloud Infrastructure | Deploy the Threat Defense Virtual on OCI: Attach the Interfaces.**Compute** > **Instances** > **Instance Details** > **Attached VNICs**. |
| 10 | Management Center | Manage the Firewall Threat Defense Virtual by using the Management Center |

# Prerequisites

- Create an OCI account at https://www.oracle.com/cloud/.
- A Cisco Smart Account. You can create one at Cisco Software Central (https://software.cisco.com/).
- License the Firewall Threat Defense Virtual.
  - Configure all license entitlements for the security services from the Firewall Management Center.
  - See "Licensing" in the *Cisco Secure Firewall Management Center Admin Guide* for more information about how to manage licenses.



**Note** All the default License entitlement offered by Cisco, previously for Firewall Threat Defense Virtual will have the IPv6 configuration support.

- Interface requirements:
  - Management interfaces (2) — One used to connect the Firewall Threat Defense Virtual to the Firewall Management Center, second used for diagnostics; cannot be used for through traffic.
  - Traffic interfaces (2) — Used to connect the Firewall Threat Defense Virtual to inside hosts and to the public network.
- Communications paths:
  - Public IPs for access into the Firewall Threat Defense Virtual.
- For Firewall Threat Defense Virtual system requirements, see Cisco Secure Firewall Threat Defense Compatibility Guide.

# Guidelines and Limitations

### Supported Features

- Deployment in the OCI Virtual Cloud Network (VCN)
- Routed mode (default)
- Licensing – Only BYOL is supported
- IPv6
- Firewall Management Center support only.
- Single Root I/O Virtualization (SR-IOV) is supported.

### Performance Tiers for FTDv Smart Licensing

The Firewall Threat Defense Virtual supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.

*Table 25: Firewall Threat Defense Virtual Licensed Feature Limits Based on Entitlement*

| Performance Tier | Device Specifications (Core/RAM) | Rate Limit | RA VPN Session Limit |
|---|---|---|---|
| FTDv5, 100Mbps | 4 core/8 GB | 100Mbps | 50 |
| FTDv10, 1Gbps | 4 core/8 GB | 1Gbps | 250 |
| FTDv20, 3Gbps | 4 core/8 GB | 3Gbps | 250 |
| FTDv30, 5Gbps | 8 core/16 GB | 5Gbps | 250 |
| FTDv50, 10Gbps | 12 core/24 GB | 10Gbps | 750 |
| FTDv100, 16Gbps | 16 core/32 GB | 16Gbps | 10,000 |

See the "Licensing" chapter in the *Cisco Secure Firewall Management Center Admin Guide* for guidelines when licensing your Firewall Threat Defense Virtual device.

**Note** To change the vCPU/memory values, you must first power off the Firewall Threat Defense Virtual device.

### Performance Optimizations

To achieve the best performance out of the Firewall Threat Defense Virtual, you can make adjustments to the both the VM and the host. See Virtualization Tuning and Optimization on OCI for more information.

**Receive Side Scaling**—The Firewall Threat Defense Virtual supports Receive Side Scaling (RSS), which is a technology utilized by network adapters to distribute network receive traffic to multiple processor cores. Supported on Version 7.0 and later. See Multiple RX Queues for Receive Side Scaling (RSS) for more information.

### Snort

- If you are observing abnormal behavior such as Snort taking a long time to shut down, or the VM being slow in general or when a certain process is executed, collect logs from the Firewall Threat Defense Virtual and the VM host. Collection of overall CPU usage, memory, I/O usage, and read/write speed logs will help troubleshoot the issues.
- High CPU and I/O usage is observed when Snort is shutting down. If a number of Firewall Threat Defense Virtual instances have been created on a single host with insufficient memory and no dedicated CPU, Snort will take a long time to shut down which will result in the creation of Snort cores.

### Unsupported Features

- Local management support via Firewall Device Manager.

- Firewall Threat Defense Virtual native HA
- Transparent/inline/passive modes
- Data Interface configuration via DHCP

### Limitations

- Firewall Threat Defense Virtual deployment on OCI does not support Mellanox 5 as vNICs in the SR-IOV mode.
- IPv6 works in only Dualstack with (VCN IPv4 and IPv6) configuration as per OCI standards.
- Separate routing rules required for Firewall Threat Defense Virtual for both static and DHCP configuration.

# Sample Network Topology

The following figure shows the recommended topology for the Firewall Threat Defense Virtual in Routed Firewall Mode with 4 subnets configured in OCI for the Firewall Threat Defense Virtual (management, diagnostic, inside, and outside).

***Figure 37: Sample Firewall Threat Defense Virtual on OCI Deployment with subnets in four VCNs***

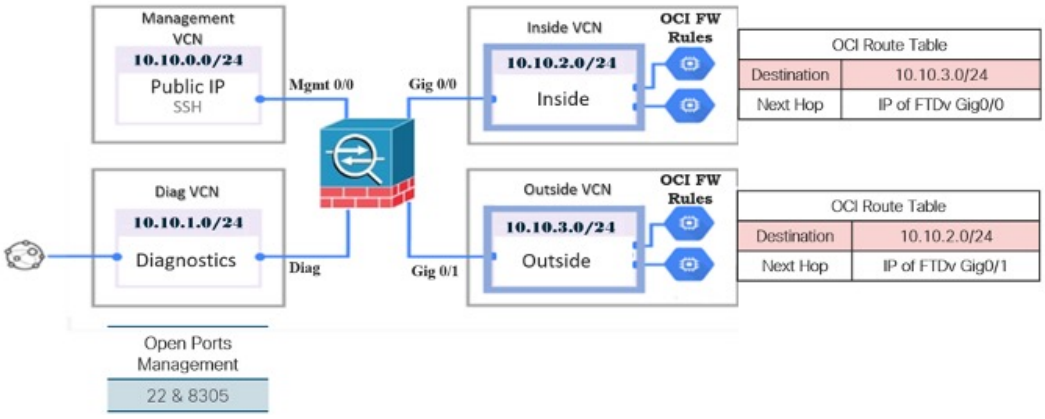
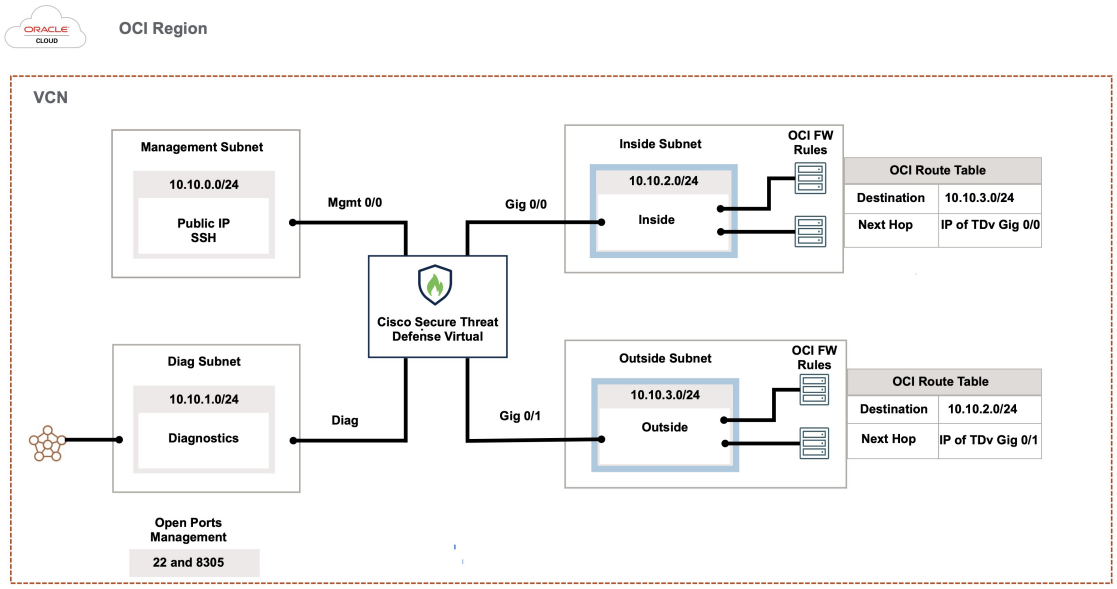
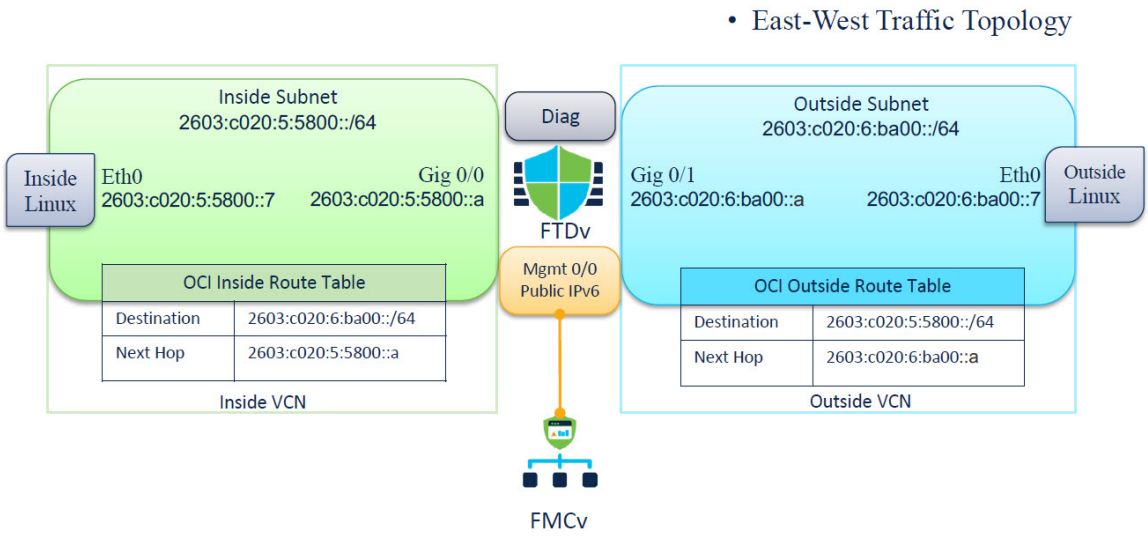


*Figure 38: Sample Firewall Threat Defense Virtual on OCI Deployment with four subnets in a VCN*




**Note** When you are using four subnets in a single VCN, the routes specific to a subnet must be added to the route table associated with the subnet.

### Firewall Threat Defense Virtual IPv6 Deployment Topology




# How to Manage Secure Firewall Threat Defense Virtual Device

You have two options to manage your Secure Firewall Threat Defense Virtual.

## Secure Firewall Management Center

If you are managing large numbers of devices, or if you want to use the more complex features and configurations that the Firewall Threat Defense allows, use the Firewall Management Center to configure your devices instead of the integrated Firewall Device Manager.

**Important** You cannot use both the Firewall Device Manager and the Firewall Management Center to manage the Firewall Threat Defense device. Once the Firewall Device Manager integrated management is enabled, it won't be possible to use the Firewall Management Center to manage the Firewall Threat Defense device, unless you disable the local management and re-configure the management to use the Firewall Management Center. On the other hand, when you register the Firewall Threat Defense device to the Firewall Management Center, the Firewall Device Manager onboard management service is disabled.

**Caution** Currently, Cisco does not have an option to migrate your Firewall Device Manager configuration to the Firewall Management Center and vice-versa. Take this into consideration when you choose what type of management you configure for the Firewall Threat Defense device.

## Secure Firewall Device Manager

The Firewall Device Manager is a web interface included on most Firewall Threat Defense devices. It lets you configure the basic features of the software that are most commonly used for small networks. It is especially designed for networks that include a single device or just a few, where you do not want to use a high-powered multiple-device manager to control a large network with many devices.

**Note** See the Cisco Secure Firewall Threat Defense Compatibility Guide for list of devices that support the Firewall Device Manager.

# Configure OCI Environment

You can configure the Virtual Cloud Network (VCN) for your threat defense virtual deployment as follows:

- **Multiple VCNs** - At a minimum, you need four VCNs, one for each interface of the Firewall Threat Defense Virtual. This allows for isolated traffic inspection between different networks.
- **Single VCN with Subnets** - Alternatively, you can configure a single VCN with four subnets, one for each interface of the threat defense virtual. In this configuration, traffic between subnets within the same VCN can be inspected and controlled by the firewall using associated route tables. This allows you to manage inter-subnet traffic effectively while using a single VCN.

You can continue with the following procedures to complete the Management VCN. Then you return to **Networking** to create VCNs for the diagnostic, inside, and outside interfaces.

### Procedure

**Step 1** Log into OCI and choose your region.

OCI is divided into multiple regions that are isolated from each other. The region is displayed in the upper right corner of your screen. Resources in one region do not appear in another region. Check periodically to make sure you are in the intended region.

**Step 2** Choose **Networking** > **Virtual Cloud Networks** and click **Create VCN**.

**Step 3** Enter a descriptive **Name** for your VCN, for example, *FTDv-Management*.

**Step 4** Enter a **CIDR block** for your VCN.

a) An IPv4 CIDR block of IP addresses. CIDR (Classless Inter-Domain Routing) notation is a compact representation of an IP address and its associated routing prefix. For example, 10.0.0.0/24.

**Note**
Use DNS hostnames in this VCN.

b) An IPv6 CIDR block of IP addresses. CIDR (Classless Inter-Domain Routing) notation is a compact representation of an IP address and its associated routing prefix. For example, [::/0].

c) Select the IPv6 CIDR block as Oracle-assigned IPv6/56 prefix to your Virtual Cloud Network.

**Step 5** Click **Add IPv6 CIDR Block** to add a new IPv6 block.

**Step 6** Add the IPv6 prefix for your VCN, for example, */54*.

**Step 7** Click **Create VCN**.

#### What to do next

Continue with the following procedures to complete the Management VCN. When you complete the management VCN you'll create VCNs for the diagnostic, inside, and outside interfaces.

**Note** After you select a service from the navigation menu, the menu on the left includes the compartments list. Compartments help you organize resources to make it easier to control access to them. Your root compartment is created for you by Oracle when your tenancy is provisioned. An administrator can create more compartments in the root compartment and then add the access rules to control which users can see and take action in them. See the Oracle document Managing Compartments for more information.

# Create the Network Security Group

A network security group consists of a set of vNICs and a set of security rules that apply to those vNICs.

**Procedure**

**Step 1** Choose **Networking** > **Virtual Cloud Networks** > **Virtual Cloud Network Details** > **Network Security Groups**, and click **Create Network Security Group**.

**Step 2** Enter a descriptive **Name** for your Network Security Group, for example, *FTDv-Mgmt-Allow-22-8305*.

**Step 3** Click **Next**.

**Step 4** Add your security rules:

a) Add a rule to allow TCP port 22 for SSH access.

b) Add a rule to allow TCP port 8305 for HTTPS access.

The Firewall Threat Defense Virtual can be managed via the Firewall Management Center, which requires port 8305 to be opened for HTTPS connections.

**Note**
You apply these security rules to the management interface/VCN.

**Step 5** Click **Create**.

# Create the Internet Gateway

An Internet gateway is required to make your management subnet publicly accessible.

**Procedure**

**Step 1** Choose **Networking** > **Virtual Cloud Networks** > **Virtual Cloud Network Details** > **Internet Gateways**, and click **Create Internet Gateway**.

**Step 2** Enter a descriptive **Name** for your Internet gateway, for example, *FTDv-IG*.

**Step 3** Click **Create Internet Gateway**.

**Step 4** Add the route to the Internet Gateway:

a) Choose **Networking** > **Virtual Cloud Networks** > **Virtual Cloud Network Details** > **Route Tables**.

b) Click on the link for your default route table to add route rules.

c) Click **Add Route Rules**.

d) From the **Target Type** drop-down, select **Internet Gateway**.

e) Enter the Destination IPv4 CIDR Block, for example 0.0.0.0/0.

f) Enter the Destination IPv6 CIDR Block, for example [::/0].

g) From the **Target Internet Gateway** drop-down, select the gateway you created.

h) Click **Add Route Rules**.

# Create the Subnet

Each VCN will have one subnet, at a minimum. You'll create a Management subnet for the Management VCN. You'll also need a Diagnostic subnet for the Diagnostic VCN, need an Inside subnet for the Inside VCN, and an Outside subnet for the Outside VCN.

If you are using one VCN, then you will create Management subnet, Diagnostic subnet, Inside subnet, and Outside subnet within the VCN.

### Procedure

**Step 1** Choose **Networking** > **Virtual Cloud Networks** > **Virtual Cloud Network Details** > **Subnets**, and click **Create Subnet**.

**Step 2** Enter a descriptive **Name** for your subnet, for example *Management*.

**Step 3** Select a **Subnet Type** (leave the recommended default of **Regional**).

**Step 4** Enter a **CIDR Block**, for example, 10.10.0.0/24. The internal (non-public) IP address for the subnet is taken from this CIDR block.

a) If you are enabling IPv6, then select the **ENABLE IPv6 CIDR BLOCK** check box.
b) In the **IPv6 CIDR Block**, enter the IPV6 prefix range.

**Step 5** Select one of the route tables you created previously from the **Route Table** drop-down.

**Step 6** Select the **Subnet Access** for your subnet.

For the Management subnet, this must be **Public Subnet**.

**Step 7** Select the **DHCP Option**.

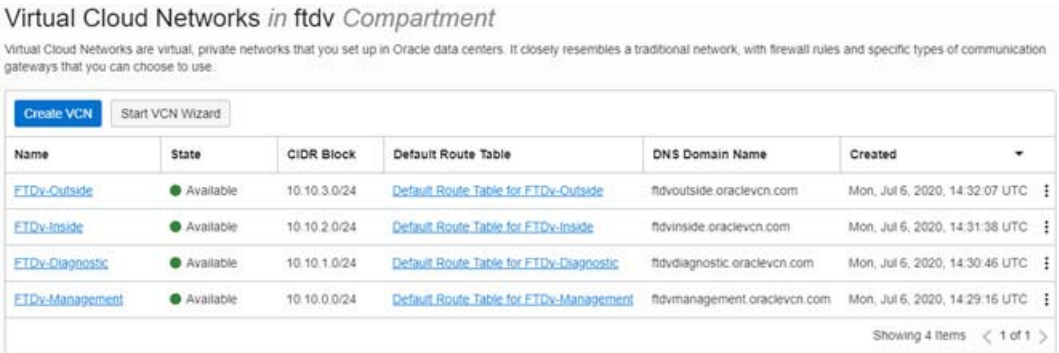
**Step 8** Select a **Security List** that you created previously.

**Step 9** Click **Create Subnet**.

### What to do next

After you configure your VCNs (Management, Diagnostic, Inside, Outside), you are ready to launch the Firewall Threat Defense Virtual. See the following figure for an example of the Firewall Threat Defense Virtual VCN configuration.

***Figure 39: Firewall Threat Defense Virtual Virtual Cloud Networks***



## Configure IPv6 Gateway Address Using Cloud Shell

In OCI, each subnet has a unique IPv6 gateway address which you must configure in Firewall Threat Defense Virtual for IPv6 traffic to work. This gateway address is retrieved from the subnet details running an OCI command in the cloud shell.

### Procedure

**Step 1** Go to **OCI** > **Open CloudShell (OCI Cloud Terminal)** .

**Step 2** Execute following command to get the IPv6 details from the subnet:

```
oci network subnet get -subnet_id <subnet_OCID>
```

**Step 3** From the command result find the `ipv6-virtual-router-ip` key.

**Step 4** Copy the value of this key and use it as required.

# Deploy the Threat Defense Virtual on OCI

Deploy the Firewall Threat Defense Virtual on OCI via a Compute instance using the Cisco Firepower NGFW virtual firewall (NGFWv) offering on the Oracle Cloud Marketplace. Select the most appropriate machine shape based on characteristics such as the number of CPUs, amount of memory, and network resources.

### Procedure

**Step 1** Log into the OCI portal.

The region is displayed in the upper right corner of your screen. Make sure you are in the intended region.

**Step 2** Choose **Marketplace** > **Applications**.

**Step 3** Search Marketplace for “Cisco Firepower NGFW virtual firewall (NGFWv)” and choose the offering.

**Step 4** Review the Terms and Conditions, and check the **I have reviewed and accept the Oracle Terms of Use and the Partner terms and conditions.**check box.

**Step 5** Click **Launch Instance**.

**Step 6** Enter a descriptive **Name** for your instance, for example *FTDv-6-7*.

**Step 7** Click **Change Shape** and select the shape with the number of CPUs, amount of RAM, and number of interfaces required for the Firewall Threat Defense Virtual, for example VM.Standard2.4 (see Overview, on page 258).

**Step 8** From the **Virtual Cloud Network** drop-down, choose the Management VCN.

**Step 9** From the **Subnet** drop-down, choose the Management subnet if it's not autopopulated.

**Step 10** Check **Use Network Security Groups to Control Traffic** and choose the security group you configured for the Management VCN.

**Step 11** Click the **Assign a Public Ip Address** radio button.

**Step 12** Under **Add SSH keys**, click the **Paste Public Keys** radio button and paste the SSH key.

Linux-based instances use an SSH key pair instead of a password to authenticate remote users. A key pair consists of a private key and public key. You keep the private key on your computer and provide the public key when you create an instance. See Managing Key Pairs on Linux Instances for guidelines.

**Step 13** Click the **Show Advanced Options** link to expand the options.

**Step 14** Under **Initialization Script**, click the **Paste Cloud-Init Script** radio button to provide the day0 configuration for your Firewall Threat Defense Virtual. The day0 configuration is applied during the firstboot of the Firewall Threat Defense Virtual.

The following example shows a sample day0 configuration you can copy and paste in the **Cloud-Init Script** field:

```
{
"Hostname": "ftdv-oci",
"AdminPassword": "myPassword@123456",
"FirewallMode": "routed",
"IPv4Mode": "dhcp",
"IPv6Mode": "dhcp",
"ManageLocally":"No",
"FmcIp": "1.2.3.4",
"FmcRegKey": "cisco123reg",
"FmcNatId": "cisco123nat"
}
```

- **FmcRegKey —** This is a one-time-use registration key used to register registering the device to a Firewall Management Center. The registration key is any user-defined alphanumeric value up to 37 characters in length.
- **FmcNatId —** This is a unique one-time-use string (user-defined). If the device and the Firewall Management Center are separated by a NAT device, you must enter a unique NAT ID, along with the unique registration key.

**Step 15** Click **Create**.

### What to do next

Monitor the Firewall Threat Defense Virtual instance, which shows the state as Provisioning after you click the **Create** button.

**Important** It's important to monitor the status. As soon as the Firewall Threat Defense Virtual instance goes from Provisioning to Running state, you need to attach the VNICs as required before the Firewall Threat Defense Virtual boot completes.

# Attach the Interfaces

The Firewall Threat Defense Virtual enters the Running state with one VNIC attached (see **Compute** > **Instances** > **Instance Details** > **Attached VNICs**). This is referred to as the Primary VNIC, and maps to the Management VCN. Before the Firewall Threat Defense Virtual completes the first boot, you need to attach the VNICs for the other VCN subnets you created previously (diagnostic, inside, outside) so that the VNICs are correctly detected on the Firewall Threat Defense Virtual.

**Procedure**

**Step 1** Select your newly launched Firewall Threat Defense Virtual instance.

**Step 2** Choose **Attached VNICs** > **Create VNIC**.

**Step 3** Enter a descriptive **Name** for your VNIC, for example, *Inside*.

**Step 4** Select the VCN from the **Virtual Cloud Network** drop-down.

**Step 5** Select your subnet from the **Subnet** drop-down.

**Step 6** Check **Use Network Security Groups to Control Traffic** and choose the security group you configured for the selected VCN.

**Step 7** Check **Skip Source Destination Check** Network Security Groups to Control Traffic.

**Step 8** (Optional) Specify a **Private IP Address**. This is only required if you want to choose a particular IP for the VNIC.

If you do not specify an IP, OCI will assign an IP address from the CIDR block you assigned to the subnet.

**Step 9** Click **Save Changes** to create the VNIC.

**Step 10** Repeat this procedure for each VNIC your deployment requires.

# Add Route Rules for the Attached VNICs

Add route table rules to the diagnostic, inside, and outside route tables.

**Procedure**

**Step 1** Choose **Networking** > **Virtual Cloud Networks** and click the default route table associated with the VCN (inside or outside).

**Step 2** Click **Add Route Rules**.

**Step 3** From the **Target Type** drop-down, select **Private IP**.

**Step 4** From the **Destination Type** drop-down, select **CIDR Block**.

**Step 5** Enter the **Destination CIDR Block**, for example 0.0.0.0/0.

**Step 6** Enter the private IP address of the VNIC in the **Target Selection** field.

If you did not explicitly assign an IP address to the VNIC, you can find the auto-assigned IP address from the VNIC details (**Compute** > **Instances** > **Instance Details** > **Attached VNICs**).

**Step 7** Click **Add Route Rules**.

If you want to configure IPv6 internet access through internet gateway, then perform the following:

a) From the **Target Type** drop-down, select **Internet Gateway**.
b) In the **Destination CIDR Block**, specify the IP address
c) From the **Target Internet Gateway** drop-down, select an existing internet gateway compartment or create new one.

**Step 8** Repeat this procedure for each VNIC your deployment requires.

**Note**
If the IPv6 address configured with the Routing rule through DHCP or IPv6 address prefix is /128, then you must add the following routes in Firewall Threat Defense Virtual route table.

ipv6 route <interface_name> <interface_subnet_CIDR> <ipv6_virtual_router_ip>

Example:

- `ipv6 route inside 2603:c020:5:5800::/64 fe80::200:17ff:fe96:921b`
- `ipv6 route outside 2603:c020:6:ba00::/64 fe80::200:17ff:fe21:748c`

# Deploy Auto Scale Solution

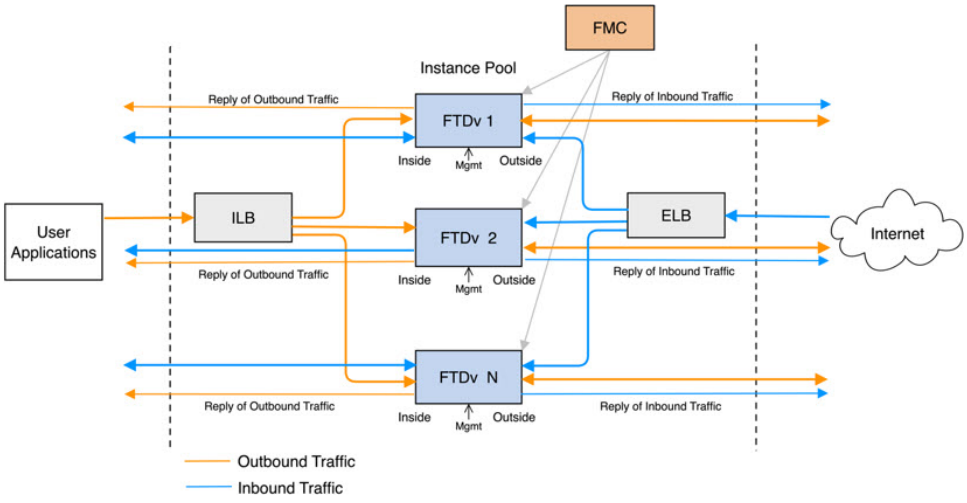
The following sections describe how the components of the Auto Scale solution work for the Firewall Threat Defense Virtual on OCI.

## Auto Scale Use Case

The use case for the Firewall Threat Defense Virtual Auto Scale on OCI solution is shown in the following figure. Internet-facing Load Balancer will have a public IP address with ports enabled using Listener and Target Group combination.

Port-based bifurcation is possible for traffic, and it can be achieved via NAT rules. This is explained in the following sections.

***Figure 40: Secure Firewall Threat Defense Virtual Auto Scale Use Case Diagram***




## Scope

This document covers the detailed procedures to deploy the Firewall Threat Defense Virtual Auto Scale for OCI solution.

**Important**

- Read the entire document before you begin your deployment.
- Make sure the prerequisites are met before you start deployment.
- Make sure you follow the steps and order of execution as described herein.

# Prerequisites

### Permission and Policies

Following are the OCI permissions and policies that you require to implement the solution:

1. **Users and Group**

**Note** You must be an OCI User or a Tenancy Administrator to create the Users and Groups.

Create Oracle Cloud Infrastructure user accounts and a group to which the user accounts belong. If the relevant group with user accounts exist, you need not create them. For instructions on creating users and groups, see Creating Groups and Users.

2. **Group Policies**

You need to create the policies and then map them to the group. To create the policies, go to **OCI** > **Identity & Security** > **Policies** > **Create Policy**. Create and add the following policies to the desired group:

- Allow group <*Group_Name*> to use metrics in compartment <*Compartment_Name*>
- Allow group <*Group_Name*> to manage alarms in compartment <*Compartment_Name*>
- Allow group <*Group_Name*> to manage ons-topics in compartment <*Compartment_Name*>
- Allow group <*Group_Name*> to inspect metrics in compartment <*Compartment_Name*>
- Allow group <*Group_Name*> to read metrics in compartment <*Compartment_Name*>
- Allow group <*Group_Name*> to use tag-namespaces in compartment <*Compartment_Name*>
- Allow group <*Group_Name*> to read log-groups in compartment <*Compartment_Name*>
- Allow group <*Group_Name*> to use instance-pools compartment <*Compartment_Name*>
- Allow group <*Group_Name*> to use cloud-shell in tenancy
- Allow group <*Group_Name*> to read objectstorage-namespace in tenancy
- Allow group <*Group_Name*> to manage repos in tenancy

**Note** You can create policies at tenancy level as well. It is at your discretion how you want to provide all the permissions.

3. **Permission for Oracle Functions**

   To enable a Oracle-Function to access another Oracle Cloud Infrastructure resource, include the function in a dynamic group, and then create a policy to grant the dynamic group access to that resource.

4. **Create Dynamic Group**

   To create dynamic groups, go to **OCI** > **Identity & Security** > **Dynamic Group** > **Create Dynamic Group**

   Specify the following rule while creating the dynamic group:

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

   For more details on dynamic groups, see:

   - 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. **Create Policy for Dynamic Group**

   To add policy, go to **OCI** > **Identity & Security** > **Policies** > **Create Policy**. Add the following policy to the group:

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

### Download files from GitHub

FTDv – OCI Autoscale solution is delivered as a GitHub repository. You can pull or download the files from the repository.

### Python3 Environment

A *make.py* file can be found in the cloned repository. This program compresses the oracle functions and template files into a Zip file; copy them to a target folder. In order to do these tasks, the Python 3 environment should be configured.

**Note** This python script can be used only on Linux environment.

### Infrastructure Configuration

The following must be configured:

1. **VCN**

   Create VCN as required for your FTDv application. Create VCN with the Internet Gateway having at least one of the subnet attached with route to internet.

For information on creating VCN, see https://docs.oracle.com/en-us/iaas/Content/GSG/Tasks/creatingnetwork.htm.

2. **Application Subnets**

   Create subnets as required for your FTDv application. To implement the solution as per this use case, FTDv instance requires 4 subnets for its operation.

   For information on creating subnet, see https://docs.oracle.com/en-us/iaas/Content/Network/Tasks/managingVCNs_topic-Overview_of_VCNs_and_Subnets.htm#.

3. **Outside Subnet**

   Subnet should have route with '0.0.0.0/0' to Internet Gateway. This subnet contains the Outside interface of Cisco FTDv and the Internet-facing Load balancer. Ensure that the NAT Gateway is added for outbound traffic.

   For more information, see the following documents:

   - 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. **Inside Subnet**

   This is similar to the Application Subnets, with or without NAT/Internet gateway.



**Note** For FTDv health probes, you can reach the metadata server (169.254.169.254) through Port 80.

5. **Management Subnet**

   Management subnet should be public so that it supports SSH accessibility to the FTDv.

6. **Function Subnet**

   This subnet is for Oracle Functions deployment.



**Note** This subnet must have `0.0.0.0/0` route to NAT GW (not Internet GW).

The Public IP of NAT GW of this subnet must be allowed in NSG (Network Security Group) of the Firewall Management Center Virtual and Firewall Threat Defense Virtual.

7. **Security Groups- Network Security Group for FTDv Instance**

   Configure the security group for FTDv instances that allows the Oracle Functions(in same VCN) perform SSH connections to FTDv's management address.

8. **Object Storage Namespace**

   This object storage namespace is used for hosting static website, having configuration.txt file. You must create a pre-authenticated requests for the configuration.txt file. This pre-authenticated URL is used during the template deployment.

**Note** Ensure that the following configurations that are uploaded are accessible by the FTDv instances through HTTP URL.

When FTDv is booted, it executes the following command`$ copy /noconfirm <configuration.txt file's pre-authenticated request URL > disk0:Connfiguration.txt`

This command enables FTDv launch to be configured with configuration.txt file.

# Secure Firewall Management Center Prerequisites

You can manage the Firewall Threat Defense Virtual devices using the Secure Firewall Management Center, a full-featured, multidevice manager. The Firewall Threat Defense Virtual registers and communicates with the FMC on the Management interface that you allocated to the Firewall Threat Defense Virtual virtual machine.

Create the objects required for Firewall Threat Defense Virtual configuration and management, including a device group to deploy policies and install updates on multiple devices. All the configurations applied on the device group is pushed to the Firewall Threat Defense Virtual instances.

The following sections provide a brief overview of basic steps to prepare the Firewall Management Center. For complete information on the procedure, refer *Secure Firewall Management Center Configuration Guide*. When you prepare the Firewall Management Center, make sure you record the following information:

- Secure Firewall Management Center Public IP Address
- Username and Password (If memory based scaling is enabled, you have to provide 2 user credentials)
- Security Zone Names
- Secure Firewall Management Center Access policy Name
- Secure Firewall Management Center NAT Policy Name
- Device Group Name

## Create User in Secure Firewall Management Center

Create a new user in Secure Firewall Management Center with Admin privileges to be used only by Autoscale Manager.

**Note** You must have an Secure Firewall Management Center user account dedicated to the Firewall Threat Defense Virtual Autoscale solution to prevent conflicts with other FMC sessions.

### Procedure

Create new user in Secure Firewall Management Center with Admin privileges. Choose **System** > **Users** and click **Create User**. The username must be Linux-valid:

- Maximum 32 alphanumeric characters, plus hyphen (-) and underscore (_)
- All lowercase
- Should not start with hyphen (-); must have alphabets; should not include a period (.), at sign (@), or slash (/)

Complete user options as required for your environment. See the *Secure Firewall Management Center Configuration Guide* for complete information.

## Create Device Group

Device groups enable you to easily assign policies and install updates on multiple devices. A device group should be created and rules should be applied on it. All the configurations applied on the device group are pushed to Firewall Threat Defense Virtual instances.

### Procedure

**Step 1** Choose **Devices** > **Device Management**.

**Step 2** From the **Add** drop-down menu select **Add Group**.

**Step 3** Enter the Device group name.

**Step 4** Click **Ok** to create the device group.

## Create Network and Host Objects

Create the following objects to be used for Firewall Threat Defense Virtual configuration.

### Procedure

**Step 1** Create Host with its name as *oci-metadata-server* and its IP as *169.254.169.254*.

**Step 2** Create a port with its name as health-check-port and its value as 8080 or any other port as required.

**Step 3** Create Inside interface, choose **Interface** > **Security Zone**. Select type as **Routed**. Provide a name for the interface, example, *inside-sz*.

**Step 4** Create Outside interface, choose **Interface** > **Security Zone**. Select type as **Routed**. Provide a name for the interface, example, *outside-sz*.

## Create NAT Policy

Create a NAT policy and create the necessary NAT rules to forward traffic from the outside interface to your application, and attach this policy to the device group you created for Autoscale.

### Procedure

**Step 1** Choose **Devices** > **NAT**

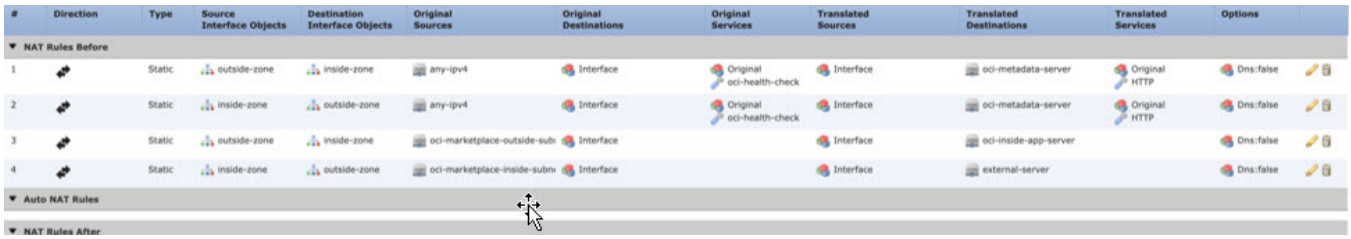
**Step 2** From the **New Policy** drop-down list, choose **Threat Defense NAT**.

**Step 3** Enter a unique **Name**.

**Step 4** Optionally, enter a **Description**.

**Step 5** Configure NAT rules. Refer Configure NAT for Threat Defense in the Secure Firewall Management Center Device Configuration Guide for guidelines on how to create NAT rules and apply NAT policies. The following figure shows a basic approach in setting the rules.

***Figure 41: NAT Rules***




**Step 6** Click **Save**.

## Create NAT Rules

A typical NAT rule converts internal addresses to a port on the outside interface IP address. This type of NAT rule is called interface Port Address Translation (PAT). See Configure NAT for Threat Defense in the Secure Firewall Management Center Device Configuration Guide for more information.

Configure the following 2 mandatory rules that are required in your NAT policy:

### Procedure

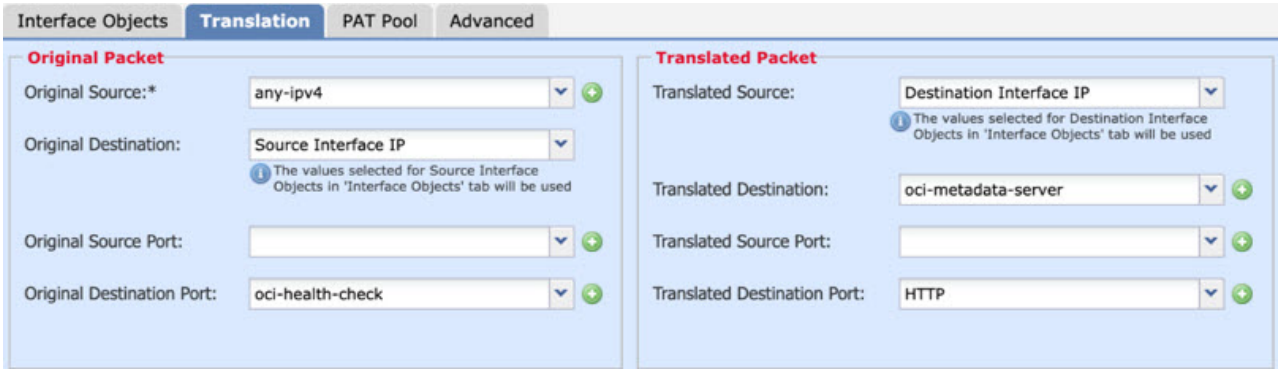
**Step 1** Configure the following NAT Rule for Inbound health check:

- Source Zone : Outside Zone
- Destination Zone : Inside Zone
- Original-sources : any-ipv4
- Original Destinations: Source Interface IP
- Original source port: Default
- Original-destination-port: health-check-port
- Translated-sources: Destination Interface IP
- Translated-destination: oci-metadata-server

- Translated source port: default
- Translated-destination-port: HTTP

The following figure shows the NAT rule for inbound health check.

***Figure 42: Inbound health NAT rule***

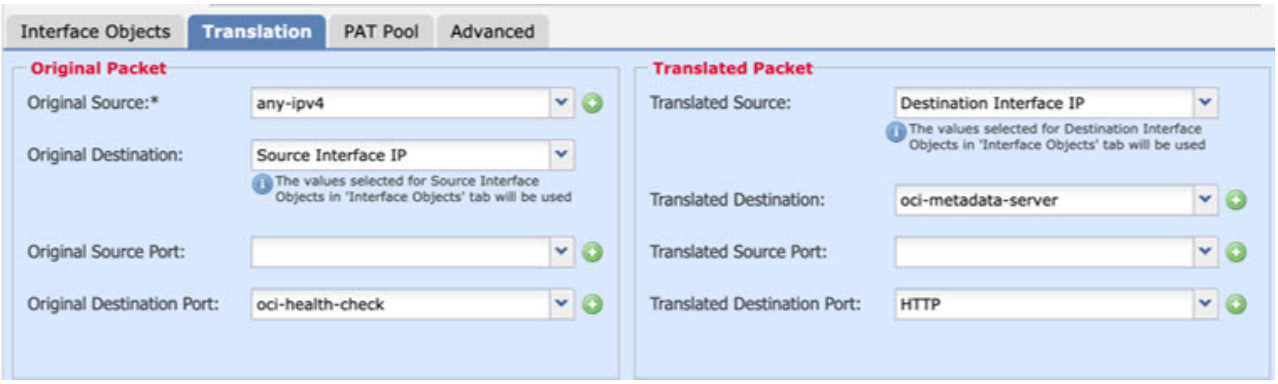



**Step 2** Configure the following NAT rule for outbound health check.

- Source Zone : Inside Zone
- Destination Zone : Outside Zone
- Original-sources : any-ipv4
- Original Destinations: Source Interface IP
- Original source port: Default
- Original-destination-port: health-check-port
- Translated-sources: Destination Interface IP
- Translated-destination: oci-metadata-server
- Translated source port: default
- Translated-destination-port: HTTP

The following figure shows the NAT rule for outbound health check.

*Figure 43: Outbound health check NAT rule*




Similarly, any NAT rules can be added for data traffic, and this configuration pushes them to Firewall Threat Defense Virtual devices.

## Create an Access Policy

Configure access control to allow traffic from inside to outside. An Access Policy with all required policies can be created, health port object should be allowed such that traffic on this port is allowed to reach the device. Within an access control policy, access control rules provide a granular method of handling network traffic across multiple managed devices. Proper configuration and sequencing of the rules are essential to build an effective deployment. See the Best Practices for Access Control Rules in the Secure Firewall Management Center Device Configuration Guide.

Assign the device group (created as part of pre-requisites) to the access policy using **Policy Assignments**.

### Procedure

**Step 1** Choose **Policies** > **Access Control**.

**Step 2** Click **New Policy**.

**Step 3** Enter a unique Name and, optionally, a Description.

**Step 4** Configure security settings and rules for your deployment. For more information, see Access Control in the Secure Firewall Management Center Device Configuration Guide.

# Encrypt Password

**Note** For more information on this procedure, see Create Vaults and Secrets.

Password for FTDv is used to configure all the FTDv instances being used while autoscaling and it is used to create connections for Rest APIs calls for several configuration purpose.

Therefore, you need to save and process the password every now and then. Owing to the frequent changes and vulnerability, editing or saving the password in the plain-text format is not allowed. Password must be in an encrypted format only.

To obtain password in encrypted form:

## Procedure

**Step 1** Create Vault.

OCI Vault provides services to create and save master encryption keys safely, and methods for encryption and decryption in using them. So Vault should be created (if not having already) in the same compartment as the rest of the autoscale solution.

Go to **OCI** > **Identity & Security** > **Vault** > **Choose or Create New Vault**

.

**Step 2** Create Master Encryption Key.

One master encryption key is needed to encrypt the plain text password.

Go to **OCI** > **Identity & Security** > **Vault** > **Choose or Create Key**

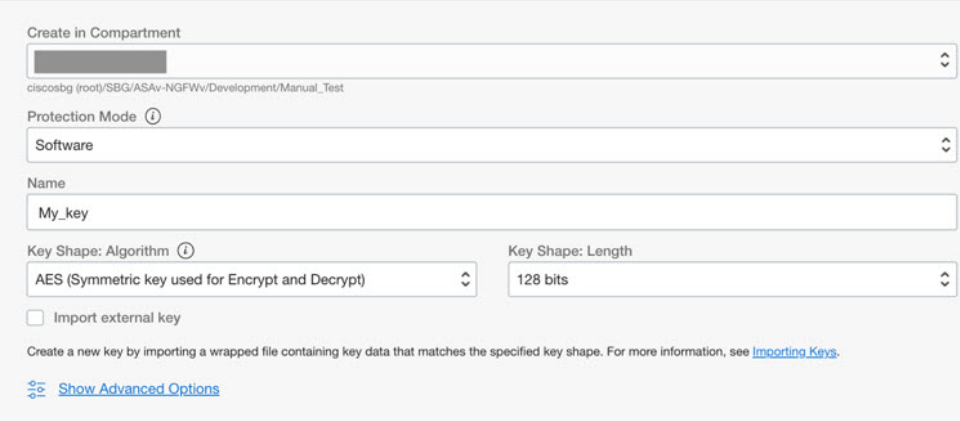
Choose any of the keys from any of the given algorithm with any bit of length.

**a.** AES – 128, 192, 256

**b.** RSA – 2048, 3072, 4096

**c.** ECDSA – 256, 384, 521

***Figure 44: Create Key***




**Step 3** Create encrypted password.

**a.** Go to **OCI** > **Open CloudShell (OCI Cloud Terminal)**

**b.** Execute following command by replacing <*Password*> as your password.

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

**c.** From the selected Vault, copy cryptographic endpoint and master encryption key OCID. Replace the following values, and then execute the encrypt command:

- KEY_OCID with Your key's OCID
- Cryptographic_Endpoint_URL with Your vault's cryptographic endpoint URL
- Password with Your password

**Encrypt Command**

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

**d.** Copy ciphertext from output of the above command and use it as required.

# Preparation of Firewall Threat Defense Virtual configuration File

It is expected that Application is either deployed or its deployment plan is available.

### Procedure

**Step 1** Collect the following input parameters before deployment:

| Parameter | Data Type | Description |
|---|---|---|
| tenancy_ocid | String | OCID of the tenancy to which your account belongs. To know how to find your tenancy OCID, see here.<br>The tenancy OCID looks something like this - `ocid1.tenancy.oc1..<unique_ID>` |
| region | String | The unique identifier of the region in which you want the resources to be created.<br>Example - us-phoenix-1, us-ashburn-1 |
| lb_size | String | A template that determines the total pre-provisioned bandwidth (ingress plus egress) of the external and internal load balancer.<br>Supported values: 100Mbps, 10Mbps, 10Mbps-Micro, 400Mbps, 8000Mbps<br>Example : 100Mbps |

| Parameter | Data Type | Description |
|---|---|---|
| availability_domain | String | Example - Tpeb:PHX-AD-1, Tpeb:PHX-AD-2<br>**Note**<br>To get the availability domain names, see here. |
| min_and_max_instance_count | comma separated value | The minimum and the maximum number of instances that you would want to retain in the instance pool.<br>Example: 1,5 |
| autoscale_group_prefix | String | The prefix to be used to name all the resources that are created using the template. For example, if the resource prefix is given as 'autoscale', all the resources are named as follows - autoscale_resource1, autoscale_resource2 etc. |
| mgmt_subnet_ocid | String | OCID of the Management subnet that is to be used. |
| inside_subnet_ocid | String | OCID of the Inside subnet that is to be used. |
| function_subnet_ocid | String | OCID of the Function subnet that is to be used. |
| outside_subnet_ocid | String | OCID of the Outside subnet that is to be used. |
| mgmt_nsg_ocid | String | OCID of the Management subnet network security group that is to be used. |
| inside_nsg_ocid | String | OCID of the Inside subnet network security group that is to be used. |
| outside_nsg_ocid | String | OCID of the Outside subnet network security group that is to be used. |
| elb_listener_port | comma separated Values | List of the communication ports for the external load balancer listener.<br>Example: 80 |
| ilb_listener_port | comma separated Values | List of the communication ports for the internal load balancer listener.<br>Example: 80 |
| health_check_port | String | The backend server port of load balancer against which to run the health check.<br>Example: 8080 |
| instance_shape | String | The shape of the instance to be created. The shape determines the number of CPUs, amount of memory, and other resources allocated to the instance.<br>Supported shapes :"VM.Standard2.4" & "VM.Standard2.8" |

| Parameter | Data Type | Description |
|---|---|---|
| lb_bs_policy | String | The load balancer policy to be used for the internal and external load balancer's backend set. To know more about how load balancer policies work, see here<br>Supported values: "ROUND_ROBIN", "LEAST_CONNECTIONS", "IP_HASH" |
| image_name | String | The name of the marketplace image to be used for creating the instance configuration.<br>Default value : " Cisco Firepower NGFW virtual firewall (NGFWv)"<br>**Note**<br>If the user wants to deploy custom image, user has to configure the custom_image_ocid parameter. |
| scaling_thresholds | Comma separated value | The CPU usage thresholds to be used for scale-in and scale-out. Provide the scale-in and scale-out threshold values as comma separated input.<br>Example : 15,50<br>where, 15 is the scale-in threshold and 50 is the scale-out threshold. |
| compartment_id | String | The OCID of the compartment in which to create the resources.<br>Example: **ocid1.compartment.oc1..<unique_ID>** |
| compartment_name | String | Name of the compartment |
| custom_image_ocid | String | OCID of the custom image to be used to create instance configuration if the marketplace image is not to be used.<br>**Note**<br>*custom_image_ocid is optional parameter* |
| ftdv_password | String | The password for Firewall Threat Defense Virtual in the encrypted form, to SSH into the Firewall Threat Defense Virtual for configuration. Use configuration guide for the instructions on how to encrypt password or see here. |
| ftdv_license_type | String | Type of Firewall Threat Defense Virtual license either BYOL or PAYG. Currently, BYOL is supported. |
| cryptographic_endpoint | String | Cryptographic endpoint is a URL, that is used for decrypting password. It can be found in the Vault. |

| Parameter | Data Type | Description |
|---|---|---|
| master_encryption_key_id | String | The OCID of key with which the password was encrypted. It can be found in the Vault.<br>**Note**<br>master_encryption_key_id and cryptographic_endpoint both must belong to same vault. |
| fmc_ip | String | IP address of Secure Firewall Management Center. IP of Firewall Management Center that will be used by customer to manage Firewall Threat Defense Virtual instances.<br>**Note**<br>*Firewall Management Center IP can be private only if it is in the same subnet as Firewall Threat Defense Virtual, otherwise Public IP must be used for all other cases.* |
| fmc_username | String | Username of the Firewall Management Center account. This username will be used to login into the Firewall Management Center to configure each time the new Firewall Threat Defense Virtual instance comes. |
| fmc_password | String | Password of Firewall Management Center in encrypted form. For procedure on how to encrypt password, see here. |
| fmc_device_group_name | String | There must be a device group in Firewall Management Center, all the Firewall Threat Defense Virtual part of this Autoscale solution will be added into that group, so that same policies and configuration can be applied to all of them. |
| enable_memory_based_scaling | Bool | Publish Firewall Threat Defense Virtual Memory usage from the Secure Firewall Management Center Virtual. By enabling this flag Scaling can happen based on Memory utilization as well. By default CPU utilization is used. |
| fmc_metrics_username | String | In case you opt for Memory Utilization by enabling flag enable_memory_based_scaling, an extra Firewall Management Center user account is needed as that will be used continuously to pull memory usage from all the running Firewall Threat Defense Virtual instances. |
| fmc_metrics_password | String | Password of extra Firewall Management Center account in encrypted form. For procedure on how to encrypt password, see here. |

| Parameter | Data Type | Description |
|---|---|---|
| Profile Name | | It is the User's profile name in OCI. It can be found under profile section of the user. Example: "oracleidentitycloudservice/ <user>@<mail>.com" |
| Object Storage Namespace | | It is unique identifier created at the time of Tenancy creation. Go to **OCI** > **Administration** > **Tenancy Details** |
| Authorization Token | | This is used as password for docker login which authorizes it to push Oracle-Functions into the OCI container registry. Go to **OCI** > **Identity** > **Users** > **User Details** > **Auth Tokens** > **Generate Token**. |

**Step 2** Create the *Configuration.json* file with the following content:

```
{
  "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"
    }
  ]
}
```

**Step 3** Update *Configuration.json* with the configuration settings.

**Step 4** Upload Configuration file to Object Storage space.

The *configuration.txt* file must be uploaded to the user created object storage space and the pre-authenticated request for the uploaded file should be created.

**Note**
Ensure that pre-authenticated request URL of configuration.txt is used in the stack deployment.

**Note**
Expiry period is needed to be defined while creating pre-authenticated URL in OCI, make sure this period is long enough to not expire during solution execution.

**Step 5** Create the Zip files.

A *make.py* file can be found in the cloned repository. Execute the `python3 make.py build` command to create the zip files. The target folder has the following files.

```
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
```

# Deploy the Auto Scale Solution

After completing the pre-requisite steps for deployment, start creating the OCI stack. You can perform a manual deployment or perform deployment using the cloud shell. Deployment scripts and templates for your version are available in the GitHub repository.

## Manual Deployment

End-to-end Autoscale solution deployment consist of three steps: Deploy Terraform Template-1 Stack, Deploy Oracle Functions, and then Deploy Terraform Template-2.

# Deploy Terraform Template-1 Stack

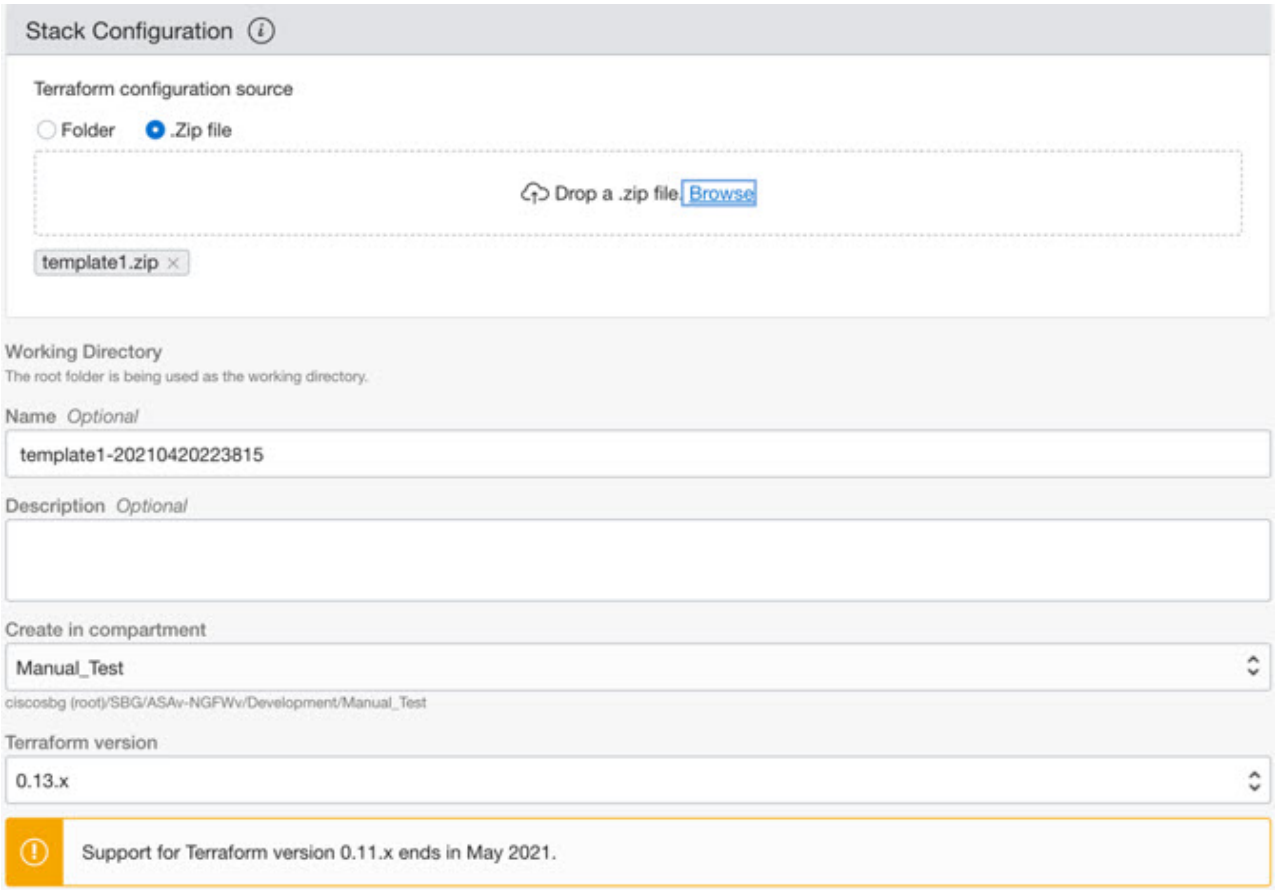
## Procedure

**Step 1** Log into the OCI portal.

The region is displayed in the upper right corner of your screen. Make sure you are in the intended region.

**Step 2** Choose **Developer Service** > **Resource Manager** > **Stack** > **Create Stack**

Choose **My Configuration**, and then select the *Terraform template1.zip* file in the target folder as Terraform Configuration Source as shown in the figure below.




**Step 3** In the **Transform version** drop-down list, select 0.13.x or 0.14.x.

**Step 4** In the next step, enter all the details as collected in Collection of Input Parameters.

**Note**
Enter valid input parameters, otherwise stack deployment may fail in further steps.

**Step 5** In the next step, choose **Terraform Actions** > **Apply**.

Post successful deployment, proceed to deploy the Oracle functions.

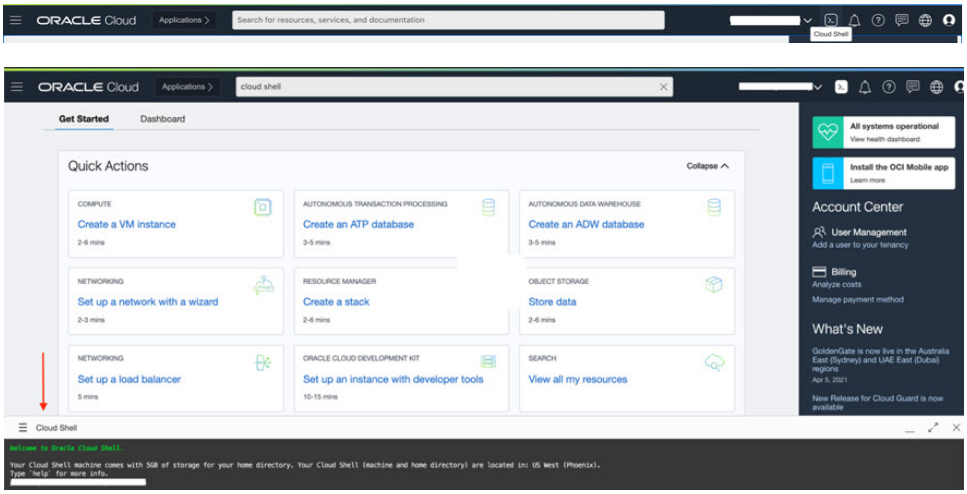
## Deploy Oracle Functions

**Note** *This step must be performed only after successful Terraform Template-1 deployment.*

In OCI, Oracle Functions are uploaded as Docker Images, which are saved into the OCI container registry. Oracle Functions are needed to be pushed into one of the OCI Application (created in Terraform Template-1) at the time of deployment.

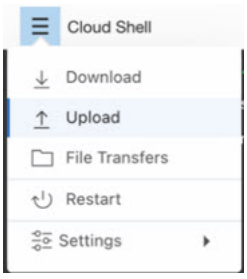
### Procedure

**Step 1** Open OCI Cloud Shell.

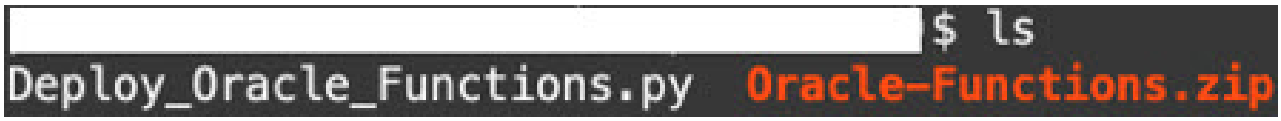



**Step 2** Upload *deploy_oracle_functions_cloudshell.py* and *Oracle-Functions.zip*.

From the Cloud Shell's hamburger menu, choose **Upload**.




**Step 3** Verify files using the **ls** command.




**Step 4** Run `python3 Deploy_Oracle_Functions.py -h`. The `deploy_oracle_functions_cloudshell.py` script requires some input parameters whose details can be found using help argument, as shown in figure below.

```
$ 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 ***

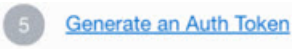
Instruction to find values of required arguments:
Application Name: Name of Application created by first Terraform Template
Region Identifier: OCI -> Administration -> Region Management
Profile Name: OCI -> Profile
Compartment OCID: OCI -> Identity -> Compartment -> Compartment Details
Object Storage 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)
```

To run the script pass the following arguments:

***Table 26: Arguments and Details***

| Argument | Particulars |
|---|---|
| **Application Name** | It is the name of OCI Application created by Terraform Template-1 deployment. Its value is obtained by combining **"autoscale_group_prefix"** given in Template-1 and suffix **"_application".** |
| **Region Identifier** | Region identifier is the region codeword fixed in the OCI for different regions.<br>Example: 'us-phoenix-1' for Phoenix or "ap-melbourne-1" for Melbourne.<br>To get the list of all region with their region identifiers, go to **OCI** > **Administration** > **Region Management**. |
| **Profile Name** | It is simple User's profile name in OCI.<br>Example: *oracleidentitycloudservice/<user>@<mail>.com*<br>The name can be found under profile section of the user. |
| **Compartment OCID** | It is the compartment's OCID (Oracle Cloud Identifier). Compartment OCID where user have the OCI Application.<br>Go to **OCI** > **Identity** > **Compartment** > **Compartment Details**. |

| Argument | Particulars |
|---|---|
| **Object Storage Namespace** | It is unique identifier created at the time of Tenancy creation.<br>Go to **OCI** > **Administration** > **Tenancy Details**. |
| **Authorization Token** | This is used as password for docker login which authorizes it to push Oracle-Functions into the OCI container registry. Specify the token in quotes in the deployment script.<br>Go to **OCI** > **Identity** > **Users** > **User Details** > **Auth Tokens** > **Generate Token**.<br>For some reason, if you are not able to see User Details then click **Developer services** > **Functions**. Go to the application created by Terraform Template-1. Click **Getting Started**, and choose Cloud Shell Setup and among the steps you will find the link to generate auth token as shown below.<br>  |

**Step 5** Run the `python3 Deploy_Oracle_Functions.py` command by passing valid input arguments. It will take some time to deploy all the functions. You can then remove the file and close the Cloud Shell.

## Deploy Terraform Template-2

Template 2 deploys the resources related to alarm creation, including alarms, ONS topics for invoking function. The deployment of template 2 is similar to Terraform Template-1 deployment.

### Procedure

**Step 1** Log into the OCI portal.

The region is displayed in the upper right corner of your screen. Make sure you are in the intended region.

**Step 2** Choose **Developer Service** > **Resource Manager** > **Stack** > **Create Stack**.

Select *Terraform template template2.zip* in the target folder as source of Terraform configuration.

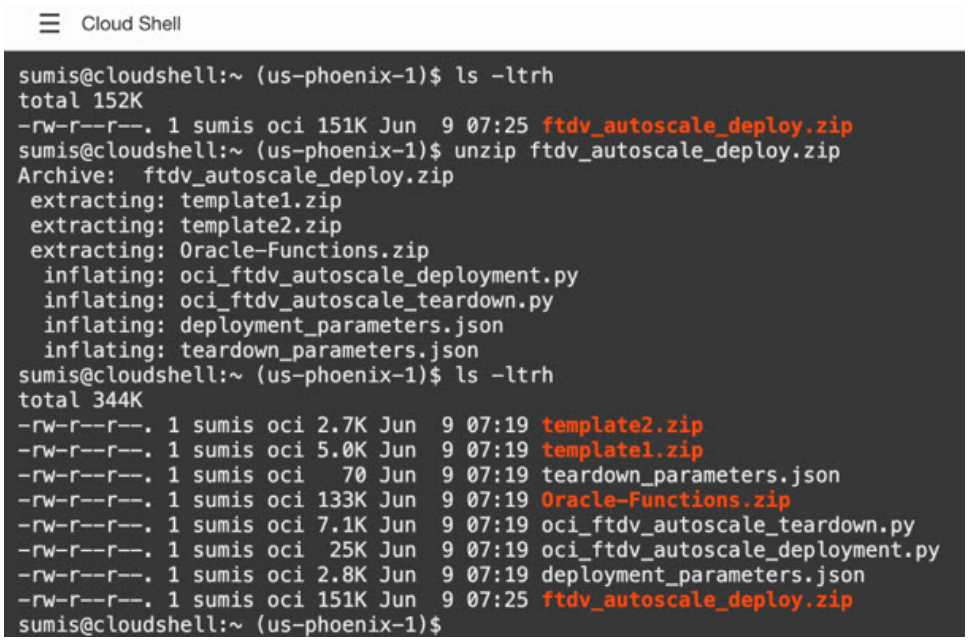
**Step 3** In next step, click **Terraform Actions** > **Apply**.

# Deployment using cloud shell

To avoid the deployment overhead, you can invoke the easy, end-to-end deployment script to deploy the autoscale solution (terraform template1, template2 and oracle functions).

### Procedure

**Step 1** Upload the *ftdv_autoscale_deploy.zip* file in the target folder to the cloud shell and extract the files.




**Step 2** Make sure you have updated the input parameters in the *deployment_parameters.json* before executing the `python3 make.py` build command.

**Step 3** To start the autoscale solution deployment, run the `python3 oci_ftdv_autoscale_deployment.py` command on the cloud shell.

It will take approximately 10-15 minutes for the solution deployment to complete.

If there is any error during the solution deployment, error log is saved.

# Validate Deployment

Validate if all resources are deployed and the Oracle Functions are connected with Alarm & Events. By default, instance pool has minimum and maximum number of instances as zero. You can edit the instance pool in OCI UI with the minimum and maximum number that you want. This will trigger new Firewall Threat Defense Virtual instances.

We recommend that you launch only one instance and check for its workflow, validate its behaviour to ensure that it is working as it is expected. Post this validation, you can deploy the actual requirements of Firewall Threat Defense Virtual.



**Note** Specify the minimum number of Firewall Threat Defense Virtual instances as **Scale-In protected** to avoid their removal by OCI scaling policies.

# Upgrade

### Upgrade Autoscale Stack

No support for upgrade in this release. Stacks should be re-deployed.

### Upgrade Firewall Threat Defense Virtual VMs

No support for upgrade for Firewall Threat Defense Virtual VMs in this release. The Stack should be re-deployed with the required Firewall Threat Defense Virtual image.

### Instance Pool

1. To change minimum and maximum number of instances in the Instance Pool:

   Click **Developer Services** > **Function** > **Application Name(created by Terraform Template 1)** > **Configuration**.

   Change the min_instance_count and max_instance_count respectively.

2. Deletion/Termination of Instance is not equal to Scale-in. If any instance in the Instance Pool is deleted/terminated due to external action and not the scale-in action, instance pool automatically initiates a new instance to recover.

3. Max_instance _count defines threshold limit for Scale-out action, but it can be surpassed by changing the instance count of the Instance Pool through the UI. Ensure that the instance count from UI is less than max_instance_count set in OCI Application. Else, increase the threshold accordingly.

4. Reducing the count of instances in Instance Pool directly from the application does not perform the clean-up actions set programmatically. Due to which backends will not be drained and removed from both the load balancers, if Firewall Threat Defense Virtual has license, it will be lost.

5. Due to some reasons, if Firewall Threat Defense Virtual instance is unhealthy, not responding and unreachable through SSH for some definite period of time, instance is removed from the instance pool forcefully, any license may be lost.

### Oracle Functions

- Oracle Functions are actually docker images. These images are saved into root directory of OCI Container registry. These images should not be deleted as it will also delete the function that are used in the Autoscale solution.
- OCI Application created by Terraform template-1, contains crucial environmental variables, which are required by Oracle Functions to work properly. Neither the value nor the format of these environment variables should be changed, unless it is mandated. Any changes made are reflected with new instances only.

# Load Balancer Backend Sets

In OCI, Load Balancer attachment to instance pool is only supported using primary interface that is configured as management interface in Firewall Threat Defense Virtual. Hence, inside interface is connected to Internal Load Balancer's backend set; outside interface is connected to External load balancer's backend set. These IPs are not automatically added or removed from backend set. The Autoscale solution programmatically handles both of this task. But in case of any external action, maintenance or troubleshooting, there could be situation demanding manual effort to operate on them.

As per requirements, more ports can be opened on Load Balancer using listener and backend sets. Upcoming instances IPs are automatically added to the backend set, however already existing instances IPs should be manually added.

**Adding Listener in Load Balancer**

To add some port as listener in Load Balancer, go to **OCI** > **Networking** > **Load Balancer** > **Listener** > **Create Listener**.

**Register a backend to Backend Set**

In order to register an Firewall Threat Defense Virtual instance to Load Balancer, Firewall Threat Defense Virtual instance Outside interface IP should be configured as a backend in the Backend Set of External Load Balancer. Inside interface IP should be configured as backend in Backend set of Internal Load Balancer. Ensure that the port you are using has been added into the listener.

# Delete Autoscale Configuration from OCI

Stacks deployed using terraform can be deleted in the same manner, using Resource Manager in OCI. Deletion of stack removes all the resources created by it and all the information associated with these resources are removed permanently.

**Note** In case of stack deletion, it is recommended to make the Minimum number of instances in Instance pool to 0, wait for instances to be terminated. This will help removal of all instances and won't leave any residue.

You can perform a manual deletion or use Cloud Shell.

# Manual Deletion

The end-to-end autoscale solution deletion consist of three steps: Delete Terraform Template-2 Stack, Delete Oracle Functions, and then Delete Terraform Template-1 Stack.

## Delete Terraform Template-2 Stack

To delete the Autoscale configuration, you must begin with Terraform Template-2 stack deletion.

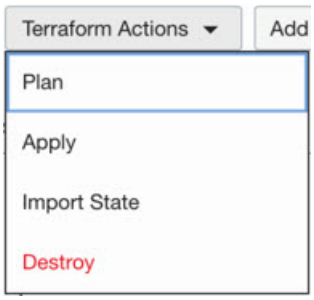
### Procedure

**Step 1** Log into the OCI portal.

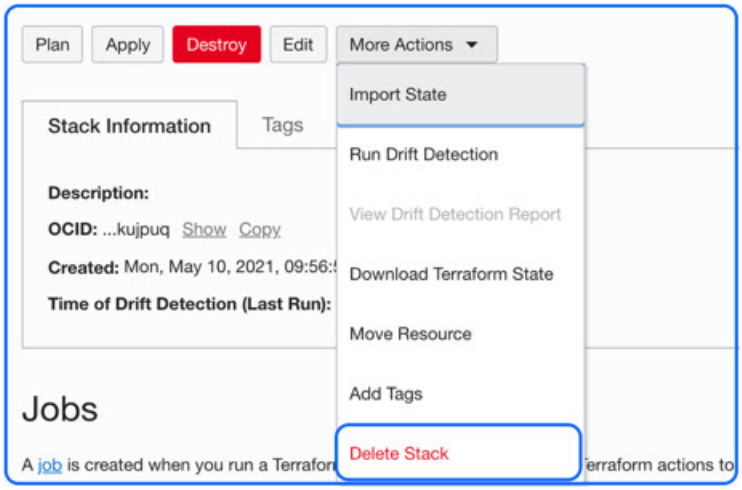
The region is displayed in the upper right corner of your screen. Make sure you are in the intended region.

**Step 2** Choose **Developer Services** > **Resource Manager** > **Stack**.

**Step 3** Select the stack created by Terraform Template-2, then select **Destroy** in **Terraform Actions** drop-down menu as shown in the figure below:




Destroy Job is created which takes some time to remove resources one after another. You can delete the stack after the destroy job is completed. as shown in the figure below:




**Step 4** Proceed to delete the Oracle functions.

## Delete Oracle-Functions

The Oracle-Function deployment is not a part of Terraform Template Stack deployment, it is uploaded separately using Cloud Shell. Hence, its deletion is also not supported by Terraform Stack deletion. You must delete all the Oracle-Functions inside the OCI application created by Terraform Template-1.

### Procedure

**Step 1** Log into the OCI portal.

The region is displayed in the upper right corner of your screen. Make sure you are in the intended region.

**Step 2** Choose **Developer Services** > **Functions**. Choose the application name that was created in Template-1 stack.

**Step 3** Inside this application visit each function and delete it.

## Delete Terraform Template-1 Stack

**Note** Template-1 Stack deletion will only succeed after deleting all Oracle-Functions.

Same as Terraform Template-2 Deletion.

### Procedure

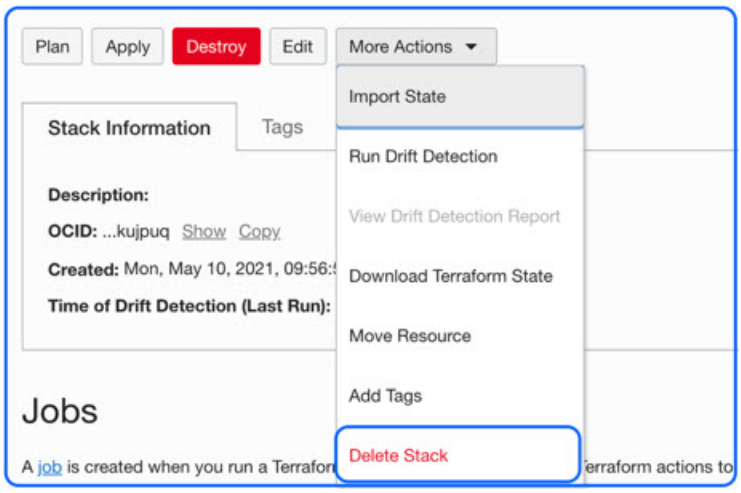
**Step 1** Log into the OCI portal.

The region is displayed in the upper right corner of your screen. Make sure you are in the intended region.

**Step 2** Choose **Developer Services** > **Resource Manager** > **Stack**.

**Step 3** Select the stack created by Terraform Template-2, then click **Destroy** in Terraform **Actions** drop-down menu. Destroy Job will be created which will take some time to remove resources one after another.

**Step 4** After the destroy job is completed, you can delete the stack from **More Actions** drop-down menu as shown in the figure below.



Post successful deletion of Terraform Template-1 stack, you must verify whether all the resources are deleted and there is no residue of any kind.

## Delete Autoscale Using Cloud Shell

User can use the script to delete the stacks and oracle functions by executing the `python3 oci_ftdv_autoscale_teardown.py` command in the cloud shell. If the stacks are deployed manually, update the stack id of the stack1 and stack2, and update the application id in the *teardown_parameters.json* file.

# Connect to the Firewall Threat Defense Virtual Instance Using SSH

To connect to the Firewall Threat Defense Virtual instance from a Unix-style system, log in to the instance using SSH.

**Procedure**

**Step 1** Use the following command to set the file permissions so that only you can read the file:

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

Where:

`<private_key>` is the full path and name of the file that contains the private key associated with the instance you want to access.

**Step 2** Use the following SSH command to access the instance:

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

`<private_key>` is the full path and name of the file that contains the private key associated with the instance you want to access.

`<username>` is the username for the Firewall Threat Defense Virtual instance.

`<public-ip-address>` is your instance IP address that you retrieved from the console.

# Connect to the Firewall Threat Defense Virtual Instance Using OpenSSH

To connect to the Firewall Threat Defense Virtual instance from a Windows system, log in to the instance using OpenSSH.

**Procedure**

**Step 1** If this is the first time you are using this key pair, you must set the file permissions so that only you can read the file.

Do the following:

a) In Windows Explorer, navigate to the private key file, right-click the file, and then click **Properties**.
b) On the **Security** tab, click **Advanced**.
c) Ensure that the **Owner** is your user account.
d) Click **Disable Inheritance**, and then select **Convert inherited permissions into explicit permissions on this object**.
e) Select each permission entry that is not your user account and click **Remove**.
f) Ensure that the access permission for your user account is **Full control**.
g) Save your changes.

**Step 2** To connect to the instance, open Windows PowerShell and run the following command:

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

Where:

`<private_key>` is the full path and name of the file that contains the private key associated with the instance you want to access.

`<username>` is the username for the Firewall Threat Defense Virtual instance.

`<public-ip-address>` is your instance IP address that you retrieved from the Console.

# Connect to the Firewall Threat Defense Virtual Instance Using PuTTY

To connect to the Firewall Threat Defense Virtual instance from a Windows system using PuTTY:

**Procedure**

**Step 1** Open PuTTY.

**Step 2** In the **Category** pane, select **Session** and enter the following:

- **Host Name (or IP address):**

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

  Where:

  `<username>` is the username for the Firewall Threat Defense Virtual instance.

  `<public-ip-address>` is your instance public IP address that you retrieved from the Console.

- **Port:** 22

- **Connection type:** SSH

**Step 3** In the **Category** pane, expand **Window**, and then select **Translation**.

**Step 4** In the **Remote character set** drop-down list, select **UTF-8**.

The default locale setting on Linux-based instances is UTF-8, and this configures PuTTY to use the same locale.

**Step 5** In the **Category** pane, expand **Connection**, expand **SSH**, and then click **Auth**.

**Step 6** Click **Browse**, and then select your private key.

**Step 7** Click **Open** to start the session.

If this is your first time connecting to the instance, you might see a message that the server's host key is not cached in the registry. Click **Yes** to continue the connection.

# IPv6 Troubleshooting

**Problem** SSH—Firewall Threat Defense Virtual with IPv6 is not working

- **Solution** Make sure the route for IPv6 public access via internet gateway is added.
- **Solution** Enable IPv6 is present in the Firewall Threat Defense Virtual management configuration.
- **Solution** Verify IPv6 related access-list been added to Firewall Threat Defense Virtual deployed.
- **Solution** Verify if “ipv6 address dhcp default” is used for configuring IPv6 on management interface. If used “ipv6 address dhcp” only then add the following route separately “ipv6 route management ::/0 <IPv6_Gateway_address>.”
- **Solution** Verify if proper ssh ingress is allowed. Use following command to set ssh access allow for all “ssh ::/0 management.”

**Problem** Not able to assign IPv6 address to existing subnet.

- **Solution** Verify if VCN to which the subnet belongs is enabled with IPv6 already.
- **Solution** Make sure that correct IPv6 CIDR is being used.
- **Solution** Subnet can only have “/64” IPv6 CIDR prefix.

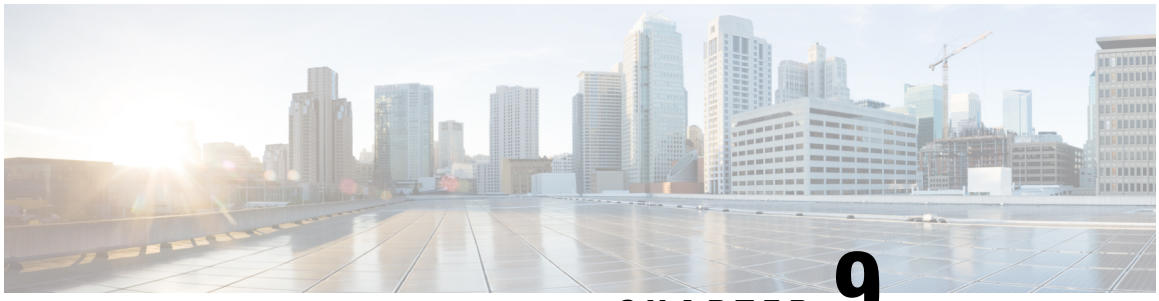
**Problem** East-West traffic not working.

- **Solution** Verify if following routes are added properly.

  **Solution** ipv6 route <interface_name> <interface_subnet_CIDR> <ipv6_virtual_router_ip>

  **Solution** Example: ipv6 route inside 2603:c020:5:5800::/56 fe80::200:17ff:fe96:921b

- **Solution** Make sure that correct IPv6 CIDR is being used.
- **Solution** Make sure if proper access list is configured for IPv6.



CHAPTER 9

# Deploy the Firewall Threat Defense Virtual on GCP

You can deploy the Firewall Threat Defense Virtual on the Google Cloud Platform (GCP), a public cloud computing service that enables you to run your applications in a highly-available, hosted environment offered by Google.

You see the GCP project information in the GCP console **Dashboard**.

- Make sure that you select your GCP project in the **Dashboard** if that is not already selected.
- To access the Dashboard, click **Navigation menu** > **Home** > **Dashboard**.

You log into the GCP Console, search the GCP Marketplace for the Cisco Firepower NGFW virtual firewall (NGFWv) offering, and launch the Firewall Threat Defense Virtual instance. The following procedures describe how to prepare your GCP environment and launch the Firewall Threat Defense Virtual instance to deploy the Firewall Threat Defense Virtual.


- Overview, on page 302
- End-to-End Procedure, on page 303
- Prerequisites, on page 305
- Guidelines and Limitations for the Firewall Threat Defense Virtual and GCP, on page 306
- NIC Mapping to Data Interfaces, on page 308
- Sample Network Topology, on page 309
- How to Manage Secure Firewall Threat Defense Virtual Device, on page 309
- Configure GCP Environment, on page 310
- Create the Firewall Rules, on page 312
- Deploy the Firewall Threat Defense Virtual, on page 315
- Connect to the Firewall Threat Defense Virtual Instance Using an External IP, on page 319
- Connect to the Firewall Threat Defense Virtual Instance Using the Serial Console, on page 320
- Connect to the Firewall Threat Defense Virtual Instance Using Gcloud, on page 320
- About Deployment of Threat Defense Virtual without Diagnostic Interface on GCP, on page 321
- Guidelines and Limitations for Deployment of Threat Defense Virtual without Diagnostic Interface, on page 321
- NIC Mapping to Data Interfaces for Deployment of Threat Defense Virtual without Diagnostic Interface on GCP, on page 322
- Deploy Threat Defense Virtual without Diagnostic Interface on GCP, on page 322
- Upgrade Scenarios, on page 323

- Deployment of Threat Defense Virtual Cluster or Auto Scale Solution without Diagnostic Interface, on page 324
- Troubleshooting, on page 324
- Auto Scale Solution, on page 324
- Download the Deployment Package, on page 327
- System Requirements, on page 327
- Prerequisites, on page 330
- Deploy the Auto Scale Solution, on page 339
- Auto Scale Logic, on page 345
- Logging and Debugging, on page 345
- Troubleshooting, on page 346


# Overview

The Firewall Threat Defense Virtual runs the same software as physical Secure Firewall Threat Defense (formerly Firepower Threat Defense) to deliver proven security functionality in a virtual form factor. The Firewall Threat Defense Virtual can be deployed in the public GCP. It can then be configured to protect virtual and physical data center workloads that expand, contract, or shift their location over time.

# System Requirements

Select the Google virtual machine type and size to meet your Firewall Threat Defense Virtual needs. Currently, the Firewall Threat Defense Virtual supports both compute-optimized and general purpose machine types (standard, high-memory, and high-CPU machine types).

**Note** Supported machine types may change without notice.

*Table 27: Supported Compute-Optimized Machine Types*

| Compute-Optimized Machine Types | Attributes | | |
|---|---|---|---|
| | vCPUs | RAM (GB) | vNICs |
| c2-standard-4 | 4 | 16 GB | 4 |
| c2-standard-8 | 8 | 32 GB | 8 |
| c2-standard-16 | 16 | 64 GB | 8 |

*Table 28: Supported General Purpose Machine Types*

| General Purpose Machine Types | Attributes | | |
|---|---|---|---|
| | vCPUs | RAM (GB) | vNICs |
| n1-standard-4 | 4 | 15 | 4 |

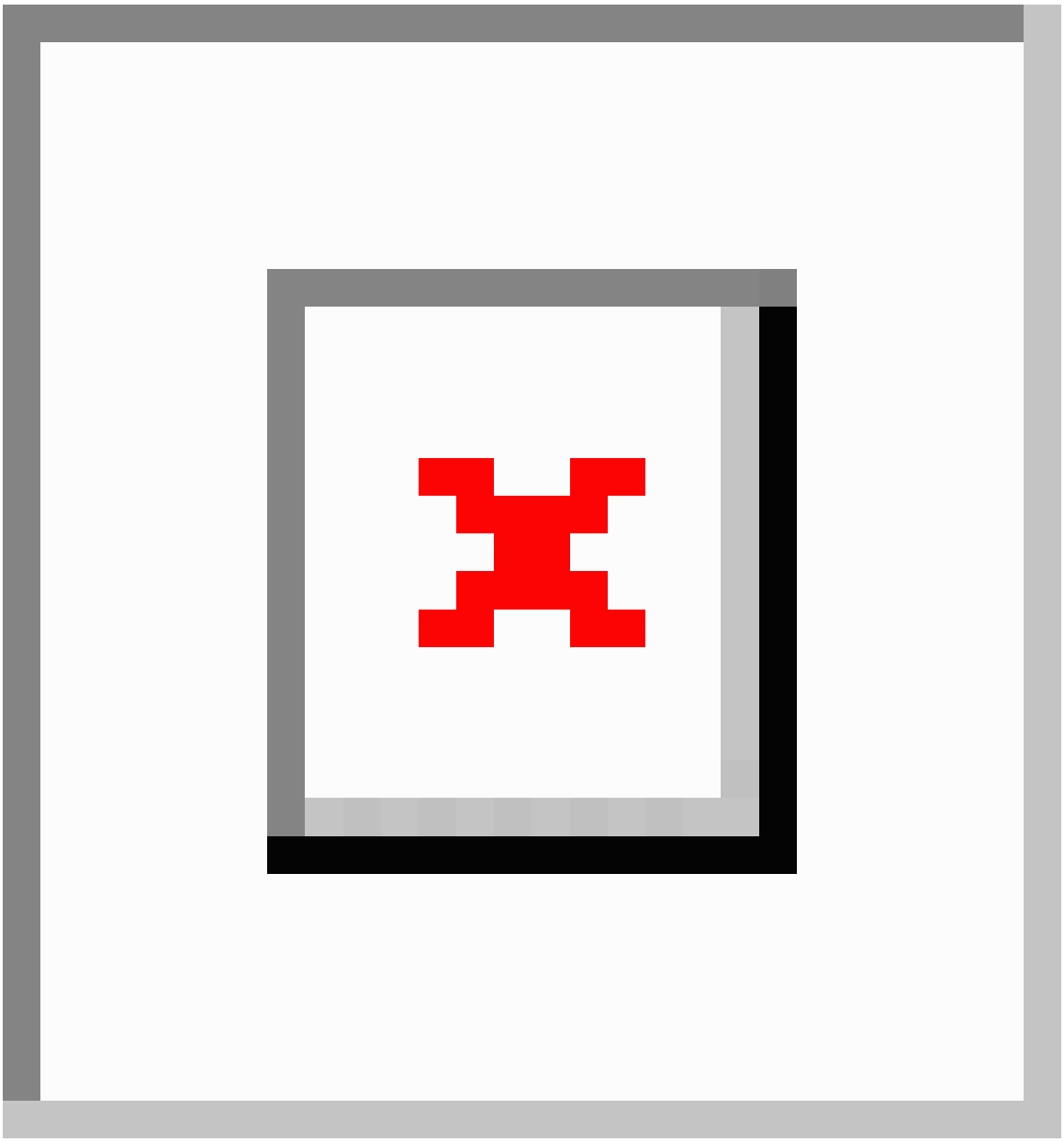
| General Purpose Machine Types | Attributes | | |
|---|---|---|---|
| | vCPUs | RAM (GB) | vNICs |
| n1-standard-8 | 8 | 30 | 8 |
| n1-standard-16 | 16 | 60 | 8 |
| n2-standard-4 | 4 | 16 | 4 |
| n2-standard-8 | 8 | 32 | 8 |
| n2-standard-16 | 16 | 64 | 8 |
| n2-highmem-4 | 4 | 32 | 4 |
| n2-highmem-8 | 8 | 64 | 8 |

- The Firewall Threat Defense Virtual requires a minimum of 4 interfaces.
- The maximum supported vCPUs is 16.

You create an account on GCP, launch a VM instance using the Cisco Firepower NGFW virtual firewall (NGFWv) offering on the GCP Marketplace, and choose a GCP machine type.

# End-to-End Procedure

The following flowchart illustrates the workflow for deploying Threat Defense Virtual on Google Cloud Platform.



| | Workspace | Steps |
|---|---|---|
| 1 | GCP | Deploy the Threat Defense Virtual on GCP: Create the VPC Network (**VPC Networks** > **Subnet** > **Region** > **IP address range**). |
| 2 | GCP | Deploy the Threat Defense Virtual on GCP: Create the firewall rules (**Networking** > **VPC networks** > **Firewall** > **Create Firewall Rule**). |
| 3 | GCP | Deploy the Threat Defense Virtual on GCP: Search for “Cisco Secure Firewall” in the GCP Marketplace. |

| | Workspace | Steps |
|---|---|---|
| 4 | GCP | Deploy the Threat Defense Virtual on GCP: Configure the threat defense virtual deployment parameters. |
| 5 | GCP | Deploy the Threat Defense Virtual on GCP: Configure network interfaces and apply firewall rules. |
| 6 | GCP | Deploy the Threat Defense Virtual on GCP: Deploy the Threat Defense Virtual on GCP. |
| 7 | Management Center or Device Manager | Manage the Firewall Threat Defense Virtual:<br>• Managing the Firewall Threat Defense Virtual with the Firewall Management Center<br>• Managing the Firewall Threat Defense Virtual with the Firewall Device Manager |

# Prerequisites

- Create a GCP account at https://cloud.google.com.
- Create your GCP project. See the Google documentation, Creating Your Project.
- A Cisco Smart Account. You can create one at Cisco Software Central (https://software.cisco.com/).
- License the Firewall Threat Defense Virtual.
  - Configure all license entitlements for the security services from the Firewall Management Center.
  - See the *Licensing* chapter in the Cisco Secure Firewall Management Center Administration Guide for more information about how to manage licenses.
- For Firewall Threat Defense Virtual system requirements, see Cisco Secure Firewall Threat Defense Compatibility Guide.

### Interface requirements

- Management interfaces (2) — One used to connect the Firewall Threat Defense Virtual to the Firewall Management Center, second used for diagnostics; cannot be used for through traffic.
- Traffic interfaces (2) — Used to connect the Firewall Threat Defense Virtual to inside hosts and to the public network.
- From Secure Firewall version 7.4.1, you can remove the diagnostic interface and deploy the Threat Defense Virtual on GCP with a minimum of 4 interfaces in the following combination – 1 management, and 3 data interfaces. We recommend that you deploy the Threat Defense Virtual on GCP without the diagnostic interface from Secure Firewall version 7.4.1. For more information, see About Deployment of Threat Defense Virtual without Diagnostic Interface on GCP, on page 321.

### Communications paths

- Public IPs for access into the Firewall Threat Defense Virtual.

# Guidelines and Limitations for the Firewall Threat Defense Virtual and GCP

### Supported Features

- Deployment in the GCP Compute Engine
- Maximum of 16 vCPUs per instance
- Routed mode (default)
- Licensing – Only BYOL is supported
- Clustering (7.2 or later). For more information, see Clustering for Threat Defense Virtual in a Public Cloud
- On Secure Firewall 7.1 and earlier versions, only Firewall Management Center is supported. Starting from Secure Firewall version 7.2, Firewall Device Manager is also supported.

### Performance Tiers for Firewall Threat Defense Virtual Smart Licensing

The Firewall Threat Defense Virtual supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.

*Table 29: Firewall Threat Defense Virtual Licensed Feature Limits Based on Entitlement*

| Performance Tier | Device Specifications (Core/RAM) | Rate Limit | RA VPN Session Limit |
|---|---|---|---|
| FTDv5, 100Mbps | 4 core/8 GB | 100Mbps | 50 |
| FTDv10, 1Gbps | 4 core/8 GB | 1Gbps | 250 |
| FTDv20, 3Gbps | 4 core/8 GB | 3Gbps | 250 |
| FTDv30, 5Gbps | 8 core/16 GB | 5Gbps | 250 |
| FTDv50, 10Gbps | 12 core/24 GB | 10Gbps | 750 |
| FTDv100, 16Gbps | 16 core/32 GB | 16Gbps | 10,000 |

See the "Licensing " chapter in the Cisco Secure Firewall Management Center Administration Guide for guidelines when licensing your Firewall Threat Defense Virtual device.

**Note** To change the vCPU/memory values, you must first power off the Firewall Threat Defense Virtual device.

### Performance Optimizations

To achieve the best performance out of the Firewall Threat Defense Virtual, you can make adjustments to the both the VM and the host. See Virtualization Tuning and Optimization on GCP for more information.

**Receive Side Scaling**—The Firewall Threat Defense Virtual supports Receive Side Scaling (RSS), which is a technology utilized by network adapters to distribute network receive traffic to multiple processor cores. Supported on Version 7.0 and later. See Multiple RX Queues for Receive Side Scaling (RSS) for more information.

### Allocation of Receive and Transmit Queues

A specific number of receive (RX) and transmit (TX) queues is assigned to each vNIC to process network packets. Based on the type of network interface used - VirtIO or gVNIC, Google Cloud uses an algorithm to assign a default number of RX and TX queues per vNIC.

The method used by GCP to assign queues to vNICs is as follow:

- VirtIO - Number of vCPUs divided by the number of vNICs, and discard any remainder value.

  For example, if the VM has 16 vCPUs and 4 vNICs, the number of queues assigned per vNIC is [16/4] = 4.

- gVNIC - Number of vCPUs divided by number of vNICs, and further divide the result by 2

  For example, if the VM has 128 vCPUs and 2 vNICs, the number of queues assigned is [128/2]/2 = 32.

You can also customize the number of queues that are allocated to each vNIC when you create a new VM by using the Compute Engine API. However, you have to adhere to the following rules if you want to do this-

- Minimum queue count: One per vNIC.
- Maximum queue count: This number is the lower of the vCPU count or the maximum queue count per vNIC, based on the driver type:
  - Maximum queue count is 32 if you are using VirtIO or a custom driver
  - Maximum queue count is 16 if you are using gVNIC
- If you customize the number of queues that is assigned to all the vNICs of the VM, the total number of queue assignments must be less than or equal to the number of vCPUs assigned to the VM instance.

For more information and examples on default and custom queue allocation, see Default queue allocation and Custom queue allocation.

### Snort

- If you are observing abnormal behavior such as Snort taking a long time to shut down, or the VM being slow in general or when a certain process is executed, collect logs from the Firewall Threat Defense Virtual and the VM host. Collection of overall CPU usage, memory, I/O usage, and read/write speed logs will help troubleshoot the issues.
- High CPU and I/O usage is observed when Snort is shutting down. If a number of Firewall Threat Defense Virtual instances have been created on a single host with insufficient memory and no dedicated CPU, Snort will take a long time to shut down which will result in the creation of Snort cores.

### Upgrade

Upgrade of Firewall Threat Defense Virtual in GCP from Secure Firewall version 7.1 to 7.2 is not supported. Perform a reimage if you are upgrading from Secure Firewall version 7.1 to 7.2.

### Unsupported Features

- IPv6
- Firewall Threat Defense Virtual native HA
- Transparent/inline/passive modes
- Jumbo Frames

# NIC Mapping to Data Interfaces

On Secure Firewall version 7.1 and earlier releases, the mapping of Network Interface Cards (NICs) to data interfaces is as given below:

- nic0 – Management interface
- nic1 – Diagnostic interface
- nic2 – Gigabit Ethernet 0/0
- nic3 – Gigabit Ethernet 0/1

From Secure Firewall version 7.2, a data interface is required on nic0 to facilitate movement of north-south traffic because the external load balancer (ELB) forwards packets only to nic0.

The mapping of NICs and data interfaces on Secure Firewall version 7.2 is as given below:

- nic0 – Gigabit Ethernet 0/0
- nic1 – Gigabit Ethernet 0/1
- nic2 – Management interface
- nic3 – Diagnostic interface
- nic4 – Gigabit Ethernet 0/2

  .

  .

  .
- nic(N-2) – Gigabit Ethernet 0/N-4
- nic(N-1) – Gigabit Ethernet 0/N-3

From Secure Firewall version 7.4.1, you can also deploy Threat Defense Virtual without the diagnostic interface. In such a scenario, the mapping of NICs and data interfaces is as given below:

- nic0 – Gigabit Ethernet 0/0

- nic1 – Gigabit Ethernet 0/1
- nic2 – Management interface
- nic3 – Gigabit Ethernet 0/2
- nic4 – Gigabit Ethernet 0/3

.

.

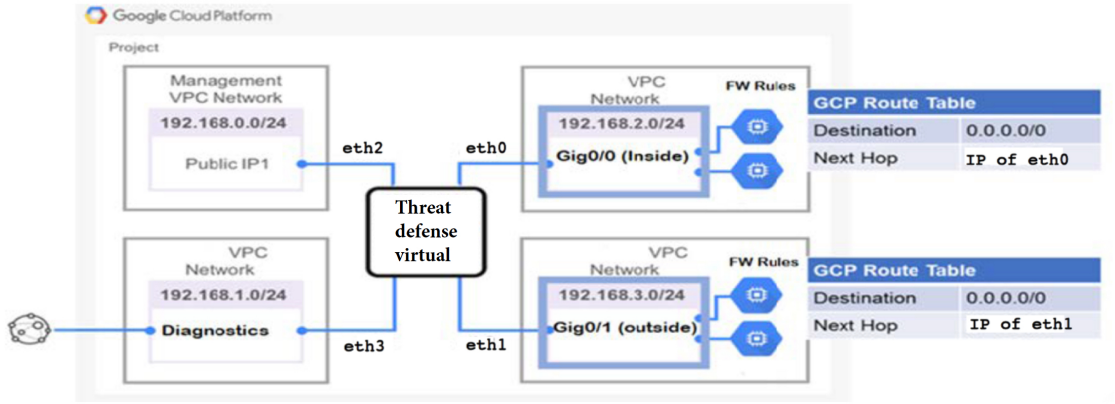
.

- nic(N-2) – Gigabit Ethernet 0/N-3
- nic(N-1) – Gigabit Ethernet 0/N-2

# Sample Network Topology

The following figure shows the recommended topology for the Firewall Threat Defense Virtual in Routed Firewall Mode with 4 subnets configured in GCP for the Firewall Threat Defense Virtual (management, diagnostic, inside, and outside).

***Figure 45: Sample Firewall Threat Defense Virtual on GCP Deployment***




# How to Manage Secure Firewall Threat Defense Virtual Device

You have two options to manage your Secure Firewall Threat Defense Virtual.

## Secure Firewall Management Center

If you are managing large numbers of devices, or if you want to use the more complex features and configurations that the Firewall Threat Defense allows, use the Firewall Management Center to configure your devices instead of the integrated Firewall Device Manager.

**Important** You cannot use both the Firewall Device Manager and the Firewall Management Center to manage the Firewall Threat Defense device. Once the Firewall Device Manager integrated management is enabled, it won't be possible to use the Firewall Management Center to manage the Firewall Threat Defense device, unless you disable the local management and re-configure the management to use the Firewall Management Center. On the other hand, when you register the Firewall Threat Defense device to the Firewall Management Center, the Firewall Device Manager onboard management service is disabled.

**Caution** Currently, Cisco does not have an option to migrate your Firewall Device Manager configuration to the Firewall Management Center and vice-versa. Take this into consideration when you choose what type of management you configure for the Firewall Threat Defense device.

## Secure Firewall Device Manager

The Firewall Device Manager is a web interface included on most Firewall Threat Defense devices. It lets you configure the basic features of the software that are most commonly used for small networks. It is especially designed for networks that include a single device or just a few, where you do not want to use a high-powered multiple-device manager to control a large network with many devices.

**Note** See the Cisco Secure Firewall Threat Defense Compatibility Guide for list of devices that support the Firewall Device Manager.

# Configure GCP Environment

The Firewall Threat Defense Virtual deployment requires four networks which you must create prior to deploying the Firewall Threat Defense Virtual. The networks are as follows:

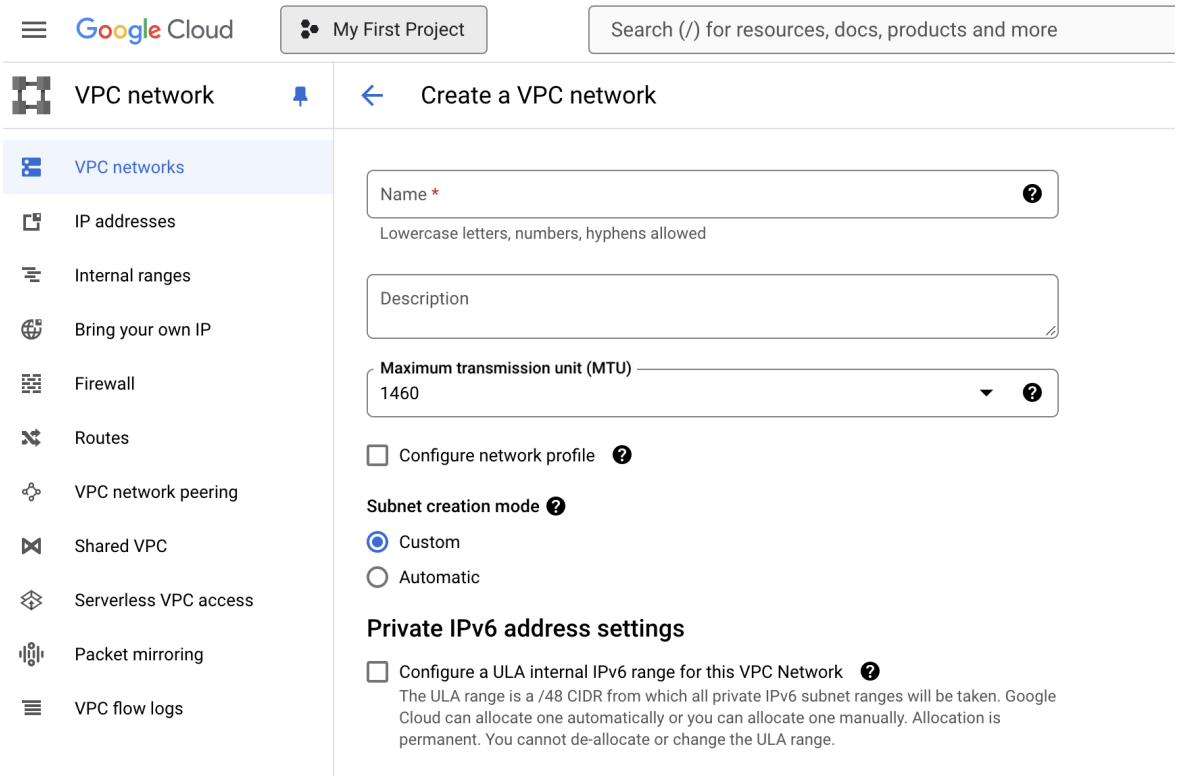
- Management VPC for the management subnet.
- Diagnostic VPC or the diagnostic subnet.
- Inside VPC for the inside subnet.
- Outside VPC for the outside subnet.

Additionally, you set up the route tables and GCP firewall rules to allow traffic flow through the Firewall Threat Defense Virtual. The route tables and firewall rules are separate from those that are configured on the Firewall Threat Defense Virtual itself. Name the GCP route tables and firewall rules according to associated network and functionality. See Sample Network Topology for FTDv on GCP as a guide.

You can set up a route for GCP health checks across all interfaces that are used to configure their health probes. You can achieve this by creating a route with a higher metric on interfaces where a route for GCP health checks is not already available.

## Procedure

**Step 1** In the GCP console, choose **VPC networks**, then click **Create VPC Network**.

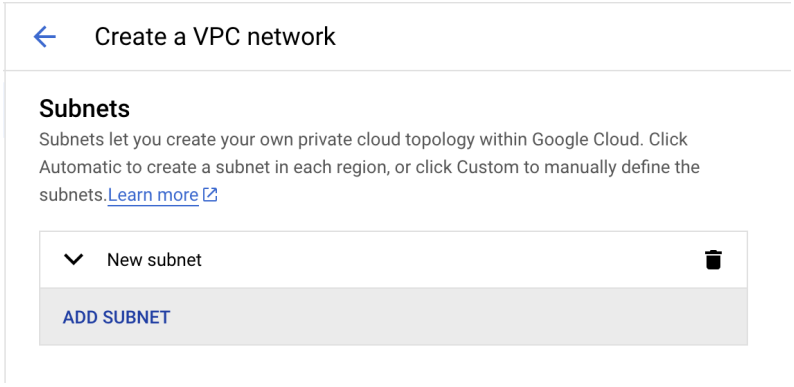



**Step 2** In the **Name** field, enter the desired name.

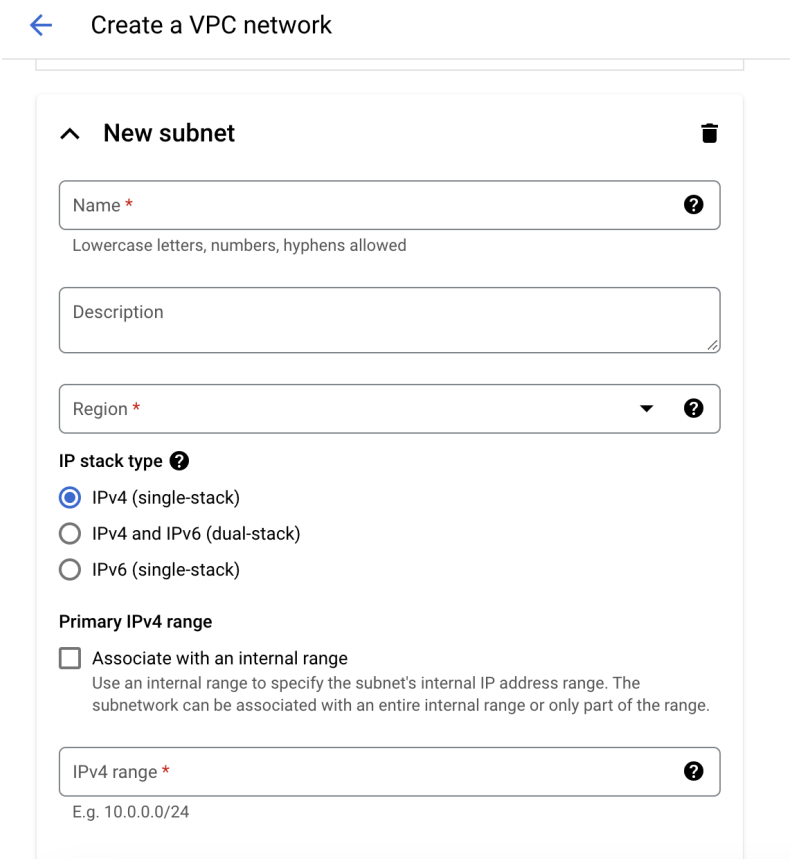
**Step 3** From the **Maximum transmission unit (MTU)** drop-down menu, choose an appropriate MTU value.

**Step 4** From the **Subnet creation mode** option, click **Custom**.

**Step 5** Under the **Subnet** section, click **ADD SUBNET** to create a new subnet.




**Step 6** In the **Name** field under **New subnet**, enter the desired name.




**Step 7** From the **Region** drop-down list, select the region appropriate for your deployment. All four networks must be in the same region.

**Step 8** From the **IP address range** field, enter the first network's subnet in CIDR format, such as 10.10.0.0/24.

**Step 9** Accept the defaults for all other settings, then click **Create**.

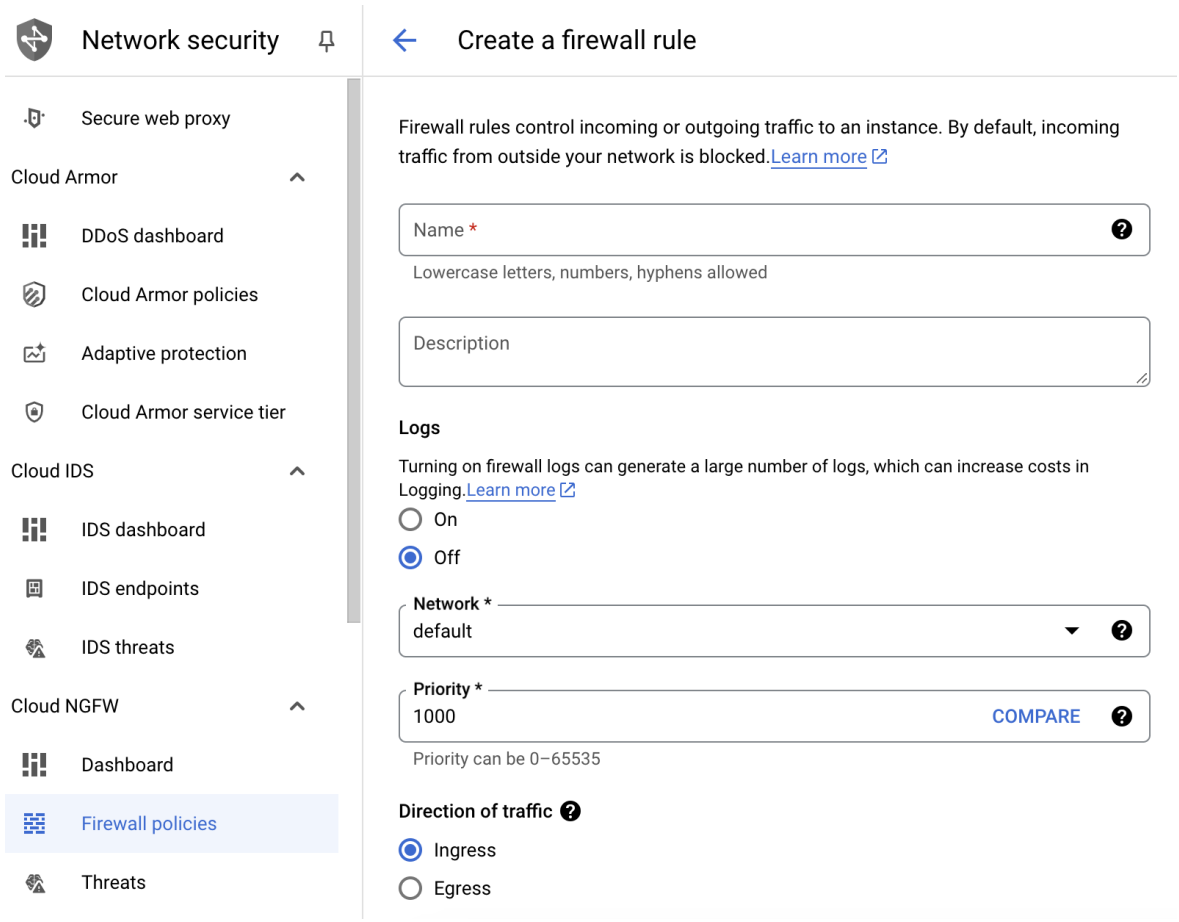
**Step 10** Repeat steps 1-7 to create the remaining three VPC networks.

# Create the Firewall Rules

You apply the firewall rules for the management interface (to allow SSH and SFTunnel communication with the Firewall Management Center) while deploying the Firewall Threat Defense Virtual instance, see Deploy the Firewall Threat Defense Virtual, on page 315. According to your requirements, you can also create firewall rules for the inside, outside, and diagnostic interfaces.

## Procedure

**Step 1** In the GCP console, choose **Networking** > **VPC network** > **Firewall**, then click **Create Firewall Rule**.

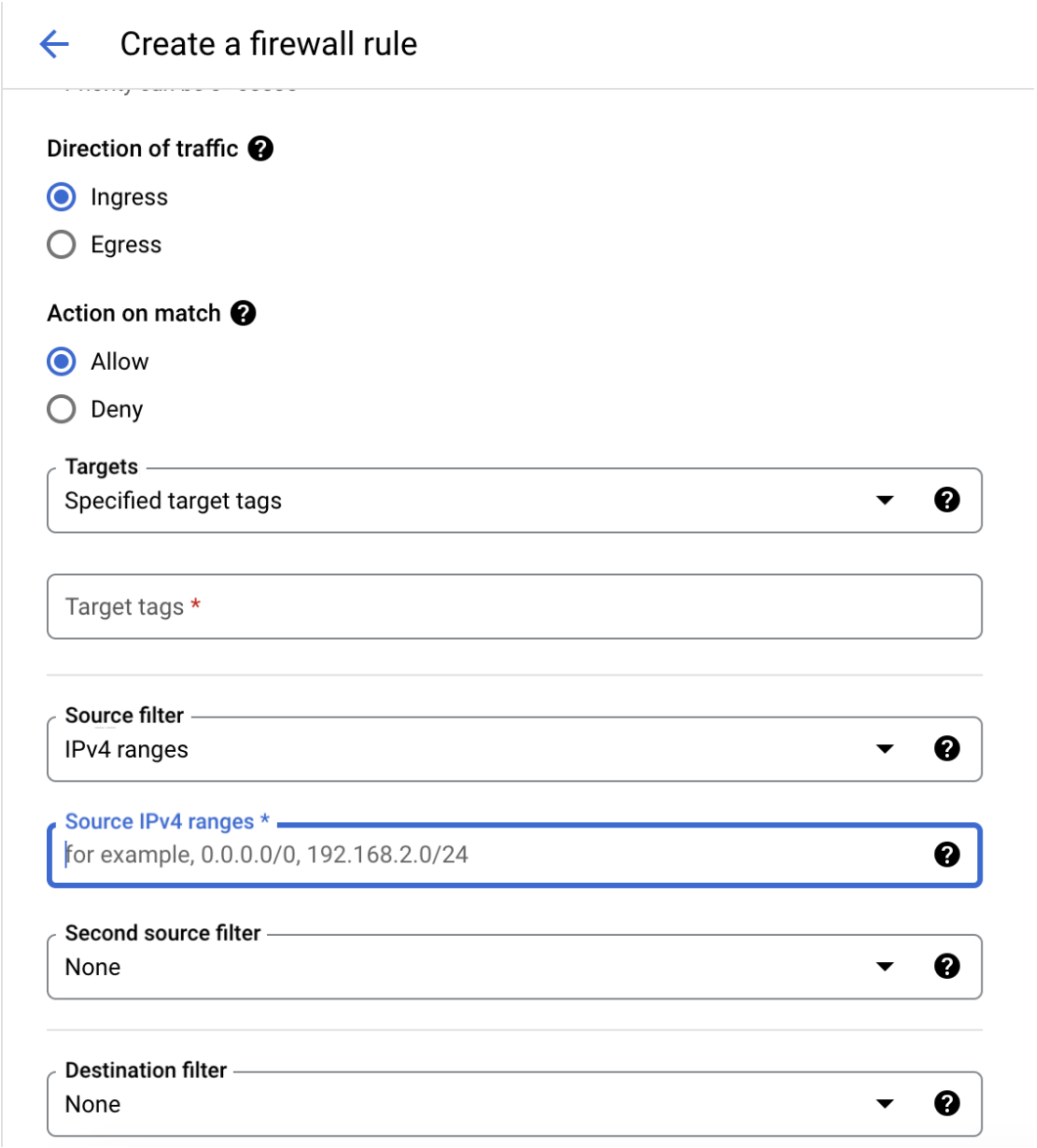



**Step 2** In the **Name** field, enter a descriptive name for your firewall rule, for example, *vpc-asiasouth-inside-fwrule*.

**Step 3** From the **Network** drop-down list, select the name of the VPC network for which you are creating the firewall rule, for example, *ftdv-south-inside*.

**Step 4** From the **Targets** drop-down list, select the option applicable for your firewall rule, for example, **All instances in the network**.

**Step 5** From the **Source filter** drop-down list, select the supported IP type, for example, **IPv4 ranges**.




**Step 6** In the **Source IP ranges** field, enter the source IP address ranges in CIDR format, for example, 0.0.0.0/0.

Traffic is only allowed from sources within these IP address ranges.

**Step 7** Under **Protocols and ports**, select **Specified protocols and ports**.

**Step 8** Add your security rules.

**Step 9** Click **Create**.

# Deploy the Firewall Threat Defense Virtual

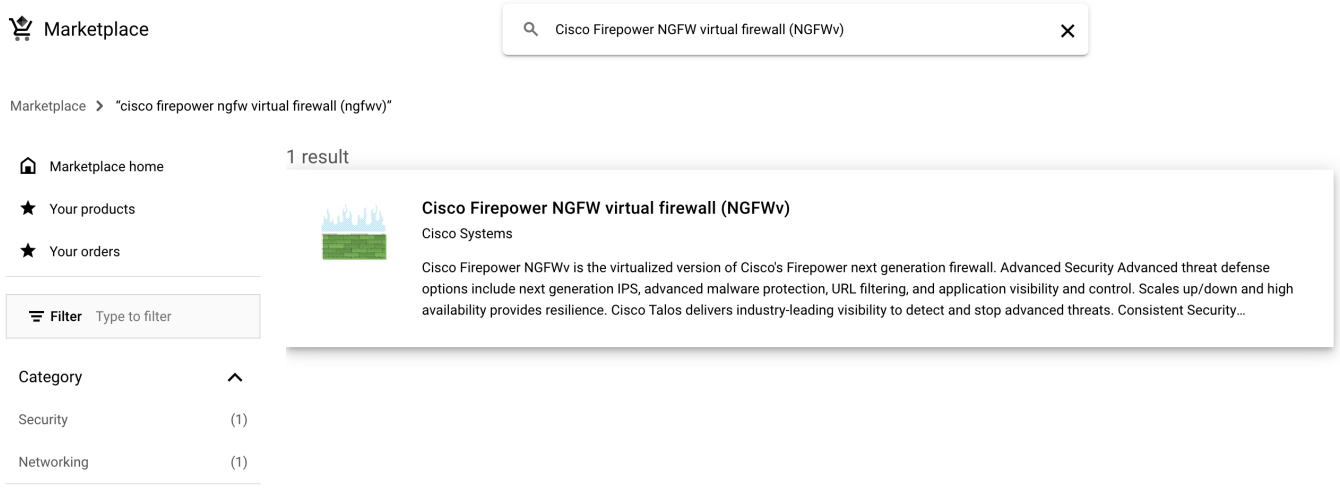
You can follow the steps below to deploy an Firewall Threat Defense Virtual instance using the Cisco Firepower NGFW virtual firewall (NGFWv) offering from the GCP Marketplace.

## Procedure

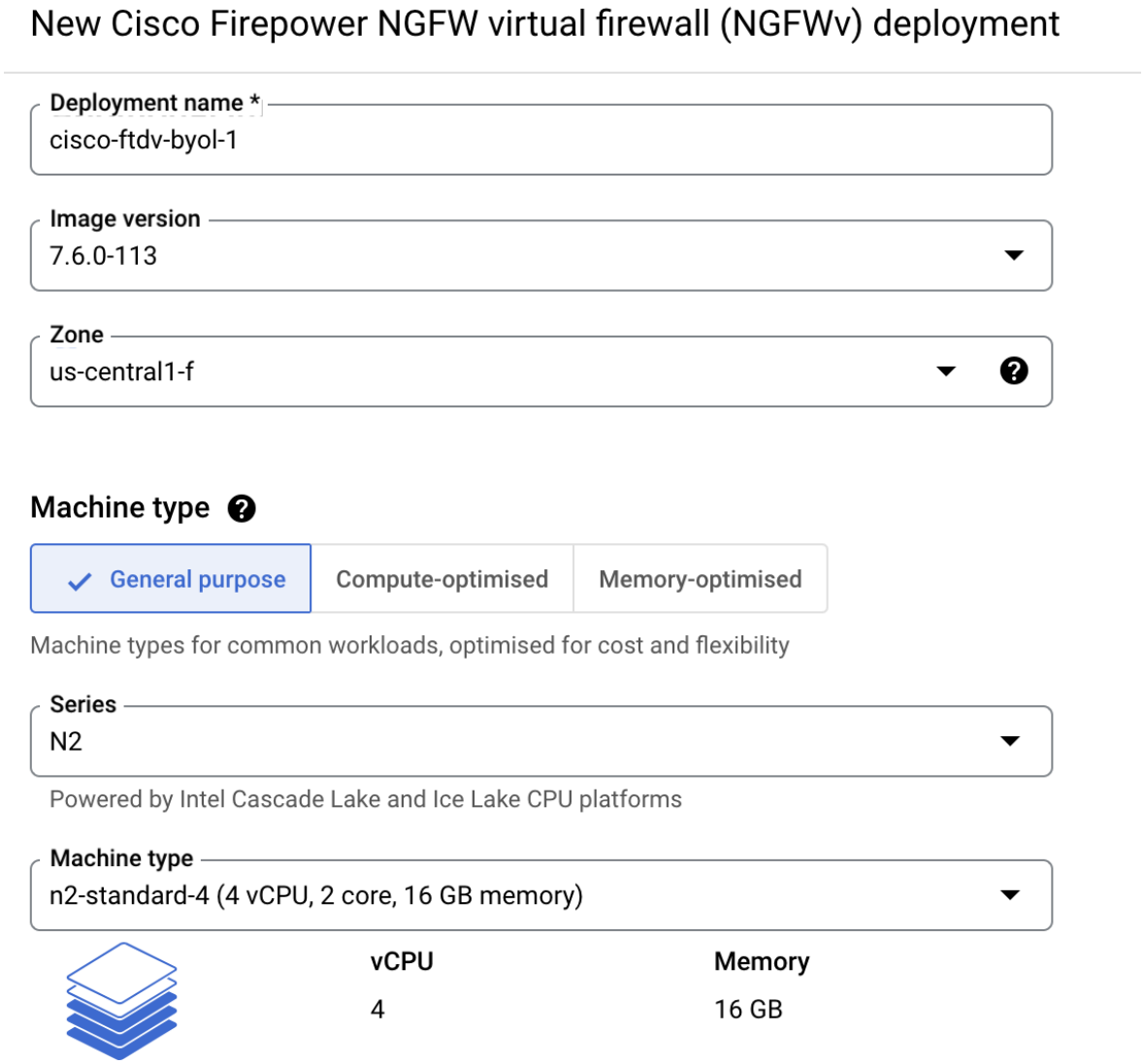
**Step 1** Log into to the GCP Console.

**Step 2** Click **Navigation menu** > **Marketplace**.

**Step 3** Search the Marketplace for "Cisco Firepower NGFW virtual firewall (NGFWv)" and choose the offering.




**Step 4** Click **Launch**.




a) **Deployment name** — Specify a unique name for the instance.
b) **Zone** — Select the zone where you want to deploy the Firewall Threat Defense Virtual.
c) **Machine type** — Choose the correct machine type based on the System Requirements, on page 302.
d) **SSH key (optional)** — Paste the public key from the SSH key pair.

   The key pair consists of a public key that GCP stores and a private key file that the user stores. Together they allow you to connect to your instance securely. Be sure to save the key pair to a known location, as it will be required to connect to the instance.

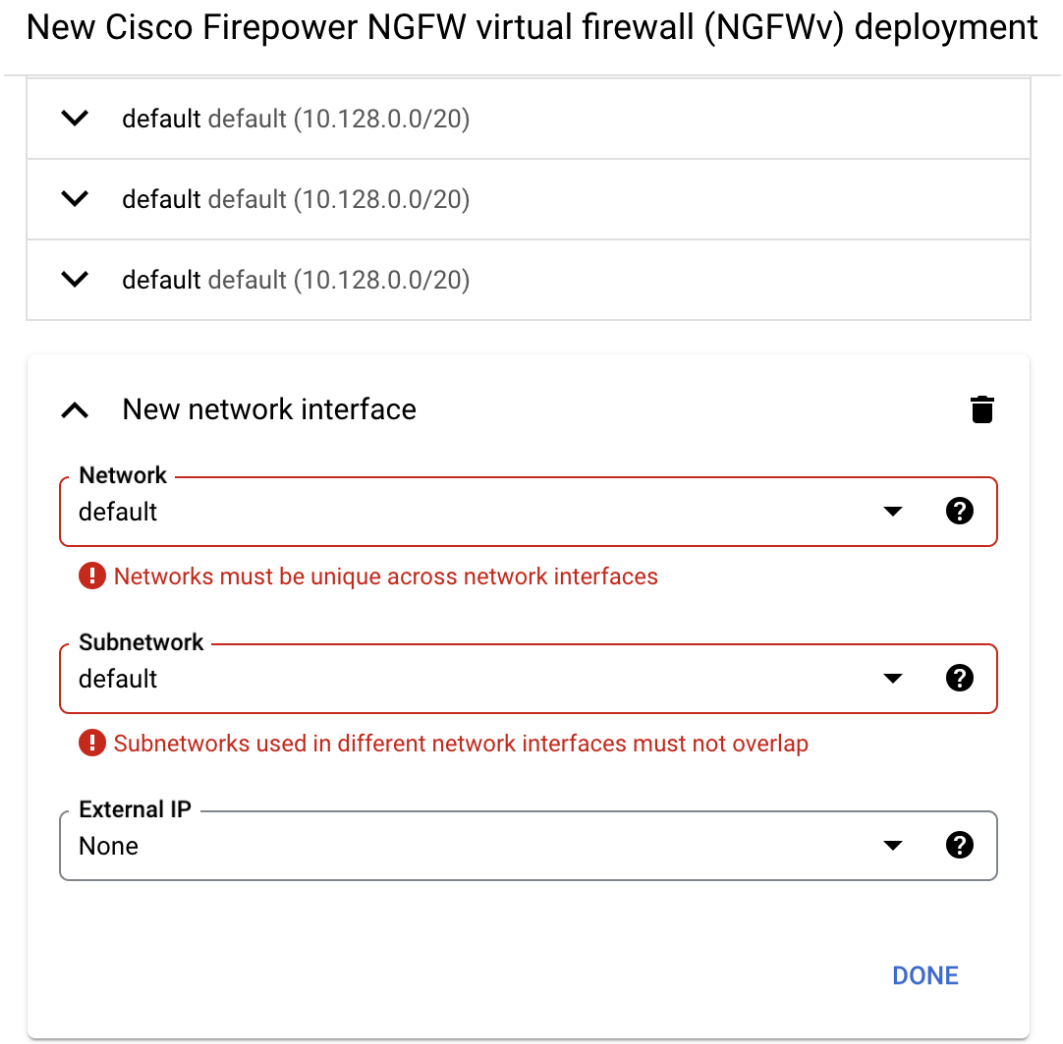
e) Choose whether to allow or block the project-wide SSH keys to access this instance. See the Google documentation Allowing or blocking project-wide public SSH keys from a Linux instance.
f) **Startup script** — You can create a startup script for your Firewall Threat Defense Virtual instance to perform automated tasks every time your instance boots up.

   The following example shows a sample Day0 configuration you copy and paste in the **Startup script** field:

```
{
"AdminPassword": "Cisco@123123",
"Hostname": "ftdv-gcp",
"DNS1": "8.8.8.8",
"FirewallMode": "routed",
"IPv4Mode": "dhcp",
"ManageLocally": "No"
}
```

**Tip**
To prevent execution errors, you should validate your Day0 configuration using a JSON validator.

g) **Network interfaces** — Configure interfaces: 1) management, 2) diagnostic, 3) inside, 4) outside.



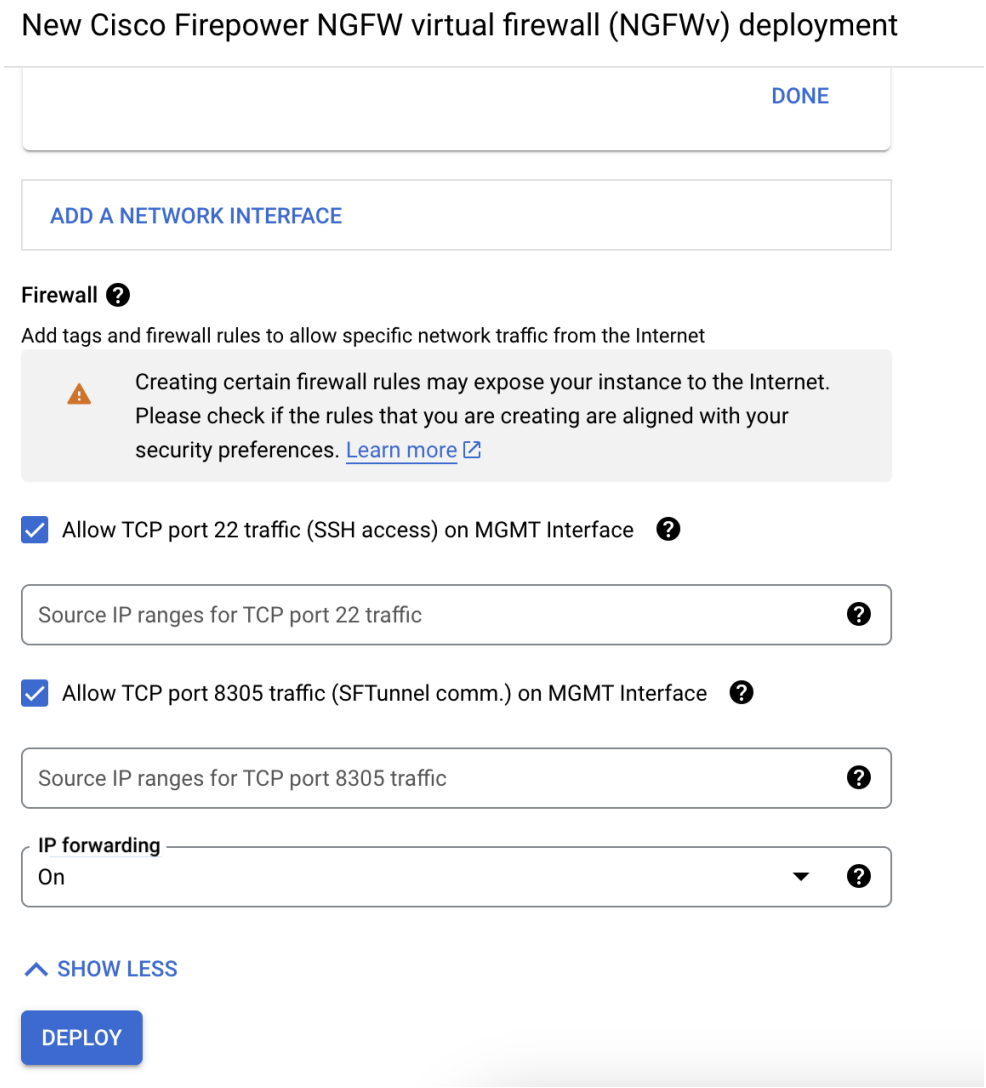

**Note**
You cannot add interfaces to an instance after you create it. If you create the instance with an improper interface configuration, you must delete the instance and recreate it with the proper interface configuration.

1. From the **Network** drop-down list, select a VPC network, for example, *vpc-asiasouth-mgmt*.

2. From the **Subnetwork** drop-down list, select a subnet.
3. From the **External IP** drop-down list, select the appropriate option.

   For the management interface, select the **External IP** to **Ephemeral**. This is optional for inside and outside interfaces.
4. Click **Done**.

h) **Firewall**— Apply the firewall rules.




- Check the **Allow TCP port 22 traffic from the Internet (SSH access)** check box to allow SSH.
- Check the **Allow HTTPS traffic from the Internet (FMC access)** check box to allow the Firewall Management Center and managed devices to communicate using a two-way, SSL-encrypted communication channel (SFTunnel ).

i) Click **More** to expand the view and make sure that **IP Forwarding** is set to **On**.

**Step 5** Click **Deploy**.

**Note**
Startup time depends on a number of factors, including resource availability. It can take between 7-8 minutes for the initialization to complete. Do not interrupt the initialization or you may have to delete the appliance and start over.

### What to do next

View the instance details from the VM instance page of the GCP console. You'll find the internal IP address, external IP address, and controls to stop and start the instance. You need to stop the instance if you need to edit it.

# Connect to the Firewall Threat Defense Virtual Instance Using an External IP

The Firewall Threat Defense Virtual instance is assigned with an internal IP and an external IP. You can use the external IP to access the Firewall Threat Defense Virtual instance.

### Procedure

**Step 1** In the GCP console, choose **Compute Engine** > **VM instances**.

**Step 2** Click the Firewall Threat Defense Virtual instance name to open the **VM instance details** page.

**Step 3** Under the **Details** tab, click the drop-down menu for the **SSH** field.

**Step 4** Select the desired option from the **SSH** drop-down menu.

You can connect to the Firewall Threat Defense Virtual instance using the following method.

- Any other SSH client or third-party tools—See the Google documentation, Connecting using third-party tools for more information.

# Connect to the Firewall Threat Defense Virtual Instance Using SSH

To connect to the Firewall Threat Defense Virtual instance from a Unix-style system, log in to the instance using SSH.

### Procedure

**Step 1** Use the following command to set the file permissions so that only you can read the file:

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

Where:

<private_key> is the full path and name of the file that contains the private key associated with the instance you want to access.

**Step 2** Use the following SSH command to access the instance:

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

<private_key> is the full path and name of the file that contains the private key associated with the instance you want to access.

<username> is the username for the Firewall Threat Defense Virtual instance.

<public-ip-address> is your instance IP address that you retrieved from the console.

# Connect to the Firewall Threat Defense Virtual Instance Using the Serial Console

**Procedure**

**Step 1** In the GCP console, choose **Compute Engine** > **VM instances**.

**Step 2** Click the Firewall Threat Defense Virtual instance name to open the **VM instance details** page.

**Step 3** Under the **Details** tab, click **Connect to serial console**.

See the Google documentation, Interacting with the serial console for more information.

# Connect to the Firewall Threat Defense Virtual Instance Using Gcloud

**Procedure**

**Step 1** In the GCP console, choose **Compute Engine** > **VM instances**.

**Step 2** Click the Firewall Threat Defense Virtual instance name to open the **VM instance details** page.

**Step 3** Under the **Details** tab, click the drop-down menu for the **SSH** field.

**Step 4** Click **View gcloud command** > **Run in Cloud Shell**.

The Cloud Shell terminal window opens. See the Google documentation, gcloud command-line tool overview, and gcloud compute ssh for more information.

# About Deployment of Threat Defense Virtual without Diagnostic Interface on GCP

On Secure Firewall version 7.3 and earlier, the Firewall Threat Defense Virtual is deployed with a minimum of 4 interfaces – 1 management, 1 diagnostic, and 2 data interfaces.

From Secure Firewall version 7.4.1, you can remove the diagnostic interface and deploy the Firewall Threat Defense Virtual with a minimum of 4 interfaces – 1 management, and 3 data interfaces. This feature enables deployment of the Threat Defense Virtual with an additional data interface on the same machine type. For example, on a c2-standard-8 machine type, instead of deploying Threat Defense Virtual with 1 management, 1 diagnostic, and 6 data interfaces, you can now deploy Threat Defense Virtual with 1 management, and 7 data interfaces.

From Secure Firewall version 7.4.1, we recommend that you deploy the Threat Defense Virtual on GCP without the diagnostic interface.

This feature is supported only on new deployments of Firewall Threat Defense Virtual instances on Google Cloud Platform (GCP).



**Note** As the maximum number of supported interfaces is 8, you can add up to 4 more interfaces to deploy the Firewall Threat Defense Virtual with the maximum of 8 interfaces.

# Guidelines and Limitations for Deployment of Threat Defense Virtual without Diagnostic Interface

- When the diagnostic interface is removed, syslog and SNMP is supported using either the Firewall Threat Defense Virtual management or the data interface instead of the diagnostic interface.
- Clustering and auto scale is supported with this deployment.
- Grouping of Firewall Threat Defense Virtual instances with diagnostic interface port and Firewall Threat Defense Virtual instances without diagnostic interface port is not supported.

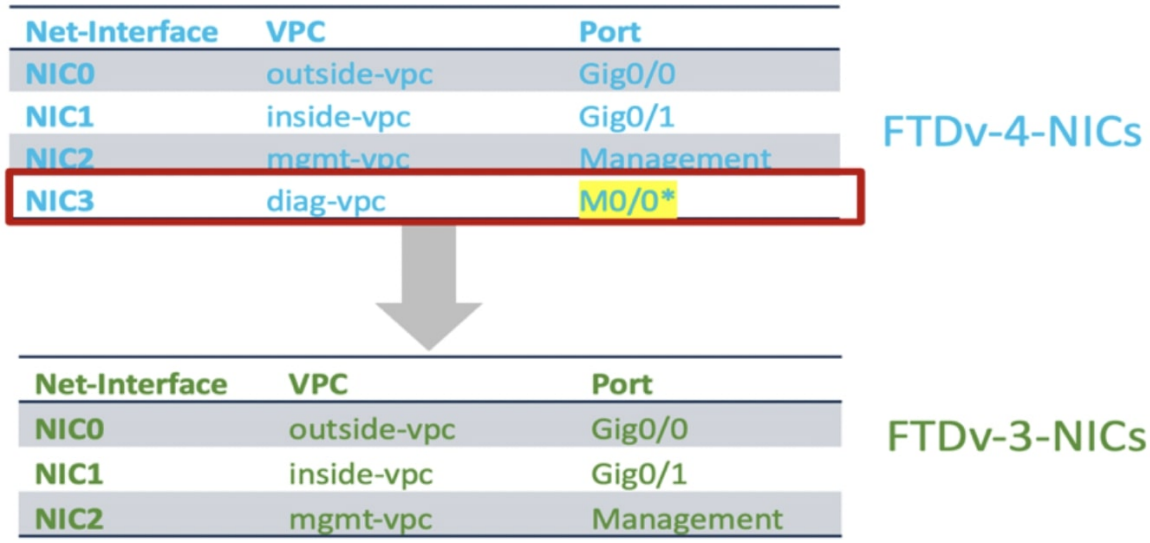


**Note** The grouping of Firewall Threat Defense Virtual instances here refers to the grouping of the instances in the instance group on GCP. This does not pertain to the grouping of Firewall Threat Defense Virtual instances on the Management Center Virtual.

- CMI is not supported.

# NIC Mapping to Data Interfaces for Deployment of Threat Defense Virtual without Diagnostic Interface on GCP

The NIC mapping to data interfaces for deployment of Threat Defense Virtual without the diagnostic interface is given below.




# Deploy Threat Defense Virtual without Diagnostic Interface on GCP

Perform the steps given below to deploy Firewall Threat Defense Virtual without the diagnostic interface.

**Procedure**

**Step 1** Enable this feature by using a key-value pair, **Diagnostic: OFF/ON**, in the day-0 configuration script (**Startup script** on the GCP console) that is used for fresh deployment. By default, the key-value pair is set to **Diagnostic: ON** and the diagnostic interface comes up. When the key-value pair is set to **Diagnostic: OFF**, the deployment comes up without the diagnostic interfaces.

A sample day-0 configuration script is given below.

```
{
"AdminPassword": "E28@2OiUrhx!",
"Hostname": "ciscothreatdefensevirtual",
"FirewallMode": "routed",
"ManageLocally": "No",
"Diagnostic": "OFF"
}
```

**Note**

The key value pair, "Diagnostic": "ON/OFF", is case-sensitive.

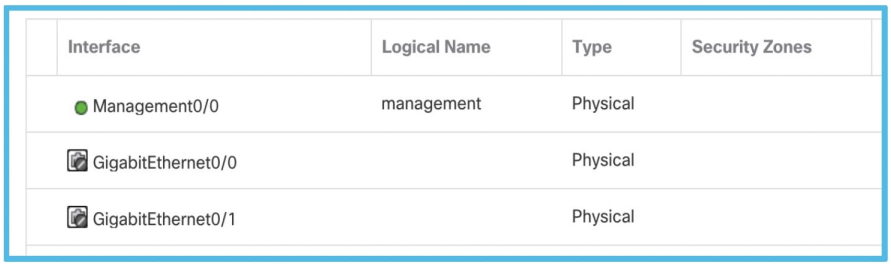
**Step 2** Attach the required minimum number of NICs - 4.

See Deploy Threat Defense Virtual on GCP for the detailed procedure to deploy the Firewall Threat Defense Virtual on GCP.

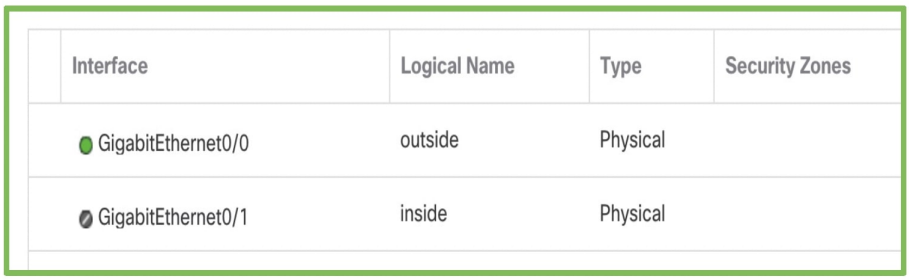
For more information on interfaces, see Interface Overview.

**Step 3** (Optional) Use the **show interface ip brief** command on the console to display interface details. You can also view interface details on the Management Center Virtual as given below

The interfaces are displayed on the Management Center Virtual as given below.




With Diagnostic Interface




Without Diagnostic Interface

# Upgrade Scenarios

You can upgrade a Firewall Threat Defense Virtual instance as per the scenarios given below.

- All Secure firewall versions – You can upgrade a Firewall Threat Defense Virtual instance deployed with a diagnostic interface to a Firewall Threat Defense Virtual instance with a diagnostic interface.
- Secure Firewall version 7.4 and later – You can upgrade a Firewall Threat Defense Virtual instance deployed without a diagnostic interface to a Firewall Threat Defense Virtual instance without a diagnostic interface.

The upgrade scenarios given below are not supported.

- All Secure firewall versions – You cannot upgrade a Firewall Threat Defense Virtual instance deployed with a diagnostic interface to a Firewall Threat Defense Virtual instance without a diagnostic interface.

- Secure Firewall version 7.4.1 and later – You cannot upgrade a Firewall Threat Defense Virtual instance deployed without a diagnostic interface to a Firewall Threat Defense Virtual instance with a diagnostic interface.



**Note** The number and order of the NICs is maintained after upgrading.

# Deployment of Threat Defense Virtual Cluster or Auto Scale Solution without Diagnostic Interface

To perform a new deployment of a Firewall Threat Defense Virtual cluster or an auto scale solution consisting of Firewall Threat Defense Virtual instances without the diagnostic interface, ensure that the key-value pair, **Diagnostic: OFF/ON**, is set to **OFF** in the day-0 configuration script.

# Troubleshooting

If the diagnostic interface is not removed when the Firewall Threat Defense Virtual is deployed, check if the key-value pair, **Diagnostic: OFF/ON**, has been set to **OFF** in the day-0 configuration script.

# Auto Scale Solution

The following sections describe how the components of the Auto Scale solution work for the Firewall Threat Defense Virtual on GCP.

## Overview

Firewall Threat Defense Virtual Auto Scale for GCP is a complete serverless implementation that makes use of serverless infrastructure provided by GCP (Cloud Functions, Load Balancers, Pub/Sub, Instance Groups, etc.).

Some of the key features of the Firewall Threat Defense Virtual Auto Scale for GCP implementation include:

- GCP Deployment Manager template-based deployment.
- Support for scaling metrics based on CPU utilization..
- Support for Firewall Threat Defense Virtual deployment and multi-availability zones.
- Support for automatic registration and de-registration of Firewall Threat Defense Virtual.
- Completely automated configuration automatically applied to scaled-out Firewall Threat Defense Virtual instances.
- Support for automatic application of NAT policy, access policy, and routes, to Firewall Threat Defense Virtual.
- Support for Load Balancers and multi-availability zones.

- Support for Firewall Management Center Virtual on other platforms.
- Cisco provides an Auto Scale for GCP deployment package to facilitate the deployment.

# Guidelines and Limitations

- Only IPv4 is supported.
- Licensing - Only BYOL is supported. PAYG licensing is not supported.
- Device functionality errors are not displayed in the logs.
- The maximum number of devices supported is 25. This is the maximum limit in a Firewall Management Center Virtual instance.
- On all Secure Firewall versions, you can use the provided templates to deploy the Threat Defense Virtual auto scale solution. The Threat Defense Virtual instances are deployed with a minimum of 4 interfaces - 1 management, 1 diagnostic, and 2 data interfaces.

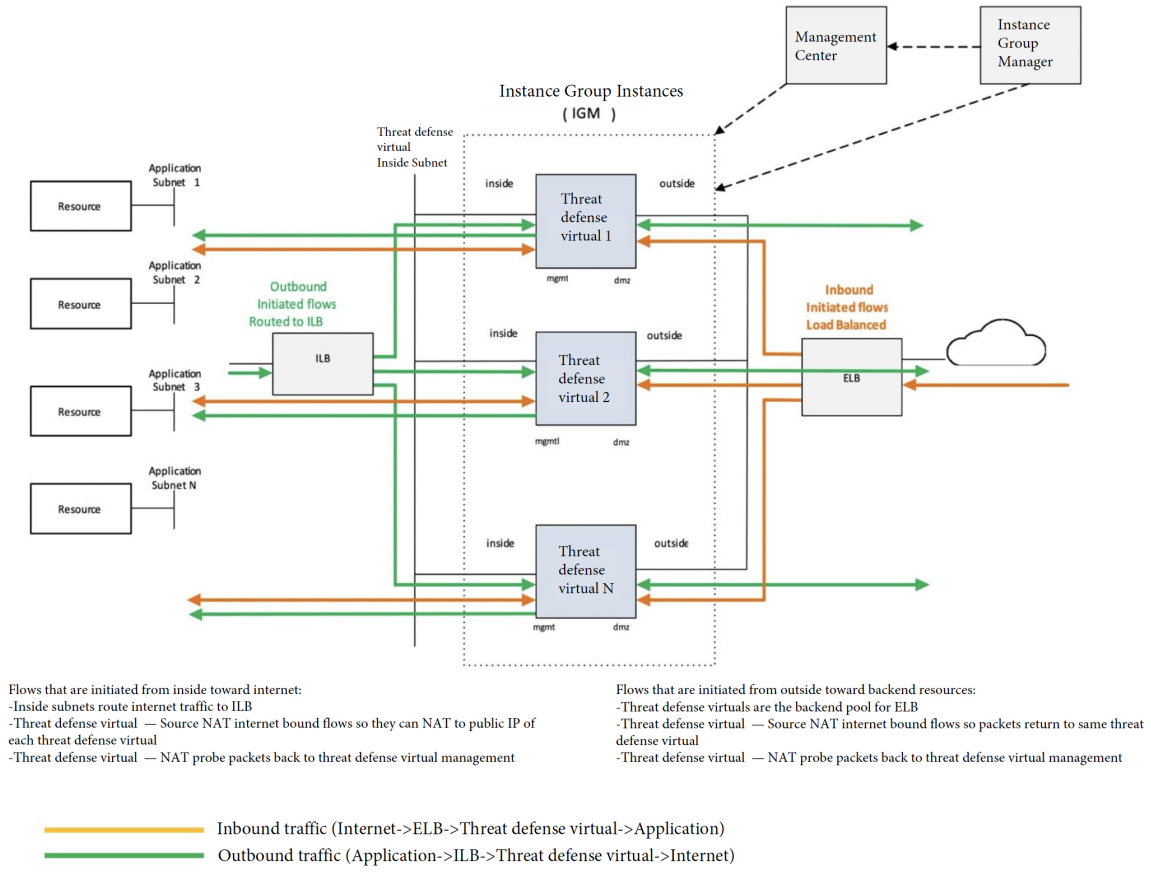
  From Secure Firewall version 7.4.1, you can also deploy the Threat Defense Virtual without the diagnostic interface. In this scenario also, the deployment is done with a minimum of 4 interfaces - 1 management, and 3 data interfaces. To do this, modify the template parameters - *diagFirewallRule*, *diagSubnetworkName*, *diagVpcName*, and *withDiagnostic*, as per the description given in Input Parameters, on page 331.
- Cold standby or Snapshot methods to reduce scale-out time are not supported.
- Schedule based scaling is not supported.
- Auto Scaling based on Average Memory Utilization is not supported.
- Scale-In/Scale-Out may decrease/increase the number of instances by more than 1. However, the Firewall Threat Defense Virtual instances will only deregister/register on the Firewall Management Center Virtual sequentially, that is, one by one.
- During scale-in, there is a connection draining time of 300s. You can also manually configure the draining time to a required period.
- The external Load Balancer is created by the template that is provided. Customizing DNS requirements of the Load Balancer's public IP is not supported.
- Users have to fit their existing infrastructure into the sandwich model of implementation.
- For details on errors faced during the scale-out and scale-in process, analyze the logs of the Cloud Functions.
- NAT, security policies attached to device group, and static routes, are applied to the newly created Firewall Threat Defense.
- If you are deploying the solution for more than 1 Firewall Threat Defense Virtual, then the deployment time will increase as the Firewall Management Center Virtual can handle only one registration request at a time. Deployment time also increases when scaling out adds more than one Firewall Threat Defense Virtual instance. Currently, all registrations and de-registrations are sequential.
- Device Group, NAT rules, and network objects, have to be created in Firewall Management Center Virtual before Auto Scaling is initiated. Note that the ILB and ELB IPs are only available after deploying the solution. So, you can create dummy objects and update the objects after the actual IPs are obtained.

# Auto Scale Use Case

The Firewall Threat Defense Virtual Auto Scale for GCP is an automated horizontal scaling solution that positions a Firewall Threat Defense Virtual instance group sandwiched between a GCP Internal load balancer (ILB) and a GCP External load balancer (ELB).

- The ELB distributes traffic from the Internet to Firewall Threat Defense Virtual instances in the instance group; the Firewall Threat Defense Virtual then forwards traffic to the application.
- The ILB distributes outbound Internet traffic from an application to Firewall Threat Defense Virtual instances in the instance group; the Firewall Threat Defense Virtual then forwards traffic to the Internet.
- A network packet will never pass through both (internal & external) load balancers in a single connection.
- The number of Firewall Threat Defense Virtual instances in the scale set will be scaled and configured automatically based on load conditions.

***Figure 46: Firewall Threat Defense Virtual Auto Scale Use Case***




On all Secure Firewall versions, you can use the provided templates to deploy the Threat Defense Virtual auto scale solution. The Threat Defense Virtual instances are deployed with a minimum of 4 interfaces - 1 management, 1 diagnostic, and 2 data interfaces.

From Secure Firewall version 7.4.1, you can also deploy the Threat Defense Virtual without the diagnostic interface. In this scenario also, the deployment is done with a minimum of 4 interfaces - 1 management, and

3 data interfaces. To do this, modify the template parameters - *diagFirewallRule*, *diagSubnetworkName*, *diagVpcName*, and *withDiagnostic*, as per the description given in Input Parameters, on page 331.

## Scope

This document covers the detailed procedures to deploy the serverless components for the Firewall Threat Defense Virtual Auto Scale for GCP solution.



**Important**

- Read the entire document before you begin your deployment.
- Make sure the prerequisites are met before you start deployment.
- Make sure you follow the steps and order of execution as described herein.

# Download the Deployment Package

The Firewall Threat Defense Virtual Auto Scale for GCP solution is a GCP Deployment Manager template-based deployment that makes use of the serverless infrastructure provided by GCP (Cloud Functions, Load Balancers, Pub/Sub, Instance Groups, etc.).

Download the files required to launch the Firewall Threat Defense Virtual auto scale for GCP solution. Deployment scripts and templates for your Firewall Threat Defense Virtual version are available in the GitHub repository.



**Attention** Note that Cisco-provided deployment scripts and templates for auto scale are provided as open source examples, and are not covered within the regular Cisco TAC support scope.

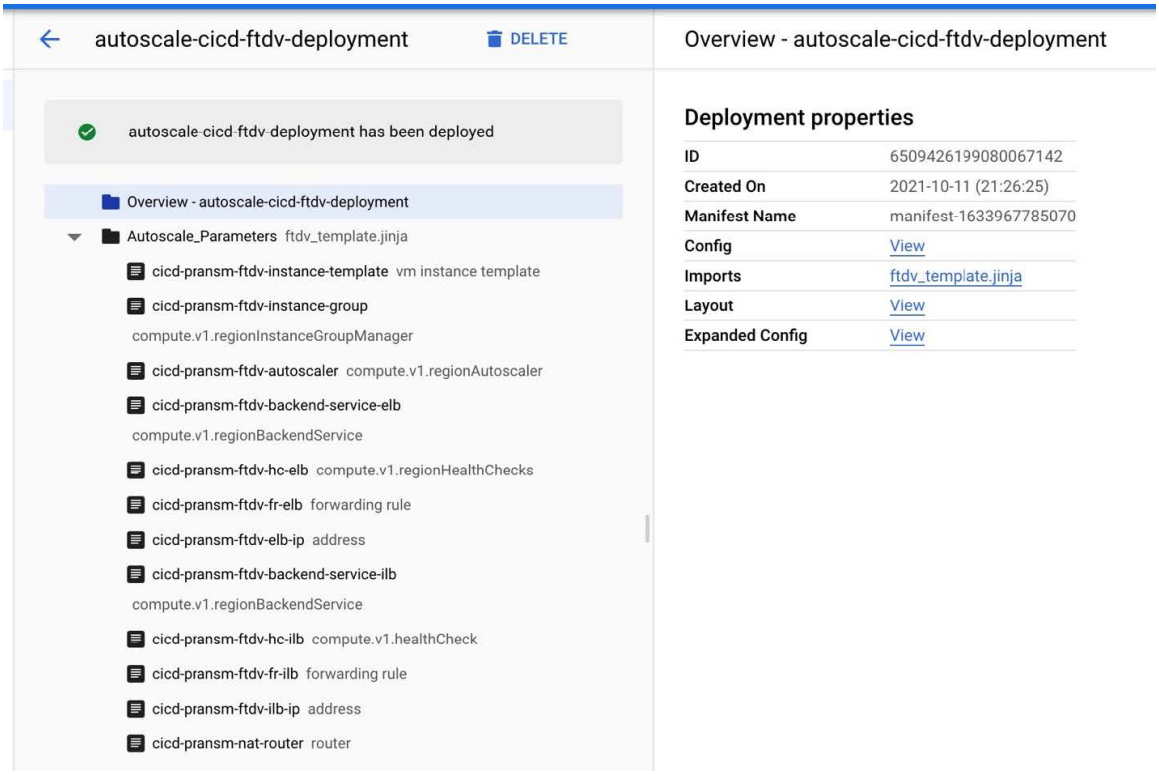
# System Requirements

The following components make up the Firewall Threat Defense Virtual Auto Scale for GCP solution.

### Deployment Manager

- Treat your configuration as code and perform repeatable deployments. Google Cloud Deployment Manager allows you to specify all the resources needed for your application in a declarative format using YAML. You can also use Jinja2 templates to parameterize the configuration and allow the reuse of common deployment paradigms.
- Create configuration files that define the resources. The process of creating those resources can be repeated over and over with consistent results. See https://cloud.google.com/deployment-manager/docs for more information.

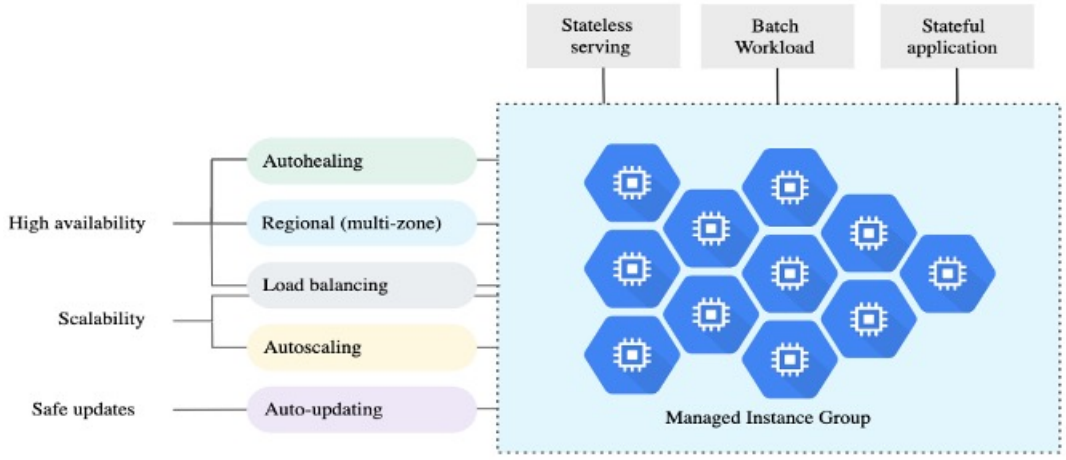
*Figure 47: Deployment Manager View*




## Managed Instance Group in GCP

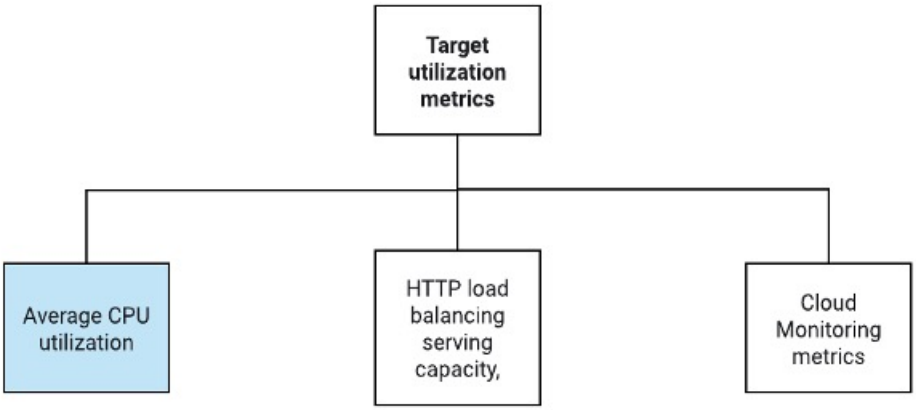
A Managed Instance Group (MIG) creates each of its managed instances based on the instance template and optional stateful configuration that you specify. See https://cloud.google.com/compute/docs/instance-groups for more information.

*Figure 48: Instance Group Features*



### Target Utilization Metrics

- The following diagram alongside shows the target utilization metrics. Only average CPU utilization metrics are used in making autoscaling decisions.
- The autoscaler continuously collects usage information based on the selected utilization metric, compares actual utilization to your desired target utilization, and uses this information to determine whether the group needs to remove instances (Scale In) or add instances (Scale Out).
- The target utilization level is the level at which you want to maintain your virtual machine (VM) instances. For example, if you scale based on CPU utilization, you can set your target utilization level at 75% and the autoscaler will maintain the CPU utilization of the specified group of instances at or close to 75%. The utilization level for each metric is interpreted differently based on the autoscaling policy. See https://cloud.google.com/compute/docs/autoscaler for more information.




### Serverless Cloud Functions

You use serverless Google Cloud functions for tasks such as changing the SSH Password, configure manager, registering Firewall Threat Defense Virtual on Firewall Management Center Virtual, deregistering Firewall Threat Defense Virtual from Firewall Management Center Virtual, and so on.

- When a new Firewall Threat Defense Virtual instance comes up in the instance group during Scale Out, you need to performs tasks such as changing the SSH Password, configure manager, registering Firewall Threat Defense Virtual on Firewall Management Center Virtual, deregistering Firewall Threat Defense Virtual from Firewall Management Center Virtual, and so on.
- Cloud functions are triggered through a Cloud Pub/Sub Topic during the Scale Out process. You also have a Log Sink with a filter that is exclusive to the addition of instances during Scale Out.

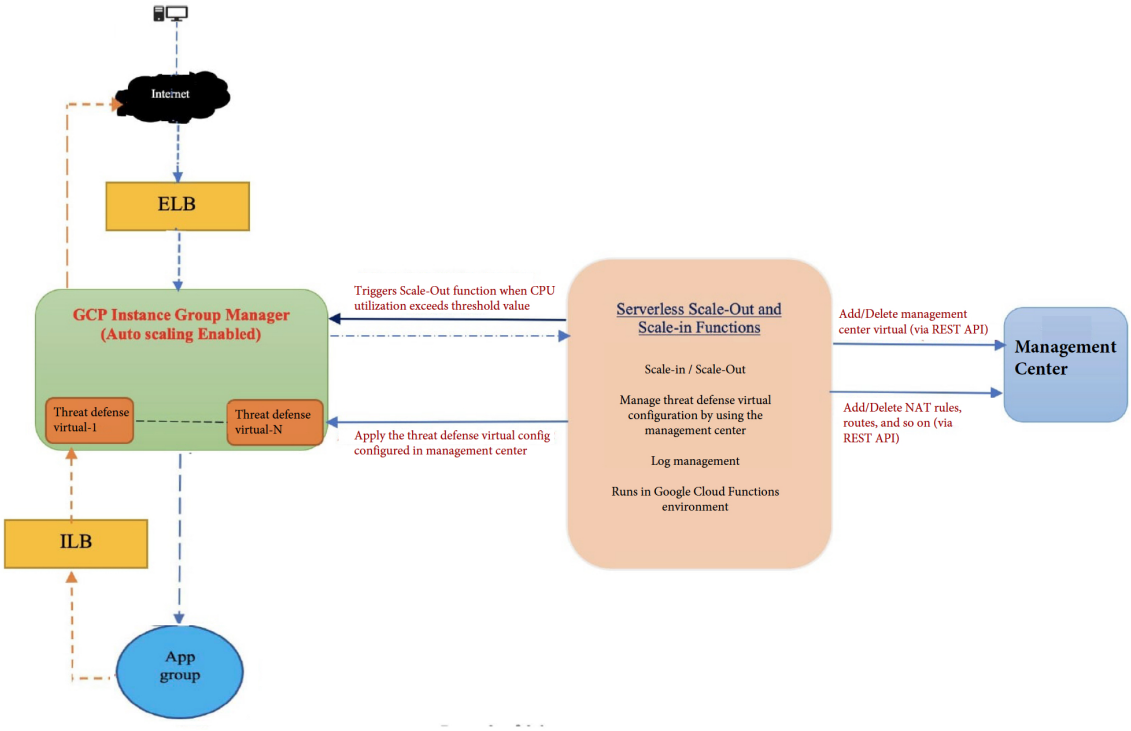
### Serverless License Deregistering using Cloud Functions

- While the instances are getting deleted during Scale In, you need to deregister the license from the Firewall Threat Defense Virtual instance and deregsister Firewall Threat Defense Virtual from Firewall Management Center Virtual.
- Cloud functions are triggered through a Cloud Pub/Sub Topic. Particularly for the deletion process, you have a Log Sink with a filter that is exclusive to the deletion of instances during Scale In.

- Cloud Function, when triggered, will SSH into the deleting Firewall Threat Defense Virtual instance and run the command for license deregistration.

### High-Level Overview of Auto Scale Solution

***Figure 49: Auto Scale Solution Overview***




# Prerequisites

## GCP Resources

### GCP Project

An existing or newly created project is required to deploy all the components of this solution.

### VPC Networks

Make sure four VPCs are available/created. An Auto Scale deployment will not create, alter, or manage any networking resources.

In addition to the existing subnetworks, create a new VPC connector in the management VPC network with a /28 subnetwork. The Cloud Function uses the VPC connector to access the Firewall Threat Defense Virtual with private IP addresses.

The Firewall Threat Defense Virtual requires 4 network interfaces, thus your virtual network requires 4 subnets for:

- Outside traffic
- Inside traffic
- Management traffic
- Diagnostic traffic

### Firewall

Firewall rules that allow inter VPC communication and also allow health probes are required to be created.

Create 4 firewall rules for the Inside, Outside, Management, and Diagnostic interfaces. Also, create a Firewall rule to allow the health check probes.

The IP addresses for the health check probes are given below:

- 35.191.0.0/16
- 130.211.0.0/22
- 209.85.152.0/22
- 209.85.204.0/22

You must note the firewall tags which are used later in the deployment manager template.

The following ports should be open in the Network Security Group to which the subnets are connected:

- SSH(TCP/22) — Required for the health probe between the Load Balancer and Firewall Threat Defense Virtual. Required for communication between the serverless functions and Firewall Threat Defense Virtual.
- Application-specific protocol/ports — Required for any user applications (for example, TCP/80, etc.).

# Build the GCP Cloud Function Package

The Firewall Threat Defense Virtual GCP Auto Scale solution requires that you build two archive files that deliver the cloud functions in the form of a compressed ZIP package.

- `ftdv_scalein.zip`
- `ftdv_scaleout.zip`

See the Auto Scale deployment instructions for information on how to build the `ftdv_scalein.zip` and `ftdv_scaleout.zip` packages.

These functions are as discrete as possible to carry out specific tasks and can be upgraded as needed for enhancements and new release support.

# Input Parameters

The following table defines the template parameters and provides an example. Once you decide on these values, you can use these parameters to create the Firewall Threat Defense Virtual device when you deploy the GCP Deployment Manager template into your GCP project.

*Table 30: Template Parameters*

| Parameter Name | Allowed Values/Type | Description |
|---|---|---|
| resourceNamePrefix | String | All the resources are created with name containing this prefix.<br>Example: demo-test |
| region | Valid regions supported by GCP<br>[String] | Name of the region where project will be deployed.<br>Example: us-central1 |
| serviceAccountMailId | String [ Email Id] | Email address that identifies the service account. |
| vpcConnectorName | String | Name of the connector that handles the traffic between your serverless environment and your VPC network.<br>Example: demo-test-vpc-connector |
| adminPassword | String | Initial password for the threat defense virtual instance. Later, this parameter is changed to 'newFtdPasswordSecret'. |
| bucketName | String | Name of the GCP storage bucket where the cloud function ZIP package will be uploaded.<br>Example: demo-test-bkt |
| coolDownPeriodSec | Integer | Number of seconds that the autoscaler should wait before it starts collecting information from a new instance.<br>Example: 30 |
| cpuUtilizationTarget | Decimal (0,1] | The average CPU utilization of the VMs in the instance group the autoscaler should maintain.<br>Example: 0.5 |

| Parameter Name | Allowed Values/Type | Description |
|---|---|---|
| deployUsingExternalIP | Boolean | Decides whether the Threat Defense Virtual management should have a public IP address.<br>Example: true<br>If set as true, the Threat Defense Virtual should have a public IP address. If set as false, a public IP address is not required. |
| diagFirewallRule | String | Name of the firewall rule that is created for the diagnostic VPC.<br>Example: cisco-ftdv-diag-firewall-rule<br>If you are deploying the Firewall Threat Defense Virtual without the diagnostic interface, leave this parameter blank or enter a dummy string. |
| diagSubnetworkName | String | Name of the VPC subnet that is used for the diagnostic interface.<br>Example: cisco-ftdv-diag-subnet<br>If you are deploying the Firewall Threat Defense Virtual without the diagnostic interface, leave this parameter blank or enter a dummy string. |
| diagVpcName | String | Name of the VPC that is used for the diagnostic interface.<br>Example: custom-ftdv-diag-vpc<br>If you are deploying the Firewall Threat Defense Virtual without the diagnostic interface, leave this parameter blank or enter a dummy string. |
| elbFePorts | Integer | ELB Fast ethernet ports.<br>Example: 80,22 |
| elbIpProtocol | String | ELB IP protocol used.<br>Example: TCP |
| elbPort | Integer | ELB port number.<br>Example: 80 |

| Parameter Name | Allowed Values/Type | Description |
|---|---|---|
| elbPortName | String | Name of the ELB port.<br>Example: tcp |
| elbPortRange | Integer | Range of ELB ports.<br>Example: 80-80 |
| elbProtocol | String | ELB protocol used.<br>Example: TCP |
| elbProtocolName | String | Name of the ELB protocol.<br>Example: TCP |
| elbTimeoutSec | Integer | ELB timeout period in seconds.<br>Example: 5 |
| elbUnhealthyThreshold | Integer | Threshold number for failed health checks.<br>Example: 2 |
| fmcIP | String | IP address of the Firewall Management Center<br>Example: 10.61.1.2 |
| fmcPasswordSecret and new FtdPasswordSecret | String | Names of the secrets created. |
| fmcUsername | String | Firewall Management Center Virtual username. |
| ftdvCheckIntervalSec | Integer | Interval between health checks.<br>Example: 300 |
| ftdvHealthCheckPort | Integer | Port number for the Firewall Threat Defense Virtual health check.<br>Example: 22 |
| ftdvHealthCheckProtocolName | String | Protocol used for the health check.<br>Example: TCP |
| ftdvPassword | String | Firewall Threat Defense Virtual password. |
| ftdvTimeoutSec | Integer | Timeout for Firewall Threat Defense Virtual connection.<br>Example: 300 |

| Parameter Name | Allowed Values/Type | Description |
|---|---|---|
| ftdvUnhealthyThreshold | Integer | Threshold number for failed health checks.<br>Example: 3 |
| grpID | String | Name of the device group created in Firewall Management Center.<br>Example: auto-group |
| healthCheckFirewallRule | String | Name of the firewall rule that allows packets from health check probe IP ranges.<br>Example: custom-ftdv-hc-firewall-rule |
| healthCheckFirewallRuleName | String | Tag of the firewall rule that allows packets from health check probe IP ranges.<br>Example: demo-test-health-allow-all |
| ilbCheckIntervalSec | Integer | Interval period for checking the ILB connection.<br>Example: 10 |
| ilbDrainingTimeoutSec | Integer | Connection draining timeout period.<br>Example: 60 |
| ilbPort | Integer | ILB port number.<br>Example: 80 |
| ilbProtocol | String | ILB protocol used.<br>Example: TCP |
| ilbProtocolName | String | ILB protocol name.<br>Example: TCP |
| ilbTimeoutSec | Integer | ILB timeout period.<br>Example: 5 |
| ilbUnhealthyThreshold | Integer | Threshold number for failed health checks.<br>Example: 3 |

| Parameter Name | Allowed Values/Type | Description |
|---|---|---|
| insideFirewallRule | String | Name of the inside firewall rule.<br>Example: custom-ftdv-in-firewall-rule |
| insideFirewallRuleName | String | Tag of the firewall rules that allows communication in Inside VPC.<br>Example: demo-test-inside-allowall |
| insideGwName | String | Name of the inside gateway.<br>Example: inside-gateway |
| insideSecZone | String | Name of the inside security zone.<br>Example: inside-zone |
| insideSubnetworkName | String | Name of the inside subnet.<br>Example: custom-ftdv-inside-subnet |
| insideVPCName | String | Name of Inside VPC.<br>Example: demo-test-inside |
| insideVPCSubnet | String | Name of Inside subnet.<br>Example: demo-test-inside-subnet |
| licenseCAPS | String | Names of the licenses used.<br>Example: BASE,MALWARE,URL Filter,THREAT |
| machineType | String | Machine type for the Firewall Threat Defense Virtual VM.<br>Example: n1-standard-4 |
| maxFTDCount | Integer | The maximum number of Firewall Threat Defense Virtual instances allowed in the instance group.<br>Example: 3 |
| maxFTDReplicas | Integer | Maximum number of Firewall Threat Defense Virtual instances in the auto scaling group.<br>Example: 2 |

| Parameter Name | Allowed Values/Type | Description |
|---|---|---|
| mgmtFirewallRule | String | Name of the management firewall rule.<br>Example: cisco-ftdv-mgmt-firewall-rule |
| mgmtFirewallRuleName | String | Tag of the firewall rules which allows communication in Management VPC.<br>Example: demo-test-mgmt-allowall |
| mgmtSubnetworkName | String | Name of the management subnet.<br>Example: custom-ftdv-mgmt-subnet |
| mgmtVPCName | String | Name of Management VPC.<br>Example: demo-test-mgmt |
| mgmtVPCSubnet | String | Name of Management Subnet.<br>Example: demo-test-mgmt-subnt |
| minFTDCount | Integer | The minimum number of Firewall Threat Defense Virtual instances available in the Instance Group at any given time.<br>Example: 1 |
| minFTDReplicas | Integer | The minimum number of Firewall Threat Defense Virtual instances in the auto scaling group.<br>Example: 2 |
| natID | String | Unique NAT ID required while registering Firewall Management Center on Firewall Threat Defense. |
| outsideFirewallRule | String | Name of the outside firewall rule.<br>Example: cisco-ftdv-out-firewall-rule |
| outsideFirewallRuleName | String | Tag of the firewall rules which allows communication in outside VPC.<br>Example: demo-test-outside-allowall |

| Parameter Name | Allowed Values/Type | Description |
|---|---|---|
| outsideGwName | String | Name of the outside gateway.<br>Example: outside-gateway |
| outsideSecZone | String | Name of the outside security zone.<br>Example: outside-zone |
| outsideSubnetworkName | String | Name of the outside subnet.<br>Example: custom-ftdv-outside-subnet |
| outsideVPCName | String | Name of Outside VPC.<br>Example: demo-test-outside |
| outsideVPCSubnet | String | Name of Outside Subnet.<br>Example: demo-test-outside-subnt |
| policyID | String | Name of the ACL policy. |
| publicKey | String | SSH key of the Firewall Threat Defense Virtual VM. |
| sourceImageURL | String | URL of the Firewall Threat Defense Virtual image which is to be used in the project. |
| sshUsingExternalIP | Boolean | Decides whether the Google functions use a public IP address or a private IP address.<br>Example: true<br>If set as true, the Google functions use a public IP address. If set as false, the Google functions use a private IP address. |
| withDiagnostic | Boolean | Decides whether the Firewall Threat Defense Virtual is deployed with the diagnostic interface or without the diagnostic interface.<br>Example: true<br>If set as true, the Firewall Threat Defense Virtual is deployed with the diagnostic interface. If set as false, the Firewall Threat Defense Virtual is deployed without the diagnostic interface. |

# Deploy the Auto Scale Solution

### Procedure

**Step 1** Clone the Git repository to a local folder.

```
git clone git_url -b branch_name
```

**Step 2** Create the bucket in gcloud CLI.

```
gsutil mb -c nearline gs://bucket_name
```

**Note**
Run any **gsutil** or **gcloud** commands in this procedure in the Google cloud shell or the Google cloud SDK installed on your system.

**Step 3** Build compressed Zip packages:

a) Create compressed Zip packages consisting of the following files from the folders `ftdv_scaleout` and `ftdv_scalein`.

- main.py
- basic_functions.py
- fmc_functions.py
- requirements.txt

**Note**
In the main.py file, use the **ssh_ip = response['networkInterfaces'] [2] ['networkIP']** command if an internal IP address is used. If an external IP address is used, enter the **ssh_ip = response['networkInterfaces'] [2] ['accessConfigs'] [0] ['natIP']** command. Also, two static routes are added in this function. You can modify the static routes using the **fmc.create_static_network_route (vm_name, 'outside', 'any_ipv4', os.getenv("OUTSIDE_GW_NAME"), metric=1)** and **fmc.create_static_network_route (vm_name, 'inside', 'any_ipv4', os.getenv("INSIDE_GW_NAME"), metric=2)** commands.

b) Rename the compressed Zip packages to `ftdv_scaleout.zip` and `ftdv_scalein.zip`.

**Note**
Navigate inside the folder, select the files, right-click, and select 'compress | archive' to make a .zip that GCP can read.

**Step 4** Upload the compressed Zip packages (`ftdv_scaleout.zip` and `ftdv_scalein.zip`) to the Cloud Editor workspace.

**Step 5** Upload the following files from the deployment manager template to the Cloud Editor workspace.

- ftdv_predeployment.yaml
- ftdv_predeployment.jinja
- ftdv_parameters.yaml

- ftdv_template.jinja

**Step 6** Copy the compressed Zip packages to the Bucket Storage.

- `gsutil cp ftdv_scaleout.zip gs://bucket_name`
- `gsutil cp ftdv_scalein.zip gs://bucket_name`

**Step 7** Create VPC and Subnet for inside, outside, management, and diagnostic interfaces.

In the management VPC, you need to have /28 subnet, for example, 10.8.2.0/28.

**Step 8** You need four firewall rules for the inside, outside, management, and diagnostic interfaces. Also, you should have a firewall rule to allow the health check probes.

**Step 9** Create two secrets for the following using the Secret Manager GUI. See https://console.cloud.google.com/security/secret-manager.

- fmc-password
- ftdv-new-password

**Step 10** Create the VPC connector.

```
gcloud beta compute networks vpc-access connectors create <vpc-connector-name>
--region <region> --subnet=</28 subnet name>
```

**Example:**

```
gcloud beta compute networks vpc-access connectors create demo-vpc-connector
--region us-central1 --subnet=outside-connect-28
Create request issued for: [demo-vpc-connector]
Waiting for operation [projects/asavgcp-poc-4krn/locations/us-central1/operations/
10595de7-837f-4c19-9396-0c22943ecf15] to complete...done.
Created connector [demo-vpc-connector].
```

**Step 11** Deploy the Firewall Management Center Virtual on any public cloud platform with a public IP. See Cisco Secure Firewall Management Center Virtual Getting Started Guide for more information on how to deploy Firewall Management Center Virtual on various public cloud platforms.

**Note**
Perform Steps 12 to 16 on the deployed Firewall Management Center Virtual instance.

**Step 12** On the Firewall Management Center Virtual instance - Create a user restapi for Firewall Management Center Virtual and use the same password that is saved in the fmcpassword secret. See Users for more information.

**Step 13** On the Firewall Management Center Virtual instance - Create a Device Group, Access Control Policy, and an Access Control Rule. See Add a Device Group, Creating a Basic Access Control Policy, and Create and Edit Access Control Rules for more information.

**Step 14** On the Firewall Management Center Virtual instance - Create the objects given below. See Object Management for more information on how to create objects on Firewall Management Center Virtual.

- ELB-IP
- ILB-IP
- Application-IP
- Health Check IP ranges (4)

- Metadata

```
object network hc1
  subnet 35.191.0.0 255.255.0.0
object network metadata
  host 169.254.169.254
object network ilb-ip
  host 10.52.1.218
object network hc2
  subnet 130.211.0.0 255.255.252.0
object network elb-ip
  host 34.85.214.40
object network hc3
  subnet 209.85.152.0 255.255.252.0
object network hc4
  subnet 209.85.204.0 255.255.252.0
object network inside-linux
  host 10.52.1.217
object network outside-gateway
  host <>
object network inside-gateway
  host <>
```

**Step 15** On the Firewall Management Center Virtual instance - Create Security Zones (Interface Objects). See Creating Security Zone and Interface Group Objects for more information.

- inside-security-zone
- outside-security-zone

**Step 16** On the Firewall Management Center Virtual instance - Create NAT Policy and NAT Rules. See Network Address Translation for more information.

```
nat (inside,outside) source dynamic hc1 interface destination static ilb-ip metadata service
SVC_4294968559 SVC_4294968559
nat (inside,outside) source dynamic hc2 interface destination static ilb-ip metadata service
SVC_4294968559 SVC_4294968559
nat (inside,outside) source dynamic any interface
nat (outside,inside) source dynamic hc1 interface destination static elb-ip metadata service
SVC_4294968559 SVC_4294968559
nat (outside,inside) source dynamic hc2 interface destination static elb-ip metadata service
SVC_4294968559 SVC_4294968559
nat (outside,inside) source dynamic hc3 interface destination static elb-ip metadata service
SVC_4294968559 SVC_4294968559
nat (outside,inside) source dynamic hc4 interface destination static elb-ip metadata service
SVC_4294968559 SVC_4294968559
nat (outside,inside) source dynamic any interface destination static elb-ip inside-linux
```

| # | Direction | Type | Source Interface Objects | Destination Interface Objects | Original Sources | Original Destinations | Original Services | Translated Sources | Translated Destinations | Translated Services | Options |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | D... | inside-zone | outside-zone | hc1 | ilb-ip | Original HTTP | Interface | metadata | Original HTTP | Dns:false |
| 2 | | D... | inside-zone | outside-zone | hc2 | ilb-ip | Original HTTP | Interface | metadata | Original HTTP | Dns:false |
| 3 | | D... | inside-zone | outside-zone | any-ipv4 | | | Interface | | | Dns:false |
| 4 | | D... | outside-zone | inside-zone | hc1 | elb-ip | Original HTTP | Interface | metadata | Original HTTP | Dns:false |
| 5 | | D... | outside-zone | inside-zone | hc2 | elb-ip | Original HTTP | Interface | metadata | Original HTTP | Dns:false |
| 6 | | D... | outside-zone | inside-zone | hc3 | elb-ip | Original HTTP | Interface | metadata | Original HTTP | Dns:false |
| 7 | | D... | outside-zone | inside-zone | hc4 | elb-ip | Original HTTP | Interface | metadata | Original HTTP | Dns:false |
| 8 | | D... | outside-zone | inside-zone | any-ipv4 | elb-ip | | Interface | inside-linux | | Dns:false |

**Step 17** Update the parameters in the Jinja and YAML files for the Pre-Deployment and Firewall Threat Defense Virtual Autoscale deployment.

a) Open the `ftdv_predeployment.yaml` file and update the following parameters:

- **resourceNamePrefix**: <resourceNamePrefix>
- **region**: <region>
- **serviceAccountMailId**: <serviceAccountMailId>
- **vpcConnectorName**: <VPC-Connector-Name>
- **bucketName**: <bucketName>
- **fmcIP**: <Firewall Management Center-IP-address>
- **regID**: <registration-ID>
- **natID**: <unique-NAT-ID>
- **grpID**: <device-group-name>
- **policyID**: <acl-policy-name>
- **licenseCAPS**: <licenses>
- **fmcPasswordSecret**: <Firewall Management Center-password>
- **newFtdPasswordSecret**: <new-Firewall Threat Defense Virtual-password>
- **fmcUsername**: <username>
- **ftdvPassword**: <password>
- **outsideGwName**: <outside-gateway-name>
- **insideGwName**: <inside-gateway-name>
- **outsideSecZone**: <outside-security-zone>
- **insideSecZone**: <inside-security-zone>
- **sshUsingExternalIP**: <true/false>

b) The `ftdv_predeployment.jinja` file takes parameters from the `ftdv_predeployment.yaml` file.

c) Open the `ftdv_parameters.yaml` file and update the following parameters.

**VPC and Firewall Parameters**

- **mgmtVpcName**: <mgmt-vpc-name>
- **diagVpcName**: <diagnostic-vpc-name>
- **outsideVpcName**: <outside-vpc-name>
- **insideVpcName**: <inside-vpc-name>
- **mgmtSubnetworkName**: <mgmt-subnet-name>
- **diagSubnetworkName**: <diagnostic-subnet-name>
- **outsideSubnetworkName**: <outside-subnet-name>
- **insideSubnetworkName**: <inside-subnet-name>
- **mgmtFirewallRule**: <mgmt-firewall-rule>
- **diagFirewallRule**: <diagnostic-firewall-rule>
- **outsideFirewallRule**: <outside-firewall-rule>
- **insideFirewallRule**: <inside-firewall-rule>
- **healthCheckFirewallRule**: <healthcheck-firewall-rule>
- **adminPassword**: <initial-Firewall Threat Defense Virtual-password>
- **deployUsingExternalIP**: <true/false>

**Instance Template parameters**

- **machineType**: <machine-type>
- **sourceImageURL**: <source-image-URL>

**FTDv Health Check**

- **ftdvHealthCheckPort**: <port-number>
- **ftdvCheckIntervalSec**: <interval-in-seconds>
- **ftdvTimeoutSec**: <timeout-in-seconds>
- **ftdvHealthCheckProtocolName**: <protocol-name>
- **ftdvUnhealthyThreshold**: <threshold-count>

**FTDv Autoscaler**

- **cpuUtilizationTarget**: <percentage-in-decimals (for example, 0.7)>
- **coolDownPeriodSec**: <cooldown-period-in-seconds>
- **minFTDReplicas**: <min-number-of-FTDv-instances>
- **maxFTDReplicas**: <max-number-of-FTDv-instances>

**ELB Services**

- **elbPort**: <port-number>
- **elbPortName**: <port-name>
- **elbProtocol**: <protocol-name>
- **elbTimeoutSec**: <timeout-in-seconds>
- **elbProtocolName**: <protocol-name>
- **elbUnhealthyThreshold**: <threshold-number-for-failed-health-checks>
- **elbIpProtocol**: <IP-Protocol>
- **elbPortRange**: <port-range>
- **elbFePorts**: <fast-ethernet-ports>

**ILB Services**

- **ilbProtocol**: <protocol-name>
- **ilbDrainingTimeoutSec**: <timeout-in-seconds>
- **ilbPort**: <port-number>
- **ilbCheckIntervalSec**: <interval-in seconds>
- **ilbTimeoutSec**: <timeout-in-seconds>
- **ilbProtocolName**: <protocol-name>
- **ilbUnhealthyThreshold**: <threshold-number-for-failed-health-checks>

**Note**
For the Firewall Threat Defense Virtual Auto Scale, the **cpuUtilizationTarget: 0.5** parameter is set and you can edit it according to your requirements. This value signifies 50% CPU usage of all the Firewall Threat Defense Virtual Instance Groups.

d) The `ftdv_template.jinja` file takes parameters from the `ftdv_parameters.yaml` file.

**Step 18** Deploy the pre-deployment YAML configuration.

```
gcloud deployment-manager deployments create <pre-deployment-name>
--config ftdv_predeployment.yaml
```

**Example:**

```
gcloud deployment-manager deployments create demo-predeployment
--config ftdv_predeployment.yaml

The fingerprint of the deployment is b'9NOy0gsTPgg16SqUEVsBjA=='
Waiting for create [operation-1624383045917-5c55e266e596d-4979c5b6-66d1025c]...done.
Create operation operation-1624383045917-5c55e266e596d-4979c5b6-66d1025c
completed successfully
```

**Step 19** Create the Firewall Threat Defense Virtual Auto Scale deployment.

```
gcloud deployment-manager deployments create <deployment-name>
--config ftdv_parameters.yaml
```

**Example:**

```
gcloud deployment-manager deployments create demo-asav-autoscale
--config ftdv_parameters.yaml
The fingerprint of the deployment is b'1JCQi7Il-laWOY7vOLza0g=='
Waiting for create [operation-1624383774235-5c55e51d79d01-1a3acf92-4f3daf16]...done.
Create operation operation-1624383774235-5c55e51d79d01-1a3acf92-4f3daf16
completed successfully.
```

**Step 20** Create a route for ILB to forward the packets from the inside application to the Internet.

```
gcloud beta compute routes create <ilb-route-name>
--network=<inside-vpc-name> --priority=1000 --destination-range=0.0.0.0/0
--next-hop-ilb=<ilb-forwarding-rule-name> --next-hop-ilb-region=<region>
```

**Example:**

```
gcloud beta compute routes create demo-ilb --network=sdt-test-asav-inside
--priority=1000 --destination-range=0.0.0.0/0 --next-hop-ilb=demo-asav-fr-ilb
--next-hop-ilb-region=us-central1
Created [https://www.googleapis.com/compute/beta/projects/asavgcp-poc-4krn/global
/routes/demo-ilb].
```

# Auto Scale Logic

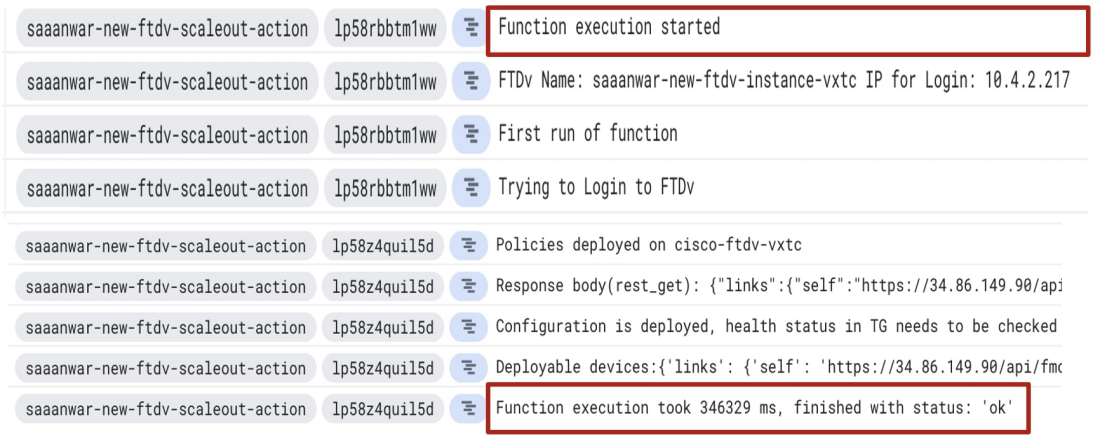
- The autoscaler treats the target CPU utilization level as a fraction of the average use of all vCPUs over time in the instance group.
- If the average utilization of your total vCPUs exceeds the target utilization, the autoscaler adds more VM instances. If the average utilization of your total vCPUs is less than the target utilization, the autoscaler removes instances.
- For example, setting a 0.75 target utilization tells the autoscaler to maintain an average utilization of 75% among all vCPUs in the instance group.
- Only CPU utilization metrics are used in scaling decisions.
- This logic is based on the assumption that load balancer will try to equally distribute connections across all Firewall Threat Defense Virtuals, and on average, all Firewall Threat Defense Virtuals should be loaded equally.

# Logging and Debugging

Logs of cloud functions can be viewed as follows.

- Scale Out function logs

*Figure 50: Scale Out Function Logs*




In the scale out function logs given above, the **Function execution started** and the **Function execution took 346329 ms, finish with status: 'ok'** entries indicate the start and the end of the function logs respectively. You can also track other operations such as the first function run, Firewall Threat Defense Virtual login, policy deployment, and so on.

- Scale In function logs




In the scale out function logs given above, the **Function execution started** and the **Function execution took 50852 ms, finish with status: 'ok'** entries indicate the start and the end of the function logs respectively. You can also track other operations such as initiation of the deregistration process, status of deregistration, obtaining a new authToken, and so on.

# Troubleshooting

The following are common error scenarios and debugging tips for Firewall Threat Defense Virtual Auto Scale for GCP:

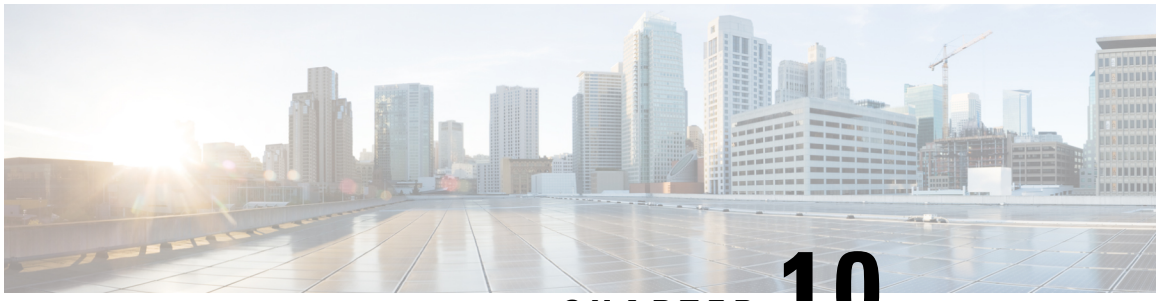
- `main.py` not found—Ensure that the Zip package is made only from the files. You can go to cloud functions and check the file tree. There should not be any folder.

- Error while deploying the template—Ensure that all the parameter values within "<>" are filled in jinja and yaml, or check if a deployment by the same name already exists.
- Google Function cannot reach Firewall Threat Defense Virtual—Ensure that the VPC connector is created and the same name is mentioned in the YAML parameter file.
- Authentication Failed while SSH-ing Firewall Threat Defense Virtual—Ensure that the Public and Private key pair is correct.
- Auth-token not found—Ensure that the Firewall Management Center Virtual password in Secret is correct.
- Unhealthy Firewall Threat Defense Virtual and traffic issues—Ensure that there are no issues in the firewall rules and routes.
- Unable to manually log in to Firewall Threat Defense Virtual—Ensure that you are using the new password. The old password is changed by the scale-out function.
- Unable to register device on Firewall Management Center Virtual—Ensure that Firewall Threat Defense Virtual is reachable from Firewall Management Center Virtual. The management interface of Firewall Threat Defense Virtual and Firewall Management Center Virtual should be in the same subnet.
- Preserved connections forming a loop between ILB and Firewall Threat Defense Virtual cause high CPU usage due to the initiation of health probe requests. To reduce high CPU usage, you can use one of the following options:

  Option 1 - On the Firewall Management Center Virtual, disable data interface, configure health probe NAT rules, and enable data interface. For more information on data interfaces and NAT, refer Interface Overview and Network Address Translation.

  Option 2 - After applying health probe NAT rules from Firewall Management Center Virtual, log in to the Firewall Threat Defense Virtual console, and use the **clear conn** command. If you have set up clustering, use the **cluster exec clear conn** command.

  Verify CPU usage using the **show cpu** command on the Firewall Threat Defense Virtual console.



CHAPTER 10

# Deploy the Firewall Threat Defense Virtual on Cisco HyperFlex

This chapter describes the procedures to deploy the Firewall Threat Defense Virtual on Cisco HyperFlex on a vCenter server or a standalone ESXi host.


- Overview, on page 349
- End-to-End Procedure, on page 350
- System Requirements, on page 350
- Guidelines and Limitations, on page 353
- How to Manage Secure Firewall Threat Defense Virtual Device, on page 356
- Overview, on page 357
- Deploy the Firewall Threat Defense Virtual, on page 357
- Complete the Firewall Threat Defense Virtual Setup using CLI, on page 360
- Enabling Jumbo Frames, on page 361
- Troubleshooting, on page 362


## Overview

The Cisco Secure Firewall Threat Defense Virtual (formerly Firepower Threat Defense Virtual) brings Cisco's Secure Firewall functionality to virtualized environments, enabling consistent security policies to follow workloads across your physical, virtual, and cloud environments, and between clouds.

HyperFlex systems deliver hyperconvergence for any application, and anywhere. HyperFlex with Cisco Unified Computing System (Cisco UCS) technology that is managed through the Cisco Intersight cloud operations platform can power applications and data anywhere, optimize operations from a core datacenter to the edge and into public clouds, and therefore increase agility through accelerating DevOps practices.

This chapter describes how the Firewall Threat Defense Virtual functions within a Cisco HyperFlex environment, including feature support, system requirements, guidelines, and limitations. This chapter also describes your options for managing the Firewall Threat Defense Virtual. It is important that you understand your management options before you begin your deployment. You can manage and monitor the Firewall Threat Defense Virtual using the Secure Firewall Management Center (formerly Firepower Management Center) or the Secure Firewall Device Manager (formerly Firepower Device Manager). Other management options maybe available.

# End-to-End Procedure

The following flowchart illustrates the workflow for deploying Threat Defense Virtual on Cisco HyperFlex.




| | Workspace | Steps |
|---|---|---|
| 1 | Hyperflex | Deploy the Threat Defense Virtual on Hyperflex: Download the threat defense virtual VI OVF template file from Cisco.com. |
| 2 | Hyperflex | Deploy the Threat Defense Virtual on Hyperflex: Review the OVF template information. |
| 3 | Hyperflex | Deploy the Threat Defense Virtual on Hyperflex: Customize the deployment configuration. |
| 4 | Hyperflex | Deploy the Threat Defense Virtual on Hyperflex: Review and verify the displayed information. Click Finish to begin deployment of the OVF template. |
| 5 | Management Center or Device Manager | Manage Threat Defense Virtual:<br>• Using Management Center<br>• Using Device Manager |

# System Requirements

### Versions

| Manager Version | Device Version |
|---|---|
| Firewall Device Manager 7.0 | Firewall Threat Defense 7.0 |
| Firewall Management Center 7.0 | |

See theCisco Secure Firewall Threat Defense Compatibility Guide for the most current information about hypervisor support for the Firewall Threat Defense Virtual.

### Firewall Threat Defense Virtual Memory, Disk Sizing and vCPUs

The specific hardware used for the Firewall Threat Defense Virtual deployments can vary, depending on the number of instances deployed and usage requirements. Each instance of the Firewall Threat Defense Virtual requires a minimum resource allocation—amount of memory, number of CPUs, and disk space—on the server.

| Settings | Value |
|---|---|
| Performance Tiers | **Version 7.0 and later**<br>The Firewall Threat Defense Virtual supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.<br>• FTDv5 4vCPU/8GB (100Mbps)<br>• FTDv10 4vCPU/8GB (1Gbps)<br>• FTDv20 4vCPU/8GB (3Gbps)<br>• FTDv30 8vCPU/16GB (5Gbps)<br>• FTDv50 12vCPU/24GB (10Gbps)<br>• FTDv100 16vCPU/32GB (16Gbps)<br>See the "Licensing the System" chapter in the*Secure Firewall Management Center Configuration* for guidelines when licensing your Firewall Threat Defense Virtual device.<br>**Note**<br>To change the vCPU/memory values, you must first power off the Firewall Threat Defense Virtual device. |
| Storage | Based on Disk Format selection.<br>• Thin Provision disk size is 48.24GB. |
| vNICs | The Firewall Threat Defense Virtual supports the following virtual network adapters:<br>• **VMXNET3**—The Firewall Threat Defense Virtual on VMware now defaults to VMXNET3 interfaces when you create a virtual device. Previously, the default was e1000. (7.1 and later) The vmxnet3 driver uses the first Ethernet adapter for management. The second adapter is unused. (7.0 and earlier)<br>The VMXNET3 driver uses two management interfaces. The first two Ethernet adapters must be configured as management interfaces; one for device management/registration, one for diagnostics. |

### Firewall Threat Defense Virtual Licenses

- Configure all license entitlements for the security services from the Firewall Management Center.
- See *Licensing the System* in the Secure firewall Management Center Configuration Guide for more information about how to manage licenses.

### Configurations and Clusters for HyperFlex HX-Series

| Configurations | Clusters |
|---|---|
| HX220c converged nodes | • Flash cluster<br>• Minimum of 3 Node Cluster (Databases, VDI, VSI) |
| HX240c converged nodes | • Flash cluster<br>• Minimum of 3 Node Cluster (VSI: IT/Biz Apps, Test/Dev) |
| HX220C and Edge (VDI, VSI, ROBO)<br>HX240C (VDI, VSI, Test/Dev | • Hybrid cluster<br>• Minimum of 3 Node Cluster |
| B200 + C240/C220 | Compute bound apps/VDI |

Deployment options for the HyperFlex HX-Series:

- Hybrid Cluster
- Flash Cluster
- HyperFlex HX Edge
- SED drives
- NVME Cache
- GPUs

For HyperFlex HX cloud powered management option, refer to the *Deploying HyperFlex Fabric Interconnect-attached Clusters* section in the Cisco HyperFlex Systems Installation Guide.

### HyperFlex Components and Versions

| Component | Version |
|---|---|
| VMware vSphere/VMware ESXI | 7.0<br>For more information on Threat Defense Virtual compatibility with VMware vSphere/VMware ESXI, see Threat Defense Virtual Compatibility: VMware. |

| Component | Version |
|---|---|
| HyperFlex Data Platform | 4.5.1a-39020 and later<br>. |

# Guidelines and Limitations

### Supported Features

- Deployment Modes–Routed (Standalone), Routed (HA), Inline Tap, Inline, Passive, and Transparent
- Licensing–Only BYOL
- IPv6
- Firewall Threat Defense Virtual native HA
- Jumbo frames
- HyperFlex Data Center Clusters (excluding Stretched Clusters)
- HyperFlex Edge Clusters
- HyperFlex All NVMe, All Flash and Hybrid converged nodes
- HyperFlex Compute-only Nodes

### Unsupported Features

Firewall Threat Defense Virtual running with SR-IOV has not been qualified with HyperFlex.

**Note** HyperFlex supports SR-IOV, but requires a PCI-e NIC in addition to the MLOM VIC.

### General Guidelines

To configure vSwitches for HyperFlex, you can either use the GUI or the command line interface. These configurations are helpful when you are installing multiple ESX servers and planning to script the vSwitch configuration. For more information, refer to the Configure the vSwitches section in the Cisco HyperFlex Systems Network and External Storage Management Guide.

The following is a concordance of Network Adapter, Source Networks and Destination Networks for the Firewall Threat Defense Virtual interfaces:

| Network Adapter | Source Network | Destination Network | Function |
|---|---|---|---|
| Network adapter 1 | Management0-0 | Management0/0 | Management |
| Network adapter 2 | Diagnostic0-0 | Diagnostic | Diagnostic |
| Network adapter 3 | GigabitEthernet0-0 | GigabitEthernet0/0 | Outside |

| Network Adapter | Source Network | Destination Network | Function |
|---|---|---|---|
| Network adapter 4 | GigabitEthernet0-1 | GigabitEthernet0/1 | Inside |
| Network adapter 5 | GigabitEthernet0-2 | GigabitEthernet0/2 | Data traffic (optional) |
| Network adapter 6 | GigabitEthernet0-3 | GigabitEthernet0/3 | Data traffic (optional) |
| ...till Network adapter 10 | | | |

### Performance Optimizations

To achieve the best performance out of the Firewall Threat Defense Virtual, you can make adjustments to the both the VM and the host. See Virtualization Tuning and Optimization on HyperFlex for more information.

**Receive Side Scaling**—The Firewall Threat Defense Virtual supports Receive Side Scaling (RSS), which is a technology utilized by network adapters to distribute network receive traffic to multiple processor cores. Supported on Version 7.0 and later. See Multiple RX Queues for Receive Side Scaling (RSS) for more information.

### Snort

- If you are observing abnormal behavior such as Snort taking a long time to shut down, or the VM being slow in general or when a certain process is executed, collect logs from the Firewall Threat Defense Virtual and the VM host. Collection of overall CPU usage, memory, I/O usage, and read/write speed logs will help troubleshoot the issues.
- High CPU and I/O usage is observed when Snort is shutting down. If a number of Firewall Threat Defense Virtual instances have been created on a single host with insufficient memory and no dedicated CPU, Snort will take a long time to shut down which will result in the creation of Snort cores.

### Modify the Security Policy Settings for a vSphere Standard Switch

For a vSphere standard switch, the three elements of the Layer 2 Security policy are promiscuous mode, MAC address changes, and forged transmits. The Firewall Threat Defense Virtual uses promiscuous mode to operate, and the Firewall Threat Defense Virtual high availability depends on switching the MAC address between the active and the standby to operate correctly.

The default settings will block correct operation of the Firewall Threat Defense Virtual. See the following required settings:

***Table 31: vSphere Standard Switch Security Policy Options***

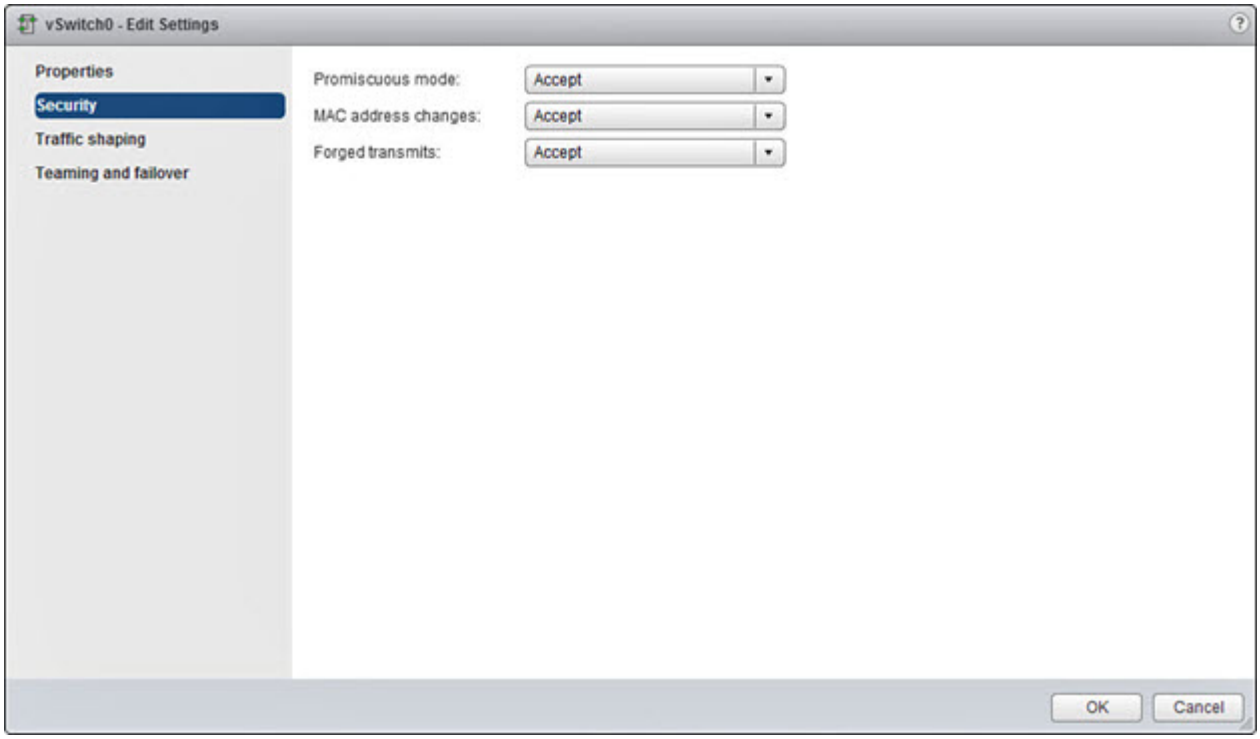
| Option | Required Setting | Action |
|---|---|---|
| Promiscuous Mode | Accept | You must edit the security policy for a vSphere standard switch in the vSphere Web Client and set the Promiscuous mode option to Accept.<br>Firewalls, port scanners, intrusion detection systems and so on, need to run in promiscuous mode. |

| Option | Required Setting | Action |
|---|---|---|
| MAC Address Changes | Accept | You should verify the security policy for a vSphere standard switch in the vSphere Web Client and confirm the MAC address changes option is set to Accept. |
| Forged Transmits | Accept | You should verify the security policy for a vSphere standard switch in the vSphere Web Client and confirm the Forged transmits option is set to Accept. |

Use the following procedure to configure the default settings for correct operation of the Firewall Threat Defense Virtual.

1. In the vSphere Web Client, navigate to the HyperFlex cluster.
2. On the **Manage** tab, click **Networking**, and select **Virtual switches**.
3. Select a standard switch from the list and click **Edit settings**.
4. Select **Security** and view the current settings.
5. **Accept** promiscuous mode activation, MAC address changes, and forged transmits in the guest operating system of the virtual machines attached to the standard switch.

***Figure 51: vSwitch Edit Settings***



6. Click **OK**.

**Note** Ensure these settings are the same on all networks that are configured for management and failover (HA) interfaces on the Firewall Threat Defense Virtual devices.

**Related Documents**

Release Notes for Cisco HX Data Platform

Configuration Guides for Cisco HX Data Platform

Cisco HyperFlex 4.0 for Virtual Server Infrastructure with VMware ESXi

Cisco HyperFlex Systems Solutions Overview

Cisco HyperFlex Systems Documentation Roadmap

# How to Manage Secure Firewall Threat Defense Virtual Device

You have two options to manage your Secure Firewall Threat Defense Virtual.

## Secure Firewall Management Center

If you are managing large numbers of devices, or if you want to use the more complex features and configurations that the Firewall Threat Defense allows, use the Firewall Management Center to configure your devices instead of the integrated Firewall Device Manager.

**Important** You cannot use both the Firewall Device Manager and the Firewall Management Center to manage the Firewall Threat Defense device. Once the Firewall Device Manager integrated management is enabled, it won't be possible to use the Firewall Management Center to manage the Firewall Threat Defense device, unless you disable the local management and re-configure the management to use the Firewall Management Center. On the other hand, when you register the Firewall Threat Defense device to the Firewall Management Center, the Firewall Device Manager onboard management service is disabled.

**Caution** Currently, Cisco does not have an option to migrate your Firewall Device Manager configuration to the Firewall Management Center and vice-versa. Take this into consideration when you choose what type of management you configure for the Firewall Threat Defense device.

## Secure Firewall Device Manager

The Firewall Device Manager is a web interface included on most Firewall Threat Defense devices. It lets you configure the basic features of the software that are most commonly used for small networks. It is especially designed for networks that include a single device or just a few, where you do not want to use a high-powered multiple-device manager to control a large network with many devices.

**Note** See the Cisco Secure Firewall Threat Defense Compatibility Guide for list of devices that support the Firewall Device Manager.

# Overview

You can deploy the Firewall Threat Defense Virtual to Cisco HyperFlex on a VMware vCenter server.

To successfully deploy the Firewall Threat Defense Virtual, you must be familiar with VMware and vSphere, including vSphere networking, ESXi host setup and configuration, and virtual machine guest deployment.

The Firewall Threat Defense Virtual for Cisco HyperFlex is distributed using the Open Virtualization Format (OVF), a standard method of packaging and deploying virtual machines. VMware provides several ways to provision vSphere virtual machines. The optimal way for your environment depends on the size and type of your infrastructure and the goals you want to achieve.

You can use the VMware vSphere Web Client to access your Cisco HyperFlex environment.

# Deploy the Firewall Threat Defense Virtual

Use this procedure to deploy the Firewall Threat Defense Virtual appliance to Cisco HyperFlex on a vSphere vCenter Server.

### Before you begin

- Ensure that you have deployed Cisco HyperFlex and performed all the post-installation configuration tasks. For more information, see Cisco HyperFlex Systems Documentation Roadmap.
- You must have at least one network configured in vSphere (for management) before you deploy the Firewall Threat Defense Virtual.
- Download the Firewall Threat Defense Virtual VI OVF template file from Cisco.com: *Cisco_Secure_Firewall_Threat_Defense_Virtual-VI-X.X.X-xxx.ovf*, where *X.X.X-xxx* is the version and build number.

### Procedure

**Step 1** Log in to the vSphere Web Client.

**Step 2** Select the HyperFlex cluster where you want to deploy the Firewall Threat Defense Virtual, and click **ACTIONS** > **Deploy OVF Template**.

**Step 3** Browse your file system for the OVF template source location, and click **NEXT**.

Select the Firewall Threat Defense Virtual VI OVF template:

*Cisco_Secure_Firewall_Threat_Defense_Virtual-VI-X.X.X-xxx.ovf*

where *X.X.X-xxx* is the version and build number of the archive file you downloaded.

**Step 4** Specify a name and location for the Firewall Threat Defense Virtual, and click **NEXT**.

**Step 5** Select a compute resource, and wait until the compatibility check is complete.

If the compatibility check succeeds, click **NEXT**.

**Step 6** Review the OVF template information (product name, version, vendor, download size, size on disk, and description), and click **NEXT**.

**Step 7** Review and accept the license agreement that is packaged with the OVF template (VI templates only), and click **NEXT**.

**Step 8** Select a deployment configuration (vCPU/memory values), and click **NEXT**.

**Step 9** Select a storage location and virtual disk format, and click **NEXT**.

On this window, select from datastores already configured on the destination HyperFlex cluster. The virtual machine configuration file and virtual disk files that are stored on the datastore. Select a datastore large enough to accommodate the virtual machine and all of its virtual disk files.

When you select **Thick Provisioned** as the virtual disk format, all storage is immediately allocated. When you select **Thin Provisioned** as the virtual disk format, storage is allocated on demand as data is written to the virtual disks. Thin provisioning can also reduce the amount of time it takes to deploy the virtual appliance.

**Step 10** Map the networks specified in the OVF template to networks in your inventory, and click **NEXT**.

Ensure the Management0-0 interface is associated with a VM Network that is reachable from the Internet. Non-management interfaces are configurable from either the Firewall Management Center or from the Firewall Device Manager, depending on your management mode.

The networks may not be in alphabetical order. If it is too difficult to find your networks, you can change the networks later from the **Edit Settings** dialog box. After you deploy, right-click the Firewall Threat Defense Virtual instance, and choose **Edit Settings**. However, that screen does not show the Firewall Threat Defense Virtual IDs (only Network Adapter IDs).

See the following concordance of Network Adapter, Source Networks and Destination Networks for the Firewall Threat Defense Virtual interfaces (note these are the default vmxnet3 interfaces):

| Network Adapter | Source Networks | Destination Networks | Function |
|---|---|---|---|
| Network adapter 1 | Management0-0 | Management0/0 | Management |
| Network adapter 2 | Diagnostic0-0 | Diagnostic0/0 | Diagnostic |
| Network adapter 3 | GigabitEthernet0-0 | GigabitEthernet0/0 | Outside data |
| Network adapter 4 | GigabitEthernet0-1 | GigabitEthernet0/1 | Inside date |
| Network adapter 5 | GigabitEthernet0-2 | GigabitEthernet0/2 | Data traffic (Optional) |
| Network adapter 6 | GigabitEthernet0-3 | GigabitEthernet0/3 | Data traffic (Optional) |
| Network adapter 7 | GigabitEthernet0-4 | GigabitEthernet0/4 | Data traffic (Optional) |
| Network adapter 8 | GigabitEthernet0-5 | GigabitEthernet0/5 | Data traffic (Optional) |
| Network adapter 9 | GigabitEthernet0-6 | GigabitEthernet0/6 | Data traffic (Optional) |
| Network adapter 10 | GigabitEthernet0-7 | GigabitEthernet0/7 | Data traffic (Optional) |

You can have a total of 10 interfaces when you deploy the Firewall Threat Defense Virtual. For data interfaces, make sure that the Source Networks map to the correct Destination Networks, and that each data interface maps to a unique subnet or VLAN. You do not need to use all the Firewall Threat Defense Virtual interfaces; for interfaces you do not intend to use, you can simply leave the interface disabled within the Firewall Threat Defense Virtual configuration.

**Step 11** Set the user-configurable properties packaged with the OVF template:

**Note**
We recommend that you configure all the required customizations in this step. If you did not configure all the required customizations, you must complete the setup by logging in to the CLI after the deployment. For instructions, see Complete the Firewall Threat Defense Virtual Setup using CLI, on page 360.

a) **Password**

Set the password for the Firewall Threat Defense Virtual admin access.

b) **Network**

Set the network information, including the Fully Qualified Domain Name (FQDN), DNS, search domain, and network protocol (IPv4 or IPv6).

c) **Management**

Set the management mode. Click the drop-down arrow for **Enable Local Manager** and select **Yes** to use the integrated Firewall Device Manager web-based configuration tool. Select **No** to use the Firewall Management Center to manage this device.

d) **Firewall Mode**

Set the initial firewall mode. Click the drop-down arrow for **Firewall Mode** and choose one of the two supported modes, either **Routed** or **Transparent**.

If you chose **Yes** for **Enable Local Manager**, you can only select **Routed** firewall mode. You cannot configure transparent firewall mode interfaces using the local Firewall Device Manager.

e) **Registration**

If you chose **No** for **Enable Local Manager**, you need to provide the required credentials to register this device to the managing **Firepower Management Center**. Provide the following:

- **Managing Defense Center**—Enter the hostname or IP address of the Firewall Management Center.
- **Registration Key**—The registration key is a user-generated one-time use key that must not exceed 37 characters. Valid characters include alphanumerical characters (A–Z, a–z, 0–9) and the hyphen (-). You must remember this registration key when you add the device to the Firewall Management Center.
- **NAT ID**—If the Firewall Threat Defense Virtual and the Firewall Management Center are separated by a Network Address Translation (NAT) device, and the Firewall Management Centerr is behind a NAT device, enter a unique NAT ID. This is a user-generated one-time use key that must not exceed 37 characters. Valid characters include alphanumerical characters (A–Z, a–z, 0–9) and the hyphen (-).

f) Click **NEXT**.

**Step 12** Review and verify the displayed information. To begin the deployment with these settings, click **FINISH**. To make any changes, click **BACK** to navigate back through the screens.

After you complete the wizard, the vSphere Web Client processes the virtual machine; you can see the “Initialize OVF deployment” status in the **Global Information** area **Recent Tasks** pane.

When it is finished, you see the Deploy OVF Template completion status.

The Firewall Threat Defense Virtual virtual instance appears under the specified data center in the Inventory. Booting up the new VM could take up to 30 minutes.

**Note**
To successfully register the Firewall Threat Defense Virtual with the Cisco Licensing Authority, the Firewall Threat Defense Virtual requires Internet access. You need to perform additional configuration after deployment to achieve Internet access and successful license registration.

### What to do next

Your next steps depend on what management mode you chose.

- If you chose **No** for **Enable Local Manager**, you'll use the Firewall Management Center to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409.

**Note** If you did not configure all the required customizations while deploying the Firewall Threat Defense Virtual, you must complete the setup using the CLI. For instructions, see Complete the Firewall Threat Defense Virtual Setup using CLI, on page 360.

# Complete the Firewall Threat Defense Virtual Setup using CLI

If you did not configure all the required customizations while deploying the Firewall Threat Defense Virtual, you must complete the setup using the CLI.

### Procedure

**Step 1** Open the VMware console.

**Step 2** At the **firepower login** prompt, log in with the default credentials of username **admin** and the password **Admin123**.

**Step 3** When the Firewall Threat Defense system boots, a setup wizard prompts you for the following information required to configure the system:

- Accept EULA
- New admin password
- IPv4 or IPv6 configuration
- IPv4 or IPv6 DHCP settings
- Management port IPv4 address and subnet mask, or IPv6 address and prefix
- System name
- Default gateway

- DNS setup
- HTTP proxy
- Management mode (local management uses the Firewall Device Manager).

**Step 4** Review the Setup wizard settings. Defaults or previously entered values appear in brackets. To accept previously entered values, press **Enter**.

The VMware console may display messages as your settings are implemented.

**Step 5** Complete the system configuration as prompted.

**Step 6** Verify the setup was successful when the console returns to the # prompt.

**Note**
To successfully register the Firewall Threat Defense Virtual with the Cisco Licensing Authority, the Firewall Threat Defense Virtual requires Internet access. You might need to perform additional configuration after deployment to achieve Internet access and successful license registration.

### What to do next

Your next steps depend on what management mode you chose.

- If you chose **No** for **Enable Local Manager**, you'll use the Firewall Management Center to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409.

# Enabling Jumbo Frames

A larger MTU allows you to send larger packets. Larger packets might be more efficient for your network. See the following guidelines:

- Matching MTUs on the traffic path—We recommend that you set the MTU on all ASAv interfaces and other device interfaces along the traffic path to be the same. Matching MTUs prevents intermediate devices from fragmenting the packets.
- Accommodating jumbo frames—You can set the MTU up to 9198 bytes. The maximum is 9000 for the ASAv.

This procedure explains how to enable jumbo frames in the following environment:

**HyperFlex Cluster on the vSphere 7.0.1 > VMware vSphere vSwitch> Cisco UCS Fabric Interconnects (FI).**

### Procedure

**Step 1** Change the MTU settings of the ASAv host where you have deployed the ASAv.

**a.** Connect to the vCenter Server using the vSphere Web Client.

**b.** In the **Advanced System Settings** of your HyperFlex host, set the value of the configuration parameter—`Net.Vmxnet3NonTsoPacketGtMtuAllowed` to `1`.

**c.** Save the changes and reboot the host.

For more information, see https://kb.vmware.com/s/article/1038578.

**Step 2** Change the MTU settings of the VMware vSphere vSwitch.

**a.** Connect to the vCenter Server using the vSphere Web Client.

**b.** Edit the properties of the VMware vSphere vSwitch, and set the value of **MTU** to `9000`.

**Step 3** Change the MTU settings of the Cisco UCS Fabric Interconnects (FI).

**a.** Log in to the Cisco UCS Management console.

**b.** To Edit QoS System Class, choose **LAN** > **LAN Cloud** > **QoS System Class.** Under the **General** tab, set the value of **MTU** to 9216.

**c.** To edit your vNIC, choose **LAN** > **Policies** > **root** > **Sub-Organizations**

<your-hyperflex-org>**vNIC Templates** <your-vnic>. Under the **General** tab, set the value of **MTU** to 9000.

# Troubleshooting

This section provides you with some basic troubleshooting steps related to your Hyperflex deployment on your virtual machine.

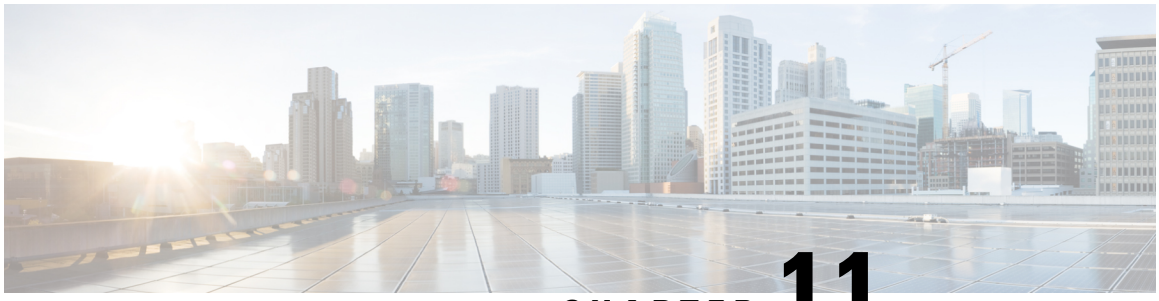
**Verify whether your virtual machine is running the HyperFlex**

If the Firewall Threat Defense Virtual appliance is installed on the HyperFlex with ESX OS, the default vSphere HA policy created by the HX post_install script is causing an error message when the Firewall Threat Defense Virtual is powered on. The error message will say:

"Power on Failures: Insufficient resources to satisfy configured failover level for vSphere HA."

**Workaround**

1. In VMware vCenter, go to **HX cluster** > **Configure** > **vSphere Availability** > **Edit Vsphere HA** > **Admission Control** > **Define host failover capacity** > **Override calculated failover capacity**.
2. Change and tune reserved failover CPU, and Memory capacity percentage.
3. Power on the Firewall Threat Defense Virtual VM.



CHAPTER 11

# Deploy the Firewall Threat Defense Virtual on Nutanix

This chapter describes the procedures to deploy the Firewall Threat Defense Virtual to a Nutanix environment.


- Overview, on page 363
- About Firewall Threat Defense Virtual Deployment On Nutanix, on page 363
- End-to-End Procedure, on page 364
- System Requirements, on page 365
- Guidelines and Limitations, on page 367
- How to Manage Secure Firewall Threat Defense Virtual Device, on page 369
- Prerequisites for Deployment on Nutanix, on page 369
- How to Deploy the Firewall Threat Defense Virtual on Nutanix, on page 369


## Overview

The Cisco Secure Firewall Threat Defense Virtual (formerly Firepower Threat Defense Virtual) brings Cisco's Secure Firewall functionality to virtualized environments, enabling consistent security policies to follow workloads across your physical, virtual, and cloud environments, and between clouds.

This chapter describes how the Firewall Threat Defense Virtual functions in the Nutanix environment with AHV hypervisor, including feature support, system requirements, guidelines, and limitations. This chapter also describes your options for managing the Firewall Threat Defense Virtual.

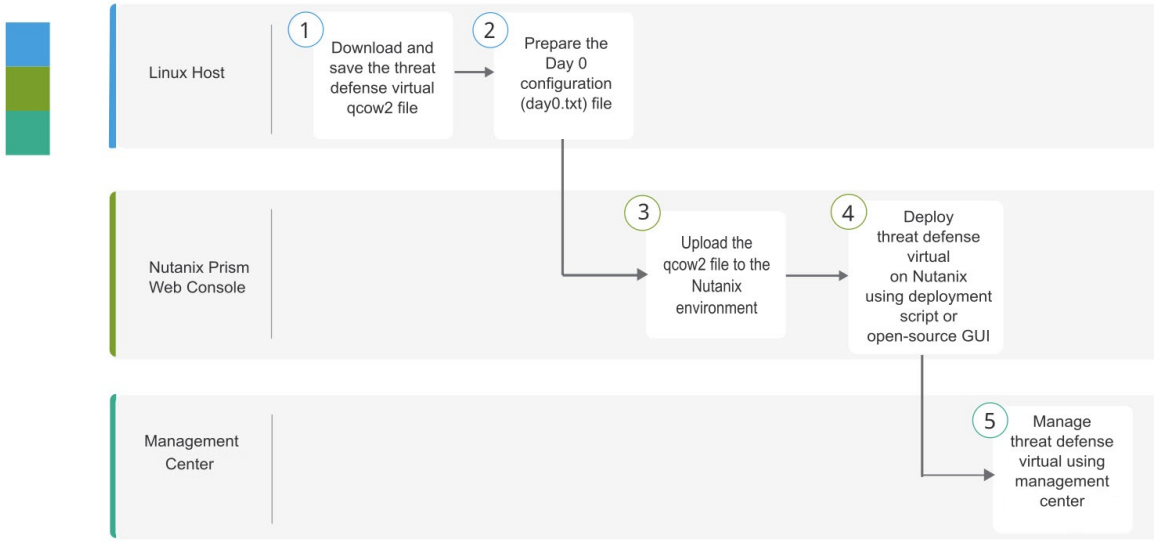
It's important that you understand your management options before you begin your deployment. You can manage and monitor the Firewall Threat Defense Virtual using the Secure Firewall Management Center. (formerly Firepower Management Center)

## About Firewall Threat Defense Virtual Deployment On Nutanix

The Nutanix Enterprise Cloud Platform is a converged, scale-out compute and storage system that is built to host and store virtual machines. You can run multiple virtual machines running unmodified OS images of Firewall Threat Defense Virtual using Nutanix AHV. Each virtual machine has private virtualized hardware: a network card, disk, graphics adapter, and so forth.

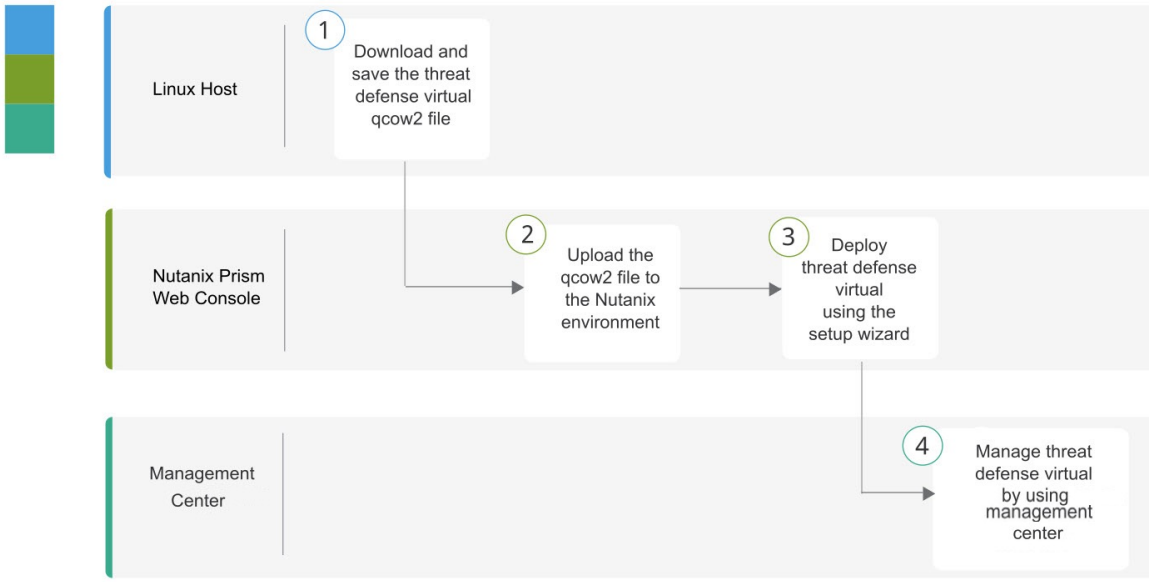
# End-to-End Procedure

The following flowchart illustrates the workflow for deploying Threat Defense Virtual on Nutanix Platform with Day-0 Configuration File.




| | Workspace | Steps |
|---|---|---|
| 1 | Linux Host | Deploy the Threat Defense Virtual on Nutanix: Download and save the threat defense virtual qcow2 file. |
| 2 | Linux Host | Deploy the Threat Defense Virtual on Nutanix: Upload the qcow2 file to the Nutanix environment. |
| 3 | Nutanix Prism Web Console | Deploy the Threat Defense Virtual on Nutanix: Prepare the Day-0 Configuration File (**Text file** > **Enter the configuration details** > **Save as day0-config.txt**. |
| 4 | Nutanix Prism Web Console | Deploy the Threat Defense Virtual on Nutanix: Deploy the Threat Defense Virtual on Nutanix. |
| 5 | Management Center | Manage Threat Defense Virtual:<br>• Using Management Center |

The following flowchart illustrates the workflow for deploying Threat Defense Virtual on Nutanix Platform without Day-0 Configuration File.




| | Workspace | Steps |
|---|---|---|
| 1 | Linux Host | Deploy the Threat Defense Virtual on Nutanix: Download and save the threat defense virtual qcow2 file. |
| 2 | Nutanix Prism Web Console | Deploy the Threat Defense Virtual on Nutanix: Upload the qcow2 file to the Nutanix environment. |
| 3 | Nutanix Prism Web Console | Deploy the Threat Defense Virtual on Nutanix: Deploy the Threat Defense Virtual on Nutanix. |
| 4 | Management Center | Manage Threat Defense Virtual:<br>• Using Management Center |

# System Requirements

### Versions

| Manager Version | Device Version |
|---|---|
| Firewall Management Center 7.0 | Firewall Threat Defense 7.0 |

See the Cisco Secure Firewall Threat Defense Compatibility Guide for the most current information about hypervisor support for the Firewall Threat Defense Virtual.

### Firewall Threat Defense Virtual Memory, vCPU, and Disk Sizing

The specific hardware used for Firewall Threat Defense Virtual deployments can vary, depending on the number of instances deployed and usage requirements. Each instance of the Firewall Threat Defense Virtual requires a minimum resource allocation—amount of memory, number of CPUs, and disk space—on the server.

| Settings | Value |
|---|---|
| Performance Tiers | **Version 7.0 and later**<br>The Firewall Threat Defense Virtual supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.<br>• FTDv5 4vCPU/8GB (100Mbps)<br>• FTDv10 4vCPU/8GB (1Gbps)<br>• FTDv20 4vCPU/8GB (3Gbps)<br>• FTDv30 8vCPU/16GB (5Gbps)<br>• FTDv50 12vCPU/24GB (10Gbps)<br>• FTDv100 16vCPU/32GB (16Gbps)<br>See the "Licensing the System" chapter in the*Secure Firewall Management Center Configuration* for guidelines when licensing your Firewall Threat Defense Virtual device.<br>**Note**<br>To change the vCPU/memory values, you must first power off the Firewall Threat Defense Virtual device. |
| Storage | 50 GB (Adjustable)<br>• Supports virtio block devices |

**Note** The minimum number of network for the Firewall Threat Defense Virtual are 4 data interfaces (management,diagnostic,outside and inside).

### Firewall Threat Defense Virtual Licenses

- Configure all license entitlements for the security services from the Firewall Management Center.
- See *Licensing the System* in the Secure firewall Management Center Configuration Guide for more information about how to manage licenses.

### Nutanix Components and Versions

| Component | Version |
|---|---|
| Nutanix Acropolis Operating System (AOS) | 5.15.5 LTS and later |
| Nutanix Cluster Check (NCC) | 4.0.0.1 |
| Nutanix AHV | 20201105.12 and later |
| Nutanix Prism Web Console | - |

# Guidelines and Limitations

### Supported Features

- Deployment Modes–Routed (Standalone), Routed (HA), Inline Tap, Inline, Passive, and Transparent
- Licensing–Only BYOL
- IPv6
- Firewall Threat Defense Virtual native HA
- Jumbo frames
- VirtIO

### Performance Optimizations

To achieve the best performance out of the Firewall Threat Defense Virtual, you can make adjustments to the both the VM and the host. See Virtualization Tuning and Optimization on Nutanix for more information.

**Receive Side Scaling**—The Firewall Threat Defense Virtual supports Receive Side Scaling (RSS), which is a technology utilized by network adapters to distribute network receive traffic to multiple processor cores. Supported on Version 7.0 and later. See Multiple RX Queues for Receive Side Scaling (RSS) for more information.

### Snort

- If you are observing abnormal behavior such as Snort taking a long time to shut down, or the VM being slow in general or when a certain process is executed, collect logs from the Firewall Threat Defense Virtual and the VM host. Collection of overall CPU usage, memory, I/O usage, and read/write speed logs will help troubleshoot the issues.
- High CPU and I/O usage is observed when Snort is shutting down. If a number of Firewall Threat Defense Virtual instances have been created on a single host with insufficient memory and no dedicated CPU, Snort will take a long time to shut down which will result in the creation of Snort cores.

### Unsupported Features

- Firewall Threat Defense Virtual on Nutanix AHV does not support hot plug of interface. Do not try to Add/Remove interface when the Firewall Threat Defense Virtual is powered on.
- Nutanix AHV does not support SR-IOV or DPDK-OVS.

**Note** Nutanix AHV supports in-guest DPDK using VirtIO. For more information, refer DPDK support on AHV.

### General Guidelines

- Requires two management interfaces and two data interfaces to boot. Supports a total of 11 interfaces.

**Note**

- The Firewall Threat Defense Virtual default configuration puts the management interface, diagnostic interface, and inside interface on the same subnet.
- When you are modifying the network interfaces, you must turn off the Firewall Threat Defense Virtual device.

- The default configuration for the Firewall Threat Defense Virtual assumes that you put both the management (management and diagnostic) and inside interfaces on the **same subnet**, and the management address uses the inside address as its gateway to the Internet (going through the outside interface).
- The Firewall Threat Defense Virtual must be powered up on firstboot with at least four interfaces. Your system will not deploy without four interfaces.
- The Firewall Threat Defense Virtual supports a total of 11 interfaces—1 management interface, 1 diagnostic interface, and a maximum of 9 network interfaces for data traffic. The interface-to-network assignments must be ordered as follows:
  1. Management interface (required)
  2. Diagnostic interface (required)
  3. Outside interface (required)
  4. Inside interface (required)
  5. 5–11 Data interfaces (optional)

**Note** The minimum number of network for the Firewall Threat Defense Virtual are 4 data interfaces.

- For the console access, terminal server is supported through telnet.
- The following are the supported vCPU and memory parameters:

| CPUs | Memory | Firewall Threat Defense Virtual Platform Size |
|---|---|---|
| 4 | 8 GB | 4vCPU/8GB (default) |
| 8 | 16 GB | 8vCPU/16GB |
| 12 | 24 GB | 12vCPU/24GB |
| 16 | 32 GB | 16vCPU/32GB |

- See the following concordance of Network Adapter, Source Networks and Destination Networks for the Firewall Threat Defense Virtual interfaces:

| Network Adapter | Source Network | Destination Network | Function |
|---|---|---|---|
| vnic0* | Management0-0 | Management0/0 | Management |

| Network Adapter | Source Network | Destination Network | Function |
|---|---|---|---|
| vnic1 | Diagnostic | Diagnostic | Diagnostic |
| vnic2* | GigabitEthernet0-0 | GigabitEthernet0/0 | Outside |
| vnic3* | GigabitEthernet0-1 | GigabitEthernet0/1 | Inside |
| *Attach to the same subnet. | | | |

### Related Documentation

- Nutanix Release Notes
- Nutanix Field Installation Guide
- Hardware Support on Nutanix

# How to Manage Secure Firewall Threat Defense Virtual Device

You can manage your Secure Firewall Threat Defense Virtual device using the following:

## Secure Firewall Management Center

If you are managing large numbers of devices, or if you want to use the more complex features and configurations that the Firewall Threat Defense allows, use the Firewall Management Center to configure your devices.

# Prerequisites for Deployment on Nutanix

- Download the Firewall Threat Defense Virtual qcow2 file from Cisco.com: https://software.cisco.com/download/navigator.html

**Note** A Cisco.com login and Cisco service contract are required.

- Review the Overview, on page 363 chapter.
- For Nutanix and System compatibility, seeCisco Secure Firewall Threat Defense Compatibility Guide .

# How to Deploy the Firewall Threat Defense Virtual on Nutanix

| Step | Task | More Information |
|---|---|---|
| 1 | Review the prerequisites. | Prerequisites for Deployment on Nutanix, on page 369 |

| Step | Task | More Information |
|---|---|---|
| 2 | Upload the Firewall Threat Defense Virtual qcow2 file to the Nutanix environment. | Upload the Firewall Threat Defense Virtual QCOW2 File to Nutanix, on page 370 |
| 3 | (Optional) Prepare a Day 0 configuration file that contains the initial configuration data that gets applied at the time a virtual machine is deployed. | Prepare the Day 0 Configuration File, on page 371 |
| 4 | Deploy the Firewall Threat Defense Virtual to the Nutanix environment. | Deploy the Firewall Threat Defense Virtual, on page 372 |
| 5 | (Optional) If you did not use a Day 0 configuration file to set up the Firewall Threat Defense Virtual, complete the setup by logging in to the CLI. | Complete the Firewall Threat Defense Virtual Setup, on page 375 |

# Upload the Firewall Threat Defense Virtual QCOW2 File to Nutanix

To deploy an Firewall Threat Defense Virtual to the Nutanix environment, you must create an image from the Firewall Threat Defense Virtual qcow2 disk file in the Prism Web Console.

### Before you begin

Download the Firewall Threat Defense Virtual qcow2 disk file from Cisco.com: https://software.cisco.com/download/navigator.html

### Procedure

**Step 1** Log in to the Nutanix Prism Web Console.

**Step 2** Click the gear icon to open the **Settings** page.

**Step 3** Click **Image Configuration** from the left pane.

**Step 4** Click **Upload Image**.

**Step 5** Create the image.

**a.** Enter a name for the image.

**b.** From the **Image Type** drop-down list, choose **DISK**.

**c.** From the **Storage Container** drop-down list, choose the desired container.

**d.** Specify the location of the Firewall Threat Defense Virtual qcow2 disk file.

You can either specify a URL (to import the file from a web server) or upload the file from your workstation.

**e.** Click **Save**.

**Step 6** Wait until the new image appears in the **Image Configuration** page.

# Prepare the Day 0 Configuration File

You can prepare a Day 0 configuration file before you deploy the Firewall Threat Defense Virtual. This file is a text file that contains the initial configuration data that gets applied at the time a virtual machine is deployed.

Keep in mind that:

- If you deploy with a Day 0 configuration file, the process allows you to perform the entire initial setup for the Firewall Threat Defense Virtual appliance.
- If you deploy without a Day 0 configuration file, you must configure System-required settings after launch; see Complete the Firewall Threat Defense Virtual Setup, on page 375 for more information.

You can specify:

- The End User License Agreement (EULA) acceptance.
- A hostname for the system.
- A new administrator password for the admin account.
- The initial firewall mode; sets the initial firewall mode, either **routed** or **transparent**.

  If you plan to manage your deployment using the local Firewall Device Manager, you can only enter **routed** for the firewall mode. You cannot configure transparent firewall mode interfaces using the Firewall Device Manager.
- The management mode; see How to Manage Secure Firewall Threat Defense Virtual Device, on page 1.

  Enter information for the Firewall Management Center fields (**FmcIp**, **FmcRegKey**, and **FmcNatId**).
- Network settings that allow the appliance to communicate on your management network.

### Procedure

**Step 1** Create a new text file using a text editor of your choice.

**Step 2** Enter the configuration details in the text file as shown in the following sample:

**Example:**

```
#Firepower Threat Defense
{
    "EULA": "accept",
    "Hostname": "ftdv-production",
    "AdminPassword": "Admin123",
    "FirewallMode": "routed",
    "DNS1": "1.1.1.1",
    "DNS2": "1.1.1.2",
    "DNS3": "",
    "IPv4Mode": "manual",
    "IPv4Addr": "10.12.129.44",
    "IPv4Mask": "255.255.0.0",
    "IPv4Gw": "10.12.0.1",
    "IPv6Mode": "disabled",
    "IPv6Addr": "",
    "IPv6Mask": "",
    "IPv6Gw": "",
```

```
    "FmcIp": "",
    "FmcRegKey": "",
    "FmcNatId": "",
    "ManageLocally":"No"
}
```

**Note**
The content of the Day 0 configuration file must be in JSON format. You must validate the text using a JSON validator tool.

**Step 3** Save the file as "**day0-config.txt**."

**Step 4** Repeat Step 1–3 to create unique default configuration files for each Firewall Threat Defense Virtual that you want to deploy.

# Deploy the Firewall Threat Defense Virtual

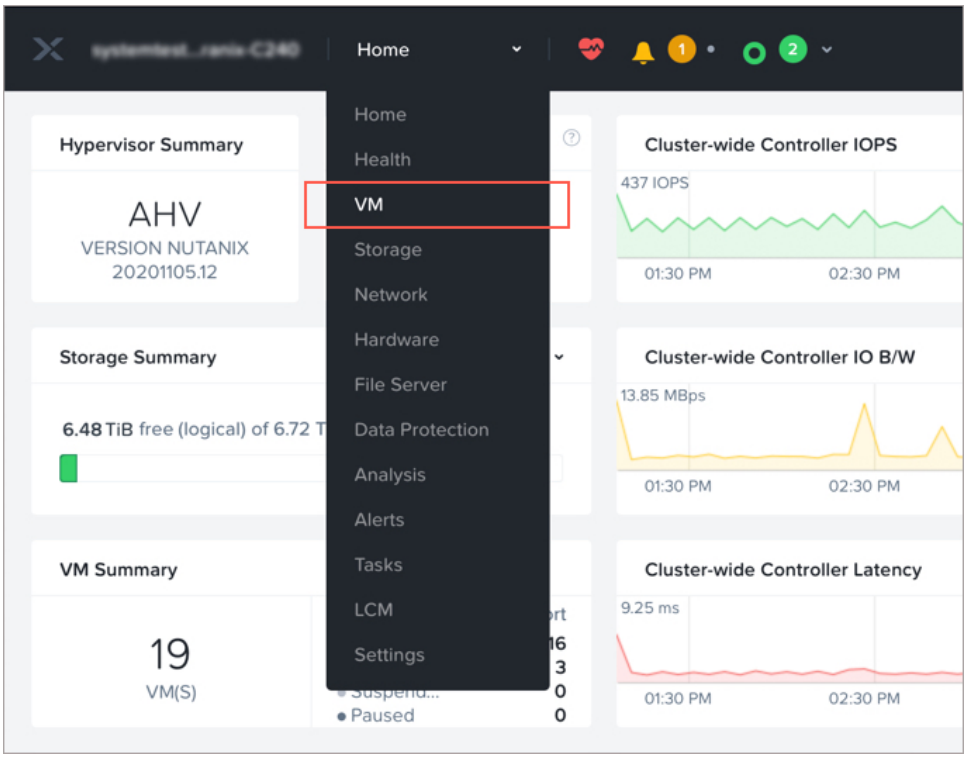
### Before you begin

Ensure that the image of the Firewall Threat Defense Virtual that you plan to deploy is appearing on the **Image Configuration** page.

### Procedure

**Step 1** Log in to the Nutanix Prism Web Console.

**Step 2** From the main menu bar, click the view drop-down list, and choose **VM**.




**Step 3** On the VM Dashboard, click **Create VM**.

**Step 4** Do the following:

**a.** Enter a name for the Firewall Threat Defense Virtual instance.

**b.** Optionally enter a description for the Firewall Threat Defense Virtual instance.

**c.** Select the timezone that you want the Firewall Threat Defense Virtual instance to use.

**Step 5** Enter the compute details.

**a.** Enter the number of virtual CPUs to allocate to the Firewall Threat Defense Virtual instance.

**b.** Enter the number of cores that must be assigned to each virtual CPU.

**c.** Enter the amount of memory (in GB) to allocate to the Firewall Threat Defense Virtual instance.

**Step 6** Attach a disk to the Firewall Threat Defense Virtual instance.

**a.** Under **Disks**, Click **Add New Disk**.

**b.** From the **Type** drop-down list, choose **DISK**.

**c.** From the **Operation** drop-down list, choose **Clone from Image Service**.

**d.** From the **Bus Type** drop-down list, choose **PCI** or **SCSI**.

**e.** From the **Image** drop-down list, choose the image that you want to use.

**f.** Click **Add**.

**Step 7** Configure at least four virtual network interfaces.

Under **Network Adapters (NIC)**, click **Add New NIC**, select a network, and click **Add**.

Repeat this process to add more network interfaces.

The Firewall Threat Defense Virtual on Nutanix supports a total of 11 interfaces—One management interface, one diagnostic interface, and a maximum of nine network interfaces for data traffic. The interface-to-network assignments must be ordered as follows:

- vnic0—Management interface (required)
- vnic1—Diagnostic interface (required)
- vnic2—Outside interface (required)
- vnic3—Inside interface (required)
- vnic4-10—Data interfaces (optional)

**Step 8** Configure affinity policy for the Firewall Threat Defense Virtual.

Under **VM Host Affinity** , click **Set Affinity**, select the hosts, and click **Save**.

Select more than one host to ensure that the Firewall Threat Defense Virtual can be run even if there is a node failure.

**Step 9** If you have prepared a Day 0 configuration file, do the following:

**a.** Select **Custom Script**.

**b.** Click **Upload A File**, and choose the Day 0 configuration file (**day0-config.txt**).

**Note**
All the other custom script options are not supported in this release.

**Step 10** Click **Save** to deploy the Firewall Threat Defense Virtual. The Firewall Threat Defense Virtual instance appears in the VM table view.

**Step 11** In the VM table view, select the newly created Firewall Threat Defense Virtual instance, and click **Power On**.

### What to do next

- If you used a Day 0 configuration file to set up the Firewall Threat Defense Virtual, your next steps depend on what management mode you chose.
  - If you chose **No** for **ManageLocally**, you'll use the Firewall Management Center to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409.
- If you did not use a Day 0 configuration file to set up the Firewall Threat Defense Virtual, complete the Firewall Threat Defense Virtual setup by logging in to the CLI. For instructions, see Complete the Firewall Threat Defense Virtual Setup, on page 375.

# Complete the Firewall Threat Defense Virtual Setup

Because the Firewall Threat Defense Virtual appliances do not have web interfaces, you must set up a virtual device using the CLI if you deployed without a Day 0 configuration file.

### Procedure

**Step 1** Open a console to the Firewall Threat Defense Virtual.

**Step 2** At the **firepower login** prompt, log in with the default credentials of **username** *admin* and the **password** *Admin123*.

**Step 3** When the Firewall Threat Defense Virtual system boots, a setup wizard prompts you for the following information that is required to configure the system:

- Accept EULA
- New admin password
- IPv4 or IPv6 configuration
- IPv4 or IPv6 DHCP settings
- Management port IPv4 address and subnet mask, or IPv6 address and prefix
- System name
- Default gateway
- DNS setup
- HTTP proxy
- Management mode

**Step 4** Review the Setup wizard settings. Defaults or previously entered values appear in brackets. To accept previously entered values, press **Enter**.

**Step 5** Complete the system configuration as prompted.

**Step 6** Verify that the setup was successful when the console returns to the # prompt.

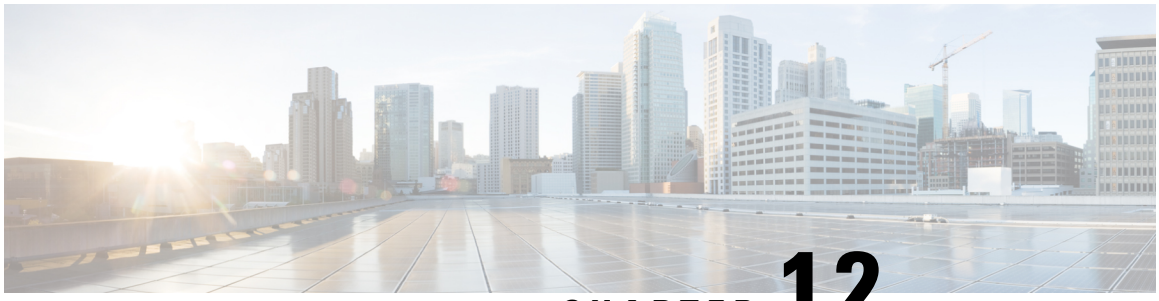
**Step 7** Close the CLI.

### What to do next

Your next steps depend on what management mode you chose.

- If you chose **No** for **Enable Local Manager**, you'll use the Firewall Management Center to manage your Firewall Threat Defense Virtual; see Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409.

See How to Manage Secure Firewall Threat Defense Virtual Device, on page 1 for an overview of how to choose your management option.



CHAPTER 12

# Deploy the Firewall Threat Defense Virtual on OpenStack


- Overview, on page 377
- End-to-End Procedure, on page 378
- Prerequisites, on page 378
- Guidelines and Limitations, on page 379
- System Requirements, on page 380
- Network Topology Example for Firewall Threat Defense Virtual on OpenStack, on page 382
- Deploy the Firewall Threat Defense Virtual, on page 383
- Upload the Firewall Threat Defense Virtual Image to OpenStack, on page 383
- Create the Network Infrastructure for OpenStack and Firewall Threat Defense Virtual, on page 384
- Deploy the Firewall Threat Defense Virtual on OpenStack, on page 385


## Overview

This guide describes how to deploy the Firewall Threat Defense Virtual in an OpenStack environment. OpenStack is a free open standard cloud computing platform, mostly deployed as infrastructure-as-a-service (IaaS) in both public and private clouds where virtual servers and other resources are made available to users.

This deployment uses a KVM hypervisor to manage virtual resources. KVM is a full virtualization solution for Linux on x86 hardware containing virtualization extensions (such as Intel VT). It consists of a loadable kernel module, kvm.ko, that provides the core virtualization infrastructure and a processor specific module, such as kvm-intel.ko.

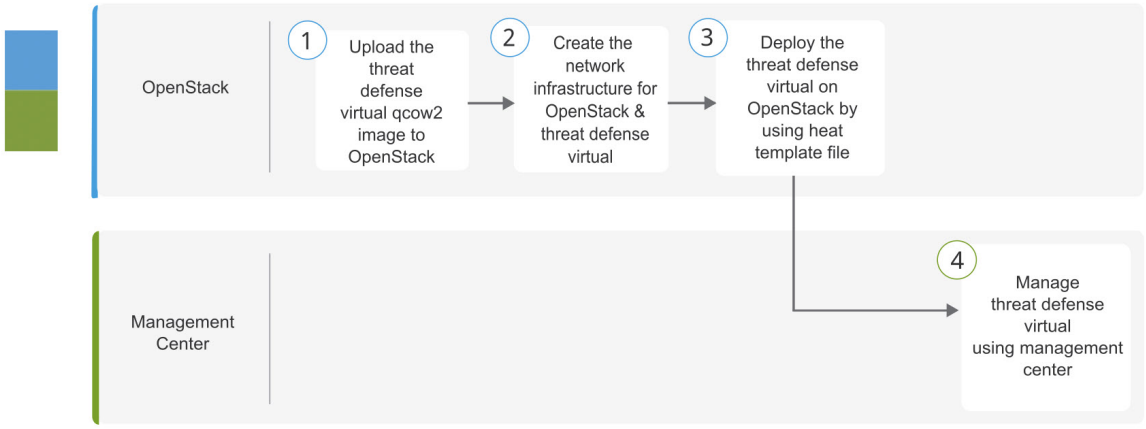
You can run multiple virtual machines running unmodified OS images using KVM. Each virtual machine has private virtualized hardware: a network card, disk, graphics adapter, and so forth.

Because devices are already supported on the KVM hypervisor, no additional kernel packages or drivers are needed to enable OpenStack support.

**Note** Firewall Threat Defense Virtual on OpenStack can be installed on any optimized multi-node environment.

# End-to-End Procedure

The following flowchart illustrates the workflow for deploying threat defense virtual on OpenStack.




| | Workspace | Steps |
|---|---|---|
| 1 | OpenStack | Deploy the Threat Defense Virtual on OpenStack: Upload the threat defense virtual image to OpenStack. |
| 2 | OpenStack | Deploy the threat defense Virtual on OpenStack: Create the network infrastructure for OpenStack and threat defense virtual. |
| 3 | OpenStack | Deploy the threat defense Virtual on OpenStack: Deploy the threat defense virtual on OpenStack by using threat defense virtual heat template file. |
| 4 | Management Center | Manage the threat defense virtual by using the Management Center |

# Prerequisites

- Get the qcow2 Firewall Threat Defense Virtual image from software.cisco.com.
- Firewall Threat Defense Virtual supports deployment on opensource OpenStack environment and Cisco VIM managed OpenStack environment.

  Set up the OpenStack environment according to the OpenStack guidelines.

  - See the opensource OpenStack document:

    Wallaby Release - https://docs.openstack.org/project-deploy-guide/openstack-ansible/wallaby/overview.html

  - See the Cisco Virtualized Infrastructure Manager (VIM) OpenStack document: Cisco Virtualized Infrastructure Manager Documentation, 4.4.3.

- A Cisco Smart Account. You can create one at Cisco Software Central.

- License the Firewall Threat Defense Virtual.
  - Configure all license entitlements for the security services from the Firewall Management Center.
  - See “Licensing” in the *Secure Firewall Management Center Admin Guide* for more information about how to manage licenses.
- Interface requirements:
  - Management interfaces (2) — One used to connect the Firewall Threat Defense Virtual to the Firewall Management Center, second used for diagnostics; cannot be used for through traffic.
  - Inside and outside interfaces — Used to connect the Firewall Threat Defense Virtual to inside hosts and to the public network.
- Communications paths:
  - Floating IPs for access into the Firewall Threat Defense Virtual.
- Minimum supported the Firewall Threat Defense Virtual version:
  - Version 7.0
- For OpenStack requirements, see System Requirements, on page 380.
- For Firewall Threat Defense Virtual system requirements, see Cisco Secure Firewall Threat Defense Compatibility Guide.

# Guidelines and Limitations

### Supported Features

The Firewall Threat Defense Virtual on OpenStack supports the following features:

- Deployment of Firewall Threat Defense Virtual on the KVM hypervisor running on a compute node in your OpenStack environment.
- OpenStack CLI
- Heat template-based deployment
- OpenStack Horizon dashboard
- IPv6
- Licensing – Only BYOL is supported
- Firewall Threat Defense Virtual management using the Firewall Management Center only.
- Drivers - virtIO and SR-IOV

### Performance Tiers for Firewall Threat Defense Virtual Smart Licensing

The Firewall Threat Defense Virtual supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.

*Table 32: Firewall Threat Defense Virtual Licensed Feature Limits Based on Entitlement*

| Performance Tier | Device Specifications (Core/RAM) | Rate Limit | RA VPN Session Limit |
|---|---|---|---|
| FTDv5 | 4 core/8 GB | 100Mbps | 50 |
| FTDv10 | 4 core/8 GB | 1Gbps | 250 |
| FTDv20 | 4 core/8 GB | 3Gbps | 250 |
| FTDv30 | 8 core/16 GB | 5Gbps | 250 |
| FTDv50 | 12 core/24 GB | 10Gbps | 750 |
| FTDv100 | 16 core/32 GB | 16Gbps | 10,000 |

See the "Licensing" chapter in the *Secure Firewall Management Center Admin Guide* for guidelines when licensing your Firewall Threat Defense Virtual device.

### Performance Optimizations

To achieve the best performance out of the Firewall Threat Defense Virtual, you can make adjustments to the both the VM and the host. See Virtualization Tuning and Optimization on OpenStack for more information.

**Receive Side Scaling**—The Firewall Threat Defense Virtual supports Receive Side Scaling (RSS), which is a technology utilized by network adapters to distribute network receive traffic to multiple processor cores. Supported on Version 7.0 and later. See Multiple RX Queues for Receive Side Scaling (RSS) for more information.

### Snort

- If you are observing abnormal behavior such as Snort taking a long time to shut down, or the VM being slow in general or when a certain process is executed, collect logs from the Firewall Threat Defense Virtual and the VM host. Collection of overall CPU usage, memory, I/O usage, and read/write speed logs will help troubleshoot the issues.
- High CPU and I/O usage is observed when Snort is shutting down. If a number of Firewall Threat Defense Virtual instances have been created on a single host with insufficient memory and no dedicated CPU, Snort will take a long time to shut down which will result in the creation of Snort cores.

### Unsupported Features

The Firewall Threat Defense Virtual on OpenStack does not support the following:

- Autoscale
- Cluster

# System Requirements

The OpenStack environment must conform to the following supported hardware and software requirements.

*Table 33: Hardware and Software Requirements for Open Source OpenStack*

| Category | Supported Versions | Notes |
|---|---|---|
| Server Hardware | UCS C240 M5 | 2 UCS servers are recommended, one each for os-controller and os-compute nodes. |
| Drivers | VIRTIO, IXGBE, and I40E | These are the supported drivers. |
| Operating System | Ubuntu Server 20.04 | This is the recommended OS on UCS servers. |
| OpenStack Version | Wallaby release | Details of the various OpenStack releases are available at:<br>https://releases.openstack.org/ |

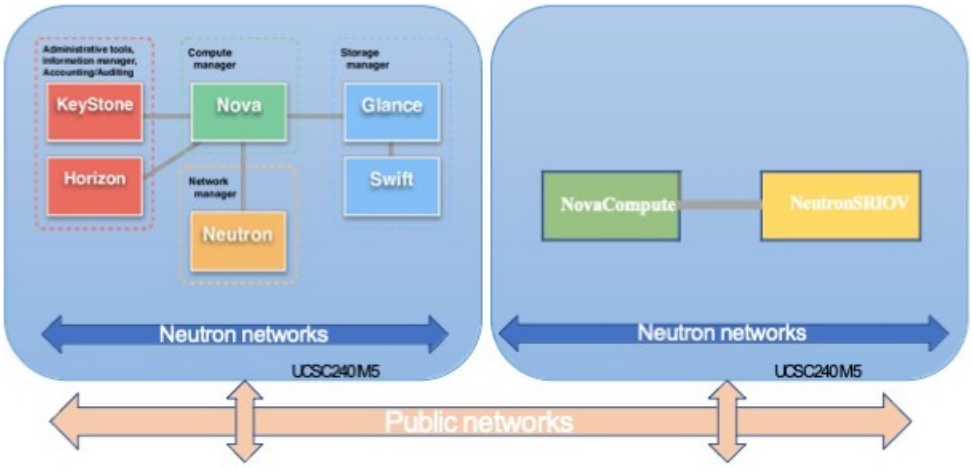
*Table 34: Hardware and Software Requirements for Cisco VIM Managed OpenStack*

| Category | Supported Versions | Notes |
|---|---|---|
| Server Hardware | UCS C220-M5/UCS C240-M4 | 5 UCS servers are recommended, three each for os-controller and Two or more for os-compute nodes. |
| Drivers | VIRTIO, IXGBE, and I40E | These are the supported drivers. |
| Cisco VIM Version | Cisco VIM 4.4.3<br>Supported on:<br>• Operating System - Red Hat Enterprise Linux 8.4<br>• OpenStack version - OpenStack 16.2 (Train Release) | See Cisco Virtualized Infrastructure Manager Documentation, 4.4.3 for more information. |

### OpenStack Platform Topology

The following figure shows the recommended topology to support deployments in OpenStack using two UCS servers.

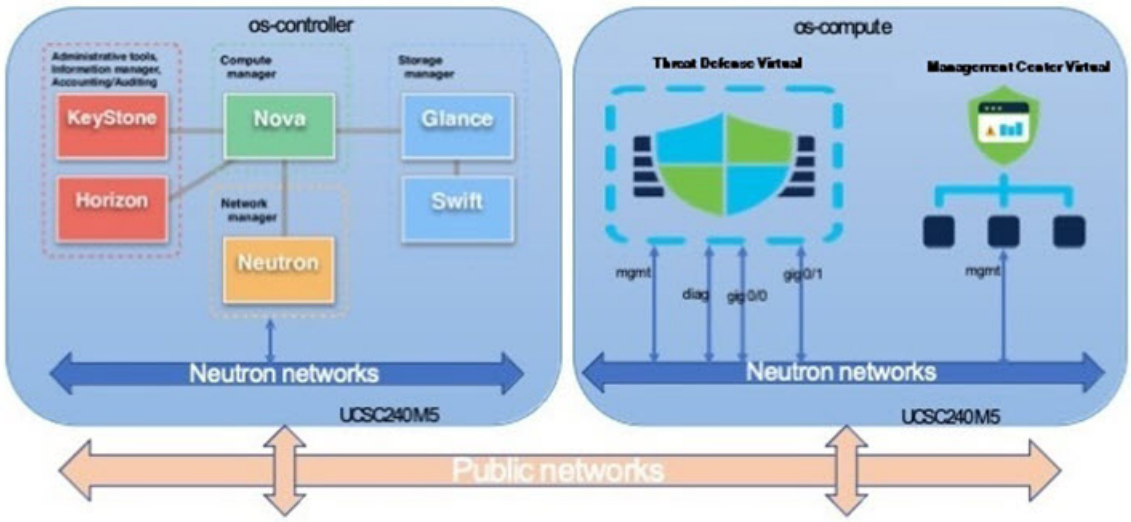
*Figure 52: OpenStack Platform Topology*




# Network Topology Example for Firewall Threat Defense Virtual on OpenStack

The following figure shows an example network topology for the Firewall Threat Defense Virtual in Routed Firewall Mode with 4 subnets configured in OpenStack for the Firewall Threat Defense Virtual (management, diagnostic, inside, and outside).

*Figure 53: Topology Example with Firewall Threat Defense Virtual and Firewall Management Center Virtual on OpenStack*



# Deploy the Firewall Threat Defense Virtual

Cisco provides sample heat templates for deploying the Firewall Threat Defense Virtual. Steps for creating the OpenStack infrastructure resources are combined in a heat template (`deploy_os_infra.yaml`) file to create networks, subnets, and router interfaces. At a high-level, the Firewall Threat Defense Virtual deployment steps are categorized into the following sections.

- Upload the Firewall Threat Defense Virtual qcow2 image to the OpenStack Glance service.
- Create the network infrastructure:
  - Network
  - Subnets
  - Router interface
- Create the Firewall Threat Defense Virtual instance:
  - Flavor
  - Security Groups
  - Floating IP
  - Instance

You can deploy the Firewall Threat Defense Virtual on OpenStack using the following steps.

# Upload the Firewall Threat Defense Virtual Image to OpenStack

Copy the Firewall Threat Defense Virtual qcow2 image to the OpenStack controller node, and then upload the image to the OpenStack Glance service.

### Before you begin

Download the Firewall Threat Defense Virtual qcow2 file from Cisco.com and put it on your Linux host:

https://software.cisco.com/download/navigator.html

**Note** A Cisco.com login and Cisco service contract are required.

### Procedure

**Step 1** Copy the qcow2 image file to the OpenStack controller node.

**Step 2** Upload the Firewall Threat Defense Virtual image to the OpenStack Glance service.

```
root@ucs-os-controller:$ openstack image create <image_name> --public --disk-
format qcow2 --container-format bare --file ./<ftdv_qcow2_file>
```

**Step 3** Verify if the Firewall Threat Defense Virtual image upload is successful.

root@ucs-os-controller:$ openstack image list

**Example:**

```
root@ucs-os-controller:$ openstack image list
+--------------------------------------+-------------------+---------+
| ID                                   | Name              | Status |+
| 06dd7975-0b6e-45b8-810a-4ff98546a39d | ftdv-7-0-image    | active |+
```

The uploaded image and it's status is displayed.

### What to do next

Create the network infrastructure using the `deploy_os_infra.yaml` template.

# Create the Network Infrastructure for OpenStack and Firewall Threat Defense Virtual

### Before you begin

Heat template files are required to create the network infrastructure and the required components for Firewall Threat Defense Virtual, such as flavor, networks, subnets, router interfaces, and security group rules:

- `deploy_os_infra.yaml`
- `env.yaml`

Templates for your Firewall Threat Defense Virtual version are available from the GitHub repository at FTDv OpenStack heat template.

**Important** Note that Cisco-provided templates are provided as open source examples, and are not covered within the regular Cisco TAC support scope. Check GitHub regularly for updates and ReadMe instructions.

### Procedure

**Step 1** Deploy the infrastructure heat template file.

**root@ucs-os-controller:$ openstack stack create <stack-name> -e <environment files name> -t <deployment file name>**

**Example:**

```
root@ucs-os-controller:$ openstack stack create infra-stack -e env.yaml -t deploy_os_infra.yaml
```

**Step 2** Verify if the infrastructure stack is created successfully.

**root@ucs-os-controller:$ openstack stack list**

### What to do next

Create the Firewall Threat Defense Virtual instance on OpenStack.

# Deploy the Firewall Threat Defense Virtual on OpenStack

Use the sample Firewall Threat Defense Virtual heat template to deploy the Firewall Threat Defense Virtual on OpenStack.

### Before you begin

A heat template is required to deploy the Firewall Threat Defense Virtual on OpenStack:

- `deploy_ftdv.yaml`

Templates for your Firewall Threat Defense Virtual version are available from the GitHub repository at FTDv OpenStack heat template.

**Important** Note that Cisco-provided templates are provided as open source examples, and are not covered within the regular Cisco TAC support scope. Check GitHub regularly for updates and ReadMe instructions.

### Procedure

**Step 1** Deploy the Firewall Threat Defense Virtual heat template file (`deploy_ftdv.yaml`) to create the Firewall Threat Defense Virtual instance.

**root@ucs-os-controller:$ openstack stack create ftdv-stack -e env.yaml-t deploy_ftdv.yaml**

**Example:**

```
+---------------------+----------------------------+
| Field               | Value                      |
+---------------------+----------------------------+
| id                  | 14624af1-e5fa-4096-bd86-c453bc2928ae |
| stack_name          | ftdv-stack                 |
| description         | FTDvtemplate               |
| updated_time        | None                       |
| stack_status        | CREATE_IN_PROGRESS         |
| stack_status_reason | Stack CREATE started       |
+---------------------+----------------------------+
```

**Step 2** Verify that your Firewall Threat Defense Virtual stack is created successfully.

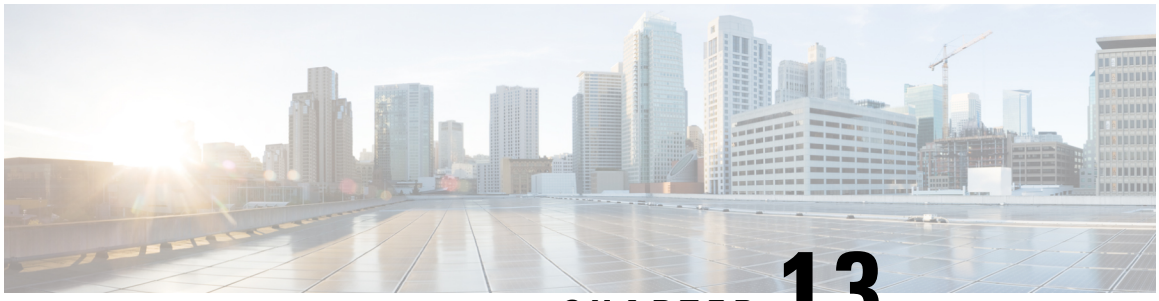
**root@ucs-os-controller:$ openstack stack list**

**Example:**

```
+--------------------------------------+-------------+----------------------------------+--------+
| ID                                   | Stack Name  | Project                          | Stack Status
```

Deploy the Firewall Threat Defense Virtual on OpenStack

```
    |
+--------------------------------------+-------------+----------------------------------+-----------------+
| 14624af1-e5fa-4096-bd86-c453bc2928ae | ftdv-stack  | 13206e49b48740fdafca83796c6f4ad5 |
CREATE_COMPLETE |
| 198336cb-1186-45ab-858f-15ccd3b909c8 | infra-stack | 13206e49b48740fdafca83796c6f4ad5 |
CREATE_COMPLETE |
+--------------------------------------+-------------+----------------------------------+-----------------+
```



CHAPTER 13

# Deploy the Threat Defense Virtual on Megaport

This guide provides information on integrating the Cisco Secure Threat Defense Virtual with Megaport by deploying the Threat Defense Virtual as a Megaport Virtual Edge (MVE).


- Overview, on page 387
- **Guidelines and Limitations to Deploy the Threat Defense Virtual as a Megaport Virtual Edge**, on page 388
- **Prerequisites to Deploy the Threat Defense Virtual as a Megaport Virtual Edge**, on page 389
- Deploy the Threat Defense Virtual as a Megaport Virtual Edge, on page 389
- Create Megaport Internet Connection, on page 390
- Connect the Threat Defense Virtual to the Management Center or Device Manager, on page 391
- Connect the Threat Defense Virtual to the Public Cloud Platforms, on page 391
- Configure High Availability (Failover), on page 392
- **Additional Resources**, on page 393


## Overview

Business critical data can originate from diverse sources ranging from multiple public clouds, private clouds, and internal servers to a remote employee's device. Securing each data entity individually is time consuming and challenging due to lack of compliance between all the data points. With the increase in such use cases, you must be able to deploy the firewall quickly and securely at your network edge in a way that provides scalability and flexibility.

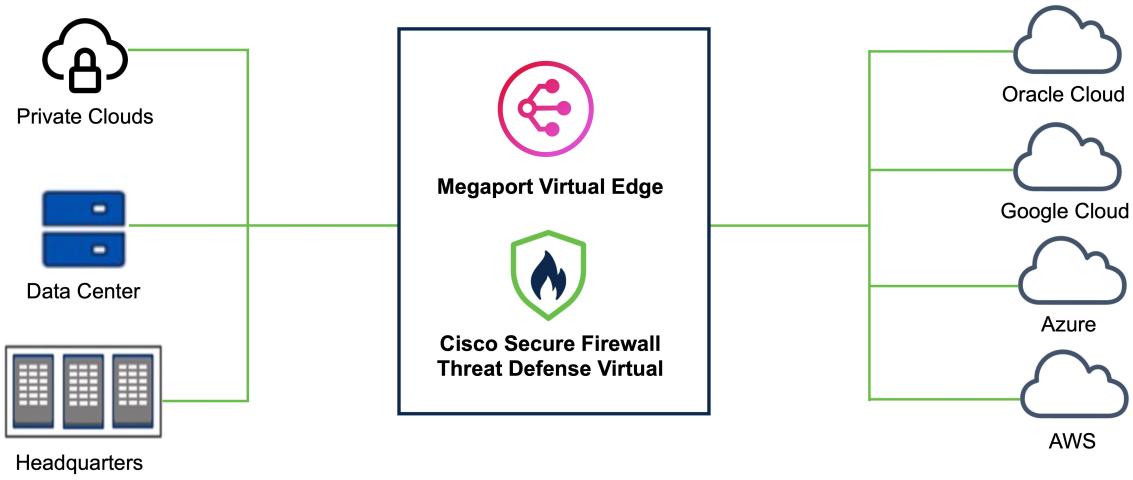
Megaport Virtual Edge (MVE) is an on-demand Network Function Virtualization (NFV) service on Megaport's Software Defined Network (SDN), with a global reach of more than 280 cloud on-ramps and more than 800 data centers. The MVE enables you to secure branch-to-cloud, cloud-to-cloud, and branch-to-branch connectivity over private networks, Layer-2 networks with dedicated bandwidth, and low latency. From the Megaport portal, you can deploy SD-WAN gateways, virtual routers, transit gateways, and virtual firewalls at the network edge.

From Secure Firewall version 7.2.8, you can deploy the Threat Defense Virtual as an MVE that enables you to create a security service chain in your hybrid and multi-cloud workflows; and deploy a single point solution for personal devices, data centers, and the closest availability zones of your cloud platforms such as AWS, Azure, and GCP. This integration reduces data transit over potentially unsecure networks and allows you to seamlessly implement your security solution without worrying about problems with robustness and scalability.

Use the Megaport portal to deploy the Threat Defense Virtual and connect all your data centers to multi-cloud applications in a single place. All the data packets are routed using Megaport's private global network. After

deployment, the Threat Defense Virtual can be managed by either the on-box Firewall Device Manager, the Cloud-Delivered Firewall Management Center, or the On-Premises Firewall Management Center.

***Figure 54: High-level Sample Topology***




# Guidelines and Limitations to Deploy the Threat Defense Virtual as a Megaport Virtual Edge

- The following MVE compute sizes are supported:
  - 4/16 - 4 CPUs, 16 GB RAM
  - 8/32 - 8 CPUs, 32 GB RAM
  - 12/48 - 12 CPUs, 48 GB RAM
- A minimum of 4 and a maximum of 5 interfaces are supported.
- High Availability (HA) is supported. For more information, see Configure High Availability (Failover).
- For initial troubleshooting if you run into any issues, see the initial boot log by using the **expert** command to enter the Threat Defense virtual expert mode, and then use the **cat firstboot.S96ovf-data.pl** command.
- Use the CLI to reboot the Threat Defense virtual instance. There is no option to reboot the Threat Defense virtual instance from the Megaport Portal.
- An interface cannot be deleted after deployment. You can only disable it.
- DHCP IPv6 addressing is not supported.
- Clustering is not supported.

# Prerequisites to Deploy the Threat Defense Virtual as a Megaport Virtual Edge

- You should have an account on the Megaport Portal with appropriate permissions to create a MVE. See Creating an Account for more information.
- (Optional) If required, you can opt for a smart license to register your firewall. Click Cisco Software Licensing and Smart Accounts for more information.



**Note** If you do not have a license, you must enable the **Evaluation** mode on your Secure Firewall Management Center or Device Manager.

# Deploy the Threat Defense Virtual as a Megaport Virtual Edge

Perform the procedure given below to deploy Cisco Secure Firewall Threat Defense Virtual as a Megaport Virtual Edge.

### Procedure

**Step 1** Open the **Megaport Portal** and click **Services**.

**Step 2** Click **Create MVE**.

**Step 3** Choose the required **MVE Location**. We recommend choosing a location that is geographically closer to your end target.

Ensure that you note the diversity zone colour of your location. We recommend deploying any other required connectors, explained later in the procedure, in the same diversity zone.

**Step 4** Click **Next**.

**Step 5** Under **MVE Configuration**, from the list of vendors, choose the required version of the **Cisco Secure Firewall Threat Defense Virtual**.

**Note**
Ensure that you choose a version that is compatible with the Firewall Management Center that you will use. The version of your Firewall Management Center should be equal to or higher than the Cisco Threat Defense Virtual version. To verify the version of your FMC, see Verify Firepower Software Versions.

**Step 6** Enter the **Cisco Secure Firewall Threat Defense Virtual Service Details**.

a) Enter a name for your Firewall in the **MVE Name** field.
b) Choose the Size, represented in CPU/RAM format,from the drop-down list depending on the needed throughput. See the Cisco Secure Firewall Threat Defense Virtual Data Sheet for more information.
c) (Optional) Enter an identifier for billing purposes in the **Service level reference** field.
d) Enter a password for your firewall in the **Admin Password** field.
e) If you want to use the on-box Device Manager, check the **Manage Locally** checkbox.

f) If you want to use the Management Center (virtual or on-prem), enter the IP address or the domain name in the **FMC IP Address** field.
g) Enter an **FMC Registration Key** for device registration. This key is a shared secret alphanumeric string (2-36 characters). Only alphanumeric characters, hyphen (-), underscore (_), and period (.) are allowed.
h) (Optional) Enter a **NAT ID**. This is an alphanumeric string that is used by the Management Center and the device during the registration process if one side does not specify an IP address. You have to specify the same NAT ID on the Management Center.
i) Add and name all the required virtual interfaces. By default, the Threat Defense Virtual has one management, one diagnostic, and two data interfaces set up. Click **+Add** to add up to 5 vNICs to an MVE in Megaport. This means that you can add one more data interface if needed.

**Note**
If you want to increase or decrease the number of vNICs on this MVE after deployment, you have to delete and recreate the entire MVE. You cannot add or delete vNICs on an MVE that is already deployed.

j) Choose the **Minimum Term** required based on your security needs and budget.

**Step 7** Click **Next**.

**Step 8** Verify all the details in the **Summary** window. Ensure that the vNICs are displayed correctly. You cannot change the vNICs after deployment.

**Step 9** Click **Add MVE**.

### What to do next

Create Megaport Internet Connection, on page 390

# Create Megaport Internet Connection

After the MVE is added, you will see a pop-up window suggesting creation of a **Megaport Internet** connection. We recommend that you create the Megaport Internet connection for your MVE to connect the deployed Threat Defense Virtual to your Management Center or use the on-box FDM.

### Procedure

**Step 1** Click **Create Megaport Internet**.

You can also create Megaport Internet later by using the **Connections** option under Threat Defense Virtual and selecting **Megaport Internet**.

**Step 2** Choose the diversity zone and the target Port. We recommend selecting the same region and diversity zone color as you did while creating the MVE.

**Step 3** Enter the required Connection details.

a) Enter a **Connection Name** for this connection.
b) Specify the **Rate Limit** that you want to put on this connection. Choosing a higher speed will incur higher cost.
c) The **VXC State** defines the initial state of the connection. By default, it is set to **Enabled**. Leave it unchanged.
d) **A-End vNIC**: This is the interface at which one end of the **Megaport Internet** will terminate. Choose the management interface (vNIC-0) from the drop-down list.

e) By default, **Preferred A-End VLAN** is set to **Untag**. You can leave it as is or you can use a unique custom VLAN ID.
f) Choose the **Minimum Term** required based on your security needs and budget.

**Step 4** Click **Next**.

**Step 5** Verify the details in the connection summary, and click **Add VXC**.

**Step 6** (Optional) Repeat the steps 1 through 5 to set up internet access for the outside interface.

**Step 7** On the left pane, click **Order**, and then click **Order Now**.

**Note**
Cisco Threat Defense Virtual can take 10-20 minutes to initialize and boot up.

# Connect the Threat Defense Virtual to the Management Center or Device Manager

After the Threat Defense Virtual is deployed, use the Public IP address of the Megaport Internet connector to register to your Secure Firewall Management Center or to access the on-box Secure Firewall Device Manager.

Perform the steps given below to find out the Public IP address of your connector.

**Procedure**

**Step 1** On the **Megaport Portal**, click the **Services** tab.

**Step 2** Search for the name of your Threat Defense Virtual instance from the filter bar.

**Step 3** Click the **gear** icon next to the Megaport Internet connector under your Threat Defense Virtual instance.

**Step 4** Click **Details** in the top menu.

**Step 5** In the **Connection Details** page, you will see the Public IPv4 and IPv6 address that you can use to connect the Threat Defense Virtual to the Management Center or Device Manager.

# Connect the Threat Defense Virtual to the Public Cloud Platforms

To secure your multicloud or hybrid cloud deployment, set up a connection between the newly deployed Threat Defense Virtual and your public cloud platforms. You can do so by creating a Virtual Cross Connect (VXC).

Perform the steps given below to create a VXC.

**Procedure**

**Step 1** On the Megaport portal, click the **Services** tab

**Step 2** Click **+ Connection** for the Threat Defense virtual instance that has to be connected to the public cloud.

**Step 3** Choose **Cloud** as the **Destination Type** to connect your cloud architecture to the Threat Defense Virtual using Megaport's secured connection.

**Step 4** Click **Next**.

**Step 5** Choose the public cloud **Provider** that has to be connected to the Threat Defense Virtual and fill in the required details.

**Step 6** Click **Next** and enter the required **Connection Details**.

# Configure High Availability (Failover)

Cisco Secure Firewall Threat Defense Virtual supports Active/Standby failover, in which one unit is the active unit and passes traffic. The standby unit does not actively pass traffic but synchronizes configuration and other state information with the active unit. When a failover occurs, the active unit fails over to the standby unit, which then becomes active.

To configure High Availability on Megaport, deploy two identical Threat Defense Virtual instances as MVEs. To understand other hardware and software requirements for the configuration of HA in the Cisco Threat Defense Virtual, see High Availability.

After configuring HA, create a Private VXC between both the Threat Defense Virtual instances. This connection is required for the failover link to work. Perform the steps given below to create a private VXC.

**Procedure**

**Step 1** Under the primary Threat Defense Virtual, click the **+Connection** icon and then click the **Private VXC** destination type.

**Step 2** Choose the secondary Threat Defense Virtual from the **Destination Port** list.

**Step 3** Enter the required Connection details.

a) Enter a **Connection Name** for your connection.
b) Specify the **Rate Limit** that you want to put on this connection. Choosing a higher speed will incur higher cost.
c) The **VXC State** defines the initial state of the connection. By default, it is set to **enabled**. Leave it unchanged.
d) For the **A-End vNIC** and **B-End vNIC**, choose the required interface from both the Threat Defense Virtual instances. This connection between these two interfaces acts as a failover link.
e) (Optional) Enter a unique **VLAN ID** for your topology.
f) Click **Next**.
g) Click **Add VXC**.
h) Finalize the order in the left pane.

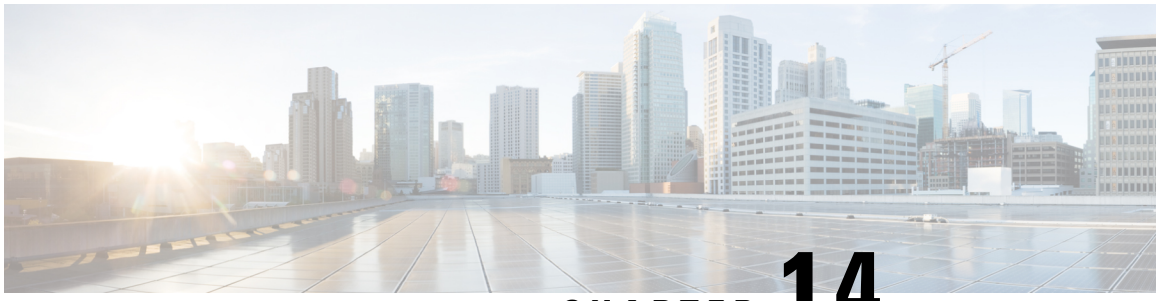
For more information on configuring HA on the Threat Defense Virtual with the Management Center or Device Manager, see the following guides:

- Management Center - High Availability

- Device Manager - High Availability (Failover).

# Additional Resources

- Configure, Verify, and Troubleshoot Firepower Device Registration



CHAPTER 14

# Deploy the Firewall Threat Defense Virtual on the Alibaba Cloud


- Overview, on page 395
- End-to-End Procedure, on page 396
- Prerequisites, on page 397
- Guidelines and Limitations, on page 398
- Configuring Policies and Device Settings, on page 400
- Configuring Alibaba Environment, on page 405
- Deploy the Threat Defense Virtual, on page 405


## Overview

Alibaba Cloud is a public cloud environment. The Firewall Threat Defense Virtual runs as a guest in the Alibaba Cloud environment.

### Alibaba Supported Instance Types

Firewall Threat Defense Virtual on Alibaba can use the following instance types:

| Network Enhanced Machine Types | | | |
|---|---|---|---|
| **Configuration** | **No of vCPUs** | **Memory (GB)** | **Maximum Interfaces Supported** |
| ecs.g5ne.xlarge | 4 | 16 | 4 |
| ecs.g5ne.2xlarge | 8 | 32 | 6 |
| ecs.g5ne.4xlarge | 16 | 64 | 8 |

**Note** Firewall Threat Defense Virtual needs a minimum of four interfaces (ENIs) to support the instance.

**Note** We do not support resizing the instance type and deploying the Firewall Threat Defense Virtual. You can only deploy a Firewall Threat Defense Virtual with a different instance size through a new deployment.

### Network Requirement

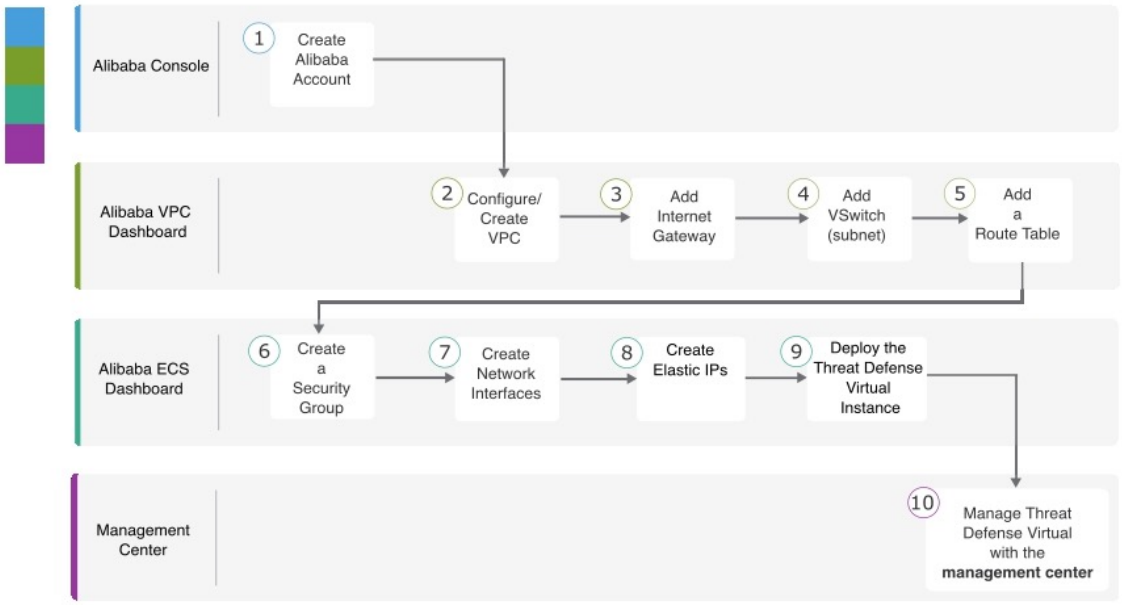
- You can create one VPC with four Vswitch (Subnet) for basic Firewall Threat Defense Virtual support.
- Management Vswitch must be available in the same zone in which instance is being deployed, otherwise, you have to create it.

### Related Documentation

For more information on instance types and their configuration, see Alibaba Cloud

# End-to-End Procedure

See the following tasks to deploy the Firewall Threat Defense Virtual on your Alibaba.




| | Workspace | Steps |
|---|---|---|
| 1 | Alibaba Console | https://marketplace.alibabacloud.com/.Create a user account in Alibaba console. |
| 2 | Alibaba VPC Dashboard | Creating the VPC, on page 400: Create and configure a VPC that is dedicated to your Alibaba account. |
| 3 | Alibaba VPC Dashboard | Adding the Internet Gateway, on page 401: Add an Internet gateway to connect your VPC to the Internet. |

| | Workspace | Steps |
|---|---|---|
| 4 | Alibaba VPC Dashboard | Adding vSwitch, on page 401: Add VSwitch (subnets) to your VPC. |
| 5 | Alibaba VPC Dashboard | Adding a Route Table, on page 402: Attach a route table to the gateway you configured for your VPC. |
| 6 | Alibaba ECS Dashboard | Creating a Security Group, on page 403: Create a security group with rules specifying allowed protocols, ports and source IP ranges. |
| 7 | Alibaba ECS Dashboard | Creating Network Interfaces, on page 403: Create network interfaces for the threat defense virtual using static IP addresses. |
| 8 | Alibaba ECS Dashboard | Creating Elastic IP Address, on page 404: Elastic IPs are reserved public IPs that are used for remote access to the threat defense virtual as well as other instances. |
| 9 | Management Center or Device Manager | Deploy the Threat Defense Virtual, on page 405: Deploy the threat defense virtual from the Alibaba portal. |
| 10 | Management Center | Manage threat defense virtual:<br>• Managing the Firepower Threat Defense Virtual with the Firepower Management Center |

# Prerequisites

- An Alibaba account. You can create one at https://www.alibaba.com/.
- An SSH client (for example, PuTTY on Windows or Terminal on Macintosh) is required to access the Firewall Threat Defense Virtual console.
- A Cisco Smart Account. You can create one at Cisco Software Central https://software.cisco.com/.
- License the Firewall Threat Defense Virtual.
  - Configure all license entitlements for the security services from the Firewall Management Center Virtual.
  - See “Licensing the Secure Firewall System” in the Secure Firewall Management Center Configuration Guide for more information about how to manage licenses.
- Firewall Threat Defense Virtual interface requirements:
  - Management interfaces (1)—Used to connect the Firewall Threat Defense Virtual to the Firewall Management Center Virtual,
  - Second Interface is used for diagnostics; can’t be used for through traffic.

    In version 6.7 and later, you can optionally configure a data interface for FMC management instead of the Management interface. The Management interface is a prerequisite for data interface

management, so you still need to configure it in your initial setup. FMC access from a data interface isn't supported in High Availability deployments.

For more information about configuring a data interface for FMC access, see the **configure network management-data-interface** command in the FTD command reference.

- Traffic interfaces (2)—Used to connect the Firewall Threat Defense Virtual to inside hosts and to the public network.

- Communication Paths:
  - Public and elastic IPs for access into the Firewall Threat Defense Virtual.

### Supported Software Platforms

The Firewall Threat Defense Virtual Auto Scale solution is a software version agnostic and it is applicable to the Firewall Threat Defense Virtual devices managed by the management center. For information about Cisco software and hardware compatibility, including operating system and hosting environment requirements, see Cisco Secure Firewall Threat Defense Compatibility Guide.

- The Firewall Management Center Virtual Compatibility Guide table lists the compatibility and virtual hosting environment requirements for the Firewall Management Center Virtual on Alibaba.
- The Cisco Secure Firewall Threat Defense Compatibility Guide table lists the compatibility and virtual hosting environment requirements for the Firewall Threat Defense Virtual on Alibaba.

# Guidelines and Limitations

### Supported Features

- Basic Product Bringup
- Day-0 Configuration
- SSH using Public Key or Password.
- Alibaba UI Console to access Firewall Threat Defense Virtual for any debugging purpose.
- Alibaba UI Stop/Restart
- Instance Type Supported: ecs.g5ne.xlarge, ecs.g5ne.2xlarge, and ecs.g5ne.4xlarge.
- Hyperthreading
- Bring Your Own License (BYOL) License Support.

### Performance Tiers for Firewall Threat Defense Virtual Smart Licensing

The supports performance-tiered licensing that provides different throughput levels and VPN connection limits based on deployment requirements.

*Table 35: Licensed Feature Limits Based on Entitlement*

| Performance Tier | Device Specifications (Core/RAM) | Rate Limit | RA VPN Session Limit |
|---|---|---|---|
| FTDv5, 100Mbps | 4 core/8 GB | 100Mbps | 50 |
| FTDv10, 1Gbps | 4 core/8 GB | 1Gbps | 250 |
| FTDv20, 3Gbps | 4 core/8 GB | 3Gbps | 250 |
| FTDv30, 5Gbps | 8 core/16 GB | 5Gbps | 250 |
| FTDv50, 10Gbps | 12 core/24 GB | 10Gbps | 750 |
| FTDv, 16Gbps | 16 core/34 GB | 16Gbps | 10,000 |

- BYOL (Bring Your Own License) using a Cisco Smart License Account.

See the "Licensing the Threat Defense Virtual System" chapter in the *Threat Defense Virtual Management Center Configuration* for guidelines when licensing your Firewall Threat Defense Virtual device.

### Performance Optimizations

To achieve the best performance out of the Firewall Threat Defense Virtual, you can make adjustments to the both the VM and the host. See Virtualization Tuning and Optimization on Alibaba Cloud for more information.

**Receive Side Scaling**—The Firewall Threat Defense Virtual supports Receive Side Scaling (RSS), which is a technology utilized by network adapters to distribute network receive traffic to multiple processor cores. Supported on Version 7.0 and later. See Multiple RX Queues for Receive Side Scaling (RSS) for more information.

### Unsupported Features

- FDM
- High Availability Functionality
- Autoscale
- IPv6
- SR-IOV

### Limitations

- Transparent, inline, and passive modes are not supported in version 7.2 release.
- East-West Traffic is not supported in Alibaba.
- Jumbo Frames is not supported as its availability is limited to a few instance types from Alibaba. For more information, see Alibaba Cloud.

**Note** Firewall Threat Defense Virtual must have four interfaces to launch.

### Licensing

- BYOL (Bring Your Own License) using a Cisco Smart License Account is supported.

# Configuring Policies and Device Settings

After you install Firewall Threat Defense Virtual and add the device to a Firewall Management Center Virtual, you can use the Firewall Management Center Virtual user interface to configure device management settings for Firewall Threat Defense Virtual running on Alibaba. You can configure and apply access control policies and other related policies to manage traffic using your Firewall Threat Defense Virtual instance.

The security policy controls the services provided by the Firewall Threat Defense Virtual, such as Next Generation IPS filtering and application filtering. You configure the security policy on the Firewall Threat Defense Virtual using the Firewall Management Center Virtual. For information about how to configure the security policy, see the *Secure Firewall Configuration Guide* or the online help in Firewall Management Center Virtual.

## Creating the VPC

A virtual private cloud (VPC) is a virtual network dedicated to your Alibaba account. It is logically isolated from other virtual networks in the Alibaba cloud. You can launch your Alibaba Cloud resources, such as the Management Center Virtual and the Threat Defense Virtual instances, into your VPC. You can configure your VPC; you can select its IP address range, create VSwitches (subnets), and configure route tables, network gateways, and security settings.

### Procedure

**Step 1** Log into https://www.alibabacloud.com and choose your region.

Alibaba Cloud is divided into multiple regions that are isolated from each other. The region is displayed in the upper right corner of your screen. Resources in one region do not appear in another region. Check periodically to make sure you are in the intended region.

**Step 2** Click **Products** > **VPC**.

**Step 3** Click **VPC Dashboard** > **Your VPCs**.

**Step 4** Click **Create VPC**.

**Step 5** Enter the following in the **Create VPC** dialog box:

a) A user-defined **Name tag** to identify the VPC.

b) An **IPv4 CIDR block** of IP addresses. CIDR (Classless Inter-Domain Routing) notation is a compact representation of an IP address and its associated routing prefix. For example, 10.0.0.0/24.

c) Select the **IPv4 CIDR block** as **Alibaba Cloud-provided IPv4 CIDR block** to enable IPv4 in the Virtual Private Cloud.

d) A **Tenancy setting** of default to ensure that instances launched in this VPC use the tenancy attribute specified at launch.

**Step 6** Click **OK** to create your VPC.

### What to do next

Add an Internet gateway to your VPC as described in the next section.

# Adding the Internet Gateway

You can add an Internet gateway (NAT Gateway) to connect your VPC to the Internet. You can route traffic for IP addresses outside your VPC to the Internet gateway.

**Before You Begin**

- Create a VPC for your Threat Defense Virtual instances.

### Procedure

**Step 1** Click **Products** > **VPC**.

**Step 2** Click **VPC Dashboard** > **Internet Gateways**, and then click **Create Internet Gateway**.

**Step 3** Enter a user-defined **Name tag** to identify the gateway and click **OK** to create the gateway.

**Step 4** Select the gateway created in the previous step.

**Step 5** Click **Bind to VPC**and select the VPC you created previously.

**Step 6** Click **OK** to bind the gateway to your VPC.

By default, the instances launched in the VPC cannot communicate with the Internet until a NAT Gateway is created and bound to the VPC.

### What to do next

Add VSwitch (subnets) to your VPC as described in the next section.

# Adding vSwitch

You can segment the IP address range of your VPC that the Threat Defense Virtual instances can be attached to. You can create vSwitch (subnets) to group instances according to security and operational needs. For the Threat Defense Virtual you need to create a vSwitch for management as well as VSwitches for traffic.

**Before You Begin**

- Create four VPCs for your Threat Defense Virtual instances. As mentioned in creating VPC section.
- Add one vSwitch (subnet) for each VPC.

**Procedure**

**Step 1** Click **Products** > **VPC**.

**Step 2** Click **VPC Dashboard** > **VSwitches**, and then click **Click vSwitch**.

**Step 3** Enter the following in the **Create vSwitch** dialog box:

a) A user-defined **Name tag** to identify the vSwitch.
b) A **VPC** to use for this vSwitch.
c) The **Zone** where this vSwitch will reside. Select **No Preference** to let Alibaba Cloud select the zone.
d) A **CIDR block** of IP addresses (IPv4). The range of IP addresses in the vSwitch must be a subset of the range of IP addresses in the VPC. Block sizes must be between a /16 network mask and a /28 network mask. The size of the vSwitch can equal the size of the VPC.

**Step 4** Click **OK** to create your vSwitch.

**Step 5** Repeat for as many vSwitchs' required. Create a separate vSwitch for management traffic and create as many vSwitchs' as needed for data traffic.

### What to do next

Add a route table to your VPC as described in the next section.

# Adding a Route Table

You can attach a route table to the gateway you configured for your VPC. You can also associate multiple subnets with a single route table, but a subnet can be associated with only one route table at a time.

**Procedure**

**Step 1** Click **Products** > **VPC**.

**Step 2** Click **VPC Dashboard** > **Route Tables**, and then click **Create Route**.

**Step 3** Enter a user-defined **Name tag** to identify the route table.

**Step 4** Select the **VPC** from the drop-down list that will use this route table.

**Step 5** Click **OK** to create your route table.

**Step 6** Select the route table that you created.

**Step 7** Click the **Routes** tab to display the route information in the details pane.

**Step 8** Click **Edit**, then click **Add another route**.

a) In the **Destination** column, enter **0.0.0.0/0**for all IPv4 traffic.
b) In the **Target** column, select your gateway.

**Step 9** Click **Save**.

### What to do next

Create a security group as described in the next section.

# Creating a Security Group

You can create a security group with rules specifying allowed protocols, ports and source IP ranges. Multiple security groups can be created with different rules which you can assign to each instance.

### Procedure

**Step 1** Click **Products** > **ECS**.

**Step 2** Click **ECS Dashboard** > **Security Groups**.

**Step 3** Click **Create Security Group**.

**Step 4** Enter the following in the **Create Security Group** dialog box:

a) A user-defined **Security Group Name** to identify the security group.

b) A **Description** for this security group.

c) The **VPC** associated with this security group.

**Step 5** Configure **Security Group Rules**:

a) Click the **Inbound Rules** tab, then click **Add Rule**.

**Note**
HTTPS and SSH access is required to manage the Management Center Virtual from outside Alibaba. You should specify the Source IP addresses accordingly. Also, if you are configuring both the Management Center Virtual and Threat Defense Virtual within the Alibaba VPC, you should allow the private IP management subnet access.

b) Click the **Outbound Rules** tab, then click **Add Rule** to add a rule for outbound traffic, or leave the defaults of **All traffic** (for **Type**) and **Anywhere** (for **Destination**).

**Step 6** Click **Create** to create your security group.

### What to do next

Create network interfaces as described in the next section.

# Creating Network Interfaces

You can create network interfaces for the Threat Defense Virtual using static IP addresses (IPv4) or DHCP. Create network interfaces (external and internal) as needed for your particular deployment.

### Procedure

**Step 1** Click **Services** > **Elastic Network Interface**.

**Step 2** Click **Network Interfaces**.

**Step 3** Click **Create Network Interface**.

**Step 4** Enter the following in the **Create Network Interface** dialog box:

a) A optional user-defined **Description** for the network interface.
b) Select a **vSwitch** from the drop-down list. Make sure to select the vSwitch of the VPC where you want to create the Threat Defense Virtual instance.
c) Enter a **Private IP** address. You can use a static IP address (IPv4) or Auto-generate (DHCP).
d) Select one or more **Security groups**. Make sure the security group has all the required ports open.

**Step 5** Click **Create network interface** to create your network interface.

**Step 6** Select the network interface that you just created.

**Step 7** Right-click and select **Modify Source/Dest. Check**.

**Step 8** Uncheck the **Enable** check box under **Source/destination check** and click **Save**.

### What to do next

Create elastic IP addresses as described in the next section.

# Creating Elastic IP Address

When an instance is created, a public IP address is associated with the instance. That public IP address (IPv4) changes automatically when you STOP and START the instance. To resolve this issue, assign a persistent public IP address to the instance using Elastic IP addressing. Elastic IP address is a reserved public IP address that are used for remotely accessing the Firewall Threat Defense Virtual as well as other instances.

### Procedure

**Step 1** Click **Products** > **Elastic Compute Service**.

**Step 2** In the **Elastic Compute Service** dashboard, click **Elastic IP** from the left-hand menu.

**Step 3** Click **Allocate Elastic IP Address**.

**Step 4** Configure EIP settings:

a) Choose the **Region** where you want to allocate the EIP.
b) Select the desired bandwidth plan for the EIP. For example, BYOL or Subscription.
c) Specify the bandwidth amount required.
d) Review your selections and click **OK** to allocate the EIP.

**Step 5** Associate the EIP with an instance:

a) After EIP allocation, go to the **Elastic IP** section in the **Elastic Compute Service** dashboard.
b) Find the EIP you created and click **Associate**.
c) Choose the ECS instance you want to associate with the EIP and click **OK**.

**Step 6** Ensure that the EIP is now listed under the associated ECS instance and verify its connectivity.

### What to do next

Deploy the Firewall Threat Defense Virtual as described in the next section.

# Configuring Alibaba Environment

To deploy the Firewall Threat Defense Virtual on Alibaba you need to configure an Alibaba VPC with your deployment-specific requirements and settings. In most situations a setup wizard can guide you through your setup. Alibaba provides online documentation where you can find useful information about the services ranging from introductions to advanced features. For more information, see Alibaba Cloud Documentation.

The Firewall Threat Defense Virtual deployment requires four network virtual private cloud (VPC) which you must create prior to deploying the Firewall Threat Defense Virtual.

The four networks VPCs are:

- Management VPC for the management subnet.
- Diagnostic VPC for the diagnostic subnet.
- Inside VPC for the inside subnet.
- Outside VPC for the outside subnet.

For greater control over your Alibaba setup, the following sections offer a guide to your VPC and EC2 configurations prior to launching the Firewall Threat Defense Virtual instances:

### Before You Begin

- Create your Alibaba Cloud account.

# Deploy the Threat Defense Virtual

### Procedure

**Step 1** Go to https://marketplace.alibabacloud.com/ and search for **`Cisco Firepower NGFW Virtual (NGFWv) - BYOL`** offering to deploy the Firewall Threat Defense Virtual.

**Note**
Alibaba is divided into multiple regions that are isolated from each other. The region is displayed in the upper right corner of your window. Resources in one region do not appear in another region. Check periodically to make sure you are in the intended region.

**Step 2** Click the offering link to open **Cisco Firepower NGFW Virtual (NGFWv) - BYOL** page.

**Step 3** Click **Choose Your Plan**. You will be redirected to the **Elastic Compute Service** page.

**Step 4** Enter the following details in the **Custom Launch** section.

- **Billing Method**: As per requirement.

**Note**

The billing method is for infrastructure on the Alibaba Cloud, which you can select according to your requirement.

- **Region**: As per requirement.
- **Network and Zone**: Select a VPC and management vSwitch, which you have previously created, from the drop-down list or use the **Create VPC** and **Create vSwitch** links to create newly.

**Step 5** Move to the **Instances and Images** page.

Under the **All Instance Types** section, perform the following:

- **Instance**: Select any of the following supported instance type - **ecs.g5ne.xlarge**, **ecs.g5ne.2xlarge**, or **ecs.g5ne.4xlarge**.
- **Image**: The latest Firewall Threat Defense Virtual marketplace version is displayed in the **Marketplace Image REC** section.
  a. Click **Reselect Image**. The Alibaba Cloud Marketplace Image dialog box is displayed with Firewall Threat Defense Virtual image details you are deploying.
  b. Choose the Firewall Threat Defense Virtual version from the drop-down list and click **Select**.

**Step 6** Move to the **Storage** section. Retain the default values and proceed.

**Step 7** Move to the **Bandwidth and Security Groups** section and perform the following:

- **ENI**
  - **Security Group**: Choose the appropriate security group.
  - **Primary ENI**: Enter the primary interface, which is the management vSwitch, as selected in the **Network and Zone** field.
  - **Secondary ENI**: Choose the secondary interface from the **Existing Secondary Interface** drop-down list or create a new secondary interface by selecting the required vSwitch.

    **Note**
    During instance launch phase, an instance can be deployed with *two* interfaces and the other *two* interfaces can be attached after deployment from ECS console.

  - **Key Pair**: Select an existing key pair from the drop-down list or create a new key pair.

**Step 8** Move to the **Advance Settings** and perform the following:

- **Instance Name**: Name of the instance as suitable.
- **User Data**: Provide the Day-0 configuration as per the requirement (Do not choose the **Enter Base64 Encoded iInformation** check box).

  **Sample Day-0 Configuration to manage Threat Defense Virtual using the Management Center:**

```
{
"AdminPassword": "<your_password>",
"Hostname": "<your_hostname>",
"ManageLocally": "No",
"FmcIp":  "<IP address of FMC>",
"FmcRegKey":"<registration_passkey>",
```

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

**Note**
If you do not provide any password in the Day-0 configuration, the default password will be the instance ID of the Firewall Threat Defense Virtual as seen on Alibaba Console or CLI.

**Step 9** Accept the **ECS Terms of Service** and click **Create Order**.
Firewall Threat Defense Virtual is launched with *two* interfaces and you can view them on ECS console.

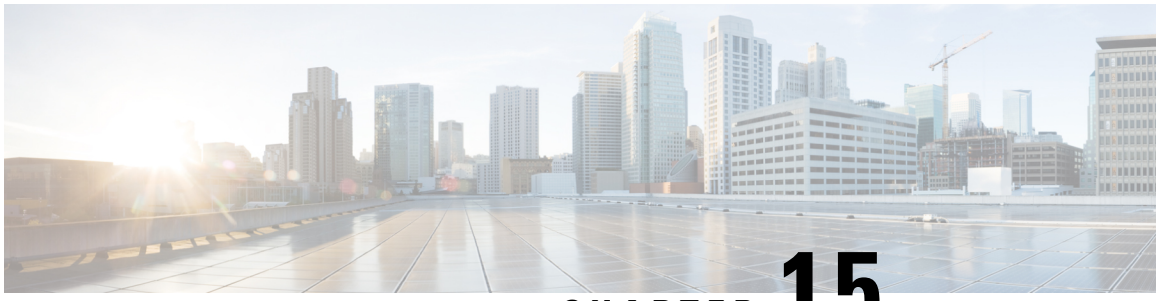
**Note**
To complete the boot process, you must configure the Firewall Threat Defense Virtual with *four* interfaces.

**Step 10** To configure the Firewall Threat Defense Virtual with two other interfaces, perform the following:

a) On the Alibaba Cloud, go to **Elastic Compute Service**.
b) Click **Elastic Network Interface** under **Network & Security** on the left pane.
c) Search for the traffic interface that is previously created.
d) Select the check box corresponding to a traffic interface, and click **Bind to Instance**. The **Bind to Instance** dialog box is displayed.
e) Enter the Firewall Threat Defense Virtual name in the **Instance** field.
f) Click **Confirm** to configure it as **eth2** interface for your instance.
g) Repeat **Step c** through **Step f** to configure **eth3** interface for your Firewall Threat Defense Virtual.

**Step 11** Click **ECS Dashboard** > **Instances**.

**Step 12** You should be able to register your Threat Defense Virtual to the Management Center Virtual after it finishes booting up.



CHAPTER 15

# Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center

This chapter describes how to deploy a standalone Firewall Threat Defense Virtual device managed with the Firewall Management Center.

**Note** This document covers the latest Firewall Threat Defense Virtual version features. If you are on an old version of software, refer to the procedures in the Firewall Management Center configuration guide for your version.


- About Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409
- Log In to the Secure Firewall Management Center, on page 410
- Register the Device with the Secure Firewall Management Center, on page 410
- Configure a Basic Security Policy, on page 413
- Access the Secure Firewall Threat Defense CLI, on page 425


## About Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center

The Secure Firewall Threat Defense Virtual is the virtualized component of the Cisco NGFW solution. The Firewall Threat Defense Virtual provides next-generation firewall services, including stateful firewalling, routing, VPN, Next-Generation Intrusion Prevention System (NGIPS), Application Visibility and Control (AVC), URL filtering, and malware defense.

You can manage the Firewall Threat Defense Virtual using the Firewall Management Center , a full-featured, multidevice manager on a separate server. The Firewall Threat Defense Virtual registers and communicates with the Firewall Management Center on the Management interface that you allocated to the Firewall Threat Defense Virtual machine.

The Firewall Threat Defense Virtual registers and communicates with the Firewall Management Center on the Management interface that you allocated to the Firewall Threat Defense Virtual machine.

For troubleshooting purposes, you can access the Firewall Threat Defense CLI using SSH on the Management interface, or you can connect to the Firewall Threat Defense from the Firewall Management Center CLI.

This guide describes how to deploy a standalone threat defense virtual device managed with the Firewall Management Center. For detailed configuration information on the Firewall Management Center, see the Management Center Administration Guide and Management Center Device Configuration Guide.

For information about installing the Firewall Management Center, see the Cisco Firepower Management Center 1600, 2600, and 4600 Hardware Installation Guide or Management Center Virtual Getting Started Guide.

# Log In to the Secure Firewall Management Center

Use the Firewall Management Center to configure and monitor the Firewall Threat Defense.

### Before you begin

For information on supported browsers, refer to the release notes for the version you are using (see https://www.cisco.com/go/firepower-notes).

### Procedure

**Step 1** Using a supported browser, enter the following URL.

**https://***fmcv_ip_address*

*fmc_ip_address* identifies the IP address or host name of the Firewall Management Center.

**Note**
https://[fmcv_ipv6_public_address] specific to IPv6

**Step 2** Enter your username and password.

**Step 3** Click **Log In**.

# Register the Device with the Secure Firewall Management Center

### Before you begin

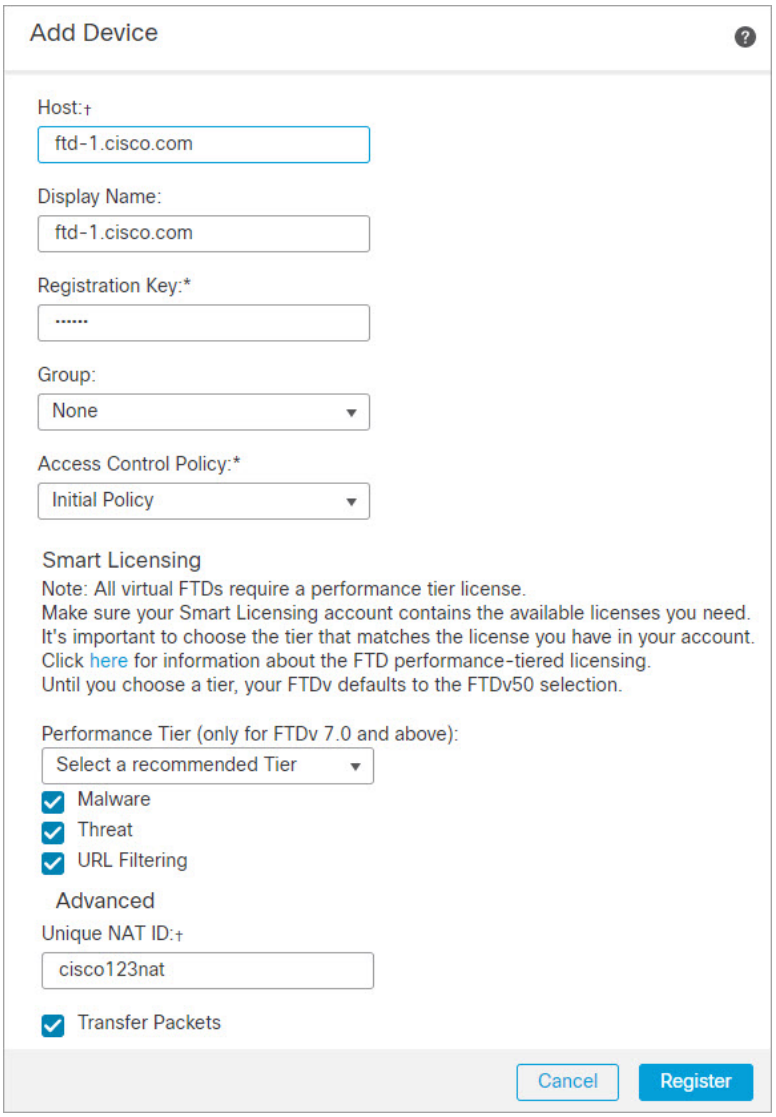
Make sure the Firewall Threat Defense Virtual machine has deployed successfully, is powered on, and has gone through its first boot procedures.

**Note** This procedure assumes that you provided the registration information for the Firewall Management Center via the day0/bootstrap script. However, all of these settings can be changed later at the CLI using **configure network** commands. See the Cisco Secure Firewall Threat Defense Command Reference.

## Procedure

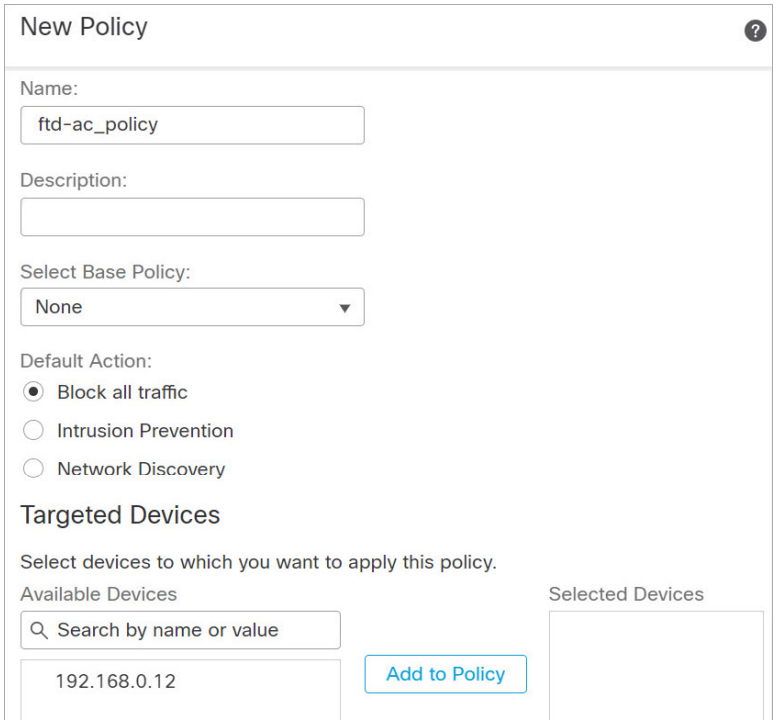
**Step 1** Choose **Devices** > **Device Management**.

**Step 2** From the **Add** drop-down list, choose **Add Device**, and enter the following parameters.




- **Host**—Enter the IP address (IPv4 and IPv6) of the device you want to add. In the case of IPv6 enable setup, you can either have Ipv4 or Ipv6 in the hostname.

- **Display Name**—Enter the name for the device as you want it to display in the Firewall Management Center.
- **Registration Key**—Enter the same registration key that you specified in the Firewall Threat Defense Virtual bootstrap configuration.
- **Domain**—Assign the device to a leaf domain if you have a multidomain environment.
- **Group**—Assign it to a device group if you are using groups.
- **Access Control Policy**—Choose an initial policy. Unless you already have a customized policy you know you need to use, choose **Create new policy**, and choose **Block all traffic**. You can change this later to allow traffic; see Configure Access Control, on page 423.




- **Smart Licensing**—Assign the Smart Licenses you need for the features you want to deploy: **Malware** (if you intend to use malware defense inspection), **Threat** (if you intend to use intrusion prevention), and **URL** (if you intend to implement category-based URL filtering).
- **Unique NAT ID**—Specify the NAT ID you specified in the Firewall Threat Defense Virtual bootstrap configuration.
- **Transfer Packets**—Allow the device to transfer packets to the Firewall Management Center. When events like IPS or Snort are triggered with this option enabled, the device sends event metadata information and packet data to the Firewall Management Center for inspection. If you disable it, only event information will be sent to the Firewall Management Center, but packet data is not sent.

**Step 3** Click **Register**, and confirm a successful registration.

If the registration succeeds, the device is added to the list. If it fails, you will see an error message. If the Firewall Threat Defense Virtual fails to register, check the following items:

- Ping—Access the Firewall Threat Defense CLI (Access the Secure Firewall Threat Defense CLI, on page 425), and ping the Firewall Management Center IP address using the following command:

**ping system** *ip_address*

If the ping is not successful, check your network settings using the **show network** command. If you need to change the Firewall Threat Defense IP address, use the **configure network** {**ipv4** | **ipv6**} **manual or DHCP** command.

- NTP—Make sure the NTP server matches the Firewall Management Center server set on the **System** > **Configuration** > **Time Synchronization** page.
- Registration key, NAT ID, and the Firewall Management Center IP address—Make sure you are using the same registration key, and if used, NAT ID, on both devices. You can set the registration key and NAT ID on the Firewall Threat Defense Virtual using the **configure manager add DONTRESOLVE<registrationkey> <NATID>** command. This command also lets you change the Firewall Management Center IP address.

# Configure a Basic Security Policy

This section describes how to configure a basic security policy with the following settings:

- Inside and outside interfaces—Assign a static IP address to the inside interface, and use DHCP for the outside interface.
- DHCP server—Use a DHCP server on the inside interface for clients.
- Default route—Add a default route through the outside interface.
- NAT—Use interface PAT on the outside interface.
- Access control—Allow traffic from inside to outside.

### Procedure

**Step 1** Configure Interfaces, on page 413
**Step 2** Configure the DHCP Server, on page 417
**Step 3** Add the Default Route, on page 418
**Step 4** Configure NAT, on page 420
**Step 5** Configure Access Control, on page 423
**Step 6** Deploy the Configuration, on page 424

# Configure Interfaces

Enable the Firewall Threat Defense Virtual interfaces, assign them to security zones, and set the IP addresses. Typically, you must configure at least a minimum of two interfaces to have a system that passes meaningful traffic. Normally, you would have an outside interface that faces the upstream router or internet, and one or more inside interfaces for your organization's networks. Some of these interfaces might be "demilitarized zones" (DMZs), where you place publically-accessible assets such as your web server.

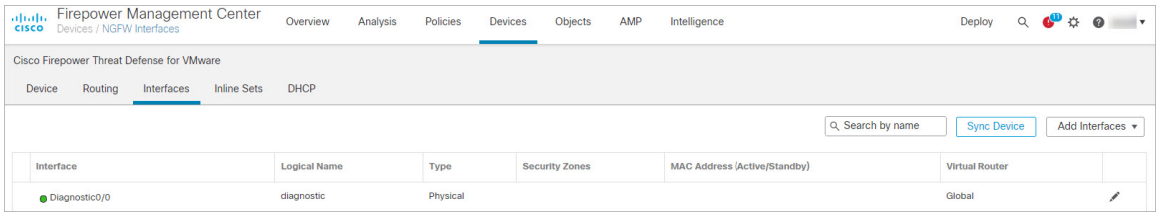
A typical edge-routing situation is to obtain the outside interface address through DHCP from your ISP, while you define static addresses on the inside interfaces.

The following example configures a routed mode inside interface with a static address and a routed mode outside interface using DHCP.

## Procedure

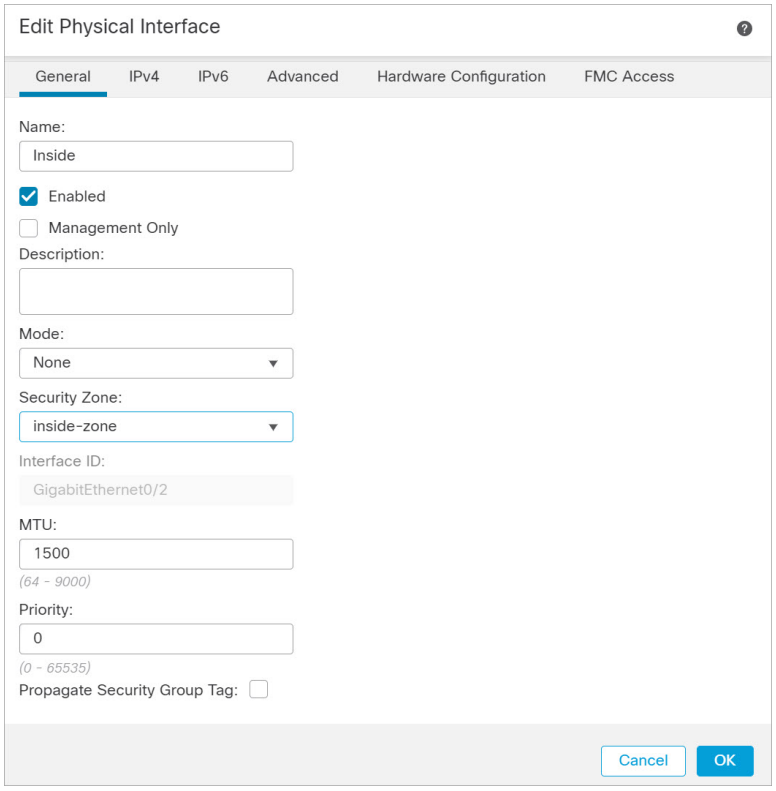
**Step 1** Choose **Devices** > **Device Management**, and click the **Edit** (✎) for the device.

**Step 2** Click **Interfaces**.




**Step 3** Click the **Edit** (✎) for the interface that you want to use for *inside*.

The **General** tab appears.




a) Enter a **Name** up to 48 characters in length.

For example, name the interface **inside**.

b) Check the **Enabled** check box.

c) Leave the **Mode** set to **None**.

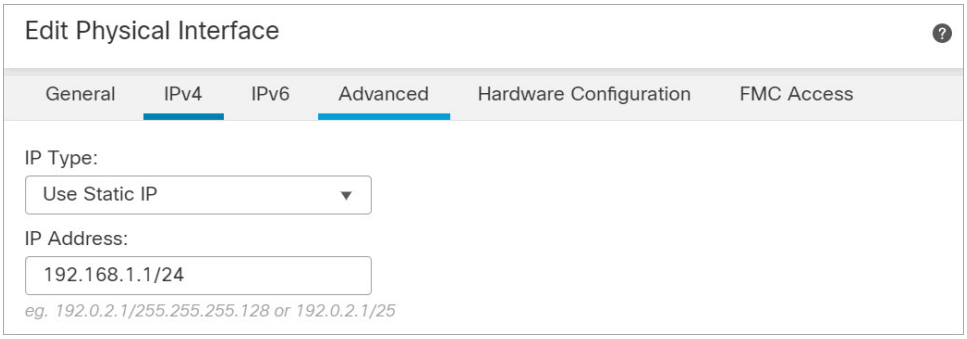
d) From the **Security Zone** drop-down list, choose an existing inside security zone or add a new one by clicking **New**.

For example, add a zone called **inside_zone**. Each interface must be assigned to a security zone and/or interface group. An interface can belong to only one security zone, but can also belong to multiple interface groups. You apply your security policy based on zones or groups. For example, you can assign the inside interface to the inside zone; and the outside interface to the outside zone. Then you can configure your access control policy to enable traffic to go from inside to outside, but not from outside to inside. Most policies only support security zones; you can use zones or interface groups in NAT policies, prefilter policies, and QoS policies.

e) Click the **IPv4** and/or **IPv6** tab.

- **IPv4**—Choose **Use Static IP** from the drop-down list, and enter an IP address and subnet mask in slash notation or DHCP option .

  For example, enter **192.168.1.1/24**

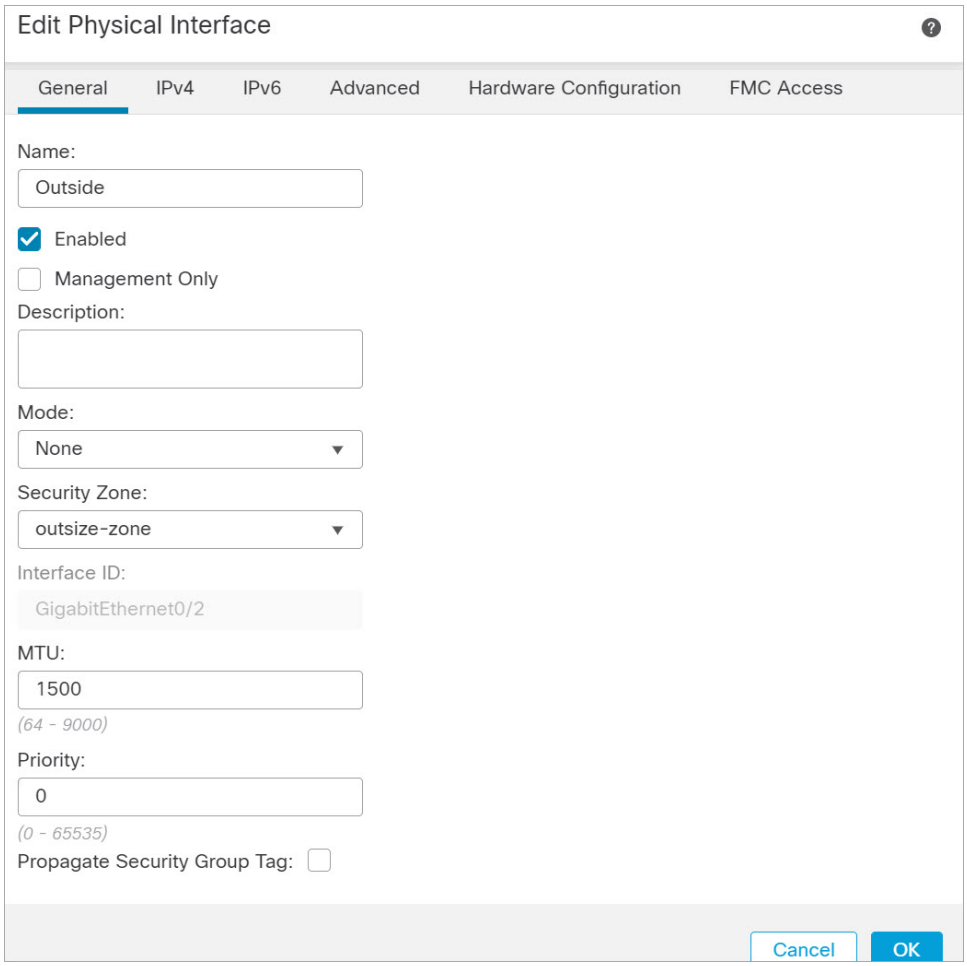



- **IPv6**—Check the **Autoconfiguration** check box for stateless auto configuration and also for IPv6 DHCP or static configuration to enable the interface.

f) Click **OK**.

**Step 4** Click the **Edit** (✎) for the interface that you want to use for *outside*.

The **General** tab appears.




a) Enter a **Name** up to 48 characters in length.

   For example, name the interface **outside**.

b) Check the **Enabled** check box.

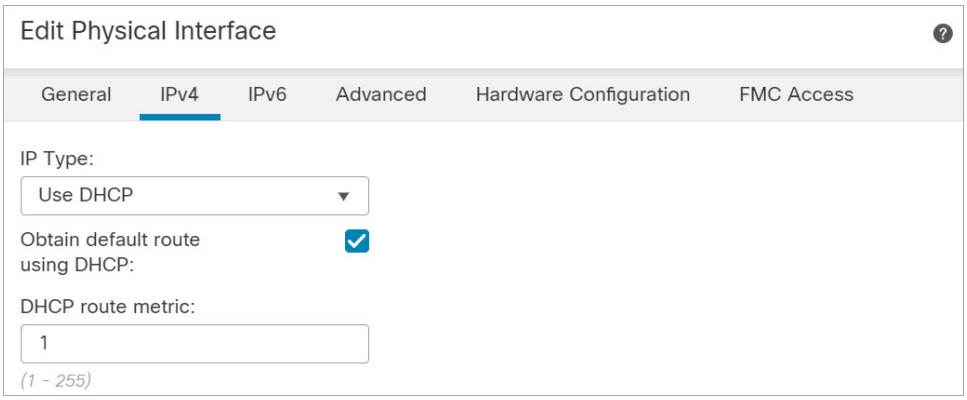
c) Leave the **Mode** set to **None**.

d) From the **Security Zone** drop-down list, choose an existing outside security zone or add a new one by clicking **New**.

   For example, add a zone called **outside_zone**.

e) Click the **IPv4** and/or **IPv6** tab.

   - **IPv4**—Choose **Use DHCP**, and configure the following optional parameters:
     - **Obtain default route using DHCP**—Obtains the default route from the DHCP server.
     - **DHCP route metric**—Assigns an administrative distance to the learned route, between 1 and 255. The default administrative distance for the learned routes is 1.




- **IPv6**—Check the **Autoconfiguration** check box for stateless auto configuration.

f) Click **OK**.

**Step 5** Click **Save**.

# Configure the DHCP Server

**Note** Skip this procedure if you are deploying to a public cloud environment such as AWS, Azure, GCP, OCI.

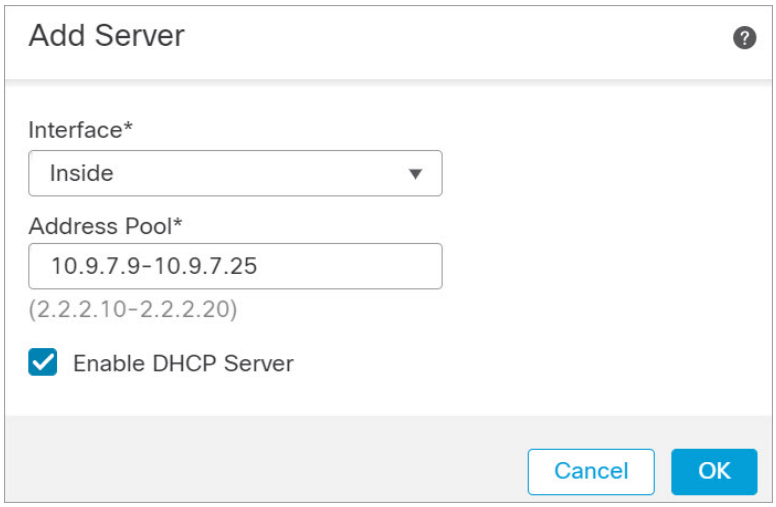
Enable the DHCP server if you want clients to use DHCP to obtain IP addresses from the Firewall Threat Defense Virtual.

## Procedure

**Step 1** Choose **Devices** > **Device Management**, and click the **Edit** (✎) for the device.

**Step 2** Choose **DHCP** > **DHCP Server**.

**Step 3** On the **Server** page, click **Add**, and configure the following options:




- **Interface**—Choose the interface from the drop-down list.
- **Address Pool**—Set the range of IP addresses from lowest to highest that are used by the DHCP server. The range of IP addresses must be on the same subnet as the selected interface and cannot include the IP address of the interface itself.
- **Enable DHCP Server**—Enable the DHCP server on the selected interface.

**Step 4** Click **OK**.

**Step 5** Click **Save**.

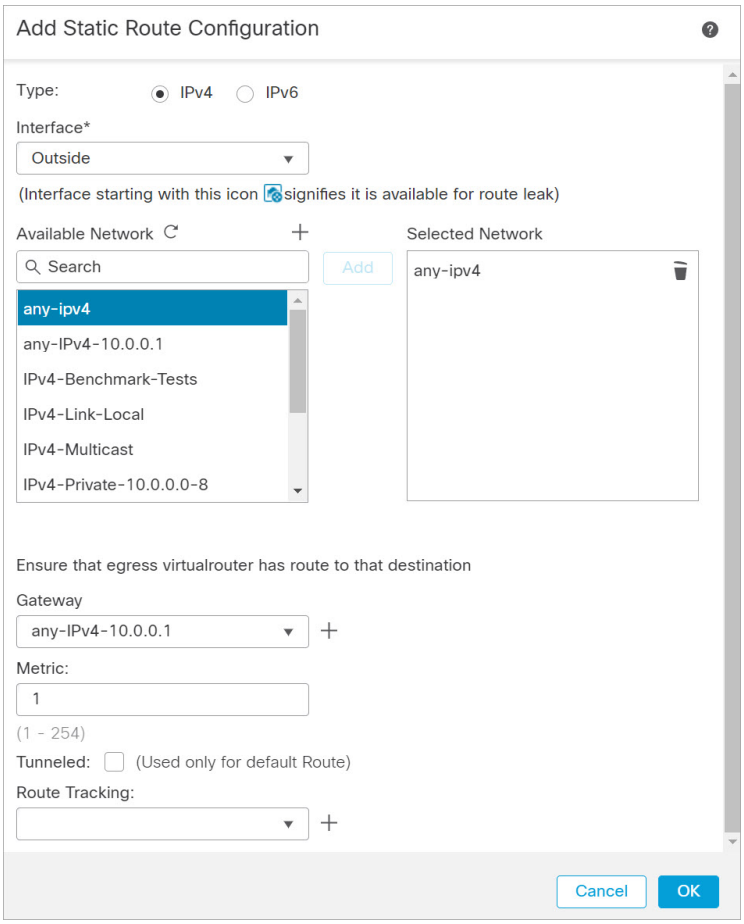
# Add the Default Route

The default route normally points to the upstream router reachable from the outside interface. If you use DHCP for the outside interface, your device might have already received a default route. If you need to manually add the route, complete this procedure. If you received a default route from the DHCP server, it will show in the **IPv4 Routes** or **IPv6 Routes** table on the **Devices** > **Device Management** > **Routing** > **Static Route** page.

## Procedure

**Step 1** Choose **Devices** > **Device Management**, and click the **Edit** (✎) for the device.

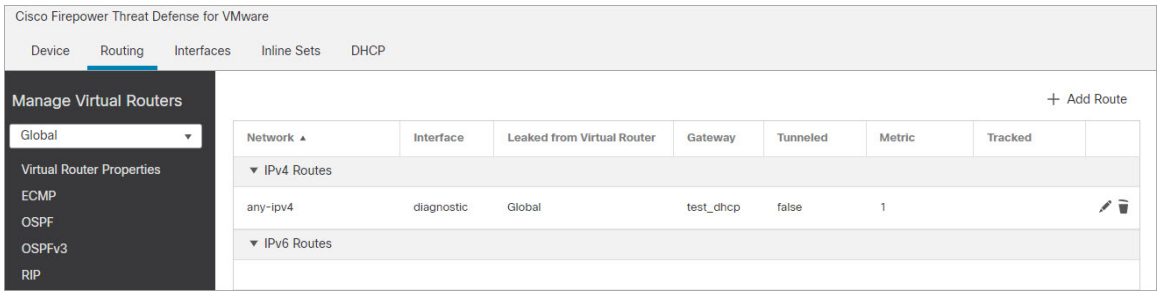
**Step 2** Choose **Routing** > **Static Route**, click **Add Route**, and set the following:




- **Type**—Click the **IPv4** or **IPv6** radio button depending on the type of static route that you are adding.
- **Interface**—Choose the egress interface; typically the outside interface.
- **Available Network**—Choose **any-ipv4** for an IPv4 default route, or **any-ipv6** for an IPv6 default route.
- **Gateway** or **IPv6 Gateway**—Enter or choose the gateway router that is the next hop for this route. You can provide an IP address or a Networks/Hosts object.
- **Metric**—Enter the number of hops to the destination network. Valid values range from 1 to 255; the default value is 1.

**Step 3** Click **OK**.

The route is added to the static route table.




**Step 4** Click **Save**.

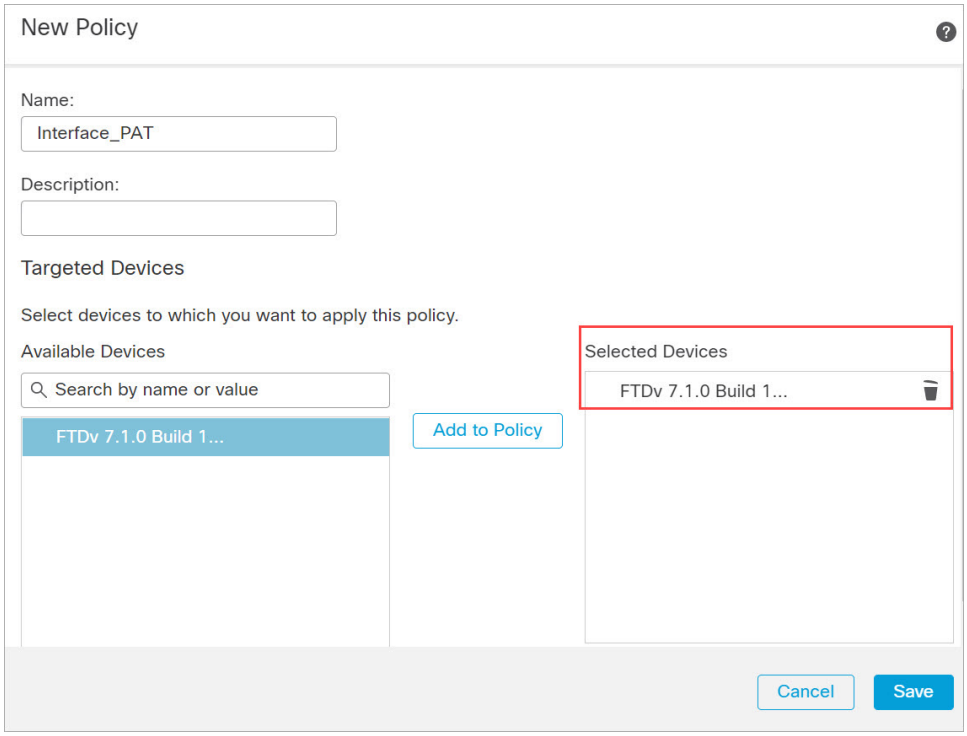
# Configure NAT

A typical NAT rule converts internal addresses to a port on the outside interface IP address. This type of NAT rule is called *interface Port Address Translation (PAT)*.

## Procedure

**Step 1** Choose **Devices** > **NAT**, and click **New Policy** > **Threat Defense NAT**.

**Step 2** Name the policy, select the device(s) that you want to use the policy, and click **Save**.

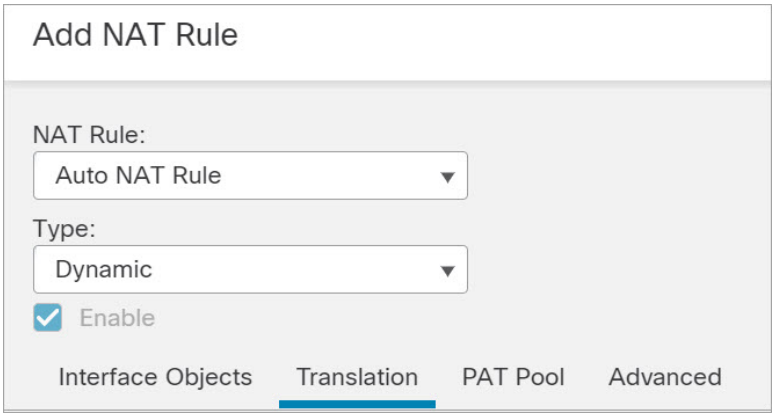



The policy is added the Firewall Management Center. You still have to add rules to the policy.

**Step 3** Click **Add Rule**.

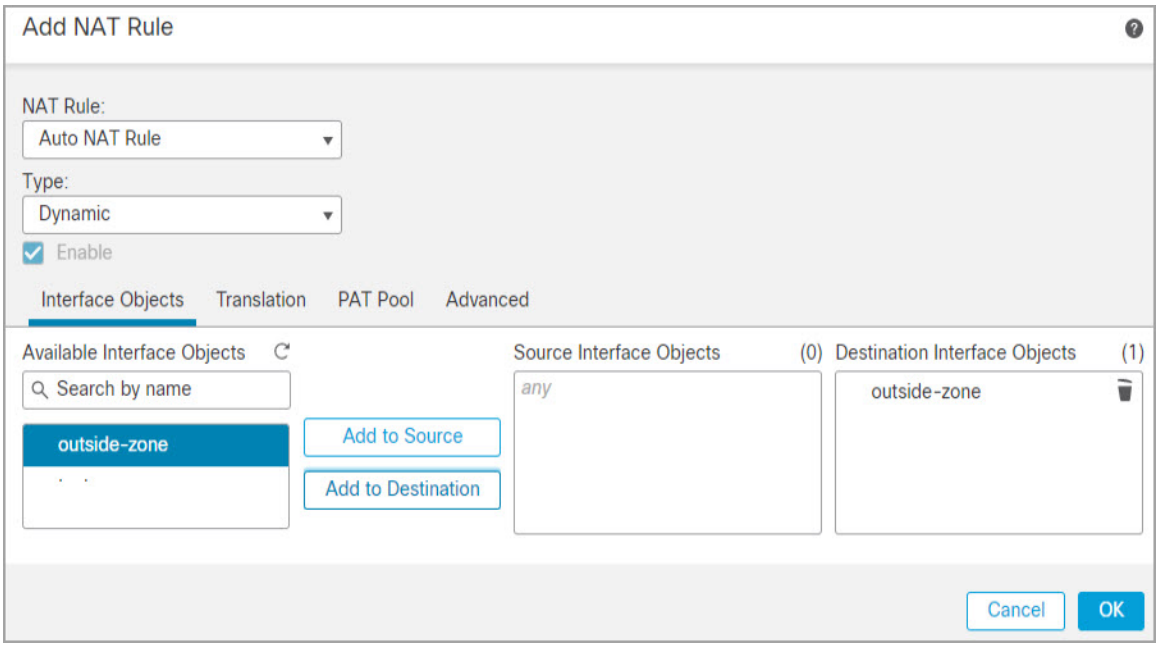
The **Add NAT Rule** dialog box appears.

**Step 4** Configure the basic rule options:

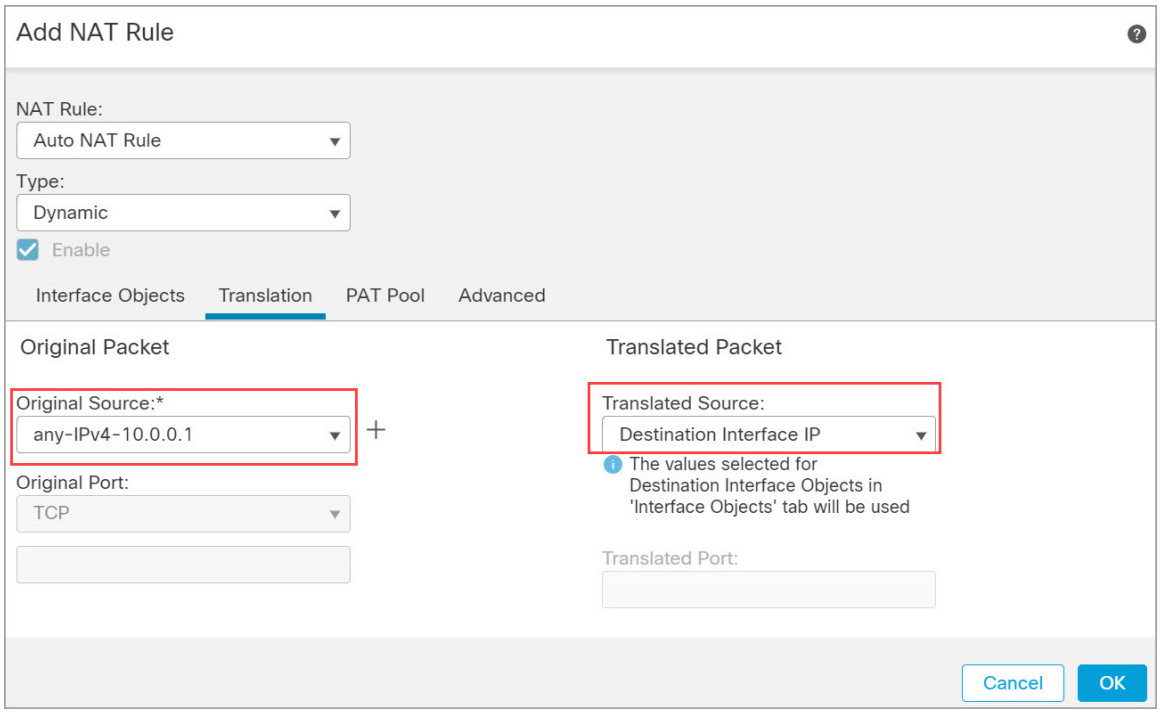



- **NAT Rule**—Choose **Auto NAT Rule**.
- **Type**—Choose **Dynamic**.

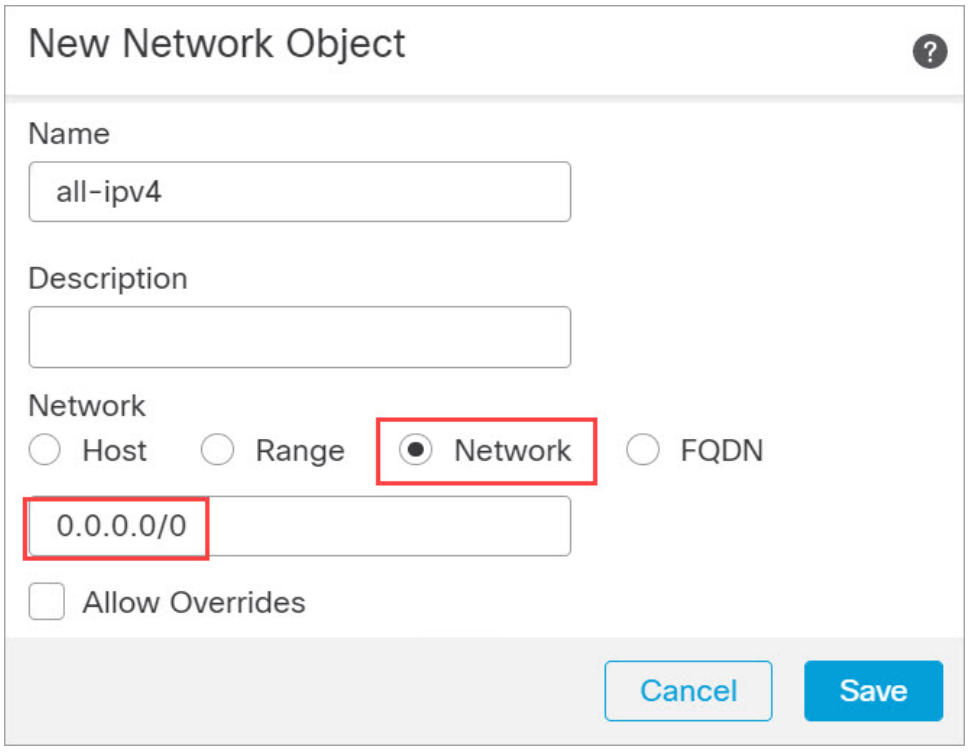
**Step 5** On the **Interface Objects** page, add the outside zone from the **Available Interface Objects** area to the **Destination Interface Objects** area.




**Step 6** On the **Translation** page, configure the following options:




- **Original Source**—Click the **Add** (+) to add a network object for all IPv4 traffic (0.0.0.0/0).



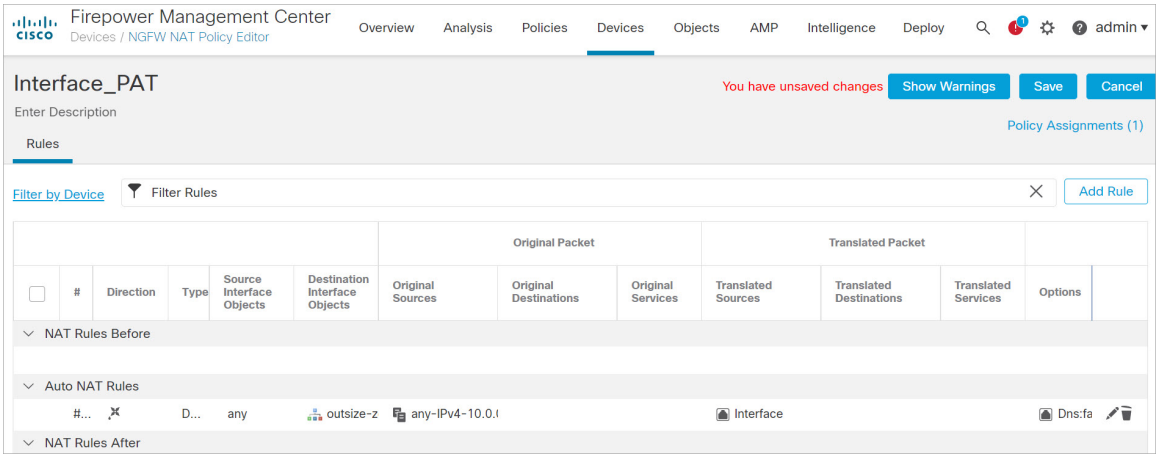

**Note**
You cannot use the system-defined **any-ipv4** object, because Auto NAT rules add NAT as part of the object definition, and you cannot edit system-defined objects.

Similarly, you can create the NAT policy with a default host network [::/0] for all IPv6 traffic.

- **Translated Source**—Choose **Destination Interface IP**.

**Step 7** Click **Save** to add the rule.

The rule is saved to the **Rules** table.




**Step 8** Click **Save** on the **NAT** page to save your changes.

# Configure Access Control

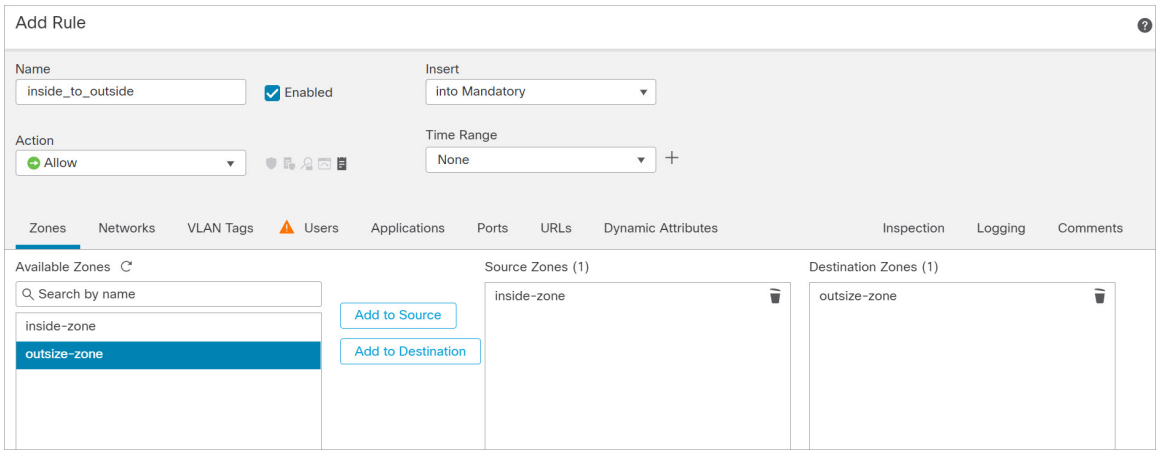
If you created a basic **Block all traffic** access control policy when you registered the Firewall Threat Defense Virtual with the Firewall Management Center, then you need to add rules to the policy to allow traffic through the device. The following procedure adds a rule to allow traffic from the inside zone to the outside zone. If you have other zones, be sure to add rules allowing traffic to the appropriate networks.

See the Firewall Management Center Configuration Guide configuration guide to configure more advanced security settings and rules.

## Procedure

**Step 1** Choose **Policy** > **Access Policy** > **Access Policy**, and click the **Edit** (✎) for the access control policy assigned to the Firewall Threat Defense.

**Step 2** Click **Add Rule**, and set the following parameters:

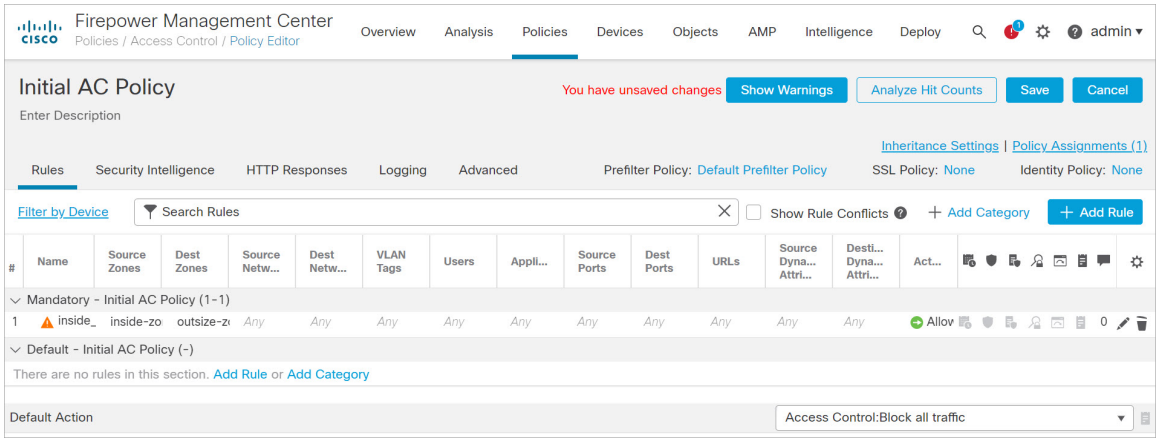



- **Name**—Name this rule, for example, **inside_to_outside**.
- **Source Zones**—Select the inside zone from **Available Zones**, and click **Add to Source**.
- **Destination Zones**—Select the outside zone from **Available Zones**, and click **Add to Destination**.

Leave the other settings as is.

**Step 3** Click **Add**.

The rule is added to the **Rules** table.



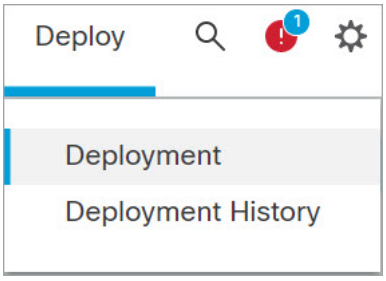

**Step 4** Click **Save**.

# Deploy the Configuration

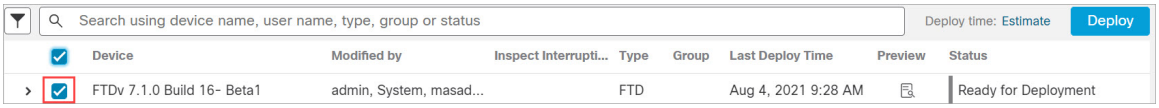
Deploy the configuration changes to the Firewall Threat Defense Virtual; none of your changes are active on the device until you deploy them.

**Procedure**

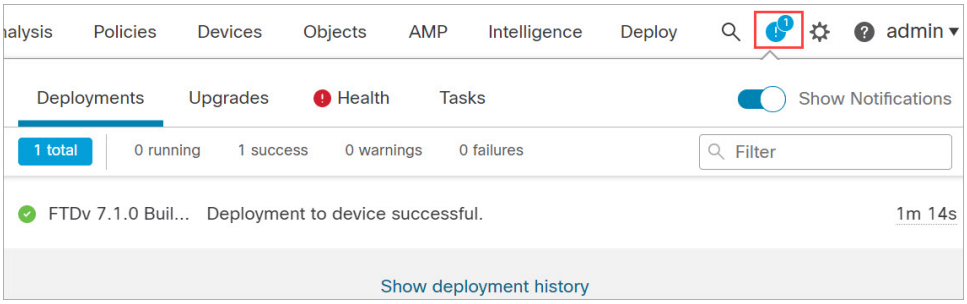
**Step 1** Click **Deploy** in the upper right.




**Step 2** Select the device in the **Deploy Policies** dialog box, then click **Deploy**.




**Step 3** Ensure that the deployment succeeds. Click the icon to the right of the **Deploy** button in the menu bar to see status for deployments.




# Access the Secure Firewall Threat Defense CLI

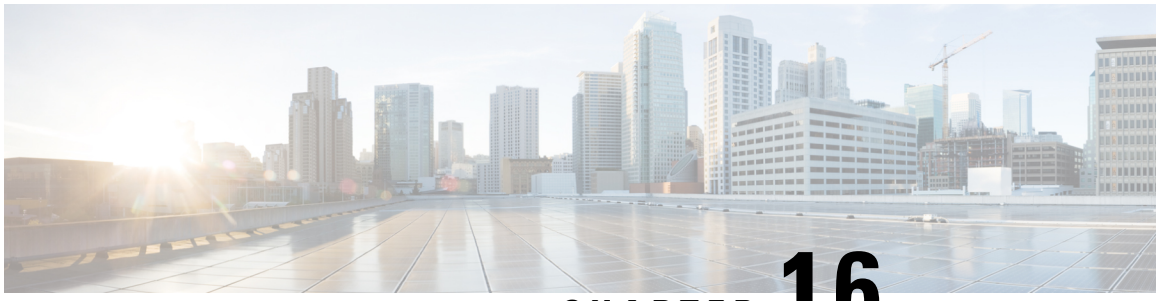
You can use the Firewall Threat Defense Virtual CLI to change management interface parameters and for troubleshooting purposes. You can access the CLI using SSH to the Management interface, or by connecting from the VMware console.

**Procedure**

**Step 1** (Option 1) SSH directly to the Firewall Threat Defense Virtual management interface IP address.

You set the management IP address when you deployed the virtual machine. Log into the Firewall Threat Defense Virtual with the **admin** account and the password you set during initial deployment.

**Step 2** (Option 2) Open the VMware console and log in with the default username **admin** account and the password you set during initial deployment.



CHAPTER 16

# Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Device Manager

This chapter describes how to deploy a standalone Firewall Threat Defense Virtual device managed with the Firewall Device Manager. To deploy a High Availability pair, see the Cisco Secure Firewall Device Manager Configuration Guide.


- About Secure Firewall Threat Defense Virtual with the Secure Firewall Device Manager, on page 427
- Initial Configuration, on page 428
- How to Configure the Device in the Secure Firewall Device Manager, on page 430


## About Secure Firewall Threat Defense Virtual with the Secure Firewall Device Manager

The Secure Firewall Threat Defense Virtual is the virtualized component of the Cisco NGFW solution. The Firewall Threat Defense Virtual provides next-generation firewall services, including stateful firewalling, routing, VPN, Next-Generation Intrusion Prevention System (NGIPS), Application Visibility and Control (AVC), URL filtering, and malware defense.

You can manage the Firewall Threat Defense Virtual using the Secure Firewall Device Manager, a web-based device setup wizard included on some of the Firewall Threat Defense models. The Firewall Device Manager lets you configure the basic features of the software that are most commonly used for small networks. It is especially designed for networks that include a single device or just a few, where you do not want to use a high-powered multiple-device manager to control a large network containing many of the Firewall Threat Defense devices.

If you are managing large numbers of devices, or if you want to use the more complex features and configurations that the Firewall Threat Defense allows, use the Firewall Management Center to configure your devices instead of the integrated Firewall Device Manager. See Managing the Secure Firewall Threat Defense Virtual with the Secure Firewall Management Center, on page 409 for more information.

For troubleshooting purposes, you can access the Firewall Threat Defense CLI using SSH on the Management interface, or you can connect to the Firewall Threat Defense from the Firewall Device Manager CLI.

# Default Configuration

The Firewall Threat Defense Virtual default configuration puts the management interface and inside interface on the same subnet. You must have Internet connectivity on the management interface in order to use Smart Licensing and to obtain updates to system databases.

Thus, the default configuration is designed so that you can connect both the Management0-0 and GigabitEthernet0-1 (inside) to the same network on the virtual switch. The default management address uses the inside IP address as the gateway. Thus, the management interface routes through the inside interface, then through the outside interface, to get to the Internet.

You also have the option of attaching Management0-0 to a different subnet than the one used for the inside interface, as long as you use a network that has access to the Internet. Ensure that you configure the management interface IP address and gateway appropriately for the network.

The Firewall Threat Defense Virtual must be powered up on firstboot with at least four interfaces:

- The first interface on the virtual machine is the management interface (Management0-0).
- The second interface on the virtual machine is reserved for internal use.
- The third interface on the virtual machine (GigabitEthernet0-0) is the outside interface.
- The fourth interface on the virtual machine (GigabitEthernet0-1) is the inside interface.

You can add up to six more interfaces for data traffic, for a total of eight data interfaces. For additional data interfaces, make sure that the Source Networks map to the correct Destination Networks, and that each data interface maps to a unique subnet or VLAN. See Configuring VMware Interfaces.

# Initial Configuration

You must complete an initial configuration to have the Firewall Threat Defense Virtual function correctly in your network, which includes configuring the addresses needed to insert the security appliance into your network and connect it to the Internet or other upstream router. You can do the initial configuration of the system in one of two ways:

- Using the Firewall Device Manager web interface (recommended). Firewall Device Manager runs in your web browser. You use this interface to configure, manage, and monitor the system.
- Using the Command Line Interface (CLI) setup wizard (optional). You can use the CLI setup wizard for initial configuration instead of Firewall Device Manager, and you can use the CLI for troubleshooting. You still use the Firewall Device Manager to configure, manage, and monitor the system; see (Optional) Launch the Firewall Threat Defense CLI Wizard.

The following topics explain how to use these interfaces to do the initial configuration of your system.

# Launch the Firewall Device Manager

When you initially log into Firewall Device Manager, you are taken through the device setup wizard to complete the initial system configuration.

### Procedure

**Step 1** Open a browser and log into Firewall Device Manager. Assuming you did not go through initial configuration in the CLI, open the Firewall Device Manager at **https://FTDv pubic IPv4 address or [FTDv IPv6 public address]**.

**Step 2** Log in with the username **admin**, password **Admin123**.

**Step 3** If this is the first time logging into the system, and you did not use the CLI setup wizard, you are prompted to read and accept the End User License Agreement and change the admin password. You must complete these steps to continue.

**Step 4** Configure the following options for the outside and management interfaces and click **Next**.

**Note**
Your settings are deployed to the device when you click **Next**. The interface will be named "outside" and it will be added to the "outside_zone" security zone. Ensure that your settings are correct.

a) **Outside Interface**—This is the data port that you connected to your gateway mode or router. You cannot select an alternative outside interface during initial device setup. The first data interface is the default outside interface.

**Configure IPv4**—The IPv4 address for the outside interface. You can use DHCP or manually enter a static IP address, subnet mask, and gateway. You can also select **Off** to not configure an IPv4 address.

**Configure IPv6**—The IPv6 address for the outside interface. You can use DHCP or manually enter a static IP address, prefix, and gateway. You can also select **Off** to not configure an IPv6 address.

b) **Management Interface**

**DNS Servers**—The DNS server for the system's management address. Enter one or more addresses of DNS servers for name resolution. The default is the OpenDNS public DNS servers. If you edit the fields and want to return to the default, click **Use OpenDNS** to reload the appropriate IP addresses into the fields.

**Firewall Hostname**—The hostname for the system's management address.

**Note**
When you configure the Firewall Threat Defense device using the device setup wizard, the system provides two default access rules for outbound and inbound traffic. You can go back and edit these access rules after initial setup.

**Step 5** Configure the system time settings and click **Next**.

a) **Time Zone**—Select the time zone for the system.
b) **NTP Time Server**—Select whether to use the default NTP servers or to manually enter the addresses of your NTP servers. You can add multiple servers to provide backups.

**Step 6** Configure the smart licenses for the system.

You must have a smart license account to obtain and apply the licenses that the system requires. Initially, you can use the 90-day evaluation license and set up smart licensing later.

To register the device now, click the link to log into your Smart Software Manager account, generate a new token, and copy the token into the edit box.

To use the evaluation license, select **Start 90 day evaluation period without registration**. To later register the device and obtain smart licenses, click the name of the device in the menu to get to the **Device Dashboard**, then click the link in the **Smart Licenses** group.

**Step 7** Click **Finish**.

### What to do next

- Configure the device using the Firewall Device Manager; see How to Configure the Device in the Secure Firewall Device Manager, on page 430.

# How to Configure the Device in the Secure Firewall Device Manager

After you complete the setup wizard, you should have a functioning device with a few basic policies in place:

- Security zones for the inside and outside interfaces.
- An access rule trusting all inside to outside traffic.
- An interface NAT rule that translates all inside to outside traffic to unique ports on the IP address of the outside interface.
- A DHCP server running on the inside interface or bridge group.

The following steps provide an overview of additional features you might want to configure. Please click the help button (**?**) on a page to get detailed information about each step.

### Procedure

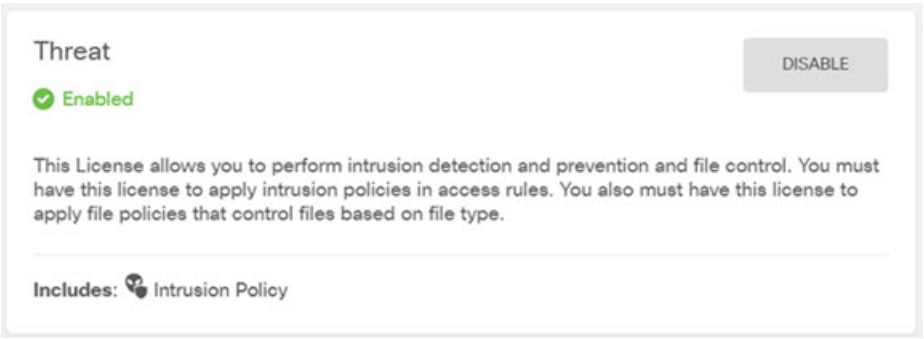
**Step 1** Choose **Device**, then click **View Configuration** in the **Smart License** group.

Click **Enable** for each of the optional licenses you want to use: IPS, malware defense, URL filtering. If you registered the device during setup, you can also enable the RA VPN license desired. Read the explanation of each license if you are unsure of whether you need it.

If you have not registered, you can do so from this page. Click **Request Register** and follow the instructions. Please register before the evaluation license expires.

For example, an enabled IPS license should look like the following:

***Figure 55: Enabled IPS License***



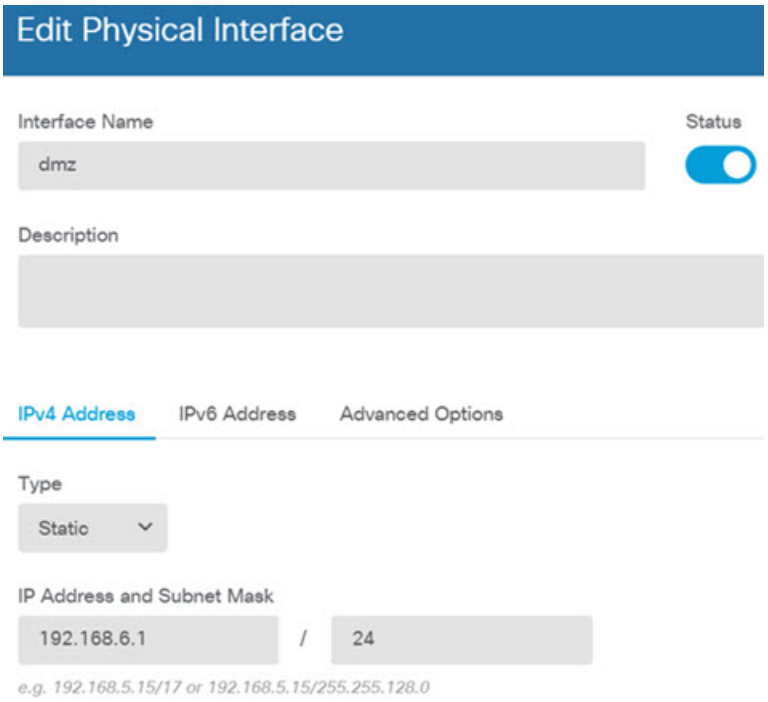

**Step 2** If you configured other interfaces, choose **Device**, then click **View Configuration** in the **Interfaces** group and configure each interface.

You can create a bridge group for the other interfaces, or configure separate networks, or some combination of both.

Click the edit icon ( ) for each interface to define the IP address and other settings.

The following example configures an interface to be used as a “demilitarized zone” (DMZ), where you place publically-accessible assets such as your web server. Click **Save** when you are finished.

***Figure 56: Edit Interface***




**Note**
To enable the IPv6 address, select the IPv6 tab and configure the IPv6 address using static or DHCP.

**Step 3** If you configured new interfaces, choose **Objects**, then select **Security Zones** from the table of contents.

Edit or create new zones as appropriate. Each interface must belong to a zone, because you configure policies based on security zones, not interfaces. You cannot put the interfaces in zones when configuring them, so you must always edit the zone objects after creating new interfaces or changing the purpose of existing interfaces.

The following example shows how to create a new dmz-zone for the dmz interface.

*Figure 57: Security Zone Object*

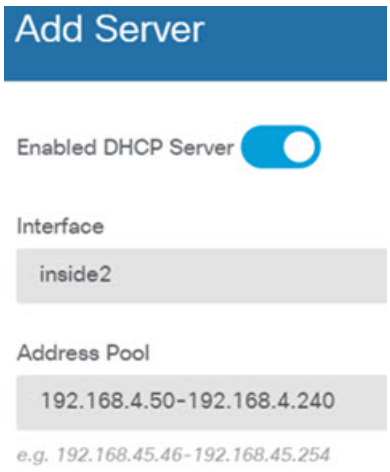



**Step 4** If you want internal clients to use DHCP to obtain an IP address from the device, choose **Device** > **System Settings** > **DHCP Server**, then select the **DHCP Servers** tab.

There is already a DHCP server configured for the inside interface, but you can edit the address pool or even delete it. If you configured other inside interfaces, it is very typical to set up a DHCP server on those interfaces. Click + to configure the server and address pool for each inside interface.

You can also fine-tune the WINS and DNS list supplied to clients on the **Configuration** tab. The following example shows how to set up a DHCP server on the inside2 interface with the address pool 192.168.4.50-192.168.4.240.

*Figure 58: DHCP Server*




**Step 5** Choose **Device**, then click **View Configuration** (or **Create First Static Route**) in the **Routing** group and configure a default route.

The default route normally points to the upstream or ISP router that resides off the outside interface. A default IPv4 route is for any-ipv4 (0.0.0.0/0), whereas a default IPv6 route is for any-ipv6 (::0/0). Create routes for each IP version you use. If you use DHCP to obtain an address for the outside interface, you might already have the default routes that you need.

**Note**
The routes you define on this page are for the data interfaces only. They do not impact the management interface. Set the management gateway on **Device** > **System Settings** > **Management Interface**.

The following example shows a default route for IPv4. In this example, isp-gateway is a network object that identifies the IP address of the ISP gateway (you must obtain the address from your ISP). You can create this object by clicking **Create New Network** at the bottom of the **Gateway** drop-down list.

***Figure 59: Default Route***




**Note**
Similarly, you can configure the IPv6 routes by selecting the IPv6 radio button.

**Step 6** Choose **Policies** and configure the security policies for the network.

The device setup wizard enables traffic flow between the inside-zone and outside-zone, and interface NAT for all interfaces when going to the outside interface. Even if you configure new interfaces, if you add them to the inside-zone object, the access control rule automatically applies to them.

However, if you have multiple inside interfaces, you need an access control rule to allow traffic flow from inside-zone to inside-zone. If you add other security zones, you need rules to allow traffic to and from those zones. These would be your minimum changes.

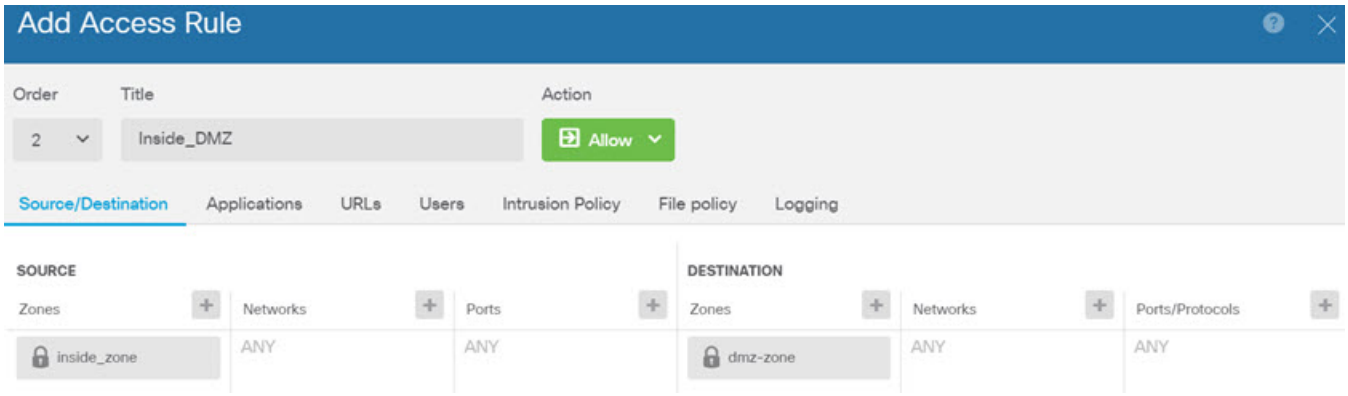
In addition, you can configure other policies to provide additional services, and fine-tune NAT and access rules to get the results that your organization requires. You can configure the following policies:

- **SSL Decryption**—If you want to inspect encrypted connections (such as HTTPS) for intrusions, malware, and so forth, you must decrypt the connections. Use the SSL decryption policy to determine which connections need to be decrypted. The system re-encrypts the connection after inspecting it.
- **Identity**—If you want to correlate network activity to individual users, or control network access based on user or user group membership, use the identity policy to determine the user associated with a given source IP address.

- **Security Intelligence**—Use the Security Intelligence policy to quickly drop connections from or to blacklisted IP addresses or URLs. By blacklisting known bad sites, you do not need to account for them in your access control policy. Cisco provides regularly updated feeds of known bad addresses and URLs so that the Security Intelligence blacklist updates dynamically. Using feeds, you do not need to edit the policy to add or remove items in the blacklist.
- **NAT (Network Address Translation)**—Use the NAT policy to convert internal IP addresses to externally routeable addresses.
- **Access Control**—Use the access control policy to determine which connections are allowed on the network. You can filter by security zone, IP address, protocol, port, application, URL, user or user group. You also apply intrusion and file (malware) policies using access control rules. Use this policy to implement URL filtering.
- **Intrusion**—Use the intrusion policies to inspect for known threats. Although you apply intrusion policies using access control rules, you can edit the intrusion policies to selectively enable or disable specific intrusion rules.

The following example shows how to allow traffic between the inside-zone and dmz-zone in the access control policy. In this example, no options are set on any of the other tabs except for **Logging**, where **At End of Connection** is selected.

***Figure 60: Access Control Policy***




**Step 7** Choose **Device**, then click **View Configuration** in the **Updates** group and configure the update schedules for the system databases.

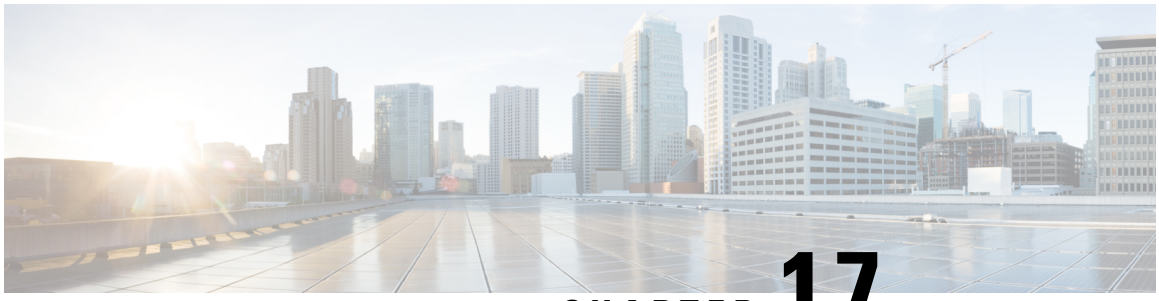
If you are using intrusion policies, set up regular updates for the Rules and VDB databases. If you use Security Intelligence feeds, set an update schedule for them. If you use geolocation in any security policies as matching criteria, set an update schedule for that database.

**Step 8** Click the **Deploy** button in the menu, then click the Deploy Now button ( ), to deploy your changes to the device.

Changes are not active on the device until you deploy them.

### What to do next

For more information about managing the Firewall Threat Defense Virtual with the Firewall Device Manager, see the Cisco Secure Firewall Threat Defense Configuration Guide for Secure firewall Device Manager, or the Secure Firewall Device Manager online help.



CHAPTER 17

# Managing the Secure Firewall Threat Defense Virtual with the Cloud-Delivered Firewall Management Center

This chapter describes how to deploy a Firewall Threat Defense Virtual device managed with the Cloud-Delivered Firewall Management Center.


- Onboarding Overview, on page 435
- Prerequisites to Onboard a Device to Cloud-Delivered Firewall Management Center, on page 436
- Create a Security Cloud Control Tenant, on page 437
- Onboard a Device with a CLI Registration Key, on page 439
- Configure a Basic Security Policy, on page 440


## Onboarding Overview

Cloud-Delivered Firewall Management Center is supported on Threat Defense Virtual devices running Secure Firewall versions 7.0.3,7.2.0, and later. To see all supported versions and product compatibility, see Secure Firewall Threat Defense Compatibility Guide for more information.

There are three different types of scenarios in which you onboard a threat defense virtual device to the cloud-delivered Firewall Management Center:

- Onboard a new threat defense virtual device
- Onboard a threat defense virtual device that is currently managed by the device manager

**Note** If you onboard a device manager-managed device to the Cloud-Delivered Firewall Management Center, you can no longer manage the device with the device manager.

- Onboard a threat defense virtual device that is currently managed by an on-prem management center. See Migrate Secure Firewall Threat Defense to Cloud for more information.

**Note** The following scenarios occur when you either move or migrate a device to the Cloud-Delivered Firewall Management Center:

- If you delete a device from an on-prem management center or Secure Firewall Threat Defense device manager to onboard to the Cloud-Delivered Firewall Management Center, the change of managers wipes any policies configured through the on-prem management center.
- If you migrate a device from an on-prem management center to the Cloud-Delivered Firewall Management Center, the device retains the majority of your previously configured policies.

If you do not know if your device is already managed by an alternative manager, use the **show managers** command on the CLI.

This guide provides information on the basics of managing threat Defense Virtual using Cloud-Delivered Firewall Management Center. For more detailed information on Security Cloud Control, see Cisco Security Cloud Control.

# Prerequisites to Onboard a Device to Cloud-Delivered Firewall Management Center

### Onboard Limitations and Requirements

Be aware of the following limitations when onboarding a device to the cloud-delivered Firewall Management Center:

- Devices **must** be running version 7.0.3, or version 7.2 and later. We **strongly** recommend version 7.2 or later.
- You can migrate an HA pair that is managed by an On-Premises Firewall Management Center by following the Migrate FTD to Cloud-Delivered Firewall Management Center process. Confirm both peers are in a healthy state prior to migrating.
- Only devices that are configured for local management and are managed by a Firewall Device Manager can be onboarded with the serial number and zero-touch provisioning methods.
- If the device is managed by an on-premises management center, you can either onboard the device to cloud-delivered Firewall Management Center or migrate the device. Migrating retains any existing policies and objects, whereas onboarding the device removes most policies and all objects. See Migrate FTD to Cloud-Delivered Firewall Management Center for more information.
- If your device is currently managed by a Firewall Device Manager, unregister all your smart licenses before you onboard the device. Even if you switch device management, the Cisco Smart Software Manager will retain the smart licenses.
- If you have previously onboarded a device that was managed by a Firewall Device Manager and deleted the device from Security Cloud Control with the intention of re-onboarding for cloud management, you **must** register the Firewall Device Manager to the Security Services Exchange cloud after deleting the

device. See the "Access Security Services Exchange" chapter in the *Firepower and Cisco SecureX Threat Response Integration Guide.*

**Tip** Onboarding a device to the cloud-delivered Firewall Management Center removes any policies and most objects configured through the previous manager. If your device is currently managed by an on-premises management center, it is possible to migrate the device and retain your policies and objects. See Migrate FTD to Cloud-Delivered Firewall Management Center for more information.

### Network Requirements

Before you onboard a device, ensure the following ports have external and outbound access. Confirm the following ports on the device are allowed. If communication ports are blocked behind a firewall, onboarding the device may fail.

**Note** You cannot configure these ports in the Security Cloud Control UI. You must enable these ports through the device's SSH.

***Table 36: Device Port Requirements***

| Port | Protocol / Feature | Details |
|---|---|---|
| 443/tcp | HTTPS | Send and receive data from the internet. |
| 443 | HTTPS | Communicate with the AMP cloud (public or private) |
| 8305/tcp | Appliance communications | Securely communicate between appliances in a deployment. |

### Management and Data Interfaces

Make sure your device is correctly configured with either a management or data interface.

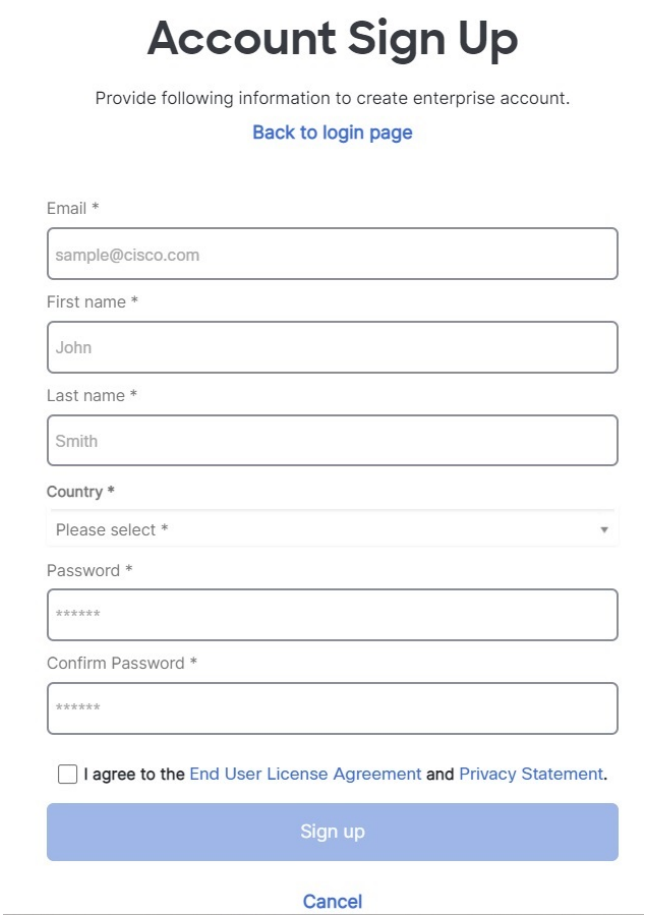
# Create a Security Cloud Control Tenant

You can provision a new Security Cloud Control tenant to onboard and manage your devices. If you use an On-Premises Firewall Management Center Version 7.2 and later, and want to integrate it with the Cisco Security Cloud, you can also create a Security Cloud Control tenant as part of the integration workflow.

### Procedure

1. Go to https://us.manage.security.cisco.com/provision.
2. Select the region where you want to provision your Security Cloud Control tenant and click **Sign Up**.
3. On the **Security Cloud Sign On** page, provide your credentials.

4. If you do not have a Security Cloud Sign On account and want to create one, click **Sign up now**.
   a. Provide the information to create an account.




   Here are some tips:

   - **Email**: Enter the email address that you will eventually use to log in to Security Cloud Control.
   - **Password**: Enter a strong password.

   b. Click **Sign up**. Cisco sends you a verification email to the address you registered with.
   c. Open the email and click **Activate account** both on the mail and the **Security Cloud Sign On** page.
   d. Configure multifactor authentication using Duo on a device of your choice and click **Log in with Duo** and **Finish**.

**Note** We recommend installing the Duo Security app on a mobile phone. Review Duo Guide to Two Factor Authentication: Enrollment Guide if you have questions about installing Duo.

5. Provide a name for your tenant and click **Create new account**.

6. A new Security Cloud Control tenant is created in the region that you have chosen; you will also receive an email about your Security Cloud Control tenant being created, with the details. If you are associated with multiple Security Cloud Control tenants already, on the **Choose a tenant** page, select the tenant you just created to log in to it. If you have created a new Security Cloud Control tenant for the first time, you get logged into your tenant directly.

# Onboard a Device with a CLI Registration Key

Use the procedure below to onboard a device for cloud-delivered Firewall Management Center with a CLI registration key.

**Note** If your device is currently managed by an on-premises management center, onboarding the device will fail. You can either delete the device from the on-premises management center and onboard as a fresh, new device with no policies or objects, or you can migrate the device and retain the existing policies and objects. See Migrate FTD to Cloud-Delivered Firewall Managmenet Center for more information.

### Before you begin

Before you onboard a device, be sure to complete the following tasks:

- Cloud-Delivered Firewall Management Center is enabled for your tenant.
- Device must be running version 7.0.3, or 7.2.0 and later.

### Procedure

**Step 1** Log in to Security Cloud Control.

**Step 2** In the left pane, click .

**Step 3** In the top-right corner, click **Onboard** ( ).

**Step 4** Click the **FTD** tile.

**Step 5** Under **Management Mode**, ensure you select **FTD**. By selecting **FTD** under **Management Mode**, you will not be able to manage the device using the previous management platform. All existing policy configurations except for interface configurations will be reset. You must re-configure policies after you onboard the device.

**Step 6** Select **Use CLI Registration Key** as the onboarding method.

**Step 7** Enter the device name in the **Device Name** field and click **Next**.

**Step 8** In the **Policy Assignment** step, use the drop-down menu to select an access control policy to deploy once the device is onboarded. If you have no policies configured, select the **Default Access Control Policy**.

**Step 9** Specify whether the device you are onboarding is a physical or virtual device. If you are onboarding a virtual device, you must select the device's performance tier from the drop-down menu.

**Step 10** Select the subscription licenses you want to apply to the device. Click **Next**.

**Step 11** Security Cloud Control generates a command with the registration key. Connect to the device you are onboarding using SSH. Log in as "admin" or a user with equivalent admin privileges and paste the entire registration key as is into the device's CLI.

**Note:** For Firepower 1000, Firepower 2100, ISA 3000, and Firewall Threat Defense Virtual devices, open an SSH connection to the device and log in as `admin`. Copy the entire registration command and paste it into the device's CLI interface at the prompt. In the CLI, enter **Y** to complete the registration. If your device was previously managed by Firewall Device Manager, enter **Yes** to confirm the submission.

**Step 12** Click **Next** in the Security Cloud Control onboarding wizard.

**Step 13** (Optional) Add labels to your device to help sort and filter the **Security Devices** page. Enter a label and select the blue plus button. Labels are applied to the device after it's onboarded to Security Cloud Control.

### What to do next

Once the device is synchronized, select the device you just onboarded from the **Security Devices** page and select any of the options listed under the **Device Management** pane located to the right. We strongly recommend the following actions:

- If you did not already, create a custom access control policy to customize the security for your environment. See Access Control Overview in *Managing Firewall Threat Defense with Cloud-Delivered Firewall Management Center in Firewall in Security Cloud Control* for more information.
- Enable Cisco Security Analytics and Logging (SAL) to view events in the Security Cloud Control dashboard **or** register the device to an Secure Firewall Management Center for security analytics. See Cisco Security Analytics and Logging in *Managing Firewall Threat Defense with Cloud-Delivered Firewall Management Center in Firewall in Security Cloud Control* for more information.

# Configure a Basic Security Policy

This section describes how to configure a basic security policy with the following settings:

- Inside and outside interfaces—Assign a static IP address to the inside interface, and use DHCP for the outside interface.
- DHCP server—Use a DHCP server on the inside interface for clients.
- Default route—Add a default route through the outside interface.
- NAT—Use interface PAT on the outside interface.
- Access control—Allow traffic from inside to outside.

### Procedure

**Step 1** Configure Interfaces, on page 413

**Step 2** Configure the DHCP Server, on page 417

**Step 3** Add the Default Route, on page 418

**Step 4** Configure NAT, on page 420

**Step 5** Configure Access Control, on page 423

**Step 6** Deploy the Configuration, on page 424

# Configure Interfaces

Enable the Firewall Threat Defense Virtual interfaces, assign them to security zones, and set the IP addresses. Typically, you must configure at least a minimum of two interfaces to have a system that passes meaningful traffic. Normally, you would have an outside interface that faces the upstream router or internet, and one or more inside interfaces for your organization's networks. Some of these interfaces might be "demilitarized zones" (DMZs), where you place publically-accessible assets such as your web server.

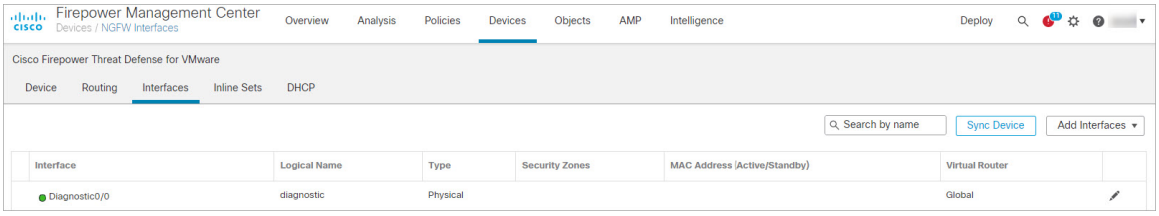
A typical edge-routing situation is to obtain the outside interface address through DHCP from your ISP, while you define static addresses on the inside interfaces.

The following example configures a routed mode inside interface with a static address and a routed mode outside interface using DHCP.

## Procedure

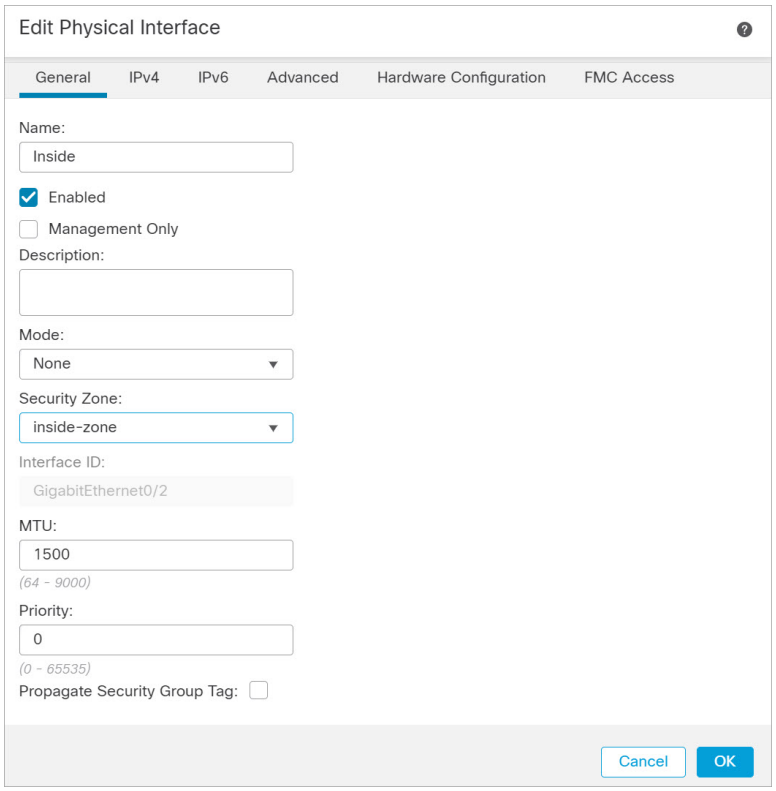
**Step 1** Choose **Devices** > **Device Management**, and click the **Edit** (✎) for the device.

**Step 2** Click **Interfaces**.




**Step 3** Click the **Edit** (✎) for the interface that you want to use for *inside*.

The **General** tab appears.




a) Enter a **Name** up to 48 characters in length.

   For example, name the interface **inside**.

b) Check the **Enabled** check box.

c) Leave the **Mode** set to **None**.

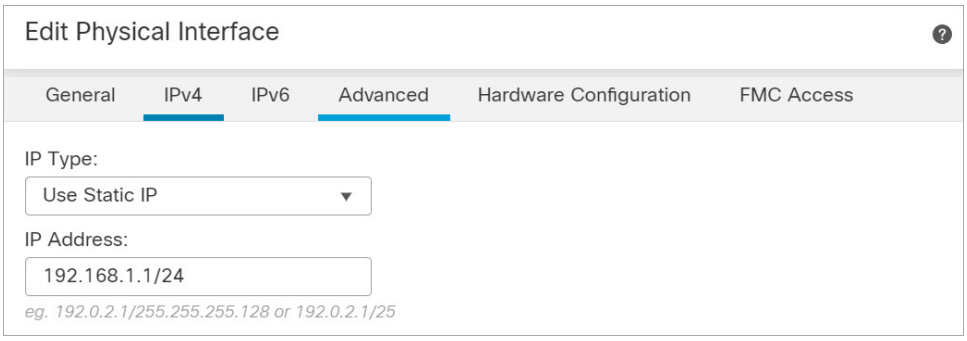
d) From the **Security Zone** drop-down list, choose an existing inside security zone or add a new one by clicking **New**.

   For example, add a zone called **inside_zone**. Each interface must be assigned to a security zone and/or interface group. An interface can belong to only one security zone, but can also belong to multiple interface groups. You apply your security policy based on zones or groups. For example, you can assign the inside interface to the inside zone; and the outside interface to the outside zone. Then you can configure your access control policy to enable traffic to go from inside to outside, but not from outside to inside. Most policies only support security zones; you can use zones or interface groups in NAT policies, prefilter policies, and QoS policies.

e) Click the **IPv4** and/or **IPv6** tab.

   - **IPv4**—Choose **Use Static IP** from the drop-down list, and enter an IP address and subnet mask in slash notation or DHCP option .

     For example, enter **192.168.1.1/24**

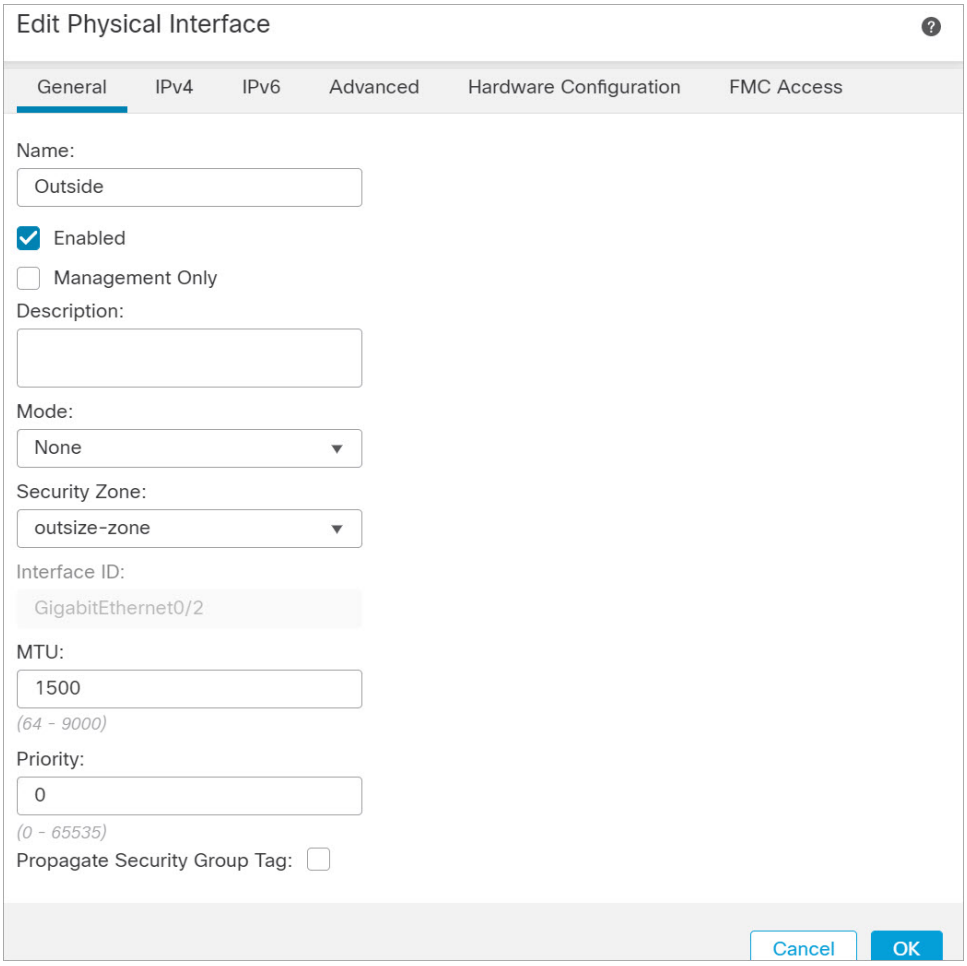



- **IPv6**—Check the **Autoconfiguration** check box for stateless auto configuration and also for IPv6 DHCP or static configuration to enable the interface.

f) Click **OK**.

**Step 4** Click the **Edit** (✎) for the interface that you want to use for *outside*.

The **General** tab appears.




a) Enter a **Name** up to 48 characters in length.

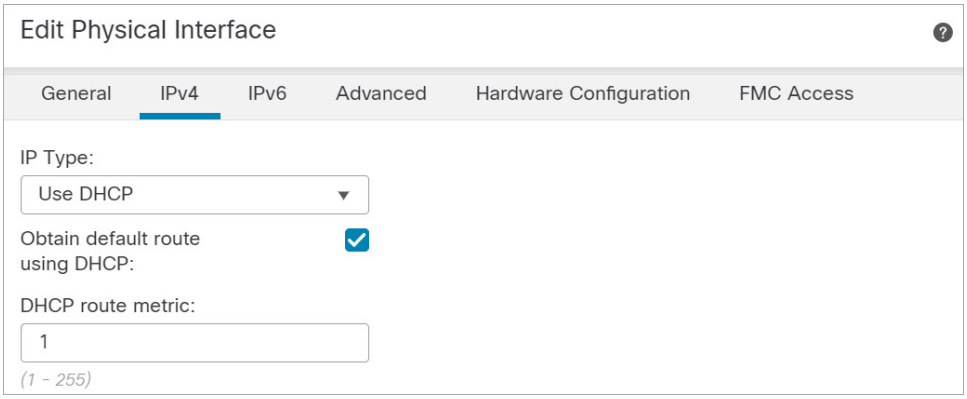
For example, name the interface **outside**.

b) Check the **Enabled** check box.
c) Leave the **Mode** set to **None**.
d) From the **Security Zone** drop-down list, choose an existing outside security zone or add a new one by clicking **New**.

For example, add a zone called **outside_zone**.

e) Click the **IPv4** and/or **IPv6** tab.

- **IPv4**—Choose **Use DHCP**, and configure the following optional parameters:
  - **Obtain default route using DHCP**—Obtains the default route from the DHCP server.
  - **DHCP route metric**—Assigns an administrative distance to the learned route, between 1 and 255. The default administrative distance for the learned routes is 1.




- **IPv6**—Check the **Autoconfiguration** check box for stateless auto configuration.

f) Click **OK**.

**Step 5** Click **Save**.

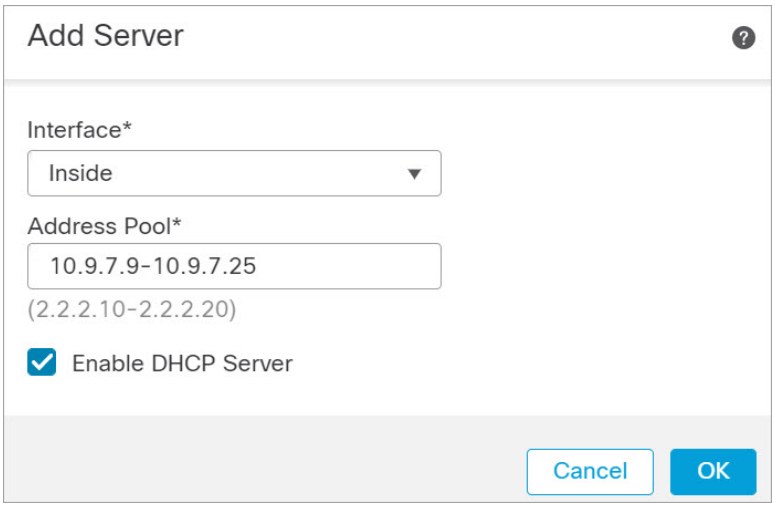
# Configure the DHCP Server

**Note** Skip this procedure if you are deploying to a public cloud environment such as AWS, Azure, GCP, OCI.

Enable the DHCP server if you want clients to use DHCP to obtain IP addresses from the Firewall Threat Defense Virtual.

### Procedure

**Step 1** Choose **Devices** > **Device Management**, and click the **Edit** (✎) for the device.
**Step 2** Choose **DHCP** > **DHCP Server**.
**Step 3** On the **Server** page, click **Add**, and configure the following options:




- **Interface**—Choose the interface from the drop-down list.
- **Address Pool**—Set the range of IP addresses from lowest to highest that are used by the DHCP server. The range of IP addresses must be on the same subnet as the selected interface and cannot include the IP address of the interface itself.
- **Enable DHCP Server**—Enable the DHCP server on the selected interface.

**Step 4** Click **OK**.

**Step 5** Click **Save**.

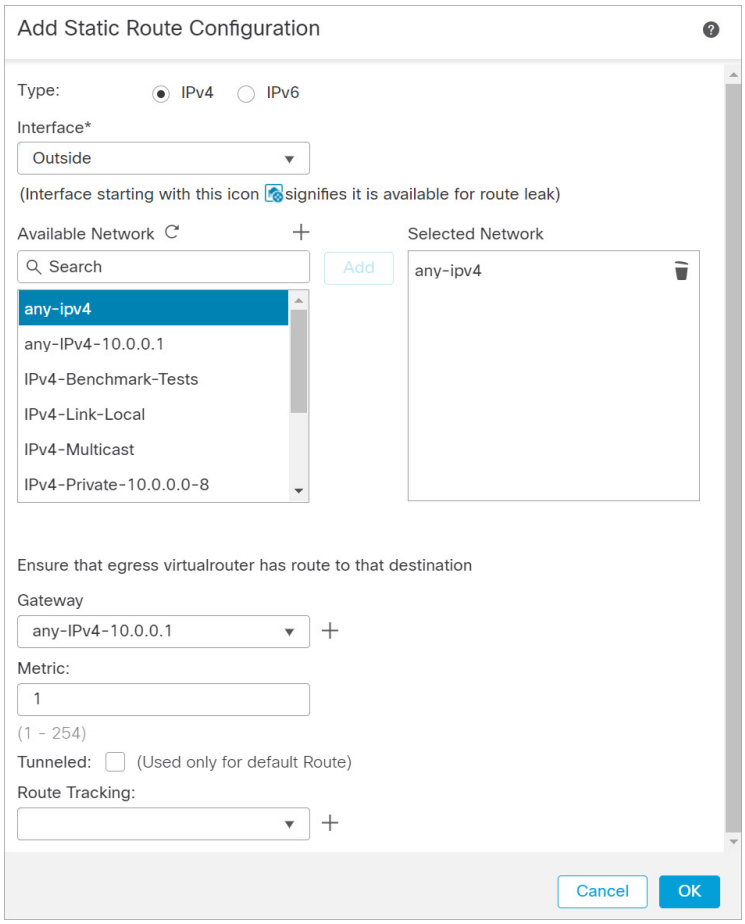
# Add the Default Route

The default route normally points to the upstream router reachable from the outside interface. If you use DHCP for the outside interface, your device might have already received a default route. If you need to manually add the route, complete this procedure. If you received a default route from the DHCP server, it will show in the **IPv4 Routes** or **IPv6 Routes** table on the **Devices** > **Device Management** > **Routing** > **Static Route** page.

## Procedure

**Step 1** Choose **Devices** > **Device Management**, and click the **Edit** (✎) for the device.

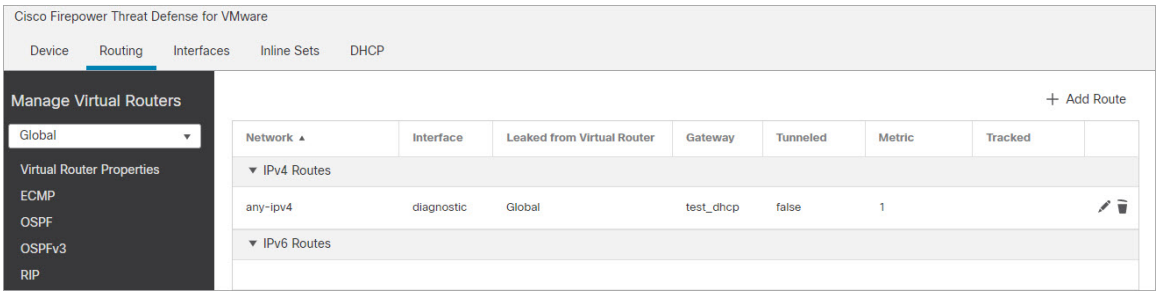
**Step 2** Choose **Routing** > **Static Route**, click **Add Route**, and set the following:




- **Type**—Click the **IPv4** or **IPv6** radio button depending on the type of static route that you are adding.
- **Interface**—Choose the egress interface; typically the outside interface.
- **Available Network**—Choose **any-ipv4** for an IPv4 default route, or **any-ipv6** for an IPv6 default route.
- **Gateway** or **IPv6 Gateway**—Enter or choose the gateway router that is the next hop for this route. You can provide an IP address or a Networks/Hosts object.
- **Metric**—Enter the number of hops to the destination network. Valid values range from 1 to 255; the default value is 1.

**Step 3** Click **OK**.

The route is added to the static route table.




**Step 4** Click **Save**.

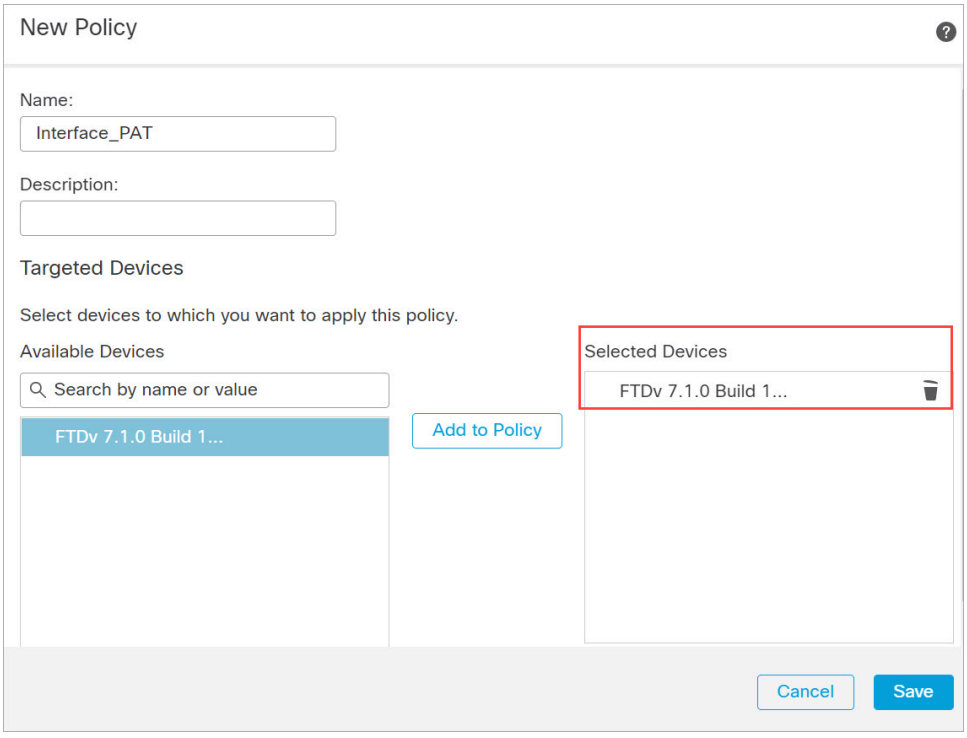
# Configure NAT

A typical NAT rule converts internal addresses to a port on the outside interface IP address. This type of NAT rule is called *interface Port Address Translation (PAT)*.

## Procedure

**Step 1** Choose **Devices** > **NAT**, and click **New Policy** > **Threat Defense NAT**.

**Step 2** Name the policy, select the device(s) that you want to use the policy, and click **Save**.

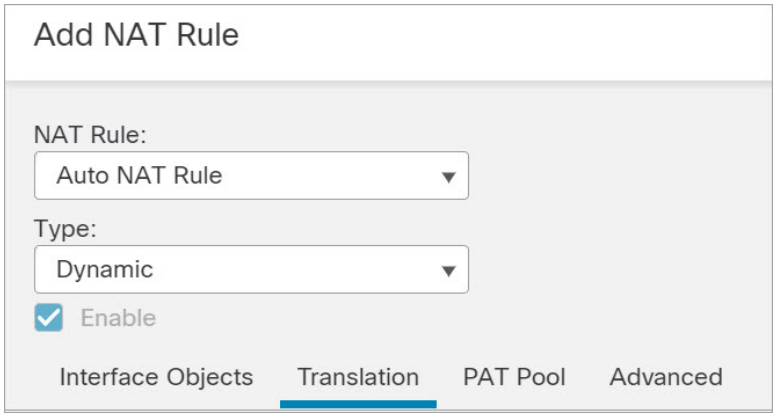



The policy is added the Firewall Management Center. You still have to add rules to the policy.

**Step 3** Click **Add Rule**.

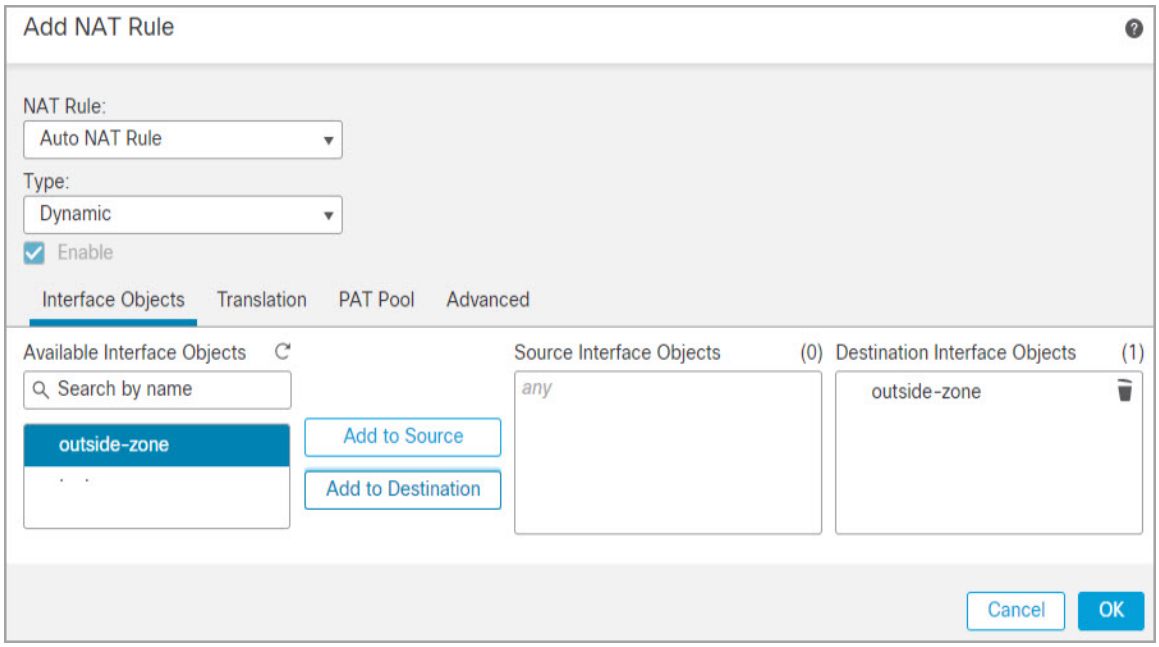
The **Add NAT Rule** dialog box appears.

**Step 4** Configure the basic rule options:

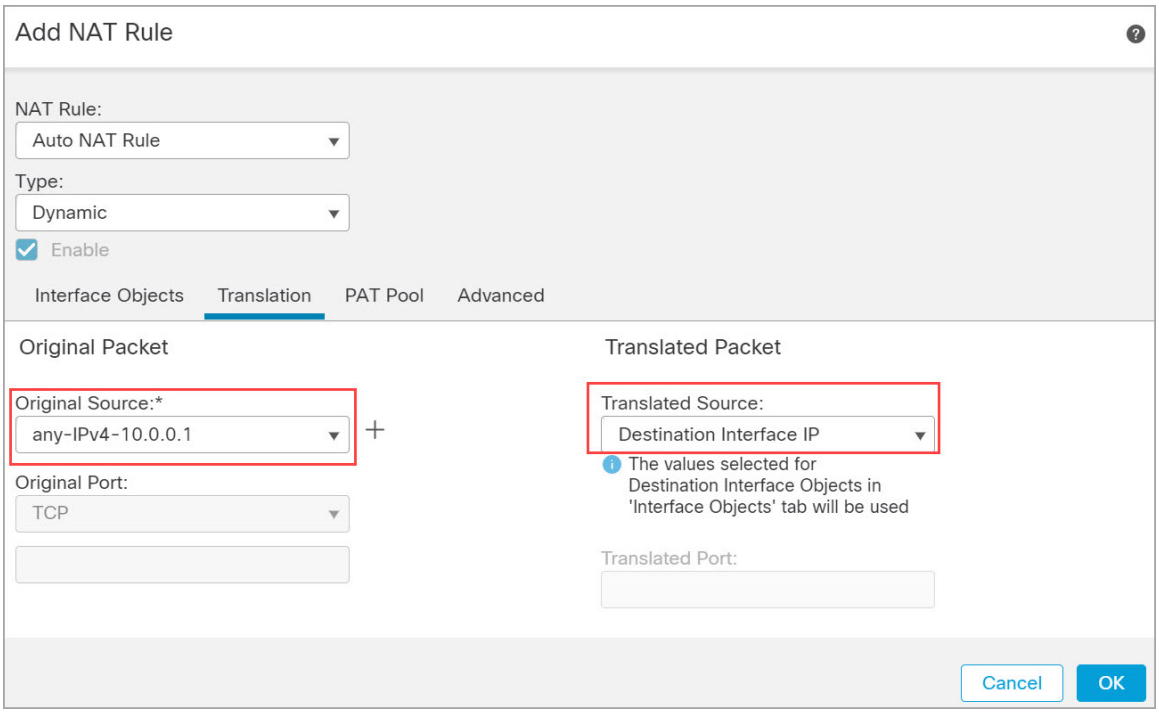



- **NAT Rule**—Choose **Auto NAT Rule**.
- **Type**—Choose **Dynamic**.

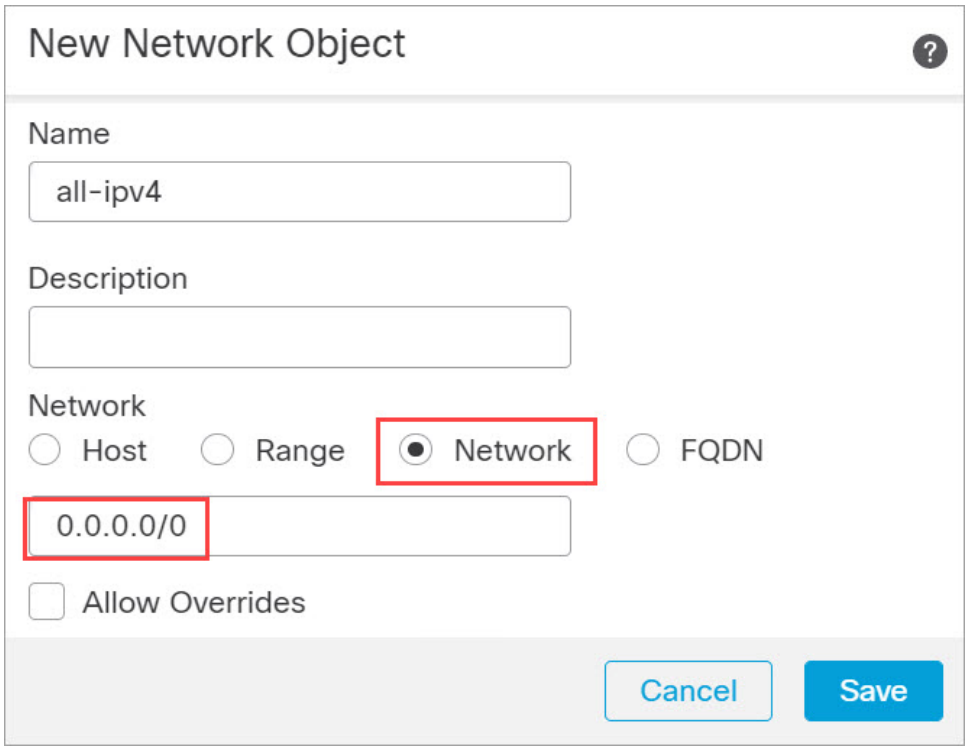
**Step 5** On the **Interface Objects** page, add the outside zone from the **Available Interface Objects** area to the **Destination Interface Objects** area.




**Step 6** On the **Translation** page, configure the following options:




- **Original Source**—Click the **Add** (+) to add a network object for all IPv4 traffic (0.0.0.0/0).



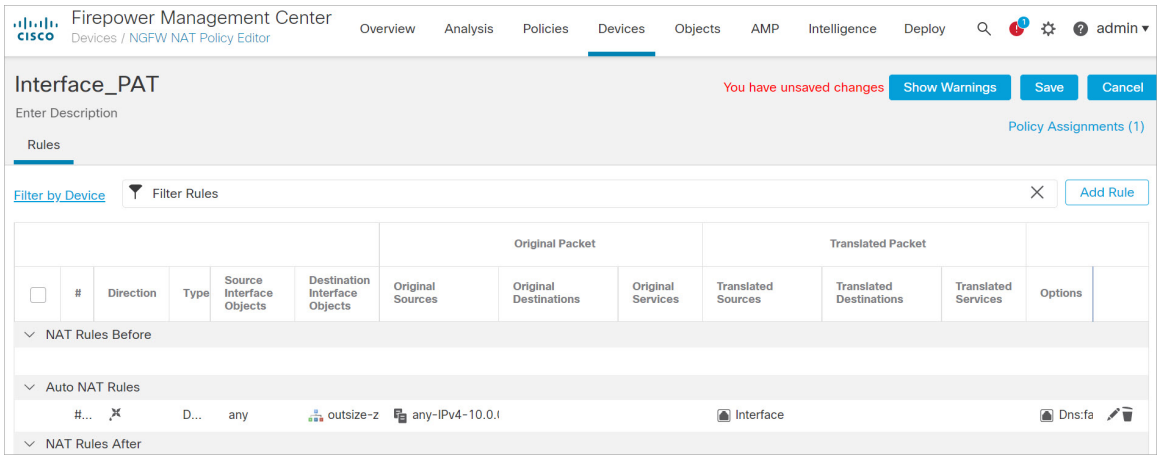

**Note**
You cannot use the system-defined **any-ipv4** object, because Auto NAT rules add NAT as part of the object definition, and you cannot edit system-defined objects.

Similarly, you can create the NAT policy with a default host network [::/0] for all IPv6 traffic.

- **Translated Source**—Choose **Destination Interface IP**.

**Step 7** Click **Save** to add the rule.

The rule is saved to the **Rules** table.




**Step 8** Click **Save** on the **NAT** page to save your changes.

# Configure Access Control

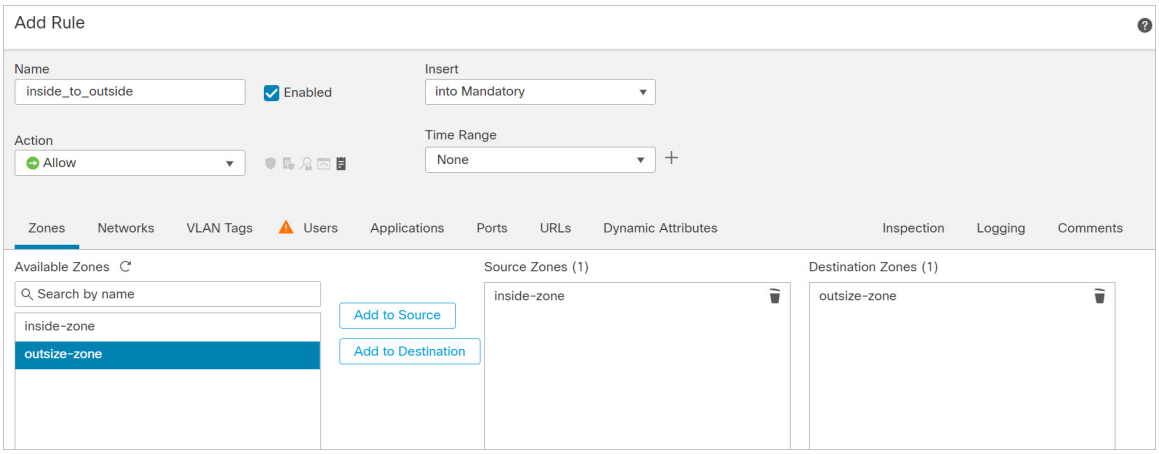
If you created a basic **Block all traffic** access control policy when you registered the Firewall Threat Defense Virtual with the Firewall Management Center, then you need to add rules to the policy to allow traffic through the device. The following procedure adds a rule to allow traffic from the inside zone to the outside zone. If you have other zones, be sure to add rules allowing traffic to the appropriate networks.

See the Firewall Management Center Configuration Guide configuration guide to configure more advanced security settings and rules.

## Procedure

**Step 1** Choose **Policy** > **Access Policy** > **Access Policy**, and click the **Edit** (✎) for the access control policy assigned to the Firewall Threat Defense.

**Step 2** Click **Add Rule**, and set the following parameters:

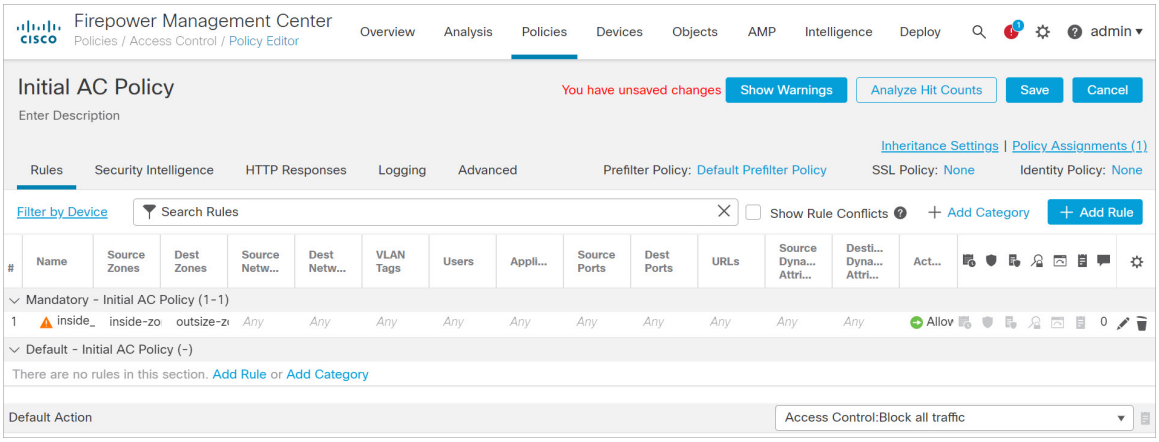



- **Name**—Name this rule, for example, **inside_to_outside**.
- **Source Zones**—Select the inside zone from **Available Zones**, and click **Add to Source**.
- **Destination Zones**—Select the outside zone from **Available Zones**, and click **Add to Destination**.

Leave the other settings as is.

**Step 3** Click **Add**.

The rule is added to the **Rules** table.



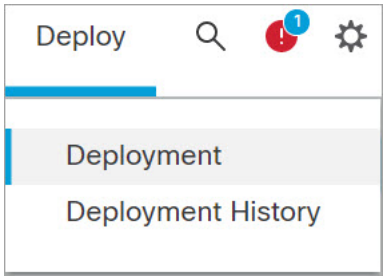

**Step 4** Click **Save**.

# Deploy the Configuration

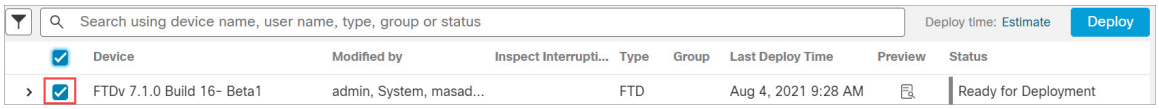
Deploy the configuration changes to the Firewall Threat Defense Virtual; none of your changes are active on the device until you deploy them.

## Procedure

**Step 1** Click **Deploy** in the upper right.




**Step 2** Select the device in the **Deploy Policies** dialog box, then click **Deploy**.




**Step 3** Ensure that the deployment succeeds. Click the icon to the right of the **Deploy** button in the menu bar to see status for deployments.

