# *Set Up the VM-Series Firewall on AWS*

docs.paloaltonetworks.com

techDOCS paloalto NETWORKS

## Contact Information

Corporate Headquarters:
Palo Alto Networks
3000 Tannery Way
Santa Clara, CA 95054
www.paloaltonetworks.com/company/contact-support

## About the Documentation

- To ensure you are viewing the most current version of this document, or to access related documentation, visit the Technical Documentation portal: docs.paloaltonetworks.com.
- To search for a specific topic, go to our search page: docs.paloaltonetworks.com/search.html.
- Have feedback or questions for us? Leave a comment on any page in the portal, or write to us at documentation@paloaltonetworks.com.

## Copyright

Palo Alto Networks, Inc.
www.paloaltonetworks.com

© 2019-2019 Palo Alto Networks, Inc. Palo Alto Networks is a registered trademark of Palo Alto Networks. A list of our trademarks can be found at www.paloaltonetworks.com/company/trademarks.html. All other marks mentioned herein may be trademarks of their respective companies.

## Last Revised

May 16, 2019

# Table of Contents

# Set Up the VM-Series Firewall on AWS

The VM-Series firewall can be deployed in the public Amazon Web Services (AWS) cloud and AWS GovCloud. It can then be configured to secure access to the applications that are deployed on EC2 instances and placed into a Virtual Private Cloud (VPC) on AWS.





© Palo Alto Networks, Inc.

# About the VM-Series Firewall on AWS

The Amazon Web Service (AWS) is a public cloud service that enables you to run your applications on a shared infrastructure managed by Amazon. These applications can be deployed on scalable computing capacity or EC2 instances in different AWS regions and accessed by users over the internet.

For networking consistency and ease of management of EC2 instances, Amazon offers the Virtual Private Cloud (VPC). A VPC is apportioned from the AWS public cloud, and is assigned a CIDR block from the private network space (RFC 1918). Within a VPC, you can carve public/private subnets for your needs and deploy the applications on EC2 instances within those subnets. To then enable access to the applications within the VPC, you can deploy the VM-Series firewall on an EC2 instance. The VM-Series firewall can then be configured to secure traffic to and from the EC2 instances within the VPC.

The VM-Series firewall is available in both the public AWS cloud and on AWS GovCloud. The VM-Series firewall in public AWS and AWS GovCloud supports the Bring Your Own License (BYOL) model and the hourly Pay-As-You-Go (PAYG), the usage-based licensing model that you can avail from the AWS Marketplace. For licensing details, see VM-Series Firewall Licenses for Public Clouds.

- AWS Instance Types
- VM-Series Firewall on AWS GovCloud
- VM-Series Firewall on AWS China
- AWS Terminology
- Management Interface Mapping for Use with Amazon ELB

## AWS EC2 Instance Types

The VM-Series firewalls support the following AWS instance types— C3, C4, C5, M3, M4, M5.

You can deploy the VM-Series firewall on an AWS instance size with more resources than the minimum VM-Series System Requirements. If you choose a larger instance size for the VM-Series firewall model, although the firewall only uses the max vCPU cores and memory shown in table, it does take advantage of the faster network performance that AWS provides.

The C3, C4, M3, M4 instance types support both DPDK and SR-IOV modes; the C5 and M5 instance types that have the Elastic Network Adapter (ENA) support SR-IOV mode only. For guidance with sizing the VM-Series firewall on AWS, refer to this article.

## VM-Series Firewall on AWS GovCloud

AWS GovCloud is an isolated AWS region that meets the regulatory and compliance requirements of the US government agencies and customers.

To secure your workloads that contain all categories of Controlled Unclassified Information (CUI) data and government-oriented, publicly available data in the AWS GovCloud (US) Region, the VM-Series firewall provides the same robust security features in the standard AWS public cloud and on AWS GovCloud. The VM-Series firewall on AWS GovCloud and the standard AWS public cloud support the same capabilities.

See AMI on AWS GovCloud to Deploy the VM-Series Firewall on AWS.

## VM-Series Firewall on AWS China

The VM-Series firewall is available as a shared AMI with the BYOL option on AWS China (Beijing) region. You must have an AWS China account that is separate from your global AWS account to access this image and use AWS resources on AWS China.



© Palo Alto Networks, Inc.

To launch the VM-Series firewall in your AWS China account, find the AMI for the VM-Series firewall on the EC2 console (**Instances** > **Launch** > **Instance** > **Community AMIs**) using the AMI ID (ami-5157873c) or by searching for Palo Alto Networks. Make sure to review the VM-Series System Requirements before Launch the VM-Series Firewall on AWS.

*You cannot bootstrap the VM-Series firewall on AWS China.*

## AWS Terminology

This document assumes that you are familiar with the networking and configuration of the AWS VPC. In order to provide context for the terms used in this section, here is a brief refresher on the AWS terms (some definitions are taken directly from the AWS glossary) that are referred to in this document:

| Term | Description |
|---|---|
| EC2 | Elastic Compute Cloud<br>A web service that enables you to launch and manage Linux/UNIX and Windows server instances in Amazon's data centers. |
| AMI | Amazon Machine Image<br>An AMI provides the information required to launch an instance, which is a virtual server in the cloud.<br>The VM-Series AMI is an encrypted machine image that includes the operating system required to instantiate the VM-Series firewall on an EC2 instance. |
| ELB | Elastic Load Balancing<br>ELB is an Amazon web service that helps you improve the availability and scalability of your applications by routing traffic across multiple Elastic Compute Cloud (EC2) instances. ELB detects unhealthy EC2 instances and reroutes traffic to healthy instances until the unhealthy instances are restored. ELB can send traffic only to the primary interface of the next hop load-balanced EC2 instance. So, to use ELB with a VM-Series firewall on AWS, the firewall must be able to use the primary interface for dataplane traffic. |
| ENI | Elastic Network Interface<br>An additional network interface that can be attached to an EC2 instance. ENIs can include a primary private IP address, one or more secondary private IP addresses, a public IP address, an elastic IP address (optional), a MAC address, membership in specified security groups, a description, and a source/destination check flag. |
| IP address types for EC2 instances | An EC2 instance can have different types of IP addresses.<br>• Public IP address: An IP address that can be routed across the internet.<br>• Private IP address: A IP address in the private IP address range as defined in the RFC 1918. You can choose to manually assign an IP address or to auto assign an IP address within the range in the CIDR block for the subnet in which you launch the EC2 instance. |



© Palo Alto Networks, Inc.

| Term | Description |
| --- | --- |
| | If you are manually assigning an IP address, Amazon reserves the first four (4) IP addresses and the last one (1) IP address in every subnet for IP networking purposes.<br>• Elastic IP address (EIP): A static IP address that you have allocated in Amazon EC2 or Amazon VPC and then attached to an instance. Elastic IP addresses are associated with your account, not with a specific instance. They are elastic because you can easily allocate, attach, detach, and free them as your needs change.<br>An instance in a public subnet can have a Private IP address, a Public IP address, and an Elastic IP address (EIP); an instance in a private subnet will have a private IP address and optionally have an EIP. |
| Instance type | Amazon-defined specifications that stipulate the memory, CPU, storage capacity, and hourly cost for an instance. Some instance types are designed for standard applications, whereas others are designed for CPU-intensive, memory-intensive applications, and so on. |
| VPC | Virtual Private Cloud<br>An elastic network populated by infrastructure, platform, and application services that share common security and interconnection. |
| IGW | Internet gateway provided by Amazon.<br>Connects a network to the internet. You can route traffic for IP addresses outside your VPC to the internet gateway. |
| IAM Role | Identity and Access Management<br>Required for enabling High Availability for the VM-Series firewall on AWS. The IAM role defines the API actions and resources the application can use after assuming the role. On failover, the IAM Role allows the VM-Series firewall to securely make API requests to switch the dataplane interfaces from the active peer to the passive peer.<br>An IAM role is also required for VM Monitoring. See List of Attributes Monitored on the AWS VPC. |
| Subnets | A segment of the IP address range of a VPC to which EC2 instances can be attached. EC2 instances are grouped into subnets based on your security and operational needs.<br>There are two types of subnets:<br>• Private subnet: The EC2 instances in this subnet cannot be reached from the internet.<br>• Public subnet: The internet gateway is attached to the public subnet, and the EC2 instances in this subnet can be reached from the internet. |
| Security groups | A security group is attached to an ENI and it specifies the list of protocols, ports, and IP address ranges that are allowed to establish inbound/outbound connections on the interface. |


© Palo Alto Networks, Inc.

| Term | Description |
| --- | --- |
| | *In the AWS VPC, security groups and network ACLs control inbound and outbound traffic; security groups regulate access to the EC2 instance, while network ACLs regulate access to the subnet. Because you are deploying the VM-Series firewall, set more permissive rules in your security groups and network ACLs and allow the firewall to safely enable applications in the VPC.* |
| Route tables | A set of routing rules that controls the traffic leaving any subnet that is associated with the route table. A subnet can be associated with only one route table. |
| Key pair | A set of security credentials you use to prove your identity electronically. The key pair consists of a private key and a public key. At time of launching the VM-Series firewall, you must generate a key pair or select an existing key pair for the VM-Series firewall. The private key is required to access the firewall in maintenance mode. |
| CloudWatch | Amazon CloudWatch is a monitoring service that allows you to collect and track metrics for the VM-Series firewalls on AWS. When enabled, the firewalls use AWS APIs to publish native PAN-OS metrics to CloudWatch. |

## Management Interface Mapping for Use with Amazon ELB

By default, the elastic network interface (ENI) eth0 maps to the MGT interface on the firewall and ENI eth1 maps to ethernet 1/1 on the firewall. Because the ELB can send traffic only to the primary interface of the next hop load-balanced EC2 instance, the VM-Series firewall must be able to use the primary interface for dataplane traffic.

The firewall can receive dataplane traffic on the primary interface in the following scenarios where the VM-Series firewall is behind the Amazon ELB Service (for a topology diagram, see Auto Scale VM-Series Firewalls with the Amazon ELB Service):

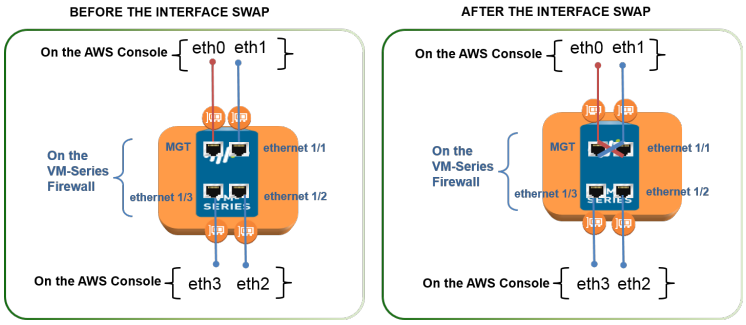
- The VM-Series firewall(s) is securing traffic outbound directly to the internet without the need for using a VPN link or a Direct Connect link back to the corporate network.
- The VM-Series firewall secures an internet-facing application when there is exactly one back-end server, such as a web server, for each firewall. The VM-Series firewalls and web servers can scale linearly, in pairs, behind ELB.

*At present, for use cases that require an ELB sandwich-type deployment to scale out firewalls and application layer EC2 instances, swapping the management interface will not allow you to seamlessly deploy the ELB solution. The ability to swap the management interface only partially solves the integration with ELB.*

To allow the firewall to send and receive dataplane traffic on eth0 instead of eth1, you must swap the mapping of the ENIs within the firewall such that ENI eth0 maps to ethernet 1/1 and ENI eth1 maps to the MGT interface on the firewall as shown below.



© Palo Alto Networks, Inc.




Swapping how the interfaces are mapped allows ELB to distribute and route traffic to healthy instances of the VM-Series firewall located in the same or different Availability Zones on AWS for increased capacity and fault tolerance.

The interface swap is only required when the VM-Series firewall is behind the Amazon ELB Service. If your requirement is to deploy the VM-Series firewalls in a traditional high availability set up, you don't need to configure the interface swap that is described in this section. Continue to High Availability for VM-Series Firewall on AWS.

To swap the interfaces, you have the following options:

- **At launch**—When you launch the firewall, you can either enter the **`mgmt-interface-swap=enable`** command in the **User data** field on the AWS management console (see Launch the VM-Series Firewall on AWS) or CLI or you can include the new **`mgmt-interface-swap`** operational command in the bootstrap configuration.
- **After launch**—After you launch the firewall, Use the VM-Series Firewall CLI to Swap the Management Interface (**`set system setting mgmt-interface-swap enable yes`** operational command) on the firewall.

- *Pick one method to consistently specify the interface swap setting—in the bootstrap configuration, from the CLI on the firewall, or using the Amazon EC2 User data field on the AWS console—to prevent unpredictable behavior on the firewall.*
- *Ensure that you have access to the AWS console (management console or CLI) to view the IP address of the eth1 interface. Also, verify that the AWS Security Group rules allow connections (HTTPS and SSH) to the new management interface.*
- *Swap the management interface before you configure the firewall or define policy rules. If you have already configured the VM-Series firewall, check whether any IP address changes for eth0 and eth1 impact policy rules.*

# Performance Tuning for the VM-Series on AWS

Make sure that you do the following:

- Pick the correct AWS Instance Types for your deployment. For example, you cannot deploy the c4.xlarge EC2 instance type because the VM-Series firewall requires 9G memory with 2 or 4 vCPUs, and the instance type only supports 4 vCPUs and 7.5G memory.

Select the VM-Series Modelsand VM-Series Firewall Licenses for Public Cloudsthat best suits your deployment needs. For help with sizing, refer to this article..

Enable DPDK using the CLI command **`set system setting dpdk-pkt-io on`** or bootstrap the firewall to use DPDK at launch, see init-cfg.txt File Components, except if deploying the firewalls in an HA configuration.

*The C5 and M5 instance types that have the Elastic Network Adapter support SR-IOV mode only.*



© Palo Alto Networks, Inc.

# Deployments Supported on AWS

The VM-Series firewall secures inbound and outbound traffic to and from EC2 instances within the AWS Virtual Private Cloud (VPC). Because the AWS VPC only supports an IP network (Layer 3 networking capabilities), the VM-Series firewall can only be deployed with Layer 3 interfaces.

- Deploy the VM-Series firewall to secure the EC2 instances hosted in the AWS Virtual Private Cloud.

  If you host your applications in the AWS cloud, deploy the VM-Series firewall to protect and safely enable applications for users who access these applications over the internet. For example, the following diagram shows the VM-Series firewall deployed in the Edge subnet to which the internet gateway is attached. The application(s) are deployed in the private subnet, which does not have direct access to the internet.

  When users need to access the applications in the private subnet, the firewall receives the request and directs it to the appropriate application, after verifying security policy and performing Destination NAT. On the return path, the firewall receives the traffic, applies security policy and uses Source NAT to deliver the content to the user. See Use Case: Secure the EC2 Instances in the AWS Cloud.

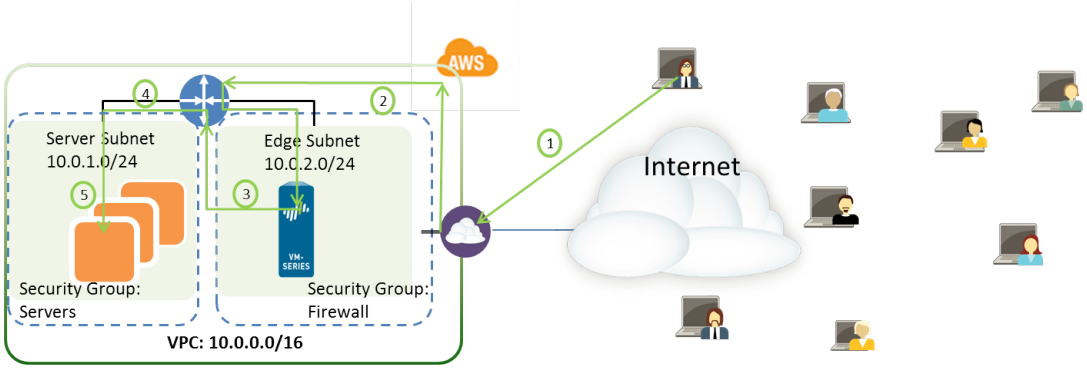



**Figure 1: VM-Series for EC2 Instances**

- Deploy the VM-Series firewall for VPN access between the corporate network and the EC2 instances within the AWS Virtual Private Cloud.

  To connect your corporate network with the applications deployed in the AWS Cloud, you can configure the firewall as a termination point for an IPSec VPN tunnel. This VPN tunnel allows users on your network to securely access the applications in the cloud.

  For centralized management, consistent enforcement of policy across your entire network, and for centralized logging and reporting, you can also deploy Panorama in your corporate network. If you need to set up VPN access to multiple VPCs, using Panorama allows you to group the firewalls by region and administer them with ease.



© Palo Alto Networks, Inc.

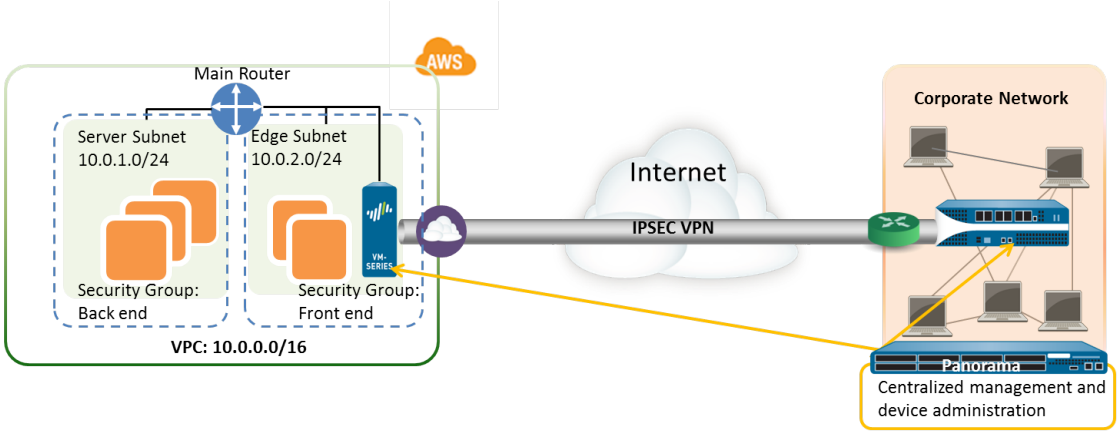



**Figure 2: VM-Series for VPN Access**

- Deploy the VM-Series firewall as a GlobalProtect gateway to secure access for remote users using laptops. The GlobalProtect agent on the laptop connects to the gateway, and based on the request, the gateway either sets up a VPN connection to the corporate network or routes the request to the internet. To enforce security compliance for users on mobile devices (using the GlobalProtect App), the GlobalProtect gateway is used in conjunction with the GlobalProtect Mobile Security Manager. The GlobalProtect Mobile Security Manager ensures that mobile devices are managed and configured with the device settings and account information for use with corporate applications and networks.

  *In each of the use cases above, you can deploy the VM-Series firewall in an active/passive high availability (HA) pair. For information on setting up the VM-Series firewall in HA, see* Use Case: Use Dynamic Address Groups to Secure New EC2 Instances within the VPC.

- Deploy the VM-Series firewall with the Amazon Elastic Load Balancing (ELB) service, whereby the firewall can receive dataplane traffic on the primary interface in the following scenarios where the VM-Series firewall is behind the Amazon ELB:
  - The VM-Series firewall(s) is securing traffic outbound directly to the internet without the need for using a VPN link or a Direct Connect link back to the corporate network.
  - The VM-Series firewall secures an internet-facing application when there is exactly one back-end server, such as a web server, for each firewall. The VM-Series firewalls and web servers can scale linearly, in pairs, behind ELB.

  If you want to Auto Scale VM-Series Firewalls with the Amazon ELB Service, use the CloudFormation Template available in the GitHub repository to deploy the VM-Series in an ELB sandwich topology with an internet-facing classic ELB and an either an internal classic load balancer or an internal application load balancer (internal ELB).


© Palo Alto Networks, Inc.

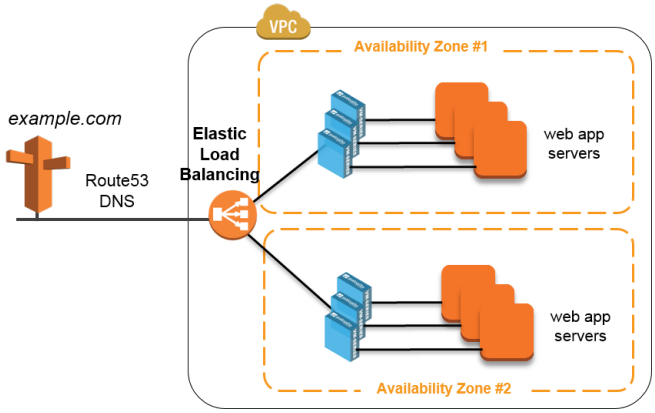



**Figure 3: VM-Series with ELB**

*You cannot configure the firewall to send and receive dataplane traffic on eth0 when the firewall is in front of ELB. The VM-Series firewall must be placed behind the Amazon ELB.*

*You can either* Use the VM-Series Firewall CLI to Swap the Management Interface *or enable it on bootstrap. For details, see* Management Interface Mapping for Use with Amazon ELB*.*

*If you want to deploy a load balancer sandwich topology, see* Auto Scale VM-Series Firewalls with the Amazon ELB Service*.*

*In addition to the links above that are covered under the Palo Alto Networks official support policy, Palo Alto Networks provides Community supported templates in the* Palo Alto Networks GitHub *repository that allow you to explore the solutions available to jumpstart your journey into cloud automation and scale on AWS. See* AWS Transit VPC *for a hub and subscribing VPC deployment that enables you to secure traffic between VPCs, between a VPC and an on-prem/hybrid cloud resource, and secure outbound traffic to the internet.*



© Palo Alto Networks, Inc.

# Deploy the VM-Series Firewall on AWS



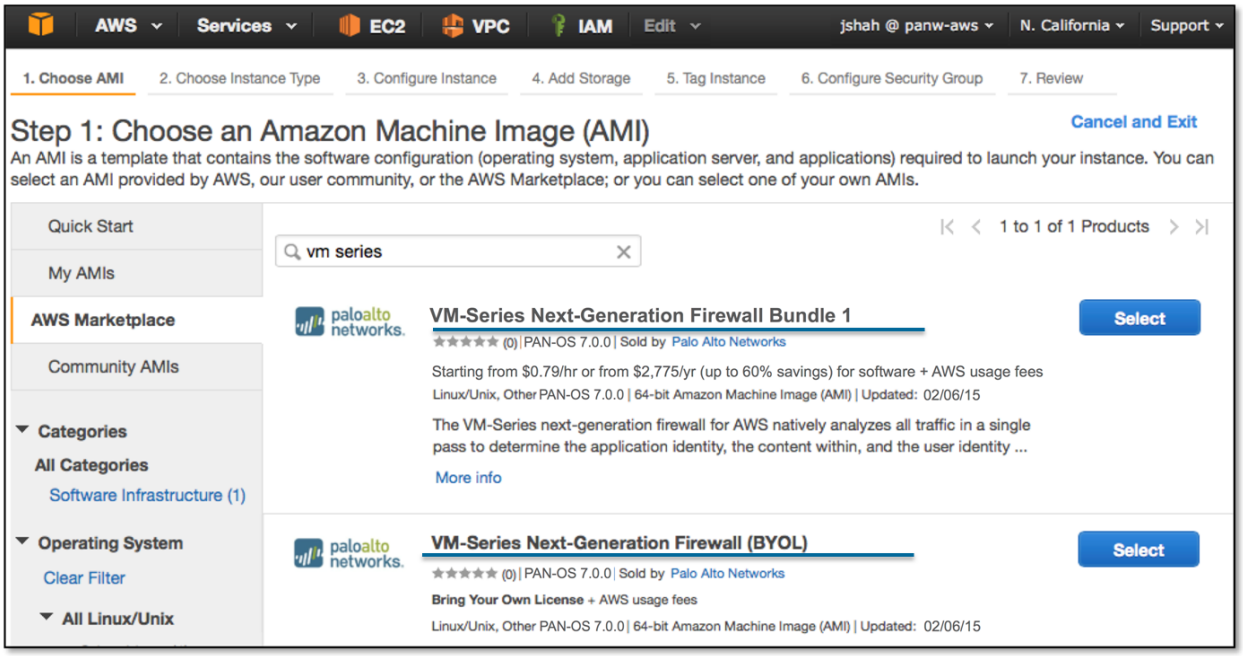
## Obtain the AMI

Get the Amazon Machine Image for the public AWS cloud and the AWS GovCloud from the respective Marketplace.



### *AMI in the Public AWS Cloud*

The AMI for the VM-Series firewall is available in the AWS Marketplace for both the Bring Your Own License (BYOL) and the Usage-based pricing options.




For purchasing licenses with the BYOL option, contact your Palo Alto Networks sales engineer or reseller.

### *AMI on AWS GovCloud*

The Bring Your Own License (BYOL) model and the usage-based model of the VM-Series firewall is available on the AWS GovCloud Marketplace.



© Palo Alto Networks, Inc.

With a GovCloud account, you can search for Palo Alto Networks and find the AMIs for the VM-Series firewall on the Marketplace. Make sure to review the supported EC2 instance types before you launch the firewall. For details, see Launch the VM-Series Firewall on AWS.

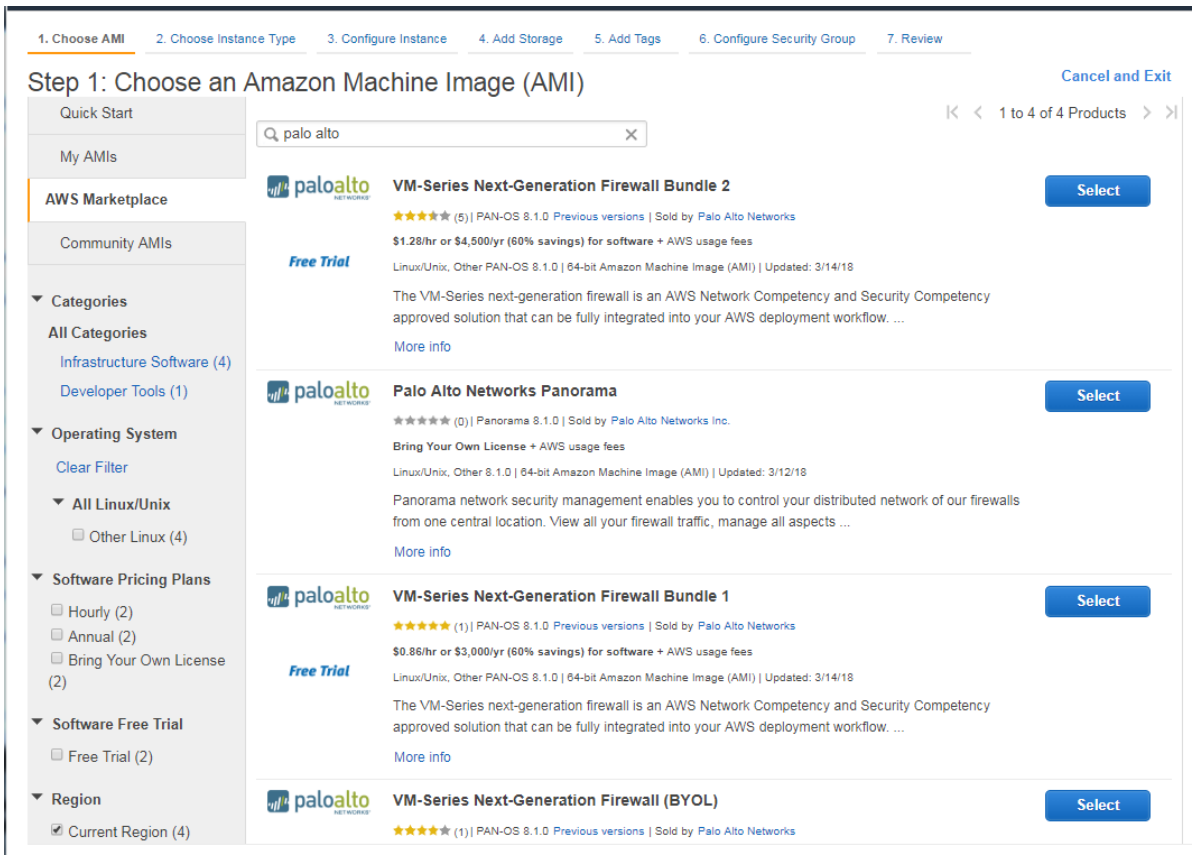



**Table 1: Review System Requirements and Limitations for VM-Series on AWS**

| Requirement | Details |
|---|---|
| EC2 instance types | The EC2 instance type you select must meet the VM-Series System Requirements for the VM-Series firewall model. If you deploy the VM-Series firewall on an EC2 instance type that does not meet these requirements, the firewall will boot into maintenance mode<br><br>*To support VM Monitoring and high availability on AWS, the VM-Series firewall must be able to directly reach the AWS API service endpoints without any proxy servers between the firewall management interface and the AWS API endpoints (such as ec2.us-west-2.amazonaws.com).* |
| Amazon Elastic Block Storage (EBS) | The VM-Series firewall must use the Amazon Elastic Block Storage (EBS) volume for storage. EBS optimization provides an optimized configuration stack and additional, dedicated capacity for Amazon EBS I/O. |
| Networking | Because the AWS only supports Layer 3 networking capabilities, the VM-Series firewall can only be deployed with Layer 3 interfaces. Layer 2 |



© Palo Alto Networks, Inc.

| Requirement | Details |
|---|---|
| | interfaces, virtual wire, VLANs, and subinterfaces are not supported on the VM-Series firewall deployed in the AWS VPC. |
| Interfaces | Support for a total of eight interfaces is available—one management interface and a maximum of seven Elastic Network Interfaces (ENIs) for data traffic. The VM-Series firewall does not support hot attachment of ENIs; to detect the addition or removal of an ENI you must reboot the firewall.<br><br>*Your EC2 instance type selection determines the total number of ENIs you can enable. For example, the c3.8xlarge supports eight (8) ENIs.* |
| Support entitlement and Licenses | For the Bring Your Own License model, a support account and a valid VM-Series license are required to obtain the Amazon Machine Image (AMI) file, which is required to install the VM-Series firewall in the AWS VPC. The licenses required for the VM-Series firewall—capacity license, support license, and subscriptions for Threat Prevention, URL Filtering, WildFire, etc—must be purchased from Palo Alto Networks. To purchase the licenses for your deployment, contact your sales representative. See VM-Series Firewall Licenses for Public Clouds.<br><br>For the usage-based licensing model, hourly and annual pricing bundles can be purchased and billed directly to AWS. You must however, register your support entitlement with Palo Alto Networks. For details see, Register the Usage-Based Model of the VM-Series Firewall for Public Clouds (no auth code). |

## Get the VM-Series Firewall Amazon Machine Image (AMI) ID

Use the following instructions to find the AMI ID for the VM-Series firewall that matches the PAN-OS version, license type, and AWS region in which you want to launch the VM-Series firewall.

STEP 1 | Install AWS CLI on the client that you are using to retrieve the AMI ID, and login with your AWS credentials.

Refer to the AWS documentation for instructions on installing the CLI.

STEP 2 | Find the AMI-ID with the following CLI command.

```
aws ec2 describe-images --filters "Name=product-code,Values=<license-type-value>" Name=name,Values=PA-VM-AWS*<PAN-OS-version>*  --region <region> --output json
```

You need to get replace the value in the angle brackets <> with the relevant information as shown below:

- Use the VM-Series product code for each license type. The values are:
  - Bundle 1—6kxdw3bbmdeda3o6i1ggqt4km
  - Bundle 2—806j2of0qy5osgjjixq9gqc6g
  - BYOL—6njl1pau431dv1qxipg63mvah
- Use the PAN-OS version—7.0, 8.0, 8.1, 9.0. If there are multiple feature releases within a PAN-OS version all the AMI-IDs are listed for you. For example, in 8.0.x, you will view a listing of the AMI IDs



© Palo Alto Networks, Inc.

for PAN-OS versions 8.0, 8.0.3, 8.0.8, 8.0.9, and 8.0.13 and you can use the AMI-ID for the PAN-OS version you need.

- Get the AWS region details from: ﷺhttps://docs.aws.amazon.com/general/latest/gr/rande.html.

For example: To find the AMI-ID for the VM-Series Bundle 1 for PAN-OS 9.0 in US California region, the CLI command is:

```
aws ec2 describe-images --filters "Name=product-
code,Values=6kxdw3bbmdeda3o6i1ggqt4km" Name=name,Values=PA-VM-AWS*9.0*  --
region us-west-1 --output json
```

The output is:

```
"Images": [                                    {
                                              "Architecture":
 "x86_64",                                                  "CreationDate":
 "2019-02-26T14:17:21.000Z",
    "ImageId": "ami-045f8b6e430535f0d",
                                                "ImageLocation": "aws-
marketplace/PA-VM-AWS-9.0.0-6f2a9521-7dc3-46cc-8891-8c4d02d29666-
ami-054da040447f62b2c
```

# Planning Worksheet for the VM-Series in the AWS VPC

For ease of deployment, plan the subnets within the VPC and the EC2 instances that you want to deploy within each subnet. Before you begin, use the following table to collate the network information required to deploy and insert the VM-Series firewall into the traffic flow in the VPC:

| Configuration Item | Value |
|---|---|
| VPC CIDR | |
| Security Groups | |
| Subnet (public) CIDR | |
| Subnet (private) CIDR | |
| Subnet (public) Route Table | |
| Subnet (private) Route Table | |
| Security Groups<br>• Rules for Management Access to the firewall (eth0/0)<br>• Rules for access to the dataplane interfaces of the firewall<br>• Rules for access to the interfaces assigned to the application servers. | |
| VM-Series firewall behind ELB | |



© Palo Alto Networks, Inc.

| Configuration Item | Value |
| --- | --- |
| EC2 Instance 1 (VM-Series firewall)<br><br>*An EIP is only required for the dataplane interface that is attached to the public subnet.* | Subnet:<br>Instance type:<br>Mgmt interface IP:<br>Mgmt interface EIP:<br>Dataplane interface eth1/1<br>• Private IP:<br>• EIP (if required):<br>• Security Group:<br>Dataplane interface eth1/2<br>• Private IP:<br>• EIP (if required):<br>• Security Group: |
| EC2 Instance 2 (Application to be secured)<br><br>Repeat these set of values for additional application(s) being deployed. | Subnet:<br>Instance type:<br>Mgmt interface IP:<br>Default gateway:<br>Dataplane interface 1<br>• Private IP |
| **Requirements for HA** | If you are deploying the VM-Series firewalls in a high availability (active/passive) configuration, you must ensure the following:<br>• Create an IAM role and assign the role to the VM-Series firewall when you are deploying the instance. See IAM Roles for HA.<br>• Deploy the HA peers in the same AWS availability zone.<br>• The active firewall in the HA pair must have at a minimum three ENIs: two dataplane interfaces and one management interface.<br>The passive firewall in the HA pair, must have one ENI for management, and one ENI that functions as dataplane interface; you will configure the dataplane interface as an HA2 interface.<br><br>*Do not attach additional dataplane interfaces to the passive firewall in the HA pair. On failover, the dataplane interfaces from the previously active firewall are moved — detached and then attached—to the now active (previously passive) firewall.* |


© Palo Alto Networks, Inc.

# Launch the VM-Series Firewall on AWS

If you have not already registered the capacity authcode that you received with the order fulfillment email, with your support account, see Register the VM-Series Firewall. After registering, deploy the VM-Series firewall using an AMI published in the Marketplace or Create a Custom Amazon Machine Image (AMI) in the AWS VPC as follows:

STEP 1 | Access the AWS Console.

Log in to the AWS console and select the EC2 Dashboard.

STEP 2 | Set up the VPC for your network needs.

Whether you launch the VM-Series firewall in an existing VPC or you create a new VPC, the VM-Series firewall must be able to receive traffic from the EC2 instances and perform inbound and outbound communication between the VPC and the internet.

Refer to the AWS VPC documentation for instructions on creating a VPC and setting it up for access.

For an example with a complete workflow, see Use Case: Secure the EC2 Instances in the AWS Cloud.

1. Create a new VPC or use an existing VPC. Refer to the AWS Getting Started documentation.
2. Verify that the network and security components are defined suitably.
   - Enable communication to the internet. The default VPC includes an internet gateway, and if you install the VM-Series firewall in the default subnet it has access to the internet.
   - Create subnets. Subnets are segments of the IP address range assigned to the VPC in which you can launch the EC2 instances. The VM-Series firewall must belong to the public subnet so that it can be configured to access the internet.
   - Create security groups as needed to manage inbound and outbound traffic from the EC2 instances/subnets.
   - Add routes to the route table for a private subnet to ensure that traffic can be routed across subnets and security groups in the VPC, as applicable.
3. If you want to deploy a pair of VM-Series firewalls in HA, you must define IAM Roles for HA before you can Configure Active/Passive HA on AWS.
4. (Optional) If you are using bootstrapping to perform the configuration of your VM-Series firewall, refer to Bootstrap the VM-Series Firewall on AWS. For more information about bootstrapping, see Bootstrap the VM-Series Firewall.

STEP 3 | Launch the VM-Series firewall.

*Although you can add additional network interfaces (ENIs) to the VM-Series firewall when you launch, AWS releases the auto-assigned Public IP address for the management interface when you restart the firewall. Hence, to ensure connectivity to the management interface you must assign an Elastic IP address for the management interface, before attaching additional interfaces to the firewall.*

If you want to conserve EIP addresses, you can assign one EIP address to the eth 1/1 interface and use this interface for both management traffic and data traffic. To restrict services permitted on the interface or limit IP addresses that can log in the eth 1/1 interface, attach a management profile to the interface.

1. On the EC2 Dashboard, click **Launch Instance**.
2. Select the VM-Series AMI. To get the AMI, see Obtain the AMI.
3. Launch the VM-Series firewall on an EC2 instance.
   1. Choose the **EC2 instance type** for allocating the resources required for the firewall, and click **Next**. See VM-Series System Requirements, for resource requirements.



© Palo Alto Networks, Inc.

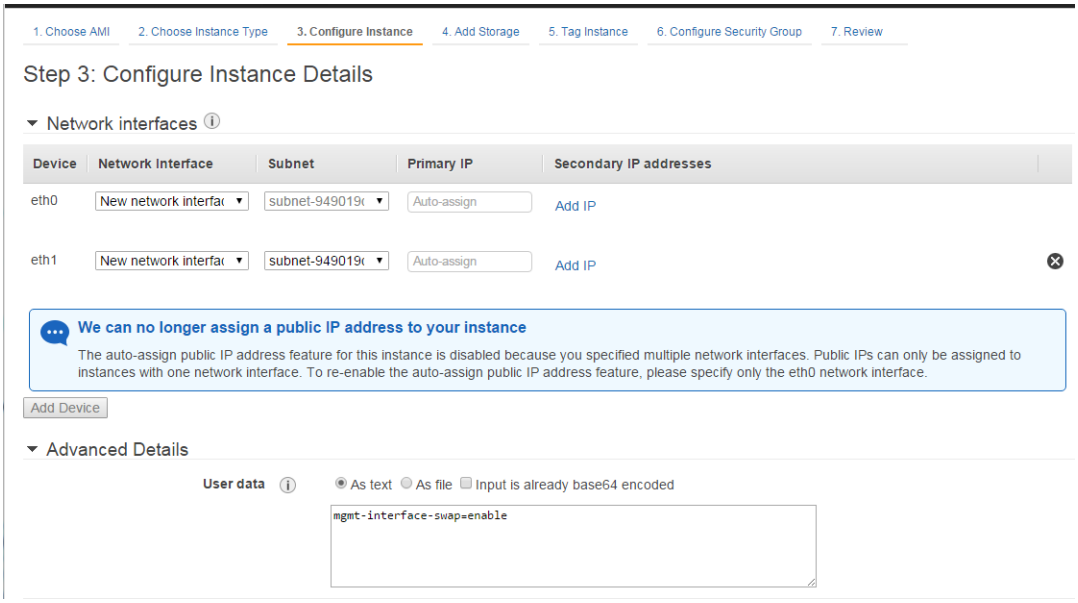
2. Select the VPC.
3. Select the public subnet to which the VM-Series management interface will attach.
4. Select **Automatically assign a public IP address**. This allows you to obtain a publicly accessible IP address for the management interface of the VM-Series firewall.

   You can later attach an Elastic IP address to the management interface; unlike the public IP address that is disassociated from the firewall when the instance is terminated, the Elastic IP address provides persistence and can be reattached to a new (or replacement) instance of the VM-Series firewall without the need to reconfigure the IP address wherever you might have referenced it.
5. Select **Launch as an EBS-optimized instance**.
6. Add another network interface for deployments with ELB so that you can swap the management and data interfaces on the firewall. Swapping interfaces requires a minimum of two ENIs (eth0 and eth1).

   - Expand the Network Interfaces section and click **Add Device** to add another network interface.

     Make sure that your VPC has more than one subnet so that you can add additional ENIs at launch.

     *If you launch the firewall with only one ENI:*
     - *The interface swap command will cause the firewall to boot into maintenance mode.*
     - *You must reboot the firewall when you add the second ENI.*
   - Expand the Advanced Details section and in the User data field enter **`mgmt-interface-swap=enable`** as text to perform the interface swap during launch.




7. Accept the default **Storage** settings. The firewall uses volume type SSD (gp2)

   *This key pair is required for first time access to the firewall. It is also required to access the firewall in maintenance mode.*
8. (Optional) **Tagging**. Add one or more tags to create your own metadata to identify and group the VM-Series firewall. For example, add a **Name** tag with a **Value** that helps you remember that the ENI interfaces have been swapped on this VM-Series firewall.


© Palo Alto Networks, Inc.

9. Select an existing **Security Group** or create a new one. This security group is for restricting access to the management interface of the firewall. At a minimum consider enabling https and ssh access for the management interface.
10. If prompted, select an appropriate **SSD** option for your setup.
11. Select **Review and Launch**. Review that your selections are accurate and click **Launch**.
12. Select an existing key pair or create a new one, and acknowledge the key disclaimer.
13. Download and save the private key to a safe location; the file extension is `.pem`. You cannot regenerate this key, if lost.

    It takes 5-7 minutes to launch the VM-Series firewall. You can view the progress on the EC2 Dashboard.When the process completes, the VM-Series firewall displays on the **Instances** page of the EC2 Dashboard.

STEP 4 | Configure a new administrative password for the firewall.

*On the VM-Series firewall CLI, you must configure a unique administrative password before you can access the web interface of the firewall. To log in to the CLI, you require the private key that you used to launch the firewall.*

1. Use the public IP address to SSH into the Command Line Interface (CLI) of the VM-Series firewall. You will need the private key that you used or created in 3 above to access the CLI.

   *If you added an additional ENI to support deployments with ELB, you must first create and assign an Elastic IP address to the ENI to access the CLI, see step 6.*

   If you are using PuTTY for SSH access, you must convert the .pem format to a .ppk format. See https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html
2. Enter the following command to log in to the firewall:

   **ssh-i *<private_key.pem>* admin@*<public-ip_address>***
3. Configure a new password, using the following command and follow the onscreen prompts:

   **configure**

   **set mgt-config users admin password**
4. If you have a BYOL that needs to be activated, set the DNS server IP address so that the firewall can aceess the Palo Alto Networks licensing server. Enter the following command to set the DNS server IP address:

   **set deviceconfig system dns-setting servers primary *<ip_address>***
5. Commit your changes with the command:

   **commit**
6. Terminate the SSH session.

STEP 5 | Shutdown the VM-Series firewall.

1. On the EC2 Dashboard, select **Instances**.
2. From the list, select the VM-Series firewall and click **Actions** > **Stop**.

STEP 6 | Create and assign an Elastic IP address (EIP) to the ENI used for management access to the firewall and reboot the VM-Series firewall.

1. Select **Elastic IPs** and click **Allocate New Address**.
2. Select **EC2-VPC** and click **Yes, Allocate**.
3. Select the newly allocated EIP and click **Associate Address**.
4. Select the **Network Interface** and the **Private IP address** associated with the management interface and click **Yes, Associate**.



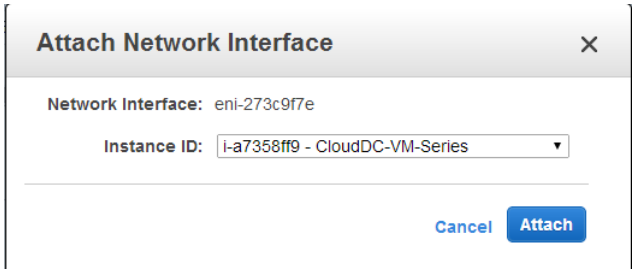
© Palo Alto Networks, Inc.

STEP 7 | Create virtual network interface(s) and attach the interface(s) to the VM-Series firewall. The virtual network interfaces are called Elastic Network Interfaces (ENIs) on AWS, and serve as the dataplane network interfaces on the firewall. These interfaces are used for handling data traffic to/from the firewall.

You will need at least two ENIs that allow inbound and outbound traffic to/from the firewall. You can add up to seven ENIs to handle data traffic on the VM-Series firewall; check your EC2 instance type to verify the maximum number supported on it.

1. On the EC2 Dashboard, select **Network Interfaces**, and click **Create Network Interface**.
2. Enter a descriptive name for the interface.
3. Select the subnet. Use the subnet ID to make sure that you have selected the correct subnet. You can only attach an ENI to an instance in the same subnet.
4. Enter the **Private IP** address to assign to the interface or select **Auto-assign** to automatically assign an IP address within the available IP addresses in the selected subnet.
5. Select the **Security group** to control access to the dataplane network interface.
6. Click **Yes, Create**.


7. To attach the ENI to the VM-Series firewall, select the interface you just created, and click **Attach**.




8. Select the **Instance ID** of the VM-Series firewall, and click **Attach**.
9. Repeat the steps above for creating and attaching at least one more ENI to the firewall.

STEP 8 | (Not required for the Usage-based licensing model) Activate the licenses on the VM-Series firewall.

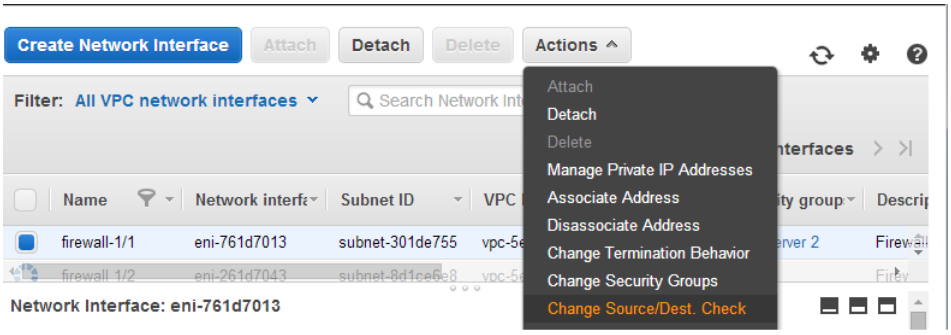
*This task is not performed on the AWS management console. Access to the Palo Alto Networks support portal and the web interface of the VM-Series firewall is required for license activation.*

See Activate the License.

STEP 9 | Disable Source/Destination check on every firewall dataplane network interface(s). Disabling this option allows the interface to handle network traffic that is not destined to the IP address assigned to the network interface.

1. On the EC2 Dashboard, select the network interface, for example eth1/1, in the **Network Interfaces** tab.
2. In the **Action** drop-down, select **Change Source/Dest. Check**.


© Palo Alto Networks, Inc.




3. Click **Disabled** and **Save** your changes.
4. Repeat Steps 1-3 for each firewall dataplane interface.

STEP 10 | Configure the dataplane network interfaces as Layer 3 interfaces on the firewall.

For an example configuration, see 14 through 17 in Use Case: Secure the EC2 Instances in the AWS Cloud.

*On the application servers within the VPC, define the dataplane network interface of the firewall as the default gateway.*

1. Using a secure connection (https) from your web browser, log in using the EIP address and password you assigned during initial configuration (https://<Elastic_IP address>). You will see a certificate warning; that is okay. Continue to the web page.
2. Select **Network** > **Interfaces** > **Ethernet**.
3. Click the link for **ethernet 1/1** and configure as follows:
   - **Interface Type**: **Layer3**
   - On the **Config** tab, assign the interface to the default router.
   - On the **Config** tab, expand the **Security Zone** drop-down and select **New Zone**. Define a new zone, for example VM_Series_untrust, and then click **OK**.
   - On the **IPv4** tab, select either **Static** or **DHCP Client**.

     If using the **Static** option, click **Add** in the IP section, and enter the IP address and network mask for the interface, for example 10.0.0.10/24.

     Make sure that the IP address matches the ENI IP address that you assigned earlier.

     If using DHCP, select **DHCP Client**; the private IP address that you assigned to the ENI in the AWS management console will be automatically acquired.
4. Click the link for **ethernet 1/2** and configure as follows:
   - **Interface Type**: **Layer3**
   - Security Zone: VM_Series_trust
   - **IP address**: Select the **Static** or **DHCP Client** radio button.

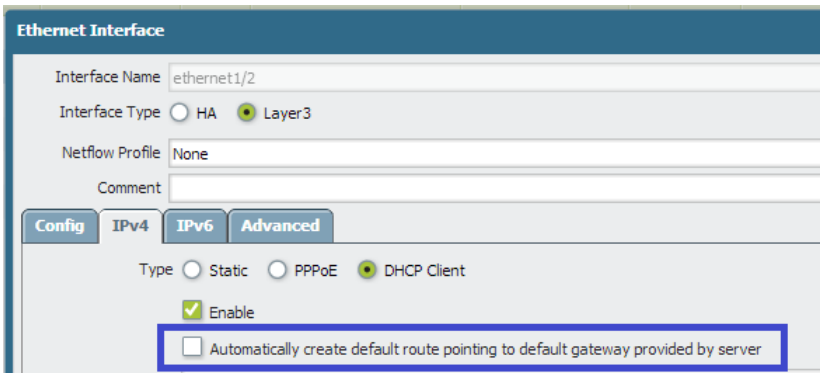
     For static, click **Add** in the IP section, and enter the IP address and network mask for the interface. Make sure that the IP address matches the attached ENI IP address that you assigned earlier.
5. Click **Commit**. Verify that the link state for the interfaces are up.


*For DHCP, clear the Automatically create default route to default gateway provided by server check box. For an interface that is attached to the private subnet in the VPC,*



© Palo Alto Networks, Inc.

*disabling this option ensures that traffic handled by this interface does not flow directly to the internet gateway on the VPC.*




STEP 11 | Create NAT rules to allow inbound and outbound traffic from the servers deployed within the VPC.

1. Select **Policies** > **NAT** on the web interface of the firewall.
2. Create a NAT rule to allow traffic from the dataplane network interface on the firewall to the web server interface in the VPC.
3. Create a NAT rule to allow outbound access for traffic from the web server to the internet.

STEP 12 | Create security policies to allow/deny traffic to/from the servers deployed within the VPC.

1. Select **Policies** > **Security** on the web interface of the firewall.
2. Click **Add**, and specify the zones, applications and logging options that you would like to execute to restrict and audit traffic traversing through the network.

STEP 13 | Commit the changes on the firewall.

Click **Commit**.

STEP 14 | Verify that the VM-Series firewall is securing traffic and that the NAT rules are in effect.

1. Select **Monitor** > **Logs** > **Traffic** on the web interface of the firewall.
2. View the logs to make sure that the applications traversing the network match the security policies you implemented.

# Create a Custom Amazon Machine Image (AMI)

A custom VM-Series AMI gives you the consistency and flexibility to deploy a VM-Series firewall with the PAN-OS version you want to use on your network instead of being restricted to using only an AMI that is published to the AWS public Marketplace or to the AWS GovCloud Marketplace. Using a custom AMI speeds up the process of deploying a firewall with the PAN-OS version of your choice because it reduces the time to provision the firewall with an AMI published on the AWS public or AWS GovCloud marketplace, and then performing software upgrades to get to the PAN-OS version you have qualified or want to use on your network. Additionally, you can then use the custom AMI in the Auto Scaling VM-Series Firewalls CloudFormation Templates or any other templates that you have created.

You can create a custom AMI with the BYOL, Bundle 1, or Bundle 2 licenses. The process of creating a custom AMI requires you to remove all configuration from the firewall and reset it to factory defaults, so in this workflow you'll launch a new instance of the firewall from the AWS Marketplace instead of using an existing firewall that you have fully configured.


© Palo Alto Networks, Inc.

*When creating a custom AMI with a BYOL version of the firewall, you must first activate the license on the firewall so that you can access and download PAN-OS software updates to upgrade your firewall, and then deactivate the license on the firewall before you reset the firewall to factory defaults and create the custom AMI. If you do not deactivate the license, you lose the license that you applied on this firewall instance.*

STEP 1 | Launch the VM-Series firewall from the Marketplace.

See 3

STEP 2 | Configure the administrative password on the firewall.

See 4

STEP 3 | (Only for BYOL) Activate the license.

STEP 4 | Install software updates and upgrade the firewall to the PAN-OS version you plan to use.

STEP 5 | (Only for BYOL) Deactivate the license.

STEP 6 | Perform a factory reset.

A factory reset allows you to remove any configuration on the firewall for the custom AMI.

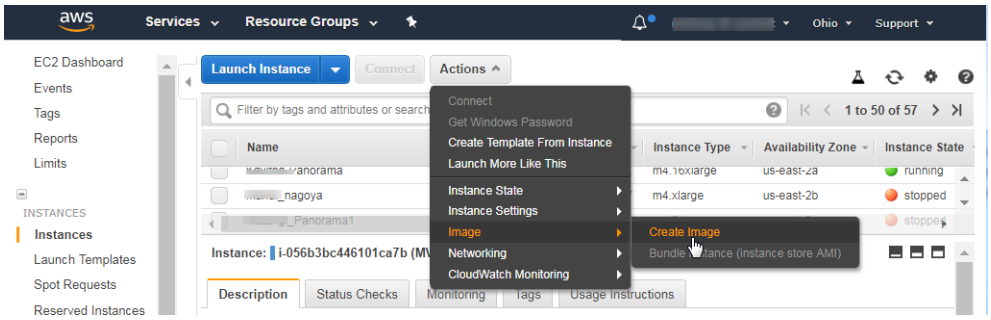
1. Access the firewall CLI.
2. Access the Maintenance Recovery Tool (MRT) to reboot the firewall in to maintenance mode.

   Use the CLI command **`debug system maintenance-mode`** and enter **`y`** to confirm. It will take approximately 2 to 3 minutes for the firewall to boot to the MRT. During this time, your SSH session will disconnect.
3. Log in as **ec2-user** and select the SSH public key that you used when you launched the firewall.
4. Select **Continue** > **Factory Reset** to access the menu.

*Do not reboot the firewall, otherwise you will need to start over again.*

STEP 7 | Create the custom AMI.

1. Log in to the AWS Console and select the EC2 Dashboard.
2. **Stop** the VM-Series firewall.
3. Select the VM-Series firewall instance, and click **Image** > **Create Image**.

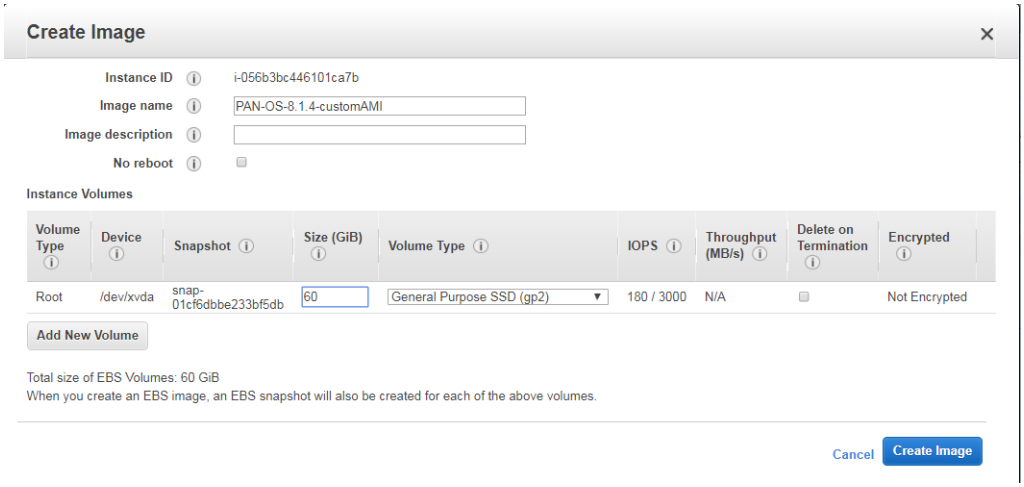



4. Enter a custom image name, and click **Create Image**.

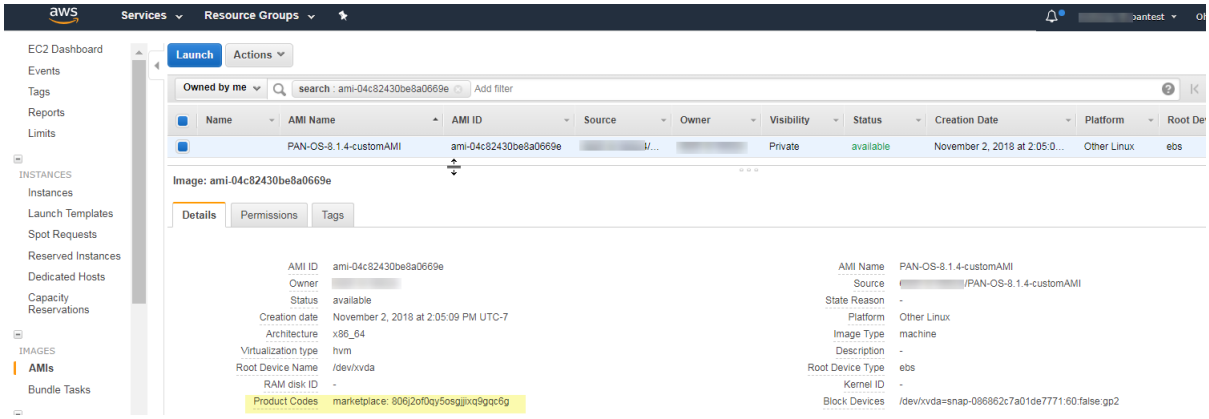
   The disk space of 60GB is the minimum requirement.



© Palo Alto Networks, Inc.




5. Verify that the custom AMI is created and has the correct product code.
    1. On the EC2 Dashboard, select **AMI**.
    2. Select the AMI that you just created. Depending on whether you selected an AMI with the BYOL, Bundle 1, or Bundle 2 licensing options, you should see one of the following **Product Codes** in the details:
        - BYOL—6njl1pau431dv1qxipg63mvah
        - Bundle 1—6kxdw3bbmdeda3o6i1ggqt4km
        - Bundle 2—806j2of0qy5osgjjixq9gqc6g




STEP 8 | Encrypt EBS Volume for the VM-Series Firewall on AWS.

If you plan to use the custom AMI with EBS encryption for an Auto Scale VM-Series Firewalls with the Amazon ELB Service deployment, you must use the default master key for your AWS account.

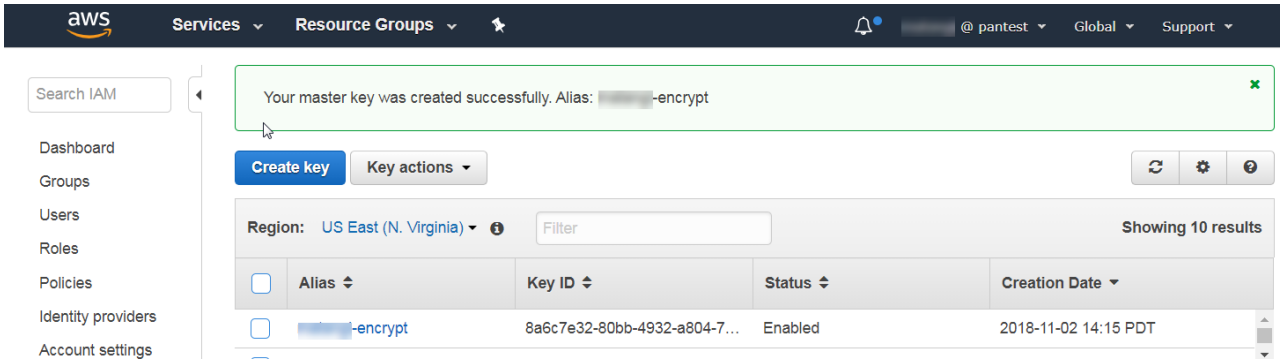
# Encrypt EBS Volume for the VM-Series Firewall on AWS

EBS encryption is available for all AWS EC2 Instance Types on which you can deploy the VM-Series firewall. To securely store data on the VM-Series firewall on AWS, you must first create a copy of an AMI that is published on the AWS public or GovCloud Marketplace, or use a custom AMI, and then encrypt the EBS volume with a customer master key (CMK) on the AWS Key Management Service (KMS). You can use the default master key for your AWS account or any CMK that you have previously created using the AWS Key Management Service, and EBS the KMS interact to ensure data security.


© Palo Alto Networks, Inc.

STEP 1 | Create an encryption key on AWS or skip this step if you want to use the default master key for your account.

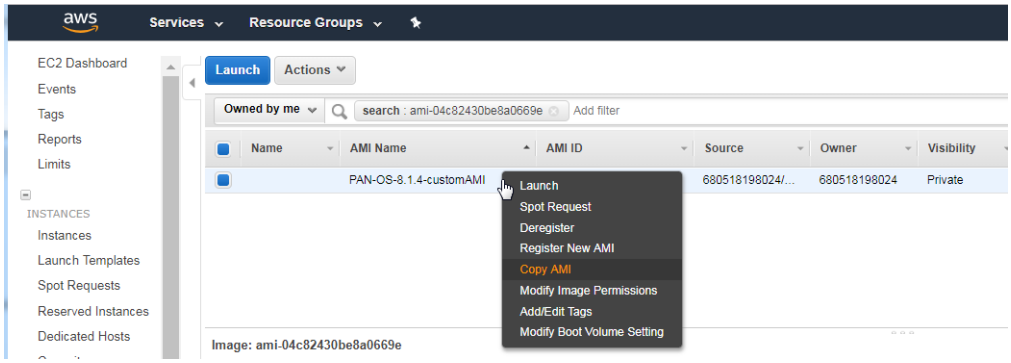
You will use this key to encrypt the EBS volume on the firewall. Note that the key is region specific.




STEP 2 | Use the key to encrypt the EBS volume on the firewall.

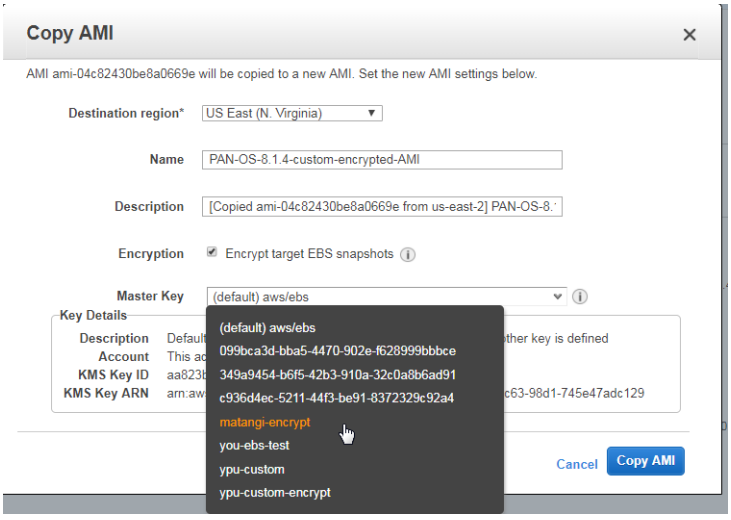
You must create a copy of the AMI that you want to encrypt. You can copy an AMI that is published on the AWS public or GovCloud Marketplace, or use a custom AMI (Create a Custom Amazon Machine Image (AMI)).

1. On the EC2 Dashboard, select the AMI and **Copy AMI**.




2. Set the details for the AMI.

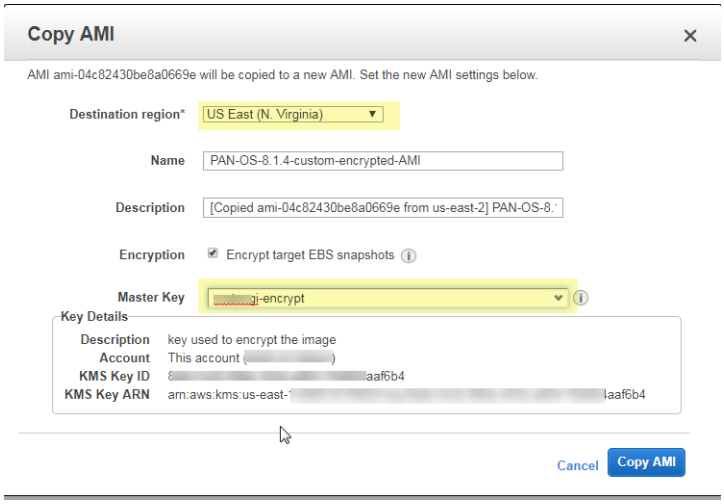
   Make sure to select **Encrypt target EBS snapshots**.



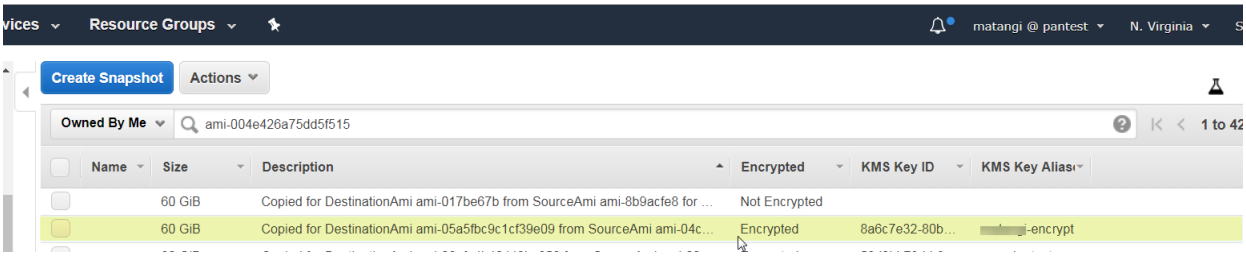

3. Select the encryption key and **Copy AMI** to create an encrypted EBS snapshot.



© Palo Alto Networks, Inc.




4. Select **EC2 Dashboard** > **Snapshots** to verify that the EBS snapshot is encrypted with the key you selected above.




# Use the VM-Series Firewall CLI to Swap the Management Interface

If you did not swap the management interface (MGT) with the dataplane interface (ethernet 1/1) when deploying the firewall, you can use the CLI to enable the firewall to receive dataplane traffic on the primary interface after launching the firewall.

STEP 1 | Complete Steps 1 through 7 in Launch the VM-Series Firewall on AWS.

*Before you proceed, verify that the firewall has a minimum of two ENIs (eth0 and eth1). If you launch the firewall with only one ENI, the interface swap command will cause the firewall to boot into maintenance mode.*

STEP 2 | On the EC2 Dashboard, view the IP address of the eth1 interface and verify that the AWS Security Group rules allow connections (HTTPS and SSH) to the new management interface (eth1).

STEP 3 | Log in to the VM-Series firewall CLI and enter the following command:

```
set system setting mgmt-interface-swap enable yes
```

STEP 4 | Confirm that you want to swap the interface and use the eth1 dataplane interface as the management interface.

STEP 5 | Reboot the firewall for the swap to take effect. Use the following command:

```
request restart system
```



© Palo Alto Networks, Inc.

STEP 6 | Verify that the interfaces have been swapped. Use the following command:

```
debug show vm-series interfaces all
Phoenix_interface   Base-OS_port   Base-OS_MAC        PCI-ID         Driver
mgt(interface-swap) eth0   0e:53:96:91:ef:29   0000:00:04.0   ixgbevf
Ethernet1/1           eth1    0e:4d:84:5f:7f:4d       0000:00:03.0
  ixgbevf
```

# Enable CloudWatch Monitoring on the VM-Series Firewall

The VM-Series firewall on AWS can publish native PAN-OS metrics to AWS CloudWatch, which you can use to monitor the firewalls. These metrics allow you to assess performance and usage patterns that you can use to take action for launching or terminating instances of the VM-Series firewalls.

The firewalls use AWS APIs to publish the metric to a *namespace* on AWS at a specified time interval. The namespace is the location to which CloudWatch collects and aggregates the selected metric for all instances configured to use the namespace. You can then monitor the metric in CloudWatch or create auto scaling policies to trigger alarms and take an action to manually deploy a new instance of the firewall when the monitored metric reaches a threshold value. Refer to the AWS CloudWatch and Auto Scaling Groups (ASG) documentation on best practices for setting the alarm conditions for a scale out or scale in action.

For a description on the PAN-OS metrics that you can publish to CloudWatch, see Custom PAN-OS Metrics Published for Monitoring.

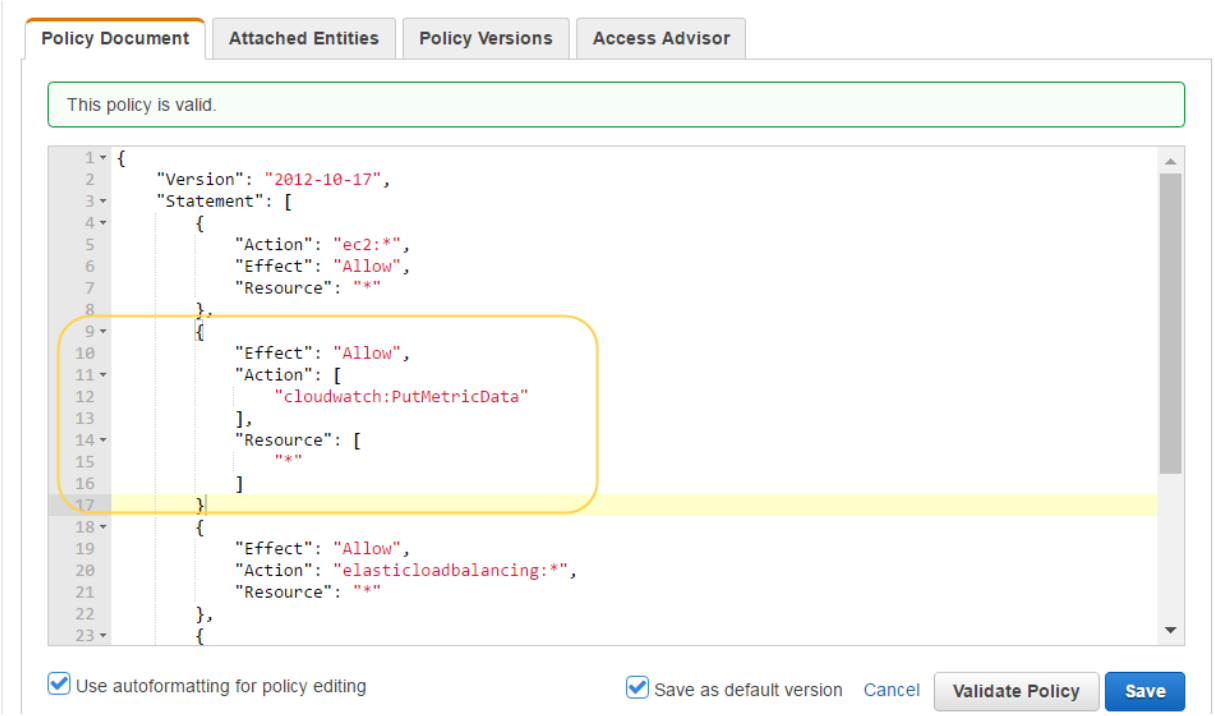
STEP 1 | Assign the appropriate permissions for the AWS Identity and Access Management (IAM) user role that you use to deploy the VM-Series firewall on AWS.

Whether you launch a new instance of the VM-Series firewall or upgrade an existing VM-Series firewall on AWS, the IAM role associated with your instance, must have permissions to publish metrics to CloudWatch.

1. On the AWS console, select IAM.
2. Edit the IAM role to grant the following permissions:

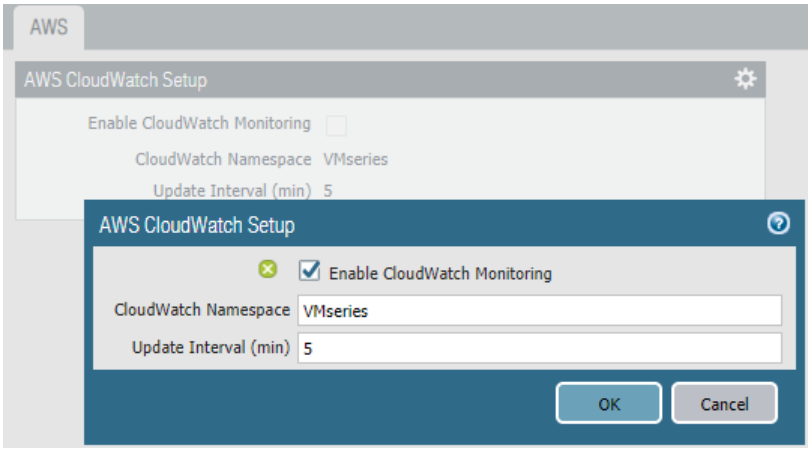


© Palo Alto Networks, Inc.




STEP 2 | Enable CloudWatch on the VM-Series firewall on AWS.

1. Log in to the web interface on the VM-Series firewall
2. Select **Device** > **VM-Series**.
3. In AWS CloudWatch Setup, click **Edit** (⚙) and select **Enable CloudWatch Monitoring**.
   1. Enter the **CloudWatch Namespace** to which the firewall can publish metrics. The namespace cannot begin with **AWS**.
   2. Set the **Update Interval** to a value between 1-60 minutes. This is the frequency at which the firewall publishes the metrics to CloudWatch. The default is 5 minutes.




4. **Commit** the changes.

   Until the firewall starts to publish metrics to CloudWatch, you cannot configure alarms for PAN-OS metrics.

STEP 3 | Verify that you can see the metrics on CloudWatch.

1. On the AWS console, select **CloudWatch** > **Metrics**, to view CloudWatch metrics by category.



© Palo Alto Networks, Inc.

2. From the Custom Metrics drop-down, select the namespace.
3. Verify that you can see PAN-OS metrics in the viewing list.

STEP 4 | Configure alarms and action for PAN-OS metrics on CloudWatch.

Refer to the AWS documentation: http://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/AlarmThatSendsEmail.html

A VM-Series firewall with bootstrap configuration will take about 7-9 minutes to be available for service. So, here are some examples on how to set alarms that trigger auto scaling for the VM-Series firewall:

- If you have deployed 2 instances of the VM-Series firewalls as Global Protect Gateways that secure remote users, use the GlobalProtect Gateway Active Tunnels metric. You can configure an alarm for when the number of active tunnels is greater than 300 for 15 minutes, you can deploy 2 new instances of the VM-Series firewall, which are bootstrapped and configured to serve as Global Protect Gateways.
- If you are using the firewall to secure your workloads in AWS, use the Session Utilization metric to scale in or scale out the firewall based on resource usage. You can configure an alarm for when the session utilization metric is greater than 60% for 15 minutes, to deploy one instance of the VM-Series instance firewall. And conversely, if Session Utilization is less than 50% for 30 minutes, terminate an instance of the VM-Series firewall.



© Palo Alto Networks, Inc.

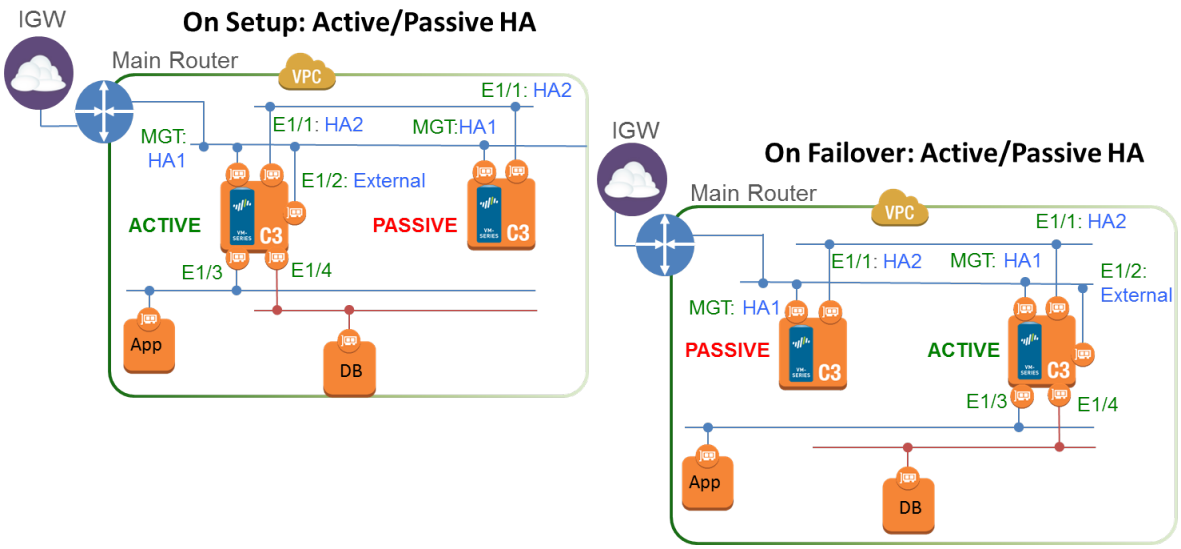
# High Availability for VM-Series Firewall on AWS

The VM-Series firewall on AWS supports active/passive HA only; if it is deployed with Amazon Elastic Load Balancing (ELB), it does not support HA (in this case ELB provides the failover capabilities).



## Overview of HA on AWS

To ensure redundancy, you can deploy the VM-Series firewalls on AWS in an active/passive high availability (HA) configuration. The active peer continuously synchronizes its configuration and session information with the identically configured passive peer. A heartbeat connection between the two devices ensures failover if the active device goes down. When the passive peer detects this failure it becomes active and triggers API calls to the AWS infrastructure to move all the dataplane interfaces (ENIs) from the failed peer to itself. The failover time can vary from 20 seconds to over a minute depending on the responsiveness from the AWS infrastructure.




## IAM Roles for HA

AWS requires that all API requests must be cryptographically signed using credentials issued by them. In order to enable API permissions for the VM-Series firewalls that will be deployed as an HA pair, you must create a policy and attach that policy to a role in the AWS Identity and Access Management (IAM) service. The role must be attached to the VM-Series firewalls at launch. The policy gives the IAM role permissions for initiating API actions for detaching and attaching network interfaces from the active peer in an HA pair to the passive peer when a failover is triggered.



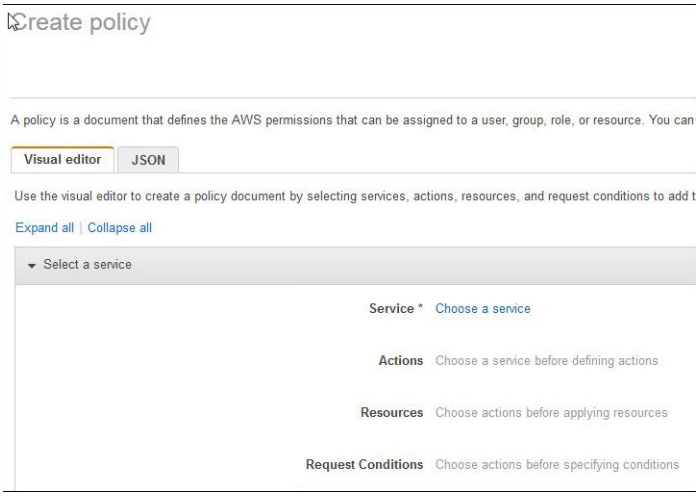
© Palo Alto Networks, Inc.

For detailed instructions on creating policy, refer to the AWS documentation on Creating Customer Managed Polices. For detailed instructions on creating an IAM role, defining which accounts or AWS services can assume the role, defining which API actions and resources the application can use upon assuming the role, refer to the AWS documentation on IAM Roles for Amazon EC2.

The IAM policy, which is configured in the AWS console, must have permissions for the following actions and resources (at a minimum):

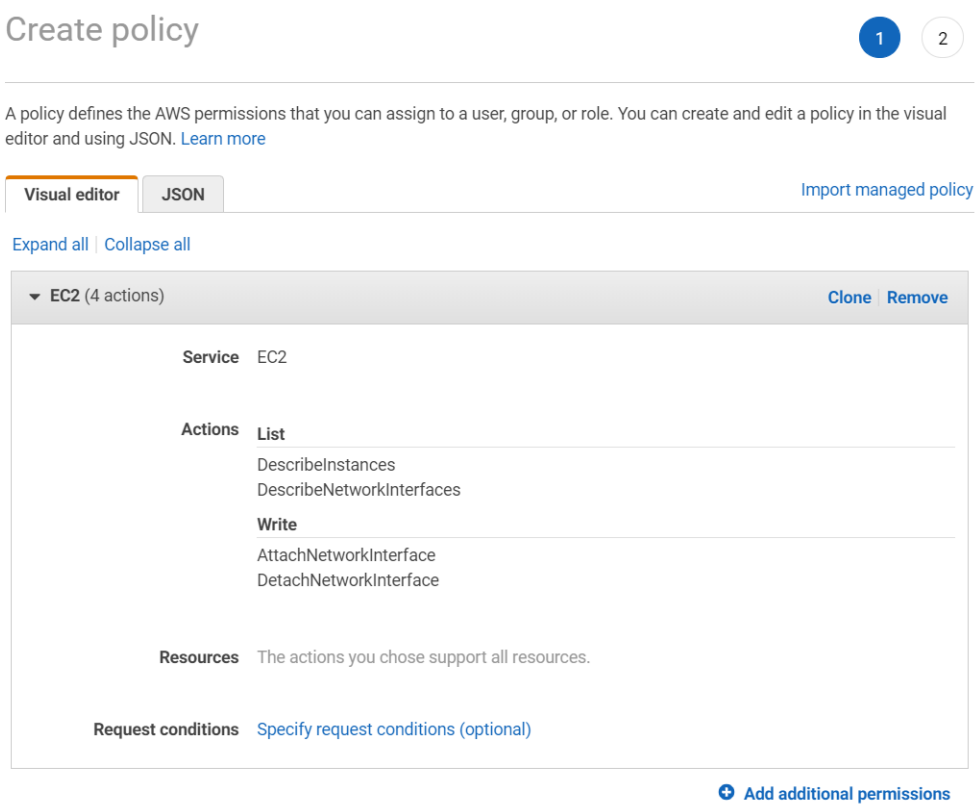
- AttachNetworkInterface—For permission to attach an ENI to an instance.
- DescribeNetworkInterface—For fetching the ENI parameters in order to attach an interface to the instance.
- DetachNetworkInterface—For permission to detach the ENI from the EC2 instance.
- DescribeInstances—For permission to obtain information on the EC2 instances in the VPC.
- Wild card (*)—In the Amazon Resource Name (ARN) field use the * as a wild card.

The following screenshot shows the access management settings for the IAM role described above:






© Palo Alto Networks, Inc.




The permissions you need are: `{"Version": "2012-10-17","Statement": [{"Sid": "VisualEditor0","Effect": "Allow","Action": ["ec2:AttachNetworkInterface","ec2:DetachNetworkInterface","ec2:DescribeInstances","ec2:I "*"}]}`

# HA Links

The devices in an HA pair use HA links to synchronize data and maintain state information. on AWS, the VM-Series firewall uses the following ports:

- **Control Link**—The HA1 link is used to exchange hellos, heartbeats, and HA state information, and management plane sync for routing and User-ID information. This link is also used to synchronize configuration changes on either the active or passive device with its peer.

  The Management port is used for HA1. TCP port 28769 and 28260 for cleartext communication; port 28 for encrypted communication (SSH over TCP).
- **Data Link**—The HA2 link is used to synchronize sessions, forwarding tables, IPSec security associations and ARP tables between devices in an HA pair. Data flow on the HA2 link is always unidirectional (except for the HA2 keep-alive); it flows from the active device to the passive device.

  Ethernet1/1 must be assigned as the HA2 link. The HA data link can be configured to use either IP (protocol number 99) or UDP (port 29281) as the transport.

The VM-Series on AWS does not support backup links for HA1 or HA2.

# Heartbeat Polling and Hello Messages

The firewalls use hello message and heartbeats to verify that the peer device is responsive and operational. Hello messages are sent from one peer to the other at the configured *Hello Interval* to verify the state of the device. The heartbeat is an ICMP ping to the HA peer over the control link, and the peer responds


© Palo Alto Networks, Inc.

to the ping to establish that the devices are connected and responsive. For details on the HA timers that trigger a failover, see HA Timers. (The HA timers for the VM-Series firewall are the same as that of the PA-5200 Series firewalls).

## Device Priority and Preemption

The devices in an HA pair can be assigned a *device priority* value to indicate a preference for which device should assume the active role and manage traffic upon failover. If you need to use a specific device in the HA pair for actively securing traffic, you must enable the preemptive behavior on both the firewalls and assign a device priority value for each device. The device with the lower numerical value, and therefore *higher priority*, is designated as active and manages all traffic on the network. The other device is in a passive state, and synchronizes configuration and state information with the active device so that it is ready to transition to an active state should a failure occur.

By default, preemption is disabled on the firewalls and must be enabled on both devices. When enabled, the preemptive behavior allows the firewall with the *higher priority* (lower numerical value) to resume as active after it recovers from a failure. When preemption occurs, the event is logged in the system logs.

## HA Timers

High availability (HA) timers are used to detect a firewall failure and trigger a failover. To reduce the complexity in configuring HA timers, you can select from three profiles: **Recommended**, **Aggressive**, and **Advanced**. These profiles auto-populate the optimum HA timer values for the specific firewall platform to enable a speedier HA deployment.

Use the **Recommended** profile for typical failover timer settings and the **Aggressive** profile for faster failover timer settings. The **Advanced** profile allows you to customize the timer values to suit your network requirements.

| HA Timer on the VM-Series on AWS | Default values for Recommended/Aggressive profiles |
|---|---|
| Promotion hold time | 2000/500 ms |
| Hello interval | 8000/8000 ms |
| Heartbeat interval | 2000/1000 ms |
| Max number of flaps | 3/3 |
| Preemption hold time | 1/1 min |
| Monitor fail hold up time | 0/0 ms |
| Additional master hold up time | 500/500 ms |

# Configure Active/Passive HA on AWS

STEP 1 | Make sure that you have followed the prerequisites.

For deploying a pair of VM-Series firewalls in HA in the AWS cloud, you must ensure the following:

- Select the IAM role you created when launching the VM-Series firewall on an EC2 instance; you cannot assign the role to an instance that is already running. See IAM Roles for HA.



© Palo Alto Networks, Inc.

For detailed instructions on creating an IAM role, defining which accounts or AWS services can assume the role, and defining which API actions and resources the application can use upon assuming the role, refer to the AWS documentation.

- DPDK cannot be enabled in a HA configuration. By default, DPDK is disabled on the VM-Series firewalls on AWS, and you do not need to disable it unless you enabled it manually.
- The active firewall in the HA pair must have at a minimum three ENIs: two dataplane interfaces and one management interface.

  The passive firewall in the HA pair, must have one ENI for management, and one ENI that functions as dataplane interface; you will configure the dataplane interface as an HA2 interface.

  *Do not attach additional dataplane interfaces to the passive firewall in the HA pair. On failover, the dataplane interfaces from the previously active firewall are moved — detached and then attached—to the now active (previously passive) firewall.*
- The HA peers must be deployed in the same AWS availability zone.

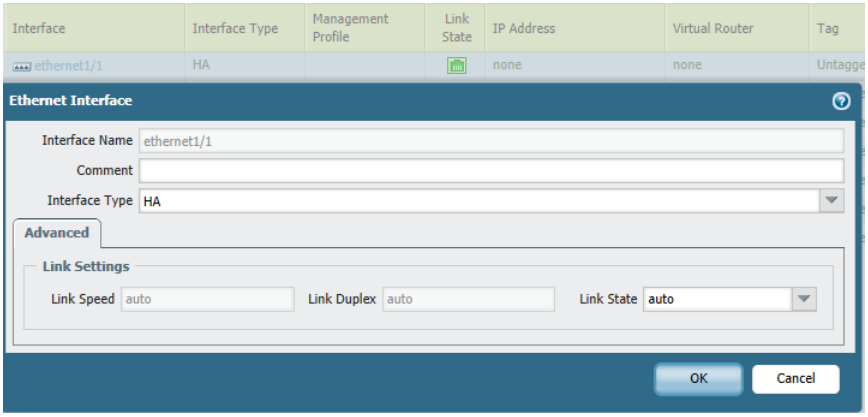
STEP 2 | Launch the VM-Series Firewall on AWS.

STEP 3 | Enable HA.

1. Select **Device** > **High Availability** > **General**, and edit the Setup section.
2. Select **Enable HA**.

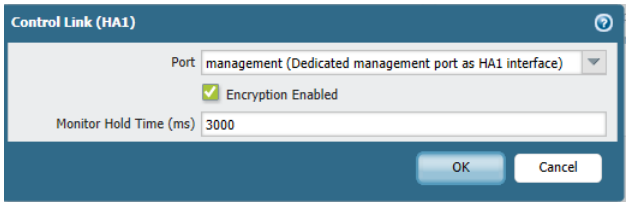
STEP 4 | Configure ethernet 1/1 as an HA interface. This interface must be used for HA2 communication.

1. Select **Network** > **Interfaces**.
2. Confirm that the link state is up on ethernet1/1.
3. Click the link for ethernet1/1 and set the **Interface Type** to HA.




STEP 5 | Set up the Control Link (HA1) to use the management port.

1. Select **Device** > **High Availability** > **General**, and edit the Control Link (HA1) section.




2. (Optional) Select **Encryption Enabled**, for secure HA communication between the peers. To enable encryption, you must export the HA key from a device and import it into the peer device.

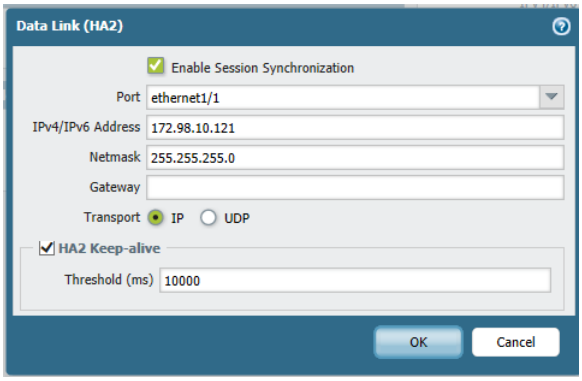


© Palo Alto Networks, Inc.

1. Select **Device** > **Certificate Management** > **Certificates**.
2. Select **Export HA key**. Save the HA key to a network location that the peer device can access.
3. On the peer device, navigate to **Device** > **Certificate Management** > **Certificates**, and select **Import HA key** to browse to the location that you saved the key and import it in to the peer device.

STEP 6 | Set up the Data Link (HA2) to use ethernet1/1.

1. Select **Device** > **High Availability** > **General**, edit the Data Link (HA2) section.
2. Select **Port** ethernet1/1.
3. Enter the IP address for ethernet1/1. This IP address must be the same that assigned to the ENI on the EC2 Dashboard.
4. Enter the **Netmask**.
5. Enter a **Gateway** IP address if the HA1 interfaces are on separate subnets.
6. Select **IP** or **UDP** for **Transport**. Use **IP** if you need Layer 3 transport (IP protocol number 99). Use **UDP** if you want the firewall to calculate the checksum on the entire packet rather than just the header, as in the IP option (UDP port 29281).




7. (Optional) Modify the **Threshold** for **HA2 Keep-alive** packets. By default, **HA2 Keep-alive** is enabled for monitoring the HA2 data link between the peers. If a failure occurs and this threshold (default is 10000 ms) is exceeded, the defined action will occur. A critical system log message is generated when an HA2 keep-alive failure occurs.

*You can configure the HA2 keep-alive option on both devices, or just one device in the HA pair. If you enable this option on one device, only that device will send the keep-alive messages.*

STEP 7 | Set the device priority and enable preemption.

Use this setting if you want to make sure that a specific device is the preferred active device. For information, see Device Priority and Preemption.

1. Select **Device** > **High Availability** > **General** and edit the Election Settings section.
2. Set the numerical value in **Device Priority**. Make sure to set a lower numerical value on the device that you want to assign a higher priority to.

*If both firewalls have the same device priority value, the firewall with the lowest MAC address on the HA1 control link will become the active device.*

3. Select **Preemptive**.

   You must enable preemptive on both the active and the passive device.
4. Modify the failover timers. By default, the HA timer profile is set to the **Recommended** profile and is suited for most HA deployments.



© Palo Alto Networks, Inc.

STEP 8 | (Optional) Modify the wait time before a failover is triggered.

1. Select **Device** > **High Availability** > **General** and edit the Active/Passive Settings.
2. Modify the **Monitor fail hold up time** to a value between 1-60 minutes; default is 1 minute. This is the time interval during which the firewall will remain active following a link failure. Use this setting to avoid an HA failover triggered by the occasional flapping of neighboring devices.

STEP 9 | Configure the IP address of the HA peer.

1. Select **Device** > **High Availability** > **General**, and edit the Setup section.
2. Enter the IP address of the HA1 port on the peer. This is the IP address assigned to the management interface (ethernet 0/0), which is also the HA1 link on the other firewall.
3. Set the **Group ID** number between 1 and 63. Although this value is not used on the VM-Series firewall on AWS, but cannot leave the field blank.

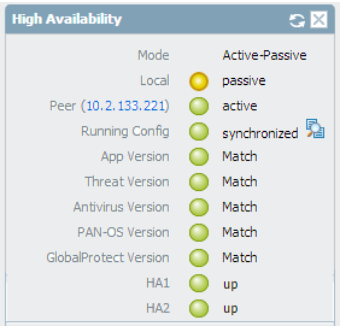
STEP 10 | Configure the other peer.

Repeat steps 3 to 9 on the HA peer.

STEP 11 | After you finish configuring both devices, verify that the devices are paired in active/passive HA.

1. Access the **Dashboard** on both devices, and view the **High Availability** widget.
2. On the active device, click the **Sync to peer** link.
3. Confirm that the devices are paired and synced, as shown below:

   - On the passive device: The state of the local device should display **passive** and the configuration is **synchronized**.




   - On the active device: The state of the local device should display **active** and the configuration is **synchronized**.


STEP 12 | Verify that failover occurs properly.

1. Shut down the active HA peer.
   1. On the EC2 Dashboard, select **Instances**.



© Palo Alto Networks, Inc.

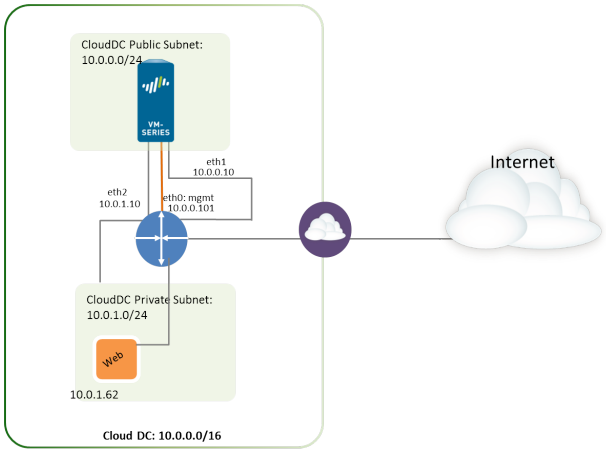
2. From the list, select the VM-Series firewall and click **Actions** > **Stop**.
2. Check that the passive peer assumes the role of the active peer and that the dataplane interfaces have moved over to the now active HA peer.



© Palo Alto Networks, Inc.

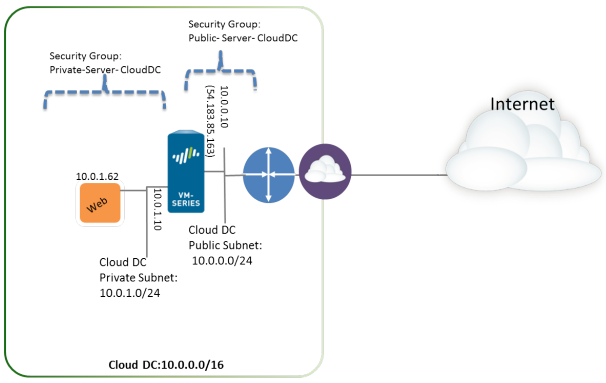
# Use Case: Secure the EC2 Instances in the AWS Cloud

In this example, the VPC is deployed in the 10.0.0.0/16 network with two /24 subnets: 10.0.0.0/24 and 10.0.1.0/24. The VM-Series firewall will be launched in the 10.0.0.0/24 subnet to which the internet gateway is attached. The 10.0.1.0/24 subnet is a private subnet that will host the EC2 instances that need to be secured by the VM-Series firewall; any server on this private subnet uses NAT for a routable IP address (which is an Elastic IP address) to access the internet. Use the Planning Worksheet for the VM-Series in the AWS VPC to plan the design within your VPC; recording the subnet ranges, network interfaces and the associated IP addresses for the EC2 instances, and security groups, will make the setup process easier and more efficient.




The following image depicts the logical flow of traffic to/from the web server to the internet. Traffic to/from the web server is sent to the data interface of the VM-Series firewall that is attached to the private subnet. The firewall applies policy and processes incoming/outgoing traffic from/to the internet gateway of the VPC. The image also shows the security groups to which the data interfaces are attached.

Logical Traffic Flow Diagram




The management IP address and the security group associated with it, are not depicted in this diagram

STEP 1 | Create a new VPC with a public subnet (or select an existing VPC).

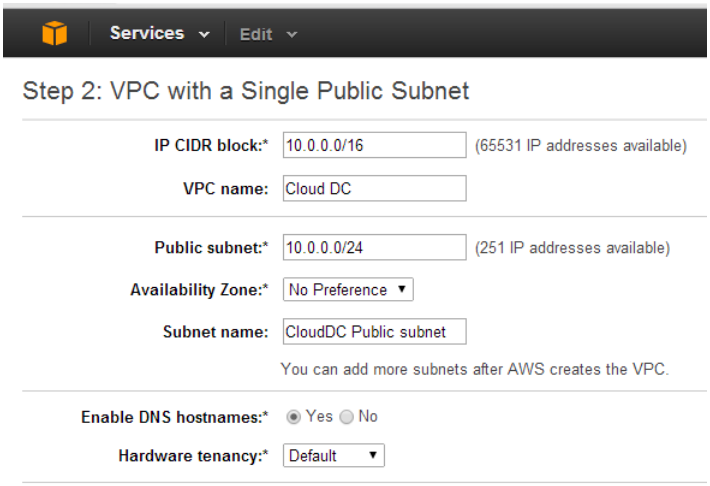
1. Log in to the AWS console and select the **VPC** Dashboard.
2. Verify that you've selected the correct geographic area (AWS region). The VPC will be deployed in the currently selected region.



© Palo Alto Networks, Inc.

3. Select **Start VPC Wizard**, and select **VPC with a Single Public Subnet**.

   In this example, the IP CIDR block for the VPC is 10.0.0.0/16, the VPC name is Cloud DC, the public subnet is 10.0.0.0/24, and the subnet name is Cloud DC Public subnet. You will create a private subnet after creating the VPC.

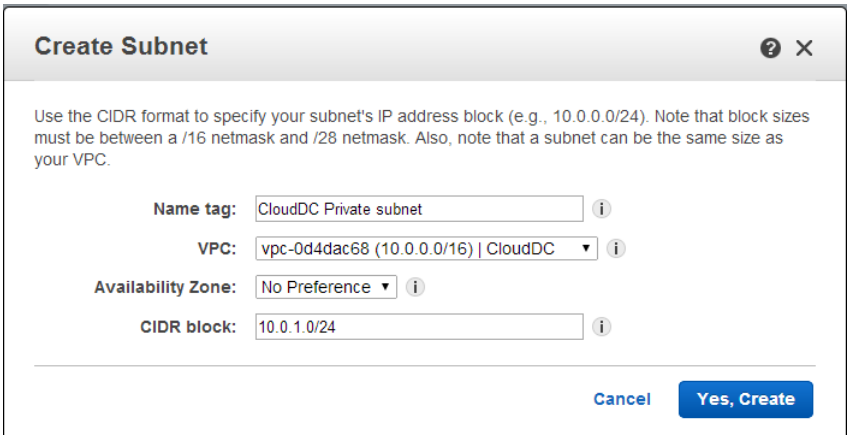



4. Click **Create VPC**.

STEP 2 | Create a private subnet.

Select **Subnets**, and click **Create a Subnet**. Fill in the information.

In this example, the **Name tag** for the subnet is Web/DB Server Subnet, it is created in the Cloud Datacenter VPC and is assigned a CIDR block of 10.0.1.0/24.




STEP 3 | Create a new route table for each subnet.

*Although a main route table is automatically created on the VPC, we recommend creating new route tables instead of modifying the default route table.*

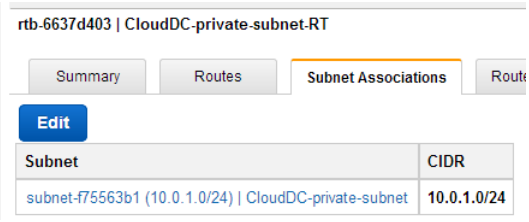
To direct outbound traffic from each subnet, you will add routes to the route table associated with each subnet, later in this workflow.

1. Select **Route Tables** > **Create Route Table**.
2. Add a **Name**, for example CloudDC-public-subnet-RT, select the **VPC** you created in step 1, and click **Yes, Create**.
3. Select the route table, click **Subnet Associations** and select the public subnet.



© Palo Alto Networks, Inc.


4. Select **Create Route Table**.
5. Add a **Name**, for example CloudDC-private-subnet-RT, select the **VPC** you created in step 1, and click **Yes, Create**.
6. Select the route table, click **Subnet Associations** and select the private subnet.




STEP 4 | Create Security Groups to restrict inbound/outbound internet access to the EC2 instances in the VPC.

By default, AWS disallows communication between interfaces that do not belong to the same security group.

Select **Security Groups** and click the **Create Security Group** button. In this example, we create three security groups with the following rules for inbound access:

- CloudDC-Management that specifies the protocols and source IP addresses that can connect to the management interface of the VM-Series firewall. At a minimum you need SSH, and HTTPS. In this example, we enable SSH, ICMP, HTTP, and HTTPS on the network interfaces that are attached to this security group.

  The management interface (eth 0/0) of the VM-Series firewall will be assigned to CloudDC-management-sg.
- Public-Server-CloudDC that specifies the source IP addresses that can connect over HTTP, FTP, SSH within the VPC. This group allows traffic from the external network to the firewall.

  The dataplane interface eth1/1 of the VM-Series firewall will be assigned to Public-Server-CloudDC.
- Private-Server-CloudDC that has very limited access. It only allows other EC2 instances on the same subnet to communicate with each other, and with the VM-Series firewall.

  The dataplane interface eth1/2 of the VM-Series firewall and the application in the private subnet will be attached to this security group.

  The following screenshot shows the security groups for this use case.

| Name tag | Group ID | Group Name | VPC | Description |
|---|---|---|---|---|
| CloudDC-private-subnet-sg | sg-6c32c409 | Private-Server-CloudDC | vpc-0d4dac68 (10.0.0.0/16) \|... | For Private Servers to comm... |
| CloudDC-public-subnet-sg | sg-6832c40d | Public-Server-CloudDC | vpc-0d4dac68 (10.0.0.0/16) \|... | External Traffic to VM-Series |
| CloudDC-management-sg | sg-9735c3f2 | CloudDC-Managment | vpc-0d4dac68 (10.0.0.0/16) \|... | CloudDC-Management |
| | sg-1035c375 | default | vpc-0d4dac68 (10.0.0.0/16) \|... | default VPC security group |

STEP 5 | Deploy the VM-Series firewall.



© Palo Alto Networks, Inc.

*Only the primary network interface that will serve as the management interface will be attached and configured for the firewall during the initial launch. The network interfaces required for handling data traffic will be added in step 6.*

See step 3 in Launch the VM-Series Firewall on AWS.

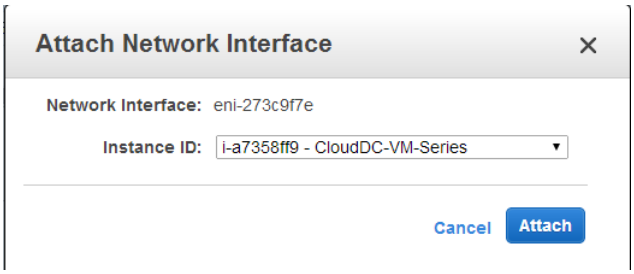
STEP 6 | Create and attach virtual network interface(s), referred to as Elastic Network Interfaces (ENIs), to the VM-Series firewall. These ENIs are used for handling data traffic to/from the firewall.

1. On the EC2 Dashboard, select **Network Interfaces**, and click **Create Network Interface**.
2. Enter a descriptive name for the interface.
3. Select the subnet. Use the subnet ID to make sure that you have selected the correct subnet. You can only attach an ENI to an instance in the same subnet.
4. Enter the **Private IP** address that you want to assign to the interface or select **Auto-assign** to automatically assign an IP address within the available IP addresses in the selected subnet.
5. Select the **Security group** to control access to the network interface.
6. Click **Yes, Create**.

   In this example, we create two interfaces with the following configuration:

| Name | Network interfa | Subnet ID | VPC ID | Zone | Security group | Description | Instance ID |
|---|---|---|---|---|---|---|---|
| CloudDC-VM-Series-Untrust | eni-bcf355e5 | subnet-ef5563a9 | vpc-0d4dac68 | us-west-1a | Public-Server-... | CloudDC-VM-Series-untrust | i-a7358ff9 |
| CloudDC-VM-Series-Trust | eni-abf355f2 | subnet-f75563b1 | vpc-0d4dac68 | us-west-1a | Private-Server-... | CloudDC-VM-Series-Trust | i-a7358ff9 |

   - For Eth1/1 (VM-Series-Untrust)
     - Subnet: 10.0.0.0/24
     - Private IP:10.0.0.10
     - Security group: Public-Server-CloudDC
   - For Eth1/2 (VM-Series-Trust)
     - Subnet: 10.0.1.0/24
     - Private IP:10.0.1.10
     - Security group: Private-Server-CloudDC
7. To attach the ENI to the VM-Series firewall, select the interface you just created, and click **Attach**.




8. Select the **Instance ID** of the VM-Series firewall, and click **Attach**.
9. Repeat steps 7 and 8 to attach the other network interface.

STEP 7 | Create an Elastic IP address and attach it to the firewall dataplane network interface that requires direct internet access.

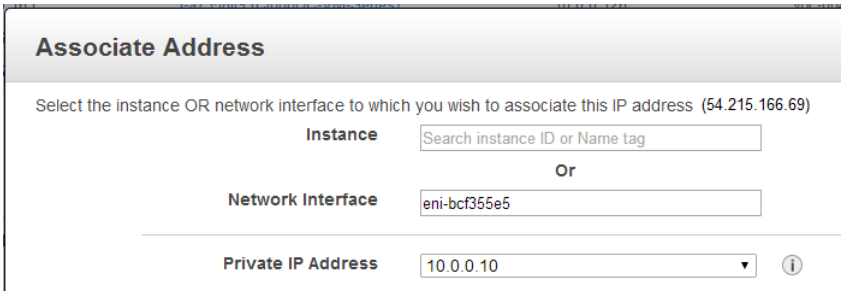
In this example, VM-Series_Untrust is assigned an EIP. The EIP associated with the interface is the publicly accessible IP address for the web server in the private subnet.

1. Select **Elastic IPs** and click **Allocate New Address**.
2. Select **EC2-VPC** and click **Yes, Allocate**.



© Palo Alto Networks, Inc.

3. Select the newly allocated EIP and click **Associate Address**.
4. Select the **Network Interface** and the **Private IP address** associated with the interface and click **Yes, Associate**.




In this example, the configuration is:

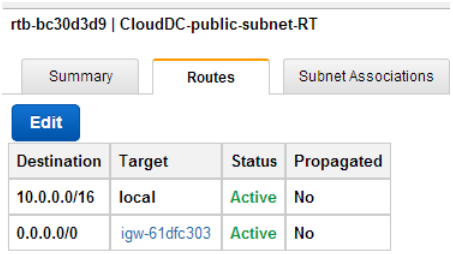
| Address | Instance | Private IP Address | Scope | Public DNS |
|---|---|---|---|---|
| 54.183.85.163 | i-a7358ff9 (CloudDC-VM-Series) | 10.0.0.126 | vpc-0d4dac68 | ec2-54-183-85-163.us-west-... |
| 54.215.166.69 | i-a7358ff9 (CloudDC-VM-Series) | 10.0.0.10 | vpc-0d4dac68 | ec2-54-215-166-69.us-west-... |

STEP 8 | Disable Source/Destination check on each network interface attached to the VM-Series firewall. Disabling this attribute allows the interface to handle network traffic that is not destined to its IP address.

1. Select the network interface in the **Network Interfaces** tab.
2. In the **Action** drop-down, select **Change Source/Dest. Check**.
3. Click **Disabled** and **Save** your changes.
4. Repeat steps 1-3 for additional network interfaces, firewall-1/2 in this example.

STEP 9 | In the route table associated with the public subnet (from step 3), add a default route to the internet gateway for the VPC.

1. From the VPC Dashboard, select **Route Tables** and find the route table associated with the public subnet.
2. Select the route table, select **Routes** and click **Edit**.
3. Add a route to forward packets from this subnet to the internet gateway. In this example, 0.0.0.0 indicates that all traffic from/to this subnet will use the internet gateway attached to the VPC.




STEP 10 | In the route table associated with the private subnet, add a default route to send traffic to the VM-Series firewall.

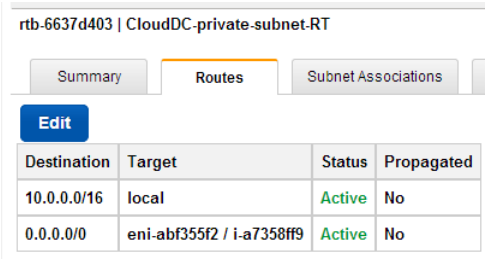
Adding this route enables the forwarding of traffic from the EC2 instances in this private subnet to the VM-Series firewall.

1. From the VPC Dashboard, select **Route Tables** and find the route table associated with the private subnet.
2. Select the route table, select **Routes** and click **Edit**.



© Palo Alto Networks, Inc.

3. Add a route to forward packets from this subnet to the VM-Series firewall network interface that resides on the same subnet. In this example, 0.0.0.0/0 indicates that all traffic from/to this subnet will use eni-abf355f2 (ethernet 1/2, which is CloudDC-VM-Series-Trust) on the VM-Series firewall.




*For each web or database server deployed on an EC2 instance in the private subnet, you must define a default route to the IP address of the VM-Series firewall so that the firewall is the default gateway for the server.*

Perform steps 11 through 16 on the VM-Series firewall

STEP 11 | Configure a new administrative password for the firewall.

*An SSH tool such as PuTTY is required to access the CLI on the firewall and change the default administrative password. You cannot access the web interface until you SSH and change the default password.*

1. Use the public IP address you configured on the firewall, to SSH into the Command Line Interface (CLI) of the VM-Series firewall.

   You will need the private key that you used or created in Launch the VM-Series Firewall on AWS, steps 3-xii to access the CLI.
2. Enter the following command to log in to the firewall:

   **ssh-i *<private_key_name>* admin@*<public-ip_address>***
3. Configure a new password, using the following command and follow the onscreen prompts:

   ```
   configure
   set mgt-config users admin password
   commit
   ```
4. Terminate the SSH session.

STEP 12 | Access the web interface of the VM-Series firewall.

Open a web browser and enter the EIP of the management interface. For example: https://54.183.85.163

STEP 13 | Activate the licenses on the VM-Series firewall. This step is only required for the BYOL license; the usage-based licenses are automatically activated.

See Activate the License.

STEP 14 | On the VM-Series firewall, configure the dataplane network interfaces on the firewall as Layer 3 interfaces.

1. Select **Network** > **Interfaces** > **Ethernet**.
2. Click the link for **ethernet 1/1** and configure as follows:
   - **Interface Type**: **Layer3**
   - Select the **Config** tab, assign the interface to the default router.



© Palo Alto Networks, Inc.

- On the **Config** tab, expand the **Security Zone** drop-down and select New Zone. Define a new zone, for example untrust, and then click OK.
- Select IPv4, select **DHCP Client**; the private IP address that you assigned to the network interface in the AWS management console will be acquired automatically.
- On the **Advanced** > **Other Info** tab, expand the Management Profile drop-down, and select **New Management Profile**.
- Enter a **Name** for the profile, such as allow_ping, and select **Ping** from the Permitted Services list, then click **OK**.
- To save the interface configuration, click **OK**.

3. Click the link for **ethernet 1/2** and configure as follows:
   - **Interface Type**: **Layer3**
   - Select the **Config** tab, assign the interface to the default router.
   - On the **Config** tab, expand the **Security Zone** drop-down and select **New Zone**. Define a new zone, for example trust, and then click **OK**.
   - Select **IPv4**, select **DHCP Client**.
   - On the **IPv4** tab, clear the **Automatically create default route to default gateway provided by server** check box. For an interface that is attached to the private subnet in the VPC, disabling this option ensures that traffic handled by this interface does not flow directly to the IGW on the VPC.
   - On the **Advanced** > **Other Info**, expand the Management Profile drop-down, and select the allow_ping profile you created earlier.
   - Click **OK** to save the interface configuration.
4. Click **Commit** to save the changes. Verify that the Link state for the interface is up 

. If the link state is not up, reboot the firewall.

| Interface | Interface Type | Management Profile | Link State | IP Address | Virtual Router | Security Zone |
|---|---|---|---|---|---|---|
| ethernet1/1 | Layer3 | allow_ping | | Dynamic-DHCP Client | default | untrust |
| ethernet1/2 | Layer3 | allow_ping | | Dynamic-DHCP Client | default | trust |

STEP 15 | On the VM-Series firewall, create Destination NAT and Source NAT rules to allow inbound/outbound traffic to/from the applications deployed within the VPC.

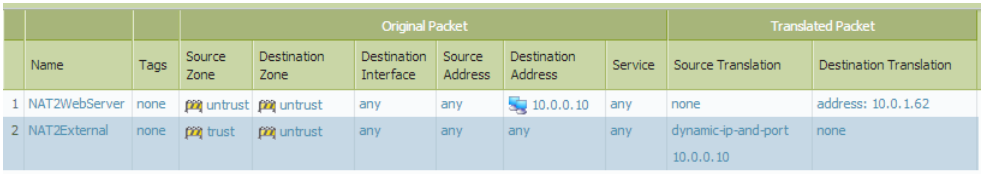
1. Select **Policies** > **NAT**.
2. Create a Destination NAT rule that steers traffic from the firewall to the web server.
   1. Click **Add**, and enter a name for the rule. For example, NAT2WebServer.
   2. In the **Original Packet** tab, make the following selections:
      - **Source Zone**: untrust (where the traffic originates)
      - **Destination Zone**: untrust (the zone for the firewall dataplane interface with which the EIP for the web server is associated.)
      - **Source Address**: Any
      - **Destination Address**: 10.0.0.10
      - In the **Translated Packet** tab, select the Destination Address Translation check box and set the **Translated Address**: to 10.0.1.62, which is the private IP address of the web server.
   3. Click **OK**.

| | | | Original Packet | | | | | | Translated Packet | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Name | Tags | Source Zone | Destination Zone | Destination Interface | Source Address | Destination Address | Service | Source Translation | Destination Translation |
| 1 | NAT2WebServer | none | untrust | untrust | any | any | 10.0.0.10 | any | none | address: 10.0.1.62 |

3. Create a Source NAT rule to allow outbound traffic from the web server to the internet.
   1. Click **Add**, and enter a name for the rule. For example, NAT2External.


© Palo Alto Networks, Inc.

2. In the **Original Packet** tab, make the following selections:
   - **Source Zone**: trust (where the traffic originates)
   - **Destination Zone**: untrust (the zone for the firewall dataplane interface with which the EIP for the web server is associated.)
   - **Source Address**: Any
   - **Destination Address**: Any
3. In the **Translated Packet** tab, make the following selections in the Source Address Translation section:
   - **Translation Type**: Dynamic IP and Port
   - **Address Type**: Translated Address
   - **Translated Address**: 10.0.0.10 (the firewall dataplane interface in the untrust zone.)
4. Click **OK**.



| | | | Original Packet | | | | | | Translated Packet | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Name | Tags | Source Zone | Destination Zone | Destination Interface | Source Address | Destination Address | Service | Source Translation | Destination Translation |
| 1 | NAT2WebServer | none | untrust | untrust | any | any | 10.0.0.10 | any | none | address: 10.0.1.62 |
| 2 | NAT2External | none | trust | untrust | any | any | any | any | dynamic-ip-and-port 10.0.0.10 | none |

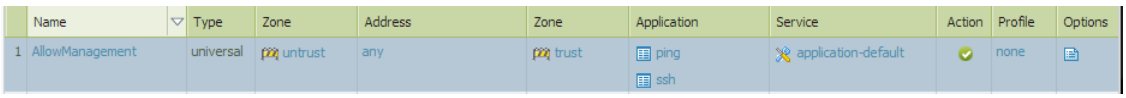
4. Click **Commit** to save the NAT policies.

STEP 16 | On the VM-Series firewall, create security policies to manage traffic.

*Instead of entering a static IP address for the web server, use a dynamic address group. Dynamic address groups allow you to create policy that automatically adapts to changes so that you do not need to update the policy when you launch additional web servers in the subnet. For details, see* Use Case: Use Dynamic Address Groups to Secure New EC2 Instances within the VPC.

1. Select **Policies** > **Security**.

   In this example, we have four rules. A rule that allows management access to the firewall traffic, a rule to allow inbound traffic to the web server, a third rule to allow internet access to the web server, and in the last rule we modify a predefined intrazone-default rule to log all traffic that is denied.
2. Create a rule to allow management access to the firewall.
   1. Click **Add** and enter a **Name** for the rule. Verify that the **Rule Type** is universal.
   2. In the **Source** tab, add untrust as the **Source Zone**.
   3. In the **Destination** tab, add trust as the **Destination Zone**.
   4. In the **Applications** tab, **Add** ping and ssh.
   5. In the **Actions** tab, set the **Action** to Allow.
   6. Click **OK**.



| | Name | Type | Zone | Address | Zone | Application | Service | Action | Profile | Options |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | AllowManagement | universal | untrust | any | trust | ping<br>ssh | application-default | | none | |

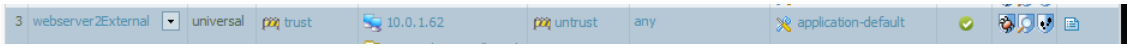
3. Create a rule to allow inbound traffic to the web server.
   1. Click **Add** and enter a **Name** for the rule and verify that the **Rule Type** is universal.
   2. In the **Source** tab, add untrust as the **Source Zone**.
   3. In the **Destination** tab, add trust as the **Destination Zone**.
   4. In the **Applications** tab, **Add** web-browsing.
   5. In the **Service/URL Category** tab, verify that the service is set to application-default.
   6. In the **Actions** tab, set the **Action** to Allow.



© Palo Alto Networks, Inc.

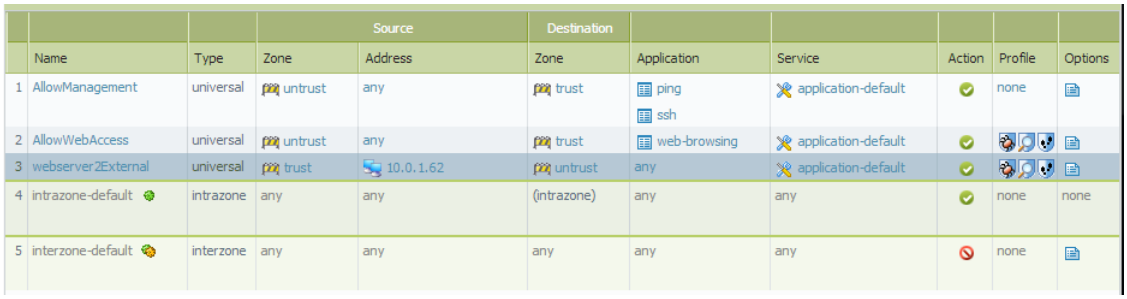
7. In the Profile Settings section of the **Actions** tab, select **Profiles** and then attach the default profiles for antivirus, anti-spyware, and vulnerability protection.
8. Click **OK**.


4. Create a rule to allow internet access to the web server.
    1. Click **Add** and enter a **Name** for the rule and verify that the Rule Type is universal.
    2. In the **Source** tab, add trust as the **Source Zone**.
    3. In the Source Address section of the **Source** tab, add 10.0.1.62, the IP address of the web server.
    4. In the **Destination** tab, add untrust as the **Destination Zone**.
    5. In the **Service/URL Category** tab, verify that the service is set to **application-default**.
    6. In the **Actions** tab, set the **Action** to Allow.
    7. In the Profile Settings section of the **Actions** tab, select **Profiles** and then attach the default profiles for antivirus, anti-spyware, and vulnerability protection.
    8. Click **OK**.




5. Edit the interzone-default rule to log all traffic that is denied. This predefined interzone rule is evaluated when no other rule is explicitly defined to match traffic across different zones.
    1. Select the **interzone-default** rule and click **Override**.
    2. In the **Actions** tab, select **Log at session end**.
    3. Click **OK**.

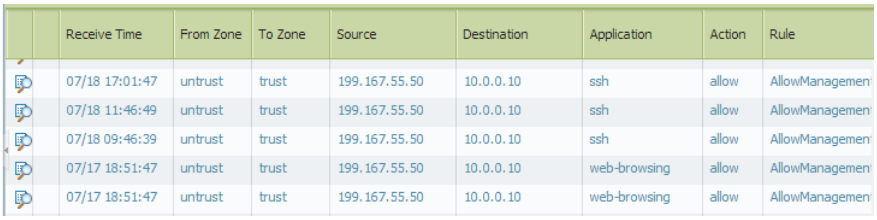

6. Review the complete set of security rules defined on the firewall.
7. Click **Commit** to save the policies.



| | | | Source | | Destination | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Name | Type | Zone | Address | Zone | Application | Service | Action | Profile | Options |
| 1 | AllowManagement | universal | untrust | any | trust | ping<br>ssh | application-default | | none | |
| 2 | AllowWebAccess | universal | untrust | any | trust | web-browsing | application-default | | | |
| 3 | webserver2External | universal | trust | 10.0.1.62 | untrust | any | application-default | | | |
| 4 | intrazone-default | intrazone | any | any | (intrazone) | any | any | | none | none |
| 5 | interzone-default | interzone | any | any | any | any | any | | none | |

## STEP 17 | Verify that the VM-Series firewall is securing traffic.

1. Launch a web browser and enter the IP address for the web server.
2. Log in to the web interface of the VM-Series firewall and verify that you can see the traffic logs for the sessions at **Monitor** > **Logs** > **Traffic**.
    - Traffic inbound to the web server (arrives at EC2 instance in the AWS VPC):

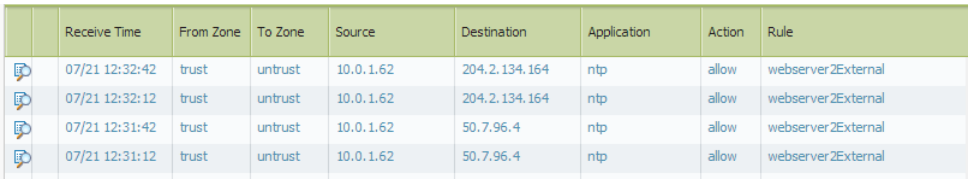


| | Receive Time | From Zone | To Zone | Source | Destination | Application | Action | Rule |
|---|---|---|---|---|---|---|---|---|
| | 07/18 17:01:47 | untrust | trust | 199.167.55.50 | 10.0.0.10 | ssh | allow | AllowManagemen |
| | 07/18 11:46:49 | untrust | trust | 199.167.55.50 | 10.0.0.10 | ssh | allow | AllowManagemen |
| | 07/18 09:46:39 | untrust | trust | 199.167.55.50 | 10.0.0.10 | ssh | allow | AllowManagemen |
| | 07/17 18:51:47 | untrust | trust | 199.167.55.50 | 10.0.0.10 | web-browsing | allow | AllowManagemen |
| | 07/17 18:51:47 | untrust | trust | 199.167.55.50 | 10.0.0.10 | web-browsing | allow | AllowManagemen |

    - Traffic outbound from the web server (EC2 instance in the AWS VPC):



© Palo Alto Networks, Inc.



| | | Receive Time | From Zone | To Zone | Source | Destination | Application | Action | Rule |
|---|---|---|---|---|---|---|---|---|---|
| | | 07/21 12:32:42 | trust | untrust | 10.0.1.62 | 204.2.134.164 | ntp | allow | webserver2External |
| | | 07/21 12:32:12 | trust | untrust | 10.0.1.62 | 204.2.134.164 | ntp | allow | webserver2External |
| | | 07/21 12:31:42 | trust | untrust | 10.0.1.62 | 50.7.96.4 | ntp | allow | webserver2External |
| | | 07/21 12:31:12 | trust | untrust | 10.0.1.62 | 50.7.96.4 | ntp | allow | webserver2External |

You have successfully deployed the VM-Series firewall as a cloud gateway!



© Palo Alto Networks, Inc.

# Use Case: Use Dynamic Address Groups to Secure New EC2 Instances within the VPC

In a dynamic environment such as the AWS-VPC where you launch new EC2 instances on demand, the administrative overhead in managing security policy can be cumbersome. Using Dynamic Address Groups in security policy allows for agility and prevents disruption in services or gaps in protection.

In this example, you can use the VM Information Source on the firewall to monitor a VPC and use Dynamic Address Groups in security policy to discover and secure EC2 instances. As you spin up EC2 instances, the Dynamic Address Group collates the IP addresses of all instances that match the criteria defined for group membership, and then security policy is applied for the group. The security policy in this example allows internet access to all members of the group.

*Instead of using VM Information Source on the firewall, you can opt to use Panorama as the central point for communicating with your VPCs. Using the AWS plugin on Panorama, you can retrieve the IP address-to-tag mapping and register the information on the managed firewalls for which you configure notification. For more details on this option, see* VM Monitoring with the AWS Plugin on Panorama.

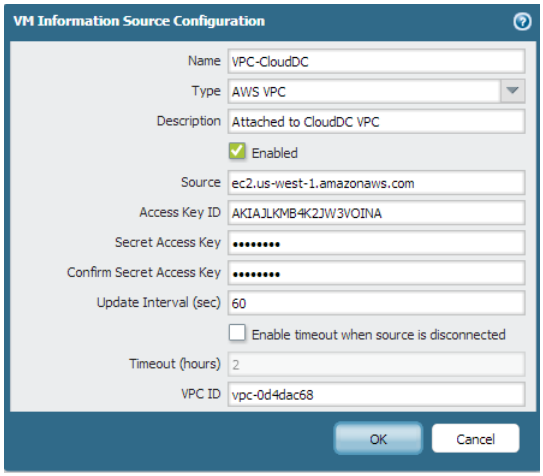
This workflow in the following section assumes that you have created the AWS VPC and deployed the VM-Series firewall and some applications on EC2 instances. For instructions on setting up the VPC for the VM-Series, see Use Case: Secure the EC2 Instances in the AWS Cloud.

STEP 1 | Configure the firewall to monitor the VPC.

1. Select **Device** > **VM Information Sources**.
2. Click **Add** and enter the following information:
    1. A **Name** to identify the VPC that you want to monitor. For example, VPC-CloudDC.
    2. Set the **Type** to AWS VPC.
    3. In **Source**, enter the URI for the VPC. The syntax is **`ec2.<your_region>.amazonaws.com`**
    4. Add the credentials required for the firewall to digitally sign API calls made to the AWS services. You need the following:
        - **Access Key ID**: Enter the alphanumeric text string that uniquely identifies the user who owns or is authorized to access the AWS account.
        - **Secret Access Key**: Enter the password and confirm your entry.
    5. (**Optional**) Modify the **Update interval** to a value between 5-600 seconds. By default, the firewall polls every 5 seconds. The API calls are queued and retrieved within every 60 seconds, so updates may take up to 60 seconds plus the configured polling interval.



© Palo Alto Networks, Inc.



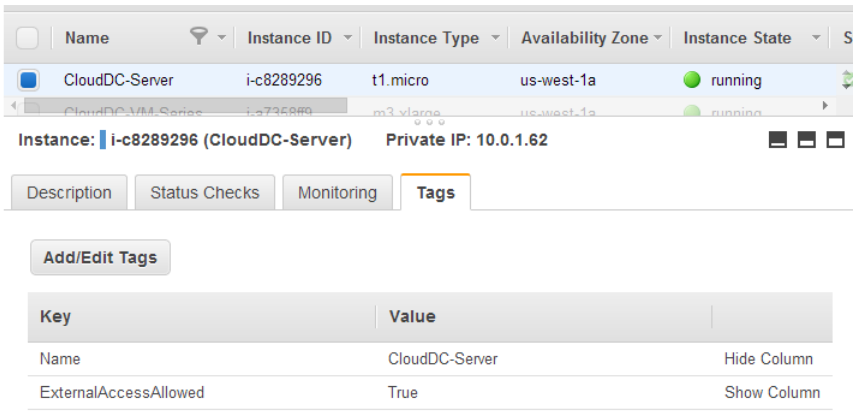

6. Enter the **VPC ID** that is displayed on the VPC Dashboard in the AWS management console.
7. Click **OK**, and **Commit** the changes.
8. Verify that the connection **Status** displays as connected

STEP 2 | Tag the EC2 instances in the VPC.

For a list of tags that the VM-Series firewall can monitor, see List of Attributes Monitored on the AWS VPC.

A tag is a name-value pair. You can tag the EC2 instances either on the EC2 Dashboard on the AWS management console or using the AWS API or AWS CLI.

In this example, we use the EC2 Dashboard to add the tag:




STEP 3 | Create a dynamic address group on the firewall.

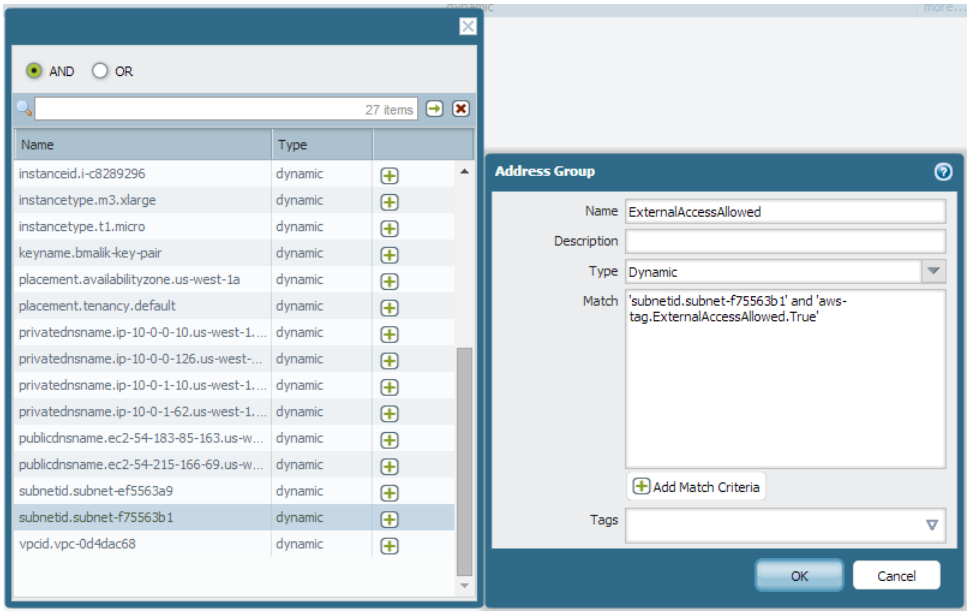
*View the tutorial to see a big picture view of the feature.*

1. Select **Object** > **Address Groups**.
2. Click **Add** and enter a **Name** and a **Description** for the address group.
3. Select **Type** as **Dynamic**.
4. Define the match criteria.
   1. Click **Add Match Criteria**, and select the **And** operator.



© Palo Alto Networks, Inc.

2. Select the attributes to filter for or match against. In this example, we select the ExternalAccessAllowed tag that you just created and the subnet ID for the private subnet of the VPC.




5. Click **OK**.
6. Click **Commit**.

## STEP 4 | Use the dynamic address group in a security policy.

To create a rule to allow internet access to any web server that belongs to the dynamic address group called ExternalServerAccess.

1. Select **Policies** > **Security**.
2. Click **Add** and enter a **Name** for the rule and verify that the **Rule Type** is universal.
3. In the **Source** tab, add trust as the **Source Zone**.
4. In the Source Address section of the **Source** tab, **Add** the ExternalServerAccess group you just created.
5. In the **Destination** tab, add untrust as the **Destination Zone**.
6. In the **Service/URL Category** tab, verify that the service is set to **application-default**.
7. In the **Actions** tab, set the **Action** to Allow.
8. In the Profile Settings section of the **Actions** tab, select **Profiles** and then attach the default profiles for antivirus, anti-spyware, and vulnerability protection.
9. Click **OK**.


10. Click **Commit**.

## STEP 5 | Verify that members of the dynamic address group are populated on the firewall.

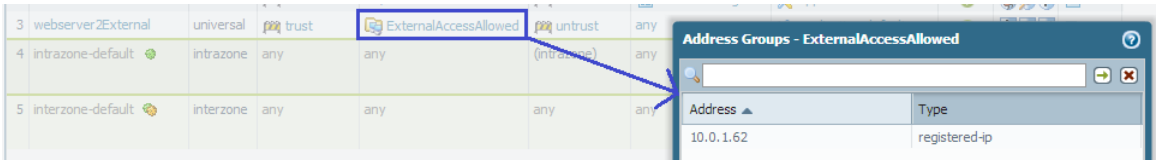
Policy will be enforced for all IP addresses that belong to this address group, and are displayed here.

1. Select **Policies** > **Security**, and select the rule.
2. Select the drop-down arrow next to the address group link, and select **Inspect**. You can also verify that the match criteria is accurate.



© Palo Alto Networks, Inc.

3. Click the **more** link and verify that the list of registered IP addresses is displayed.






© Palo Alto Networks, Inc.

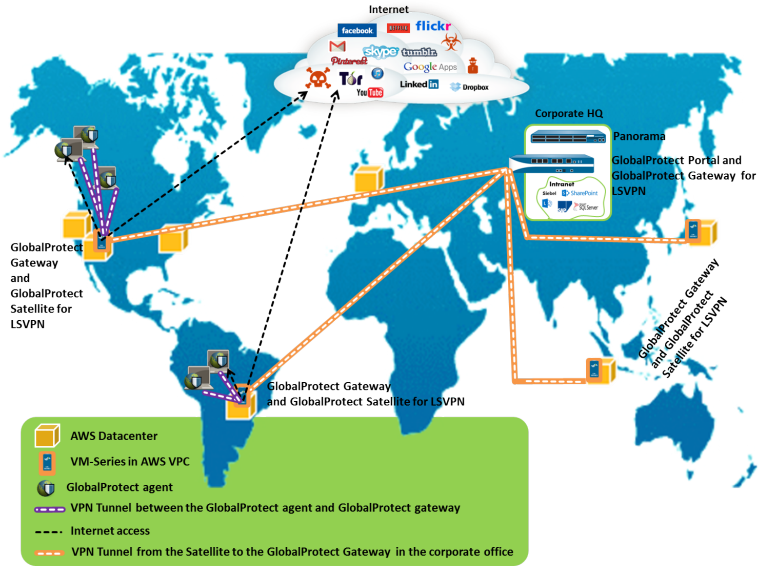
# Use Case: VM-Series Firewalls as GlobalProtect Gateways on AWS

Securing mobile users from threats and risky applications is often a complex mix of procuring and setting up the security and IT infrastructure, ensuring bandwidth and uptime requirements in multiple locations around the globe while staying within your budget.

The VM-Series firewall on AWS melds the security and IT logistics required to consistently and reliably protect devices used by mobile users in regions where you do not have a presence. By deploying the VM-Series firewall in the AWS cloud, you can quickly and easily deploy GlobalProtect™ gateways in any region without the expense or IT logistics that are typically required to set up this infrastructure using your own resources.

To minimize latency, select AWS regions that are closest to your users, deploy the VM-Series firewalls on EC2 instances, and configure the firewalls as GlobalProtect gateways. With this solution, the GlobalProtect gateways in the AWS cloud enforce security policy for internet traffic so there is no need to backhaul that traffic to the corporate network. Additionally, for access to resources on the corporate network, the VM-Series firewalls on AWS leverage the LSVPN functionality to establish IPSec tunnels back to the firewall on the corporate network.

For ease of deployment and centralized management of this distributed infrastructure, use Panorama to configure the GlobalProtect components used in this solution. Optionally, to ensure that mobile devices, such as smartphones and tablets, are safe for use on your network, use a Mobile Device Manager to configure and manage mobile devices.




- Components of the GlobalProtect Infrastructure
- Deploy GlobalProtect Gateways on AWS

## Components of the GlobalProtect Infrastructure

To block risky applications and protect mobile users from malware, you must set up the GlobalProtect infrastructure, which includes the GlobalProtect portal, the GlobalProtect gateway, and the GlobalProtect app. Additionally, for access to corporate resources, you must set up an IPSec VPN connection between the



© Palo Alto Networks, Inc.

VM-Series firewalls on AWS and the firewall in the corporate headquarters using LSVPN (a hub and spoke VPN deployment).

- The GlobalProtect agent/app is installed on each end-user system that is allowed to access corporate applications and resources. The agent first connects to the portal to obtain information on the gateways and then establishes a secure VPN connection to the closest GlobalProtect gateway. The VPN connection between the end-user system and the gateway ensures data privacy.
- The GlobalProtect portal provides the management functions for the GlobalProtect infrastructure. Every end-user system receives configuration information from the portal, including information about available gateways as well as any client certificates that may be required to connect to the GlobalProtect gateway(s). In this use case, the GlobalProtect portal is a hardware-based firewall that is deployed in the corporate headquarters.
- The GlobalProtect gateway delivers mobile threat prevention and policy enforcement based on applications, users, content, device, and device state. In this use case, the VM-Series firewalls on AWS function as the GlobalProtect gateways. The GlobalProtect gateway scans each user request for malware and other threats, and, if policy allows, sends the request to the internet or to the corporate network over the IPSec tunnel (to the LSVPN gateway).
- For LSVPN, you must configure the GlobalProtect portal, GlobalProtect gateway for LSVPN (hub), and the GlobalProtect Satellites (spokes).

  In this use case, the hardware-based firewall in the corporate office is deployed as the GlobalProtect portal and the LSVPN gateway. The VM-Series firewalls on AWS are configured to function as GlobalProtect satellites. The GlobalProtect satellites and gateway are configured to establish an IPSec tunnel that terminates on the gateway. When a mobile user requests an application or resource that resides on the corporate network, the VM-Series firewall routes the request over the IPSec tunnel.

## Deploy GlobalProtect Gateways on AWS

To secure mobile users, in addition to deploying and configuring the GlobalProtect gateways on AWS, you need to set up the other components required for this integrated solution. The following table includes the recommended workflow:

- Deploy the VM-Series firewall(s) on AWS.

  See Deploy the VM-Series Firewall on AWS.

- Configure the firewall at the corporate headquarters.

  In this use case, the firewall is configured as the GlobalProtect portal and the LSVPN gateway.

  - Configure the GlobalProtectportal.
  - Configure the GlobalProtectportal for LSVPN.
  - Configure the portal to authenticateLSVPN satellites.
  - Configure the GlobalProtectgateway for LSVPN.

- Set up a template on Panorama for configuring the VM-Series firewalls on AWS as GlobalProtect gateways and LSVPN satellites.

  To easily manage this distributed deployment, use Panorama to configure the firewalls on AWS.

  - Create template(s) on Panorama.

  Then use the following links to define the configuration in the templates.

  - Configure the firewall asa GlobalProtect gateway.
  - Prepare the satellite tojoin the LSVPN.



© Palo Alto Networks, Inc.

- Create device groups on Panorama to define the network access policies and internet access rules and apply them to the firewalls on AWS.

  See Create device groups.

- Apply the templates and the device groups to the VM-Series firewalls on AWS, and verify that the firewalls are configured properly.

- Deploy the GlobalProtect client software.

  Every end-user system requires the GlobalProtect agent or app to connect to the GlobalProtect gateway.

  See Deploy the GlobalProtectclient software.

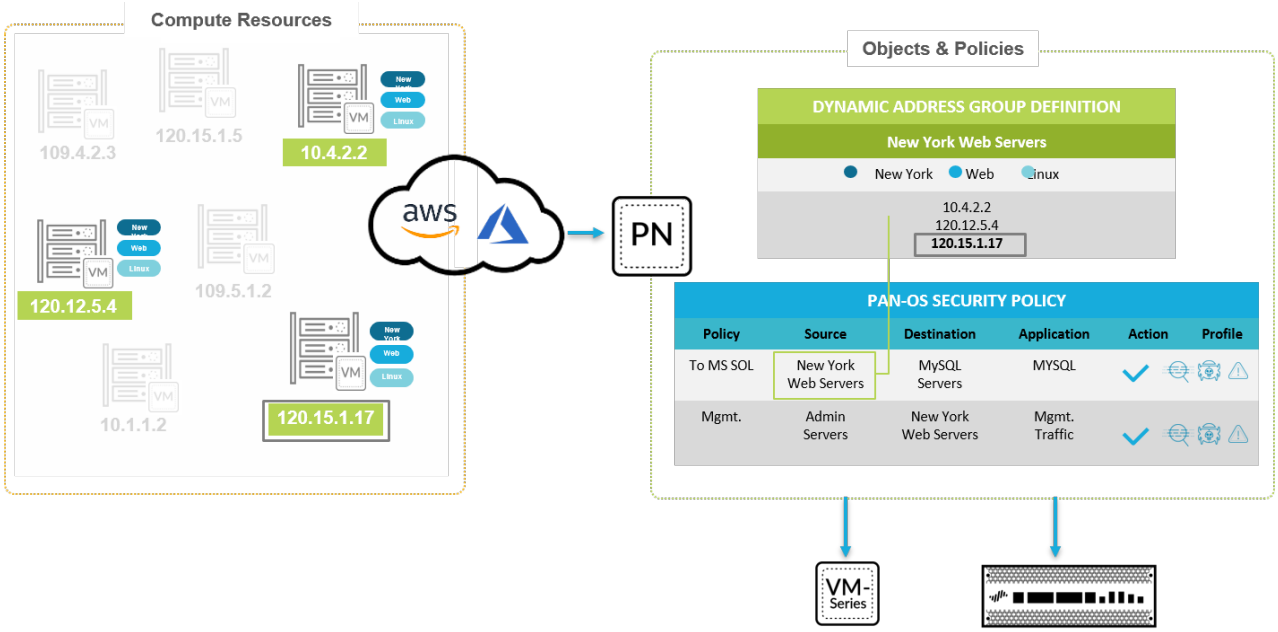


© Palo Alto Networks, Inc.

# VM Monitoring on AWS

As you deploy or terminate virtual machines in the AWS public cloud, you can either use the Panorama plugin for AWS or use the VM Information sources on the firewall to consistently enforce security policy rules on these workloads.

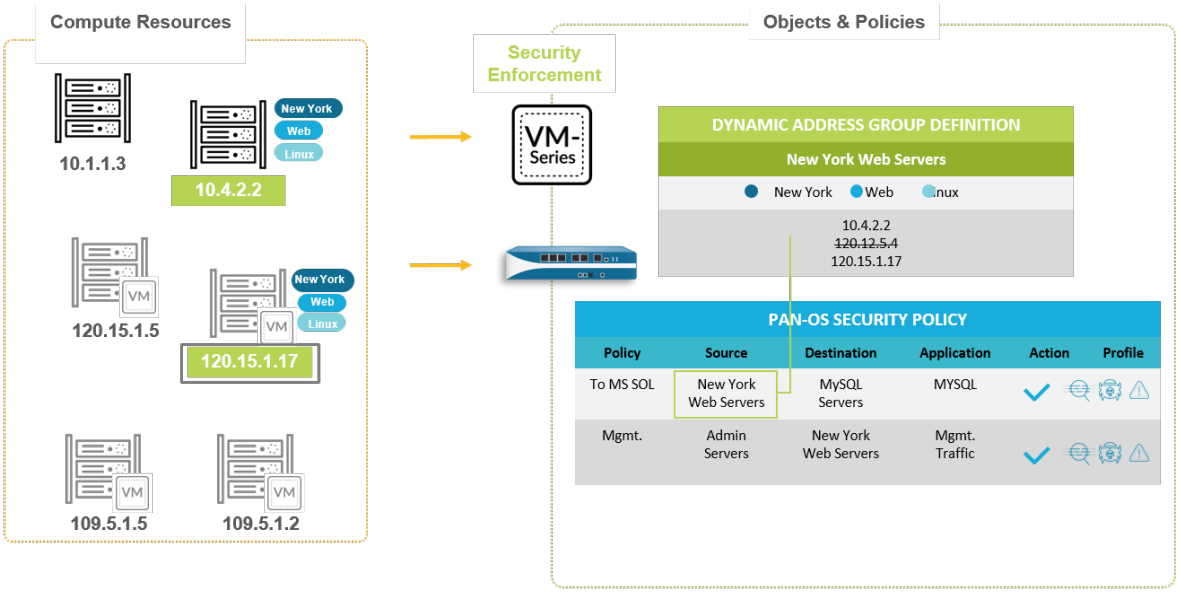
The Panorama plugin for AWS is built for scale and allows you to monitor up to 100 AWS VPCs on the AWS public cloud. With this plugin, you use Panorama as an anchor to poll your subscriptions for tags, and then distribute the metadata (IP address-to-tag mapping) to many firewalls in a device group. Because Panorama communicates with your AWS subscriptions to retrieve VM information, you're able to streamline the number of API calls made to the cloud environment. When using Panorama and the AWS plugin, you can centralize the retrieval of tags and Security policy management to ensure consistent policies for hybrid and cloud-native architectures. See VM Monitoring with the AWS Plugin on Panorama.




If you do not have Panorama or you have a simpler deployment and need to monitor 10 VPCs or fewer, you can use the VM Information Source on the firewall (hardware or VM-Series firewall) to monitor your AWS workloads. You can use the metadata, which the firewall retrieves, in Dynamic Address Groups and reference them in Security policies to secure your VM workloads as they spin up or down and IP addresses change frequently. See Use Case: Use Dynamic Address Groups to Secure New EC2 Instances within the VPC.



© Palo Alto Networks, Inc.




© [ © 2018, Palo Alto Networks, Inc. All Rights Reserved.

# VM Monitoring with the AWS Plugin on Panorama

As you deploy or terminate virtual machines in the AWS public cloud, you need a way to synchronously update Security policy on your Palo Alto Networks® firewall(s) so that you can secure these EC2 instances. To enable this capability from Panorama, you must install the AWS plugin on Panorama and enable API communication between Panorama and your AWS VPCs. Panorama can then collect a predefined set of attributes (or metadata elements) as tags for your EC2 instances and register the information to your Palo Alto Networks® firewall(s). When you reference these tags in Dynamic Address Groups and match against them in Security policy rules, you can consistently enforce policy across all assets deployed within your VPCs.

- Set Up the AWS Plugin for VM Monitoring on Panorama
- List of Attributes Monitored on the AWS VPC

# Set Up the AWS Plugin for VM Monitoring on Panorama

To find all the virtual machine workloads that your organization has deployed in the AWS public cloud, you need to install the AWS plugin on Panorama and configure *Monitoring Definitions* that enable Panorama to authenticate to your AWS VPC(s) and retrieve VM information on the workloads. Panorama retrieves the IP address of the VMs that are running— public IP address, and primary and secondary private IP addresses—and the associated tags. For a list of the metadata elements that Panorama supports, see List of Attributes Monitored on the AWS VPC.

After Panorama fetches the attributes, to push the virtual machine information from Panorama to the firewalls, you must add the firewalls (hardware or VM-Series) as managed devices on Panorama, and group the firewalls into one or more Device Groups. You can then specify which device groups are part of the *Notify Group*, which is a configuration element in a Monitoring Definition, that Panorama uses to register the IP address-to-tag mapping it retrieves from AWS.

Finally, to consistently enforce Security policies across the EC2 instances, you must set up Dynamic Address Groups and reference them in policy rules that allow or deny traffic to the IP addresses of the VMs. For streamlining your configuration and managing policies and objects centrally from Panorama, you can define



© Palo Alto Networks, Inc.

the Dynamic Address Groups and Security policy rules on Panorama and push them to the firewalls instead of managing the Dynamic Address Groups and Security policy rules locally on each firewall.

*The AWS plugin is for monitoring EC2 instances for upto 100 VPCs on the AWS public cloud. Version 1.0 does not support AWS GovCloud or AWS China.*

.

- Planning Checklist for VM Monitoring on AWS
- Install the AWS Plugin
- Configure the AWS Plugin for VM Monitoring

## Planning Checklist for VM Monitoring on AWS

For Panorama to interact with the AWS APIs and collect information on your EC2 instances, you need to create an IAM role and assign the policies that grant the permissions required to authenticate to AWS and access the EC2 instances within your VPC.

- Gather the VPC ID.
- Get the credentials that Panorama requires to digitally sign API calls to the AWS services. You need the Access Key ID that uniquely identifies the user who owns or is authorized to access the AWS account, and the Secret Access Key.

  The json format for the minimum permissions associated with the IAM role is as follows -

```
{ "Path": "/", "UserName": "panorama_vm_programmatic",
```

```
"UserId": "AIDAIZXXXXCR5JPII4XYZ",
```

```
 "Arn": "arn:aws:iam::412383210500:user/panorama_vm_programmatic",
```

```
 "CreateDate": "2018-07-06T19:14:31Z",
```

```
 "GroupList": [],
```

```
"AttachedManagedPolicies": [ { "PolicyName": "ReadOnlyAccess", "PolicyArn":
 "arn:aws:iam::aws:policy/ReadOnlyAccess" } ] },
```

- Tag your EC2 instances on AWS. You can tag (define a name-value pair) the EC2 instances either on the EC2 Dashboard on the AWS management console or using the AWS API or AWS CLI. See List of Attributes Monitored on the AWS VPC for the list of supported attributes.
- Review the requirements for Panorama and the managed firewalls:
  - Minimum system requirements—Panorama virtual appliance or hardware-based Panorama appliance. Panorama must have an active support license and a device management license for managing firewalls.

    Next-generation firewalls with a valid support license.



© Palo Alto Networks, Inc.

- You must add the firewalls as managed devices on Panorama and create Device Groups so that you can configure Panorama to notify these groups with the VM information it retrieves. Device groups can include VM-Series firewalls or virtual systems on the hardware firewalls.
- If your Panorama appliances are in a high availability configuration, you must manually install the same version of the AWS plugin on both Panorama peers.

  *You configure the AWS plugin on the active Panorama peer only. On commit, the configuration is synced to the passive Panorama peer. Only the active Panorama peer polls the AWS subscriptions you have configured for VM Monitoring.*

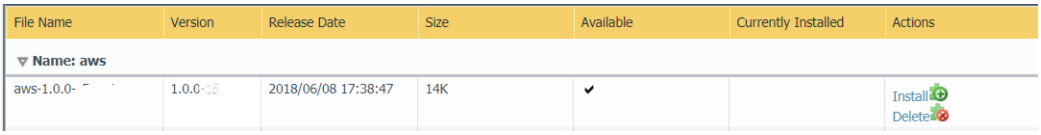
- Before you enable the Panorama plugin for monitoring for AWS VPCs, if you are using VM Information Source for AWS on the firewalls, you must disable it to avoid conflicts and unexpected behavior with tags.
- If you are using the Panorama plugin for Azure and AWS, you cannot target the same firewall or virtual system with tags from both environments. Ensure that there is no overlap of the Device Groups that you add to the Monitoring definitions for AWS and Azure.
- Check for duplicate IP addresses across the VPCs for which you will enable monitoring. If you have duplicate IP addresses across AWS VPCs, the metadata will be appended together or swapped and this may cause unexpected results in policy enforcement.

## Install the AWS Plugin

To get started with monitoring your EC2 instances on AWS, you need to download and install the AWS plugin on Panorama. If you have a Panorama HA configuration, repeat this installation process on each Panorama peer.

*If you currently have installed a Panorama plugin, the process of installing (or uninstalling) another plugin requires a Panorama reboot to enable you to commit changes. So, install additional plugins during a planned maintenance window to allow for a reboot.*

STEP 1 | Log in to the Panorama Web Interface, select **Panorama** > **Plugins** and click **Check Now** to get the **AWS** plugin .

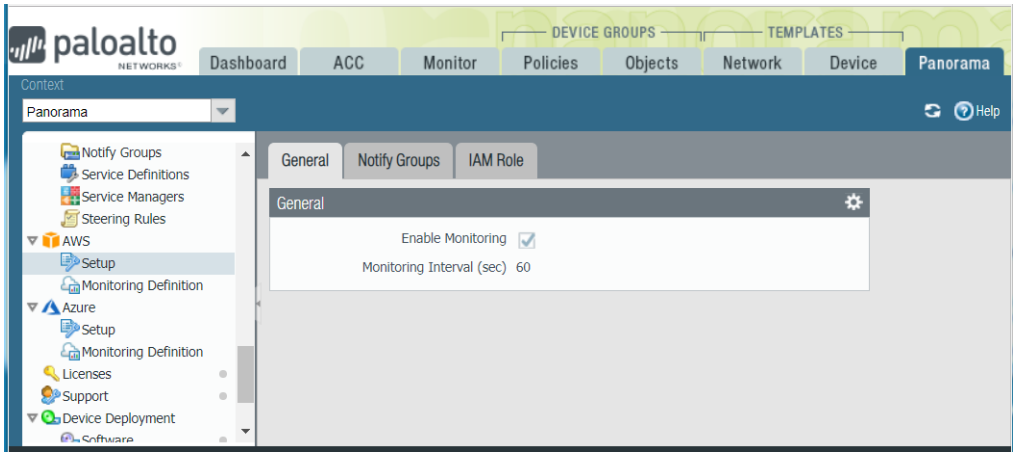



STEP 2 | **Install** the plugin.

After you successfully install, Panorama refreshes and the AWS plugin displays on the **Panorama** > **Plugins** tab.



© Palo Alto Networks, Inc.




STEP 3 | Restart Panorama.

Select **Panorama** > **Setup** > **Operations** > **Reboot Panorama**

## *Configure the AWS Plugin for VM Monitoring*

To begin monitoring the virtual machines in your AWS public cloud deployment, after you Install the AWS Plugin you must create a Monitoring Definition. This definition specifies the IAM Role that is authorized to access the EC2 instances within the AWS VPC you want to monitor and the Notify Group that includes the firewalls to which Panorama should push all the IP-address-to-tag mappings it retrieves. In order to enforce policy, you must then create Dynamic Address Groups and reference them in Security policy. The Dynamic Address Groups enable you to filter the tags you want to match on, so that the firewall can get the public and private IP addresses registered against each tag, and then allow or deny access to traffic to and from the workloads based on the policy rules you define.

*The number of tags that the Panorama plugin can retrieve and register is 7000 IP addresses each with 10 tags or 6500 IP addresses each with 15 tags per the firewalls or virtual systems included within a device group.*

STEP 1 | Log in to the Panorama web interface.

STEP 2 | Set up the following objects for enabling VM Monitoring on AWS.

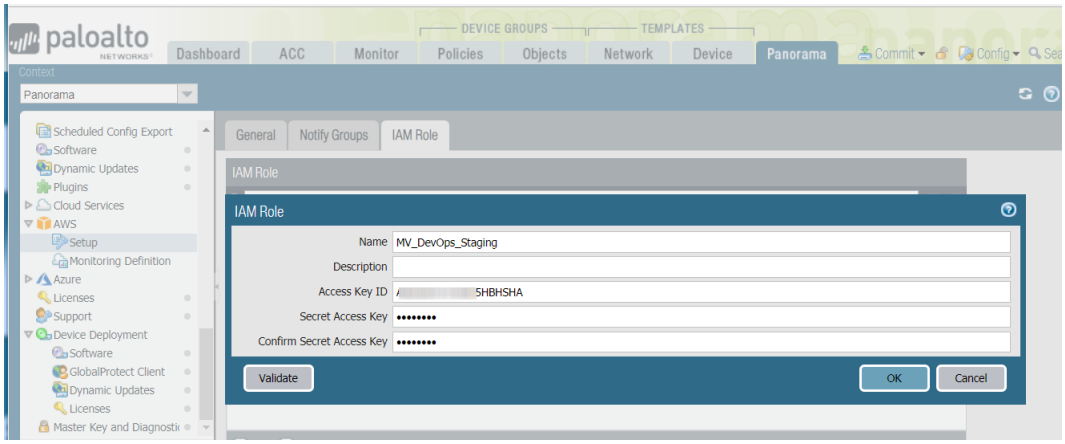
- Add an IAM Role.

  An IAM role is an entity that allows you delegate access so that the firewall can make service request on your behalf to the AWS resources (virtual machines that are deployed as EC2 instances).

  1. Select **Panorama** > **Plugins** > **AWS** > **Setup** > **IAM Role** > **Add**.



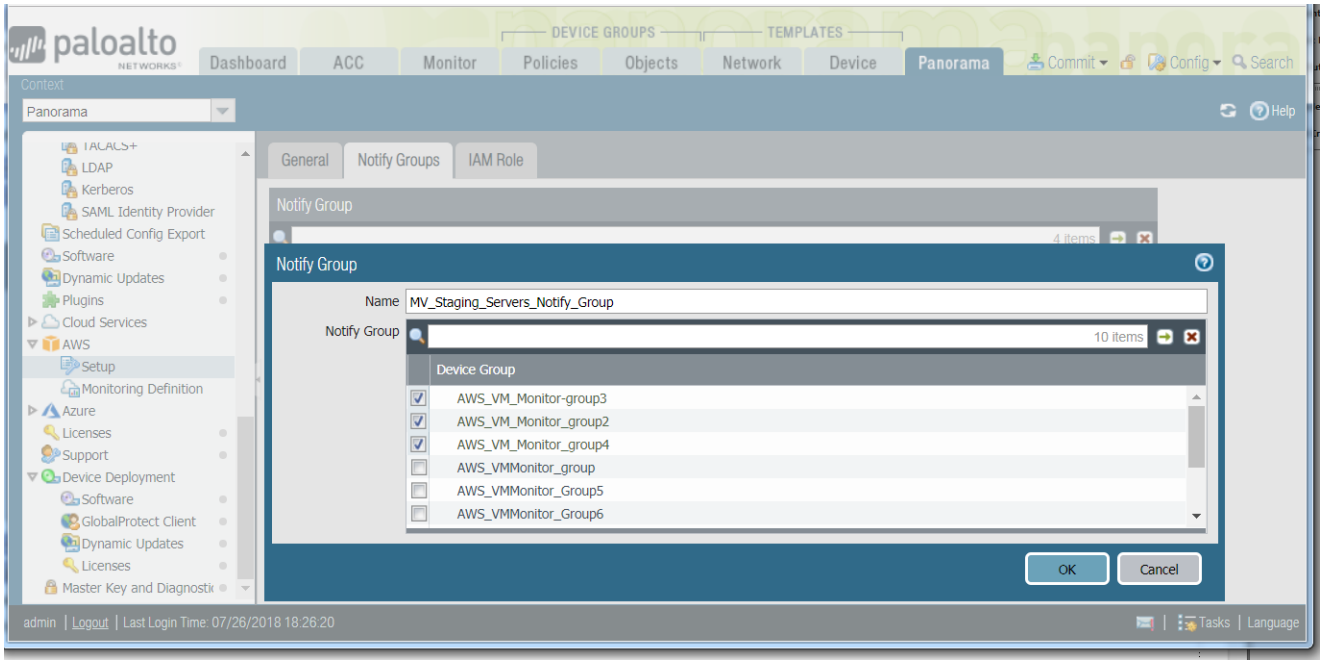
© Palo Alto Networks, Inc.




2. Enter a **Name** and optionally a **Description** to identify the IAM role.
3. Enter the **Access Key ID** for the AWS VPC you want to monitor. You must login to the AWS management console to get this key.
4. Enter the **Secret Access Key** and re-enter it to confirm.
5. Click **Validate** to verify that the keys and IDs you entered are valid, and Panorama can communicate with the AWS VPC using the API.

- Add a notify group.

1. Select **Panorama** > **Plugins** > **AWS** > **Setup** > **Notify Groups** > **Add**.




2. Enter a **Name** and optionally a **Description** to identify the group of firewalls to which Panorama pushes the VM information it retrieves.
3. Select the **Device Groups**, which are a group of firewalls or virtual systems, to which Panorama will push the VM information (IP address-to-tag mapping) it retrieves from your AWS VPCs. The firewalls use the update to determine the most current list of members that constitute dynamic address groups referenced in policy.

*Think through your Device Groups carefully.*

- *Because a Monitoring Definition can include only one notify group, make sure to select all the relevant Device Groups within your notify group. If you want to*



© Palo Alto Networks, Inc.

*deregister the tags that Panorama has pushed to a firewall included in a notify group, you must delete the Monitoring Definition.*

- *To register tags to all virtual systems on a firewall enabled for multiple virtual systems, you must add each virtual system to a separate device group on Panorama and assign the device groups to the notify group. Panorama will register tags to only one virtual system, if you assign all the virtual systems to one device group.*

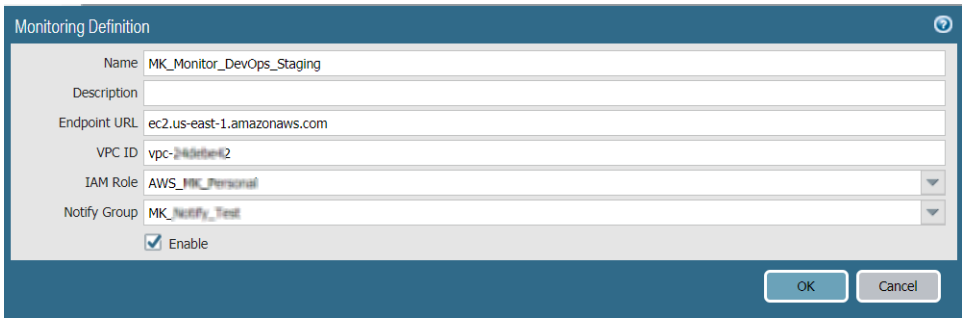
4. Verify that monitoring is enabled on the plugin. This setting must be enabled for Panorama to communicate with the AWS public cloud for VM Monitoring.

   The checkbox for **Enable Monitoring** is on **Panorama** > **Plugins** > **AWS** > **Setup** > **General**.

STEP 3 | Create a **Monitoring Definition**.

When you add a new Monitoring definition, it is enabled by default.

- Select **Panorama** > **Plugins** > **AWS** > **Monitoring Definition**, to **Add** a new definition.
- Enter a **Name** and optionally a **Description** to identify the AWS VPC for which you use this definition.
- Enter the **Endpoint URL**. The syntax is ec2.<your_region>.amazonaws.com
- Enter the **VPC ID** displayed on the VPC Dashboard on the AWS management console.
- Select the **IAM Role** and **Notify Group**.



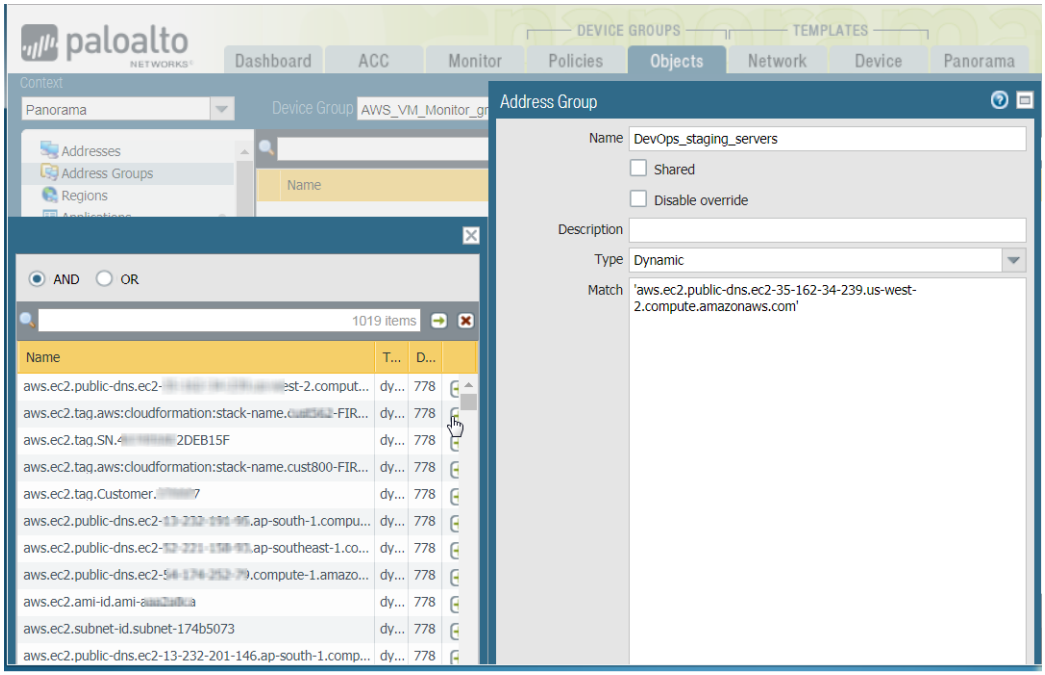

STEP 4 | **Commit** the changes on Panorama.

Verify that the status for the Monitoring Definition displays as Success. If it fails, verify that you entered the AWS VPC ID accurately and provided the correct keys and IDs for the authorizing access.

STEP 5 | Verify that you can view the VM information on Panorama, and define the match criteria for Dynamic Address Groups.



© Palo Alto Networks, Inc.




*On HA failover, the newly active Panorama attempts to reconnect to the AWS cloud and retrieve tags for all monitoring definitions. If Panorama is unable to reconnect with even one of the monitoring definitions that you have configured and enabled, Panorama generates a system log message*

```
Unable to process subscriptions after HA switch-over; user-intervention required.
```

*If this happens, you must log into Panorama and verify the monitoring definitions to fix invalid credentials or remove invalid subscriptions. Although Panorama is disconnected from the AWS cloud, all tags that were retrieved for the monitoring definitions before the failover, are retained and the firewalls can continue to enforce policy on that list of IP addresses. Panorama removes all tags associated with the subscriptions only when you delete a monitoring definition. As a best practice, to monitor this issue, you can configure action-oriented* log forwarding to an HTTPS destination *from Panorama so that you can take action immediately.*



© Palo Alto Networks, Inc.

# Auto Scale VM-Series Firewalls with the Amazon ELB Service

Palo Alto Networks delivers the VM-Series Auto Scale templates to enable auto scaling VM-Series next generation firewalls in AWS, protecting applications deployed on AWS.

The templates leverage AWS scalability features to independently scale the VM-Series firewalls to meet surges in application workload resource demand.

- VM-Series automation capabilities include the PAN-OS API and bootstrapping (using a bootstrap file for version 2.0, and Panorama for version 2.1).
- AWS automation technology includes CloudFormation templates and scripts for AWS services such as Lambda, auto scaling groups (ASGs), Elastic Load Balancing (ELB), S3, and SNS.

The templates are available on the Palo Alto Networks GitHub repository for Auto Scaling VM-Series Firewalls in AWS:

- Version 2.0 provides a firewall template and an application template. These templates and the supporting scripts deploy VM-Series firewalls, an internet facing firewall, an internal firewall, and application auto scaling groups in a single VPC or multiple VPCs.

  In version 2.0, Palo Alto Networks supports the firewall template, and the application template is community-supported. See VM-Series Auto Scale Template for AWS Version 2.0 for deployment details.
- Version 2.1 also supports deployment in a single VPC, and adds support for a load balancer sandwich topology that enables deploying the firewalls into a front end VPC, and the back end applications into one or more application VPCs connected by VPC peering or AWS PrivateLink.

  Version 2.1 can implement both application load balancers (ALBs) and network load balancers (NLBs) in VPCs. It supplies two firewall templates and five application templates. See VM-Series Auto Scale Templates for AWS Version 2.1 for deployment details.

*If you have an existing template deployment, there is no migration procedure.*

The following table compares some high-level features of each template version.

| Features / Requirements | Version 2.0 | Version 2.1 |
|---|---|---|
| Panorama Version 9.0.1 in Panorama mode<br>*Panorama in HA is not supported.* | Optional. If you choose to use Panorama you must configure VPC peering between the VM-Series firewall VPC and the application VPCs. Peered traffic traverses the public internet. | **Required** to deploy the Version 2.1 templates. |
| Bootstrapping | `bootstrap.xml` config file in an S3 bucket. | An `init-cfg.txt` file for Panorama. |
| Palo Alto Networks S3 bucket sample | Use your own S3 bucket or use the sample in panw-aws-autoscale-v20-us-west-2. | Use your own S3 bucket for the deployment. |



© Palo Alto Networks, Inc.

| Features / Requirements | Version 2.0 | Version 2.1 |
|---|---|---|
| Single VPC or separate VPCs (hub and spoke) | Yes | Yes |
| New VPC | Yes | Yes |
| Existing VPC (brown field) | No | Yes |
| Availability zones per VPC | 2 | 2-4 |
| External load balancer | ALB only | ALB or NLB |
| Internal load balancer | NLB only | ALB or NLB |
| AWS Private Link connection to the VM-Series firewall VPC and the backend servers. | No | Yes |

For details on the templates see:

- VM-Series Auto Scale Template for AWS Version 2.0
- VM-Series Auto Scale Templates for AWS Version 2.1

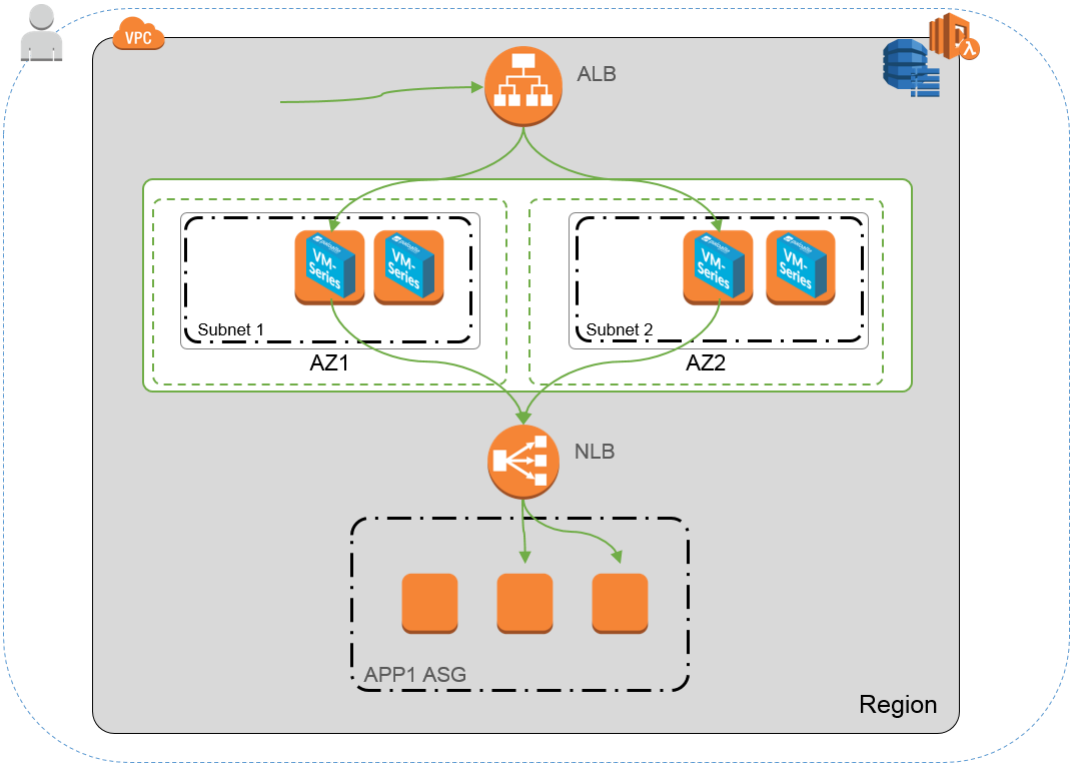
## VM-Series Auto Scale Template for AWS Version 2.0

To help you manage increased application scale, version 2.0 of the auto scaling VM-Series firewall template provides a hub and spoke architecture that simplifies deployment. This version of the solution provides two templates that support a single and multi-VPC deployment both within a single AWS account and across AWS accounts.

- **Firewall Template**—The firewall template deploys an application load balancer and VM-Series firewalls within auto scaling groups across two Availability Zones (AZs). This internet-facing application load balancer distributes traffic that enters the VPC across the pool of VM-Series firewalls. The VM-Series firewalls automatically publish custom PAN-OS metrics that enable auto scaling.

  Palo Alto Networks officially supports the firewall template, and with a valid support entitlement, you can request assistance from Palo Alto Networks Technical Support.


© Palo Alto Networks, Inc.




*The following application template deploys the network load balancer (NLB) depicted in the preceding image.*

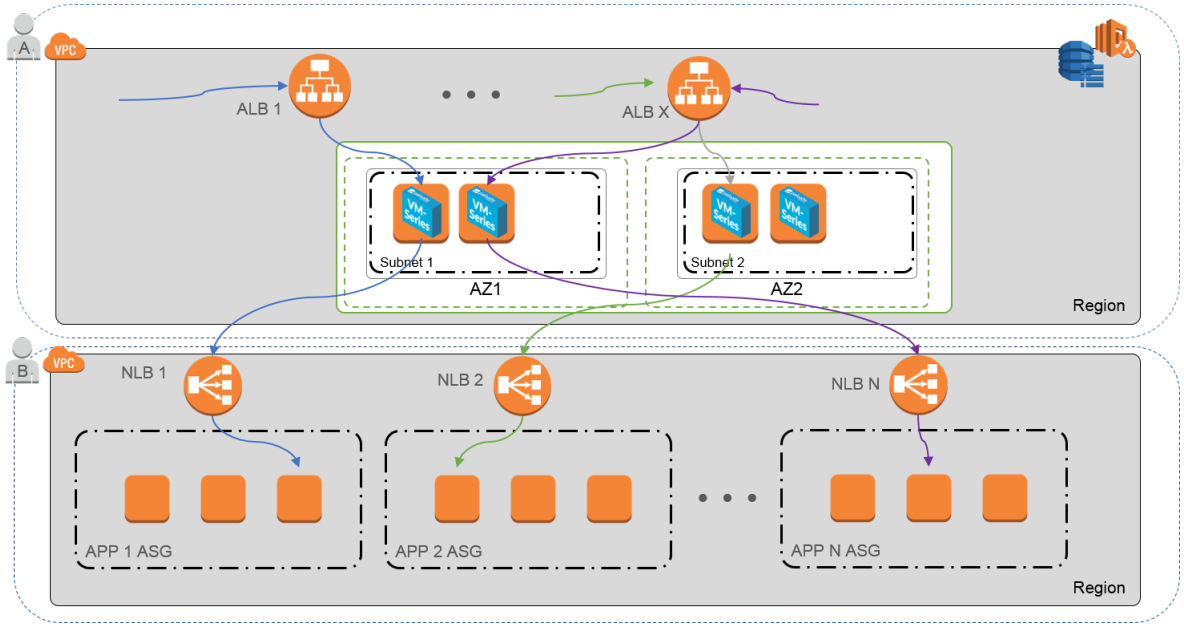
- **Application Template**—The application template deploys a network load balancer and one auto scaling group with a web server in each AZ.

  The application template is community supported. This template is provided as an example to help you get started with a basic web application. For a production environment, either use your own application template or customize this template to meet your requirements.

Together these templates allow you to deploy a load balancer sandwich topology with an internet-facing application load balancer and an internal network load balancer. The application load balancer is accessible from the internet and distributes traffic that enters the VPC across a pool of VM-Series firewalls. The firewalls then route traffic using NAT policy to the internal network load balancer(s), which distributes traffic to an auto scaling tier of web or application servers. The VM-Series firewalls are enabled to publish custom PAN-OS metrics to AWS CloudWatch where you can monitor the health and resource load on the VM-Series firewalls and then use that information to trigger a scale in or scale out event in the respective auto scaling group of firewalls.

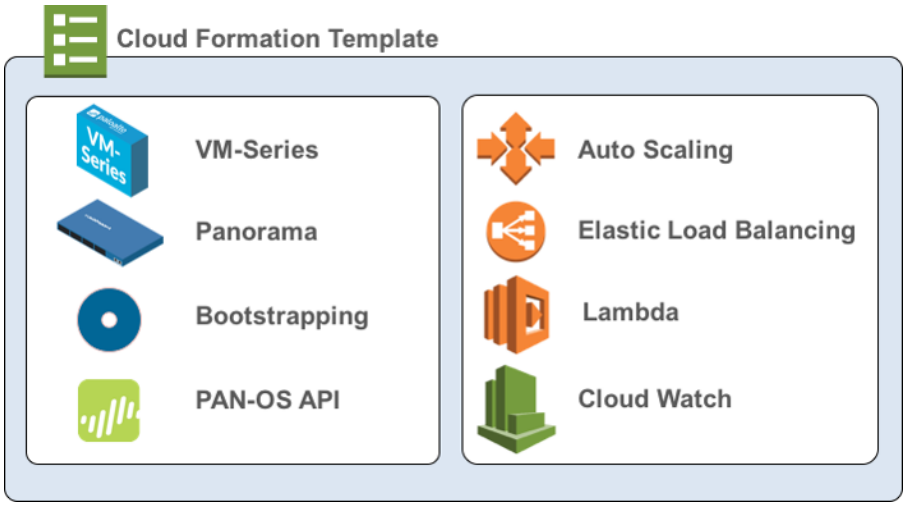


© Palo Alto Networks, Inc.






## *What Components Does the VM-Series Auto Scaling Template for AWS (v2.0) Leverage?*




The VM-Series Auto Scaling template for AWS includes the following building blocks:



© Palo Alto Networks, Inc.

| Building Block | Description |
| --- | --- |
| **Firewall template**<br>(Palo Alto Networks officially supported template) | The firewall-v2.0.template deploys a new VPC with two Availability Zones (AZs), subnets, route tables, and security groups required for routing traffic across these AZs, and an AWS NAT gateway. It also deploys an external application load balancer, and an Auto Scaling Group (ASG) with a VM-Series firewall in each AZ.<br>Due to the many variations in a production environment including but not limited to the number of subnets, availability zones, route tables, security groups etc., you must deploy the firewall-v2.0.template in a new VPC.<br>*VM-Series Auto Scaling template for AWS does not deploy Panorama, and Panorama is optional. Panorama provides ease of policy management and central visibility. If you want to use Panorama to manage the VM-Series firewalls that the solution deploys, you can either use an M-Series appliance or Panorama virtual appliance inside your corporate network, or a Panorama virtual appliance on AWS.*<br>This solution includes an AWS NAT gateway that the firewalls use to initiate outbound requests for retrieving updates, connecting to Panorama, and publishing metrics to AWS CloudWatch. |
| **Application template**<br>(Community supported template) | The application template deploys a network load balancer and an ASG with a web server in each AZ. Because the network load balancer has a unique IP address per AZ, and the NAT policy rule on the firewalls must reference a single IP address, there is one ASG for each of the two AZs. All the firewalls in an ASG have identical configuration.<br>This version of the auto scaling solution includes two application templates:<br>• The panw_aws_nlb-v2.0.template allows you to deploy the application template resources within same VPC as the one in which you deployed the firewall template (same AWS account).<br>• The panw_aws_nlb_vpcv-2.0.template allows you to deploy the application template resources in a separate VPC. This template supports both single and cross AWS account deployments. |
| **Lambda functions** | AWS Lambda provides robust, event-driven automation without the need for complex orchestration software. In the firewall-v2.0.template, AWS Lambda monitors a Simple Queue Service (SQS) to learn about network load balancers that publish to the queue. When the lambda function detects a new network load balancer, it creates a new NAT policy rule and applies it to the VM-Series firewalls within the ASG. The firewalls have a NAT policy rule for each application, and the firewalls use the NAT policy rule (that maps the port to network load balancer IP address) to forward traffic to the network load balancer in front of the application web servers.<br>*You need to create the Security policy rule to allow or deny application traffic for your deployment. The sample bootstrap.xml file does not include any Security policy rules. Using Panorama to centrally manage the firewalls simplifies the process of creating Security policy rules.* |



© Palo Alto Networks, Inc.

| Building Block | Description |
|---|---|
| | The Lambda functions also add or remove elastic network interfaces (ENIs) when the firewall is launched or terminated, delete all the associated resources when an instance is terminated or the stack is deleted, remove the firewall as a managed device on Panorama, and deactivate the BYOL license when a firewall is terminated on a scale in event.<br><br>To learn more about the lambda functions, refer to **http://paloaltonetworks-aws-autoscale-2-0.readthedocs.io/en/latest/** |
| **Bootstrap files**<br><br>The bootstrap.xml file provided in the GitHub repository is provided for testing and evaluation only. For a production deployment, you must modify the sample credentials in the bootstrap.xml prior to launch. | This solution requires the init-cfg.txt file and the bootstrap.xml file so that the VM-Series firewall has the basic configuration for handling traffic.<br><br>• The init-cfg.txt file includes the mgmt-interface-swap operational command to enable the firewall to receive dataplane traffic on its primary interface (eth0). This auto-scaling solution requires the swapping of the dataplane and management interfaces to enable the application load balancer to forward web traffic to the auto-scaling tier of VM-Series firewalls. For details see Management Interface Mapping for Use with Amazon ELB.<br>• The bootstrap.xml file enables basic connectivity for the firewall network interfaces and allows the firewall to connect to AWS CloudWatch namespace that matches the stack name you enter when launching the template. |

To deploy the solution, see Launch the VM-Series Auto Scaling Template for AWS (v2.0).

## How Does the VM-Series Auto Scaling Template for AWS (v2.0 and v2.1) Enable Dynamic Scaling?

The VM-Series firewalls that are deployed using the auto scaling template version 2.0 scale in and scale out based on custom PAN-OS metrics. The VM-Series firewalls natively publish these metrics to the Amazon CloudWatch console, and based on the metric(s) that you choose as the scaling parameter(s), you can define CloudWatch alarms and policies to dynamically deploy or terminate instances to handle the application traffic in your AWS deployment.

The firewalls publish metrics to AWS CloudWatch at a five-minute frequency (by default). When a metric that is being monitored reaches the configured threshold for the defined time interval, CloudWatch triggers an alarm and initiates an auto-scaling event.

When the auto-scaling event triggers the deployment of a new firewall, the new instance bootstraps at launch and a lambda function configures the firewall with NAT policy rules. A NAT policy rule is created for each application, and the rule references the IP addresses for each network load balancer in your deployment. When the application load balancer receives a request, it forwards the request to the firewall on the assigned TCP port. The firewall then inspects the traffic and forwards it to the corresponding network load balancer, which in turn forwards the request to a web server in its target group.

## Plan the VM-Series Auto Scaling Template for AWS (v2.0 and v2.1)

The items in this checklist are actions and choices you must make for implementing this solution.

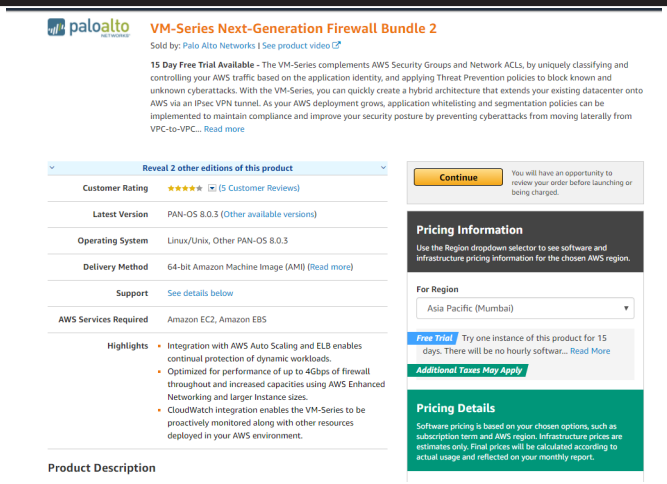
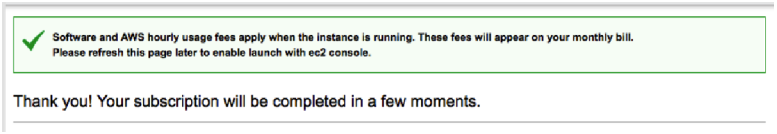


© Palo Alto Networks, Inc.

| Planning Checklist for Version 2.0 and 2.1 | |
|---|---|
| ❑ Verify the requirements for deploying the VM-Series Auto Scaling template. | The auto scaling template requires AWS Lambda and S3 Signature versions 2 or 4, and can deploy VM-Series firewalls running PAN-OS 8.1, or 9.0. You need to look up the list of supported regions and the AMI IDs, to provide as an input in the firewall template. |
| ❑ Assign the appropriate permissions for the IAM user role. | The user who deploys the VM-Series Auto Scaling template must either have administrative privileges or have the permissions listed in the iam-policy.json to launch this solution successfully. Copy and paste the permissions from this file in to a new IAM policy and then attach the policy to a new or existing IAM role.<br>For a cross-account deployment, to access resources that are in a different AWS accounts, the IAM role for the user who deploys the application template must have full SQS access permissions and a trust relationship that authorizes her to write to the SQS queue that belongs to the firewall template. |
| ❑ Collect the details required for a cross-account deployment. | For a deployment where the firewall template and the application template are in different accounts, the account that hosts the firewall template resources is the trusting account and the other AWS account(s) that hold the application template resources are the trusted accounts. To launch the application template in a cross-account deployment, you need the following information:<br>• Cross-account Role Amazon Resource Name (ARN) of the account in which you are deploying the application template.<br>• External ID, which you defined when creating the IAM role that grants full SQS access to the trusting account.<br>• The 10-digit account number for every AWS account in which you plan to launch the application template. Because the account that hosts the firewall template resources serves as a trusting account, and it owns the resources that the users of the application template need, you need to list the account number for each trusted account that can access the firewall resources. |
| ❑ Create a support account on the Palo Alto Networks Support portal, if you don't already have one. | You can opt for the BYOL or PAYG licenses.<br>• For BYOL, you must register an auth code to your Palo Alto Networks support account prior to launching the VM-Series Auto Scaling template and add the auth-code to the `/license` folder with filename as authcodes in the bootstrap package. See Launch the VM-Series Auto Scaling Template for AWS (v2.0) or Launch the Firewall Template (v2.1) for details.<br>• For PAYG, you must register the VM-Series firewalls to activate your support entitlement. |
| ❑ (For PAYG only) Review and accept the End User License Agreement (EULA). | In the AWS Marketplace, search for Palo Alto Networks, and select the bundle you plan to use. The VM-Series firewalls will fail to deploy if you have not accepted the EULA for the bundle you plan to use.<br>• Search for **VM-Series Next Generation Firewall Bundle 2, for example**. |



© Palo Alto Networks, Inc.

| | |
|---|---|
| Required, if you are launching a VM-Series firewall in an AWS account for the first time. |  <br>• Click **Continue**, and select **Manual Launch**. Review the agreement and click **Accept Software Terms** to accept the EULA.<br>  <br>You can now close the browser. |
| ❑ Decide whether you plan to use the public S3 buckets or your private S3 bucket for AWS Lambda, Python scripts, and templates. | Palo Alto Networks provides public S3 buckets in all AWS regions included in the supported regions list. These S3 buckets include all the templates, AWS Lambda code, and the bootstrap files that you need.<br><br>*Palo Alto Networks recommends using the bootstrap files in the public S3 bucket only for evaluating this solution. For a production deployment, you must create a private S3 bucket for the bootstrap package.*<br><br>The naming convention for the S3 bucket is `panw-aws-autoscale-v20-<region_name>`. For example, the bucket in the AWS Oregon region is panw-aws-autoscale-v20-us-west-2.<br><br>To use your private S3 bucket, you must download and copy the templates, AWS Lambda code, and the bootstrap files to your private S3 bucket. You can place all the required files for both the firewall template and the application template in one S3 bucket or place them in separate S3 buckets. |
| ❑ Download the templates, AWS Lambda code, and the bootstrap files. | • Get the files for deploying the firewall template (application load balancer and the VM-Series firewalls) from the GitHub repository at: https://github.com/PaloAltoNetworks/aws-elb-autoscaling/tree/master/Version-2.0<br><br>*Do not mix and match files across VM-Series Auto Scaling template versions.*<br><br>• Templates and Lambda code: |



© Palo Alto Networks, Inc.

| Planning Checklist for Version 2.0 and 2.1 | |
|---|---|
| | • panw-aws.zip<br>• firewall-v2.0.template<br>• Bootstrap files:<br>• init-cfg.txt<br>• bootstrap.xml<br>The bootstrap.xml file bundled with this solution is designed to help you get started, and is provided for testing and evaluation only. For a production deployment, you must modify the bootstrap.xml prior to launch.<br>• iam-policy: The user who deploys the VM-Series Auto Scaling template must have either the administrative privileges or the permissions listed in this file to successfully launch this solution.<br>*The firewall template is supported by Palo Alto Networks Technical Support.*<br>• Get the files for deploying the NLB and the web servers from the GitHub repository at: https://github.com/PaloAltoNetworks/pan_nlb_v1<br>• Templates:<br>• pan_aws_nlb-2.0.template—Use this template to deploy the application template resources within same VPC as the one in which you deployed the firewall template (same AWS account).<br>• pan_aws_nlb_vpc-2.0.template—Use this template to deploy the application template resources in a different VPC. This template allows you to deploy the resources within the same AWS account or in a different AWS account as long as you have the appropriate permissions to support a cross-account deployment.<br>• pan_nlb_lambda.template<br>• Lambda code and Python scripts. |
| ❑ Customize the bootstrap.xml file for your production environment. | To ensure that your production environment is secure, you must customize the bootstrap.xml file with a unique administrative username and password for production deployments. The default username and password are pandemo/demopassword. You can also use this opportunity to create an optimal firewall configuration with interfaces, zones, and security policy rules that meet your application security needs. |
| ❑ Decide whether you want to use Panorama for centralized logging, reporting, and firewall management. | Panorama is an option for administrative ease and is the best practice for managing the firewalls. It is not required to manage the auto scaling tier of VM-Series firewalls deployed in this solution.<br>If you want to use Panorama, you can either a Panorama virtual appliance on AWS or use an M-Series appliance or a Panorama virtual appliance inside your corporate network.<br>*The Panorama must be in Panorama mode and not Management Only mode.* |



© Palo Alto Networks, Inc.

| Planning Checklist for Version 2.0 and 2.1 | |
|---|---|
| | To successfully register the firewalls with Panorama, you must collect the following details:<br>• API key for Panorama—So that AWS Lambda can make API requests to Panorama, you must provide an API key when you launch the VM-Series Auto Scaling template. As a best practice, in a production deployment, create a separate administrative account just for the API call and generate an associated API key.<br>• Panorama IP address—You must include the IP address in the configuration (init-cfg.txt) file. The firewalls must be able to access this IP address from the VPC; to ensure a secure connection, use a direct connect link or an IPSec tunnel.<br>• VM auth key—Allows Panorama to authenticate the firewalls so that it can add each firewall as a managed device. You must include this key in the configuration (init-cfg.txt) file.<br>The vm auth key is required for the lifetime of the deployment. Without a valid key in the connection request, the VM-Series firewall will be unable to register with Panorama. For details on the key, see Generate VM Auth Key.<br>• Template stack name and the device group name to which to assign the firewalls—You must first add a template and assign it to a template stack, create a device group on Panorama, and then include the template stack name and the device group name in the configuration (init-cfg.txt) file.<br>*In order to reduce the cost and scale limits of using Elastic IP addresses, the firewalls do not have public IPs. If you are not using Panorama to manage the firewalls, you must deploy a jump server (a bastion host with an EIP address) that attaches to the Untrust subnet within the VPC to enable SSH and/or HTTPS access to the VM-Series firewalls. By default, this solution includes an AWS NAT gateway that the firewalls use to initiate outbound requests for retrieving updates, connecting to Panorama, and publishing metrics to AWS CloudWatch.* |
| **Get started** | Launch the VM-Series Auto Scaling Template for AWS (v2.0) |

## *Customize the Firewall Template Before Launch (v2.0 and v2.1)*

To simplify the deployment workflow, the firewall displays a limited set of parameters for which you need to provide inputs when launching the template. If you would like to view and customize other options included in the template, you can use a text editing tool such as Notepad or Visual Studio Code to specify values that you prefer before you launch the VM-Series Auto Scaling template for AWS v2.0 or 2.1.

Use the following table to view the list of parameters that you are allowed to customize for your deployment of the auto scaling firewall template for AWS. Modifying parameters from this list is within the official support policy of Palo Alto Networks through the support options that you've purchased.



© Palo Alto Networks, Inc.

| Parameter | Description | Default Value |
|---|---|---|
| CIDR Block for the VPC | The IP address space that you want to use for the VPC.<br>  *The subnets you modify below must belong to this VPC CIDR block and be unique.* | 192.168.0.0/16 |
| Management Subnet CIDR Block | Comma-delimited list of CIDR blocks for the management subnet of the firewalls. | 192.168.0.0/24, 192.168.10.0/24 |
| Untrust Subnet CIDR Block | Comma-delimited list of CIDR blocks for the Untrust subnet. | 192.168.1.0/24, 192.168.11.0/24 |
| Trust Subnet CIDR Block | Comma-delimited list of CIDR blocks for the Trust subnet. | 192.168.2.0/24, 192.168.12.0/24 |
| NAT Gateway Subnet CIDR Block | Comma-delimited list of CIDR blocks for the AWS NAT Gateway. | 192.168.100.0/24, 192.168.101.0/24 |
| Lambda Subnet CIDR Block | Comma-delimited list of CIDR blocks for the Lambda functions. | 192.168.200.0/24, 192.168.201.0/24 |
| Firewall Instance size | AWS Instance Types and size that you want for the VM-Series firewalls in your deployment. | M4.xlarge |
| Choose your Scaling Parameter<br>  *You do not need to modify the template for the scaling parameter. You can set AWS CloudWatch alarms on the AWS console for one or more custom PAN-OS metrics on which you want to trigger autoscaling.* | The template publishes all the following metrics to AWS CloudWatch:<br>• CPU—DataPlane CPU Utilization<br>• AS—Active Sessions<br>• SU—Session Utilization<br>• SSPU—SSL Proxy Utilization<br>• GPU—GlobalProtect Gateway Utilization<br>• GPAT—GlobalProtect Gateway Utilization ActiveTunnels<br>• DPB—Dataplane Packet Buffer Utilization | Dataplane CPU Utilization |
| Choose time in seconds for Scaling Period | The period in seconds over which the average statistic is applied. Must be a multiple of 60. | 900 |



© Palo Alto Networks, Inc.

| Parameter | Description | Default Value |
|---|---|---|
| Maximum VM-Series Instances | Maximum number of VM-Series firewalls in the auto scaling group. | 3 |
| Minimum VM-Series Instances | Minimum number of VM-Series firewalls in the auto scaling group. | 1 |
| ScaleDown threshold value in percentage/value | Value at which a scale in event is triggered. | 20 |
| ScaleUp threshold value in percentage/value | Value at which scale out event is triggered. | 80 |

## *Launch the VM-Series Auto Scaling Template for AWS (v2.0)*

You can choose to deploy the firewall template in one VPC and the sample application template in the same VPC as the one in which you deployed the firewalls, or in a different VPC.

If the applications that you want to secure belong to a separate AWS account, the sample application template includes support for cross-account deployments. The solution supports a hub and spoke architecture whereby you can deploy the firewall template in one AWS account and use it as a hub to secure your applications (spokes) that belong to the same or to different AWS accounts.

- Launch the VM-Series Firewall Template
- Launch the Application Template
- (Required only if you deploy more than one internal load balancer) Enable Traffic to the ELB Service (v2.0 and v2.1)

### Launch the VM-Series Firewall Template

This workflow tells you how to deploy the application load balancer and the VM-Series firewalls using the firewall template.

*This firewall template includes an AWS NAT gateway that the firewalls use to initiate outbound requests for retrieving updates, connecting to Panorama, and publishing metrics to AWS CloudWatch. If you are not using Panorama to manage the firewalls, you must deploy a jump server (a bastion host with an EIP address) that attaches to the Untrust subnet within the VPC to enable SSH and/or HTTPS access to the VM-Series firewalls. This jump server is required because the management interface on the VM-Series firewalls has a private IP address only.*

STEP 1 | Reviewed the checklist for Plan the VM-Series Auto Scaling Template for AWS (v 2.0).

Make sure that you have completed the following tasks:

- (For PAYG only) Reviewed and accepted the EULA for the PAYG bundle you plan to use.
- (For BYOL only) Obtained the auth code. You need to enter this auth code in the /license folder of the bootstrap package.
- Downloaded the files required to launch the VM-Series Auto Scaling template from the GitHub repository.

STEP 2 | (Optional) Modify the init-cfg.txt file.

For more details read about the bootstrapping process and the init-cfg.txt file.



© Palo Alto Networks, Inc.

If you're using Panorama to manage the firewalls, complete the following tasks:

1. Generate the VM-auth key on Panorama. The firewalls must include a valid key in the connection request to Panorama. Set the lifetime for the key to 8760 hours (1 year).
2. Open the init-cfg.txt file with a text editor, such as Notepad. Make sure that you do not alter the format as this causes a failure in deploying the VM-Series Auto Scaling template. Add the following information as name-value pairs:
   - IP addresses for the primary Panorama and optionally a secondary Panorama. Enter:

     **panorama-server=**

     **panorama-server-2=**
   - Specify the template stack name and the device group to which you want to assign the firewall. Enter:

     **tplname=**

     **dgname=**
   - VM auth key. Enter:

     **vm-auth-key=**
3. Verify that you have not deleted the command for swapping the management interface (mgmt) and the dataplane interface (ethernet 1/1) on the VM-Series firewall on AWS. For example, the file must include name-value pairs as shown here:

   **op-command-modes=mgmt-interface-swap**

   **vm-auth-key=755036225328715**

   **panorama-server=10.5.107.20**

   panorama-server-2=10.5.107.21

   tplname=FINANCE_TG4

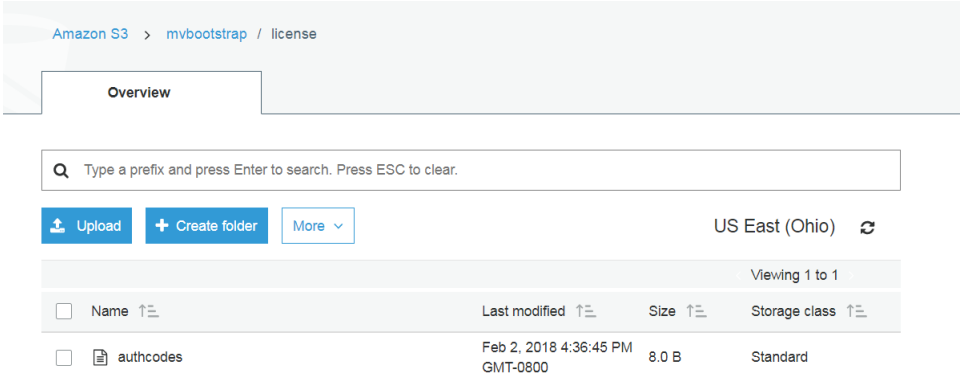
   dgname=finance_dg
4. Save and close the file.

STEP 3 | (For BYOL only) Add the license auth code in the /license folder of the bootstrap package. For more information see prepare the bootstrap package.

1. Create a new .txt file with a text editor, such as Notepad.
2. Add the authcode for your BYOL licenses to this file,then save the file with authcodes (no file extension) and upload it to the /license folder. The auth code must support the number of firewalls that may be required for your deployment. You must use an auth code bundle instead of individual auth codes so that the firewall can simultaneously fetch all license keys associated with a firewall. If you use individual auth codes instead of a bundle, the firewall retrieves only the license key for the first auth code included in the file.



© Palo Alto Networks, Inc.




STEP 4 | Change the default credentials for the VM-Series firewall administrator account defined in the bootstrap.xml file.

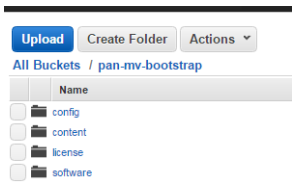
Required for using the VM-Series Auto Scaling template in a production environment.

The bootstrap.xml file in the GitHub repository is provided for testing and evaluation only. For a production deployment, you must Customize the Bootstrap.xml File (v2.0) prior to launch.

STEP 5 | Prepare the Amazon Simple Storage (S3) buckets for launching the VM-Series Auto Scaling template to a production environment.

*Make sure to create the S3 buckets in the same region in which you plan to deploy the template; the bootstrapping files hosted in the public S3 bucket are provided only to make it easier for you to evaluate the template.*

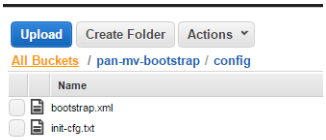
1. Create a new S3 bucket for the bootstrap files.
    1. Sign in to the AWS Management Console and open the S3 console.
    2. Click **Create Bucket**.
    3. Enter a **Bucket Name** and a **Region**, and click **Create**. The bucket must be at the S3 root level. If you nest the bucket, bootstrapping fails because you cannot specify a path to the location of the bootstrap files.
2. Upload the bootstrap files to the S3 bucket. The bootstrap folders must be in the root folder of the S3 bucket.
    1. Click the name of bucket and then click **Create folder**.
    2. Create the following folder structure for bootstrapping.



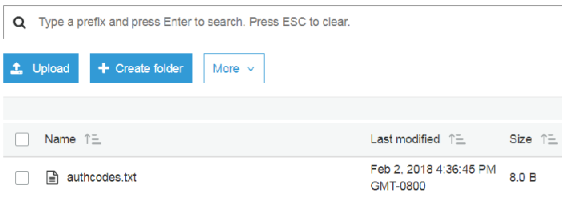

    3. Click the link to open the **config** folder.
    4. Select **Actions** > **Upload** and **Add Files**, browse to select the init-cfg.txt file and bootstrap.xml file, and click **Open**.
    5. Click **Start Upload** to add the files to the config folder. The folder can contain only two files: init-cfg.txt and the bootstrap.xml.



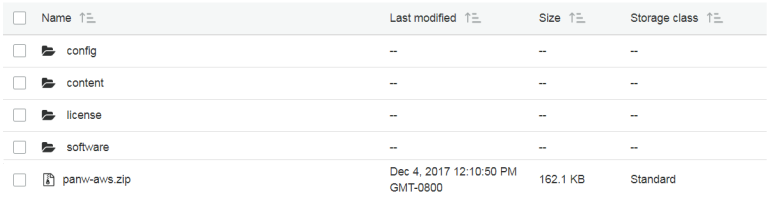
© Palo Alto Networks, Inc.




6. (For BYOL only) Click the link to open the **license** folder and upload the txt file with the auth code required for licensing the VM-Series firewalls.




3. Upload the AWS Lambda code (panw-aws.zip file) to an S3 bucket. In this example, the AWS Lambda code is in the same S3 bucket as the bootstrap package.
   1. Click the bucket name.
   2. Click **Add Files** to select the panw-aws.zip file, click **Open**.
   3. Click **Start Upload** to add the zip file to the S3 bucket.




STEP 6 | Select the firewall template.

If you need to Customize the Firewall Template Before Launch (v2.0), do that now and select the modified template.

1. In the AWS Management Console, select **CloudFormation** > **Create Stack**.
2. Select **Upload a template to Amazon S3**, choose the firewall-v2.0.template and click **Open** and **Next**.
3. Specify the **Stack name**. The stack name allows you to uniquely identify all the resources that this template deploys.

STEP 7 | Configure the parameters for the VPC.

1. Enter the parameters for the **VPC Configuration** as follows:
   1. Enter a **VPCName**.
   2. Select the two Availability Zones that your setup spans in **Select two AZs**.

STEP 8 | Select your preferences for the VM-Series firewalls.



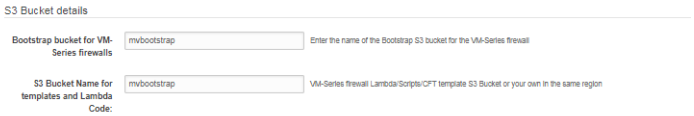
© Palo Alto Networks, Inc.


1. Look up the AMI ID for the VM-Series firewall and enter it. Make sure that the AMI ID matches the AWS region, PAN-OS version and the BYOL or PAYG licensing option you opted to use.
2. Select the EC2 **Key pair** (from the drop-down) for launching the firewall. To log in to the firewalls, you must provide the name of this key pair and the private key associated with it.
3. Restrict SSH access to the firewall's management interface. Make sure to supply a CIDR block that corresponds to your dedicated management IP addresses or network. Do not make the allowed source network range larger than necessary and do not ever configure the allowed source as 0.0.0.0/0. Verify your IP address before configuring it on the template to make sure that you do not lock yourself out.
4. Select **Yes** if you want to **Enable Debug Log**. Enabling the debug log generates more verbose logs that help with troubleshooting issues with the deployment. These logs are generated using the stack name and are saved in AWS CloudWatch.

By default, the template uses CPU utilization as the scaling parameter for the VM-Series firewalls. Custom PAN-OS metrics are automatically published to the CloudWatch namespace that matches the stack name you specified earlier.

STEP 9 | Specify the name of the Amazon S3 bucket(s).

*You can use one S3 bucket for the bootstrap package and the zip file.*

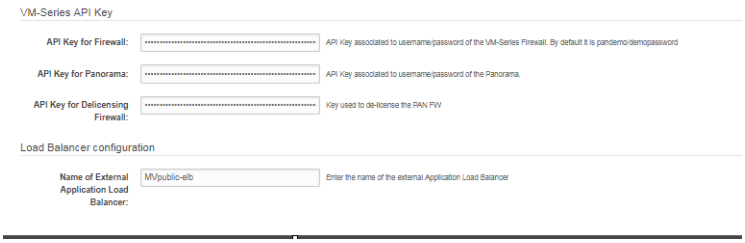



1. Enter the name of the S3 bucket that contains the bootstrap package.

   If the bootstrap bucket is not set up properly or if you enter the bucket name incorrectly, the bootstrap process fails and you cannot be able to log in to the firewall. Health checks for the load balancers also fail.
2. Enter the name of the S3 bucket that contains the panw-aws.zip file.

STEP 10 | Specify the keys for enabling API access to the firewall and Panorama.


© Palo Alto Networks, Inc.




1. Enter the key that the firewall must use to authenticate API calls. The default key is based on the sample bootstrap.xml file and you should only use it for testing and evaluation. For a production deployment, you must create a separate PAN-OS login just for the API call and generate an associated key.
2. Enter the API Key to allow AWS Lambda to make API calls to Panorama, if you are using Panorama for centralized management. For a production deployment, you should create a separate login just for the API call and generate an associated key.
3. Copy and paste the license deactivation API key for your account. This key is required to successfully deactivate licenses on your firewalls when a scale-in event occurs. To get this key:
    1. Log in to the Customer Support Portal.
    2. From the **Go To** drop-down, select **License API**.
    3. Copy the API key.

STEP 11 | Enter the name for the application load balancer.

STEP 12 | (Optional) Apply tags to identify the resources associated with the VM-Series Auto Scaling template.

Add a name-value pair to identify and categorize the resources in this stack.

STEP 13 | Review the template settings and launch the template.

1. Select **I acknowledge that this template might cause AWS CloudFormation to create IAM resources**.
2. Click **Create** to launch the template. The CREATE_IN_PROGRESS event displays.
3. On successful deployment the status updates to CREATE_COMPLETE.

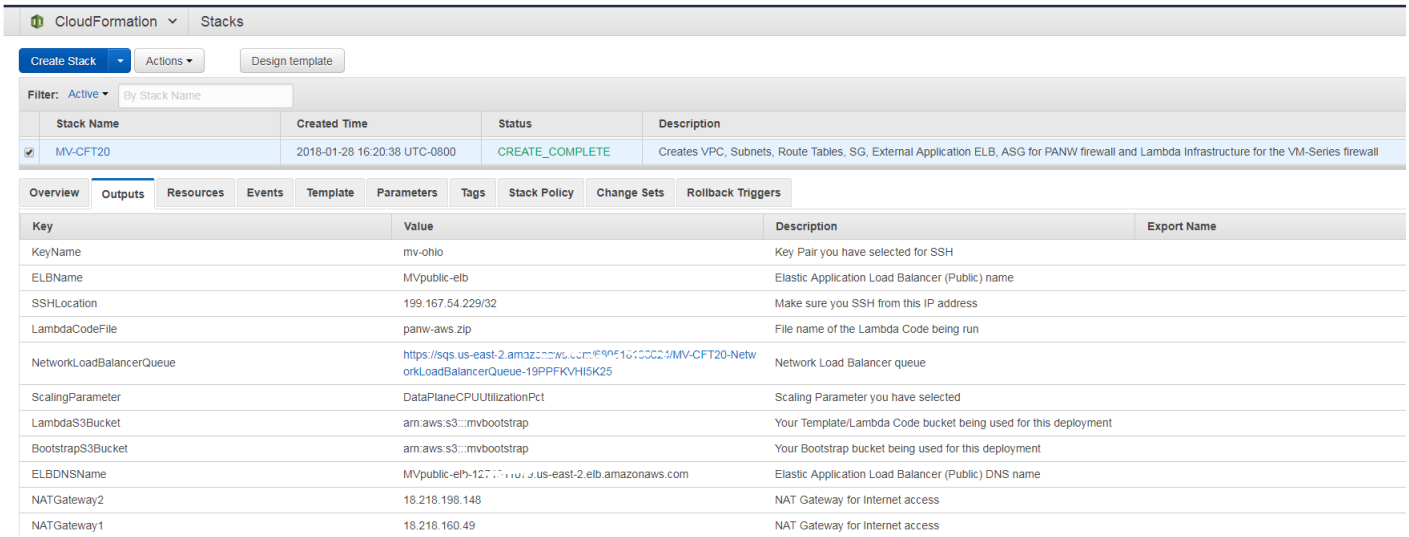

Unless you customized the template, the VM-Series Auto Scaling template launches an ASG that includes one VM-Series firewall in each AZ, behind the application load balancer.

STEP 14 | Verify that the template has launched all required resources.

1. On the AWS Management Console, select the stack name to view the **Output** for the list of resources.



© Palo Alto Networks, Inc.




2. On the EC2 Dashboard, select **Auto Scaling Groups**. Verify that in each AZ, you have one ASG for the VM-Series firewalls with the one firewall in each ASG. The ASG name prefix includes the stack name.
3. Log in to the VM-Series firewall. You must deploy a jump server or use Panorama to access the user interface on the firewall.

- *It may take up to 20 minutes for the firewalls to boot up and be available to handle traffic.*
- *When you finish testing or a production deployment, the only way to ensure charges stop occurring is to completely delete the stack. Shutting down instances, or changing the ASG maximum to 0 is not sufficient.*

STEP 15 | Save the following information. You need to provide these values as inputs when deploying the application template.

- IP addresses of the NAT Gateway in each AZ. You need this IP address to restrict HTTP access to the web servers if you deploy the application in a different VPC. Specifying this IP address ensures that the firewall secures access your applications in a different VPC, and that nobody can bypass the firewall to directly access the web server. The sample application template (panw_aws_nlb_vpc-2.0.template) displays a template validation error if you do not enter the NAT Gateway IP addresses; you must enter the IP addresses as a comma-separated list.
- Network Load Balancer SQS URL. A lambda function in the firewall stack monitors this queue so that it can learn about any network load balancers that you deploy, and create NAT policy rules (one per application) on the VM-Series firewalls that enable the firewalls to send traffic to the network load balancer IP address.

### Launch the Application Template

The application template allows you to complete the sandwich topology and is provided so that you can evaluate the auto scaling solution. This application template deploys a network load balancer and a pair of web servers behind the auto scaling group of VM-Series firewalls, which you deployed using the firewall template. The web servers in this template have a public IP address for direct outbound access to retrieve software updates. Use this template to evaluate the solution, but build your own template to deploy to production. For a custom template, make sure to enable SQS Messaging Between the Application Template and Firewall Template.

When launching the application template, you must select the template based on whether you want to deploy the application template within the same VPC (panw_aws_nlb-2.0.template) in which you deployed the firewall template or in a separate VPC (panw_aws_nlb_vpc-2.0.template). For a separate VPC, the

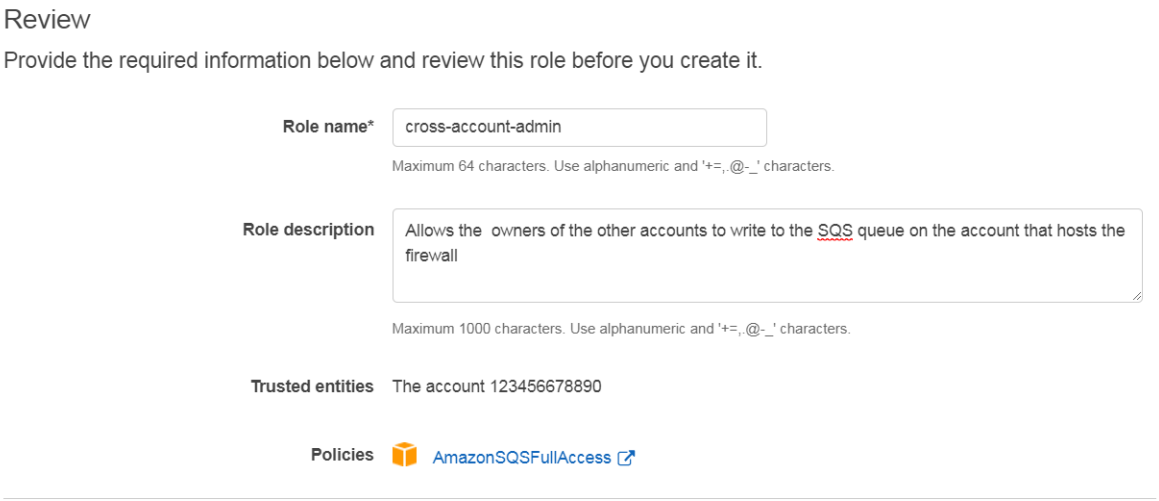

© Palo Alto Networks, Inc.

template provides supports for cross-account deployments. A cross-account deployment requires you to create an IAM role and enable permissions and trust relationship between the trusting AWS account and the trusted AWS account, and the account information is required as input when launching the template.

STEP 1 | (Required only for a cross-account deployment) Create the IAM role. Refer to AWS documentation.

This role grants access to a user who belongs to a different AWS account. This user requires permissions to access the Simple Queue Service (SQS) resource in the firewall template. The firewall uses this queue to learn about each network load balancer that you deploy so that it can create NAT policy to send traffic to the web servers that are behind the network load balancer.

- For **Account ID**, type the AWS account ID of the account into which you are deploying the application template. Specifying that account ID allows you to grant access to the resources in your account that hosts the firewall template resources.
- Select **Require external ID** and enter a value that is a shared secret. Specifying an external ID allows the user to assume the role only if the request includes the correct value.
- Choose **Permissons** to allow **Amazon SQS Full Access**.



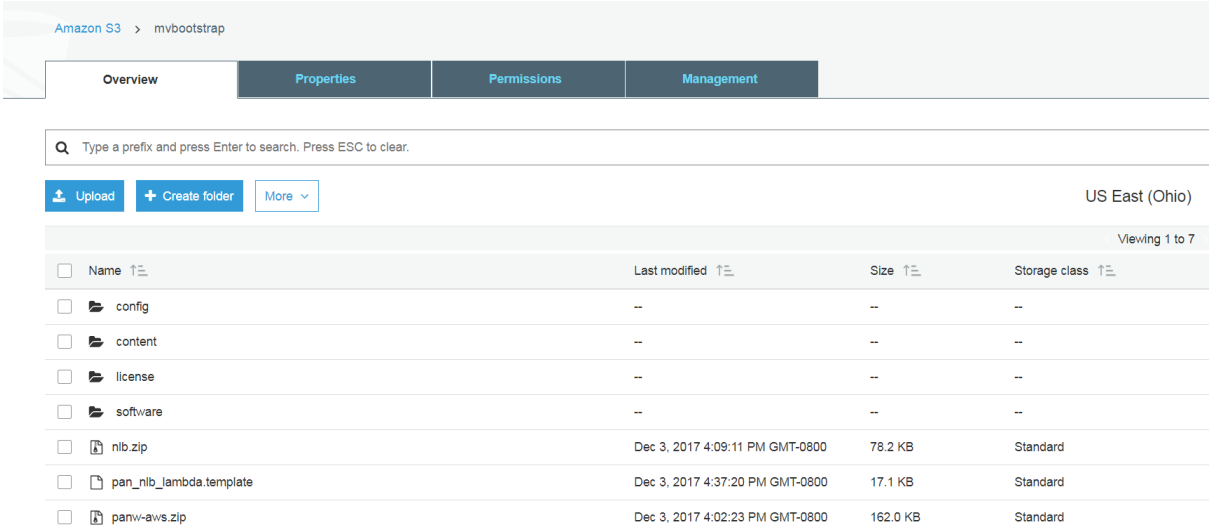

STEP 2 | Use the Palo Alto Networks public S3 bucket or prepare your private (S3) bucket for launching the application template.

The application template is available at: https://github.com/PaloAltoNetworks/pan_nlb_v1.

1. Create a zip file with all the files in the GitHub repository, excluding the three .template files, named nlb.zip in the screenshot below.
2. Upload the zip file to the S3 bucket you created earlier or to a new bucket.



© Palo Alto Networks, Inc.




3. Copy the pan_nlb_lambda template into the same bucket to which you copied the nlb.zip file.

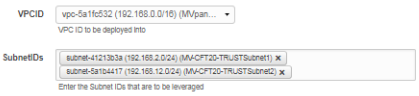
STEP 3 | Select the application template to launch.

1. In the AWS Management Console, select **CloudFormation** > **Create Stack**.
2. Select **Upload a template to Amazon S3**, to choose the panw_aws_nlb-2.0.template to deploy the resources that the template launches within the same VPC as the firewalls, or the panw_aws_nlb_vpc-2.0.template to deploy the resources in to a different VPC. Click **Open** and **Next**.
3. Specify the **Stack name**. The stack name allows you to uniquely identify all the resources that are deployed using this template.

STEP 4 | Configure the parameters for the VPC and network load balancer.

1. Select the two Availability Zones that your setup will span in **Select list of AZ**. If you are deploying within the same VPC make sure to select the same Availability Zones that you selected for the firewall template.
2. Enter a **CIDR Block for the VPC**. The default CIDR is 192.168.0.0/16.


3. (Only if you are using the panw_aws_nlb-2.0.template to deploy the applications within the same VPC)

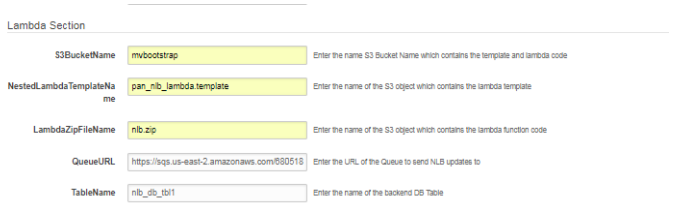



Select the **VPC ID** and the **Subnet IDs** associated with the trust subnet on the firewalls in each AZ. The network load balancer is attached to the trust subnet on the firewalls, to complete the load balancer sandwich topology.

4. Enter a name for the network load balancer.

STEP 5 | Configure the parameters for Lambda.


© Palo Alto Networks, Inc.



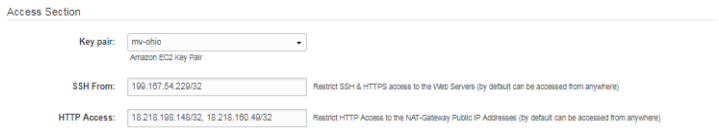

1. Enter the S3 bucket name where nlb.zip and the pan_nlb_lambda.template is stored.
2. Enter the name of the pan_nlb_lambda.template and the zip file name.
3. Paste the SQS URL that you copied earlier.
4. Enter a unique **TableName**. This table stores a mapping of the port and IP address for the applications associated with the network load balancer in your deployment.

   When you delete the application stack this table is deleted. Therefore, if multiple instances of the network load balancer write to the same table and the table is deleted, the NAT rules on the firewalls not function properly and the application traffic maybe be inaccurately forwarded to the wrong port/ network load balancer.

STEP 6 | Modify the web server EC2 instance type to meet your deployment needs.

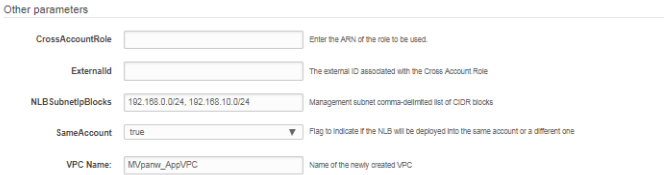
STEP 7 | Select the EC2 **Key pair** (from the drop-down) for launching the web servers. To log in to the web servers, you must provide the key pair name and the private key associated with it.

STEP 8 | (Only if you are using the panw_aws_nlb_vpc-2.0.template) Lock down access to the web servers.




1. Restrict **SSH From** access to the web servers. Only the IP addresses you list here can log in to the web servers.
2. Restrict HTTP access to the web servers. Enter the public IP addresses of the NAT gateway from the firewall template output, and make sure to separate IP addresses with commas. Entering the NAT gateway IP address allows you to ensure that all web traffic to the application servers are secured by the VM-Series firewalls.

STEP 9 | (Only if you are using the panw_aws_nlb_vpc-2.0.template) Configure the other parameters requires to launch the application template stack in a different VPC.




1. Select SameAccount **true** if you are deploying this application template within the same AWS account as the firewall template, and leave the cross account role and external ID blank; select **false** for a cross-account deployment.

   For a cross-account deployment, enter the Amazon Resource Number (ARN) for the **CrossAccountRole** and **ExternalId** that you defined in (Required only for a cross-account deployment) Create the IAM role. Refer to AWS documentation.You can get the ARN from **Support** > **Support Center** on the AWS Management Console.

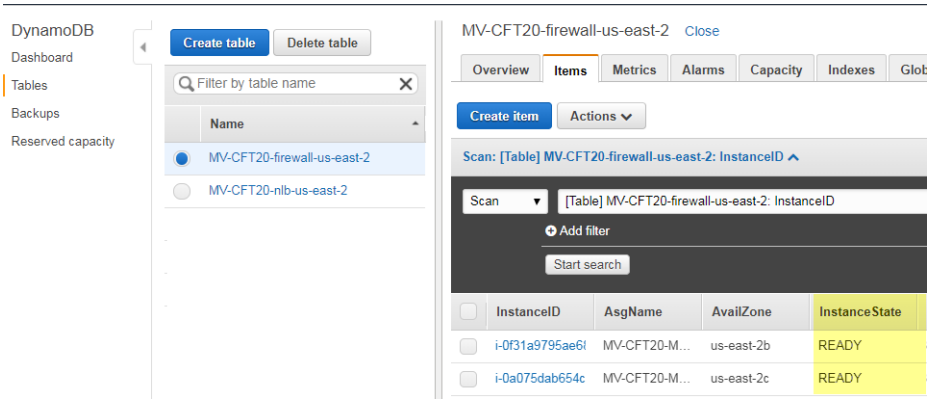


© Palo Alto Networks, Inc.

2. Enter the **VPC Name**in which you want to deploy the application template resources.
3. Optional Change the **NLBSubnetIPBlocks** for the Management subnet for the network load balancer.

STEP 10 | Review the template settings and launch the template.

STEP 11 | Verify that the network load balancer is deployed and in a ready state.




STEP 12 | Get the **DNS name** for the application load balancer, and enter it into a web browser.

For example: http://MVpublic-elb-123456789.us-east-2.elb.amazonaws.com/

When the web page displays, you have successfully launched the auto scaling template.

STEP 13 | Verify that each firewall has a NAT policy rule to the IP address of each network load balancer.

When you deploy the application template to launch another instance of a network load balancer and pair of web servers, the firewall learns about the port allocated for the next network load balancer instance and creates another NAT policy rule. So, if you deploy the application template three times, the firewall has three NAT policy rules for ports 81, 82, and 83.


STEP 14 | If you have launched the application template more than once, you need to Enable Traffic to the ELB Service.

### Enable Traffic to the ELB Service (v2.0 and v2.1)

If you add a second or additional internal load balancers (ILBs) in your deployment, you must complete additional configuration so that the internal load balancer, the VM-Series firewalls auto scaling groups, and the web servers can report as healthy and traffic is load balanced across all your AWS resources.

*In v2.0, the ILB can only be a network load balancer. In v2.1 the ILB can be an application load balancer or a network load balancer.*

STEP 1 | On the AWS management console, verify the ports allocated for each network balancer on the DynamoDB table.

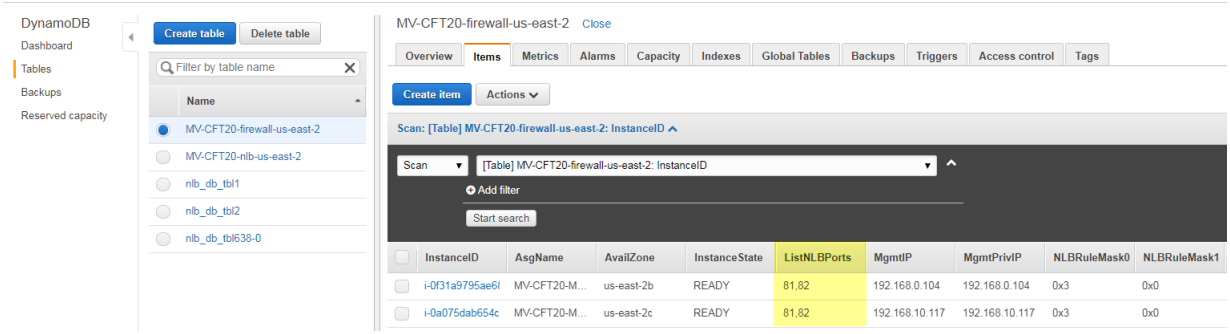


© Palo Alto Networks, Inc.

When you launch a new internal load balancer, the application template must send an SQS message to the SQS URL you provided as input when you launched the template. The lambda function in the firewall template monitors the SQS and adds the port mapping to the DynamoDB table for the firewall template. Starting at port 81, the port allocated for every additional internal load balancer you deploy increments by 1. So, the second internal load balancer uses port 82, and the third port uses port 83.

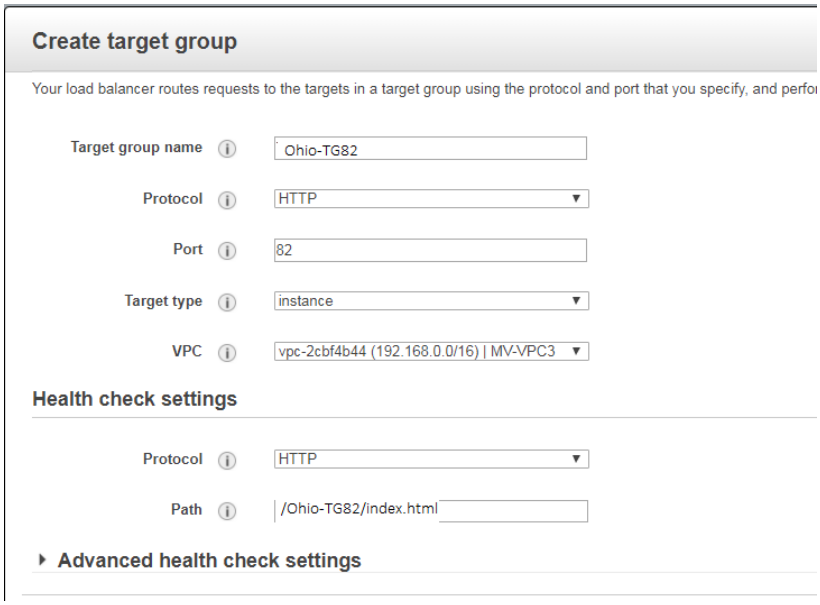
1. Select the **DynamoDB** service on the AWS management console.
2. Select **Tables** and click the table that matches the stack name for your firewall template. For example, MV-CFT20-firewall-us-east-2.

   In the Items list, view the ports used by the internal load balancers that are publishing to the SQS associated with the firewall template.




STEP 2 | Create a target group. The internal load balancer sends requests to registered targets using the port and protocol that you specify for the servers in the target group.

When you add a new target group, use the port information that you verified on the DynamoDB table.

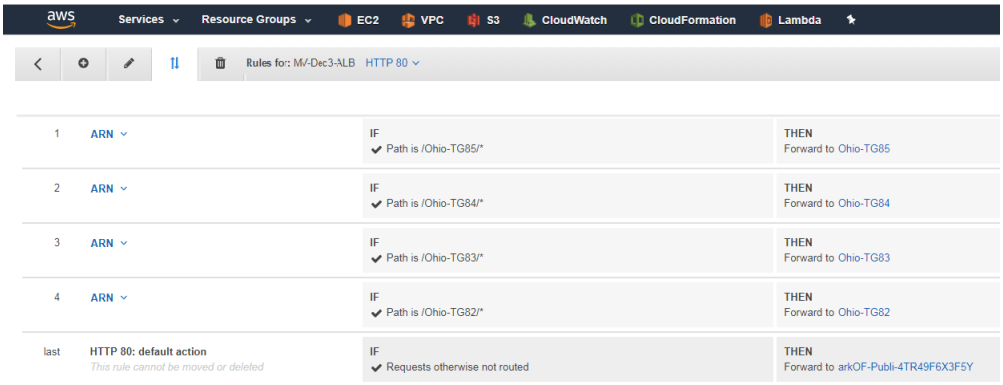



STEP 3 | Edit the listener rules on the internal load balancer to route requests to the target web servers.

1. On the AWS management console, select **Load Balancers** in the Load Balancing section, and select the internal load balancer that matches your stack name.
2. Select **View/edit rules** to modify the rules for the listener.
3. Select **Insert rule** and add a path-based route to forward traffic to the target group you defined above as follows:



© Palo Alto Networks, Inc.




STEP 4 | Attach the target group to both VM-Series firewalls auto scaling groups.

1. Select **Auto Scaling Groups** in the Auto Scaling section and select an auto scaling group that matches the stack name.
2. Select **Details** > **Edit**and select the new target group from the **Target Groups** drop-down.


STEP 5 | Log in to each web server that was deployed by the application template, create a new directory with the target group name and copy the index.html file into the directory. Until you set up the path to the index.html file, the health check for this web server reports as unhealthy.

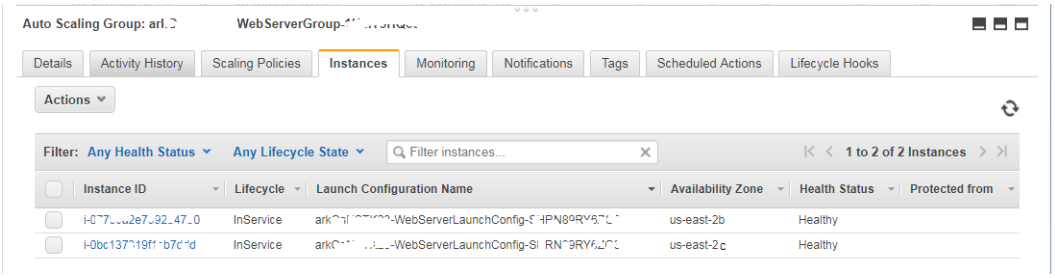
```
sudo su
cd/var/www/html
mkdir <target-groupname>
cp index.html <target-groupname>
```

STEP 6 | Verify the health status of the web servers.

Select **Auto Scaling Groups**, and use the application stack name to find the webserver auto scaling group to verify that the web servers are reporting healthy.



© Palo Alto Networks, Inc.




## Customize the Bootstrap.xml File (v2.0)

The bootstrap.xml file provided in the GitHub repository uses a default username and password for the firewall administrator. Before deploying the VM-Series Auto Scaling template in a production environment, at a minimum, you must create a unique username and password for the administrative account on the VM-Series firewall. Optionally, you can fully configure the firewall with zones, policy rules, security profiles and export a golden configuration snapshot. You can then use this configuration snapshot as the bootstrap.xml file for your production environment.

You have two ways to customize the bootstrap.xml file for use in a production environment:

- **Option 1**: Launch a VM-Series firewall on AWS using the bootstrap files provided in the GitHub repository, modify the firewall configuration and export the configuration to create a new bootstrap.xml file for the VM-Series Auto Scaling template. See Use the GitHub Bootstrap Files as Seed.
- **Option 2**: Launch a new VM-Series firewall on AWS without using the bootstrap files, add a NAT policy rule to ensure that the VM-Series firewall handles traffic properly, and export the configuration to create a new bootstrap.xml file for the VM-Series Auto Scaling template. See Create a new Bootstrap File from Scratch.

*If you have deployed the template and now need to change the credentials for the administrative user or add a new admin user and update the template stack, see* Modify Administrative Account and Update Stack.

### Create a new Bootstrap File from Scratch

Launch a new VM-Series firewall on AWS using the AMI for the PAN-OS version (8.0 or 8.1), without using the sample bootstrap.xml file, and export the configuration to create a new bootstrap.xml file for use with the VM-Series Auto Scaling template v2.0.

STEP 1 | Deploy the VM-Series Firewall on AWS (no bootstrapping required) and use the public IP address to SSH into the Command Line Interface (CLI) of the VM-Series firewall. You will need to configure a new administrative password for the firewall.

STEP 2 | Log in to the firewall web interface.

STEP 3 | (Optional) Configure the firewall. You can configure the dataplane interfaces, zones and policy rules.

STEP 4 | **Commit** the changes on the firewall.

STEP 5 | Export the configuration file and name it as `bootstrap.xml`. (**Device** > **Setup** > **Operation** > **Export Named Configuration Snapshot**).



© Palo Alto Networks, Inc.

STEP 6 | Download the bootstrap.xml file from the GitHub repository, open it with a text editing tool, and copy lines 353 to 356. These lines define the AWS CloudWatch namespace to which the firewall publishes custom PAN-OS metrics that are required for the firewalls to auto scale.

STEP 7 | Edit the configuration file you exported earlier to include the AWS CloudWatch information.
Search for `</management>` and paste the lines 353 to 356 after `</management>`.

```
352          </management>
353          <aws-cloudwatch>
354            <enabled>yes</enabled>
355            <name>autoscale-default-panw-asg-name</name>
356          </aws-cloudwatch>
357        </setting>
```

STEP 8 | Delete the management interface configuration.

1. Search for `</service>` and delete the ip-address, netmask and default gateway that follow.
2. Search for `</type>` and delete the ip-address, netmask, default gateway, and public-key that follow.

```
326          </service>
327          <ip-address>192.168.10.16</ip-address>
328          <netmask>255.255.255.0</netmask>
329          <default-gateway>192.168.10.1</default-gateway>
330          <hostname>PA-VM</hostname>
331        </system>
332        <setting>
333          <config>
334            <rematch>yes</rematch>
335          </config>
336          <management>
337            <hostname-type-in-syslog>FQDN</hostname-type-in-syslog>
338            <initcfg>
339              <type>
340                <dhcp-client>
341                  <send-hostname>yes</send-hostname>
342                  <send-client-id>no</send-client-id>
343                  <accept-dhcp-hostname>no</accept-dhcp-hostname>
344                  <accept-dhcp-domain>no</accept-dhcp-domain>
345                </dhcp-client>
346              </type>
347              <ip-address>192.168.10.16</ip-address>
348              <netmask>255.255.255.0</netmask>
349              <default-gateway>192.168.10.1</default-gateway>
350              <public-key>c3NoLXJzYSBBQUFBQjNOemFDMXljMkVBQUFBREFRQUJBQUFCQVFDQTRCSjJwZFBSZlh0TjF2SDVqWW5GRUdYTVdvTmZlaU1FcCtBSlZRaVU4c2hEMHBmSUtOVTVSeHdGRFd4OVZZckRRRFVrLzQ5VDdkeThXcXorcXZ1em44dlFpamY1RDJVEeHB2WXk3ZWtiam1IallYdTVXUFR4MnZSaXdiMmVzcS91K3FXbm9hSlQ1cXdjU2srbHRxN0prVjlGcC9HSy9jQkRDT0FqOVhmSHMvWi8xQ0VZRk9uZ0UrNTd5L2VwSjFFWWtxZXlLczZuRTBvbWxRajBHSmNhb1FMcUlqZDZmQW5Dcm93dEZ4Y3c3YWVTR</public-key>
351            </initcfg>
352          </management>
```

STEP 9 | Save the file. You can now proceed with Launch the VM-Series Auto Scaling Template for AWS (v2.0).

### Use the GitHub Bootstrap Files as Seed

Launch a VM-Series firewall on AWS from the AWS Marketplace using the bootstrap files provided in the GitHub repository, modify the firewall configuration for your production environment. Then, export the configuration to create a new bootstrap.xml file that you can now use for the VM-Series Auto Scaling template.

STEP 1 | To launch the firewall see Bootstrap the VM-Series Firewall on AWS.

STEP 2 | Add an elastic network interface (ENI) and associate an elastic IP address (EIP) to it, so that you can access the web interface on the VM-Series firewall. See Launch the VM-Series Firewall on AWS for details.

STEP 3 | Use the EIP address to log in to the firewall web interface with admin as the username and password.

STEP 4 | Add a secure password for the admin user account (**Device** > **Local User Database** > **Users**).

STEP 5 | (Optional) Configure the firewall for securing your production environment.



© Palo Alto Networks, Inc.

STEP 6 | **Commit** the changes on the firewall.

STEP 7 | Generate a new API key for the administrator account. Copy this new key to a new file. You will need to enter this API key when you launch the VM-Series Auto Scaling template; the AWS services use the API key to deploy the firewall and to publish metrics for auto scaling.

STEP 8 | Export the configuration file and save it as **`bootstrap.xml`**. (**Device** > **Setup** > **Operation** > **Export Named Configuration Snapshot**).

STEP 9 | Open the bootstrap.xml file with a text editing tool and delete the management interface configuration.

```
        </service>
        <ip-address>192.168.10.16</ip-address>
        <netmask>255.255.255.0</netmask>
        <default-gateway>192.168.10.1</default-gateway>
        <hostname>PA-VM</hostname>
      </system>
      <setting>
        <config>
          <rematch>yes</rematch>
        </config>
        <management>
          <hostname-type-in-syslog>FQDN</hostname-type-in-syslog>
          <initcfg>
            <type>
              <dhcp-client>
                <send-hostname>yes</send-hostname>
                <send-client-id>no</send-client-id>
                <accept-dhcp-hostname>no</accept-dhcp-hostname>
                <accept-dhcp-domain>no</accept-dhcp-domain>
              </dhcp-client>
            </type>
            <ip-address>192.168.10.16</ip-address>
            <netmask>255.255.255.0</netmask>
            <default-gateway>192.168.10.1</default-gateway>
            <public-key>c3NoLXJzYSBBQUFBQjNOemFDMXljMkVBQUFBREFRQUJBQUFCQVFDQTRCSjJwZFBSZlh0TjF2SDVqWW5GRUdYTVdvTmZlaU1FcCtBSlZRaVU4c2hEMHBmSUtOVTVSeHdGRFd4OVZZckRRRFVrLzQ5VDdkeThXcXorcXZ1em44dlFpamY1RDJ4dldEaHFWQ3
VEeHB2WXk3ZWtiam1IallYdTVXUFR4MnZSaXdiMmVzcS91K3FXbm9hSlQ1cXdjU2srbHRxN0prVjlGcC9HSy9jQkRDT0FqOVhmSHMvWi8xQ0VZRk9uZ0UrNTd5L2VwSjFFWWtxZXllczZuRTBvbWxRajBHSmNhb1FMcUlqZDZmQW5Dcm93dEZ4Y3c3YWVTRnhZdDdXU1FY
          </initcfg>
        </management>
```

STEP 10 | (Required if you exported a PAN-OS 8.0 configuration) Ensure that the setting to validate the Palo Alto Networks servers is disabled. Look for **`<server-verification>no</server-verification>`**.

STEP 11 | If the check is **`yes`**, change it to **`no`**.

STEP 12 | Save the file. You can now proceed with Launch the VM-Series Auto Scaling Template for AWS (v2.0).

## *SQS Messaging Between the Application Template and Firewall Template*

So that the VM-Series firewalls deployed using the firewall-v2.0.template can detect and send traffic to the network load balancers to which you want to automatically distribute incoming traffic, the firewall template includes a lambda function that monitors a Simple Queue Service for messages. The message allows the lambda function to learn about a new network load balancer and then automatically create a NAT policy rule on the firewall to send traffic to the IP address of the network load balancer. In order to route traffic properly within the AWS infrastructure, the message must also include basic information on the DNS, VPC ID, and the AZ to which the network load balancer belongs.

If you are building your own application template, you must set up your application template to post two types of messages to the SQS URL that the firewall template in the VM-Series autoscaling template version 2.0 uses to learn about network load balancers to which it must distribute traffic in your environment:

- ADD-NLB message that informs the firewalls when a new network load balancer is available.
- DEL-NLB message that informs the firewalls when a network load balancer has been terminated and is no longer available.



© Palo Alto Networks, Inc.

The following examples of each message type includes sample values. You need to modify these message with values that match your deployment.

**ADD-NLB Message**

```
msg_add_nlb= { 'MSG-TYPE': 'ADD-NLB', 'AVAIL-ZONES': [{'NLB-
IP':'192.168.2.101', 'ZONE-NAME':'us-east-2a', 'SUBNET-ID':
 'subnet-2a566243'}, {'NLB-IP':'192.168.12.101', 'ZONE-NAME':'us-east-2b',
 'SUBNET-ID': 'subnet-2a566243 '}], 'DNS-NAME': 'publicelb1-2119989486.us-
east-2.elb.amazonaws.com', 'VPC-ID': 'vpc-42ba9f2b', 'NLB-NAME':
 'publicelb1' }
```

**DEL-NLB Message**

```
msg_del_nlb= { 'MSG-TYPE': 'DEL-NLB', 'DNS-NAME': 'publicelb1-2119989486.us-
east-2.elb.amazonaws.com', }
```

Refer to the AWS documentation for details on how to send a message to an Amazon SQS Queue, or review the **describe_nlb_dns.py** in the sample application template package to see how the application template constructs the messages.

## *Stack Update with VM-Series Auto Scaling Template for AWS (v2.0)*

A stack update allows you to modify the resources that the VM-Series Auto Scaling template—firewall-v2.0.template—deploys. Instead of deleting your existing deployment and redeploying the solution, use the stack update to modify the following parameters:

- License—Switch from BYOL to PAYG and vice versa or switch from one PAYG bundle to another.
- Other stack resources— Change the launch configuration parameters such as the Amazon Machine Image (AMI) ID, the AWS instance type, key pair for your auto scaling groups. You can also update the API key associated with the administrative user account on the firewall.

*Changing the AMI-ID allows you to deploy new instances of the VM-Series firewalls with a different PAN-OS version.*

When you deploy the VM-Series Auto Scaling template, the auto scaling groups and the launch configuration are automatically created for you. The launch configuration is a template that an auto scaling group uses to launch EC2 instance, and it specifies parameters such as the AMI ID, the instance type, key pair for your auto scaling group. To launch VM-Series firewalls with your updated parameters, you must first update the stack and then delete the existing auto scaling groups in each AZ. To prevent service disruption, delete the auto scaling group in one AZ first, and wait for the new firewall instances to launch with the updated stack parameters. Then, verify that the firewalls have inherited the updates you made before you proceed to complete the changes in the other AZ.

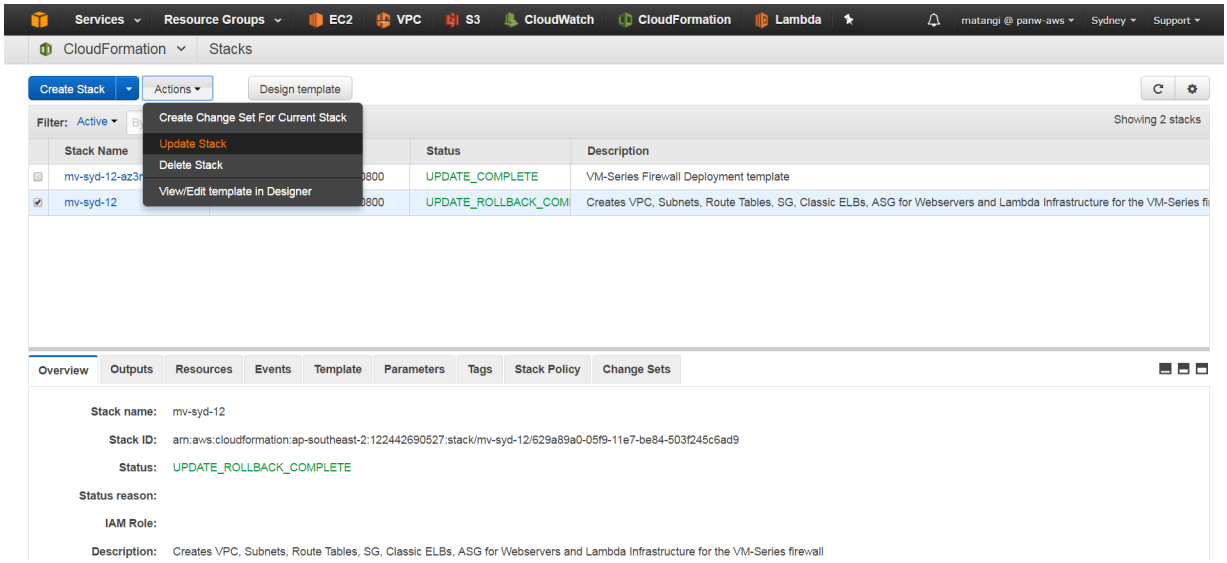
*For critical applications, perform a stack update during a maintenance window.*

You can update stack directly or create change sets. The workflow in this document takes you through the manual stack update.

STEP 1 | In the AWS CloudFormation console, select the parent stack that you want to update and choose **Actions** > **Update Stack**.



© Palo Alto Networks, Inc.




STEP 2 | Modify the resources that you want to update.

- PAN-OS version—To modify the PAN-OS version look up the AMI ID for the version you want to use and enter the ID.
- License option—Switch from BYOL to PAYG or across PAYG bundles 1 and 2.

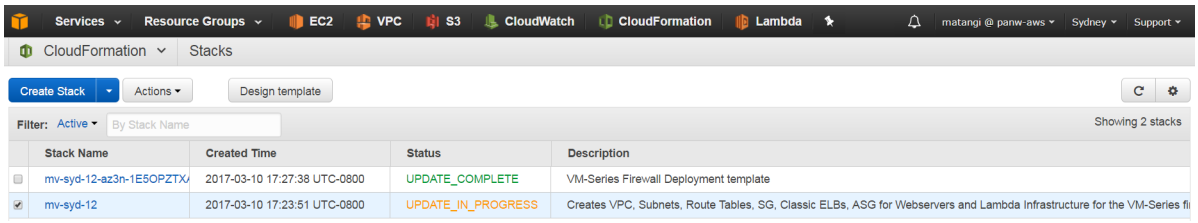
If you're switching to BYOL, make sure to include the auth code in the bootstrap package (See steps 3 and 5).

If you're switching between PAYG bundle version 1 and 2, look up the AMI ID for the VM-Series firewall.

- Other stack resources— You can modify the AMI ID, the instance type, security group, key pair for the stack resources, or the API key associated with the administrative user account on the firewall.

If you create a new administrative user account or modify the credentials of the existing administrator on the firewall, in order to update that stack and deploy new firewalls with the updated API key, you need to follow the workflow in Modify Administrative Account and Update Stack.

STEP 3 | Acknowledge the notifications and review the changes and click **Update** to initiate the stack update.

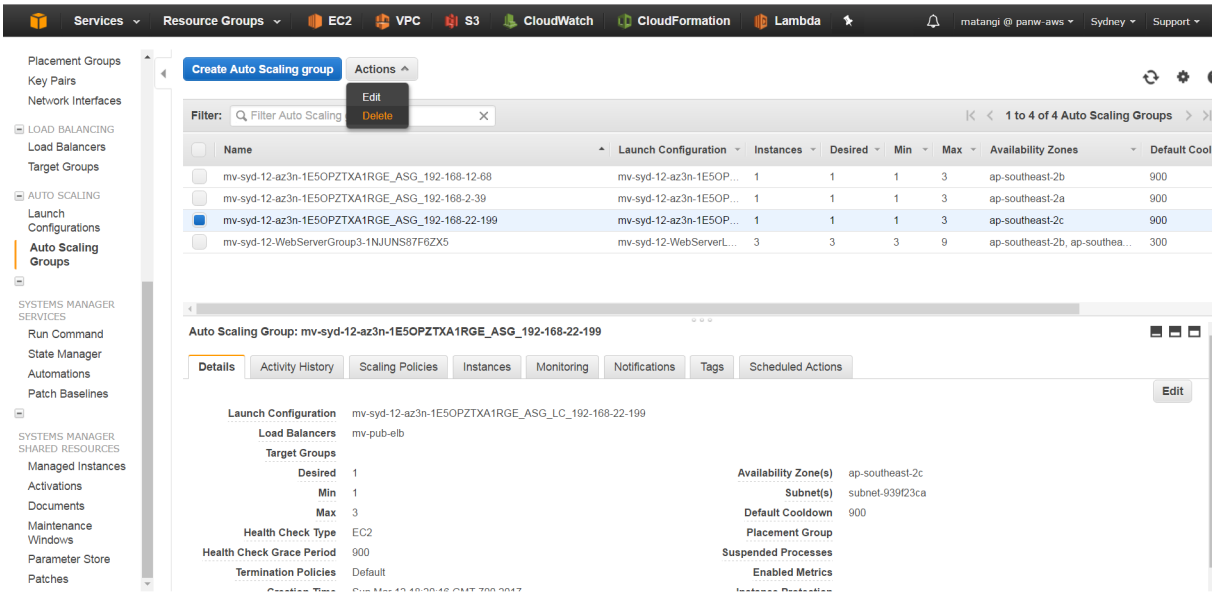



STEP 4 | On the **EC2 dashboard** > **Auto Scaling Groups** and pick an AZ in which to delete the ASG.

Deleting an ASG automatically triggers the process of redeploying a new ASG. The firewalls in the new ASG use the updated stack configuration.



© Palo Alto Networks, Inc.




STEP 5 | Verify that the updated parameters are used to launch the VM-Series firewalls in the new ASG.

Use a phased rollout process, where you test the new ASG thoroughly and ensure that the firealls are properly handling traffic. Then, wait one hour before continuing to the next ASG.

STEP 6 | Repeat steps 4 and 5 to replace the ASG in the other AZ.

## *Modify Administrative Account and Update Stack (v2.0)*

If you have already deployed the template and now want to change the password for the administrative account or create a new administrative user account on the VM-Series firewall, you must generate a new API key and update the template stack with the new API key for the administrative user account. And in order to ensure that new firewall instances are configured with the updated administrative user account, you need to export the firewall configuration and rename it to bootstrap.xml, then upload it to the S3 bootstrap folder that the VM-Series AutoScaling template uses.

STEP 1 | Log in to the web interface of the firewall and change the credentials for an existing administrative user or create a new account.

STEP 2 | Generate the API key.

STEP 3 | Export the current running configuration and rename it to bootstrap.xml.

STEP 4 | Upload this bootstrap.xml file to the S3 bootstrap folder; see Customize the Bootstrap.xml File (v2.0).

STEP 5 | Update the API key in the stack to ensure that newly launched firewalls will have the updated administrator account.

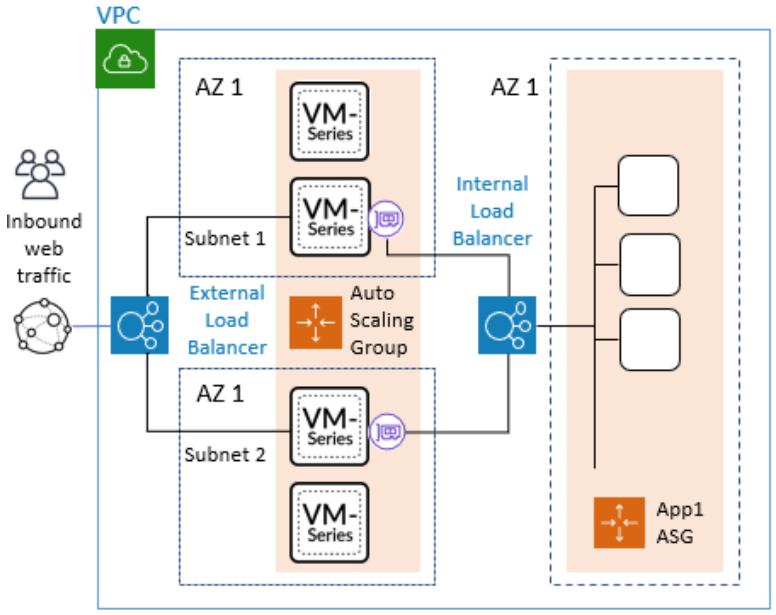
See Stack Update with VM-Series Auto Scaling Template for AWS (v2.0).



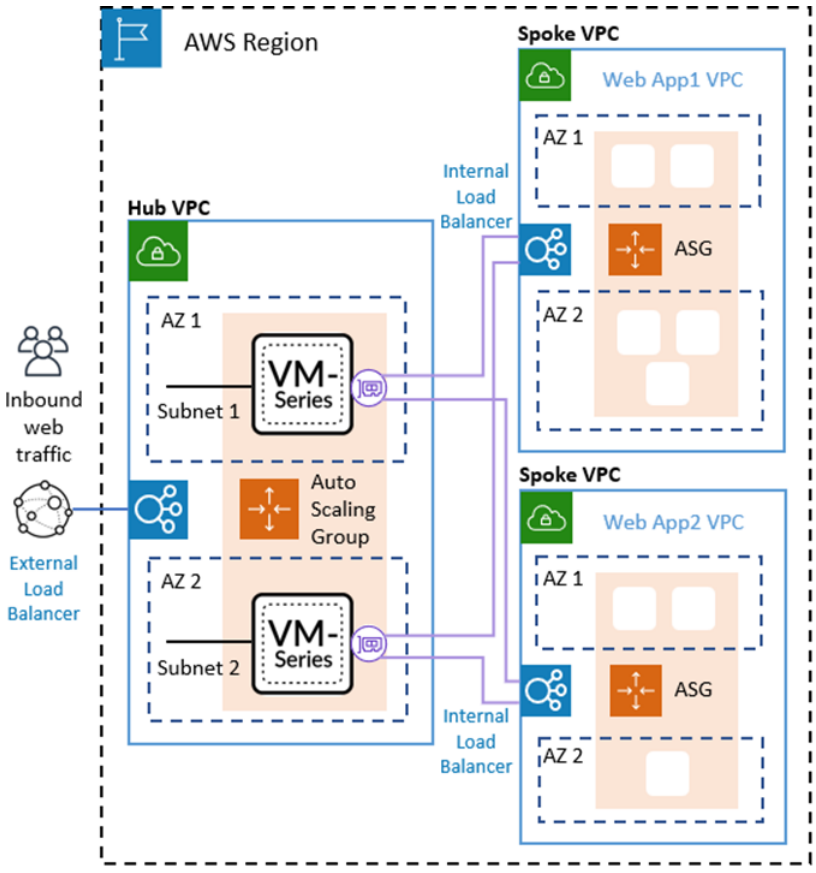
© Palo Alto Networks, Inc.

# VM-Series Auto Scale Templates for AWS Version 2.1

The VM-Series Auto Scaling templates enable you to deploy a single auto scaling group (ASG) of VM-Series firewalls to secure inbound traffic from the internet to your application workloads on AWS. You can deploy the VM-Series firewall ASG and the application workloads within a single VPC as shown below.




You can also deploy the firewall ASG in a centralized VPC and your application workloads in separate VPCs within the same region, forming a hub and spoke architecture, as shown below.






© Palo Alto Networks, Inc.

With the hub and spoke architecture you can streamline the delivery of centralized security and connectivity for AWS deployments with many applications, VPCs, or accounts. This architecture can increase agility. Your network security administrators manage the firewall VPC, and DevOps administrators or application developers can manage the application VPCs.

*Ensure that the application VPCs connected to the firewall VPC, do not have an Internet Gateway (IGW), and use a continuous monitoring and security compliance service such as RedLock.*

You can use a single AWS account or multiple AWS accounts to monitor and secure traffic between VPCs and the internet. Centralizing firewalls in a single VPC can reduce costs for deployments with multiple VPCs and/or multiple accounts.

To provide flexibility with securing your application workloads, version 2.1 allows you to deploy an application load balancer or a network load balancer for both the external load balancer that fronts your VM-Series firewall ASG, and the internal load balancer (ILB) that fronts your application workloads.

When an application load balancer fronts the application workloads, you can connect the firewall VPC to the application VPC using VPC peering. When an NLB fronts the application workloads you can use VPC Peering or an AWS Private Link to connect the firewall and application VPCs, as summarized below:

| Firewall VPC LB(External) | Application VPC LB (Internal) | Connection Method |
|---|---|---|
| ALB | NLB | AWS Private Link |
| NLB | NLB | AWS Private Link |
| NLB | ALB | VPC Peering |
| ALB | ALB | VPC Peering |

If you deploy in a single VPC you can use all the load balancing combinations in the previous table.

You can deploy the templates in both and greenfield (new VPC and applications) and brownfield (existing VPC and applications) use cases.

| Template | New | Existing |
|---|---|---|
| Firewall | firewall-new-vpc-v2.1.template<br>panw-aws-same-vpc-v2.1.template | firewall-existing-vpc-v2.1.template<br>panw-aws-same-vpc-v2.1.template |
| Application | panw-aws-nlb-new-vpc-v2.1.template<br>panw-aws-alb-new-vpc-v2.1.template | panw-aws-alb-existing-vpc-v2.1.template<br>panw-aws-nlb-existing-vpc-v2.1.template |

## *What Components Do the VM-Series Auto Scaling Template for AWS (v2.1) Leverage?*

The VM-Series Auto Scaling template for AWS includes the following building blocks.

- VM-Series Firewall Templates
- Application Templates
- Lambda Functions



© Palo Alto Networks, Inc.

- Panorama
- Bootstrap Files

### VM-Series Firewall Templates

The firewall templates deploy an internet-facing external load balancer and VM-Series firewalls within an auto scaling group that spans a minimum of two Availability Zones (AZs). The external load balancer distributes incoming VPC traffic across the pool of VM-Series firewalls. It can be an application load balancer (ALB) or a network load balancer (NLB). The VM-Series firewalls automatically publish custom PAN-OS metrics that enable auto scaling.

| Template | Description |
|---|---|
| firewall-new-vpc-v2.1.template | Deploys a firewall stack with two to four availability zones in a new VPC. |
| firewall-existing-vpc-v2.1.template | Deploys a firewall stack with two to four availability zones in an existing VPC.<br>To deploy in an existing VPC you must enter:<br>• VPC ID<br>• Internet Gateway ID. This is an existing gateway.<br>• Subnet CIDR lists for the Management, Untrust, Trust, NAT Gateway and Lambda subnets. The template uses the CIDRs to create these subnets.<br>If you choose to create a new ELB, the template connects the firewall ASG to the ELB backend pool. If you use an existing ELB, you must manually connect the firewall ASG to the existing load balancer backend. |

See Customize the Firewall Template Before Launch (v2.0 and v2.1) for more on these parameters.

### Application Templates

The application template deploys an internal load balancer (ILB) and one auto scaling group with a web server in each availability zone (AZ).

| Template | Description |
|---|---|
| panw-aws-same-vpc-v2.1.template | Deploy application in same VPC as the firewall VPC. You can choose a network or application load balancer. |
| panw-aws-alb-new-vpc-v2.1.template | Deploy application in a new VPC, using ALB as the internal load balancer, and using VPC Peering between the firewall VPC and application VPC. Supports both same account and cross-account deployments.<br>You must supply the following parameters:<br>• Hub account ID<br>• Hub VPC ID for VPC peering<br>• Hub VPC trust subnet CIDRs. The template uses these for route table construction after VPC peering is established, one CIDR per availability zone. |



© Palo Alto Networks, Inc.

| Template | Description |
| --- | --- |
| | • StsAssumeRoleARN (output from the Hub template for SQS access) |
| panw-aws-nlb-new-vpc-v2.1.template | Deploy application in a new VPC, using NLB as the internal load balancer, and using NLB Endpoint Services/Interfaces to communicate between the firewall VPC and application VPC.<br>You must supply these parameters.<br>• Hub account ID<br>• StsAssumeRoleARN (output from the Hub template for SQS access) |
| panw-aws-alb-existing-vpc-v2.1.template | Deploy ALB in an existing Application VPC. You must supply the VPC ID for your application, and an existing Subnet ID.<br>This template deploys the load balancer in the application VPC and establishes the lambda resources. You must detach your target workload from any existing load balancer, and connect it to the new load balancer. |
| panw-aws-nlb-existing-vpc-v2.1.template | Deploy NLB in an existing Application VPC. Deploy application in a new VPC, using NLB as the internal load balancer, and using NLB Endpoint Services/ Interfaces to communicate between the firewall VPC and application VPC. |

### Lambda Functions

AWS Lambda provides robust, event-driven automation without the need for complex orchestration software. AWS Lambda monitors a Simple Queue Service (SQS) to learn about load balancers (ALBs or NLBs) that publish to the queue. When the Lambda function detects a new load balancer, it creates a new NAT policy rule and applies it to the VM-Series firewalls within the ASG. The firewalls have a NAT policy rule for each application, and the firewalls use the NAT policy rule (that maps the port to the load balancer IP address) to forward traffic to the load balancer in front of the application web servers.

The Lambda functions also delete all the configuration items that Lambda added to the device group and template stack in Panorama. This includes the NAT rule, Address Object, and Static Routes that were pushed to the VM-Series firewall. The Lambda function handles delicensing as well.

To learn more about the Lambda functions, refer to the **Palo Alto Networks AWS AutoScale Documentation**.

### Panorama

You must have Panorama management server in Panorama mode to configure Auto Scaling v2.1.

The Panorama management server provides centralized monitoring and management of multiple Palo Alto Networks next-generation firewalls from a single location. Panorama allows you to oversee all applications, users, and content traversing your network, and use this knowledge to create application enablement policies that protect and control the network. If you are not familiar with Panorama please see the Panorama Administrator's Guide .



© Palo Alto Networks, Inc.

Managed firewalls are bootstrapped with an `init-config.txt` file. A sample file is included in the GitHub repository so that you can copy the configuration from the template stack and device group when you create them in your existing Panorama.

*The untrust and trust zones created in Panorama must be all lower case.*

In Panorama you must configure your network interfaces using DHCP.

- Only eth1/1 should automatically create default route trust and untrust zones.
- The Security Policy zones are named `untrust` and `trust`.

*All zone names must be lower case*

- The templates configure an Administrator account named `pandemo` and the password `demopassword`.
- Create a virtual router with the naming convention VR-<*TemplateStackName*>. On the virtual router ECMP tab, enable ECMP.
- To set the DNS server address on Panorama, select **Device** > **Setup** > **Services**. Set the **Primary DNS Server** to 169.254.169.253, the **Secondary DNS Server** to 8.8.8.8, and the **FQDN Refresh Time (sec)** to 60. Panorama requires the AWS DNS server IP address to resolve the FQDN of the internal load balancer on AWS. The FQDN refresh time is the interval at which Panorama commits newly detected internal load balancers.

After the application template has launched, Lambda populates the following in Panorama:

- NAT policy
- Address object for LB in Application Template
- Static routes in the virtual router
- Tcp81 service object

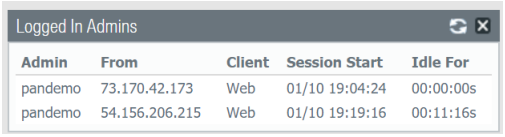
The v2.1 firewall template includes an AWS NAT gateway that the firewalls use to initiate outbound requests for retrieving updates, connecting to Panorama, and publishing metrics to AWS CloudWatch. The NAT Gateways also have Elastic IP addresses attached to them for each zone.

You need the following Panorama resources to work with the Auto Scale templates for AWS.

| | |
|---|---|
| Panorama API Key | You need a Panorama API key to authenticate the API. Lambda uses your API key to autoconfigure template and device group options. To generate the API key, see Get Your API Key. |
| Panorama License Deactivation Key | The template requires a license deactivation API key and the “Verify Update Server Identity” to be enabled to deactivate the license keys from Panorama. The license deactivation key should be obtained from Palo Alto Customer Support Portal as described in Install a License Deactivation API Key. |
| Panorama VM-Auth-Key | You need a vm-auth-key to enable bootstrapped firewalls to connect to Panorama and receive their bootstrap configuration. See Generate the VM Auth Key on Panorama. |
| Panorama Management Interface Access | • **Port 443 (HTTPS)**—Upon initial deployment of the firewall template, leave HTTPS open so Lambda can connect to Panorama. Wait to receive the following confirmation of connection in Panorama: |



© Palo Alto Networks, Inc.




When you secure port 443 you specify an IP range from which you will allow connections, as well as the EIPs assigned to the NAT gateways. To find NAT gateway EIPs in AWS, go to **VPC** > **NAT Gateways**. There are two NAT gateways and the EIPs associated with them. Note the EIP information for the security group for HTTPS.

- **Port 3978**—Port 3978 must be able to receive traffic from any IP address.

### Bootstrap Files

The GitHub auto scaling repository includes an `init-cfg.txt` file so that the VM-Series firewall has the basic configuration to:

- Perform interface swap so the VM-Series firewall `untrust` traffic uses AWS ENI for eth0.
- Communicate to Panorama for device group and template configuration.

The auto scaling GitHub repository has the basic configuration to get started. This auto scaling solution requires swapping the dataplane and management interfaces to enable the load balancer to forward web traffic to the VM-Series firewall auto scaling tier. For details on management interface mapping with the Amazon ELB as shown in Managment Interface Mapping for Use with Amazon ELB.

## Plan to Deploy VM-Series Auto Scaling Templates for AWS (v2.1)

Before starting the deployment, review the following resources.

- See Auto Scale VM-Series Firewalls with the Amazon ELB Service for an overview of template features, and account planning.
- Customize the Firewall Template Before Launch (v2.0 and v2.1). The basic parameters in this topic apply to all template versions.
- How Does the VM-Series Auto Scaling Template for AWS (v2.0 and v2.1) Enable Dynamic Scaling?

  These concepts apply to all template versions.

## Launch the Firewall Template (v2.1)

You can choose to deploy the firewall and application templates in the same VPC or in separate VPCs.

It supports a hub and spoke architecture in which you can deploy the firewall template in one AWS account and use it as a hub to secure applications (spokes) that belong to the same or to different AWS accounts.

This workflow tells you how to deploy the external load balancer and the VM-Series firewalls using the firewall template. The vm-auth-key must be configured on Panorama prior to launching this template.

STEP 1 | Review the checklists in Plan to Deploy VM-Series Auto Scaling Templates for AWS (v2.1) and Plan the VM-Series Auto Scaling Template for AWS (v2.0 and v2.1).

Verify that you have completed the following tasks:

- (For PAYG only) Review and accept the EULA for the PAYG bundle you plan to use.
- (For BYOL only) Obtain the auth code for a bundle that supports the number of firewalls that might be required for your deployment. You must save this auth code in a text file named `authcodes` (no extensions), and put the `authcodes` file in the `/license` folder of the bootstrap package.



© Palo Alto Networks, Inc.

*If you use individual auth codes instead of a bundle, the firewall only retrieves the license key for the first auth code in the file.*

- Download the files required to launch the VM-Series Auto Scaling v2.1 template from the GitHub repository.

STEP 2 | Modify the **`init-cfg.txt`** file and upload it to the **`/config`** folder.

Because you use Panorama to bootstrap the VM-Series firewalls, your `init-cfg.txt` file should be modified as follows. No `bootstrap.xml` file is needed.

```
type=dhcp-client
```

```
ip-address=
```

```
default-gateway=
```

```
netmask=
```

```
ipv6-address=
```

```
ipv6-default-gateway=
```

```
hostname=
```

```
vm-auth-key=
```

```
panorama-server=
```

```
panorama-server-2=
```

```
tplname=AWS-tmplspoke1
```

```
dgname=AWS-dgspoke1
```

```
dns-primary=169.254.169.253
```

```
dns-secondary=8.8.8.8
```

```
op-command-modes=mgmt-interface-swap
```



© Palo Alto Networks, Inc.

```
dhcp-send-hostname=yes
```

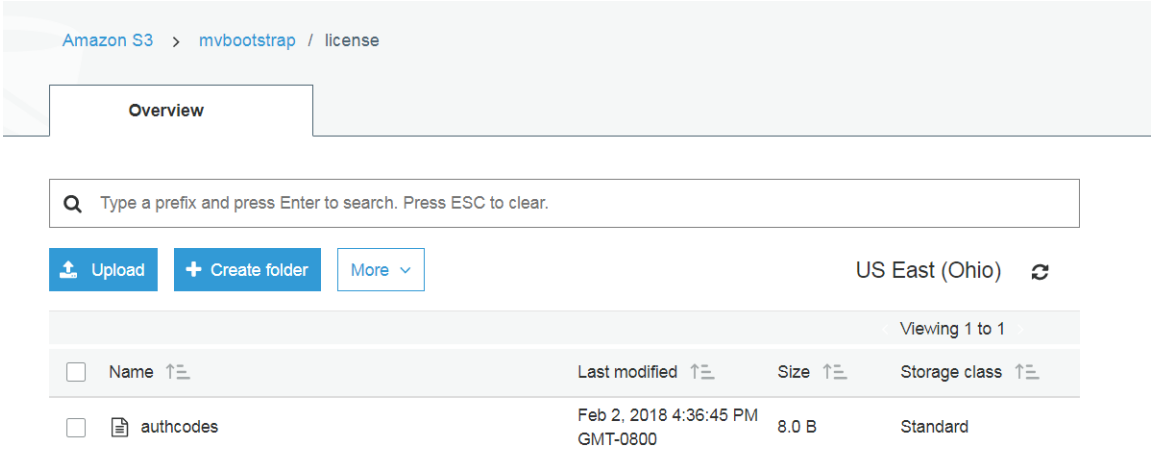
```
dhcp-send-client-id=yes
```

```
dhcp-accept-server-hostname=yesdhcp-accept-server-domain=yes
```

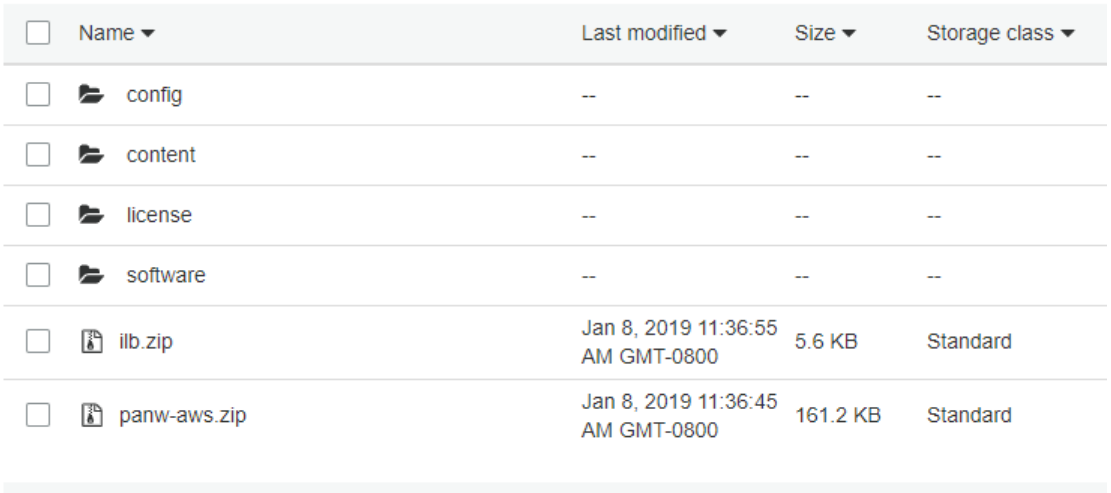
Verify that **`op-command-modes=mgmt-interface-swap`** exists. This is the command for swapping the management interface (mgmt) and the dataplane interface (ethernet 1/1) on the VM-Series firewall on AWS. Use the AWS DNS server IP of 169.254.169.253 for faster load balancer DNS name resolution.

STEP 3 | (For BYOL only) Add the license auth code in the **`/license`** folder of the bootstrap package.

1. Use a text editor to create a new text file named **`authcodes`** (no extension).
2. Add the authcode for your BYOL licenses to this file, and save. The authcode must represent a bundle, and it must support the number of firewalls that might be required for your deployment. If you use individual authcodes instead of a bundle, the firewall only retrieves the license key for the first authcode in the file.




STEP 4 | Upload Lambda code for the firewall template (**`panw-aws-zip`**) and the Application template (**`ilb.zip`**) to an S3 bucket. You can use the same S3 bucket that you use for bootstrapping.






© Palo Alto Networks, Inc.

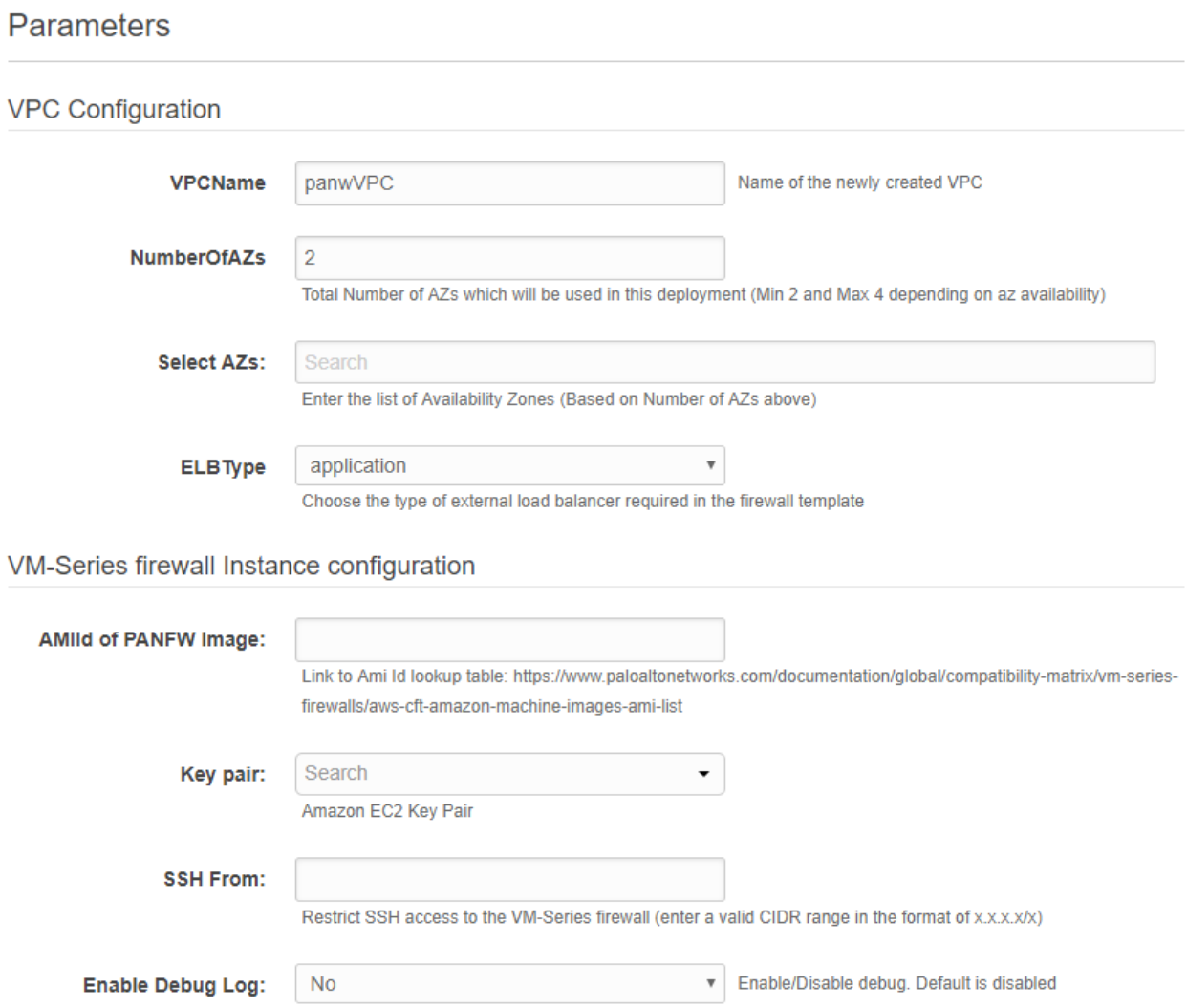
If the Application stack is managed by a different account than the firewall, use the Application account to create another s3 bucket in the same AWS region as the firewall template and copy `ilb.zip` to that s3 bucket.

STEP 5 | Select the firewall template.

1. In the AWS Management Console, select **CloudFormation** > **Create Stack**.
2. Select Upload a template to Amazon S3, to choose the application template to deploy the resources that the template launches within the same VPC as the firewalls, or to a different VPC. Click Open and Next.
3. Specify the Stack name. The stack name allows you to uniquely identify all the resources that are deployed using this template.

STEP 6 | Configure the parameters for the VPC.

1. Be sure to select at least two availability zones




2. Look up the AMI ID for the VM-Series firewall and enter it. Make sure that the AMI ID matches the AWS region, PAN-OS version and the BYOL or PAYG licensing option you opted to use.
3. Select the EC2 **Key pair** (from the drop-down) for launching the firewall. To log in to the firewalls, you must provide the name of this key pair and the private key associated with it.

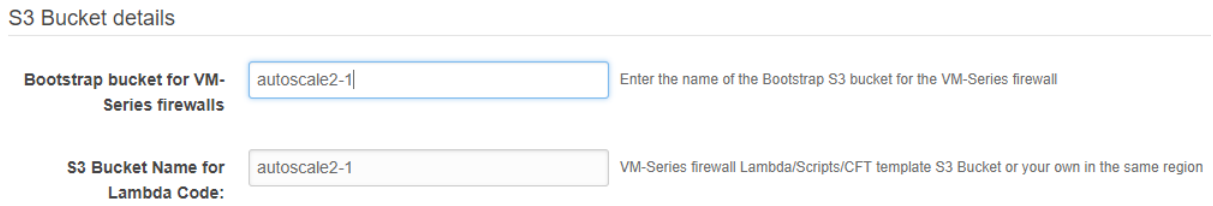


© Palo Alto Networks, Inc.

4. For the **SSH from field**, the firewalls will be managed by Panorama and do NOT have an EIP for the management interface. But just in case you decide to assign an EIP configure the IP range you would connect from.
5. Select **Yes** if you want to **Enable Debug Log**. Enabling the debug log generates more verbose logs that help with troubleshooting issues with the deployment. These logs are generated using the stack name and are saved in AWS CloudWatch.

By default, the template uses CPU utilization as the scaling parameter for the VM-Series firewalls. Custom PAN-OS metrics are automatically published to the CloudWatch namespace that matches the stack name you specified earlier.

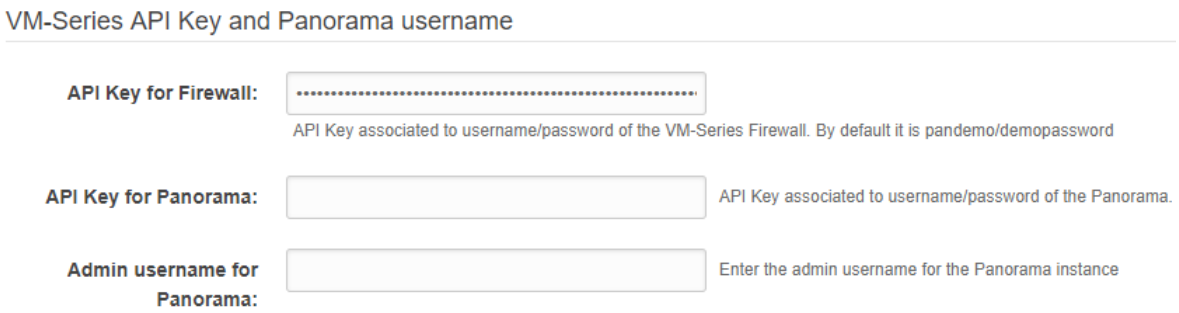
STEP 7 | Specify the name of the Amazon S3 bucket(s).




1. Enter the name of the S3 bucket that contains the bootstrap package.

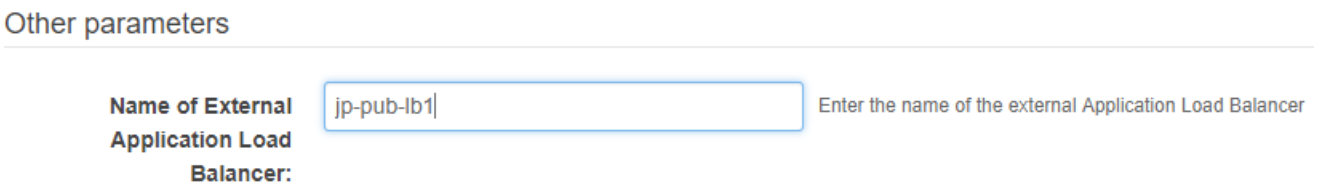
   If the bootstrap bucket is not set up properly or if you enter the bucket name incorrectly, the bootstrap process fails, and you cannot be able to log in to the firewall. Health checks for the load balancers also fail.
2. Enter the name of the S3 bucket that contains the panw-aws.zip file. As mentioned earlier you can use one S3 bucket for the Bootstrap and Lambda code.

STEP 8 | Specify the keys for enabling API access to the firewall and Panorama.




1. Enter the key that the firewall must use to authenticate API calls. The default key is based on the sample file and you should only use it for testing and evaluation. For a production deployment, you must create a separate PAN-OS login just for the API call and generate an associated key.
2. Enter the API Key to allow AWS Lambda to make API calls to Panorama. For a production deployment, you should create a separate login just for the API call and generate an associated key.

STEP 9 | Enter the name for the application load balancer.

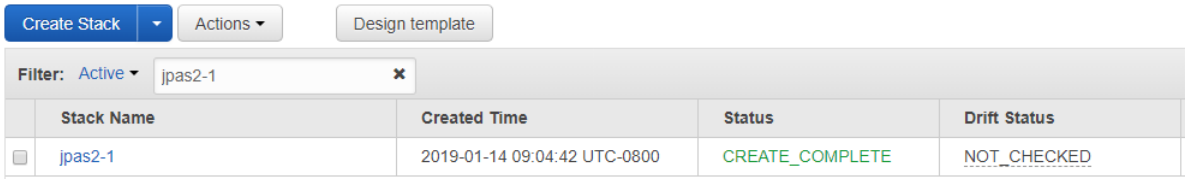




© Palo Alto Networks, Inc.

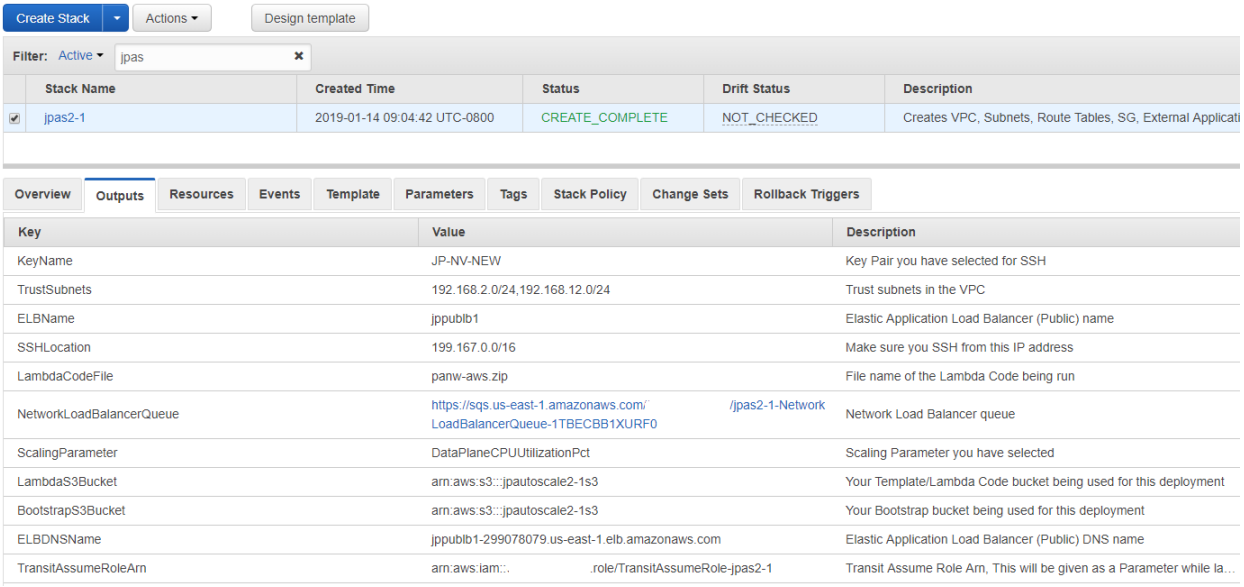
STEP 10 | Review the template settings and launch the template.

1. Select **I acknowledge that this template might cause AWS CloudFormation to create IAM resources**.
2. Click **Create** to launch the template. The CREATE_IN_PROGRESS event displays.
3. On successful deployment the status updates to CREATE_COMPLETE.




STEP 11 | Verify that the template has launched all required resources.

1. On the EC2 Dashboard, select Auto Scaling Groups. Verify that in each AZ, you have one ASG for the VM-Series firewalls. The ASG name prefix includes the stack name.
2. On the AWS Management Console, select the stack name to view the Output for the list of resources.
3. Your output should look similar to the output in the following image.
   - Take note of the Network Load Balancer Queue name.
   - Take note of the Elastic Load Balancer public DNS name.




*It may take up to 20 minutes for the firewalls to boot up and be available to handle traffic.*

*When you are finished with a testing or a production deployment, the only way to ensure charges stop occurring is to completely delete the stack. Shutting down instances, or changing the ASG maximum to 0 is not sufficient.*

STEP 12 | Save the following firewall template information. You must provide these values as inputs when deploying the application template.

- IP addresses of the NAT Gateway in each AZ. You need this IP address to restrict HTTPS access to your Panorama so that Lambda using the EIP's for the NAT Gateway can communicate with Panorama when needed.



© Palo Alto Networks, Inc.

- Network Load Balancer SQS URL. A Lambda function in the firewall stack monitors this queue so that it can learn about any network load balancers that you deploy and create NAT policy rules (one per application) in the Panorama that enable the firewalls to send traffic to the network load balancer IP address.

## *Launch the Application Template (v2.1)*

The application templates allow you to complete the sandwich topology and are provided so that you can evaluate the auto scaling solution. This application template deploys either an application or network load balancer and a pair of web servers behind the auto scaling group of VM-Series firewalls, which you deployed using the firewall template.

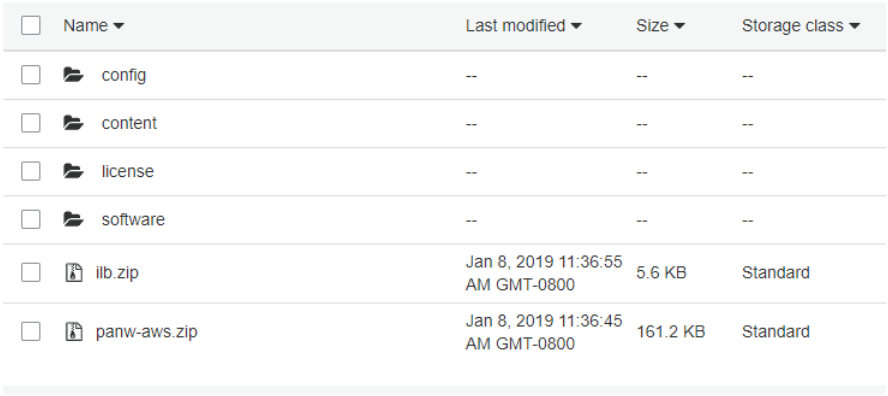
Use this template to evaluate the solution but customize your own template to deploy to production. For a custom template, make sure to enable SQS messaging between the Application template and the Firewall template.

When launching the application template, you must select the template based on whether you want to deploy the application template within the same VPC in which you deployed the firewall template or in a separate VPC. See Enable Traffic to the ELB Service (v2.0 and v2.1).

STEP 1 | Create an S3 bucket from which you will launch the application template.

- If this is a cross-account deployment, create a new bucket.
- If there is one account you can create a new bucket or use the S3 bucket you created earlier (you can use one bucket for everything).

STEP 2 | Upload the ilb.zip file into the S3 bucket.




STEP 3 | Select the application launch template you want you launch.

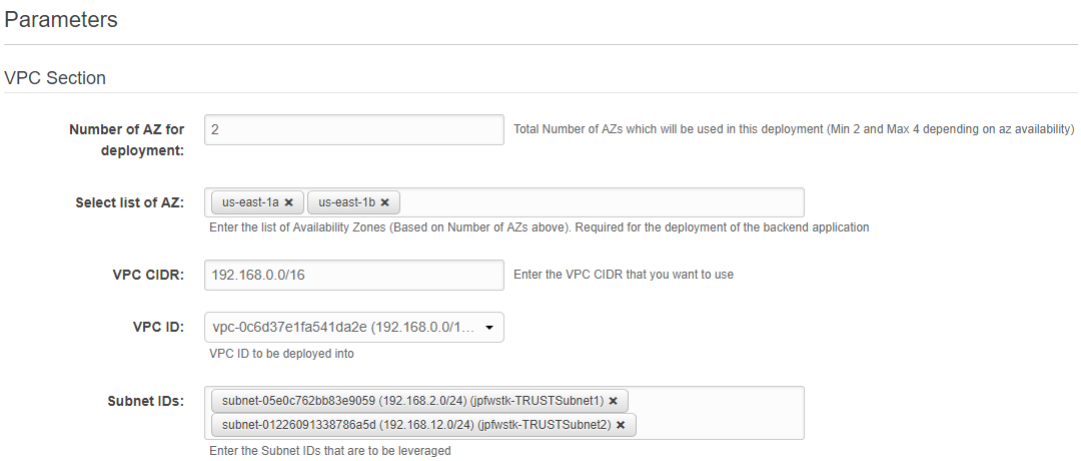
1. In the AWS Management Console, select **CloudFormation** > **CreateStack**
2. Select Upload a template to Amazon S3, to choose the application template to deploy the resources that the template launches within the same VPC as the firewalls, or to a different VPC. Click **Open** and **Next**.
3. Specify the Stack name. The stack name allows you to uniquely identify all the resources that are deployed using this template.

STEP 4 | Configure the parameters for the VPC and network load balancer.

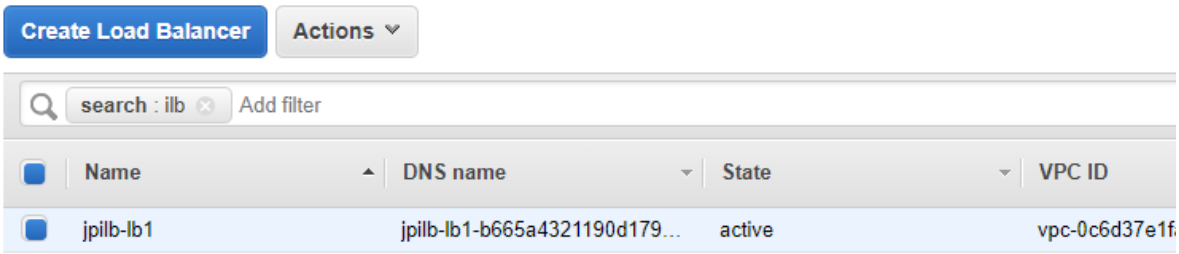
1. Select the two Availability Zones that your setup will span in Select list of AZ. If you are deploying within the same VPC make sure to select the same Availability Zones that you selected for the firewall template.
2. If deploying to a new VPC enter a CIDR Block for the VPC. The default CIDR is 192.168.0.0/16.
3. If deploying to the same VPC you will select the previous VPC and use the Trust subnets.



© Palo Alto Networks, Inc.

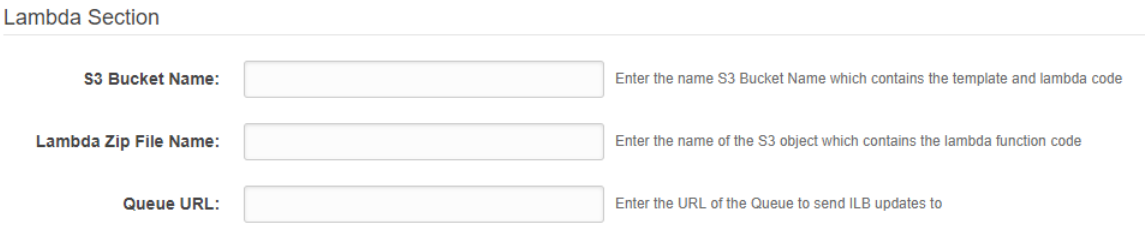



STEP 5 | Select the load balancer type.




STEP 6 | Configure the parameters for Lambda.

1. Enter the S3 bucket name where ilb.zip is stored.
2. Enter the name of the zip file name.
3. Paste the SQS URL that you copied earlier.



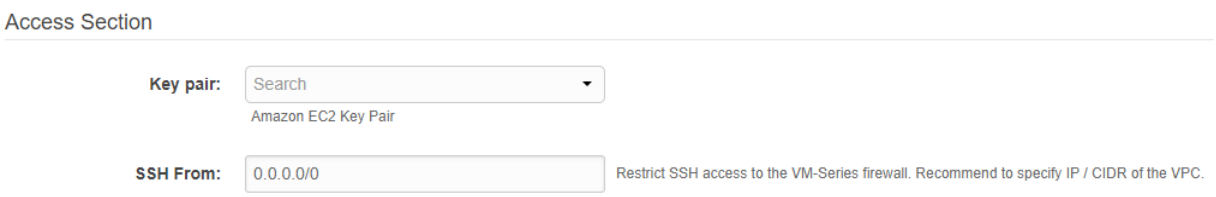

STEP 7 | Modify the web server EC2 instance type to meet your needs.


STEP 8 | Select the EC2 Key pair (from the drop-down) for launching the web servers. To log in to the web servers, you must provide the key pair name and the private key associated with it.

STEP 9 | Select the IP address of the network you will be accessing the servers from for management access only. Web traffic comes through the ELBDNS name you copied when you launched the firewall template.



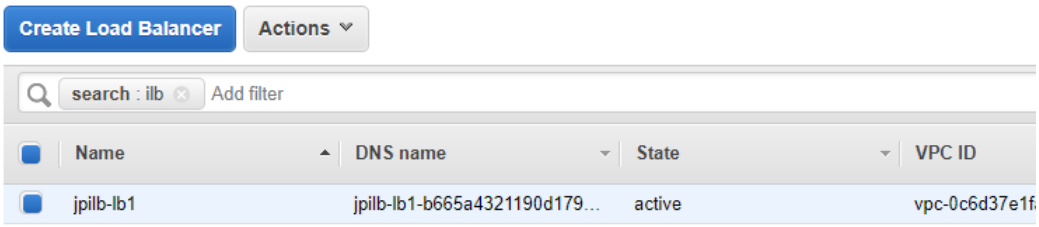
© Palo Alto Networks, Inc.



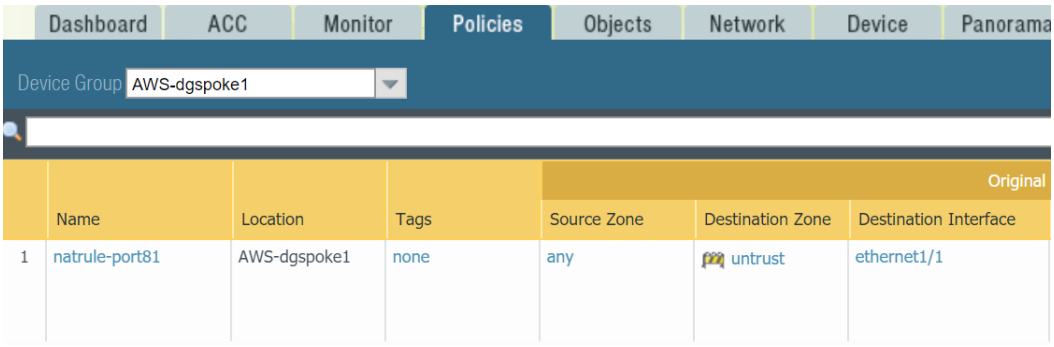

STEP 10 | Review the template settings and launch the template.

STEP 11 | After completion of the application template it can take up to 20 minutes for the web pages to become active.

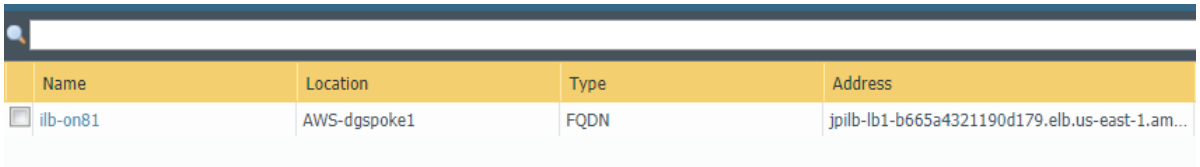
1. Verify that the application template load balancer is marked active.



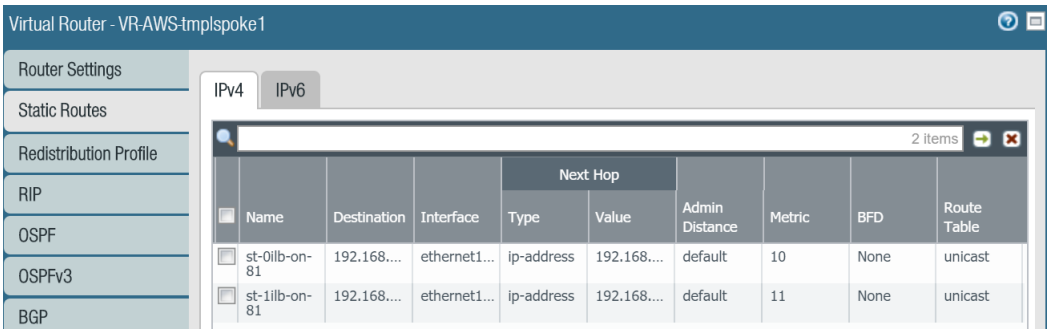

2. Verify that Panorama has a NAT object in the device group.




3. Verify that Panorama has an address object in the device group.




4. Verify that Panorama has static routes in the template stack.






© Palo Alto Networks, Inc.

STEP 12 | Get the DNS name you saved earlier for the application load balancer and enter it into a web browser.

STEP 13 | Upon successful launch your browser should look like this output.

**Congratulations, you have successfully launched VM-Series ASG CloudFormation. This file is coming from Webserver Region: us-east-1**

**StackID: arn:aws:cloudformation:us-east-1: :stack/jpappstk1/1c197cb0-18f4-11e9-b92e-0ab87eb901cc**

**StackName: jpappstk1**

## *Create a Custom Amazon Machine Image (v2.1)*

A custom VM-Series AMI gives you the consistency and flexibility to deploy a VM-Series firewall with the PAN-OS version you want to use on your network instead of being restricted to using only an AMI that is published to the AWS public Marketplace or to the AWS GovCloud Marketplace. Using a custom AMI speeds up the process of deploying a firewall with the PAN-OS version of your choice because it reduces the time to provision the firewall with an AMI published on the AWS public or AWS GovCloud marketplace, and then perform software upgrades to get to the PAN-OS version you want to use on your network. Additionally, you can use the custom AMI in the Auto Scaling VM-Series Firewalls CloudFormation Templates or any other templates that you have created.

You can create a custom AMI with the BYOL, Bundle 1, or Bundle 2 licenses. The process of creating a custom AMI requires you to remove all configuration from the firewall and perform a private data reset, so in this workflow you'll launch a new instance of the firewall from the AWS Marketplace instead of using an existing firewall that you have fully configured.

*When creating a custom AMI with a BYOL version of the firewall, you must first activate the license on the firewall so that you can access and download PAN-OS content and software updates to upgrade your firewall, and then deactivate the license on the firewall before performing the private data reset and creating the custom AMI. If you do not deactivate the license, you lose the license that you applied on this firewall instance.*

STEP 1 | Launch the VM-Series firewall from the Marketplace.

See Launch the VM-Series firewall.

STEP 2 | Configure the administrative password on the firewall.

See Configure a new administrative password on the firewall.

STEP 3 | (Only for BYOL) Activate the license.



© Palo Alto Networks, Inc.

STEP 4 | Install latest content on the firewall.

STEP 5 | (Only for BYOL) Deactivate the license.

STEP 6 | Perform a private data reset.

A private data reset removes all logs and restores the default configuration.

The system disks are not erased, so the content updates from Step 4 are intact.

1. Access the firewall CLI.
2. Export a copy of the configuration.
3. Remove all logs and restore the default configuration.

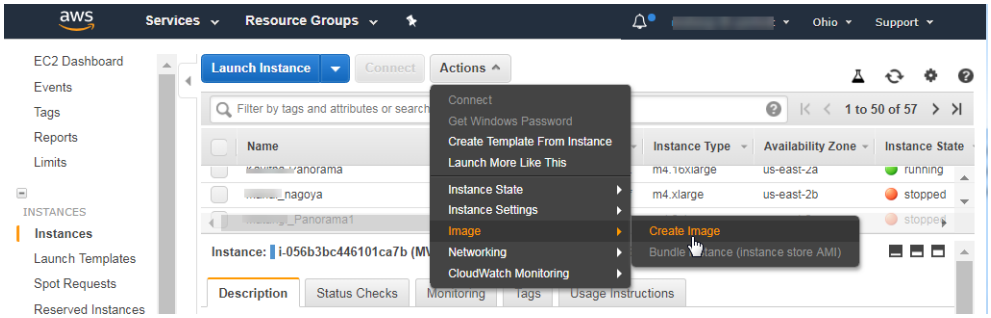
```
request system private-data-reset
```

Enter **y** to confirm.

The firewall reboots to initialize the default configuration.

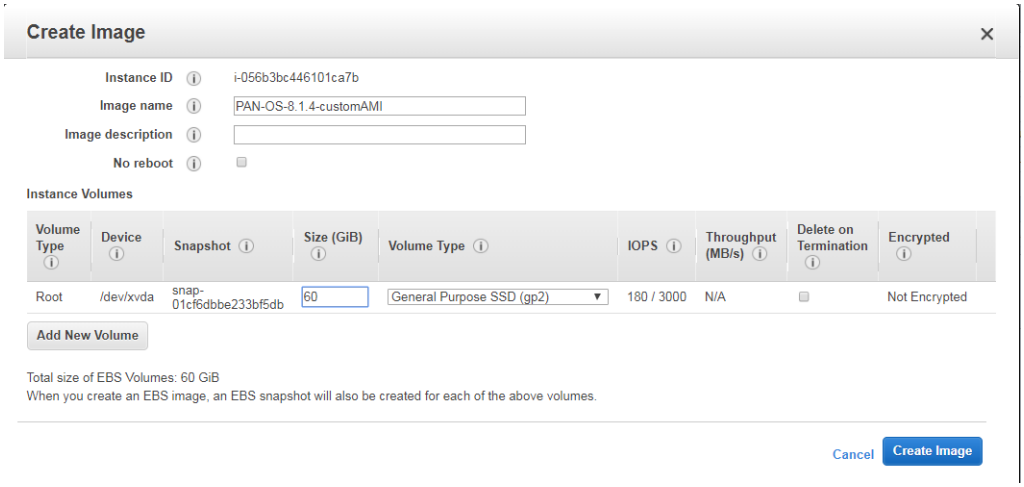
STEP 7 | Create the custom AMI.

1. Log in to the AWS Console and select the EC2 Dashboard.
2. **Stop** the VM-Series firewall.
3. Select the VM-Series firewall instance, and click **Image** > **Create Image**.




4. Enter a custom image name, and click **Create Image**.

   The disk space of 60GB is the minimum requirement.

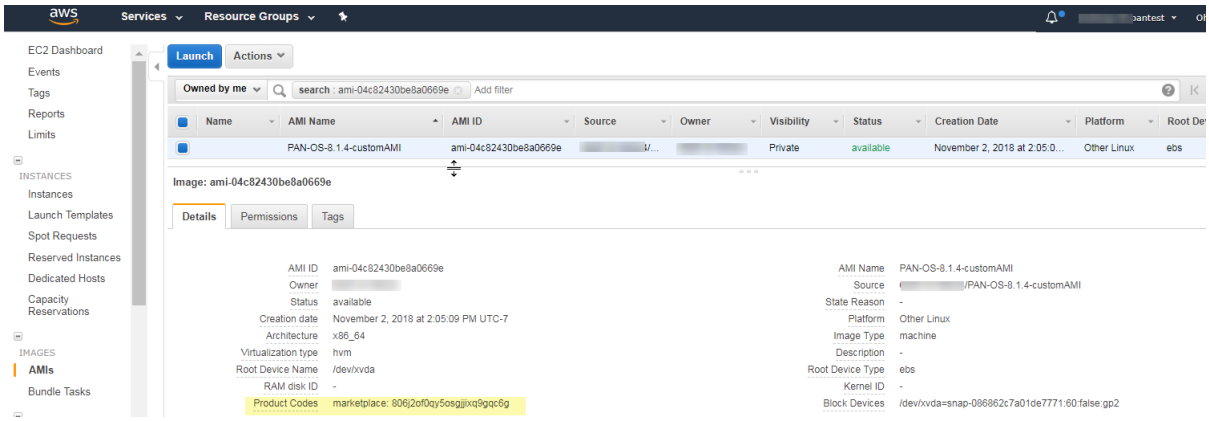



5. Verify that the custom AMI is created and has the correct product code.
   1. On the EC2 Dashboard, select **AMI**.



© Palo Alto Networks, Inc.

2. Select the AMI that you just created. Depending on whether you selected an AMI with the BYOL, Bundle 1, or Bundle 2 licensing options, you should see one of the following **Product Codes** in the details:
   - BYOL—6njl1pau431dv1qxipg63mvah
   - Bundle 1—6kxdw3bbmdeda3o6i1ggqt4km
   - Bundle 2—806j2of0qy5osgjjixq9gqc6g




STEP 8 | Encrypt EBS Volume for the VM-Series Firewall on AWS.

If you plan to use the custom AMI with EBS encryption for an Auto Scale VM-Series Firewalls with the Amazon ELB Service deployment, you must use the default master key for your AWS account.

## VM-Series Auto Scaling Template Cleanup (v2.1)

If you deployed the templates as a test, delete them to save resources and lower costs.

STEP 1 | In the AWS Management Console, select **Cloud Formation** > **Create Stack**.

STEP 2 | Locate the firewall template and application template you launched previously and delete both templates.

For more information on deleting template stacks see, "What is AWS CloudFormation?"

*Failure to delete your template stack incurs charges from AWS.*

## SQS Messaging Between the Application Template and Firewall Template (v2.1)

VM-Series firewalls deployed using one of the firewall templates can detect and send traffic to the load balancers to which you want to automatically distribute incoming traffic. To accomplish this, the firewall template includes a lambda function that monitors a Simple Queue Service for messages. The message allows the lambda function to learn about a new load balancer and then automatically create a NAT policy rule on the firewall to send traffic to the load balancer's IP. To route traffic properly within the AWS infrastructure, the message must also include basic information on the DNS, VPC ID, and the AZ to which the load balancer belongs.

If you are building your own application template, you must set up your application template to post ADD and DEL messages to the SQS URL that the firewall template uses to learn about load balancers to which it must distribute traffic in your environment:



© Palo Alto Networks, Inc.

- ADD-NLB message that informs the firewalls when a new network load balancer is available.
- DEL-NLB message that informs the firewalls when a network load balancer has been terminated and is no longer available.
- ADD-ALB message that informs the firewalls when a new application load balancer is available.
- DEL-ALB message that informs the firewalls when a application load balancer has been terminated and is no longer available.

The following examples of each message type include sample values. You must modify these messages with values that match your deployment.

**ADD-NLB Message**

```
msg_add_nlb= {

        "MSG-TYPE": "ADD-NLB",

        "AVAIL-ZONES": [

        {

            "NLB-IP":"192.168.2.101",

            "ZONE-NAME":"us-east-2a",

            "SUBNET-ID": "subnet-2a566243"

        },

        {

            "NLB-IP":"192.168.12.101",

            "ZONE-NAME":"us-east-2b",

            "SUBNET-ID": "subnet-2a566243 "

        }

        ],

             "DNS-NAME": "publicelb1-2119989486.us-east-2.elb.amazonaws.com",
```



© Palo Alto Networks, Inc.

```
            "VPC-ID": "vpc-42ba9f2b",

            "NLB-NAME": "publicelb1"

}
```

**DEL-NLB Message**

```
msg_del_nlb= {

 "MSG-TYPE": "DEL-NLB",

 "DNS-NAME": "publicelb1-2119989486.us-east-2.elb.amazonaws.com",

}
```

**ADD-ALB**

```
{   "AVAIL-ZONES": [

        {

            "SUBNET-CIDR": "172.32.0.0/24",

            "SUBNET-ID": "subnet-0953a3a8e2a8208a9",

            "ZONE-NAME": "us-east-2a"

        },

        {

            "SUBNET-CIDR": "172.32.2.0/24",

            "SUBNET-ID": "subnet-0a9602e4fb0d88baa",

            "ZONE-NAME": "us-east-2c"

        },
```



© Palo Alto Networks, Inc.

```
        {
            "SUBNET-CIDR": "172.32.1.0/24",
            "SUBNET-ID": "subnet-0b31ed16f308b3c4d",
            "ZONE-NAME": "us-east-2b"
        }
    ],
    "VPC-PEERCONN-ID": "pcx-0538bb05dbe2e1b8e",
    "VPC-CIDR": "172.32.0.0/16",
    "ALB-NAME": "appILB-908-0",
    "ALB-ARN":"arn:aws:elasticloadbalancing:us-
east-2:018147215560:loadbalancer/app/appILB-908-0/1997ed20eeb5bcef",
    "VPC-ID": "vpc-0d9234597da6d9147",
    "MSG-TYPE": "ADD-ALB",
    "DNS-NAME": "internal-appILB-908-0-484644265.us-east-2.elb.amazonaws.com"
}
```

**DEL-ALB Message**

```
{
    "MSG-TYPE": "DEL-ALB",
    "DNS-NAME": "internal-appILB-908-0-484644265.us-east-2.elb.amazonaws.com"
}
```

Refer to the AWS documentation for details on how to send a message to an Amazon SQS Queue.



© Palo Alto Networks, Inc.

## Stack Update with VM-Series Auto Scaling Template for AWS (v2.1)

A stack update allows you to modify the resources that the VM-Series Auto Scaling template firewall template deploys. Instead of deleting your existing deployment and redeploying the solution, use the stack update to modify launch configuration parameters.

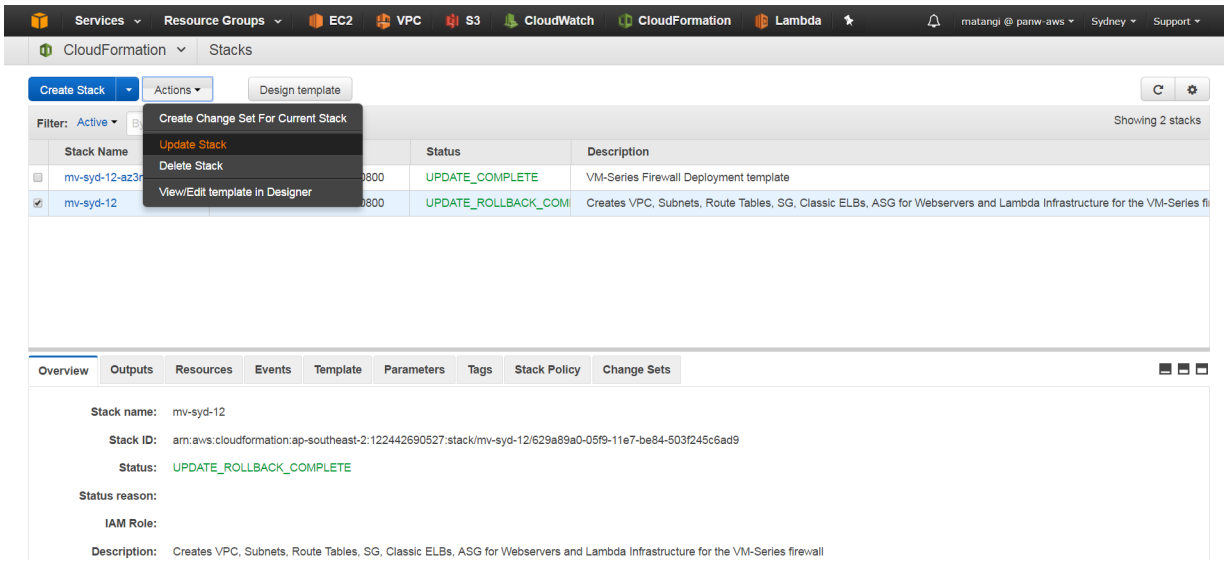
You can modify the AWS instance type, the key pair for your auto scaling groups, and the APi key associated with the adminstrative user account on the firewall.

When you deploy the VM-Series Auto Scaling template, the auto scaling groups and the launch configuration are automatically created for you. The launch configuration is a template that an auto scaling group uses to launch EC2 instance, and it specifies parameters such as the instance type, the key pair for your auto scaling group, or the API key associated with the administrative user account on the firewall.

*For critical applications, perform a stack update during a maintenance window.*

You can update your stack directly or create change sets. The workflow in this document takes you through the manual stack update.

STEP 1 | In the AWS CloudFormation console, select the parent stack that you want to update and choose **Actions** > **Update Stack**.




STEP 2 | Modify the resources that you want to update.

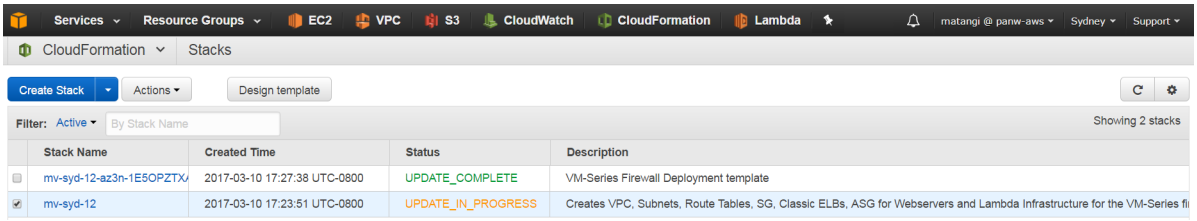
You can modify the instance type, security group, key pair for the stack resources, or the API key associated with the administrative user account on the firewall.

If you create a new administrative user account or modify the credentials of the existing administrator on the firewall, in order to update that stack and deploy new firewalls with the updated API key, you need to follow the workflow in Change a Firewall License Bundle or Version in an AWS Auto Scale Deployment (v2.1).

STEP 3 | Acknowledge the notifications and review the changes and click **Update** to initiate the stack update.



© Palo Alto Networks, Inc.




## *Change a Firewall License Bundle or Version in an AWS Auto Scale Deployment (v2.1)*

You can switch a firewall license from BYOL to PAYG and vice versa, switch from one PAYG bundle to another, or change the PAN-OS version on your firewall.

Because the VM-Series Auto Scaling Template for AWS version 2.1 supports multiple Availability Zones, the templates create a single Auto Scaling Group (ASG). The template includes a Launch Template to update and version the templates, and create and update the ASG. These changes require you to know the Amazon Machine Image (AMI) ID for the target resource. See Look Up the AMI ID.

STEP 1 | Navigate to **Launch Templates** > **Create Launch Template** > **Create a New Version of the Launch Template**.

1. For the Cloud Formation source template, select the template created when the templates were deployed.
2. Specify an AMI ID corresponding to a different PAN-OS version. The storage, instance type, and advanced details should be the same as in the source template.

STEP 2 | To make the new launch template the default version, go to **Launch Template** > **Actions** > **Set Default Version**.

STEP 3 | Manually delete an EC2 instance and wait until a new one boots up. The new instance uses the newly specified AMI ID in your launch template version.

*You must wait until the instance boots up before deleting another. To avoid disrupting traffic, there must be at least one instance running.*

## *Modify Administrative Account (v2.1)*

If you have already deployed the template and now want to change the password for the administrative account or create a new administrative user account on the VM-Series firewall, you must generate a new API key and update the template stack with the new API key for the administrative user account.

STEP 1 | Log in to the web interface of the firewall and change the credentials for an existing administrative user or create a new account.

STEP 2 | Generate the API key.

STEP 3 | Update the API key in the stack to ensure that newly launched firewalls have the updated administrator account.

See Stack Update with VM-Series Auto Scaling Template for AWS (v2.0).



© Palo Alto Networks, Inc.

# List of Attributes Monitored on the AWS VPC

As you provision or modify virtual machines in your AWS VPCs, you have two ways of monitoring these instances and retrieving the tags for use as match criteria in dynamic address groups.

- **VM Information Source**—On a next-gen firewall, you can monitor up to a total of 32 tags—14 pre-defined and 18 user-defined key-value pairs (tags).
- **AWS Plugin on Panorama**—The Panorama plugin for Microsoft AWS allows you to connect Panorama to your AWS VPC on the public cloud and retrieve the IP address-to-tag mapping for your virtual machines. Panorama then registers the VM information to the managed Palo Alto Networks® firewall(s) that you have configured for notification. With the plugin, Panorama can retrieve a total of 32 tags for each virtual machine, 11 predefined tags and up to 21 user-defined tags.

*The maximum length of a tag can be 127 characters. If a tag is longer than 127 characters, Panorama does not retrieve the tag and register it on the firewalls.*

| Attributes Monitored on the AWS-VPC | | VM Information Source on the Firewall | AWS Plugin on Panorama |
|---|---|---|---|
| Architecture | Architecture.<Architecture string> | Yes | No |
| Guest OS | GuestOS.<guest OS name> | Yes | No |
| AMI ID | ImageId.<ImageId string> | Yes | Yes |
| IAM Instance Profile | Iam-instance-profile.<instanceProfileArn> | No | Yes |
| Instance ID | InstanceId.<InstanceId string> | Yes | No |
| Instance State | InstanceState.<instance state> | Yes | No |
| Instance Type | InstanceType.<instance type> | Yes | No |
| Key Name | KeyName.<KeyName string> | Yes | Yes |
| Owner ID<br>The value for this attribute is fetched from the ENI. | Account-number.<OwnerId> | No | Yes |
| Placement—Tenancy, | Placement.Tenancy.<string> | Yes | Yes |



© Palo Alto Networks, Inc.

| Attributes Monitored on the AWS-VPC | | VM Information Source on the Firewall | AWS Plugin on Panorama |
|---|---|---|---|
| Group Name, Availability Zone | Placement.GroupName.<string><br>Placement.AvailabilityZone.<string> | | |
| Private DNS Name | PrivateDnsName.<Private DNS Name> | Yes | No |
| Public DNS Name | PublicDnsName.<Public DNS Name> | Yes | Yes |
| Subnet ID | SubnetID.<subnetID string> | Yes | Yes |
| Security Group ID | Sg-id.<sg-xxxx> | No | Yes |
| Security Group Name | Sg-name.<SecurityGroupName> | No | Yes |
| VPC ID | VpcId.<VpcId string> | Yes | Yes |
| Tag (key, value) | aws-tag.<key>.<value> | Yes;<br>Up to a maximum of 18 user defined tags are supported. The user-defined tags are sorted alphabetically, and the first 18 tags are available for use on the firewalls. | Yes;<br>Up to a maximum of 21 user defined tags are supported. The user-defined tags are sorted alphabetically, and the first 21 tags are available for use on Panorama and the firewalls. |

## IAM Permissions Required for Monitoring the AWS VPC

In order to enable VM Monitoring the user's AWS login credentials tied to the AWS Access Key and Secret Access Key must have permissions for the attributes listed above. These privileges allow the firewall to initiate API calls for monitoring the virtual machines in the AWS VPC.

The IAM policy associated with the user must either have global read-only access such as AmazonEC2ReadOnlyAccess, or must include individual permissions for all of the monitored attributes. The following IAM policy example lists the permissions for initiating the API actions for monitoring the resources in the AWS VPC:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeAvailabilityZones",
                "ec2:DescribeImages",
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceStatus",
```



© Palo Alto Networks, Inc.

```
                "ec2:DescribeKeyPairs",
                "ec2:DescribePlacementGroups",
                "ec2:DescribeRegions",
                "ec2:DescribeSubnets",
                "ec2:DescribeTags",
                "ec2:DescribeVpcs"
            ],
            "Resource": [
                "*"
            ]
}
    ]
}
```



© Palo Alto Networks, Inc.