

Palo Alto Networks: Securing Applications in AWS Transit Gateway via June 2020 Deployment Guide

Contents

About	3
Topology	3
Support Policy	4
Prerequisites	4
Panorama Connection Methods	5
Panorama Configuration.....	5
Modify Deployment Files.....	5
\setup_panorama\variables.tf	5
\setup_aws\variables.tf.....	6
\setup_aws\init-cfg._inbound_a.txt.....	6
\setup_aws\init-cfg._inbound_b.txt	6
\setup_aws\init-cfg._outbound_a.txt	6
\setup_aws\init-cfg._outbound_b.txt	6
Environment Information	6
Deploying the setup_panorama Template	7
Deploying the setup_aws Template.....	9
Verifying Deployment.....	13
Cleanup.....	13

About

This document explains how to deploy the environment shown in the document *aws-transit-gateway-deployment-guide.pdf* for the June 2020 deployment guide, Securing Applications in AWS Transit Gateway using VM-Series appliances and Panorama in AWS. This document can be found in the *docs* folder.

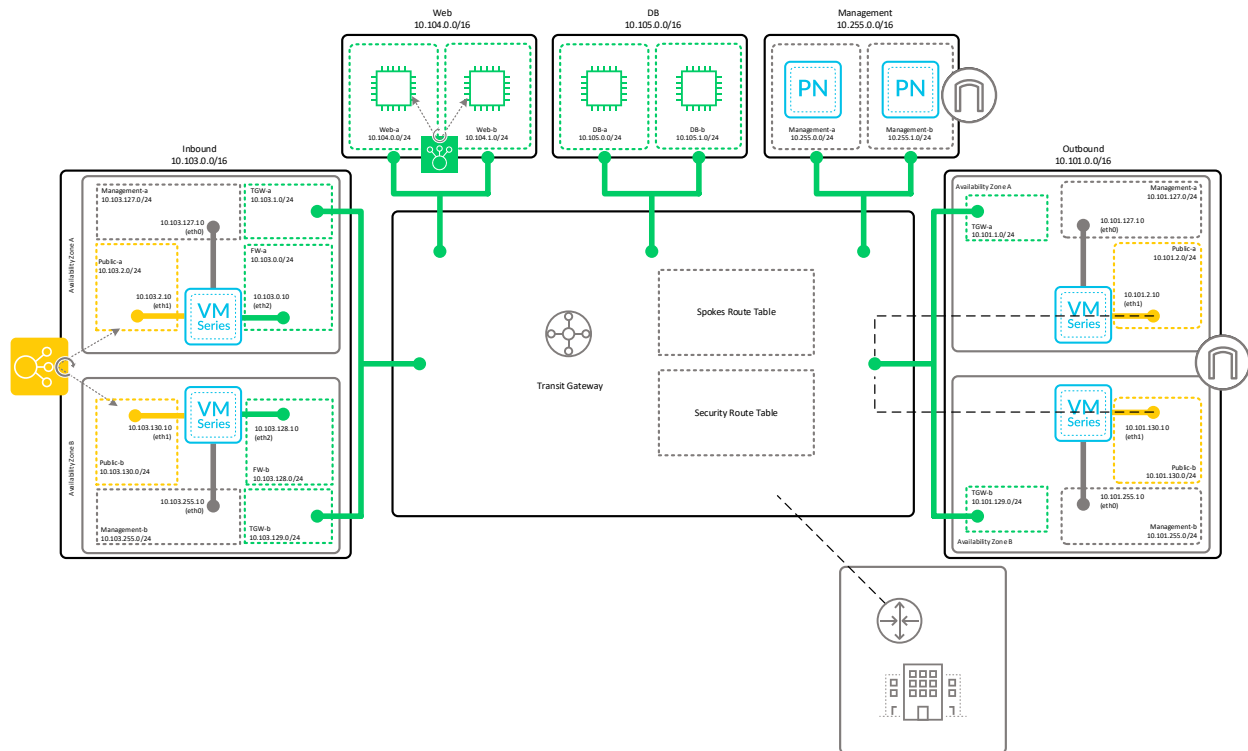
AWS Transit Gateway provides a high-performance solution for connecting large quantities of VPCs, with a scalable solution to support inbound, outbound, and east-west traffic flows through separate dedicated security VPCs.

This implements the second design option, Multiple security VPCs with VPC and VPN attachments, which is the recommended approach because it routes around failures faster by using dynamic routing and Equal Cost Multipath (ECMP) for Outbound Traffic.

East/West Traffic is not covered in this particular implementation. Outbound and Inbound traffic is covered.

Topology

The below topology displays the Transit Gateway and VPCs to be deployed. It also shows the preexisting Panorama instance which will be needed in the same account and region for the purposes of the deployment and is only a requirement and pre-requisite for this template. This topology is deployed in a single account. Inbound traffic and Outbound traffic is handled by separate VPCs and VM-Series. The Outbound VM-Series are attached via a VPN Attachment. East/West Traffic is not implemented but this model could be extended to add E/W traffic and may be implemented in a future version. In addition, though pictured, the On-Premises Network portion is not implemented but could be implemented independently to the Transit Gateway.



Support Policy

This solution is released under an as-is, best effort, support policy. These scripts should be seen as community supported and Palo Alto Networks will contribute our expertise as and when possible. We do not provide technical support or help in using or troubleshooting the components of the project through our normal support options such as Palo Alto Networks support teams, or ASC (Authorized Support Centers) partners and backline support options. The underlying product used (the VM-Series firewall) by the scripts or templates are still supported, but the support is only for the product functionality and not for help in deploying or using the template or script itself.

Prerequisites

The following are the prerequisites to successfully launch the templates:

1. AWS account
2. Clone or download the files and folders to your local machine
3. Deploy Panorama in the same AWS account using the deployment guide, *Panorama on AWS - Deployment Guide.pdf*, which can be found in the *docs* folder.
4. Generate a Panorama VM-Auth Key via the following document, <https://docs.paloaltonetworks.com/vm-series/9-1/vm-series-deployment/bootstrap-the-vm-series-firewall/generate-the-vm-auth-key-on-panorama.html>.
5. If one does not already exist, generate an api key for your primary panorama instance per <https://docs.paloaltonetworks.com/pan-os/9-1/pan-os-panorama-api/get-started-with-the-pan-os-xml-api/get-your-api-key.html>.
6. Ensure AWS credentials are setup for terraform to use. Here are to guides that may assist with this:

- a. <https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-credentials.html>
 - b. <https://www.terraform.io/docs/providers/aws/index.html>
7. BYOL licensing is used and requires four authcode(s) or an authcode that has at least four vm-series associated with it. The authcode(s) must be registered with the support portal. The following link provides instructions on how to do this <https://docs.paloaltonetworks.com/vm-series/9-1/vm-series-deployment/license-the-vm-series-firewall/register-the-vm-series-firewall.html>.
8. In addition to the Panorama AWS Elastic IP, an additional 6 Elastic IP (EIP) will be required. By default, AWS generally only allots 5 initial Elastic IP addresses. You can log a support case for additional EIP. In total you will need at least 7 or if Panorama is in HA, 8.

Panorama Connection Methods

VM-Series connection to Panorama uses bootstrap method which is described at <https://docs.paloaltonetworks.com/vm-series/9-1/vm-series-deployment/bootstrap-the-vm-series-firewall/vm-series-firewall-bootstrap-workflow.html>. The bootstrap package does not use the bootstrap.xml file and all configuration information is pushed by Panorama once registered during the bootstrap process into the appropriate device group and template stack.

Panorama Configuration

This document does not cover the deployment and initial configuration of Panorama. This guide assumes that has been done and that the panorama instance(s) are accessible by the VM-Series during the bootstrap process. Use the deployment guide, *Panorama on AWS - Deployment Guide.pdf*, which can be found in the *docs* folder.

Modify Deployment Files

Use a IDE such as VSCode or file editor such as Notepad++ to modify the following files:

`\setup_panorama\variables.tf`

The items which need to be modified are in a section at the top as shown below:

```

1  //////////////////////////////////////////////////
2  //This section should be verified and modified accordingly.
3  //////////////////////////////////////////////////
4  variable primary_panorama{
5      description = "Primary Panorama Information"
6      default = [
7          "ip" = "x.x.x.x"
8          "api_key" = "
9      ]
10 }
11 }
12
13 #####DO NOT ALTER BELOW VARIABLES#####
14
15 # 3.1 Configure Device Groups
16 variable device_groups{
17     default = {
18         "baseline"      = "TGW_AWS-Baseline"
19         "inbound"       = "TGW_AWS-Inbound"
20         "outbound"      = "TGW_AWS-Outbound"
21     }
22 }
23

```

`\setup_aws\variables.tf`

The items which need to be modified are in a section at the top.

`\setup_aws\init-cfg._inbound_a.txt`

`\setup_aws\init-cfg._inbound_b.txt`

`\setup_aws\init-cfg._outbound_a.txt`

`\setup_aws\init-cfg._outbound_b.txt`

In the four init-cfg files you will need to change the following key pair with the generated vm-auth key generated above:

- vm-auth-key=XXXXXXXXXXXX

Environment Information

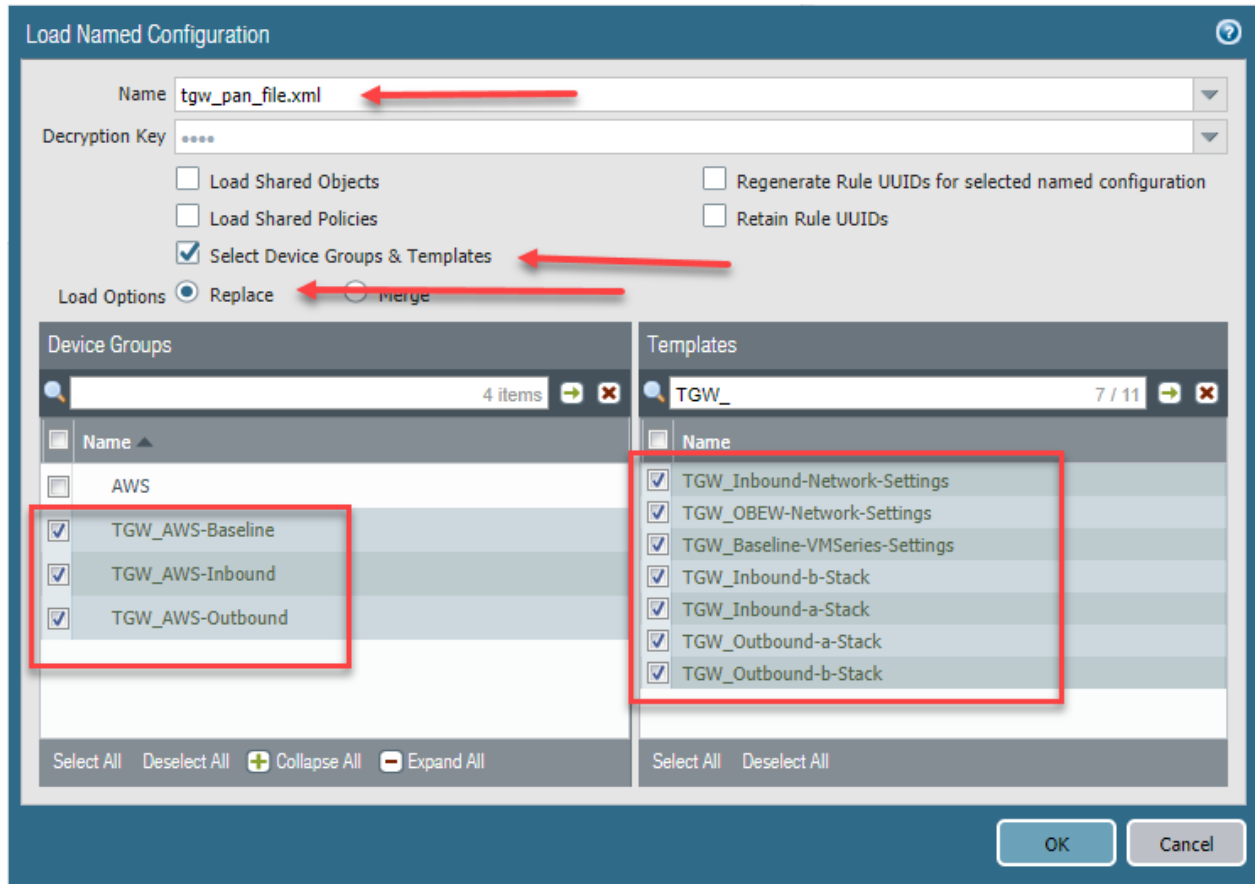
This guide used the following versions:

- Panorama: 9.1.2
- VM-Series: 9.1.3

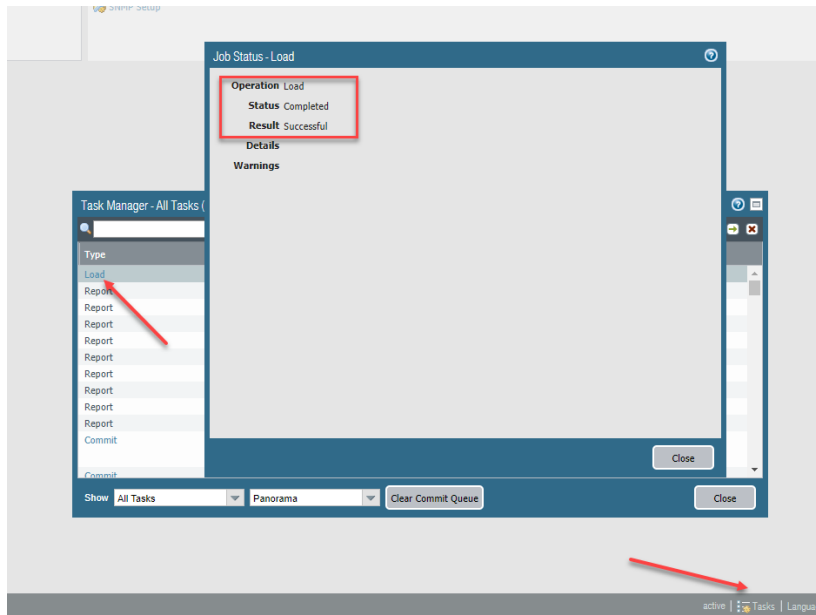
Deploying the setup_panorama Template

1. Under the context of the setup_panorama folder, on the command line:
 - a. Run *terraform init*
 - b. Optionally, before you run apply you may want to run *terraform plan*
 - c. Run *terraform apply --auto-approve*
2. In the Panorama UI, select Commit ->Commit to Panorama
3. Navigate to Panorama -> Setup -> Operations.
4. Select "Import named Panorama configuration snapshot"
5. Select the file *tgw_pan_file.xml* from the \setup_panorama\panorama_configuration_file folder and then select *OK* to import the configuration.
6. Next, select "Load named Panorama configuration snapshot".
7. In the resulting dialog
 - a. Select the file *tgw_pan_file.xml*
 - b. Check the box for "Select Device Groups & Templates"
 - c. For Load Options, select "Replace"
 - d. Under Device Groups, select:
 - i. TGW_AWS-Baseline
 - ii. TGW_AWS-Inbound

- iii. TGW_AWS-Outbound
 - e. Under Templates, select
 - i. TGW_Baseline-VMSeries-Settings
 - ii. TGW_Inbound-Network-Settings
 - iii. TGW_OBEW-Network-Settings
 - iv. TGW_Inbound-a-Stack
 - v. TGW_Inbound-b-Stack
 - vi. TGW_Outbound-a-Stack
 - vii. TGW_Outbound-b-Stack
 - f. The Dialog with look like the below:



- 8. Click OK to close the dialog. Ensure the Load job has completed by viewing the Tasks Manager. You can select the tasks manager by clicking on the bottom right icon which says “Tasks”. See picture below.

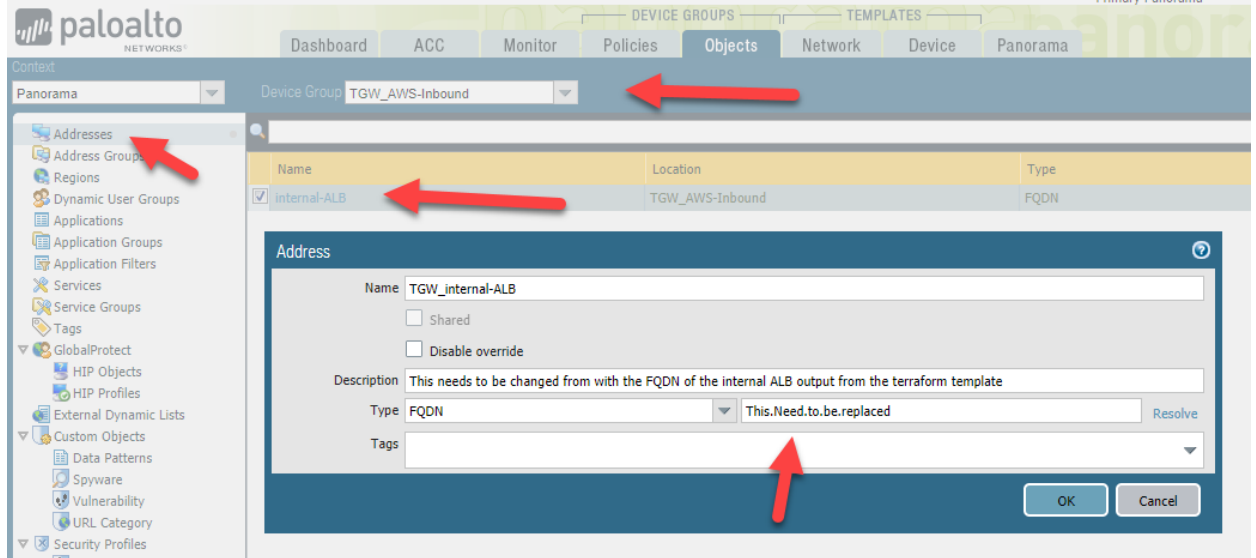


9. Once the load job has completed, select Commit->Commit to Panorama.

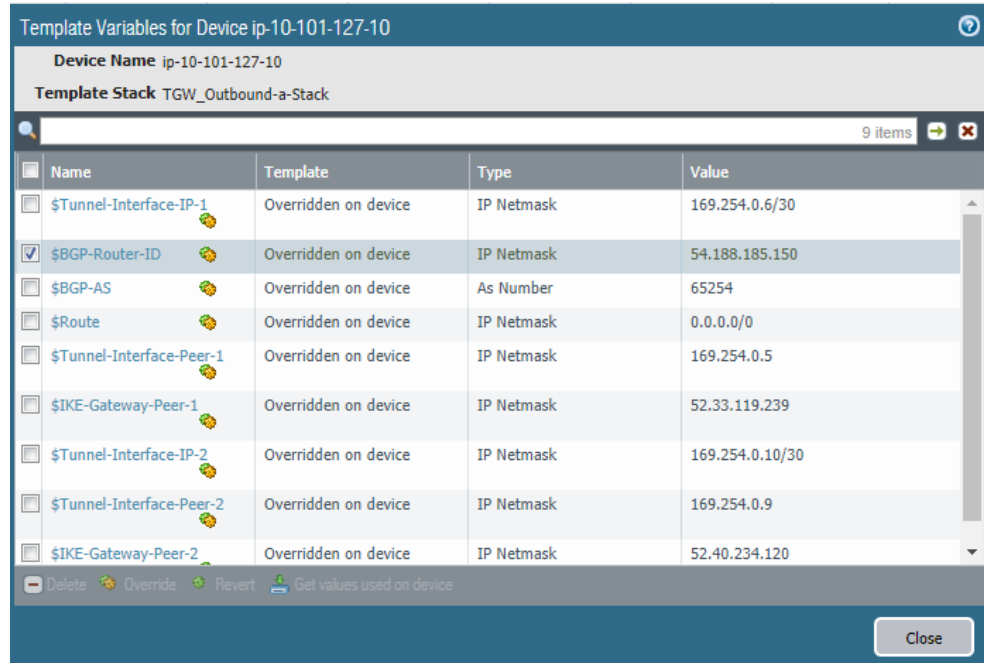
Deploying the setup_aws Template

1. On the command line under the context of the setup_aws directory:
 - a. Run terraform init
 - b. Before you run apply you may want to run terraform plan -var="create_webserver=0"
 - c. Run terraform apply -var="create_webserver=0" --auto-approve
 - i. Ensure there are no errors, it is common to not have enough Elastic IP addresses for the region. In that case, you will need to open a support case with AWS to get additional Elastic IP. Same may happen with VPCs. If you get an error about an Elastic Network Interface not being allocated correctly or not in a correct state you can attempt to immediately run the template again. If the four vm-series firewalls do not successfully register with Panorama, then you may need to run terraform destroy and restart this step.
2. The firewalls take quite a while to come up and register with Panorama once the terraform template has completed (20-30 minutes). Validate the firewalls have come up by navigating on Panorama to Panorama->Managed Devices->Summary. The four new firewalls should come up under the correct template stacks and device groups.
3. Once all the firewalls are registered with Panorama:
 - a. Navigate to Panorama-> Device Deployment-> Licenses.
 - b. Click refresh
 - c. In the resulting dialog under the device name column choose the four newly added firewalls and then click Refresh.

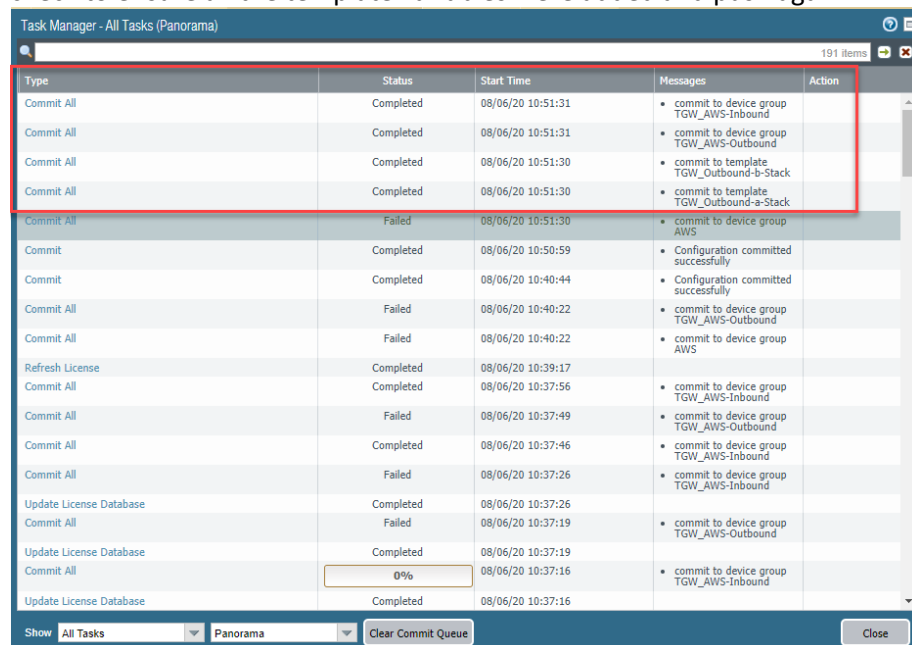
4. Next, update the internal load balancer as picture below in the address object from the output of the terraform template and then Commit -> Commit and Push.



5. Next you will need to manually add all the template variables defined for each outgoing Firewall. The variables can be found in the output of the terraform run.
 - a. Navigate to Panorama -> Managed Devices -> Summary
 - b. Go to the row with device name *ip-10-101-127-10* and click [Create](#).
 - c. On the Create Device Variable Definition dialog box, select *No*, and then click *OK*.
 - d. On the Template Variables for Device *ip-10-101-127-10* dialog box, select the row for *\$Tunnel-Interface-IP-1*.
 - e. Click *Override*, enter the corresponding value from the terraform output under the output header, *Outbound-FW-1__Tunnel-Interface-IP-1*, and then click *OK*.
 - f. Continue to do this for every variable in the dialog till each has a value as pictured below:



- g. Once completed click close.
- h. Repeat steps b-g for the second device with name ip-10-101-255-10 using the output from the terraform template with prefix *Outbound-FW-2__*.
- i. Now Commit->Commit and then Commit->Push to Devices. Ensure that the pushes were successful for both the device push and template push as shown below. If not check to ensure all the template variables were added and push again.



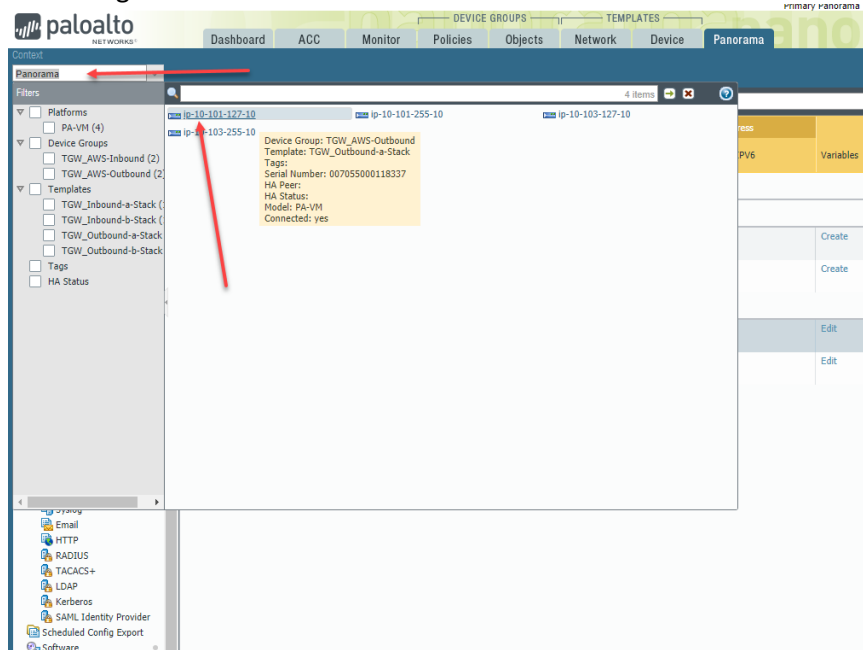
6. Before proceeding, ensure all devices are showing synched for both device group and template (i.e. green dots in the picture below). You may need to hit the refresh button indicated by the

red arrow in the picture below a few times. If they are not you may need to Commit->Commit and Push again.

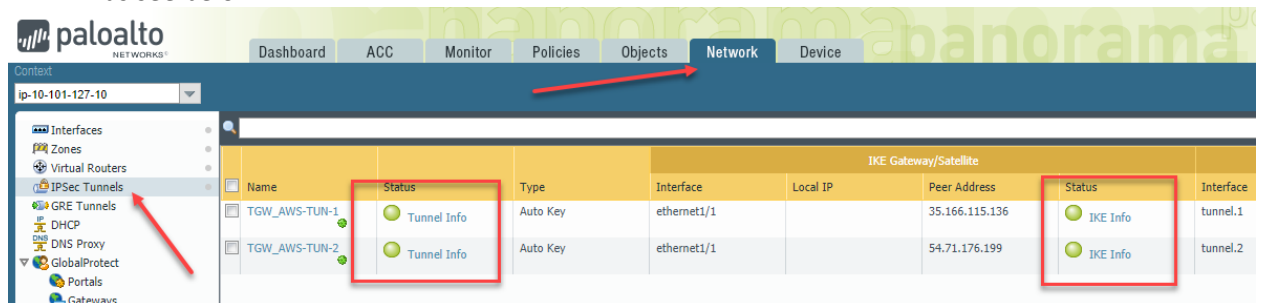
Manual

Device Name	Virtual System	Model	Tags	Serial Number	IPV4	IPV6	Variables	Template	Device State	Device Certificate	Device Certificate Expiry Date	HA Status	Shared Policy	Template	Certificate	Status	
AWS (0/3 Devices Connected): Shared > AWS																	
TGW_AWS-Inbound (2/2 Devices Connected): Shared > TGW_AWS-Inbound																	
<input type="checkbox"/> ip-10-103-127-10	PA-VM			0070550001183...	10.103.127.10 (DHCP)		Create	TGW_Inbound-a-Stack	Connected	None	N/A				<div><div></div>In Sync</div>	<div><div></div>In Sync</div>	pre-defined
<input type="checkbox"/> ip-10-103-255-10	PA-VM			0070550001183...	10.103.255.10 (DHCP)		Create	TGW_Inbound-b-Stack	Connected	None	N/A				<div><div></div>In Sync</div>	<div><div></div>In Sync</div>	pre-defined
TGW_AWS-Outbound (2/2 Devices Connected): Shared > TGW_AWS-Outbound																	
<input checked="" type="checkbox"/> ip-10-101-255-10	PA-VM			0070550001183...	10.101.255.10 (DHCP)		Edit	TGW_Outbound-b-Stack	Connected	None	N/A				<div><div></div>In Sync</div>	<div><div></div>In Sync</div>	pre-defined
<input type="checkbox"/> ip-10-101-127-10	PA-VM			0070550001183...	10.101.127.10 (DHCP)		Edit	TGW_Outbound-a-Stack	Connected	None	N/A				<div><div></div>In Sync</div>	<div><div></div>In Sync</div>	pre-defined

7. Once all devices show synced, navigate to the two outbound firewalls to ensure the VPN tunnels for the VPN attachments to the Transit Gateway are up.
 - a. First navigate to one of the two outbound firewalls via Panorama as shown below.



- b. Once on the firewall navigate to Network->IPSec Tunnels and ensure all dots are green as see below:



Network							
Context: ip-10-101-127-10							
IPSec Tunnels							
Name	Status	Type	Interface	Local IP	Peer Address	Status	Interface
TGW_AWS-TUN-1	Tunnel Info	Auto Key	ethernet1/1		35.166.115.136	IKE Info	tunnel.1
TGW_AWS-TUN-2	Tunnel Info	Auto Key	ethernet1/1		54.71.176.199	IKE Info	tunnel.2

- c. Repeat for the second firewall.

8. Now run the template a second time but change the *create_webserver* variable to create the webserver and attach them to the internal ALB.
 - a. Run the following command: `terraform apply -var="create_webserver=1" --auto-approve`

Verifying Deployment

Once the web servers are up and configured you can quickly determine outgoing traffic by looking at the Monitors->Logs-Traffic and you should see outgoing traffic from the web servers to ntp and yum. You can validate incoming traffic by using the output from the template run, specifically the the Inbound Public Application Load Balancer FQDN can be put as the address in a browser and should resolve to one of the two web servers. Refresh several times and you will see the hostname in the response change.

Cleanup

To destroy the environment:

1. Run the command, *terraform destroy* in the context of setup_aws folder. This will destroy the AWS environment.
2. Run the command *terraform destroy* in the context of setup_panorama folder.
3. In Panorama, navigate to Panorama->Managed Devices->Summary and delete the four vm-series firewalls which were already destroyed. They should now be under, *No Device Group Assigned*.
4. Commit the changes to Panorama via Commit->Commit to Panorama

Once completed, all resources in both AWS and Panorama should be removed.