Set Up the VM-Series Firewall on Google Cloud Platform

docs.paloaltonetworks.com

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

Last Revised

March 26, 2020

Table of Contents

Set Up the VM-Series Firewall on Google Cloud Platform

You can deploy a VM-Series firewall on a Google Compute Engine instance on the Google Cloud Platform.

About the VM-Series Firewall on Google Cloud Platform

VM-Series firewalls bring next-generation firewall features to the Google® Cloud Platform (GCP™).

To maximize performance, VM-Series firewalls on GCP support the Data Plane Development Kit (DPDK) libraries, which provide fast packet processing and improve network performance based on specific combinations of VM-Series firewall licenses and Google Cloud Platform virtual machine (VM) sizes.

- Google Cloud Platform and the VM-Series Firewall
- Minimum System Requirements for the VM-Series Firewall on GCP

Google Cloud Platform and the VM-Series Firewall

The VM-Series firewall integration with GCP allows you to deploy the VM-Series firewall as a virtual machine (VM) running on a Google Compute Engine (GCE) instance. This process is simplified when you Deploy the VM-Series Firewall from Google Cloud Platform Marketplace to deploy a single firewall, or use Panorama Plugin for GCP wiht templates or scripts to deploy multiple firewalls.

After you deploy the VM-Series firewall, you can manually configure the following optional services:

- Enable Google Stackdriver Monitoring on the VM Series Firewall—From the firewall, push PAN-OS metrics to the Google Stackdriver service.
- Enable VM Monitoring to Track VM Changes on Google Cloud Platform—Set up a VM information source that monitors the specific GCP zone containing your instances. The monitored VM metadata can include predefined GCP properties (such as the project ID) and user-defined properties (such as labels and network tags).

Minimum System Requirements for the VM-Series Firewall on GCP

You must choose a VM-Series Firewall License for Public Clouds and a license method: bring-your-own-license (BYOL) or pay-as-you-go (PAYG). To deploy a VM-Series firewall on a Google Compute Engine instance, you must choose a machine type that supports the VM-Series System Requirementsfor your license.

Refer to the table below for the minimum recommended predefined standard machine types for each license. You can choose a higher performing machine type or you can create your own custom machine type if the resource requirements are compatible with your VM-Series firewall license.

A single Google Compute Engine instance supports up to eight network interfaces. If you want to configure eight interfaces, choose n1-standard-8 or a larger machine type.

Capacity	BYOL	Bundles 1 and 2		
		PAYG	Marketplace	Recommended Predefined Machine Type
VM-100 Firewall	✓			n1-standard-4
VM-200 Firewall	✓			

Capacity	BYOL	Bundles 1 and 2		
		PAYG	Marketplace	Recommended Predefined Machine Type
VM-300 Firewall	✓	✓	✓	
VM-1000-HV Firewall	✓			
VM-500 Firewall	✓			n1-standard-8
VM-700 Firewall	✓			n1-standard-16

Supported Deployments on Google Cloud Platform

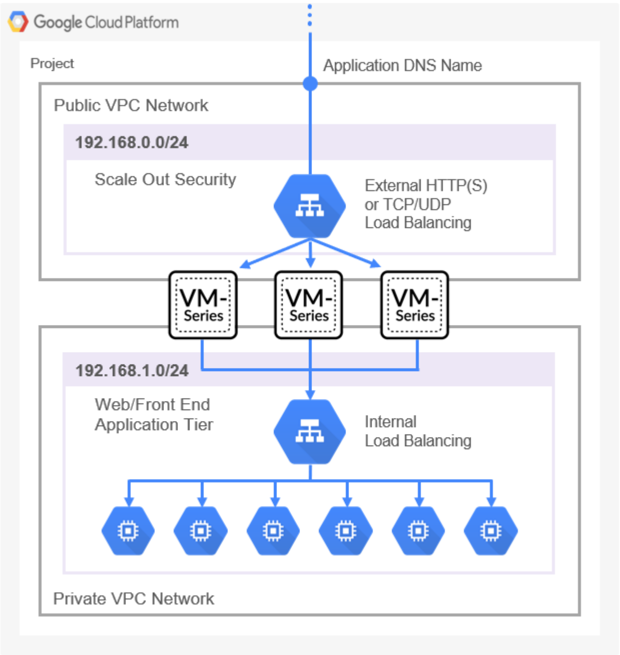
You can deploy the VM-Series firewall on a Google® Compute Engine instance in a network in your virtual private cloud (VPC). The deployment types are:

- Internet Gateway
- Segmentation Gateway
- Hybrid IPSec VPN

Internet Gateway

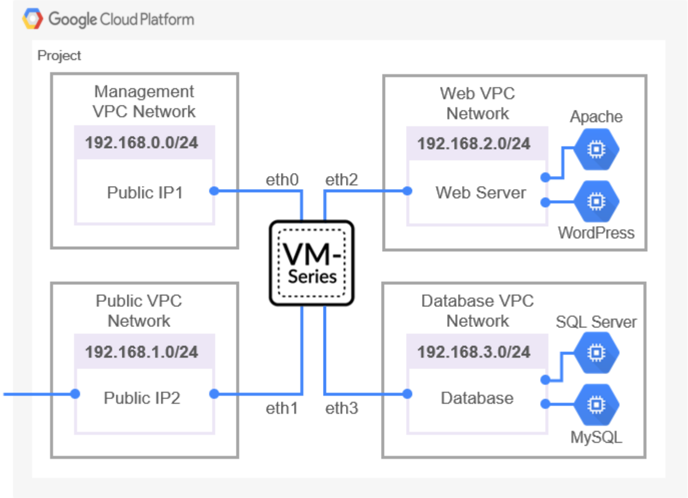
The VM-Series firewall secures North/South traffic to and from the internet to protect applications from known and unknown threats. A Google project can have up to five VPC networks. For a typical example of an internet gateway, refer to the Google configuration examples.

In public cloud environments, it is a common practice to use a scale-out architecture (see the figure below) rather than larger, higher performing VMs. This architecture (sometimes called a *sandwich* deployment) avoids a single point of failure and enables you to add or remove firewalls as needed.



Segmentation Gateway

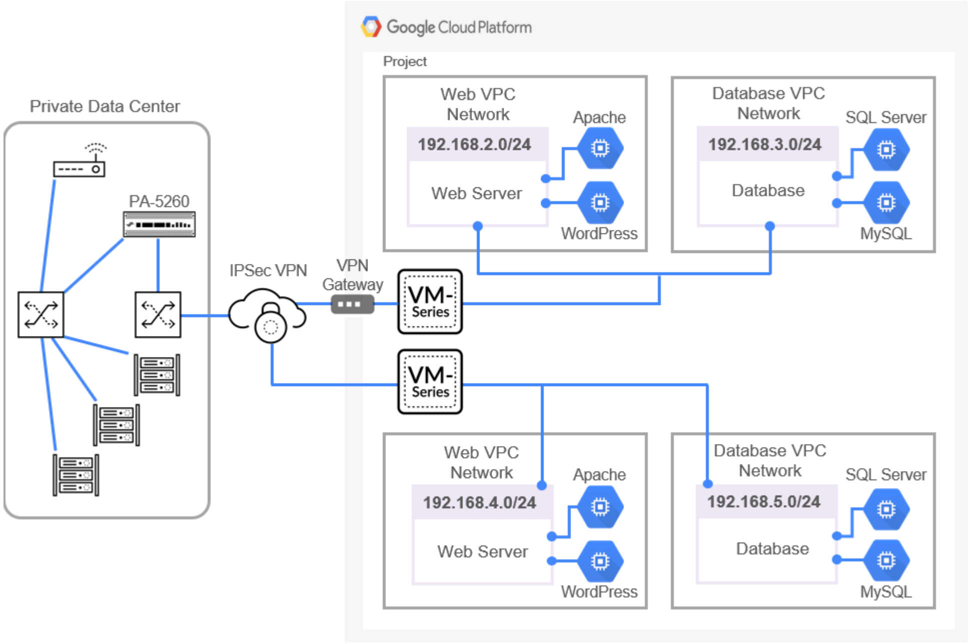
A segmentation gateway secures East/West traffic between virtual private clouds (VPCs) to ensure data protection compliance and application access. The following figure shows a firewall securing both North/South and East/West traffic.



Hybrid IPSec VPN

The VM-Series firewall serves as an IPSec VPN termination point, which enables secure communications to and from applications hosted on Google Cloud Platform (GCP).

The deployment in the figure below shows a site-to-site VPN from an on-premises network to a VM-Series firewall deployed on GCP and an IPSec connection from an on-premises network to a Google Cloud VPN gateway.



Prepare to Set Up VM-Series Firewalls on Google Public Cloud

The process to Deploy the VM-Series Firewall from Google Cloud Platform Marketplace requires preparation tasks.

If you are deploying using the Google Marketplace, you must create your project networks and subnetworks, and plan networks and IP address assignments for the VM-Series firewall interfaces in advance. During the deployment, you must choose from existing networks and subnetworks.

Refer to the following topics when planning your deployment:

- General Requirements
- Install the VM-Series Plugin on Panorama
- Install the Panorama Plugin for GCP
- Prepare to Deploy from the GCP Marketplace

General Requirements

The components in this checklist are common to deploying a VM-Series firewall that you manage directly or with Panorama. Additional requirements apply for Panorama plugin for services such as Stackdriver monitoring, VM monitoring, auto scaling or securing Kubernetes deployments.

Always consult the Compatibility Matrix for Panorama plugin information for public clouds.This release requires the following software:

- **GCP account**—You must have a GCP user account with a linked email address and you must know the username and password for that email address.
- **Google Cloud SDK**—If you have not done so, install Google Cloud SDK, which includes Google Cloud APIs, gcloud and other command line tools. You can use the command line interface to deploy the firewall template and other templates.
- **PAN-OS on VM-Series firewalls on GCP**—VM-Series firewalls running a PAN-OS version available from the Google Marketplace.
    - **VM-Series firewalls**—VM-Series firewalls that you want to manage from Panorama must be deployed in Google Cloud Platform using a Palo Alto Networks image from the Google Marketplace. Firewalls must meet the Minimum System Requirements for the VM-Series Firewall on GCP.
    - **VM-Series Licenses**—You must license a VM-Series firewall to obtain a serial number. A serial number is required to add a VM-Series firewall as a Panorama managed device. If you are using the Panorama plugin for GCP to deploy VM-Series firewalls you must supply a BYOL auth code. The Google Marketplace handles your service billing, but the firewalls you deploy will directly interface with the Palo Alto Networks licensing server.
    - **VM-Series plugin on the firewall**—VM-Series firewalls running PAN-OS 9.0 and later include the VM-Series plugin, which manages integration with public and private clouds. As shown in the Compatibility Matrix, the VM-Series plugin has a minimum version that corresponds to each PAN-OS release.

      When there is a major PAN-OS upgrade the VM-Series plugin version is automatically upgraded. For minor releases it is up to you to determine whether a VM-Series plugin upgrade is necessary, and if so, perform a manual upgrade. See Install the VM-Series Plugin on Panorama.
- **Panorama running in Management mode**—A Panorama physical or virtual appliance running a PAN-OS version that is the same or later than the managed firewalls. Virtual instances do not need to be deployed in GCP.

- You must have a licensed version of Panorama.
- Panorama must have network access to the VPCs in which the VMs you want to manage are deployed.
- If you intend to manage VMs deployed in GCP, or configure features such as auto scaling, your PAN-OS and VM-Series plugin versions must meet the Public Cloud requirements to support the Panorama plugin for GCP.
- VM-Series plugin on Panorama. See Install the VM-Series Plugin on Panorama

- **Panorama plugin for GCP version 2.0.0**—The GCP plugin manages the interactions required to license, bootstrap and configure firewalls deployed with the VM Monitoring or Auto Scaling templates. The GCP plugin, in conjunction with the VM Monitoring or Auto Scaling templates, uses Panorama templates template stacks, and device groups to program NAT rules that direct traffic to managed VM-Series firewalls.

  See Install the Panorama Plugin for GCP.

Install the VM-Series Plugin on Panorama

On Panorama, install or upgrade to the VM-Series plugin version that supports the GCP features you want to configure, as detailed in the Compatibility Matrix table for Public Clouds.

Initial installation—Because the VM-Series plugin is optional on Panorama, the first time you install you must download the VM-Series plugin from support.paloaltonetworks.com, then go to **Panorama** > **Device Deployment** > **Plugins** to upload and install.

Upgrade—Go to **Panorama** > **Device Deployment** > **Plugins** and click **Check Now**. Install a version that meets the requirements in the Compatibility Matrix table for Public Clouds.

Install the Panorama Plugin for GCP

The Panorama plugin for GCP is required if you want to use Panorama to manage VM Monitoring or Auto Scaling deployments created with Palo Alto Networks templates. Install the plugin version that supports the GCP features you want to configure, as detailed in the Compatibility Matrix table for Public Clouds.

You cannot upgrade the Panorama Plugin for GCP from version 1.0.0 to version 2.0.x. If you have installed version 1.0.0, remove it before installing 2.0.x.

STEP 1 | Verify your Panorama installation.

On Panorama, ensure that your PAN-OS version meets the requirements to support GCP auto scaling.

STEP 2 | Remove the Panorama plugin for GCP v1.0.

If you have the Panorama plugin v1.0 installed you must remove it.

STEP 3 | Install the Panorama plugin for GCP.

Select **Panorama** > **Plugins**, and type `gcp` in the search bar. **Install** the plugin version that supports the features you want to configure (see the Compatibility Matrix table for Public Clouds).

After the installation you can see the plugin in the Panorama dashboard **General Information** list. View **Panorama** > **Google Cloud Platform** and you see the **Setup**, **Monitoring Definition**, and **AutoScaling** interfaces.

STEP 4 | (Optional) If your Panorama appliances are in a high availability configuration, you must manually install the same version of the Google plugin on both Panorama peers.

Configure the Google plugin on the active Panorama peer only. On commit, the configuration syncs to the passive Panorama peer. Only the active Panorama peer polls Google VMs you have configured for VM Monitoring.

Prepare to Deploy from the GCP Marketplace

Review these requirements to ensure that you have proper accounts and permissions before you use the Google Marketplace to deploy the firewall on a Google Compute Engine (GCE) instance.

- General Accounts and Permissions
- Available Google Resources
- Google Authentication Methods
- SSH Key Pair

General Accounts and Permissions

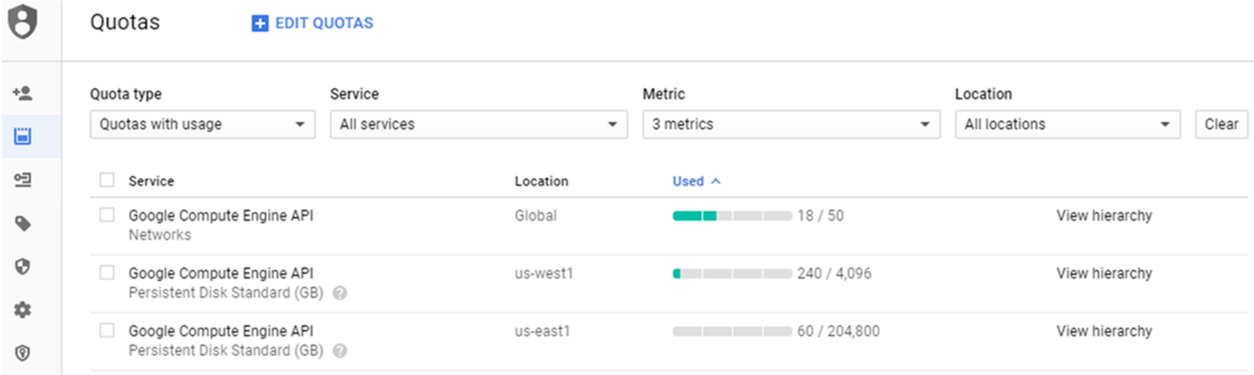
- □ You, and any users you allow, must have the following minimal roles or equivalent Identity and Access Management (IAM) permissions to connect to the VM-Series firewall:
  - □ **Compute Viewer**—Compute Viewer enables you to get and list compute engine resources without being able to read the data stored on those resources.
  - □ **Storage Object Viewer**—Enables you to bootstrap using a Google storage bucket in the same project.

  Users in your organization might have IAM permissions or predefined roles that are more permissive than required. Ensure that you appropriately restrict VM-Series firewall access.

  You can also restrict access with service accounts, as described in Google Authentication Methods.
- □ **Monitoring Metric Writer**—Required for Stackdriver.

Available Google Resources

Your project must have sufficient resources to deploy the VM-Series firewall as a Google Compute Engine instance. If you are deploying a GCP Marketplace solution, determine whether the solution deploys other VMs in addition to the firewall. In the Google Cloud Console, select **IAM & admin** > **Quotas** to review the resource quotas for your project and the networks and disk space consumed. If you are running out of resources you can ask Google to allocate more for your organization.



Quotas

Quota type: Quotas with usage
Service: All services
Metric: 3 metrics
Location: All locations

Service	Location	Used	
Google Compute Engine API Networks	Global	18 / 50	View hierarchy
Google Compute Engine API Persistent Disk Standard (GB)	us-west1	240 / 4,096	View hierarchy
Google Compute Engine API Persistent Disk Standard (GB)	us-east1	60 / 204,800	View hierarchy

Google Authentication Methods

GCP supports multiple ways to connect to an instance. You can authenticate with a service account or an SSH key pair.

1. **Service Accounts**—Service Accounts apply to applications or VMs—not to end users. They are commonly used to control access when you use programs or scripts, or when you access the firewall from the gcloud command line. If you are using Google Service Accounts to authenticate instances or applications, you must know the email address for the account(s). Refer to Creating and Managing Service Account Keys.

   Using a service account is necessary if you want to connect to the VM-Series firewall from outside the project—either from a different project or from the command line. For example, if you want to enable a physical next generation firewall to monitor your VM-Series firewall, you must save the VM-Series firewall service account information to a JSON file. In the physical firewall, you upload the file when you configure the connection.

   1. Select **IAM & Admin** > **Service accounts** and choose **+Create Service Account**.

      Enter the service account name and description, and click **Create**.
   2. Select a role type from the drop menu, and on the right, select an appropriate access level.

      For example, select Project > Editor. You can select multiple roles for a service account. When you are finished, click **Continue**.
   3. Grant specific users permission to access this service account. Select members from the **Permissions** column on the right to give them permission to access the roles in the previous step.
2. **SSH Keys**—If you deploy the VM-Series firewall from the Marketplace, you must supply one Open SSH key in RSA format for the Google Compute Engine instance metadata.

   The VM-Series firewall only accepts one key at deployment.

   At deployment time, you paste the public key into the Marketplace deployment, as described in SSH Key Pair. After deployment you use the private key to SSH in to the firewall to configure the administrator account. To add users, see Manage Firewall Administrators.

You can authenticate in several ways:

- **Create service accounts for instances**—You can create a service account for a specific instance or instance group, and grant specific permissions, which in turn can be granted to users.
- **Use the default service account for your project**—If you are using the Google Cloud Platform (GCP™) Console, then you logged in with your email address and can access a GCE instance based on whatever permissions or roles the project administrator assigned to your account.

  Every Google Compute Engine instance created with the Google Cloud Console or the gcloud command line tool has a default service account with the name in email address format:

  ```
  <project-number>-compute@developer.gserviceaccount.com
  ```

  To see the service account name for the firewall instance, view the instance details and scroll to the bottom (refer to the Compute Engine default service account).

  The default service account can manage authentication to VMs in the same project as a VM-Series firewall. Access scopes allow the firewall to initiate API calls to VMs in the Google Cloud project.
- **Use IAM permissions and the Google APIs**—If you use the Google SDK APIs and gcloud, then you must call the APIs to authenticate.

  - You typically use the Google SDK when you want to manage the firewall from a command line or you want to run a script to configure the firewall.
  - You need to access the Google APIs if a virtual machine you connect to has a custom image with applications that require Google APIs.

SSH Key Pair

When you deploy the VM-Series firewall from the Google Marketplace you need an SSH key pair to authenticate with the VM-Series firewall.

Create the key pair according to your key generator documentation. Do not edit the public key file. Editing risks introducing illegal characters.

The VM-Series firewall manages authentication differently than GCE instances. After deployment, you first log in with the **`admin`** user. The VM-series firewall default user name is accepted only once. After a successful login you set an administrator username and password for the VM-Series web interface (see Deploy the VM-Series Firewall from Google Cloud Platform Marketplace).

The Google Marketplace deployment interface **SSH key** field displays the following placeholder:

```
admin:ssh-rsa your-SSH-key
```

`admin` is the VM-Series firewall Administrator user name required to log in to the firewall for the first time. You add the **`admin:`** prefix into the Marketplace field when you deploy the VM-Series firewall.

You cannot log in to the VM-Series firewall if you do not supply the entire public key, or your key has illegal characters when you paste the key into the Marketplace **SSH key** field. When you SSH in to the VM-Series firewall for the first time, the public key is transferred to the firewall.

If the public key is corrupted, you must delete the deployment and start over. Any networks and subnetworks remain, but the firewall rules must be recreated.

STEP 1 | Create an SSH key pair and store the SSH Key pair in the default location for your operating system mentioned in Locating an SSH key.

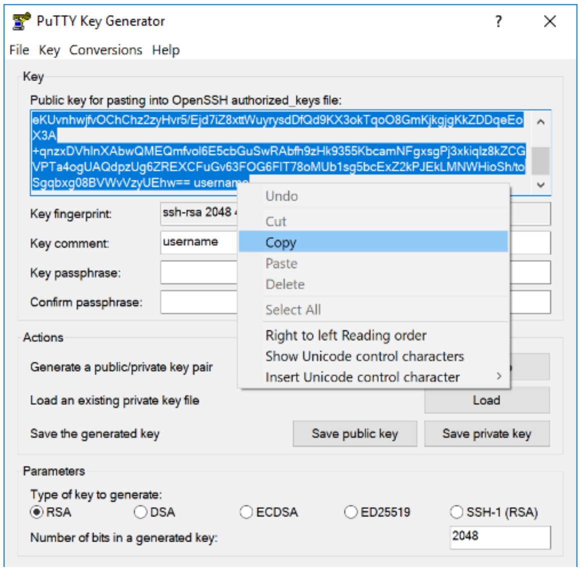
- **Linux or MacOS**—Use `ssh-keygen` to create the key pair in your .ssh directory.
- **Windows**—Use PuTTYgen to create the key pair.

  The content of the **Key comment** field does not matter to the VM-Series firewall; you can accept the default (the key creation date) or enter a comment that helps you remember the name of the key pair. Use the **Save private key** button to store the private key in your .ssh directory.

STEP 2 | Select the full public key.

- **Linux or MacOS**—Open your public key in a text editor and copy the public key.
- **Windows**—You must use the PuTTY Key Generator to view the public key. Launch PuTTYgen, click **Load**, and browse to private key you saved in your .ssh directory.

  In PuTTYgen, scroll down to ensure you select the entire key, right click, and choose Copy.



PuTTY Key Generator

Public key for pasting into OpenSSH authorized_keys file:

```
eKUvnhwjfvOChChz2zyHvr5/Ejd7iZ8xttWuyrysdDfQd9KX3okTqoO8GmKjkgjgKkZDDqeEo
X3A
+qnzxDVhlnXAbwQMEQmfvol6E5cbGuSwRAbfh9zHk9355KbcamNFgxsgPj3xkiqlz8kZCG
VPTa4ogUAQdpzUg6ZREXCFuGv63FOG6FIT78oMUb1sg5bcExZ2kPJEkLMNWHioSh/to
Sgqbxg08BVWvVzyUEhw== username
```

Key fingerprint: ssh-rsa 2048

Key comment: username

Key passphrase:

Confirm passphrase:

Type of key to generate: ● RSA ○ DSA ○ ECDSA ○ ED25519 ○ SSH-1 (RSA)

Number of bits in a generated key: 2048

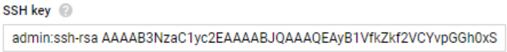
File
Key
Conversions
Help
Key
Undo
Cut
Copy
Paste
Delete
Select All
Right to left Reading order
Show Unicode control characters
Insert Unicode control character
Actions
Generate a public/private key pair
Load an existing private key file
Load
Save the generated key
Save public key
Save private key
Parameters

STEP 3 | Enter the public key in the SSH key field as detailed below.

1. In the Marketplace **SSH key** field, delete the placeholder text, and type:

   `admin:`

   Make sure there are no extra spaces following the colon.
2. Insert the cursor after **`admin:`** and choose **Paste as plain text**. The key must be on a single line, as shown below.

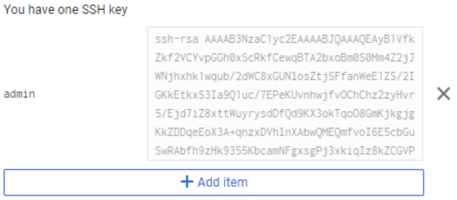


SSH key

```
admin:ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAQEAyB1VfkZkf2VCYvpGGh0xS
```

STEP 4 | Check the key.

After the deployment, and before you attempt to log in to the firewall, view the management instance and check the key for linefeeds or extra spaces:



You have one SSH key

admin

```
ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAQEAyB1Vfk
Zkf2VCYvpGGh0xScRkfCewqBTA2bxoBm0S0Mm4Z2jJ
WNjhxhk1wqub/2dWC8xGUNlosZtjSFfanWeE1ZS/2I
GKkEtkxS3Ia9Q1uc/7EPeKUvnhwjfvOChChz2zyHvr
5/Ejd7iZ8xttWuyrysdDfQd9KX3okTqoO8GmKjkgjg
KkZDDqeEoX3A+qnzxDVhlnXAbwQMEQmfvoI6E5cbGu
SwRAbfh9zHk9355KbcamNFgxsgPj3xkiqIz8kZCGVP
```

If the key looks right you are finished.

STEP 5 | (optional) If something is wrong you must replace the key.

1. Click the **X** to delete the key, then click **+ Add item**.
2. Type in **`admin:`** (no spaces) and copy and paste in the key again.
3. Click **Save** to deploy the updated deployment.
4. Re-check the key.

Virtual Private Cloud (VPC) Network Planning

Before you deploy from the Google Market place, make a plan for VPC networks (referred to as *networks*), subnetworks (also called *subnets*), and Google firewall rules. You must create networks and subnetworks before you start to Deploy the VM-Series Firewall from Google Cloud Platform Marketplace.

The Marketplace deployment page displays only networks and subnetworks that exist when you start the deployment. If a network is missing, you must exit the deployment, create the network, and start over.

- □ **VPC networks**—You must create a custom network specifically for each VM-Series firewall network interface.
  - □ See VM-Series Firewall Licenses for Public Clouds to determine the number of network interfaces needed based on your VM-Series firewall license. At a minimum, set up the three VPC networks and subnets required to launch the VM-Series firewall.
  - □ A GCP project has a default network with preset configurations and firewall rules; you can delete the default network, if unused.
  - □ By default, there are up to five networks in a project. Your GCP administrator can request additional networks for your project.
  - □ To connect to the management interface you must create a GCP firewall rules that allows access. You can do this during the deployment if you choose **Enable GCP Firewall rule for connections to Management interface** then supply a CIDR block for **Source IP in GCP Firewall rule for connections to Management Interface**.

  Be sure your networks include all instances you want to secure.
- □ **Subnetworks**—A compute engine instance can support up to eight Layer 3 interfaces on a single instance. The Management, Trust, and Untrust interfaces consume three interfaces and you can create up to five additional dataplane interfaces. Typically the dataplane interfaces represent application networks.
- □ **IP address**—You supply IP address ranges when you create interface subnetworks, and you have the option to enable an external address when you deploy a subnetwork.
  - When you create a network subnet, you must specify an IP address range. This range is used for your internal network, so it cannot overlap with other subnets.
  - During deployment, you can choose to enable an external IP address when you create a network interface. By default, you are given an ephemeral IP address. You cannot supply a reserved static IP address during the deployment, but you can promote the ephemeral address to a static IP address after you complete the deployment process (see Promoting an ephemeral external IP address).

Network Interface Planning

When you deploy from Google Cloud Platform Marketplace, the default VM-Series firewall deployment has three interfaces: the Management plane interface and the Untrust and Trust dataplane interfaces. You can define additional dataplane instances, depending on the available compute resources on your VM; see VM-Series Firewall Licenses for Public Clouds.

During the deployment you have the opportunity to name these interfaces.

Interface Order

When you deploy with Marketplace, the order of the network interfaces is predefined. The Management interface maps to eth0, Untrust to eth1, and Trust to eth2. Marketplace uses this order because mapping the Management interface to eth0 and the Untrusted interface to eth1 is a requirement if you need to Swap the Management Interface for load balancing.

Management Interface

The first network interface you add is mapped to eth0 on the firewall and includes the option to enable IP forwarding. You use this network interface to manage the VM-Series firewall. Typically, this interface has an external IP address.

> *An external IP address is only required if a dataplane interface is attached to the public subnet. At creation time, you can receive an ephemeral IP address and later promote it to a static IP address after you complete the deployment (refer to* Promoting an ephemeral external IP address*).*

Dataplane Interfaces (Untrust, Trust)

When you deploy from Marketplace, the order in which you add interfaces is predetermined.

- You configure the Untrust interface after the Management interface. This order means that the untrusted interface is mapped to eth1. The Untrust interfaces are typically attached to the public subnet, and have an external IP address.

  > *An external IP address is only required if a dataplane interface is attached to the public subnet. At creation time, you can receive an ephemeral IP address, then promote it to a static IP address, as discussed in* Promoting an ephemeral external IP address*.*

- The Trust interface follows the Untrust interface, and it is mapped to eth2. The Trust network often does not have an external IP address. You can add any additional dataplane interfaces after the Trust interface.

Additional Dataplane Interfaces

Plan interfaces for applications you must secure, such as web servers, databases, and other applications in your network. You can create up to five additional dataplane interfaces in addition to the three required to launch your firewall. Ensure that the applications you want to secure are in networks that connect to the VM-Series firewall.

Deploy the VM-Series Firewall on Google Cloud Platform

To deploy the VM-Series firewall using the GCP market place template, you must first create a VPC network for each interface on the firewall. After you deploy the firewall from the Google Marketplace, you can log in to the firewall to adjust the configuration to work within your GCP VPC configuration. You can also enable monitoring so you can collect metrics that enable you to improve resource management or create Security policy rules that automatically adapt to changes in your application environment.

- Deploy the VM-Series Firewall from Google Cloud Platform Marketplace
- Management Interface Swap for Google Cloud Platform Load Balancing
- Use the VM-Series Firewall CLI to Swap the Management Interface
- Enable Google Stackdriver Monitoring on the VM Series Firewall
- Enable VM Monitoring to Track VM Changes on GCP
- Secure Firewalls Deployed in GCP with Dynamic Address Groups
- Locate VM-Series Firewall Images in the GCP Marketplace

Deploy the VM-Series Firewall from Google Cloud Platform Marketplace

You can use Google® Cloud Platform Marketplace to deploy the VM-Series firewall on a VM-300 capacity license. The licensed images available from Cloud are:

- VM-Series Next-Generation Firewall Bundle 1
- VM-Series Next-Generation Firewall Bundle 2
- VM-Series Next-Generation Firewall (BYOL)

See Deploy the VM-Series Firewall from Google Cloud Platform Marketplace for more about these license options.

The Marketplace deploys an instance of the VM-Series firewall with a minimum of one management interface and two dataplane interfaces (Trust and Untrust). You can add additional dataplane interfaces for up to five Google Compute Engine instances in your virtual private cloud (VPC).

Before you deploy the VM-Series firewall, you must create or choose a project in your organization and create any networks and subnets that will connect to the firewall, as described in VPC Network Planning and Network Interface Planning.

You cannot attach multiple network interfaces to the same VPC network. Every interface you create must have a dedicated network with at least one subnet. Ensure that your networks include any additional dataplane instances you create.

STEP 1 | Locate the VM-Series firewall listing in the Marketplace.

1. Log in to the Google Cloud Console.
2. From the Products and Services menu, select **Marketplace**.
3. Search for **`VM-Series`**.
4. Select one of the VM-Series firewall licensing options.

STEP 2 | Click **Launch on Compute Engine**.

STEP 3 | Name the instance and choose resources.

1. Enter the **Deployment Name** (this name is displayed in the Deployment Manager). The name must be unique and cannot conflict with any other deployment in the project.
2. Select a Zone. See Regions and Zones for a list of supported zones.
3. Select a **Machine Type** based on the VM-Series System Requirements for your license and the Minimum System Requirements for the VM-Series Firewall on Google Cloud Platform.

STEP 4 | Specify instance metadata. The options **Bootstrap Bucket** and **Interface Swap** affect the initial configuration the first time the VM-Series firewall boots.

1. **Bootstrap Bucket** (Optional)—If you plan to use a bootstrap file, enter the name of a storage bucket with the bootstrap configuration described in Minimum System Requirements for the VM-Series Firewall on Google Cloud Platforms.
2. **Interface Swap** (Optional)—Swap the Management interface (eth0) and the first dataplane interface (eth1) at deployment time. Interface swap is only necessary when you deploy the VM-Series firewall behind Google Cloud Platform HTTP(S) Load Balancing. For details, see Management Interface Swap for Google Cloud Platform Load Balancing.
3. **SSH key**—Paste in the public key from an SSH key pair. Follow the instructions for your OS in SSH Key Pair, to create, copy, and paste the key. Windows users must view the key in PuTTY, copy from the user interface, and paste into Marketplace deployment.

   If the key is not formatted properly, the VM-Series firewall does not allow you to log in. You must delete the deployment and start over.

4. Click **More** to reveal additional metadata options. The options **blockProjectKeys**, and **enableSerialConsole** are properties of the instance; you can change these metadata values after a successful deployment.

   - **blockProjectKeys** (Optional)—If you Block Project Keys, you can use only the public SSH key you supply to access the instance.
   - **enableSerialConsole** (Optional)—Interacting with the Serial Console enables you to monitor instance creation and perform interactive debugging tasks.

STEP 5 | Configure the boot disk.

1. **Boot disk type**—Select from SSD Persistent disk or Standard Persistent Disk. See Storage Options.
2. Enter the **Boot disk size**—60GB is the minimum size. You can edit the disk size later but you must stop the VM to do so.

STEP 6 | Configure the management interface.

1. **Management VPC Network name**—Choose an existing network
2. **Management Subnet name**—Choose an existing subnet.
3. **Enable External IP for Management interface** (Optional)—If you enable this option, you can use the IP address assigned to the VM-Series firewall management interface to use SSH to access the VM-Series firewall web interface.
4. **Enable GCP Firewall rule for connections to Management interface** (Optional)—This option automatically creates a GCP firewall Allow rule for an external source IP address that you supply.
5. **Source IP in GCP Firewall rule for connections to Management Interface**—If you **Enable GCP Firewall rule for connections to Management interface**, enter a source IP address or a CIDR block.

   - Do not use 0.0.0.0/0. Supply an IP address or a CIDR block that corresponds to your dedicated management IP addresses or network. Do not make the source network range larger than necessary.
   - Verify the address to ensure that you do not lock yourself out.

STEP 7 | Configure the Untrust dataplane interface.

1. **Untrust VPC Network name**—Choose an existing network.

2. **Untrust Subnet name**—Choose an existing subnet.
3. **Enable External IP for Untrust**—Enable GCP to provide an ephemeral IP address to act as the external IP address.

STEP 8 | Configure the Trust dataplane interface.

1. **Trust VPC Network name**—Choose an existing network.
2. **Trust Subnet name**—Choose an existing network.
3. **Enable External IP for Trust**—Enable GCP to provide an ephemeral IP address to act as the external IP address.

STEP 9 | Configure additional interfaces. You must enter the number of dataplane interfaces you want to add; the default is 0 (none). The deployment page always displays fields for five additional dataplanes numbered 4 through 8.

1. **Additional Dataplane interfaces**—Enter the number of additional dataplane instances.

   If this number is 0 (default), dataplane numbers 4 through 8 are ignored even if you fill out the interface fields. If, for example, you specify 2 and then fill out information for three interfaces, only the first two are created.
2. **Additional Dataplane # VPC name**—Choose an existing network.
3. **Dataplane # Subnet name**—Choose a subnet that exists.
4. **Enable External IP for dataplane # interface**—Enable GCP to provide an ephemeral IP address to act as the external IP address.

STEP 10 | **Deploy** the instance.

STEP 11 | Use Google Cloud Deployment Manager to view and manage your deployment.

STEP 12 | Use the CLI to change the administrator password on the firewall.

1. Log in to the VM-Series firewall from the command line. In your SSH tool, connect to the External IP for the management interface, and specify the path to your private key.

   Windows users: Use PuTTY to connect to the VM-Series firewall and issue command line instructions. To specify the path to the private key, select **Connection** > **SSH** > **Auth**. In **Private key file for authentication**: click **Browse** to select your private key.
2. Enter configuration mode:

   ```
   VMfirewall> configure
   ```
3. Enter the following command:

   ```
   VMfirewall# set mgt-config users admin password
   ```
4. Enter and confirm a new password for the administrator.
5. Commit your new password:

   ```
   VMfirewall# commit
   ```
6. Return to command mode:

   ```
   VMfirewall# exit
   ```
7. (Optional) If you used a bootstrap file for interface swap, use the following command to view the interface mapping:

   ```
   VMfirewall> debug show vm-series interfaces all
   ```

STEP 13 | Access the VM-Series firewall web interface.

1. In a browser, create a secure (https) connection to the IP address for the management interface.

If you get a network error, check to see that you have a GCP firewall rule that allows the connection.

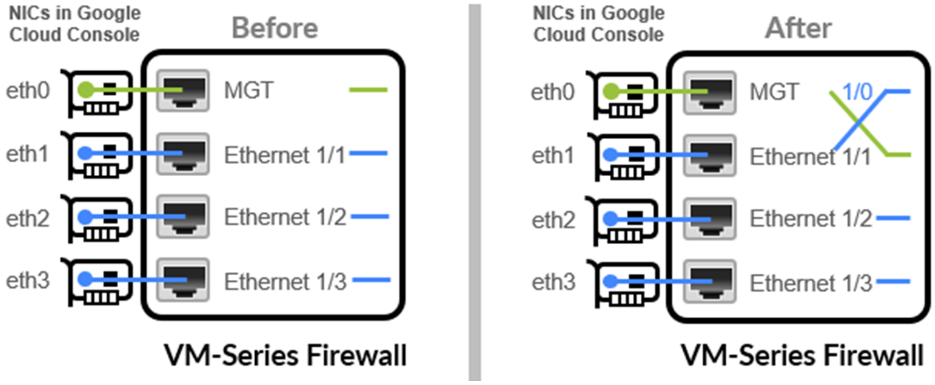
2. When prompted, enter the username (admin) and the administrator password you specified from the CLI.
3. (Optional) If you bootstrapped, then Verify Bootstrap Completion.

   If you see problems, search the log information on the VM-Series firewall. Choose **Monitor** > **System** and, in the manual search field, enter `description contains 'bootstrap'` and look for a message in the results that indicates that the bootstrap was successful.

After you log in to the firewall, you can add administrators and create interfaces, zones, NAT rules, and policy rules, just as you would on a physical firewall.

Management Interface Swap for Google Cloud Platform Load Balancing

Because internal load balancing can send traffic only to the primary interface of the next hop load-balanced Google Compute Engine instance, the VM-Series firewall must be able to use eth0 for dataplane traffic.



The firewall can receive dataplane traffic on eth0 if the VM-Series firewall is behind the Google Cloud Platform internal load balancing interface.

- The VM-Series firewalls secure traffic outbound directly to the internet without requiring 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 the Google internal load balancing address.

To allow the firewall to send and receive dataplane traffic on eth0 instead of eth1, you must swap the mapping of the internal load balancing network interface within the firewall so that eth0 maps to ethernet 1/1, and eth1 maps to the MGT interface on the firewall.

Swap the management interface mapping before you configure the firewall and define policy rules.

Swapping how the interfaces are mapped allows Google Cloud Platform to distribute and route traffic to healthy instances of the VM-Series firewall located in the same or different zones.

Swap the Management Interface

You can swap the interfaces when you Deploy the VM-Series Firewall from Google Cloud Platform Marketplace, or you can configure the firewall after it is created.

At Creation

When you deploy the VM-Series firewall, you can enable interface swap in two ways.

- **Google Cloud Console** — In the Create Instance form, enter a key-value pair in the **Metadata** field, where **`mgmt-interface-swap`** is the key, and **`enable`** is the value.
- **Bootstrap File** — Create a bootstrap file the includes the **`mgmt-interface-swap`** operational command in the bootstrap configuration, as described in Bootstrap the VM-Series Firewall on Google Cloud Platform. In the Create Instance form, enter a key-value pair in the **Metadata** field to enable the bootstrap option.

After Deployment

Log in to the firewall, and Use the VM-Series Firewall CLI to Swap the Management Interface. In operational mode, issue the following command:

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

If you configured the VM-Series firewall before swapping, check whether any IP address changes for eth0 and eth1 impact policy rules.

From the Google Cloud Console you cannot confirm whether you have swapped eth0 and eth1. After swapping, you must remember that load balancing is on eth0 and the firewall management interface is eth1 so that you can properly configure Google Cloud Platform load balancing, and create security policy rules to secure load balancing to one or more VM-Series firewalls.

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

This task is only required if your architecture places the VM-Series firewall behind the GCP internal load balancer.

If you did not specify metadata to swap the management interface (MGT) with the dataplane interface when you deployed the firewall, you can use the VM-Series firewall CLI to enable the firewall to receive dataplane traffic on the primary interface.

STEP 1 | Deploy the VM-Series Firewall from Google Cloud Platform Marketplace.

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

STEP 2 | On the Google Cloud Console, view the VM instance details to verify the network interface IP addresses of the eth1 interface and verify that any security 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
```

You can view the default mapping from the command line interface. The output is similar to this:

```
> debug show vm-series interfaces all
Interface_name  Base-OS_port
```

```
mgt               eth0
 Ethernet1/1       eth1
 Ethernet1/2       eth2
```

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

STEP 5 | Reboot the firewall for the swap to take effect:

```
request restart system
```

STEP 6 | Verify that the interfaces have been swapped:

```
debug show vm-series interfaces all
```

Enable Google Stackdriver Monitoring on the VM Series Firewall

A VM-Series firewall on a Google® Compute Engine instance can publish custom PAN-OS metrics to Google Stackdriver. These metrics allow you to assess performance and usage patterns so that you can manage your firewall resources accordingly.

- Google Stackdriver Permissions
- Enable Google Stackdriver

Google Stackdriver Permissions

Authentication requirements vary based on whether you can use the default service account to authenticate or need to use Google APIs to authenticate.

You can authenticate in two ways:

- **Use the default service account for the VM-Series firewall instance**—If you are using the Google Cloud Platform (GCP™) Console, then you logged in with your email address and can access the instance based on whatever permissions or roles the project administrator assigned to your account.
- **Use IAM permissions and the Google APIs**—If you use the Google SDK APIs and gcloud, then you must call the APIs to authenticate. You typically use the Google SDK when you want to manage the firewall from a command line or you want to run a script to configure the firewall.

Every Google Compute Engine instance created with the Google Cloud Console or the gcloud command line tool has a default service account with the name in email address format:

```
<project-number>-compute@developer.gserviceaccount.com
```

To see the service account name for the firewall instance, view the instance details and scroll to the bottom (refer to the Compute Engine default service account).

The default service account can manage authentication for monitoring VMs in the same project as a VM-Series firewall.

- Access scopes allow the firewall to initiate API calls to monitor VMs in a Google Cloud project.
- You don't need to access the Google APIs unless one of the monitored virtual machines has a custom image with applications that require Google APIs.

If you want to set up monitoring from a physical firewall or from a VM-Series firewall in a different project, you must use the Google APIs to authenticate. There are two prerequisites:

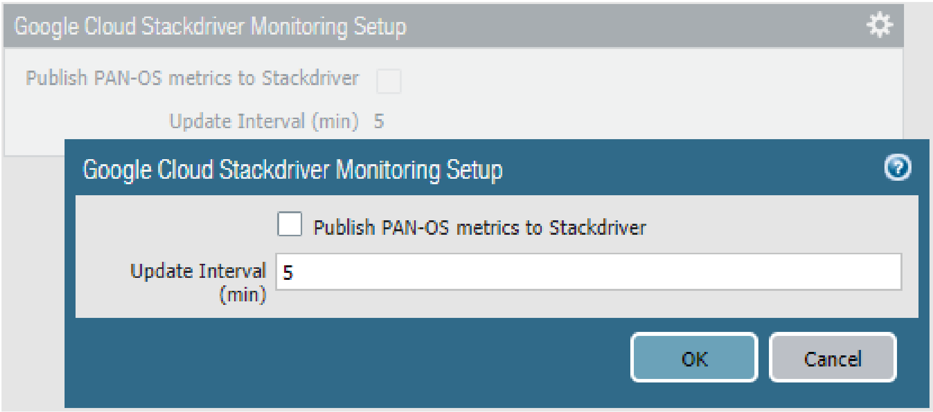
- Google APIs must be installed. See Install the Panorama Plugin for GCP.
- Your account must have the roles Monitoring Metric Writer and Stackdriver Account Viewer.

Enable Google Stackdriver

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

STEP 1 | Push PAN-OS metrics from a VM-Series firewall on a Google Compute Engine instance to Stackdriver.

1. Log in to the web interface on the VM-Series firewall.
2. Select **Device** > **VM-Series**. Under Google Cloud Stackdriver Monitoring Setup, click the Edit cog .
    1. Check **Publish PAN-OS metrics to Stackdriver**.



Google Cloud Stackdriver Monitoring Setup

Publish PAN-OS metrics to Stackdriver □

Update Interval (min) 5

Google Cloud Stackdriver Monitoring Setup

□ Publish PAN-OS metrics to Stackdriver

Update Interval (min) 5

    2. Set the **Update Interval** (range is 1 - 60 minutes; default is 5). This is the frequency at which the firewall publishes the metrics to Stackdriver.
    3. Click **OK**.
3. **Commit** your changes.

    Wait until the firewall starts to publish metrics to Stackdriver before you configure alarms for PAN-OS metrics.

STEP 2 | Verify that you can see the metrics on Stackdriver.

1. In the Google Cloud Console, select **Products and Services** > **Monitoring**.
2. In Stackdriver, choose **Resources** > **Metrics Explorer**.
3. In the **Find resource type and metric** section, enter `custom` in the search field to filter the PAN-OS metrics.

STEP 3 | Configure alerts and actions for PAN-OS metrics on Stackdriver. See Monitoring Quickstart for Google Compute Engine and Stackdriver Introduction to Alerting.

Enable VM Monitoring to Track VM Changes on GCP

You can enable any firewall that runs PAN-OS 9.0 (virtual or physical) to monitor application workloads deployed on Google Compute Engine instances. In this procedure you manually log in to the firewall to enable VM monitoring. If you want to use the Panorama plugin for GCP to configure VM Monitoring, see Configure VM Monitoring with the Panorama Plugin for GCP.

VM Monitoring enables you to monitor a predefined set of metadata elements or attributes on the VM-Series firewall. In the PAN-OS 9.0 Administrator's Guide, see Attributes Monitored on Virtual Machines in Cloud Platforms.

With an awareness of virtual machine adds, moves, and deletes within a Google VPC, you can create Security policy rules that automatically adapt to changes in your application environment. As you deploy or move virtual machines, the firewall collects attributes (or metadata elements). You can use this metadata for policy matching and to define Dynamic Address Groups (see Use Dynamic Address Groups to Secure Instances Within the VPC).

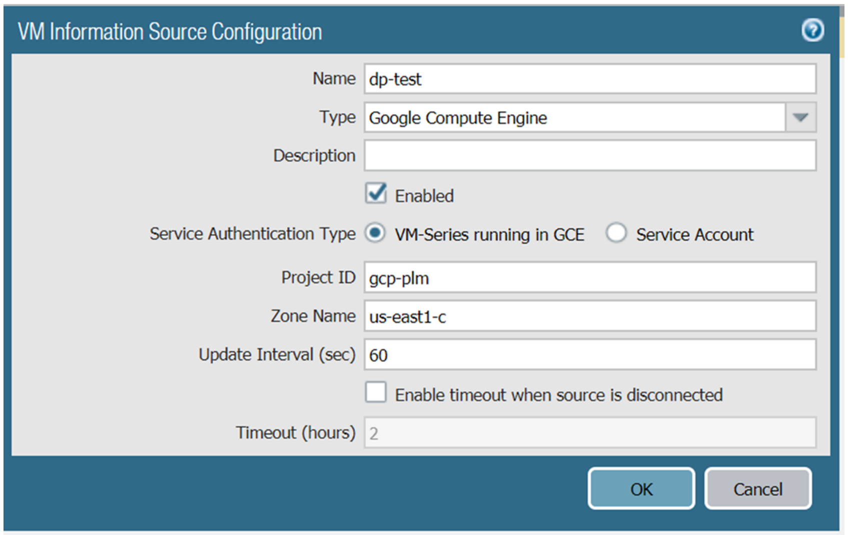
You can configure up to ten VM information sources on each firewall or on each virtual system on a firewall capable of multiple virtual systems. Information sources can also be pushed using Panorama templates.

To perform VM monitoring, you must have the IAM role Monitoring Metric Writer.

STEP 1 | Log in to your deployed firewall.

STEP 2 | Enable VM Monitoring.

1. Select **Device** > **VM Information Sources**.
2. **Add** a VM information source and enter the following information:
   - Specify a **Name** to identify the instance that you want to monitor.
   - Select the Google Compute Engine **Type**.
   - Select **Enabled**.
   - Choose the **Service Authentication Type**.
     - If you choose **VM-Series running in GCE**, you are authenticating with the default service account generated when an instance is created. This is part of the instance metadata.
     - If you want to monitor from a firewall outside the current project, choose **Service Account**. You must upload the service account credentials in JSON format. See Creating and Managing Service Account Keys.
   - (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 every 60 seconds—an update takes up to 60 seconds plus the configured polling interval.



VM Information Source Configuration

Name dp-test

Type Google Compute Engine

Description

☑ Enabled

Service Authentication Type ● VM-Series running in GCE ○ Service Account

Project ID gcp-plm

Zone Name us-east1-c

Update Interval (sec) 60

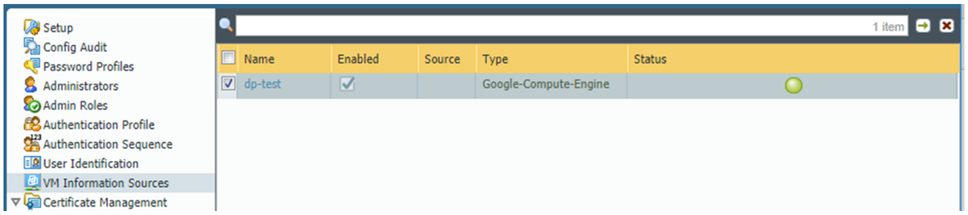
☐ Enable timeout when source is disconnected

Timeout (hours) 2

   - (Optional) To change the number of hours before timeout, check **Enable timeout when the source is disconnected** and enter the Timeout (in hours) before the connection to the monitored source is closed (range is 2 to 10; default is 2).

If the firewall cannot access the host and the specified limit is reached, the firewall closes the connection to the source.

- Click **OK** and **Commit** your changes.



□	Name	Enabled	Source	Type	Status
☑	dp-test	✓		Google-Compute-Engine	●

STEP 3 | Verify the connection status.

If the connection status is pending or disconnected, verify that the source is operational and that the firewall is able to access the source. If you use a port other than the Management (MGT) port for communicating with the monitored source, then you must change the service route (select **Device** > **Setup** > **Services**, click **Service Route Configuration**, and modify the **Source Interface** for the **VM Monitor** service).

Secure Firewalls Deployed in GCP with Dynamic Address Groups

In a dynamic environment such as the Google® Cloud Platform (GCP™), where you launch new instances on demand, the administrative overhead in managing Security policy can be cumbersome. Using use dynamic address groups in policy enables agility and prevents disruption in services or gaps in protection.

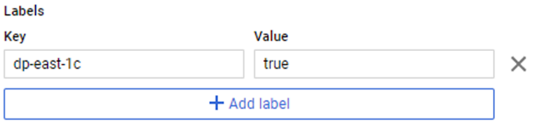
This workflow assumes that you have deployed the VM-Series firewall, configured some applications on instances, and enabled Google Stackdriver Monitoring.

STEP 1 | Configure the deployed firewall to monitor the VPC as described in Enable VM Monitoring to Track VM Changes on GCP.

STEP 2 | Label instances in the VPC.

A label is a name-value pair. You can label resources from the Google Cloud Console, from Google API calls, or from the Google Cloud Shell. In this task we are labeling instances; however, labels can be applied to many resources, as described in Labeling Resources.

You can also add labels from the Instance browser.



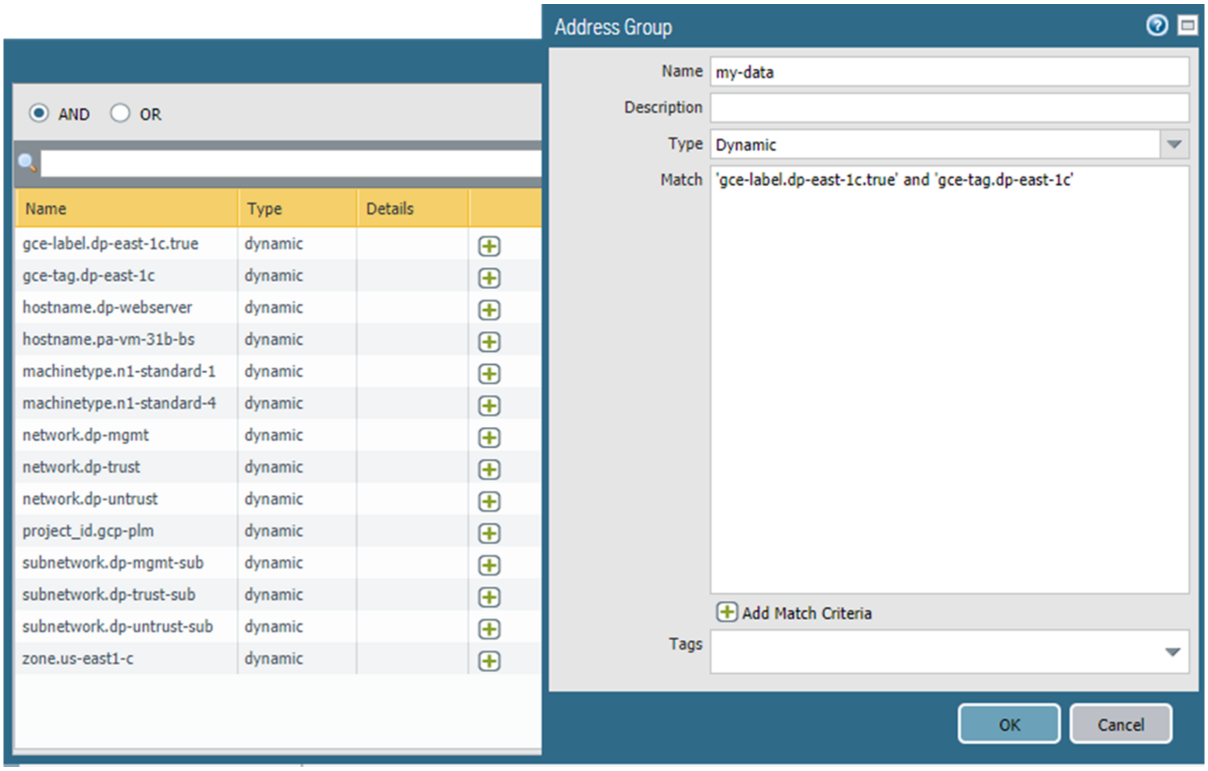
Key	Value
dp-east-1c	true

The labels you create support your strategy for differentiating your resources in ways that are useful to your Security policy.

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

1. Select **Objects** > **Address Groups**.
2. **Add** a dynamic address group and specify a **Name** and a **Description**.
3. Set **Type** to **Dynamic**.
4. Define the match criteria.

    1. **Add Match Criteria** and select the **And** operator.
    2. Select the attributes to filter for or to match against.



Name	Type	Details
gce-label.dp-east-1c.true	dynamic	
gce-tag.dp-east-1c	dynamic	
hostname.dp-webserver	dynamic	
hostname.pa-vm-31b-bs	dynamic	
machinetype.n1-standard-1	dynamic	
machinetype.n1-standard-4	dynamic	
network.dp-mgmt	dynamic	
network.dp-trust	dynamic	
network.dp-untrust	dynamic	
project_id.gcp-plm	dynamic	
subnetwork.dp-mgmt-sub	dynamic	
subnetwork.dp-trust-sub	dynamic	
subnetwork.dp-untrust-sub	dynamic	
zone.us-east1-c	dynamic	

Name: my-data

Description:

Type: Dynamic

Match: 'gce-label.dp-east-1c.true' and 'gce-tag.dp-east-1c'

Tags:

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

STEP 4 | Use the dynamic address group in a Security policy rule.

Create a rule to allow internet access to any web server that belongs to the dynamic address group called my-data.

1. Select **Policies** > **Security**.
2. **Add** a rule and 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, **Add** your new my-data group.
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, set the **Profile Type** to **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 that are displayed here.

1. Select **Policies** > **Security** and select the rule.
2. Select **Inspect** from the drop-down. You can also verify that the match criteria is accurate.
3. Click **more** to verify that the list of registered IP addresses is displayed.

Locate VM-Series Firewall Images in the GCP Marketplace

The official VM-Series images published on Google Cloud Platform Marketplace are available in the **`paloaltonetworksgcp-public`** project. You need to know the secure path to these images if you want to call them from the **`gcloud`** command line, or refer to them in a template you have written or adapted.

- BYOL: vmseries-byol-<*version*>
- PAYG Bundle 1: vmseries-bundle1-<*version*>
- PAYG Bundle 2: vmseries-bundle2-<*version*>

Use the gcloud CLI to find the current image names and project:

```
gcloud compute images list --project paloaltonetworksgcp-public
--no-standard-images
```

```
NAME                  PROJECT                   FAMILY  DEPRECATED  STATUS
vmseries-bundle1-810  paloaltonetworksgcp-public                      READY
vmseries-bundle2-810  paloaltonetworksgcp-public                      READY
vmseries-byol-810     paloaltonetworksgcp-public                      READY
```

Add the **`--uri`** flag to see the image paths:

```
gcloud compute images list --project paloaltonetworksgcp-public
--no-standard-images --uri
```

```
https://www.googleapis.com/compute/v1/projects/paloaltonetworksgcp-public/
global/images/vmseries-bundle1-810
https://www.googleapis.com/compute/v1/projects/paloaltonetworksgcp-public/
global/images/vmseries-bundle2-810
https://www.googleapis.com/compute/v1/projects/paloaltonetworksgcp-public/
global/images/vmseries-byol-810
```

VM Monitoring with the Panorama Plugin for GCP

The Panorama plugin for Google Cloud Platform (GCP) version 2.0.0 enables you to create a VM monitoring configuration that authenticates with a GCP project and monitors VM-Series firewalls and other VMs deployed within it. Once you establish a connection to your project, the plugin can retrieve IP-address-to-tag communication between Panorama and GCP assets. Tags can be predefined attributes, user-defined labels for VMs, and user-defined network tags (see Review and Create Tags).

The Panorama plugin for GCP retrieves the internal and external IP addresses from running VMs, and periodically retrieves IP-to-tag mappings from VMs in connected GCP VPCs.

You can use tags to organize VMs into dynamic address groups, and then reference your tags in Security policy rules that allow or deny traffic to specific VM IP addresses. To consistently enforce Security policy, you can then push rules to your VM-Series firewalls.

- Configure VM Monitoring with the Panorama Plugin for GCP

Configure VM Monitoring with the Panorama Plugin for GCP

This topic describes the steps to prepare your GCP assets for VM monitoring, review the required Panorama elements, and describes how to configure VM Monitoring in the Panorama plugin for Google Cloud Platform (GCP).

- Configure GCP Assets for VM Monitoring
- Configure VM Monitoring with the Panorama Plugin for GCP
  - Prepare Panorama to Configure VM Monitoring
  - Set Up VM Monitoring

Configure GCP Assets for VM Monitoring

You can monitor VM-Series firewalls you deployed from the GCP marketplace, firewalls you deployed with auto scaling Firewall templates, GCE instances you created from to the GCP console or the gcloud command line, or other virtual machines deployed in GCP. If you deploy PAN-OS VMs from the Marketplace, follow the instructions in Set Up the VM-Series Firewall on Google Cloud Platform.

Review IAM Roles

Ensure that you have the following minimum permissions for VM Monitoring tasks:

- In GCP console, create a service account for your project and grant the permission project owner or editor.

  Service account creation cannot be automated. If you do not have permission to create a service account you can ask an administrator to create it and assign an appropriate role to you.
- View your service account: read-only.
- View PAN-OS VMs deployed from the Google Marketplace: Compute viewer.
- Assign a user-defined tag to an instance: Project owner, editor or Instance Admin.

Create a Service Account

Before you use the GCP plugin on Panorama to configure VM Monitoring, you must use the GCP console to create a service account that grants permissions to access your GCP project, VM-Series firewalls deployed within it, any other VMs that you want Panorama to manage, and related networks and subnetworks. The

GCP plugin for Panorama retrieves pre-defined attributes for Google assets, user defined VM labels, and user-defined network tags.

Every project has a default service account that was automatically created when the project was created. If you create a separate service account specifically for VM Monitoring you have greater control of users and their roles. You can configure up to 100 service accounts per project.

STEP 1 | In the Google Cloud Platform console, select the project you want to monitor.

STEP 2 | Select **IAM & Admin** > **Service accounts** and choose **+Create Service Account**.

Enter the service account name and description, and click **Create**.

STEP 3 | Select a role type from the drop menu, and on the right, select an appropriate access level.

For example, select Project > Editor. You can select multiple roles for a service account.

When you are finished, click **Continue**.

STEP 4 | Grant specific users permission to access this service account. Select members from the **Permissions** column on the right to give them permission to access the roles in the previous step.

STEP 5 | (Optional) Click **+CREATE KEY** to create a credential that allows you to authenticate with the Google Cloud CLI to access VM-Series firewalls, networks, and other VMs associated with this service account.

The key is downloaded automatically. Be sure to store it in a secure location. The JSON format for the generated private key is as follows:

```
{
  "type": "service_account",
  "project_id": "gcp-xxx",
  "private_key_id": "252e1e7a2e9c84b5d4dbb6195b1de074594b6499",
  "private_key": "-----BEGIN PRIVATE KEY-----
\nMIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQDAd0i+RMKCtrsO
\n4KHnzTAPrgoBjRgpjyNcvQmdUqHr\n-----END PRIVATE KEY-----\n",
  "client_email": "dlp-vm-monit-svc-acct@gcp-xxx.iam.gserviceaccount.com",
  "client_id": "108932514695821539229",
  "auth_uri": "https://accounts.google.com/o/oauth2/auth",
  "token_uri": "https://oauth2.googleapis.com/token",
  "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/
certs",
  "client_x509_cert_url": "https://www.googleapis.com/robot/v1/metadata/
x509/dlp-vm-monit-svc-acct%40gcp-xxx.iam.gserviceaccount.com"
}
```

Review and Create Tags

"Tag" is a general term for predefined attributes, user-defined labels, and user-defined network tags.

- Predefined tags (attributes) are automatically created for Google VMs. When you configure VM Monitoring you can choose to monitor all 8 of the predefined attributes, or you can create a customized list of attributes to monitor.
- You can define your own tags for VM labels and network tags.

Tag VMs and networks so that you can identify and group them so that you can structure rules to enforce Security policy. You can tag any VM deployed in your Google project—for example, a VM-Series firewall, a web server, an application server, or a load balancer.

- Tags must be associated with a VM. This also applies to networks and subnetworks.
- If there are multiple IP addresses associated with an instance (for example if you tagged the VM-Series firewall trust and untrust interfaces), Panorama generates multiple sets of tag information.

The total number of tags that the Panorama plugin can retrieve and register depends on the PAN-OS version Panorama is running and the version of the managed VM-Series firewalls:

Google zone, Google region, VPC name, and Subnet name are used to tag network interfaces on VMs with multiple interfaces. specific to network interface.

Predefined Attributes

The Google Cloud Platform plugin for Panorama retrieves the following predefined tags from any managed VM:

- **Project ID**—For example: google.project-id.myProjectId.

  To find your project information in the Google console, select your project, then select **IAM & Admin** > **Settings**.
- **Service account**—Your service account in the form of an email address. For example: google.svc-accnt.sa-name@project-id.iam.gserviceaccount.com.

  To find your Service account, view the VM instance details.
- **VPC name**—The name of the VPC network for a managed VM. For example: google.vpc-name.myvnet.
- **Subnet name**—The name of a subnet you created for a managed VM interface. For example, for the VM-Series firewall untrust interface, the name of the subnet you created for the untrust interface: google.subnet-name-untrust.web.
- OS SKU—The operating system you chose when you deployed the managed VM. For example: google.os-sku.centos-7.

  This attribute is not supported if the VM uses a custom image.
- **Google zone**—The zone you selected when you deployed the VM. For example: google.zone.us-east1-c.
- **Google region**—The region containing the zone you selected. For example: google.region.us-east1.
- **Instance group name**—For example: google.instance-group.myInstanceGroup. To view or create an instance group in the Google console, select **Compute Engine** > **Instance Group**.

User-defined Labels

Panorama uses up to 16 user-defined labels. If you have more than 16 labels, Panorama sorts your user-defined labels alphabetically and uses the first 16 tags.

Review the Google requirements for label key-value pairs: Keys have a minimum length of 1 character and a maximum length of 63 characters, and cannot be empty. Values can be empty, and have a maximum length of 63 characters.

To create or view labels in the GCP console, go to **Compute Engine** > **VM Instances** and select **Show Info Panel**. Select one or more VMs and in the **Info** Panel, select **Labels**. Click **+Add a label**, add a key and value, and click **Save**.

User-defined Network Tags

Panorama uses up to 8 user-defined network tags, If you have more than 8 tags, Panorama sorts your user-defined labels alphabetically and uses the first 8 tags.

Note that Google limits network tags as follows:

- Maximum 63 characters per tag.
- You can use lowercase letters, numbers, and dashes; a tag must start with a lowercase letter, and end with a number or a lowercase letter.

To create or view network tags in the GCP console, go to **Compute Engine** > **VM Instances** and select an instance. **Edit** the instance, and scroll down to **Network Tags**, enter tags (separated by commas), and **Save**. See Configuring Network Tags.

Configure VM Monitoring with the Panorama Plugin for GCP

After you tag your GCP assets and create a service account, make your assets available to Panorama so you can set up VM monitoring.

Prepare Panorama to Configure VM Monitoring

Follow these steps to enable Panorama to manage and monitor your GCP assets. Any VM deployed in GCP can be a managed device in Panorama.

STEP 1 | In Panorama, add the VM-Series firewalls and other VMs associated with your GCP project as managed devices.

STEP 2 | Add a Device Group and assign managed devices to it. A Device Group is a group of firewalls or virtual systems that you want to manage as a group.

A VM can be a member of only one Device Group. Plan your Device Groups carefully.

STEP 3 | Add a template. Name the template and accept the default VPC.

STEP 4 | Add a template stack. **Add** the stack, **Add** the template you just created, and select your devices.

STEP 5 | Commit the changes.

Set Up VM Monitoring

STEP 1 | If you have not done so, Install the Panorama Plugin for GCP.

STEP 2 | Log in to the Panorama web interface and select **Panorama** > **Google Cloud Platform**.

STEP 3 | Set up VM monitoring.

1. Configure general settings.
    1. Select **Panorama** > **Google Cloud Platform** > **Setup** > **General**. To edit the settings, click the gear.
        - Check **Enable Monitoring** to permit VM monitoring on all projects for which you configure a service account.
        - Enter the **Monitoring Interval** in seconds. This is the length of time between tag retrieval events.
2. **Add** a notify group. A notify group is a list of Device Groups to which Panorama pushes IP-address-to-tag mappings and updates.

    A project can have only one notify group.

    1. **Select Panorama** > **Google Cloud Platform** > **Setup** > **Notify Groups** and click **Add**.
    2. Enter a **Name** to identify the group of firewalls to which Panorama pushes the VM information (IP address-to-tag mappings) it retrieves.

3. Select the **Device Groups** to which Panorama will push the VM information (IP address-to-tag mappings) retrieved from your project. The VM-Series firewalls use the update to determine the current member list for Dynamic Address Groups referenced in Security policy.

   Plan your Device Groups carefully.

4. Select predefined or custom tags.

   - **Select All 8 Predefined Tags**—Choose this option to select all predefined attributes (tags).
   - **Custom Tags**—Choose this option to create tag lists for predefined attributes, user-defined labels, and user-defined network tags.

5. - Make sure to include all relevant Device Groups in a single notify group.
   - If you want to 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.

3. **Add** a GCP Service Account Credential.

   - Name the service account credential.
   - (Optional) Enter a description of the service account.
   - **Browse** to upload the JSON file generated when you created the service account.

   You must use the Panorama web interface. You cannot use the CLI to add a service account

   You can only use a service account for one credential. Do not create multiple credentials from a single JSON file.

STEP 4 | Create a **Monitoring Definition**.

A monitoring definition consists of the service account credential for your project and a notify group. All the networking assets in your project are monitored, and the tags retrieved are pushed to the Device Groups you list in your monitoring definition. When you add a new monitoring definition, it is enabled by default.

A project can have only one monitoring definition, and a monitoring definition can include only one notify group.

1. **Select Panorama** > **Google Cloud Platform** > **Monitoring Definition** and click Add.
2. **Name** the monitoring definition.
3. Enter an optional **Description** for the project and assets you are monitoring.
4. Select the **Service Account** credential you created in the previous step.
5. Select a **Notify Group**.
6. **Enable** monitoring for the elements associated with this service account.

STEP 5 | **Commit** the changes on Panorama.

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

STEP 6 | Verify that you can view the VM information on Panorama, and define the match criteria for Dynamic Address Groups.

On HA failover, the newly active Panorama attempts to reconnect to Google Cloud Platform and retrieve tags for all monitoring definitions. If there is an error with reconnecting even one monitoring definition, Panorama generates a system log message:

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

If you see this error, fix the issue in Panorama. For example remove an invalid subscription or provide valid credentials, and commit your changes to enable Panorama to reconnect and retrieve the tags for all monitoring definitions. Even when Panorama is disconnected from Google Cloud Platform, the firewalls have the list of all tags that had been retrieved before failover, and can continue to enforce policy on that list of IP addresses. When you delete a monitoring definition, Panorama removes all tags associated with registered VMs. As a best practice, configure action-oriented log forwarding to an HTTPS destination *from Panorama so that you can take immediate action.*

Auto Scaling the VM-Series Firewall on Google Cloud Platform

The Panorama plugin for Google Cloud Platform (GCP) version 2.0.0 assists you in deploying the VM-Series firewall in GCP and enables Panorama to manage VM-Series firewalls securing VM monitoring or auto scaling deployments in GCP. Using Panorama for centralized policy and firewall management increases operational efficiency in managing and maintaining a distributed network of firewalls.

With Panorama maintaining your GCP managed instance groups you can create application enablement policies that protect and control the network.

The auto scaling deployment supports using a shared VPC network configuration or VPC network peering to create a common VPC network in which a host project contains shared VPC networks and the VM-Series firewalls, and a service project contains a vm-based or container-based application deployment (a Kubernetes cluster). Palo Alto networks supplies templates to help you deploy the VM-Series firewalls in the host project and deploy an optional sample application in the service project.

BYOL and PAYG licenses can be used for the VM-Series firewalls. During licensing, VM-Series firewall instances talk directly to the Palo Alto Networks license server.

If you choose BYOL your deployment can deactivate license instances in response to a scale-down event. If a VM-Series firewall's deployment information is configured in the Panorama plugin for GCP and the firewall is automatically removed, Panorama detects the firewall status and automatically deregisters the firewall.

- Auto Scaling Components for Google Cloud Platform
- Deploy GCP Auto Scaling Templates

Auto Scaling Components for Google Cloud Platform

Typical GCP auto scaling deployments use a host project and a service project and form a common VPC network between the two. The Panorama plugin for GCP can secure an auto scaling deployment in a single project with host and service VPCs, or host and service projects in a shared VPC or peered VPC network configuration, where the host project contains the VM-Series firewalls and the shared VPC networks, and the service project contains your application deployment. If your application is deployed in a Kubernetes cluster, a peered VPC is required.

- Auto Scaling Requirements
- Prepare to Deploy the Auto Scaling Templates

Auto Scaling Requirements

- □ **General Requirements**—Ensure your environment meets the basic Requirements.
- □ **Panorama Plugin for GCP**—If you have not done so, Install the Panorama Plugin for GCP.

  If you previously installed the Panorama plugin for GCP version 1.0.0, remove it before you install 2.0.0. You cannot upgrade.

- □ **Palo Alto Networks Auto Scale templates version 1.0**—Palo Alto Networks provides the templates to deploy VM-Series firewall instances in the host project and configure and deploy a sample application in a service project. See About the Auto Scaling Templates for more about the templates.

  Download the templates from GitHub. The zip file contains separate zip files for the firewall and application templates.

Prepare to Deploy the Auto Scaling Templates

Complete the following tasks before you deploy the auto scaling templates.

- Prepare a Host Project and Required Service Accounts
- Obtain a Licensing API Key
- Configure the Panorama Plugin for GCP to Secure an Auto Scaling Deployment
- Prepare a VM-Series Firewall Bootstrap Package for Auto Scaling

Prepare a Host Project and Required Service Accounts

You need a host project and a service project to form the shared VPC topology that supports the firewall and application templates. You can create a new host project or prepare an existing project to act as your host.

To set up the Shared VPC an organization administrator must grant the host project administrator the Shared VPC Admin role. The Shared VPC Admin can enable a project to act as a host, and grant the Service Project Admin role to the service project administrator. Review the GCP documentation on Administrators and IAM roles.

STEP 1 | In the GCP console, create a GCP project to act as the host. If you want to use an existing project, skip to the next step.

To create a new project, select your organization or **No organization**, click **New Project** and fill in your project information. Note, this is your only chance to **EDIT** the project ID.

The Google Cloud SDK must be installed and configured so that you can authenticate with your host project from the CLI. You will use the command line interface to deploy the firewall template and the application template, and to attach the service project to the host project.

STEP 2 | Enable APIs and services required for auto scaling. The required APIs are:

- □ Cloud Pub/Sub API
- □ Cloud Deployment Manager API
- □ Cloud Storage API
- □ Compute Engine API
- □ Google Compute Engine Instance Group Manager API
- □ Google Compute Engine Instance Group Updater API
- □ Google Compute Engine Instance Groups API
- □ Kubernetes Engine API
- □ Stackdriver API
- □ Stackdriver Logging API
- □ Stackdriver Monitoring API

You can enable APIs from the GCP console or the GCP CLI, as shown below.

Enable APIs from the GCP console

1. Select the host project, and from the Navigation menu, select **APIs & Services**.
2. Search for and view each required API.
3. **ENABLE** any APIs that do not display the “API enabled” status.

Enable APIs from the CLI

1. In the CLI, view your configuration to ensure that you are in the correct project.

```
gcloud config list
```

If not, set the project as follows:

```
gcloud config set project <project-name>
```

2. Issue the following commands to enable the required APIs.

```
gcloud services enable pubsub.googleapis.com
gcloud services enable deploymentmanager.googleapis.com
gcloud services enable storage-component.googleapis.com
gcloud services enable compute.googleapis.com
gcloud services enable replicapool.googleapis.com
gcloud services enable replicapoolupdater.googleapis.com
gcloud services enable resourceviews.googleapis.com
gcloud services enable container.googleapis.com
gcloud services enable stackdriver.googleapis.com
gcloud services enable logging.googleapis.com
gcloud services enable monitoring.googleapis.com
```

3. Confirm that the required APIs are enabled.

```
gcloud services list --enabled
```

STEP 3 | Create a service account for deploying the VM-Series firewall, and assign the IAM roles required for auto scaling a service or a Kubernetes cluster.

When you configure the firewall templates you add the email address for this service account to the VM-Series firewall `.yaml` file. Within the host project, the template uses credentials from this service account to create a host VPC with subnets, deploy VM-Series firewalls in the VPC, configure Stackdriver custom metrics, create a Pub/Sub topic, and more.

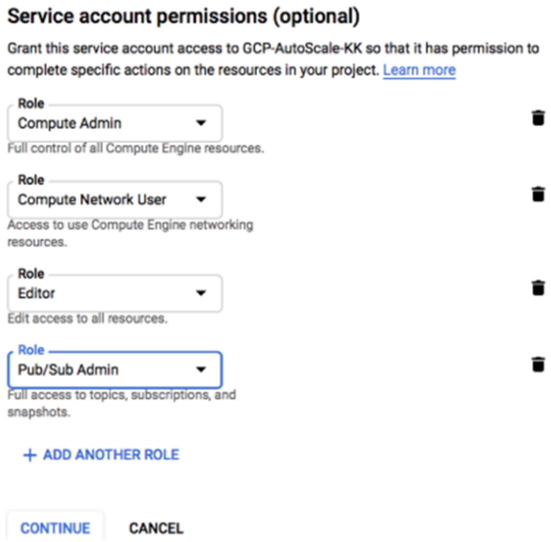
1. In the GCP console select **IAM & Admin** > **Service accounts** and select **+CREATE SERVICE ACCOUNT**.

   Fill in the service account details and click **CREATE**.

2. Give the service account permission to auto-scale resources in this project.

   Select a role type from the drop menu, and on the right, select an appropriate access level. For example, select Project > Editor. You can select multiple roles for a service account.

   □ Compute Engine > Compute Admin
   □ Compute Engine > Compute Network User
   □ Pub/Sub > Admin
   □ Monitoring > Monitoring Metric Writer
   □ Stackdriver > Stackdriver Accounts Editor
   □ Storage > Storage Admin
   □ (GKE only) Kubernetes > Kubernetes Engine Cluster Admin
   □ (GKE only) Kubernetes > Kubernetes Engine Viewer

3. 

Service account permissions (optional)

Grant this service account access to GCP-AutoScale-KK so that it has permission to complete specific actions on the resources in your project. Learn more

Role: Compute Admin
Full control of all Compute Engine resources.

Role: Compute Network User
Access to use Compute Engine networking resources.

Role: Editor
Edit access to all resources.

Role: Pub/Sub Admin
Full access to topics, subscriptions, and snapshots.

   Continue when you are finished adding roles.
4. Click **+CREATE KEY** to create a key for the host service account.
   - (Optional) Add email addresses to grant other users or administrators access to this service account.
   - Click JSON to download the private key in JSON form.
   - Store the key in a safe location. You will need this key when you Deploy GCP Auto Scaling Templates.
5. Click **DONE**.

STEP 4 | Create a service account that a Panorama administrator can use to interact with this host project.

1. In the GCP console select **IAM & Admin** > **Service accounts** and select **+CREATE SERVICE ACCOUNT**.
2. Fill in the service account details and click **CREATE**.
3. Grant service account access.

   Select a role type from the drop menu, and on the right, select an appropriate access level. For example, select Project > Editor. You can select multiple roles for a service account.

   □ Compute Engine > Compute Viewer
   □ Deployment Manager > Viewer
   □ Pub/Sub > Admin

   Click **CONTINUE**.
4. Click **+CREATE KEY** to create a key for the host service account.
   - (Optional) Add email addresses to grant other users or administrators access to this service account.
   - Select JSON to download the private key in JSON form.
   - Store the key in a safe location. You will need this key when you Configure the Panorama Plugin for GCP to Secure an Auto Scaling Deployment.

STEP 5 | (optional) In the CLI, ensure you can communicate with your new host project.

1. Set your project to the host project you just created.

```
gcloud set project <your-autoscale-host-project-name>
```

2. Create a configuration for auto scaling. Your new configuration is automatically activated unless you disable activation.

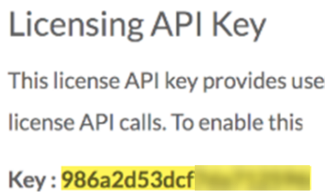
   ```
   gcloud config configurations create <CONFIGURATION_NAME> gcloud config
   list
   ```

Obtain a Licensing API Key

You need a Licensing API key so Panorama can license and de-license managed assets in GCP.

STEP 1 | On support.paloaltonetworks.com select**Assets** > **Licensing API** and click **Enable**. The key is displayed.

> *Only a Super User can view the Enable link to generate this key. See* How to Enable, Regenerate, Extend the Licensing API Key.



Licensing API Key

This license API key provides user

license API calls. To enable this

Key : 986a2d53dcf[redacted]

STEP 2 | Select the key and copy it.

STEP 3 | From the CLI, SSH in to Panorama and issue the following command, replacing <key> with the API key you copied from the support portal:

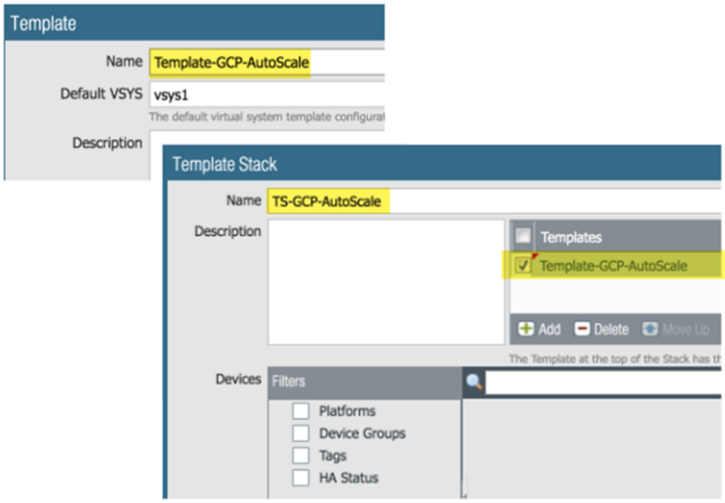
```
request license api-key set key <key>
```

```
API Key is successfully set
```

Configure the Panorama Plugin for GCP to Secure an Auto Scaling Deployment

In Panorama, create assets to support the auto scaling firewall deployment.

STEP 1 | Create a template, and a template stack that includes the template, and **Commit** the changes.



Template

Name Template-GCP-AutoScale

Default VSYS vsys1

The default virtual system template configurat

Description

Template Stack

Name TS-GCP-AutoScale

Description

Templates

☑ Template-GCP-AutoScale

The Template at the top of the Stack has th

Devices

Filters

□ Platforms

□ Device Groups

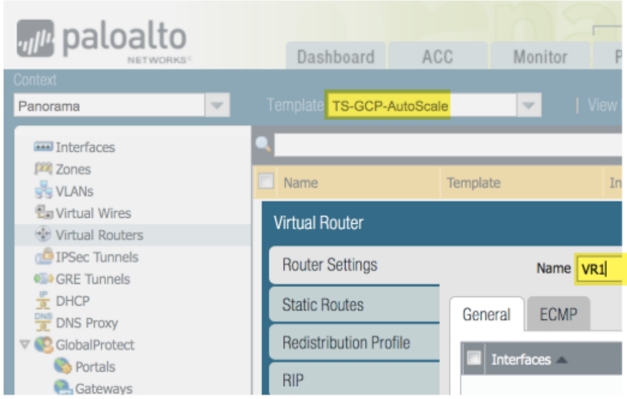
□ Tags

□ HA Status

STEP 2 | In the **Network** context, select either the template or the template stack. Select **Virtual Routers** and **Add** a virtual router.

When the firewall template creates static routes, they are added to this virtual router.

Define only one router for the auto scale deployment.



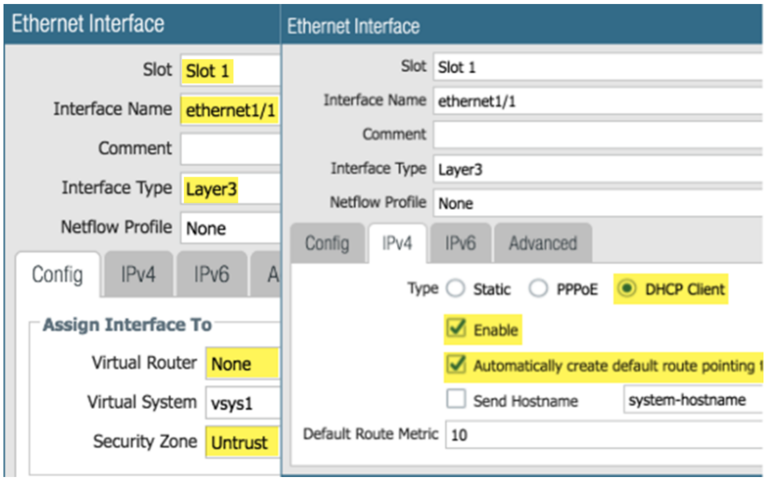
Context: Panorama

Template: TS-GCP-AutoScale

Name: VR1

STEP 3 | In the **Network** context, select the template you created, select **Interfaces** and **Add Interface**.

- On the Config tab, select a slot, select the **Interface name** and select the Layer3 **Interface Type**. From the **Security Zone** menu, select **New Zone**, name the zone Untrust and click **OK**.
- On the **IPv4** tab enable **DHCP Client** and **Automatically create default route pointing to default gateway provided by server** (enabled by default) and click **OK**.



Ethernet Interface

Slot: Slot 1

Interface Name: ethernet1/1

Comment:

Interface Type: Layer3

Netflow Profile: None

Assign Interface To

Virtual Router: None

Virtual System: vsys1

Security Zone: Untrust

Ethernet Interface

Slot: Slot 1

Interface Name: ethernet1/1

Comment:

Interface Type: Layer3

Netflow Profile: None

Type ○ Static ○ PPPoE ● DHCP Client

☑ Enable

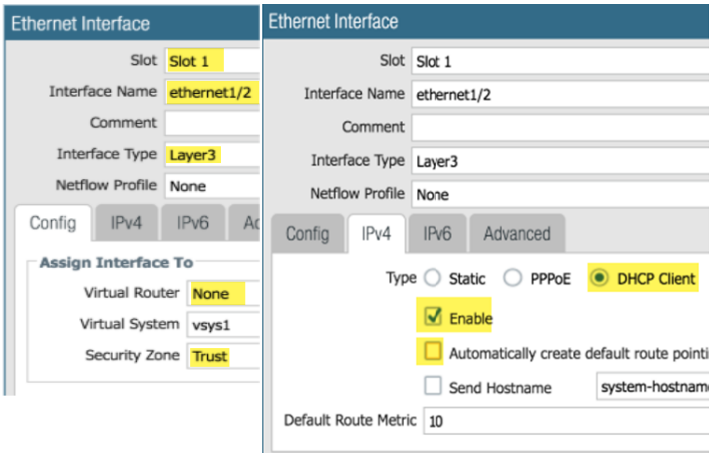
☑ Automatically create default route pointing t

□ Send Hostname system-hostname

Default Route Metric: 10

STEP 4 | Add the ethernet1/2 (Trust) Layer 3 interface.

- On the Config tab, chose the same slot as the previous step, select the **Interface name** (ethernet1/2), and select the Layer3 **Interface Type**. From the **Security Zone** menu, select **New Zone** name the zone Trust and click **OK**.
- On the **IPv4** tab enable **DHCP Client**, disable **Automatically create default route pointing to default gateway provided by server** and click **OK**.



Ethernet Interface

Slot: Slot 1
Interface Name: ethernet1/2
Comment:
Interface Type: Layer3
Netflow Profile: None

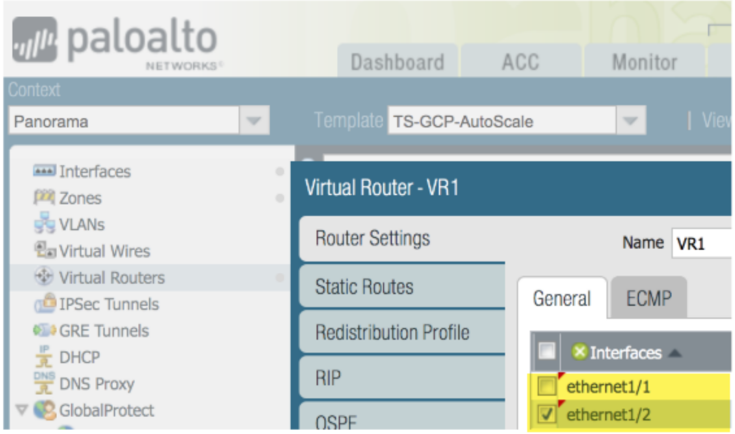
Assign Interface To
Virtual Router: None
Virtual System: vsys1
Security Zone: Trust

Ethernet Interface

Slot: Slot 1
Interface Name: ethernet1/2
Comment:
Interface Type: Layer3
Netflow Profile: None

Type ○ Static ○ PPPoE ● DHCP Client
☑ Enable
□ Automatically create default route pointi
□ Send Hostname system-hostnam
Default Route Metric: 10

STEP 5 | Return to your template stack and the virtual router you created earlier. Place the untrust and trust interfaces (ethernet1/1 and ethernet1/2) in the virtual router, and click **OK**.



Context: Panorama
Template: TS-GCP-AutoScale

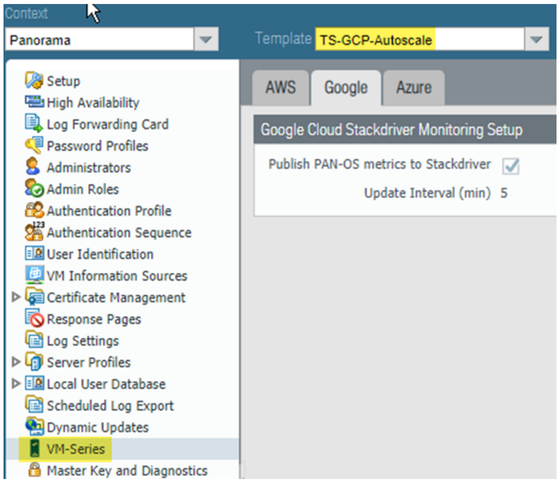
Virtual Router - VR1
Name: VR1

Interfaces
□ ethernet1/1
☑ ethernet1/2

STEP 6 | Configure Stackdriver for your auto scaling deployment.

You must have the VM-Series plugin on Panorama to configure Stackdriver.

1. In the **Device** context, select the template stack you created earlier from the Template drop menu.
2. Select **Device** > **VM-Series** > **Google** and click the Edit cog (⚙). Enable **Publish PAN-OS metrics to Stackdriver**.



Context: Panorama
Template: TS-GCP-Autoscale

Google Cloud Stackdriver Monitoring Setup
Publish PAN-OS metrics to Stackdriver ☑
Update Interval (min) 5

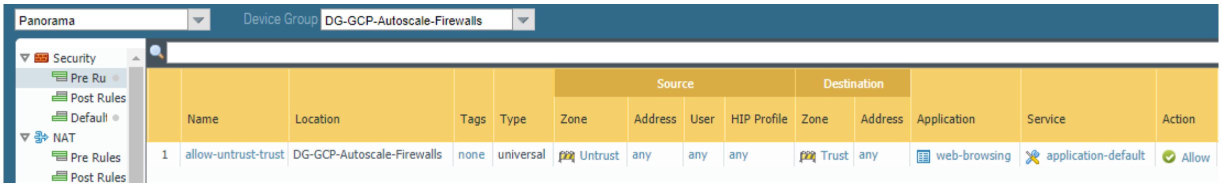
3. Commit your changes.

STEP 7 | Create a Device Group that references the template or template stack you created in step 1.

This Device Group will contain the VM-Series firewalls you create with the firewall template.

1. Add a security policy that allows web-browsing traffic from Untrust to Trust.

   In the Policies context, select the Device Group you just created. Select **Security** > **Pre Rules** and **Add** the following security policy.



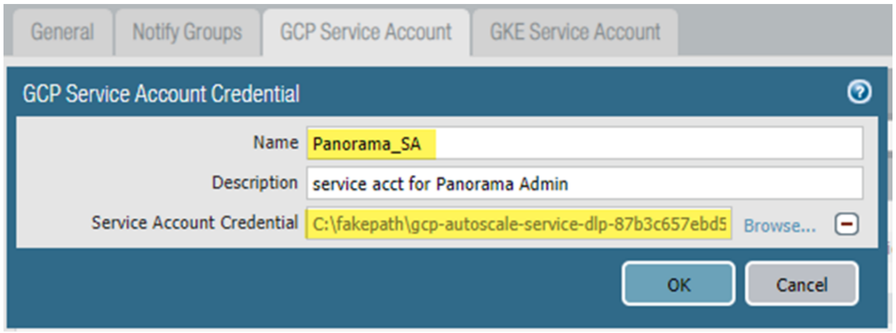
Panorama

Device Group: DG-GCP-Autoscale-Firewalls

					Source				Destination				
	Name	Location	Tags	Type	Zone	Address	User	HIP Profile	Zone	Address	Application	Service	Action
1	allow-untrust-trust	DG-GCP-Autoscale-Firewalls	none	universal	Untrust	any	any	any	Trust	any	web-browsing	application-default	Allow

STEP 8 | Set up the GCP service account for the host project.

1. In the Panorama context, expand Google Cloud Platform, select **Setup**, and click **Add**.
2. Supply a name and description for the host service account you created in Step 4.
3. Upload the JSON credentials file you created in Step 4.4.



GCP Service Account Credential

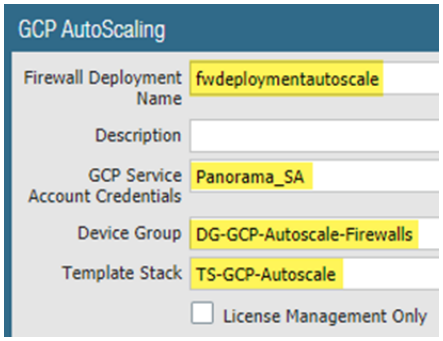
Name: Panorama_SA

Description: service acct for Panorama Admin

Service Account Credential: C:\fakepath\gcp-autoscale-service-dlp-87b3c657ebd5

STEP 9 | Set up auto scaling on the Panorama plugin for GCP.

1. In the Panorama context, expand Google Cloud Platform, select AutoScaling, and click **Add**.
2. Supply the Firewall Deployment Name and an optional description for the deployment.
3. For the GCP Service Account Credential, supply the GCP service account name from Step 8.



GCP AutoScaling

Firewall Deployment Name: fwdeploymentautoscale

Description:

GCP Service Account Credentials: Panorama_SA

Device Group: DG-GCP-Autoscale-Firewalls

Template Stack: TS-GCP-Autoscale

□ License Management Only

4. Chose the Device Group you created in Step 7, and the Template Stack you created in Step 1.
5. Disable **License Management Only** to ensure traffic is secured.

STEP 10 | Commit your changes.

Prepare a VM-Series Firewall Bootstrap Package for Auto Scaling

During bootstrap, the initial request from the firewall provides the host IP address and serial number, and the VM auth key so Panorama can validate the VM auth key and add the firewall as a managed device. Panorama can then assign the firewall to the appropriate device group and template so that you can centrally configure and administer the firewall using Panorama.

In this case, you must generate a VM auth key on Panorama and include the key in the init-cfg.txt file that you use for bootstrapping. The VM auth key allows Panorama to authenticate the newly bootstrapped VM-Series firewall. The bootstrap package must include.

- In the /config directory, an init-cfg.txt file that includes the Panorama IP address
- In the /license directory, the VM authentication key in a file named authcodes.

  The lifetime of the key can vary between 1 hour and 8760 hours (1 year). After the specified time, the key expires and Panorama will not register VM-Series firewalls without a valid auth-key in this connection request.

STEP 1 | Set up a Google storage bucket with the folders required to Bootstrap the VM-Series Firewall on Google Cloud Platform. You can use an existing bootstrap package or create a new bootstrap package, for these folders.

STEP 2 | Edit the values in the sample **init-cfg.txt** file to customize the file for your environment.

The firewall templates include a sample init-cfg.txt file.

Parameter	Value	Comment
type	dhcp-client	
hostname	<pa-vm>	Optional name you assigned when you prepared the host project. Only required if a specific host is necessary, and dhcp-send-hostname is no.
vm-auth-key	<vmauthkey>	A key that Panorama must validate before adding a firewall as a managed device. See Generate the VM Auth Key On Panorama.
panorama-server	<panorama-ip>	The IP address of the Panorama management device you configured in Configure the Panorama Plugin for GCP to Secure an Auto Scaling Deployment
tplname	<template-stack-name>	The template stack you created in Configure the Panorama Plugin for GCP to Secure an Auto Scaling Deployment.

Parameter	Value	Comment
dgname	<dg-name>	The name of the Device Group you created in the Panorama Plugin for GCP.
dns-primary		Your primary DNS server.
dns-secondary		Your secondary DNS server.
dhcp-send-hostname	yes	Leave as is.
dhcp-send-client-id	yes	Leave as is.
dhcp-accept-server-hostname	yes	Leave as is.
dhcp-accept-server-domain	yes	Leave as is.

STEP 3 | Upload your edited **`init-cfg.txt`** file to the **`/config`** folder in your bootstrap package.

STEP 4 | If you are using BYOL, create a text file named **`authcodes`** (no extension), add your auth code, and upload the file to the **`/license`** folder.

Deploy GCP Auto Scaling Templates

- About the Auto Scaling Templates
- Deploy the Firewall Template
- Prepare a Service Project
- Configure the Shared VPC
- Deploy the Application Template
- Onboard a New Application
- Sample GKE Service Templates

About the Auto Scaling Templates

Download the Palo Alto Networks auto scaling templates from https://github.com/PaloAltoNetworks/GCP-AutoScaling. The zip file contains separate zips for firewall and application templates. Each zip is a template directory containing several files, but you only need to edit the YAML files.

- Firewall Templates
- Application Template

Firewall Templates

The firewall directory files create VM-Series firewalls and other deployment resources. They create new networks and the familiar subnetworks for the VM-Series firewall: management, untrust, and trust. They also deploy a Cloud Pub/Sub messaging service to relay information from GCP to the Panorama plugin for GCP. With this infrastructure in place, the plugin can leverage dynamic address groups to apply security policy on inbound traffic routed to services running on GCP, and use auto scale metrics to deploy VM-Series firewalls to meet increased demand for application workload resources or to eliminate firewalls that are no longer needed.

To configure your load balancer, edit the `.yaml` file for an external application load balancer (ALB) or network load balancer (NLB).

- **ALB** (HTTP External Load Balancer)

  To customize an ALB, edit `vm-series-fw-alb.yaml`.

  HTTP external load balancer is a proxy-based load balancer that performs SNAT and DNAT on the inbound traffic from Internet. The HTTP load balancer is designed to support only the 80 and 8080 TCP ports.

  To support multiple applications using HTTP load balancer in load balancer sandwich architecture, we can use the GCP HTTP load balancer *urlMap* and *namedPort* to map different URLs to different ports in the load balancer. In turn, the VM-Series firewall can translate the ports to different applications, each represented by one internal load balancer per application.
- **NLB** (TCP Load Balancer)

  To customize an NLB, edit `vm-series-fw-nlb.yaml`.

  TCP load balancer is a non-proxy based load balancer, which means it doesn't perform NATing on inbound traffic from the Internet.

  TCP load balancer in GCP allows adding multiple frontend IP addresses with an arbitrary port, making it possible to support multiple applications.

  Another advantage of TCP load balancer is that the original client IP address is preserved, which is desirable for some applications.

Application Template

The application directory provides a sample application. You configure and deploy an internal load balancer (ILB) to enable your application servers to subscribe to the Pub/Sub service and communicate with your VM-Series firewalls and the GCP plugin on Panorama.

To customize the application template, edit `apps.yaml` as described in Deploy the Firewall Template and Application Template.

Deploy the Firewall Template

Edit the Firewall Templates from the host project.

STEP 1 | Edit the **`vm-series-fw-nlb.yaml`** or **`vm-series-fw-alb.yaml`** environment variables to reflect your cloud environment.

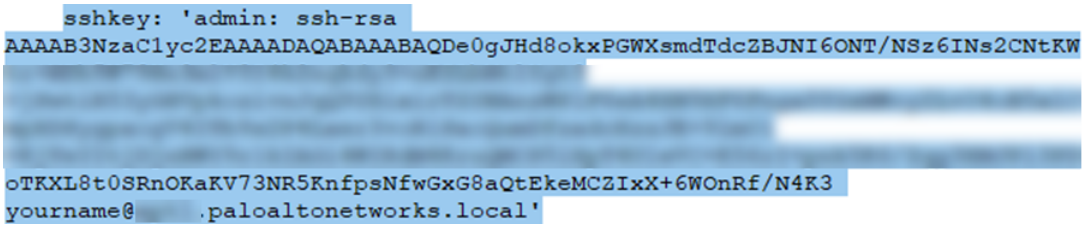
The sample in this workflow is for the NLB. See vm-series-fw-nlb.yaml and vm-series-fw-alb.yaml for further explanation of the template parameters.

```
properties:
    region: us-east1
    zones:
    -us-east1-b
    # Do not modify the lb-type field.
    lb-type: nlb
    cloud-nat: yes
    forwarding-rule-port: 80
```

```
# Only one app is allowed
    urlPath-namedPort-maps:
    - appName: app1
```

```
# ssh key PUBLIC:
    - optional
```

The autoscaling firewall template requires you to enter the value in single quotes and prepend the key with **`admin:`** followed by a space. This is the same convention used for the Google Marketplace template, as detailed in SSH Key Pair. For example:



```
    sshkey: 'admin: ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDe0gJHd8okxPGWXsmdTdcZBJNI6ONT/NSz6INs2CNtKW
[redacted]
oTKXL8t0SRnOKaKV73NR5KnfpsNfwGxG8aQtEkeMCZIxX+6WOnRf/N4K3
yourname@[redacted].paloaltonetworks.local'
```

```
    bootstrap-bucket: bootstrap-autoscale
```

```
    image: vmseries-byol-814
    machine-type: n1-standard-4
```

For the service-account, supply the email address for the host project service account you created earlier (step 3).

```
     service-account: sa-pan@gcp-autoscale-kk.iam.gserviceaccount.com
```

The fw-instance-tag value will be the managed instance group name in the deployment.

```
     fw-instance-tag: vm-series-fw
```

Choose one metric for auto scaling. Possible values are: panSessionActive, panSessionUtilization, DataPlaneCPUUtilizationPct, DataPlanePacketBufferUtilization, or panSessionUtilization.

```
metric: custom.googleapis.com/VMSeries/panSessionActive
```

```
max-size: 2
min-size: 1
target-type: GAUGE
util-target: 100
```

```
# Greenfield deployment
mgmt-network-cidr: 172.22.2.0/24
untrust-network-cidr: 172.22.1.0/24
trust-network-cidr: 172.22.3.0/24
mgmt-network-access-source-range:
- 199.167.54.229/32
- 199.167.52.5/32
mgmt-network-access-ports:
- 22
- 443
```

STEP 2 | Deploy the firewall template.

```
gcloud deployment-manager deployments create <your-template>
```

```
--config apps.yaml
--automatic-rollback-on-error
```

Take note of the outputs the CLI prints after the deployment—the subnet names, the deployment name, and the Panorama Pub/Sub topic name. You need these values to configure the Shared VPC and for the application template deployment.

The firewall deployment name must be configured in the Panorama plugin for GCP auto scaling definition.

Prepare a Service Project

Create a separate service project, or choose an existing project, for your application.

To learn more about host and service projects in a shared VPC, see the Shared VPC Overview, and review the Administrators and IAM roles. A host project administrator must have the proper role to set up the Shared VPC and make the application project a service project for the host project. See the instructions in Provisioning Shared VPC.

STEP 1 | Make the application project a service project for the host project.

Add the service account from Service/application project administrator as a member in host project with following roles:

- Compute Network User
- Pub/Sub Admin

STEP 2 | Choose a VPC configuration.

- If Service project will share the networks in the host project, continue to Configure the Shared VPC.
- If the Service project has its own VPC network for the application deployment, continue to Configure a Peered VPC.

Configure the Shared VPC

After the firewall template is deployed in the host project, configure the service project that supports your applications. An administrator with shared VPC credentials performs these tasks from the host project. To understand more about the host project and service projects in the context of shared VPC, see the Shared VPC Overview.

STEP 1 | Create a shared VPC using the Trust VPC created when you deployed the firewall template.

Set up a shared VPC for the host (firewall) project:

```
gcloud compute shared-vpc enable HOST_PROJECT_ID
```

STEP 2 | Attach the service/application project to the host project.

```
gcloud compute shared-vpc associated-projects add [SERVICE_PROJECT_ID]--
host-project [HOST_PROJECT_ID]
```

Additional options are available to share only specific subnets, rather than all subnets in the host project.

STEP 3 | Continue to Deploy the Application Template.

Configure a Peered VPC

A VPC network peering connection must be made between two VPCs. If the VPCs are in two different projects, a connection must be created in **both** projects.

STEP 1 | In the host project, peer the Trust VPC network of the Firewall deployment with the Application VPC.

```
gcloud beta compute networks peerings create [PEERING-NAME] \
    --network=[MY-LOCAL-NETWORK] \
    --peer-project [SERVICE-PROJECT-ID] \
    --peer-network [PEER-NETWORK-NAME] \
   [--import-custom-routes] \
   [--export-custom-routes]
```

STEP 2 | In the service project, peer the Trust VPC network of the application deployment with the Trust VPC network of the Firewall deployment.

```
gcloud beta compute networks peerings create [PEERING-NAME] \
    --network=[MY-LOCAL-NETWORK] \
    --peer-project [HOST-PROJECT-ID] \
    --peer-network [PEER-NETWORK-NAME] \
    [--import-custom-routes] \
    [--export-custom-routes]
```

STEP 3 | Continue to Deploy the Application Template.

Deploy the Application Template

The Service project administrator deploys the Application Template from the service project.

STEP 1 | Create a separate application project (service project) to deploy the application (see Prepare a Service Project).

STEP 2 | Prepare the `apps.yaml` file as outlined in apps.yaml.

STEP 3 | Deploy a new application with the application template and define a label for the named port.

```
gcloud deployment-manager deployments create <your-template>
--config apps.yaml
--automatic-rollback-on-error
```

Continue to Onboard a New Application.

Onboard a New Application

When you use the application template to deploy an application, it takes care of the connection to the host project. You can secure applications you did not deploy with the application template, provided they are deployed in a service project with the capabilities described in Prepare a Service Project.

- Manually Onboard an Application to an Existing Auto Scaling Deployment
- Onboard a GKE Cluster

Manually Onboard an Application to an Existing Auto Scaling Deployment

To secure an application you have deployed using an external load balancer and an auto-scaled VM-Series firewall deployment, follow these steps. For each application you onboard, you must supply the application name, the named ports, and the path.

STEP 1 | Prepare to add a new named port and URL path to the HTTP external load balancer created when you deployed the firewall template.

STEP 2 | Update all instance groups named-ports with an additional service name and port values. The following sample onboards the applications **app2** and **app3**.

```
gcloud compute instance-groups set-named-ports
fw-template2-fw-igm-us-east1-b
--zone us-east1-b
--named-ports=app1:80,app2:81,app3:82

gcloud compute instance-groups set-named-ports
fw-template2-fw-igm-us-east1-c
--zone us-east1-c
--named-ports=app1:80,app2:81,app3:82
```

STEP 3 | Create a new http-health-check.

```
gcloud compute backend-services create fw-template2-backend-app3
--protocol="HTTP"
--port-name=app3
--http-health-checks=fw-template2-healthcheck-app3
--load-balancing-scheme="EXTERNAL"
--global
```

STEP 4 | Create a new backend service with the port-name created earlier on the HTTP external load balancer.

```
gcloud compute backend-services create fw-template2-backend-app3
--protocol="HTTP" --port-name=app3
--http-health-checks=fw-template2-healthcheck-app3 --load-balancing-
scheme="EXTERNAL"
--global
```

Check to see if the new backend service is visible.

```
gcloud compute backend-services list
```

STEP 5 | Edit url-maps and add new path rule. For example:

```
- paths:
    - /app3
    - /app3/*service:
    https://www.googleapis.com/compute/v1/projects/<project-name>/global/
backendServices/fw-template2-backend-app3
```

```
gcloud compute url-maps edit fw-template2-ext-loadbalancer
```

STEP 6 | To secure this application with the VM-Series firewall, manually trigger the pub/sub message through the gcloud CLI. This sends a message to the topic created in the firewall template.

```
gcloud pubsub topics publish
  projects/topics/hj-asg-891ca3-gcp-pavmqa-panorama-apps-deployment
  --attribute ilb-ip=172.22.9.34,
    app-deployment-name=hj-asg-891ca3-app1,
    ilb-port=80,
    named-port=81,
    network-cidr=172.22.9.0/24,
    fw-deployment-name=hj-asg-891ca3,
    host-project=gcp-pavmqa,
    type=ADD-APP
  --message "ADD-APP"
```

STEP 7 | View the Onboarded Application in the Panorama Plugin for GCP.

STEP 8 | To update application attributes, such as ilb-ip, ilb-port, or named-port, issue the pubsub command:

```
gcloud pubsub topics publish projects/gcp-pavmqa/topics/hj-asg-891ca3-gcp-
pavmqa-panorama-apps-deployment
  --attribute ilb-ip=172.22.9.34,
    app-deployment-name=hj-asg-891ca3-app1,
    ilb-port=80,
    named-port=81,
    network-cidr=172.22.9.0/24,
    fw-deployment-name=hj-asg-891ca3,
    host-project=gcp-pavmqa,
    type=UPDATE-APP
  --message "UPDATE-APP"
```

STEP 9 | (Optional) To stop securing the application, issue the following command:

```
gcloud pubsub topics publish projects/gcp-pavmqa/topics/hj-asg-891ca3-gcp-
pavmqa-panorama-apps-deployment
  --attribute ilb-ip=172.22.3.20,app-deployment-name=fw-templ-3-app-1,
    ilb-port=80,
    named-port=80,
    fw-deployment-name=hj-asg-891ca3,
    type=DEL-APP
  --message "DEL-APP"
```

Onboard a GKE Cluster

To onboard a private GKE cluster, the GCP plugin for Panorama requires the following information.

- In GCP, expose the ELB frontend for the cluster to the GKE service so the VM-Series firewall can get the named port information for the service.
- The cluster API server address.
- The service account credential for the service in which the cluster is deployed, in json format.

The GKE cluster name must not exceed 24 characters. This ensures that if you deploy auto scaling in a peered VPC configuration the static route name does not exceed 31 characters.

- Onboard a GKE Cluster in a Shared VPC
- Onboard a GKE Cluster in a Peered VPC
- View the Onboarded Application in the Panorama Plugin for GCP
- View the Deployment Status from the CLI

Onboard a GKE Cluster in a Shared VPC

To onboard the GKE cluster you must share the Host project Trust network VPC with the Service project. See Configure the Shared VPC.

For security reasons, only private clusters should be used in an auto scaling deployment. See Creating a private cluster.

STEP 1 | Set the Host project ID.

```
gcloud config set project [HOST_PROJECT_ID]
```

STEP 2 | (optional) Set compute zone or region for clusters.

If the cluster is zonal, enter the following:

```
gcloud config set compute/zone [COMPUTE_ZONE]
```

If the cluster is regional, enter the following:

```
gcloud config set compute/region [COMPUTE_REGION]
```

STEP 3 | In the Host project, update secondary ranges in the Trust VPC subnet.

```
gcloud compute networks subnets update [TRUST_SUBNETWORK_NAME]
  --add-secondary-ranges
  [PODS_IP_RANGE_NAME] = [POD_RANGE_CIDR],
  [SERVICE_IP_RANGE_NAME]=[SERVICE_RANGE_CIDR]
```

Pods and service IP ranges must be within: 10.0.0.0/8, 172.16.0.0/12 or 192.168.0.0/16, and cannot collide with existing IP ranges in the subnetwork.

STEP 4 | In the Service project, create a private cluster in the shared VPC.

1. Set the Service project ID.

```
gcloud config set project [SERVicE_PROJECT_ID]
```

2. Create a private cluster in the shared VPC.

```
gcloud container clusters create  [CLUSTER_NAME]
  --project [SERVICE_PROJECT_ID]
  --zone=[ZONE_NAME]
  --enable-ip-alias
  --enable-private-nodes
```

```
--network projects/[HOST_PROJECT_ID]/global/networks/[NETWORK_NAME]
--subnetwork projects/[HOST_PROJECT_ID]/regions/[REGION_NAME]
/subnetworks/[TRUST_SUBNETWORK_NAME]
--cluster-secondary-range-name=[PODS_IP_RANGE_NAME]
--services-secondary-range-name=[SERVICE_IP_RANGE_NAME]
--master-ipv4-cidr=[MASTER_IPV4_CIDR]
--enable-master-authorized-networks
--master-authorized-networks=[PANORAMA_MANAGEMENT_IP/32],
[MY_MANAGEMENT_IP/32]
```

STEP 5 | Check your current cluster context:

```
kubectl config current-context
```

STEP 6 | Check all cluster contexts.

```
kubectl config get-context
```

STEP 7 | Change to another cluster.

```
kubectl config use-context [CONTEXT_NAME]
```

If you created your cluster in the GCP console, generate a kubeconfig entry:

```
gcloud container clusters get-credentials [CLUSTER_NAME]
```

STEP 8 | Create a cluster role in a **.yaml** file—for example, **gke_cluster_role.yaml**.

```
apiVersion: rbac.authorization.k8s.io/v1beta1
kind: ClusterRole
metadata:
  name: gke-plugin-role
rules:
  - apiGroups:
      - ""
    resources:
      - services
    verbs:
      - list
```

STEP 9 | Apply the cluster role.

```
kubectl apply -f gke_cluster_role.yaml
```

STEP 10 | Create a cluster role binding in a **.yaml** file—for example, **gke_cluster_role_binding.yaml**.

```
kind: ClusterRoleBinding
apiVersion: rbac.authorization.k8s.io/v1beta1
metadata:
  name: gke-plugin-role-binding
```

```
subjects:
  - kind: ServiceAccount
    name: [SERVICEACCOUNT_NAME]
    namespace: default
roleRef:
  kind: ClusterRole
  name: gke-plugin-role
  apiGroup: rbac.authorization.k8s.io
```

STEP 11 | Apply the cluster role binding.

```
kubectl apply -f gke_cluster_role_binding.yaml
```

STEP 12 | Create a service account.

```
kubectl create serviceaccount [SERVICEACCOUNT_NAME]
```

STEP 13 | Export the service account secret token in JSON format.

```
MY_TOKEN=`kubectl get serviceaccounts [SERVICEACCOUNT_NAME]
-o jsonpath='{.secrets[0].name}'`

kubectl get secret $MY_TOKEN -o json > [FILE_NAME].json
```

STEP 14 | Get the API server address.

```
kubectl config view --minify | grep server | cut -f 2- -d ":" | tr -d " "
```

STEP 15 | In the Panorama plugin for GCP, add the service account information.

Select **Panorama** > **Google Cloud Platform** > **Setup**.

Name the credential, enter a description, and enter the **API server address** from step 14, and for **GKE Service Account Credential** upload the JSON file from step 13.

STEP 16 | Set up auto scaling on the Panorama plugin for GCP.

1. In the Panorama context, expand Google Cloud Platform, select AutoScaling, and click **Add**.
2. Supply the **Firewall Deployment Name** and an optional description for the deployment.
3. For the **GCP Service Account Credential**, supply the GCP service account name from Step 15.
4. Chose the Device Group and the Template Stack you created in when you configured the Panorama plugin.
5. Disable **License Management Only** to ensure traffic is secured.
6. Enter the exact **GKE Cluster Name**.
7. (Optional) Enter a **Description** of the GKE cluster.
8. Enter the **Network CIDR** for the GKE cluster.
9. Select the **GKE Service Account** corresponding to the GKE cluster.

STEP 17 | Commit your changes.

STEP 18 | (optional) Deploy the sample template according to Using the Sample GKE Service Templates. To see the results, see View the Onboarded Application in the Panorama Plugin for GCP.

Onboard a GKE Cluster in a Peered VPC

To onboard the GKE cluster you must create and peer the Service VPC with the firewall Trust network in the Host project, as described in Configure a Peered VPC.

For security reasons, only private clusters should be used in an auto scaling deployment. See Creating a private cluster.

STEP 1 | Set the project ID.

```
gcloud config set project [PROJECT_ID]
```

STEP 2 | Set compute zone or region for clusters.

If the cluster is zonal, enter the following:

```
gcloud config set compute/zone [COMPUTE_ZONE]
```

If the cluster is regional, enter the following:

```
gcloud config set compute/region [COMPUTE_REGION]
```

STEP 3 | Update the service project VPC network with the secondary IP ranges for the pods and services.

```
gcloud compute networks subnets update [GKE_PEERED_VPC_SUBNETWORK]
 --region=[REGION]
--add-secondary-ranges PODS_IP_RANGE_NAME=[ip cidr],
  SERVICE_IP_RANGE_NAME=[ip cidr]
```

STEP 4 | Enable cloud NAT.

Cloud NAT is required to deploy a private cluster.

```
gcloud compute routers create [ROUTER_NAME]
  --network [NETWORK_NAME]
  --region [REGION_NAME]
```

```
gcloud compute routers nats create [NAT_CONFIG_NAME]
  --router-region [REGION_NAME]
  --router [ROUTER_NAME]
  --nat-all-subnet-ip-ranges
  --auto-allocate-nat-external-ip
```

STEP 5 | If you plan to use an existing network for the Service VPC, you can skip this step. To create a new private cluster in the Service VPC.

```
gcloud container clusters create [CLUSTER_NAME]
  --project [SERVICE_PROJECT_ID]
  --zone=[ZONE_NAME]
  --enable-ip-alias
  --network [NETWORK_NAME]
  --subnetwork [SUBNETWORK_NAME]
  --enable-private-nodes
  --cluster-secondary-range-name=[PODS_IP_RANGE_NAME]   --services-
secondary-range-name=[SERVICE_IP_RANGE_NAME]   --master-ipv4-
cidr=[MASTER_IPV4_CIDR]
  --enable-master-authorized-networks   --master-authorized-
networks=[PANORAMA_MANAGEMENT_IP/32],
  [MY_MANAGEMENT_IP/32]
```

STEP 6 | Check your current cluster context:

```
kubectl config current-context
```

STEP 7 | Check all cluster contexts.

```
kubectl config get-context
```

STEP 8 | Change to another cluster.

```
kubectl config use-context [CONTEXT_NAME]
```

If you created your cluster in the GCP console, generate a kubeconfig entry:

```
gcloud container clusters get-credentials [CLUSTER_NAME]
```

STEP 9 | Create a cluster role in a **`.yaml`** file—for example, **`gke_cluster_role.yaml`**.

```
apiVersion: rbac.authorization.k8s.io/v1beta1
kind: ClusterRole
metadata:
  name: gke-plugin-role
rules:
  - apiGroups:
      - ""
    resources:
      - services
    verbs:
      - list
```

STEP 10 | Apply the cluster role.

```
kubectl apply -f gke_cluster_role.yaml
```

STEP 11 | Create a cluster role binding in a **`.yaml`** file—for example, **`gke_cluster_role_binding.yaml`**.

```
kind: ClusterRoleBinding
apiVersion: rbac.authorization.k8s.io/v1beta1
```

```
metadata:
  name: gke-plugin-role-binding
subjects:
  - kind: ServiceAccount
    name: [SERVICEACCOUNT_NAME]
    namespace: default
roleRef:
  kind: ClusterRole
  name: gke-plugin-role
  apiGroup: rbac.authorization.k8s.io
```

STEP 12 | Apply the cluster role binding.

```
kubectl apply -f gke_cluster_role_binding.yaml
```

STEP 13 | Create a service account.

```
kubectl create serviceaccount [SERVICEACCOUNT_NAME]
```

STEP 14 | Export the service account secret token in JSON format.

```
MY_TOKEN=`kubectl get serviceaccounts [SERVICEACCOUNT_NAME]
-o jsonpath='{.secrets[0].name}'`

kubectl get secret $MY_TOKEN -o json >[FILE_NAME].json
```

STEP 15 | Get the API server address.

```
kubectl config view --minify | grep server | cut -f 2- -d ":" | tr -d " "
```

STEP 16 | In the Panorama plugin for GCP, add the service account information.

Select **Panorama** > **Google Cloud Platform** > **Setup**.

Name the credential and enter the **API server address** from Step 15, and upload the JSON file you exported in Step14.

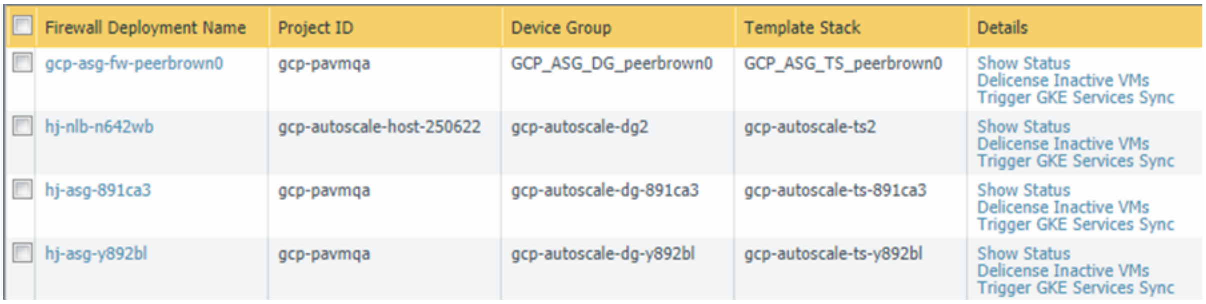
STEP 17 | Set up auto scaling on the Panorama plugin for GCP.

1. In the Panorama context, expand Google Cloud Platform, select AutoScaling, and click **Add**.
2. Supply the **Firewall Deployment Name** and an optional description for the deployment.
3. For the **GCP Service Account Credential**, supply the GCP service account name from Step 15.
4. Chose the Device Group and the Template Stack you created in when you configured the Panorama plugin.
5. Disable **License Management Only** to ensure traffic is secured.
6. Enter the exact **GKE Cluster Name**.
7. (Optional) Enter a **Description** of the GKE cluster.
8. Enter the **Network CIDR** for the GKE cluster.
9. Select the **GKE Service Account** corresponding to the GKE cluster.

STEP 18 | (optional) Deploy the sample template according to Using the Sample GKE Service Templates, or deploy your own services. To see the results, see View the Onboarded Application in the Panorama Plugin for GCP.

View the Onboarded Application in the Panorama Plugin for GCP

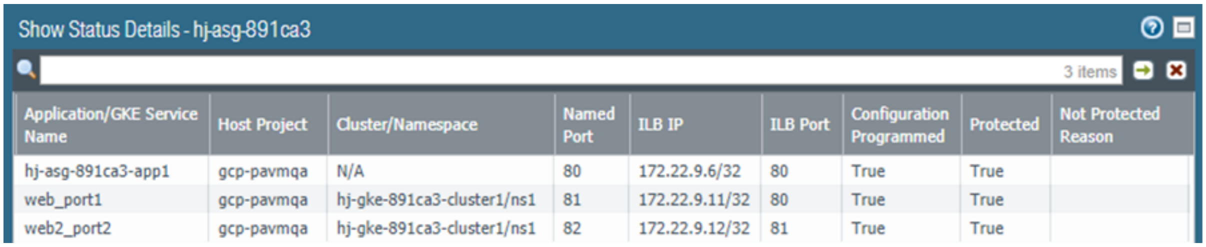
Select **Panorama** > **Google Cloud Platform** > **Autoscaling** to view your onboarded application. The **Details** column is only visible if you have an onboarded application.



□	Firewall Deployment Name	Project ID	Device Group	Template Stack	Details
□	gcp-asg-fw-peerbrown0	gcp-pavmqa	GCP_ASG_DG_peerbrown0	GCP_ASG_TS_peerbrown0	Show Status Delicense Inactive VMs Trigger GKE Services Sync
□	hj-nlb-n642wb	gcp-autoscale-host-250622	gcp-autoscale-dg2	gcp-autoscale-ts2	Show Status Delicense Inactive VMs Trigger GKE Services Sync
□	hj-asg-891ca3	gcp-pavmqa	gcp-autoscale-dg-891ca3	gcp-autoscale-ts-891ca3	Show Status Delicense Inactive VMs Trigger GKE Services Sync
□	hj-asg-y892bl	gcp-pavmqa	gcp-autoscale-dg-y892bl	gcp-autoscale-ts-y892bl	Show Status Delicense Inactive VMs Trigger GKE Services Sync

Each link in the Details column triggers an action.

- **Show Status**— View the details for applications onboarded to a GCP VM-Series firewall deployment.



Show Status Details - hj-asg-891ca3

3 items

Application/GKE Service Name	Host Project	Cluster/Namespace	Named Port	ILB IP	ILB Port	Configuration Programmed	Protected	Not Protected Reason
hj-asg-891ca3-app1	gcp-pavmqa	N/A	80	172.22.9.6/32	80	True	True	
web_port1	gcp-pavmqa	hj-gke-891ca3-cluster1/ns1	81	172.22.9.11/32	80	True	True	
web2_port2	gcp-pavmqa	hj-gke-891ca3-cluster1/ns1	82	172.22.9.12/32	81	True	True	

The following fields display information obtained from the selected deployment. You specified these values in the pub/sub message or through GKE cluster service polling.

- **Application/GKE Service Name**—An application deployment name, or the name of a GKE service.
- **Host Project**—The name of the host project.
- **Cluster/Namespace**—A GKE cluster name followed by the namespace for example, **`mycluster/namespace9`**.
- **Named Port**—The port assigned to the named port for the service.
- **ILB IP**—The ILB IP address.
- **ILB Port**—The ILB port number.

  For autoscaling an application, this property is **`ilb-port`** in `apps.yaml`.

  For securing a GKE cluster, this value is the port number of the GKE cluster, as specified in the `.yaml` file you used to deploy the service in your cluster.
- **Configuration Programmed**— True if a NAT Rule exists, False if not.
- **Protected**— True when an application is onboarded successfully, or False if onboarding failed. If False, see the **Not Protected Reason** column for an explanation.
- **Not Protected Reason— If Protected** is False, displays the reason the application is not protected. Some common reasons are:
  - **Configuration Programmed**—True if a NAT Rule exists, False if not.
  - **Protected**—True when an application is onboarded successfully, or False if onboarding failed. If False, see the **Not Protected Reason** column for an explanation.
  - **Not Protected Reason**—If **Protected** is False, displays the reason the application is not protected. Some common reasons are:
    - You deployed a UDP service in the GKE cluster.
    - You specified a named port that is already in use. Only one application can listen on a specific named port.
    - You chose the **License management only** option, so we do not program the configuration.

    - No matching label found for GKE services.
- **Delicense Inactive VMs**—Answer **Yes** to trigger the delicensing function for inactive VMs.
- **Trigger GKE Services Sync**—Answer **Yes** to poll the services running in the clusters, and program the NAT, address, and service objects, and static routes if necessary. By default, Panorama automatically polls 10 minutes after the completion of the previous poll.

View the Deployment Status from the CLI

After you have deployed a service you can use the Panorama CLI to check the deployment. The command line results are the same those mentioned in View the Onboarded Application in the Panorama Plugin for GCP. The `autoscaling_name` is the Firewall Deployment Name you entered in the auto scaling configuration.

```
show plugins gcp show-protected-apps autoscaling_name <auto-scale-name>
```

Parameters in the Auto Scaling Templates for GCP

You can download the template `.zip` file from https://github.com/PaloAltoNetworks/GCP-AutoScaling. The `.zip` file contains directories to support firewall templates for network load balancer and application load balancer configurations, and the application template.

The template YAML files have the following general format:

```
#Copyright and license information
    :
    :
imports:                     <do not change>
    :
    :
resources:
    -name: vm-series-fw      <do not change>
    -type:vm-series-fw.py    <do not change
    -properties:
    :
    :
outputs:                     <do not change>
    :
    :
```

In all `.yaml` files, you customize the `resources` properties for your deployment. Do not change anything in the `imports` or `outputs` sections.

- Firewall Templates
- Application Template

Firewall Templates

The following sections detail the parameters for the NLB and ALB `.yaml` files.

- vm-series-fw-nlb.yaml
- vm-series-fw-alb.yaml

vm-series-fw-nlb.yaml

In the `vm-series-fw-nlb.yaml` template, edit the `-properties`.

Parameter	Sample Value	Comment
region	us-central1	https://cloud.google.com/compute/docs/regions-zones
zones - <list of zones>	zones- us-central1-a	If applicable, list multiple zones as follows: zones- us-central1-a- us-central1-b- us-central1-c- us-central1-f
lb-type	nlb	Do not change.
cloud-nat	yes	Do not change.
forwarding-rule-port	80	80 or 8080
urlPath-namedPort-maps-appname	urlPath-namedPort-maps-MyApplication	Enter your application name.
sshkey	'admin:ssh-rsa <PASTE KEY>'	Review SSH Key Pair. In single quotes, type **admin:** followed by a space, and paste in your key. This is the same convention used for the Google Marketplace template.
bootstrap-bucket	bootstrap-autoscale	The name of the GCP bucket that contains your bootstrap file.
image	vm-series-byol-814	The BYOL image currently available from the Google marketplace. If you are using PAYG or another license model, the image might be different.
machine-type	n1-standard-4	n1-standard-4 is default for BYOL. If your license permits it, you can use any machine type in Minimum System Requirements for the VM-Series Firewall on GCP.
service-account		The unique service account name for the host project.
fw-instance-tag	vm-series-fw	The instance tag you provided in GCP.

Parameter	Sample Value	Comment

metric	custom.googleapis.com/VMSeries/panSessionActive	The custom API path for VM-Series, and your chosen auto scaling metric.
		Supply only one of the following metrics.
		`panSessionActive`
		`panSessionUtilization`
		`DataPlaneCPUUtilizationPct`
		`DataPlanePacketBufferUtilization`
		`panSessionUtilization`

max-size	2	
min-size	1	
target-type	GAUGE	Currently GAUGE is the only valid type.
util-target	100	
To deploy the VM-Series firewall you need a dedicated network and subnetwork for the firewall's managment, untrust, and trust interfaces. Fill out the information for either a greenfield deployment (configure the template to create new networks) or brownfield deployment (use existing networks). Be sure to remove or comment out the network deployment parameters you are not using.		
Greenfield Deployment Enter values to create management, untrust, and trust networks and subnetworks for the firewall.		
mgmt-network-cidr	172.22.2.0/24	
untrust-network-cidr	172.22.1.0/24	
trust-network-cidr	172.22.3.0/24	

mgmt-network-access-source-range- <permitted-ip-range>	`mgmt-network-access-source-range`
	`- <permitted-ip-range-1>`
	`- <permitted-ip-range-2>`	

mgmt-network-access-ports-<port-number>	`mgmt-network-access-ports`
	`- 22`
	`- 443`	

Brownfield Deployment Enter the name of each existing network or subnetwork		
mgmt-network	my-mgmt-network	

Parameter	Sample Value	Comment
mgmt-subnet	my-mgmt-subnet	
trust-network	my-trust-network	
trust-subnet	my-trust-subnet	
untrust-network	my-untrust-network	
untrust-subnet	my-untrust-subnet	

vm-series-fw-alb.yaml

In the `vm-series-fw-alb.yaml` template, edit the `-properties`.

Parameter	Sample Value	Comment
region	us-central1	https://cloud.google.com/compute/docs/regions-zones
zones - <list of zones>	zones- us-central1-a	If applicable, list multiple zones as follows: zones- us-central1-a- us-central1-b- us-central1-c- us-central1-f
lb-type	alb	Do not change.
cloud-nat	yes	Do not change.
forwarding-rule-port	80	80
connection-draining-timeout	300	The timeout value in seconds.

`urlPath-namedPort-maps:`
`- appname  namedPort:`
`  urlMapPaths:`
`  - '/app1'`
`  - '/app1/*'`
	`urlPath-namedPort-`
	`maps:`
	`- appName: app1`
	`  namedPort: 80`
	`   urlMapPaths:`
	`  - '/app1'`
	`  - '/app1/*'`
	`- appName: app2`
	`  namedPort: 81`
	` urlMapPaths:`
	`  - '/app2'`
	`  - '/app2/*'`
		List your apps and the corresponding named port

sshkey	'admin:ssh-rsa <PASTE KEY>'	Review SSH Key Pair. In single quotes, type **`admin:`** followed by a space, and paste in your key. This is the same convention

Parameter	Sample Value	Comment
		used for the Google Marketplace template.
bootstrap-bucket	bootstrap-bucket-name	The name of the GCP bucket that contains your bootstrap file.
image	vm-series-byol-814	The BYOL image currently available from the Google marketplace. If you are using PAYG or another license model, the image might be different
machine-type	n1-standard-4	n1-standard-4 is default for BYOL. If your license permits it, you can use any machine type in Minimum System Requirements for the VM-Series Firewall on GCP.
service-account	The unique service account name for the service project.	
fw-instance-tag	vm-series-fw	The instance tag you provided in GCP.

metric	custom.googleapis.com/VMSeries/panSessionActive	The custom API path for VM-Series, and your chosen auto scaling metric.
		Supply only one of the following metrics.
		`panSessionActive`
		`panSessionUtilization`
		`DataPlaneCPUUtilizationPct`
		`DataPlanePacketBufferUtilization`
		`panSessionUtilization`

max-size	2	
min-size	1	
target-type	GAUGE	Currently GAUGE is the only valid type.
util-target	100	Enter the goal utilization target value for the auto scaling.
Greenfield Deployment Enter values to create management, untrust, and trust networks and subnetworks for the firewall.		
mgmt-network-cidr	192.168.12.0/24	

Parameter	Sample Value	Comment
untrust-network-cidr	192.168.11.0/24	
trust-network-cidr	192.168.11.0/24	
mgmt-network-access-source-range- <permitted-ip-range>	`mgmt-network-access-source-range- <permitted-ip-range-1>- <permitted-ip-range-2>`	
mgmt-network-access-ports- <port-number>	`mgmt-network-access-ports- 22- 443`	
Brownfield Deployment Enter the name of each existing network or subnetwork		
mgmt-network	existing-vpc-mgmt	
mgmt-subnet	existing-subnet-mgmt	
trust-network	existing-vpc-trust	
trust-subnet	existing-subnet-trust	
untrust-network	existing-vpc-untrust	
untrust-subnet	existing-subnet-untrust	

Application Template

apps.yaml

The application template creates the connection between the host project (which contains the VM-Series firewalls) and the service project, which contains the application or services that the firewall deployment secures.

Parameter	Sample Value	Comment
host-project	your-host-project-name	The name of the project containing the VM-Series firewall deployment.
fw-deployment-name	my-vm-series-firewall-name	
region	us-central1	https://cloud.google.com/compute/docs/regions-zones

zones
- <list of zones>
	zones- us-central1-a	If applicable, list multiple zones as follows:
		`zones- us-central1-a-`
		`us-central1-b-`
		`us-central1-c-`
		`us-central1-f`

Parameter	Sample Value	Comment
app-machine-type	n1-standard-2	The machine type for the VM running your application or service. If your license permits it, you can use any machine type in Minimum System Requirements for the VM-Series Firewall on GCP.
app-instance-tag	web-app-vm	You applied this tag (label) in GCP.
sshkey	'admin:ssh-rsa <PASTE KEY>'	Review SSH Key Pair. In single quotes, type **admin:** followed by a space, and paste in your key. This is the same convention used for the Google Marketplace template.
trust-network	<project-name>/<vpc-network-name>	For a shared VPC, the <project-name> is the host project name. For peered VPCs the <project-name> is the Service project name.
trust-subnet	<project-name>/<subnet-name>	For a shared VPC, the <project-name> is the host project name. For peered VPCs the <project-name> is the Service project name.
trust-subnet-cidr	10.2.0.0/24	For a greenfield deployment, the Host project Trust subnet CIDR (the trust-network-cidr parameter in the firewall template). For a brownfield deployment, the CIDR for the Trust network.
vm-series-fw-template-topic	<pubsub-topic>	Enter the topic name created by the firewall deployment. The application template posts a message to the topic to program the firewall configuration to forward traffic.
ilb-port	80	Enter the port number for your application's internal-load-balancer-port. output.

Parameter	Sample Value	Comment
urlPath-namedPort	83	Enter the port number for the urlPath-namedPort output.

Sample GKE Service Templates

These sample templates demonstrates how to configure a GKE service so it is secured by the VM-Series firewall. For the basics on creating your own cluster services, see Creating a private cluster.

- Using the Sample GKE Service Templates
- gke_cluster_role.yaml
- gke_cluster_role_binding.yaml
- web-deployment.yaml
- web-service.yaml
- web-deployment-v2.yaml
- web-service-v2.yaml
- Multiple Ports in a Service

Using the Sample GKE Service Templates

You can create a `.yaml` file with the content in the above .yaml files.

- To be secured by the VM-Series firewall, services in the cluster must be labeled "pavm-named-port=<named_port>" as shown in web-service.yaml or web-service-v2.yaml.
- Deploy the sample as follows:

```
kubectl apply -f [FILE_NAME].yaml
```

- Delete all services deployed in this sample:

```
kubectl delete -f [FILE_NAME].yaml
```

- Configure the VPC deployment.
  - In a shared VPC deployment, launch the GKE cluster in the shared VPC. See Configure the Shared VPC.
  - In a peered VPC deployment, peer the GKE cluster VPC to the host project Trust network. See Configure a Peered VPC.

gke_cluster_role.yaml

```
apiVersion: rbac.authorization.k8s.io/v1beta1
kind: ClusterRole
metadata:
  name: gke-plugin-role
rules:
  - apiGroups:
      - ""
    resources:
      - services
    verbs:
      - list
```

gke_cluster_role_binding.yaml

```
kind: ClusterRoleBinding
apiVersion: rbac.authorization.k8s.io/v1beta1
metadata:
  name: gke-plugin-role-binding
subjects:
  - kind: ServiceAccount
    name: hj-gke-891ca3-cluster1-sa
    namespace: default
roleRef:
  kind: ClusterRole
  name: gke-plugin-role
  apiGroup: rbac.authorization.k8s.io
```

web-deployment.yaml

```
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: web
  namespace: default
spec:
  selector:
   matchLabels:
    run: web
  template:
   metadata:
    labels:
      run: web
   spec:
    containers:
    - image: gcr.io/google-samples/hello-app:1.0
      imagePullPolicy: IfNotPresent
      name: web
      ports:
      - containerPort: 8080
       protocol:  TCP
```

web-service.yaml

```
apiVersion: v1
kind: Service
metadata:
  name: web
  namespace: default
  annotations:
    cloud.google.com/load-balancer-type: "Internal"
  labels:
    pavm-named-port-port1: "80"
spec:
  ports:
  # the port that this service should serve on
  - name: port1
    port: 80
    protocol: TCP
    targetPort: 8080
  selector:
    run: web
  type: LoadBalancer
```

web-deployment-v2.yaml

```
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: web2
  namespace: default
spec:
  selector:
    matchLabels:
      run: web2
  template:
    metadata:
      labels:
        run: web2
    spec:
      containers:
      - image: gcr.io/google-samples/hello-app:2.0
        imagePullPolicy: IfNotPresent
        name: web2
        ports:
        - containerPort: 8080
          protocol: TCP
```

web-service-v2.yaml

```
apiVersion: v1
  kind: Service
  metadata:
    name: web2
    namespace: default
    annotations:
      cloud.google.com/load-balancer-type: "Internal"
    labels:
      pavm-named-port-port2: "81"
  spec:
    ports:
    # the port that this service should serve on
    - name: port2
      port: 81
      protocol: TCP
      targetPort: 8080
    selector:
      run: web2
    type: LoadBalancer
```

Multiple Ports in a Service

For multiple ports in one service, edit labels and map the target port name and number in the format panw-named-port-<service-spec-port-name>, as shown below.

```
apiVersion: v1
kind: Service
metadata:
name: carts
annotations:
cloud.google.com/load-balancer-type: "Internal"
```

```
labels:
panw-named-port-carts-http: "6082"
panw-named-port-carts-https: "6083"
namespace: default
spec:
type: LoadBalancer
ports:
# the port that this service should serve on
- name: carts-http
protocol: TCP
port: 80
targetPort: 80
- name: carts-https
protocol: TCP
port: 443
targetPort: 443
selector:
name: carts
```