

# VM-Series with a Gateway Load Balancer CloudFormation Templates

---

## Contact Information

Corporate Headquarters:

Palo Alto Networks

3000 Tannery Way

Santa Clara, CA 95054

[www.paloaltonetworks.com/company/contact-support](http://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](https://docs.paloaltonetworks.com).
- To search for a specific topic, go to our search page: [docs.paloaltonetworks.com/search.html](https://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](mailto:documentation@paloaltonetworks.com).

## Copyright

Palo Alto Networks, Inc.

[www.paloaltonetworks.com](http://www.paloaltonetworks.com)

© 2020-2020 Palo Alto Networks, Inc. Palo Alto Networks is a registered trademark of Palo Alto Networks. A list of our trademarks can be found at [www.paloaltonetworks.com/company/trademarks.html](http://www.paloaltonetworks.com/company/trademarks.html). All other marks mentioned herein may be trademarks of their respective companies.

## Last Revised

November 9, 2020

---

# Table of Contents

**VM-Series Auto Scaling Group with an AWS Gateway Load Balancer.....5**

    VM-Series Auto Scaling Group with AWS Gateway Load Balancer..... 7

    Before Launching the Templates..... 10

    Launch the Firewall Template..... 12

    Launch the Application Template..... 15



# VM-Series Auto Scaling Group with an AWS Gateway Load Balancer


Use the following topics to launch a VM-Series firewall auto scaling group with an AWS gateway load balancer.

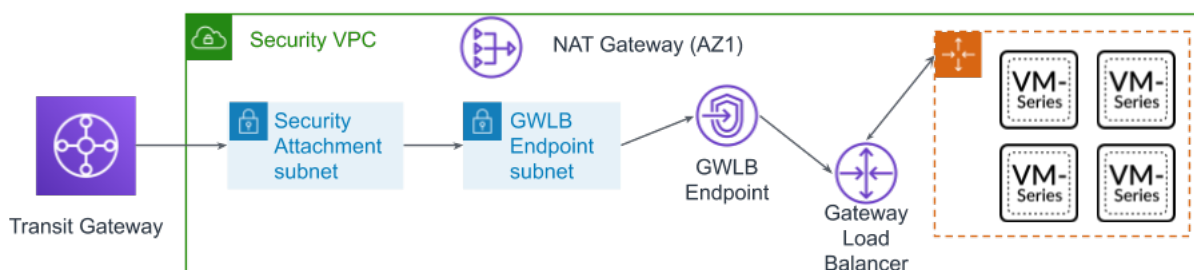
- > [VM-Series Auto Scaling Group with AWS Gateway Load Balancer](#)
- > [Before Launching the Templates](#)
- > [Launch the Firewall Template](#)
- > [Launch the Application Template](#)



# VM-Series Auto Scaling Group with AWS Gateway Load Balancer

The Palo Alto Networks auto scaling template for AWS help you integrate and configure the VM-Series firewall with a GWLB to protect applications deployed in AWS. The template leverage AWS scalability features to independently and automatically scale VM-Series firewalls deployed in AWS to meet surges in application workload resource demand.



 These templates are **community supported**.



This solution provides a security VPC template and an application template. The security VPC template deploys the VM-Series firewall auto scaling group, a GWLB, a GWLBE, GWLBE subnet, security attachment subnet, and a NAT gateway for each availability zone. Download the CloudFormation templates from the [Palo Alto Networks GitHub Repository](#).

The VM-Series Auto Scaling template for integration with an AWS GWLB includes the following building blocks:

Building Block	Description
PAN Components	<ul style="list-style-type: none"><li>• Panorama running 10.0.2 or later</li><li>• PAN-OS 10.0.2 or later</li><li>• VM-Series plugin 2.0.2 or later installed on Panorama</li></ul>
<b>Firewall template</b> (Community supported template)	<p>Based on the number of availability zones (AZs) you choose, the <code>firewall-new-vpc-v3.0.template</code> deploys the following:</p> <ul style="list-style-type: none"><li>• Subnets for Lambda management, transit gateway attachments, GWLB endpoints, and NAT gateways, as well as trust subnets.</li><li>• Routes tables for each subnet</li><li>• Transit gateway attachments and route tables</li><li>• NAT and internet gateways</li><li>• An auto scaling group with one VM-Series firewall per AZ.</li><li>• One GWLB and a GWLB endpoint in each AZ.</li></ul> <p>The VPC CIDR for the firewall template should be larger than /23.</p> <p>Due to the many variations in a production environment that includes but is not limited to a specific number components, such as subnets, availability zones, route tables, and security groups. You must deploy the <code>firewall-new-vpc-v3.0.template</code> in a new VPC.</p>

Building Block	Description
	 <i>VM-Series Auto Scaling template for AWS does not deploy a transit gateway or Panorama. You must deploy a transit gateway and Panorama before launching firewall-new-vpc-v3.0.template.</i>
<b>Application template</b>  (Community supported template)	<p>Based on the number of availability zones (AZs) you choose, the <code>panw-aws-app-v3.0.template</code> deploys the following:</p> <ul style="list-style-type: none"> <li>• Subnets for Lambda, transit gateway attachments, GWLB endpoints, application load balancers.</li> <li>• Routes tables for each subnet, as well as an inbound route table associated with the internet gateway to direct inbound traffic to the GWLB endpoint.</li> <li>• One application load balancer</li> <li>• One internet gateway</li> <li>• An auto scaling group with one Ubuntu instance per AZ.</li> </ul> <p>The VPC CIDR for the application template should be larger than /23.</p> <p> <i>The application template is intended to be used as an example for validating the security template.</i></p>
<b>Lambda functions</b>	<p>AWS Lambda provides robust, event-driven automation without the need for complex orchestration software. In addition to deploying the components described in the rows above, the <code>firewall-new-vpc-v3.0.template</code> performs the following functions:</p> <ul style="list-style-type: none"> <li>• Adds or removes an interface (ENI) when a firewall is launched or terminated.</li> <li>• Deletes all the associated resources when you delete a stack or terminate an instance.</li> <li>• Removes a firewall as a Panorama managed device when there is a scale-in event.</li> <li>• Deactivates the license when a scale-in event results in a firewall termination.</li> <li>• Monitors the transit gateway periodically for new attachments or detachments, and updates the route tables accordingly in the security VPC.</li> </ul>
<b>Bootstrap files</b>  The <code>bootstrap.xml</code> file provided in the GitHub repository is provided for testing and evaluation only. For a production deployment, you must modify the sample credentials in the <code>bootstrap.xml</code> prior to launch.	<p>This solution requires the <code>init-cfg.txt</code> file and the <code>bootstrap.xml</code> file so that the VM-Series firewall has the basic configuration for handling traffic.</p> <ul style="list-style-type: none"> <li>• The <code>init-cfg.txt</code> file includes the <code>mgmt-interface-swap</code> operational command to enable the firewall to receive dataplane traffic on its primary interface (<code>eth0</code>). This auto-scaling solution requires the swapping of the dataplane and management interfaces to enable the GWLB to forward web traffic to the auto-scaling tier of VM-Series firewalls.</li> <li>• The <code>bootstrap.xml</code> file enables basic connectivity for the firewall network interfaces and allows the firewall to connect to the AWS CloudWatch namespace that matches the stack name you enter when you launch the template.</li> </ul>



*If you need to delete these templates from AWS, always delete the application template first. Attempting to delete the firewall template causes the deletion to fail.*



- 
- [Before Launching the Templates](#)
  - [Launch the Firewall Template](#)
  - [Launch the Application Template](#)

---

# Before Launching the Templates

Before you launch the templates to integrate a VM-Series firewall auto scaling group with an AWS GWLB, you must complete the following procedure.

## STEP 1 | Ensure that you have the following before you begin.

- Obtain the auth code for a bundle that supports the number of firewalls that might be required for your deployment. You must save this auth code in a text file named `authcodes` (no extensions), and put the `authcodes` file in the `/license` folder of the bootstrap package.
- Download the files required to launch the [VM-Series Gateway Load Balancer](#) template from the GitHub repository.
- Create a [Transit Gateway](#). This transit gateway connects your security and application VPCs. Take note of the transit gateway ID; you will need it later when deploying the template.

Ensure that Default route table association and Default route table propagation are disabled.

- The recommended VPC CIDR for the firewall and application templates should be larger than `/23`.



*The target group of the gateway GWLB cannot use HTTP for health checks because the VM-Series firewall does not allow access with an unsecured protocol. Instead use HTTPS or TCP.*

## STEP 2 | Deploy Panorama running 10.0.2 and configure the following.

Panorama must allow AWS public IP addresses. The VM-Series firewall accesses Panorama using the external IP address of the NAT gateway created by the template.

## STEP 3 | Download and install the VM-Series plugin on Panorama.

1. Select Panorama > Plugins and use Check Now to look for new plugin packages. The VM-Series plugin name is `vm_series`.
2. Consult the plugin release notes to determine which version provides upgrades useful to you.
3. Select a version of the plugin and select Download in the Action column.
4. Click Install in the Action column. Panorama alerts you when the installation is complete.
5. To view the plugin, select Device > VM-Series.

## STEP 4 | Configure the template.

1. Log in to the Panorama web interface.
2. Select Panorama > Templates and click Add.
  1. Enter a descriptive Name.
  2. Click OK.
3. Configure the virtual router.
  1. Select Network > Virtual Routers.
  2. Ensure that you have selected the template you create above from the Template drop-down.
  3. Click Add.
  4. Name the virtual router using the following format: VR-<tempstackname>.
  5. Enable ECMP on the virtual router.
  6. Click OK.
4. Configure the interface and create the zone.
  1. Select Network > Interfaces and click Add Interface.
  2. Select Slot 1.

- 
3. Set Interface Type to Layer 3.
  4. On the Config tab, select New Zone from the Security Zone drop-down. In the Zone dialog, define a Name for new zone, for example Internet, and then click OK.
  5. In the Virtual Router drop-down, select virtual router you created above.
  6. Select IPv4 and click DHCP Client.
  7. Click OK.
5. Configure the DNS server and FQDN refresh time.
    1. Select Device > Setup > Services and click the Edit icon.
    2. Set the Primary DNS Server to 169.254.169.253. This is the AWS DNS address.
    3. Set the Minimum FQDN Refresh Time to 60 seconds.
    4. Click OK.
  6. Commit your changes. This is required before proceeding to the next step.
  7. Create an administrator.
    1. Select Device > Administrators.
    2. Enter **pandemo** as the Name.
    3. Set the Password to **demopassword** and Confirm.
    4. Click OK.
  8. Commit your changes.

#### STEP 5 | Create the Device Group.

1. Select Panorama > Device Groups.
2. Ensure that you have selected the device group you create above from the Device Group drop-down.
3. Click Add.
4. Enter a descriptive Name.
5. Click OK.
6. Add an allow all security pre-rule.
  1. Select Policies > Security > Pre Rules and click Add.
  2. Enter a descriptive Name.
  3. Under Source, User, Destination, Application, and Service/URL Category, select any.
  4. Under Actions, select Allow.
  5. Click OK.
7. Commit your changes.

#### STEP 6 | Add the license deactivation API key for the firewall to Panorama.

1. Log in to the Customer Support Portal.
2. Select Assets > Licensing API.
3. Copy the API key.
4. Use the CLI to install the API key copied in the previous step.

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

---

# Launch the Firewall Template

This workflow describes how to deploy the firewall template.

## STEP 1 | Modify the `init-cfg.txt` file and upload it to the `/config` folder.

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

Ensure that you use the device group and template names you created above in the `init-cfg.txt` file.

```
type=dhcp-client
ip-address=
default-gateway=
netmask=
ipv6-address=
ipv6-default-gateway=
hostname=
vm-auth-key=
panorama-server=
panorama-server-2=
tplname=
dgroup=
dhcp-send-hostname=yes
dhcp-send-client-id=yes
dhcp-accept-server-hostname=yesdhcp-accept-server-domain=yes
plugin-op-commands=aws-gwlb-inspect:enable
```

Your `init-cfg.txt` file must include `plugin-op-commands=aws-gwlb-inspect:enable`. This is required when integrating the VM-Series firewall with a GWLB.

You must add the device certificate auto-registration PIN to the `init-cfg.txt` file to automatically install a [device certificate](#) when your VM-Series firewall instance is deployed.

## STEP 2 | Add the license auth code in the `/license` folder of the bootstrap package.

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

## STEP 3 | Upload Lambda code for the firewall template (`panw-aws.zip`) and the Application template (`app.zip`) to an S3 bucket. You can use the same S3 bucket that you use for bootstrapping.

If the Application stack is managed by a different account than the firewall, use the Application account to create another S3 bucket in the same AWS region as the firewall template and copy `app.zip` to that S3 bucket.

## STEP 4 | Select the firewall template.

1. In the AWS Management Console, select CloudFormation > Create Stack.
2. Select Upload a template to Amazon S3, to choose the application template to deploy the resources that the template launches within the same VPC as the firewalls, or to a different VPC. Click Open and Next.

- 
3. Specify the Stack name. The stack name allows you to uniquely identify all the resources that are deployed using this template.

**STEP 5 | Enter a descriptive Name for your stack. The name must be 28 characters or less.**

**STEP 6 | Configure the parameters for the VPC.**

1. Enter the number of availability zones and select the region from the availability zone drop-down.
2. Look up the AMI ID for the VM-Series firewall and enter it. Make sure that the AMI ID matches the AWS region, PAN-OS version and the BYOL or PAYG licensing option you opted to use. See [Get the Amazon Machine Image IDs](#) for more information.
3. Select the EC2 Key pair (from the drop-down) for launching the firewall. To log in to the firewalls, you must provide the name of this key pair and the private key associated with it.
4. Select Yes if you want to Enable Debug Log. Enabling the debug log generates more verbose logs that help with troubleshooting issues with the deployment. These logs are generated using the stack name and are saved in AWS CloudWatch.

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

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

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

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

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

**STEP 9 | Add your AWS account number(s). You must provide the account number used to deploy any VPC that is connected to your GWLB. Add these values as a comma-separated list. You can add additional account numbers after deploying the template.**

To locate your account number, click your AWS username in the top right of the AWS console and select My Security Credentials.

**STEP 10 | Enter the maximum and minimum VM-Series firewalls for your auto scaling group.**

**STEP 11 | Enter the transit gateway ID. The transit gateway ID is required to secure east-west and outbound traffic. If you do not enter a transit gateway ID, the template assumes that only inbound traffic should be inspected by firewalls integrated with the GWLB.**

**STEP 12 | Enter the CIDR for the security VPC.**

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

1. Select I acknowledge that this template might cause AWS CloudFormation to create IAM resources.
2. Click Create to launch the template. The CREATE\_IN\_PROGRESS event displays.

- 
3. On successful deployment the status updates to CREATE\_COMPLETE.

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

---

# Launch the Application Template

Complete the following procedure to launch the application template.

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

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

## **STEP 2 | Upload the app.zip file into the S3 bucket.**

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

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

## **STEP 4 | Select the Availability Zones (AZ) that your setup will span in Select list of AZ.**

## **STEP 5 | Enter a descriptive VPC Name.**

## **STEP 6 | Configure the parameters for Lambda.**

1. Enter the S3 bucket name where app.zip is stored.
2. Enter the name of the zip file name.

## **STEP 7 | Select the EC2 instance type for the Ubuntu web server launched by this template.**

## **STEP 8 | Enter your Amazon EC2 key pair.**

## **STEP 9 | Enter the name of the service configuration (Service Name) for the GWLB endpoint in the security VPC.**

1. Select DynamoDB from the Services drop-down in the AWS console.
2. Select Tables and locate your security VPC table.
3. Click the Items tab and copy the Service Name.
4. Paste the Service Name into the template configuration parameters.

## **STEP 10 | Enter the transit gateway ID. This is the same transit gateway you created before deploying the firewall template.**

## **STEP 11 | Review the template settings and launch the template.**

## **STEP 12 | After the application has been deployed, you must add a route to the transit gateway route table to enable east-west and outbound traffic inspection.**

1. Log in to the AWS VPC console.
2. Select Transit Gateway Route Tables and choose your transit gateway route table. This route table is created by the template and is called <app-stack-name>-<region>-PANWAppAttRt.
3. Select Routes and click Create static route.
4. Enter 0.0.0.0/0 in the CIDR field.

- 
5. From the Choose attachment drop-down, select the VM-Series firewall VPC attachment.
  6. Click Create static route.

**STEP 13** |(Optional) **Create a bastion host (also called a jump box) to access the web server created by the application template.**

1. Create a public-facing [subnet](#) in your application VPC.
2. Add a route to this subnet from your IP address to the internet gateway.
3. Create a new EC2 instance in the public subnet with a public IP address.
4. Create a security group for this EC2 instance that allows SSH from your IP address.