Transit VPC on AWS with the VM-Series- Step by Step Guide

This document provides a step by step guide to manually build a Transit VPC on AWS with 3 subscribing VPC's protected by the VM-Series (deployed in the transit hub). The Transit VPC allows you to deploy services such as outbound internet access through a hub and spoke architecture where the hub (Transit VPC) houses common services such as security and spokes house the workloads. We will build the Transit VPC with 2 Availability Zones and a firewall in each Availability Zone for HA. We will use the Virtual Gateway (VGW) in each subscribing VPC to provide redundant tunnels to each firewall in the Transit VPC. If you choose, you can deploy the subscribing VPC in multiple AWS Availability Zones to include redundancy. This document will also utilize eBGP as a dynamic routing protocol to provide routes to subscribing VPC's.

NOTE: This guide takes a manual, step by step approach to building the Transit VPC. A more automate approach can be found in this guide
https://github.com/PaloAltoNetworks/aws-transit-vpc

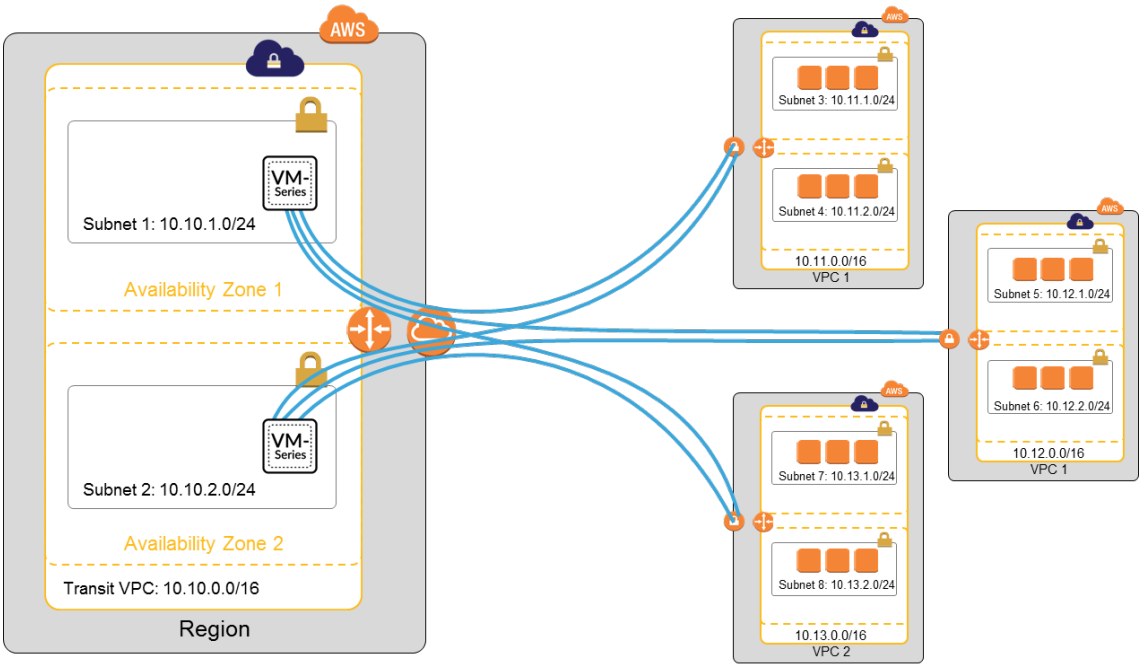
Overview of the steps to create a Transit VPC:

1. Create a Transit VPC acting as the hub with multiple Availability Zones and AWS IGW
2. Create (3) Subscriber VPC's with a VGW for VPN peering
3. Create VPN tunnels from Subscriber VPC VGW's to Palo Alto Networks VM-Series firewalls located in the hub
4. Configure BGP on VM-Series firewall's

Note: This document will assume that you have the following:

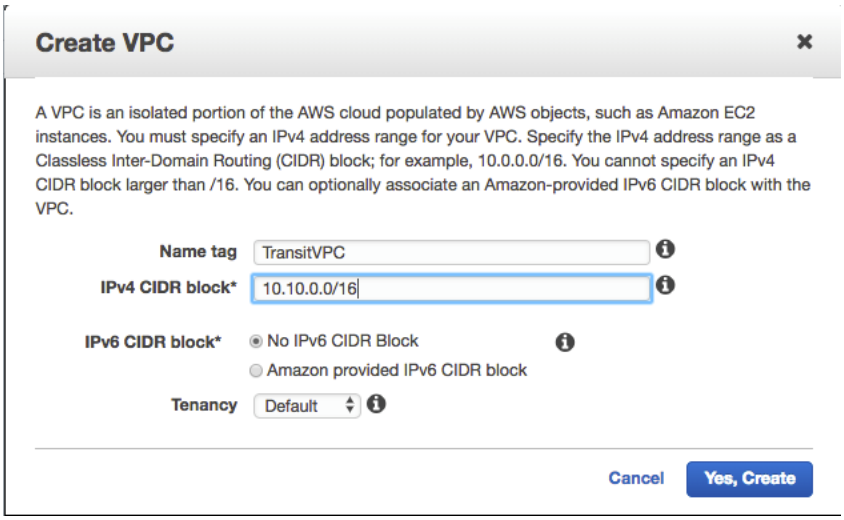
- AWS account
- Proper permissions to create VPC's, route tables, subnets, etc.
- Created a Key pair for the AWS region where you will be building this solution.
- Permissions to deploy EC2 instances from the AWS marketplace.

Overview of the architecture that will be built:



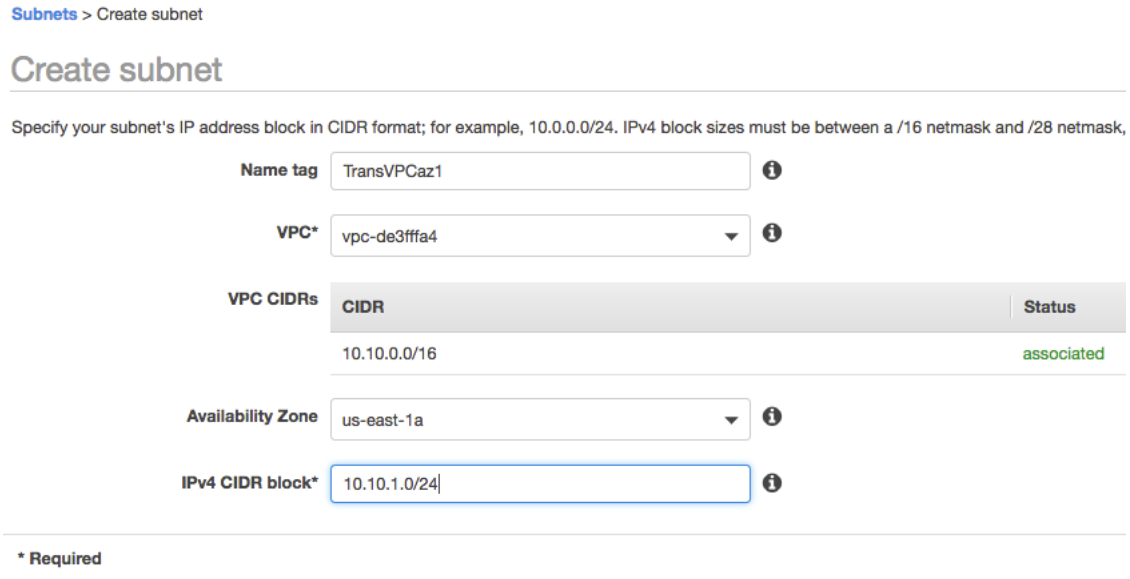
1. Create Transit VPC with multiple Availability Zones and IGW

In the AWS console, create the new Transit VPC from the VPC dashboard as depicted below:

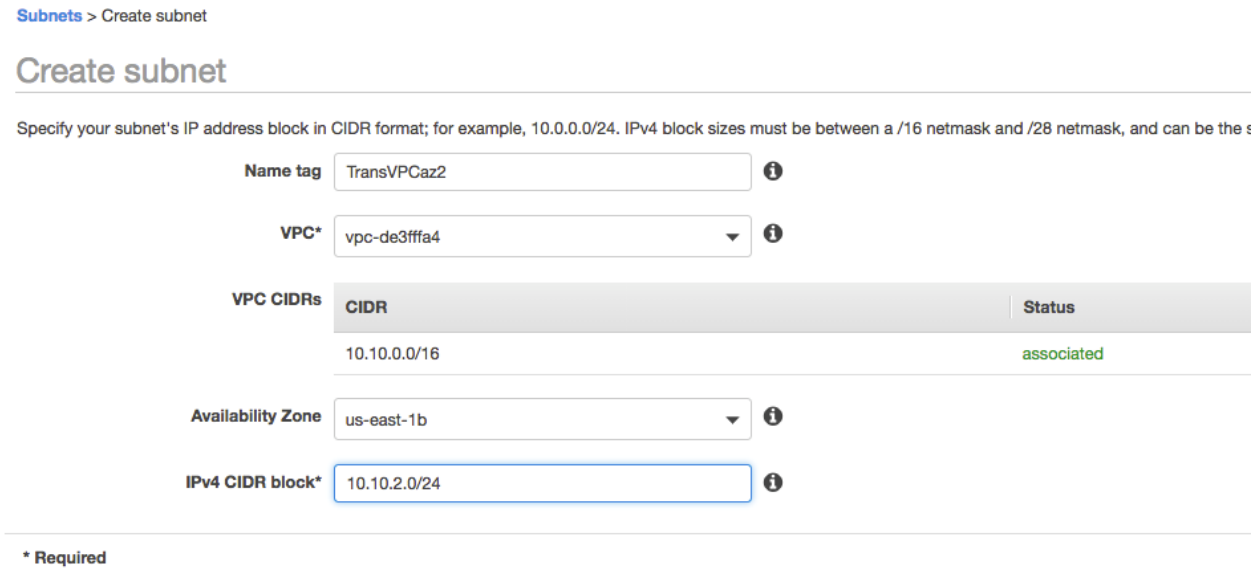


Next we will create the Subnets and Availability Zone associations:

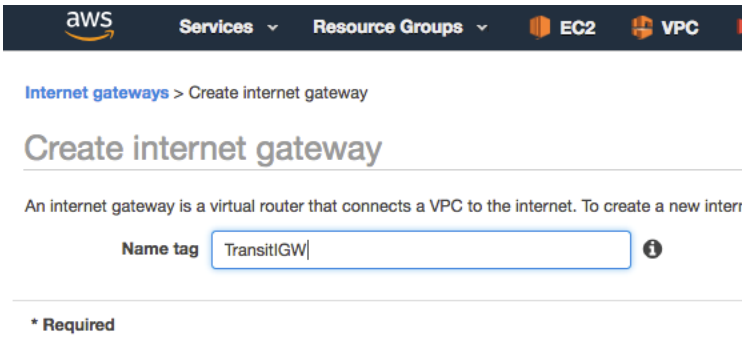
From the VPC dashboard click on the Subnets menu and configure your subnets as below:



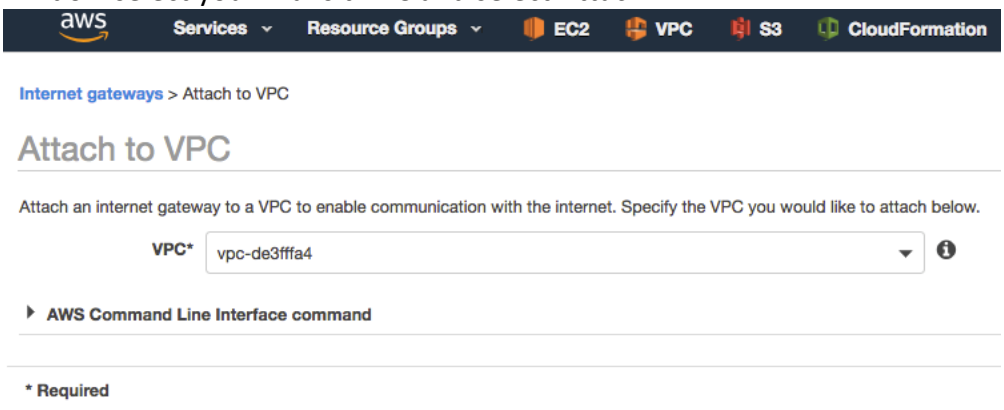
Next, create a second subnet:



We will now deploy an AWS Internet Gateway (IGW) from the VPC dash board.
From the VPC dashboard click on the Internet Gateways menu:

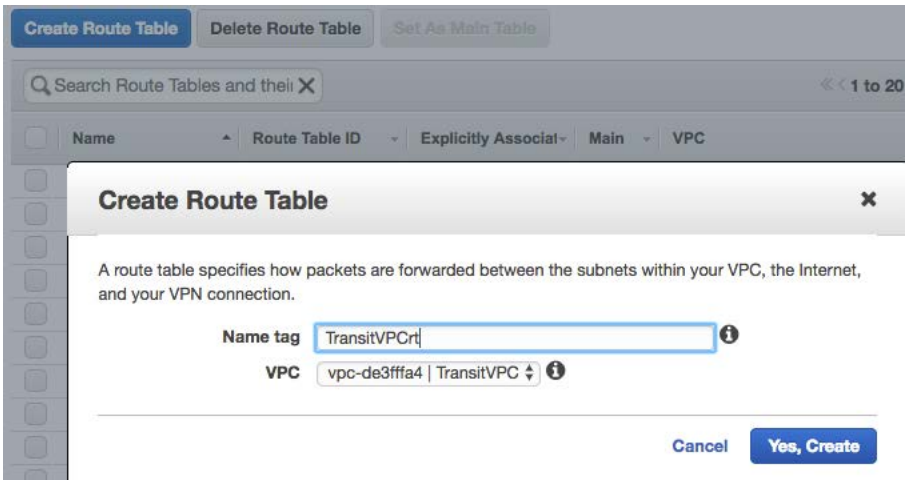


Attach the IGW. Click on the ***attach to VPC*** button and from the Actions drop down in the window select your Transit VPC and select Attach.



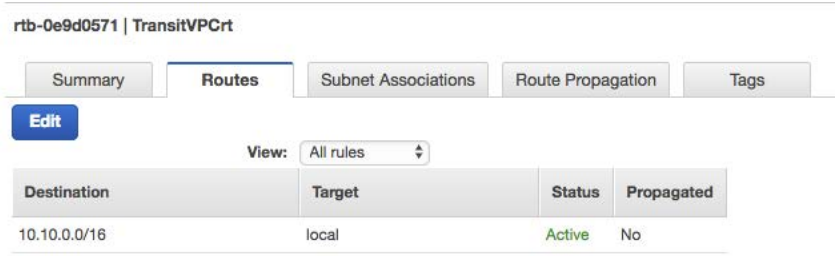
Your Internet Gateway screen should look like the screen below:

Next, we will create a route table for the Transit VPC: from the VPC dashboard select Route Tables menu and create Route table as below:

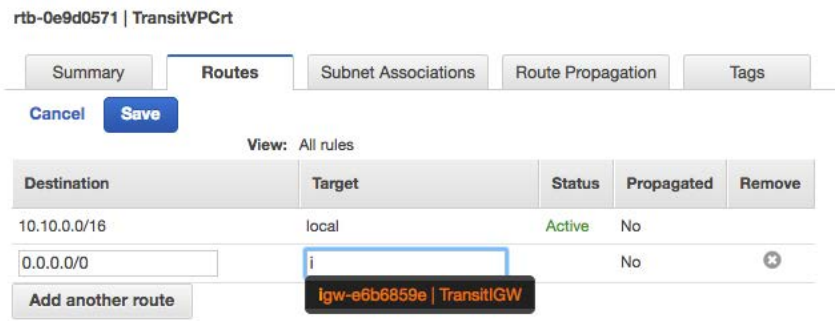


The route table must be associated with your Transit VPC as highlighted above. Once you provide a name and select the Transit VPC from the drop-down menu, select the yes, create button.

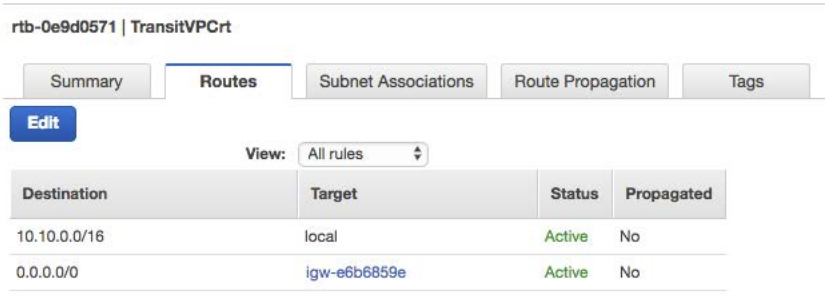
Next step creates routes for the new Transit VPC route table. Click in the empty check box next to your new route table and click on the edit button in the bottom window:



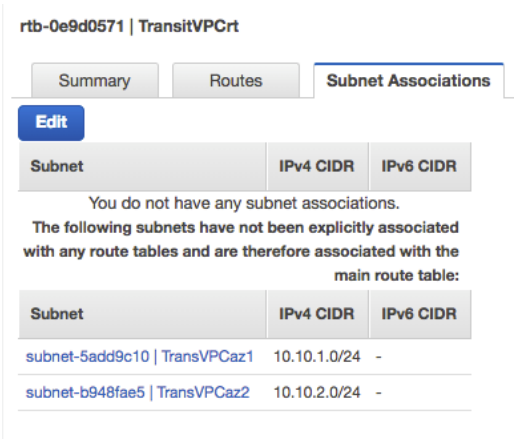
From the edit screen click the add another route button to create a default route to the IGW. Make sure you click the save button when you are done:



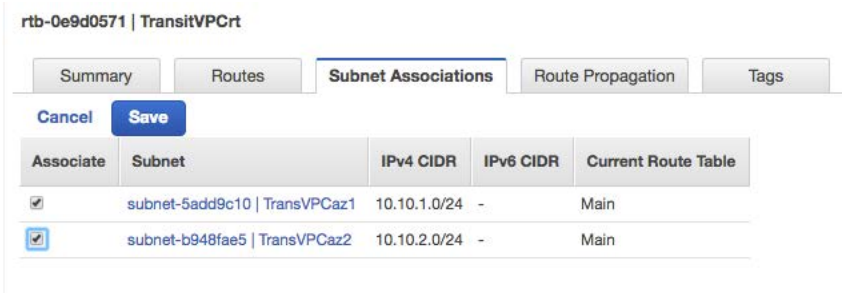
Your route table should like the image below:



Next, from the same window, select the subnet associations tab and select the edit button:



From the edit screen, select the empty check boxes to the left of the subnets you created earlier. Once complete select the save button:

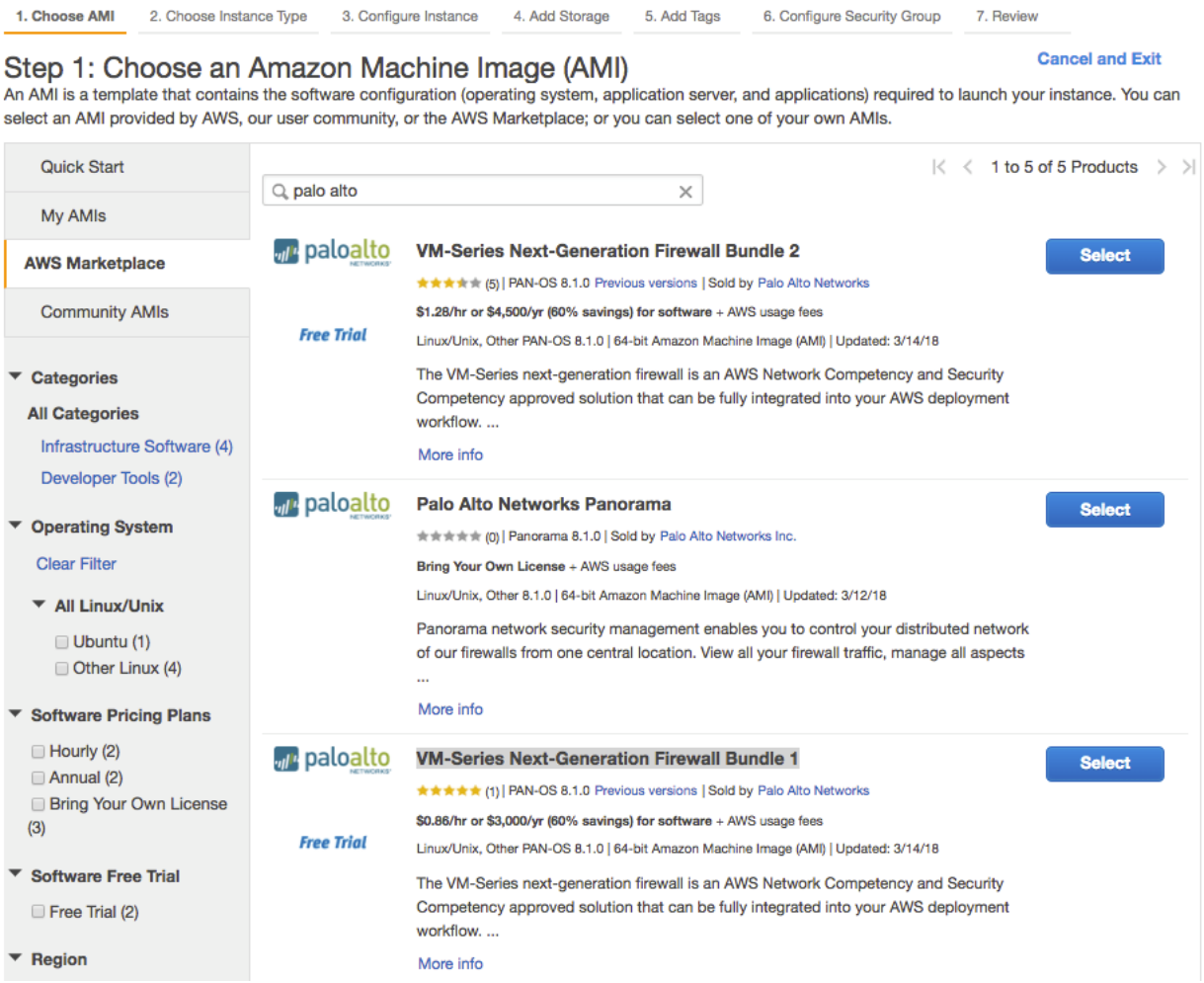


Next you are going to deploy a VM-Series firewall into each of the Availability Zones:

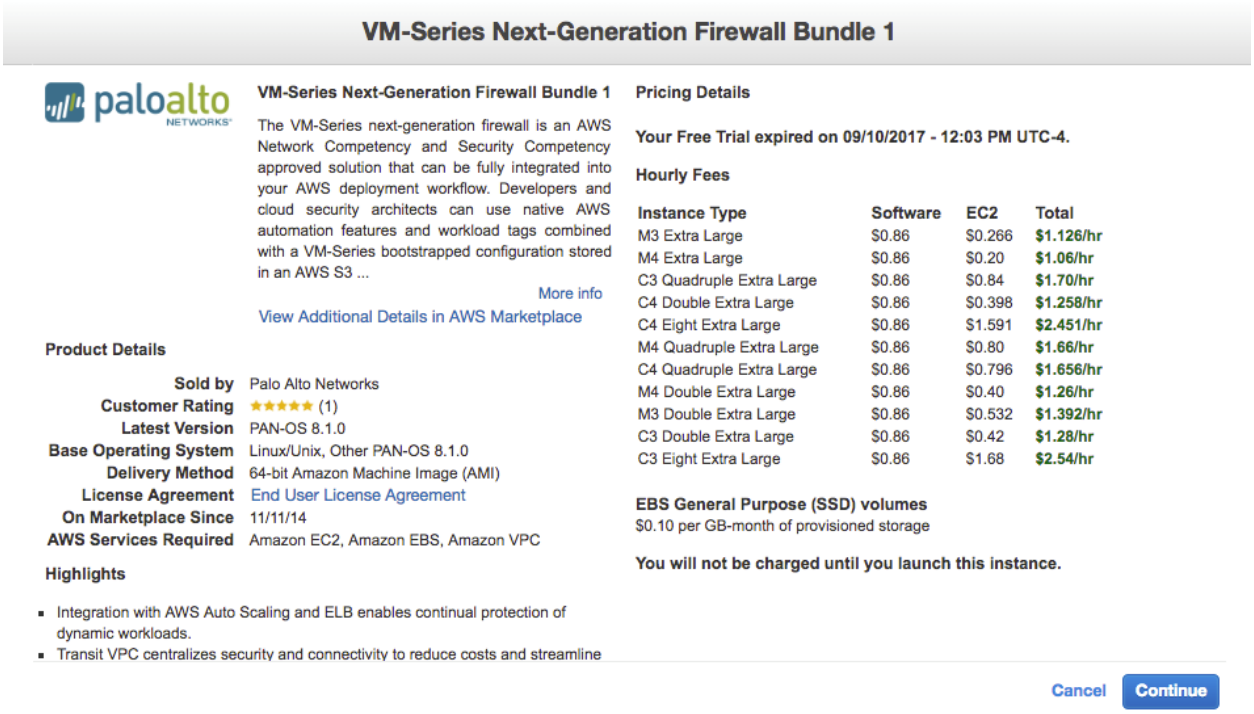
From the EC2 Dashboard, select the Launch Instance button.

From the Left-Hand Menu, select AWS Market Place and search for Palo Alto Networks.

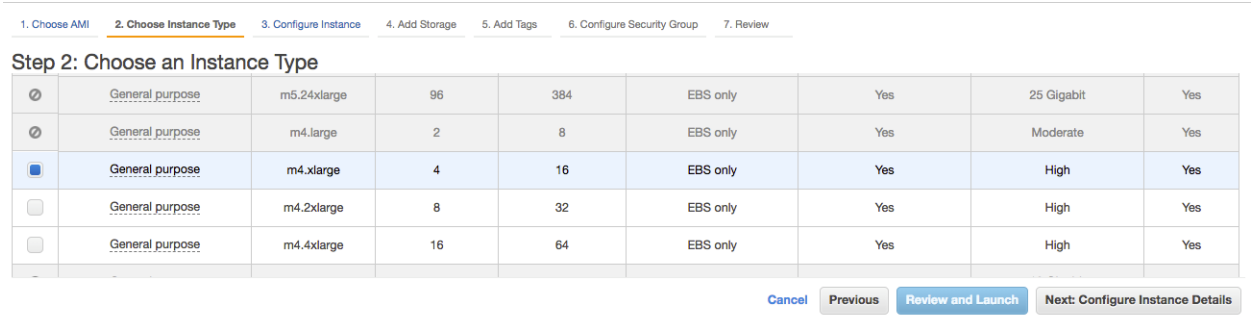
Select VM-Series Next-Generation Firewall Bundle 1 from the list.
Bundle 2 will provide the highest levels of security. We are selecting Bundle 1 here to easy to cost in a demo environment.



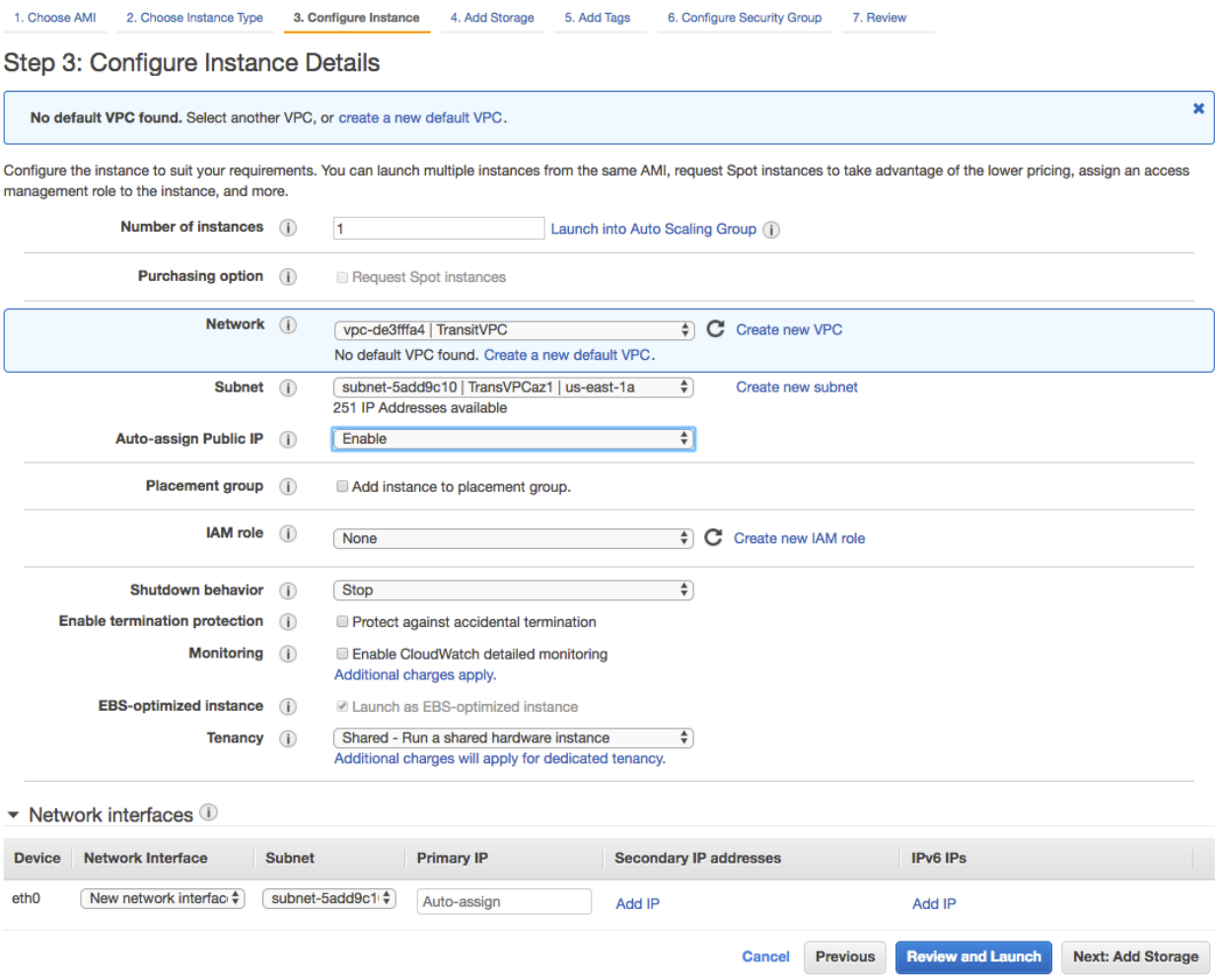
Select Continue for the Pricing Details.



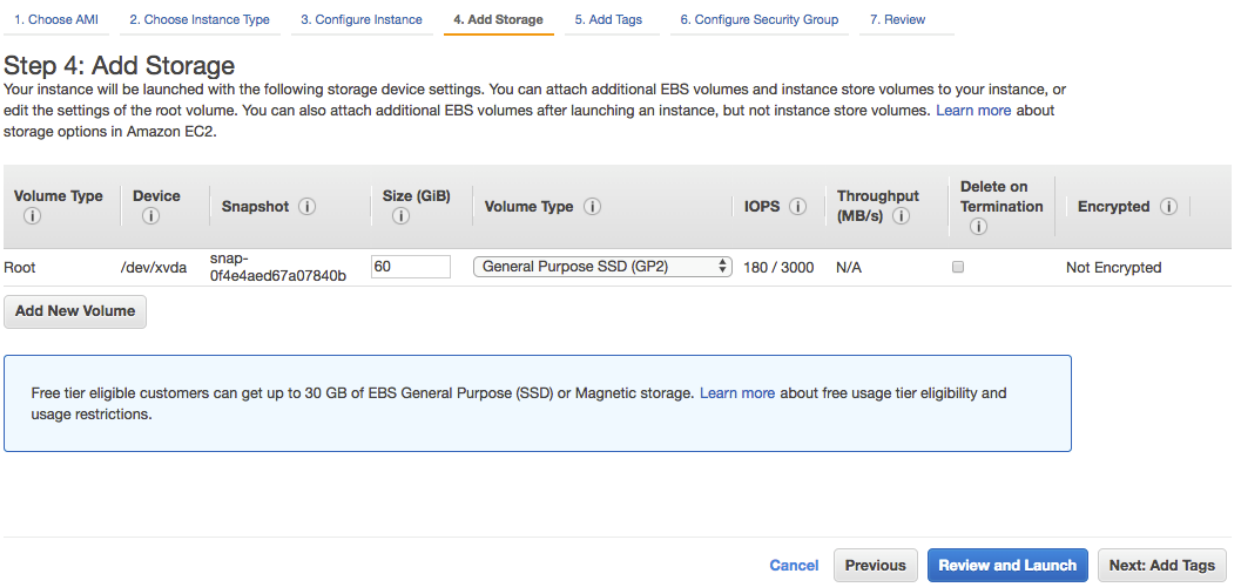
Select the Smallest available Instance type. For this guide, we chose m4.xlarge, select Next: Configurate Instance Details.



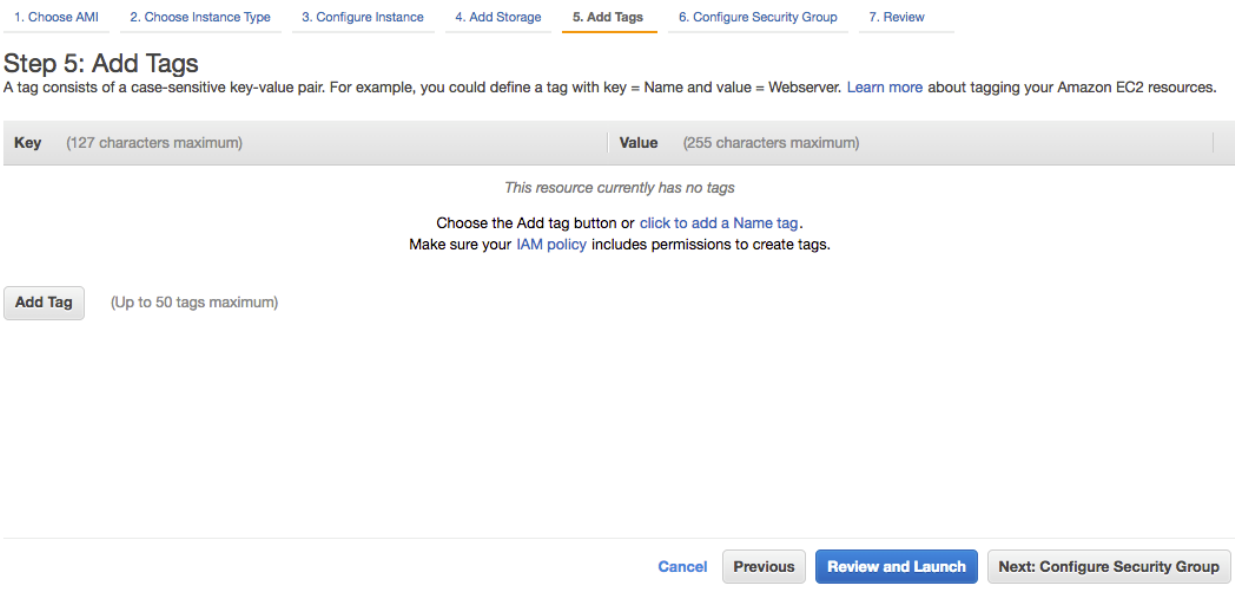
Configure the instance deployment for the Availability Zones in your Transit VPC. We will start with AZ-1A.Note that you will need to repeat these steps to deploy a firewall instance into AZ-1B. Once you have verified your settings, select Next: Add Storage:



Keep the storage defaults and click on the Next: Add Tags button at the bottom of the page:

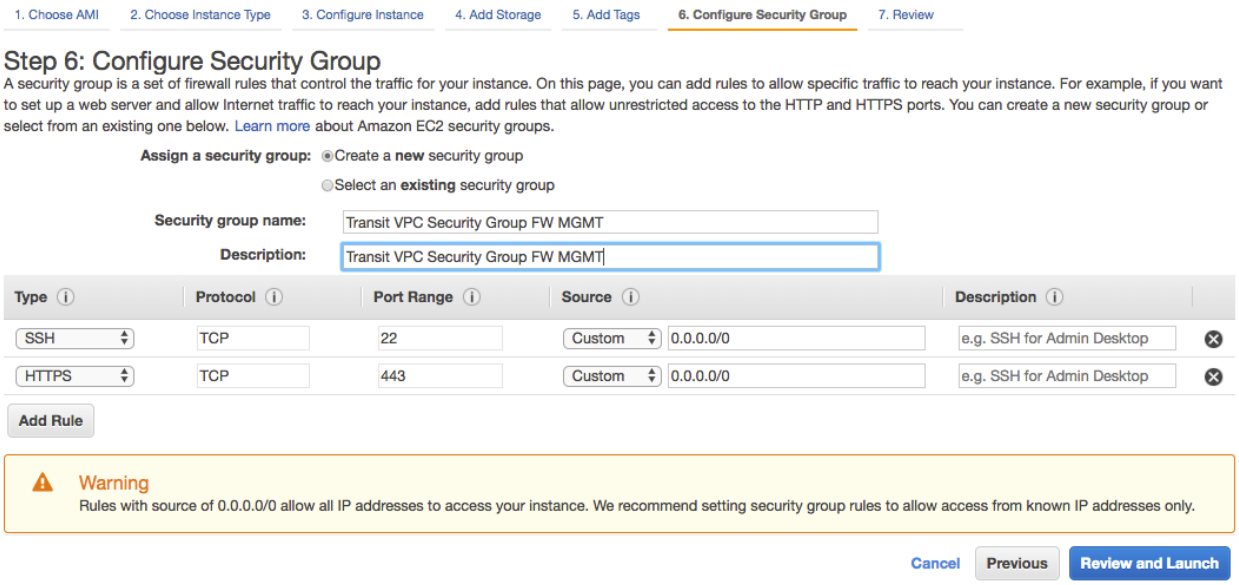


You can add tags if you would like to identify your instance or you can skip this step and move on to Nets: Security group configuration:

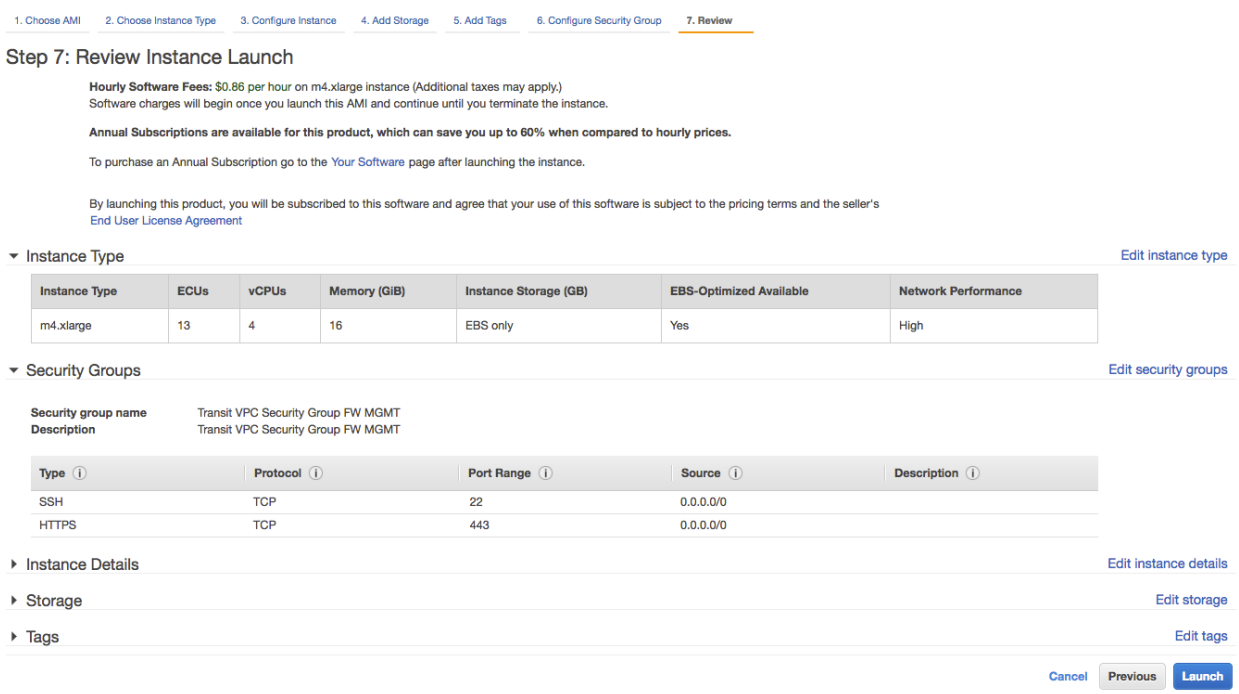


Create a new Security Group that you can name for the Transit VPC. For the firewall, we need to make sure that the Security Group allows SSH and HTTPS access as shown below. Once the Security Group is configured, select the review and Launch button.

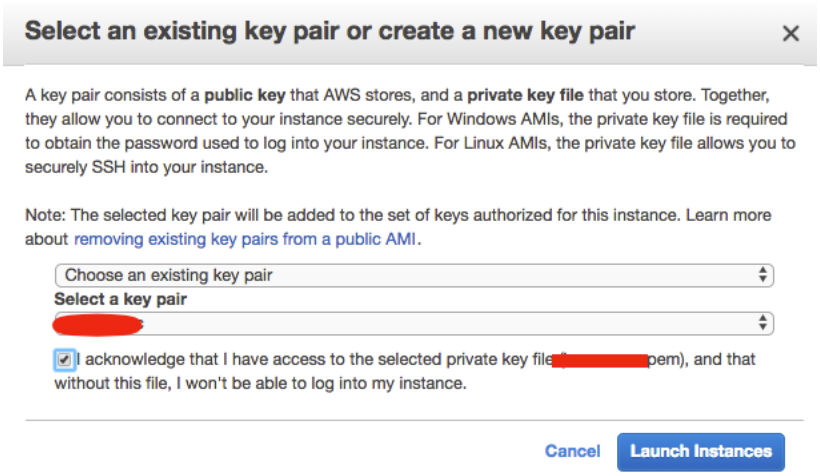
Important It is recommended to set the Source of the Security Group to be specific to your location. When clicking the Custom drop-down, AWS will resolve your External IP and designate it as a /32.



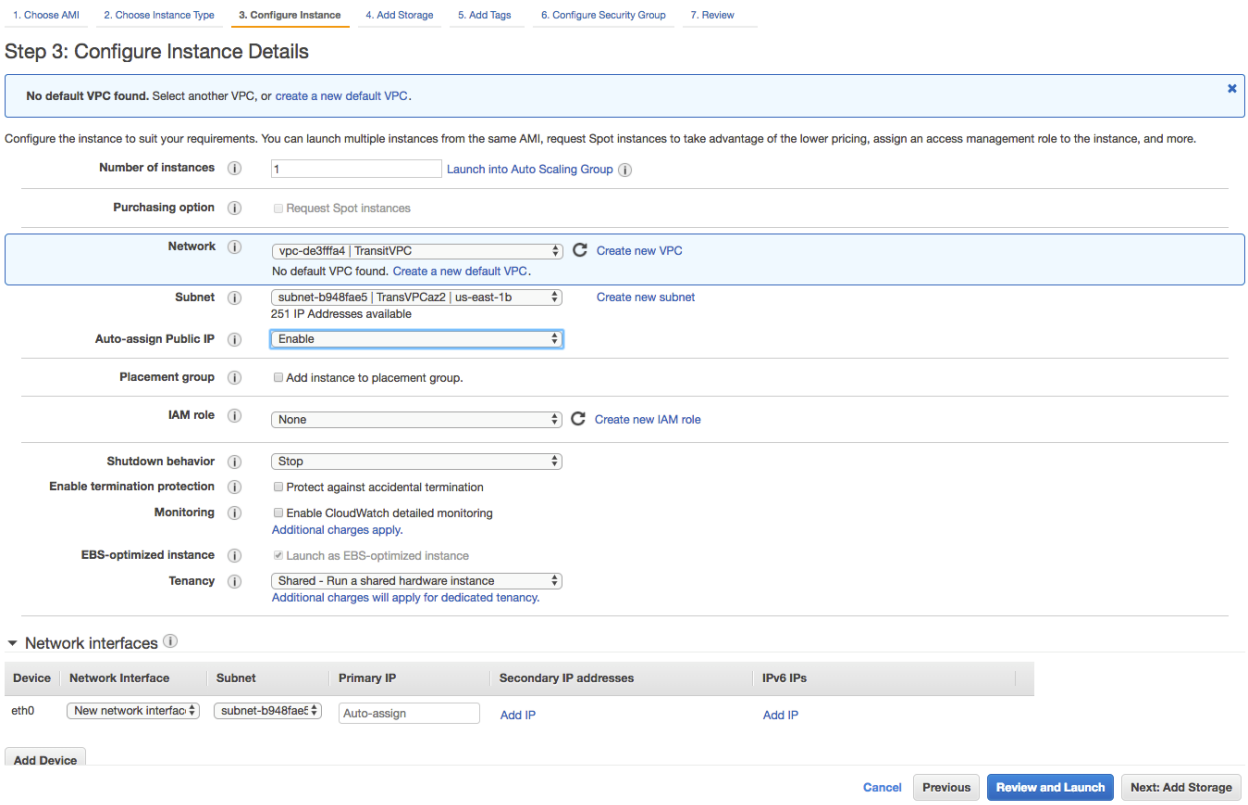
Review the settings and select the Launch button:



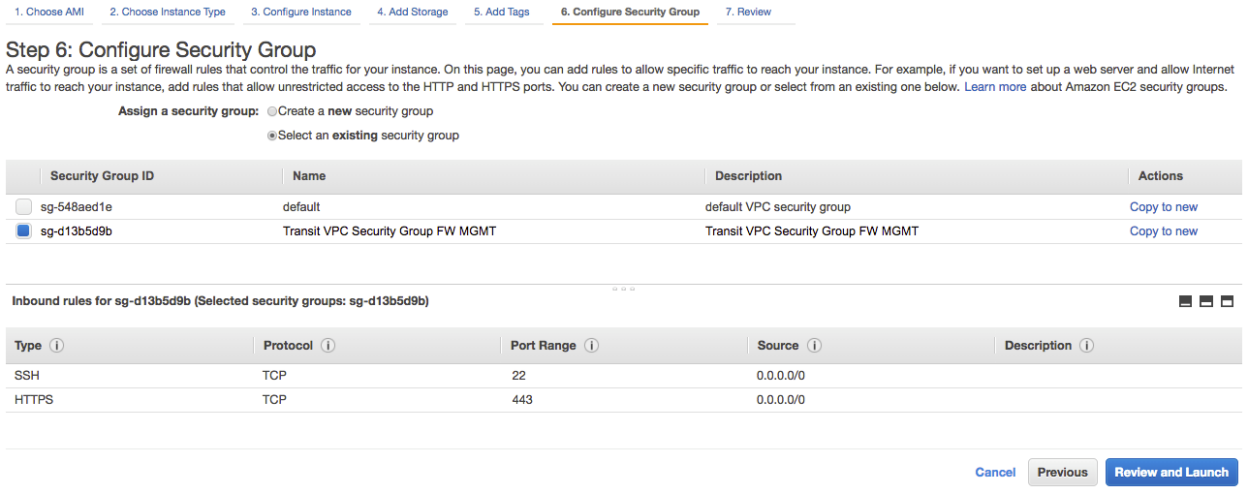
Before the instance will launch, you will have to select a Keypair to associate to the instance for access and select the acknowledge box:



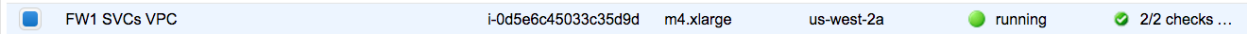
You will need to repeat the steps for deploying a firewall instance for the second Availability Zone. If you chose AZ-1a in the above section, you will need to repeat the steps in AZ-1b.



For the second instance, you can use the same Security Group that was created in the deployment of the first firewall instance:
Important – If using a new Security Group, ensure to use a more specific Source as recommended above.

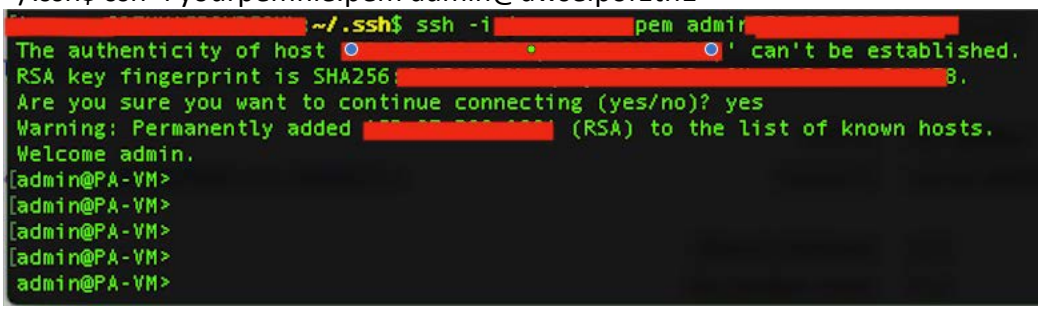


Once the firewalls are deployed and from the EC2 Dashboard click on the instance option from the ec2 dashboard. You will see a list of all instances in your environment. Once the new firewall instance shows a status like below, you will need to SSH into the firewall to setup a username and password to enable the Web Interface:



From the instance you can scroll to the right to find the public IP address. Once you have the public IP address, open an SSH client to access your device. You will need to use the keypair .pem file used to deploy the firewalls to access the new firewall. My public IP address is blocked out. You will need to provide your public IP address for the firewall in the command:

~/.ssh$ ssh -i yourpemfile.pem admin@awseipofEth1



NOTE If you are using MAC/Linux as receive a Public Key denied error, change the permission of you PEM file to proper restrict it.
chmod 600 yourpemfile.pem

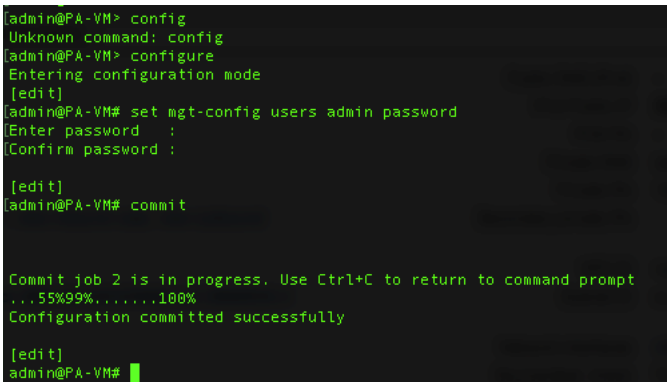
Once you gain access to the firewall, you will need to issue the following commands:

Configure

set mgt-config users admin password

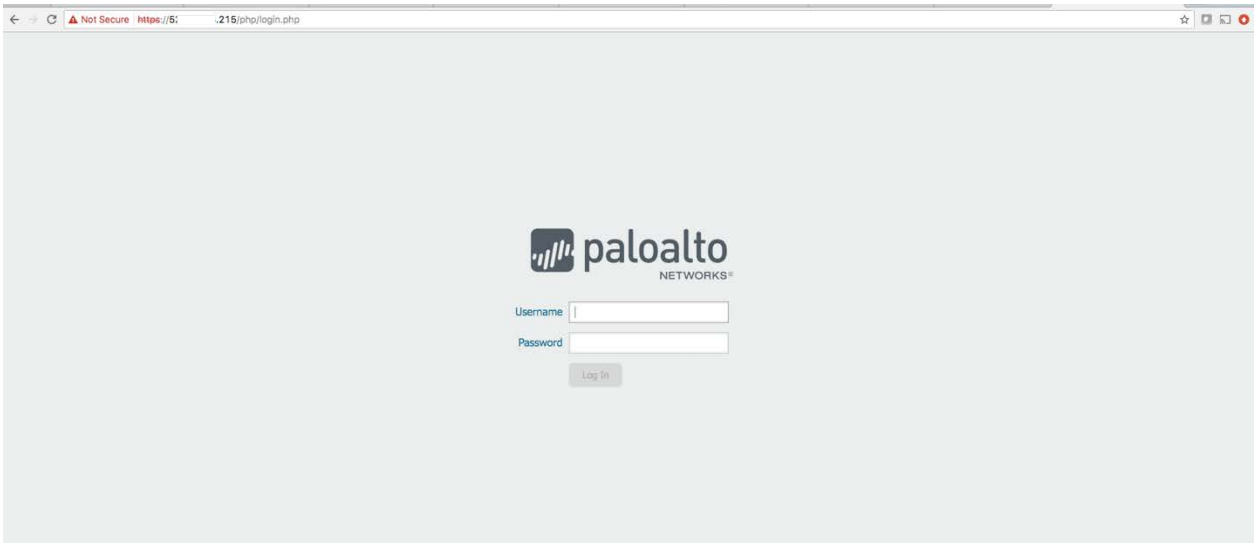
This will allow you to set an admin password after you type the password and confirm the password:

Once you have created the password, you must commit the configuration changes.



You will need to complete the SSH steps for both firewalls to access the firewalls from a web browser.

You should now be able to access your firewalls via web browser using https. The username should be admin and the password will be the password you set via the SSH commands:



If you chose to use the VM-Series BYOL AWS Marketplace AMI, you will need to license each of the VM-Series firewalls. If you are not familiar with this process, the following link can be used to help license the firewalls:

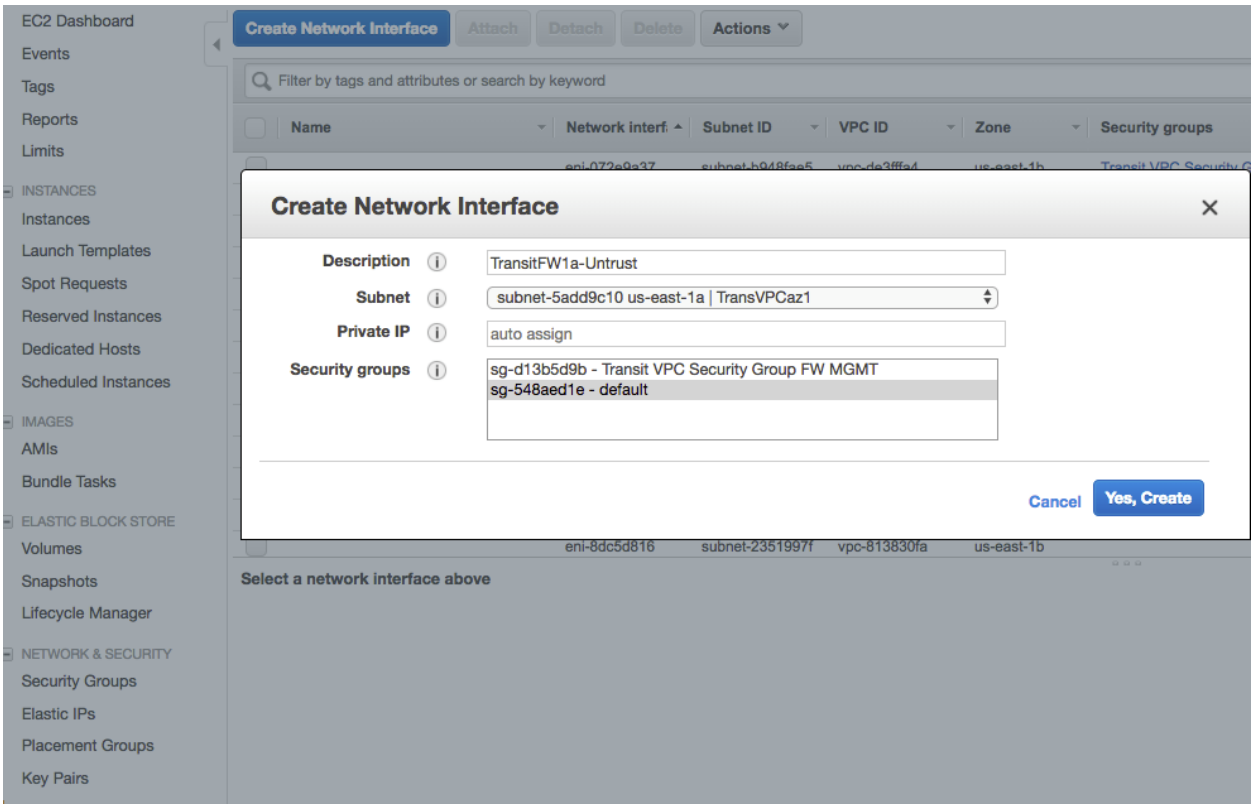
https://www.paloaltonetworks.com/documentation/81/virtualization/virtualization/license-the-vm-series-firewall/activate-the-license/activate-the-license-for-the-vm-series-firewall-standalone-version

If you selected either of the AWS Marketplace Bundles, you can ignore this step.

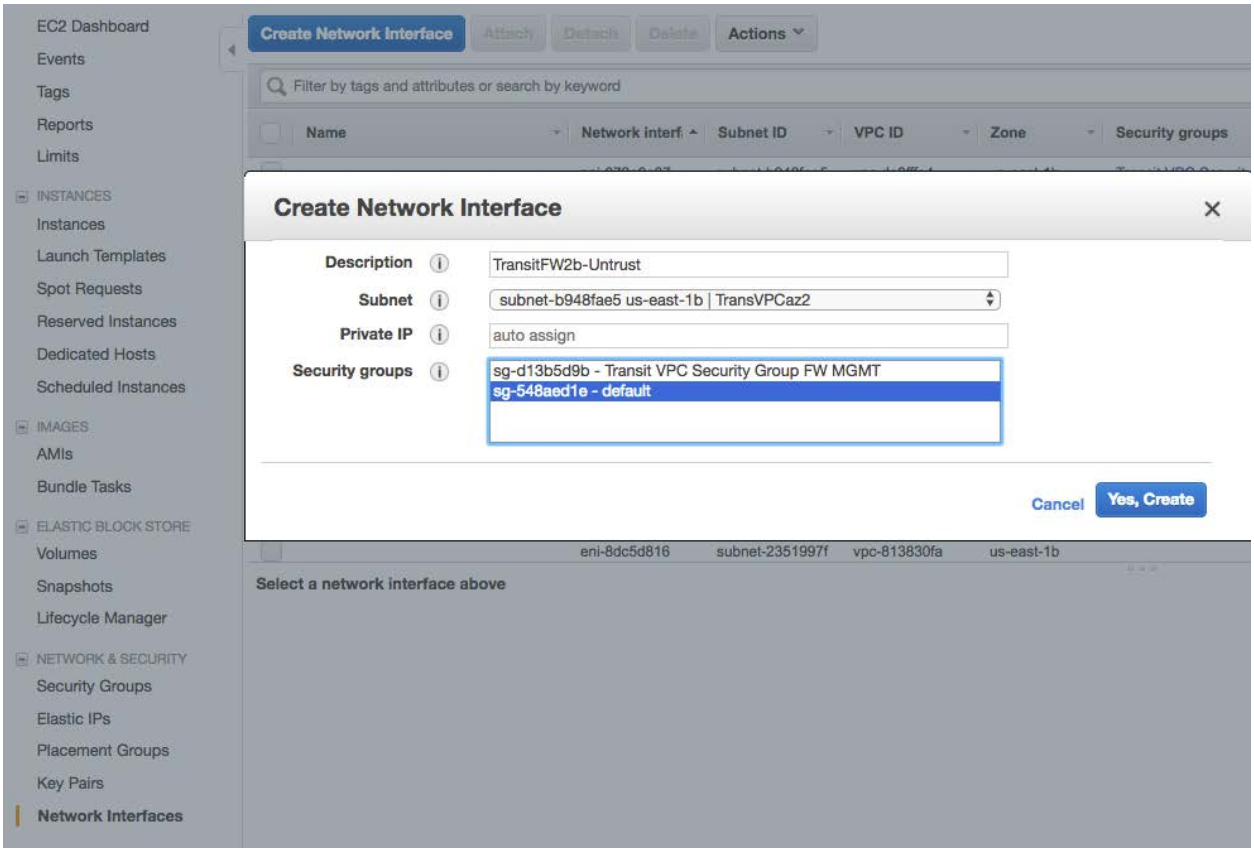
Create the Untrust interface for the new firewalls:

In the deployment of the VM-Series firewall, the management interface was configured. We will now add an Untrust interface to the firewall. From the AWS console, go to the EC2 dashboard. From the EC2 dashboard select the Network Interfaces option from the left panel.

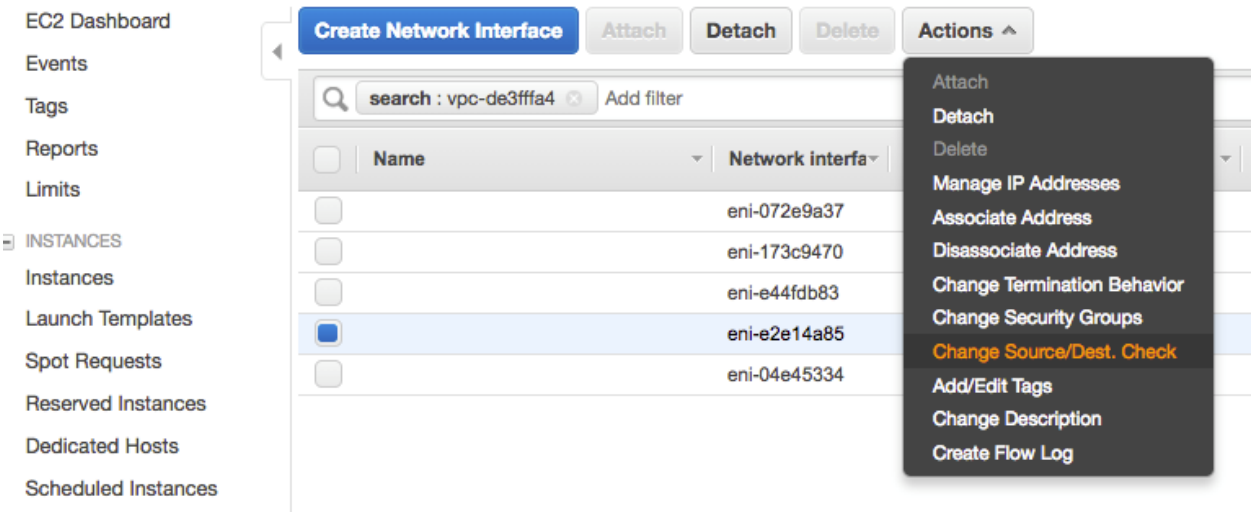
From the main Window select the Create Network Interface. You will need to create a label to identify the eth 1/1 of each firewall for the Untrust interface. You will need to assign a network interface in each subnet. Firewall1 should have been deployed in AZ-a and firewall 2 in AZ-b. Make sure that each label is set accordingly to the correct AZ.

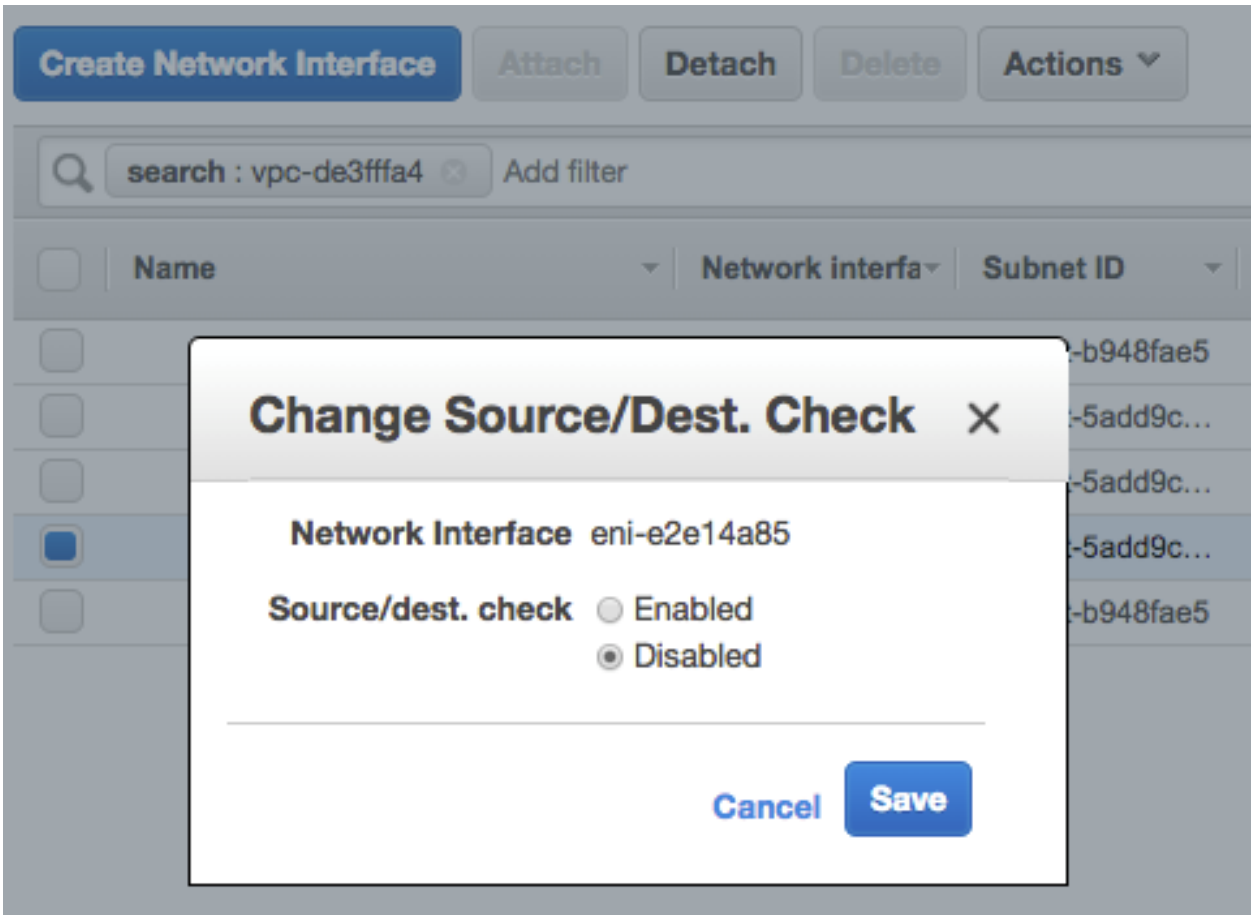


Firewall2 Network Interface:

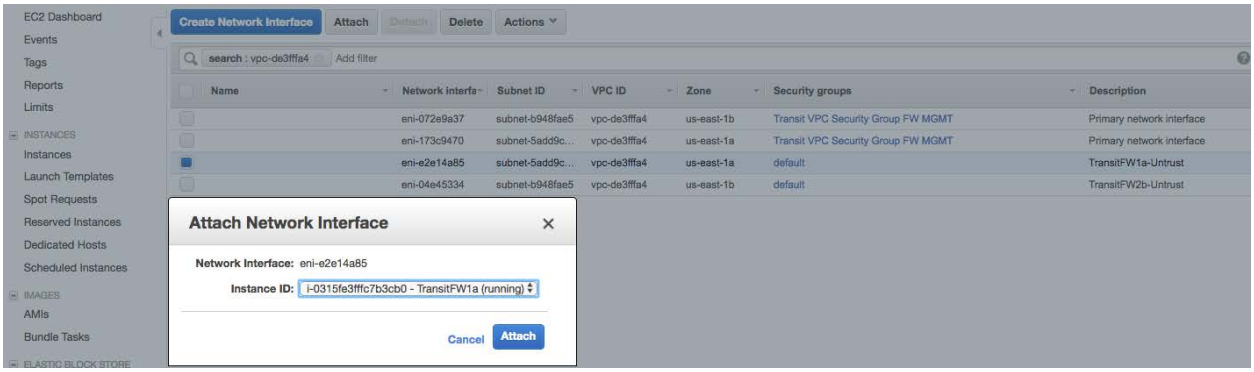


Next, highlight the Network Interface and hit the Actions Drop-Down. Choose the Change Source/Dest Check Option and Disable Source/Destination Check. This allows traffic to route through the firewall even if it is not destined for the firewall's assigned IP address.

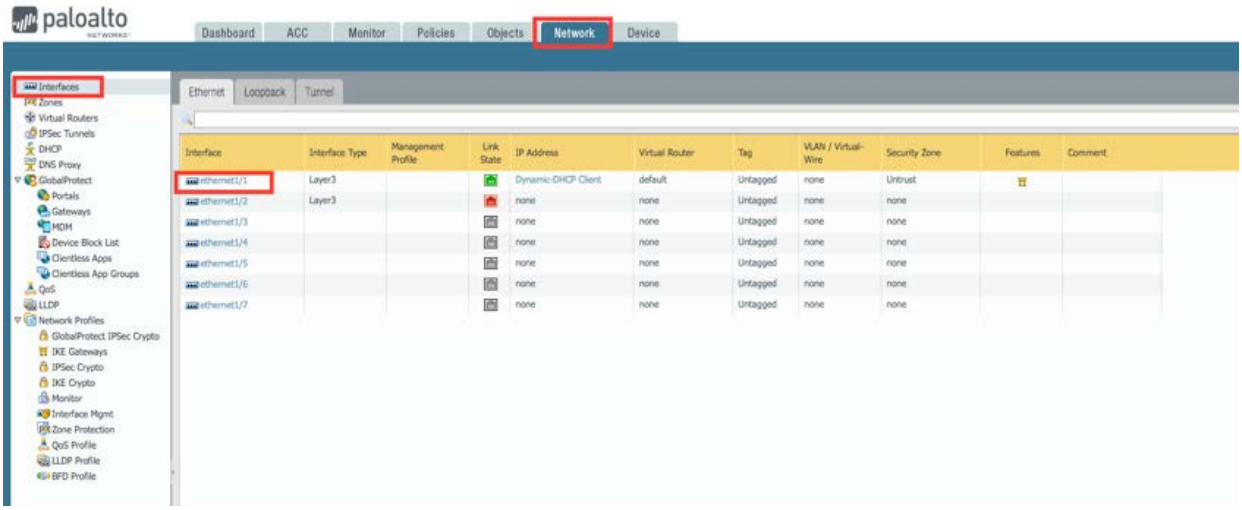




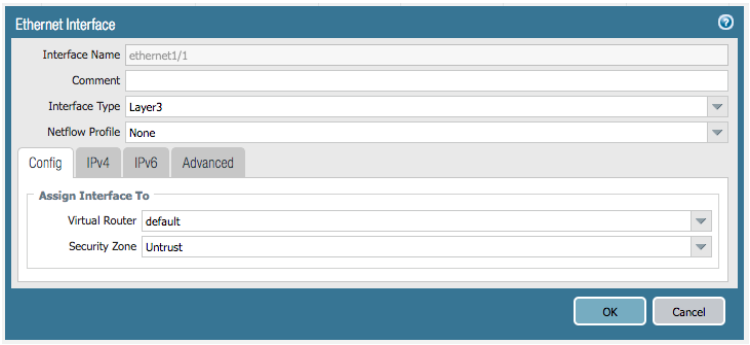
After your interfaces are created, select each interface individually to attach to the firewall instance. Check the box next to the interface and select the Attach button. In the pop up window, open the drop-down menu and select the firewall instance. You will need to do this for both interfaces.



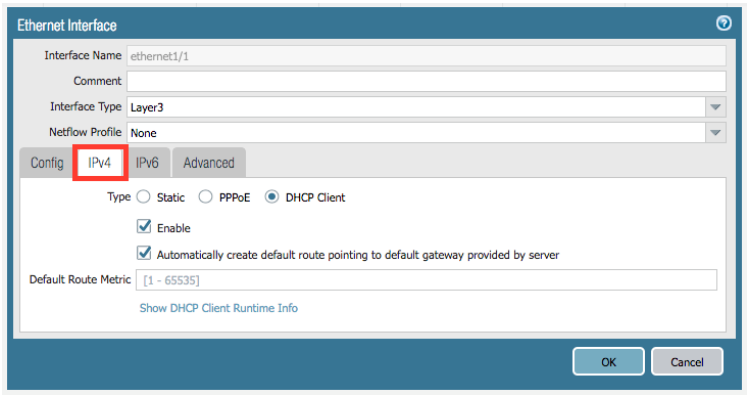
Once the network interfaces have been created, enable the interface on each of the firewalls. Log into each of the firewalls. Once you are in each firewall, go to the network tab and interfaces screen. Select Ethernet 1/1:



From the interface configuration window, configure the interface as below:
NOTE Ensure to select a Virtual Router or you will receive an error on commit.
You can also choose to create your Untrust Security Zone now.



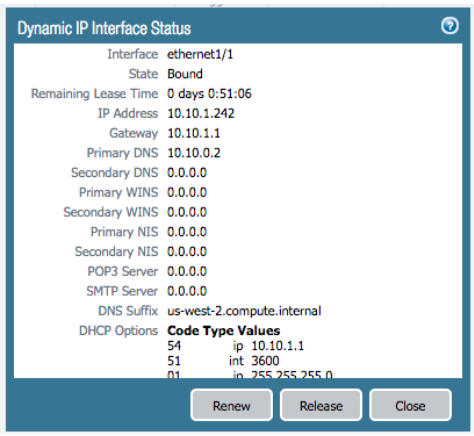
From the IPv4 tab configure as below. Once configured select ok and commit the changes:



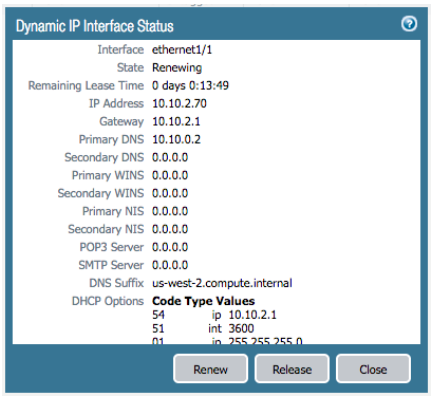
Once the commit completes, you should see the following on your network interface window:

Interface	Interface Type	Management Profile	Link State	IP Address	Virtual Router	Tag	VLAN / Virtual-Wire	Security Zone
ethernet1/1	Layer3			Dynamic-DHCP Client	default	Untagged	none	Untrust

if you click on the Dynamic DHCP Client link you should have a status as the example below:

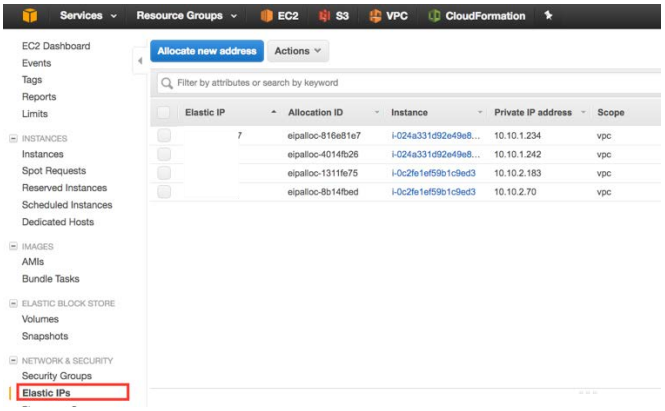


Repeat the above interface configuration for the second firewall. Once complete, you should see a DHCP Client link

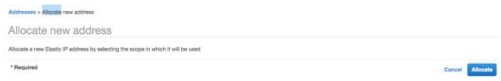


Once you are at this point commit the changes to the firewall.

Next, we will need to create Elastic IP (EIP) addresses for the new dataplane interfaces created in the last step for each firewall. From the EC2 Dashboard, select Elastic IP's and select allocate new address.

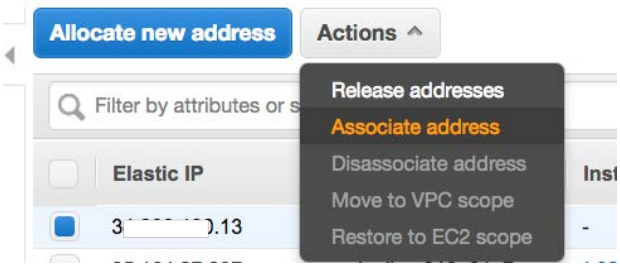


From the Allocate New Address window, click on the allocate button.

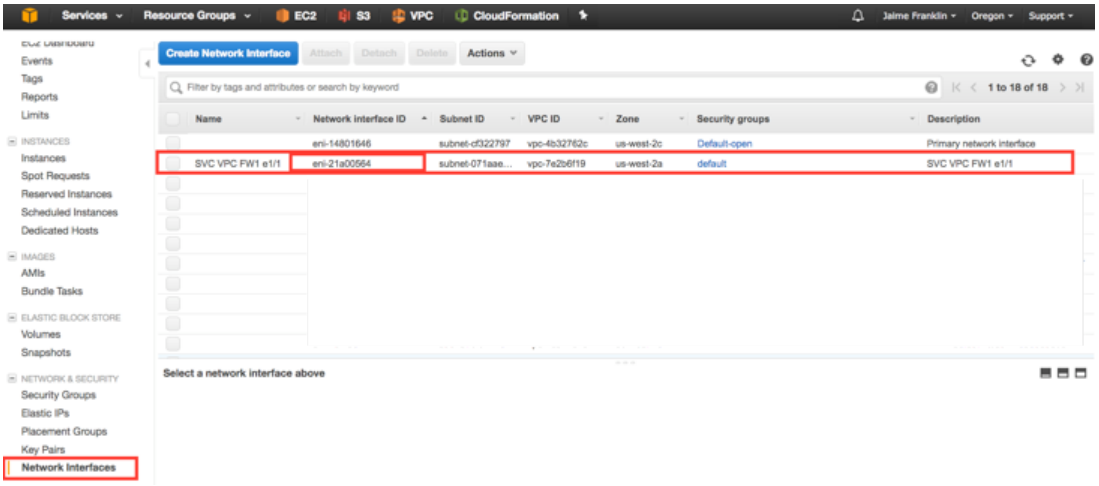


Once you allocate a new address a window will pop up with new EIP that you will associate with the dataplane interface of the firewall.

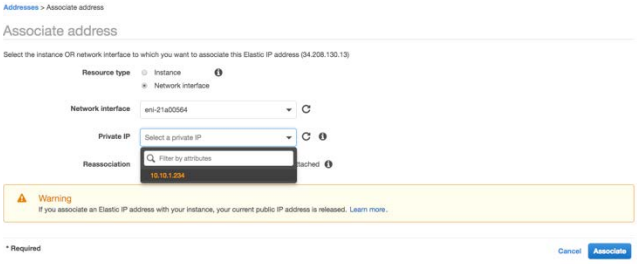
From the Elastic IP's window, select the box next to the new address that was created. From the actions drop down select Associate address:



You will need to document the eni from the AWS interfaces console. Below shows an example of the eni from the EC2 dashboard, network interface menu. You will look for and document the Network Interface ID for eth 1/1 of firewall 1 and firewall 2.



From the associate address window, select resource type, network interface from the network interface drop down select enter the eni ID that was document in the last step for eth 1/1 on the firewalls.

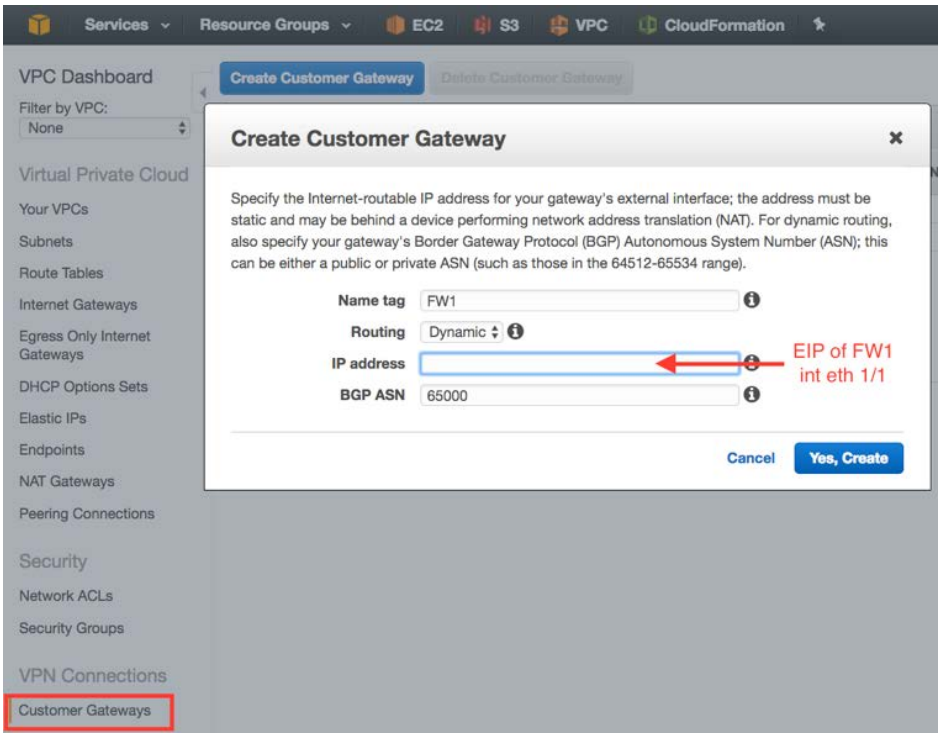


Repeat the above steps for the firewall2 to associate a new EIP to firewall 2.

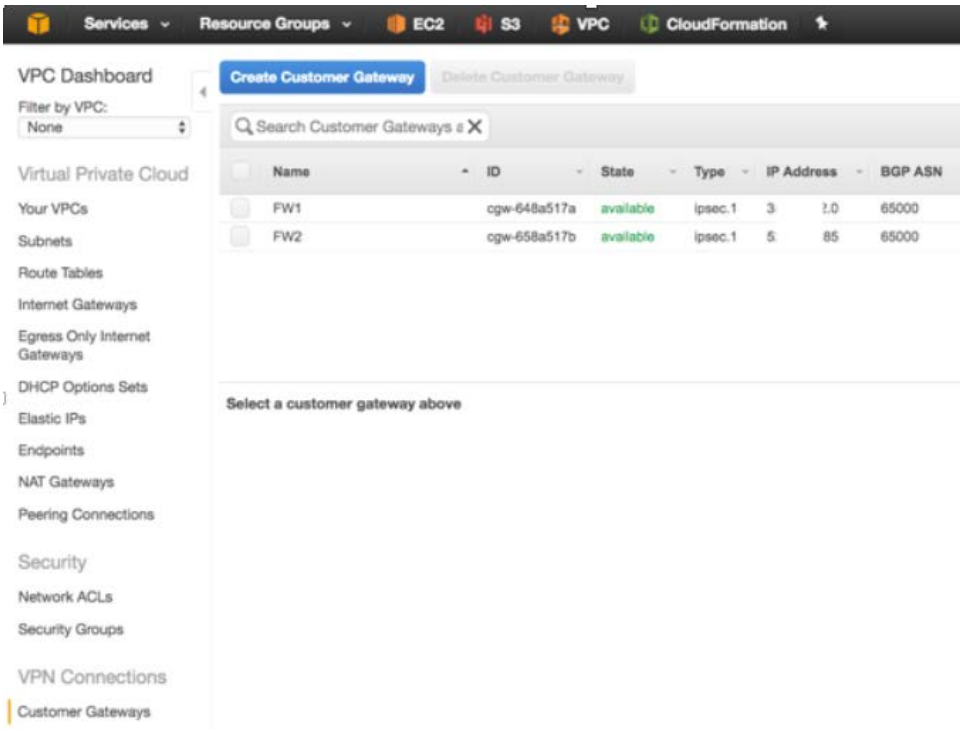
Create Customer Gateways

In this section, we will create the customer gateways for each of the firewalls in the Transit VPC. We will use the EIP address that was just associated on each of the firewall eth 1/1 interfaces.

From the VPC dashboard select Customer Gateways. From the customer gateway window select Create Customer Gateway. Make sure to match up the EIP with the correct firewall.



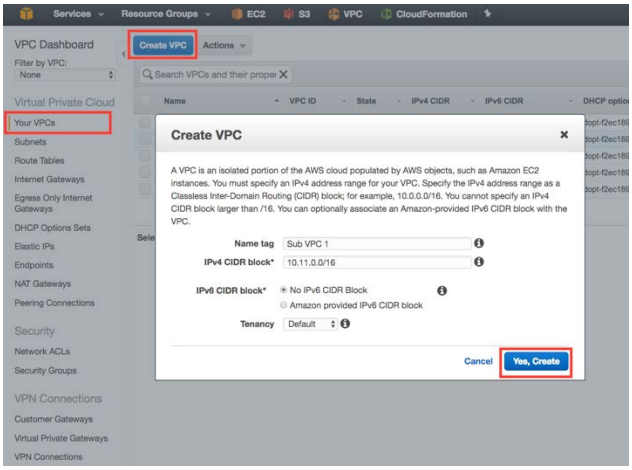
Create a second Customer gateway as above for firewall 2 using the EIP of int eth 1/1 of firewall 2. Your completed screen should look like this:



Section 2 Create the subscribing VPC's

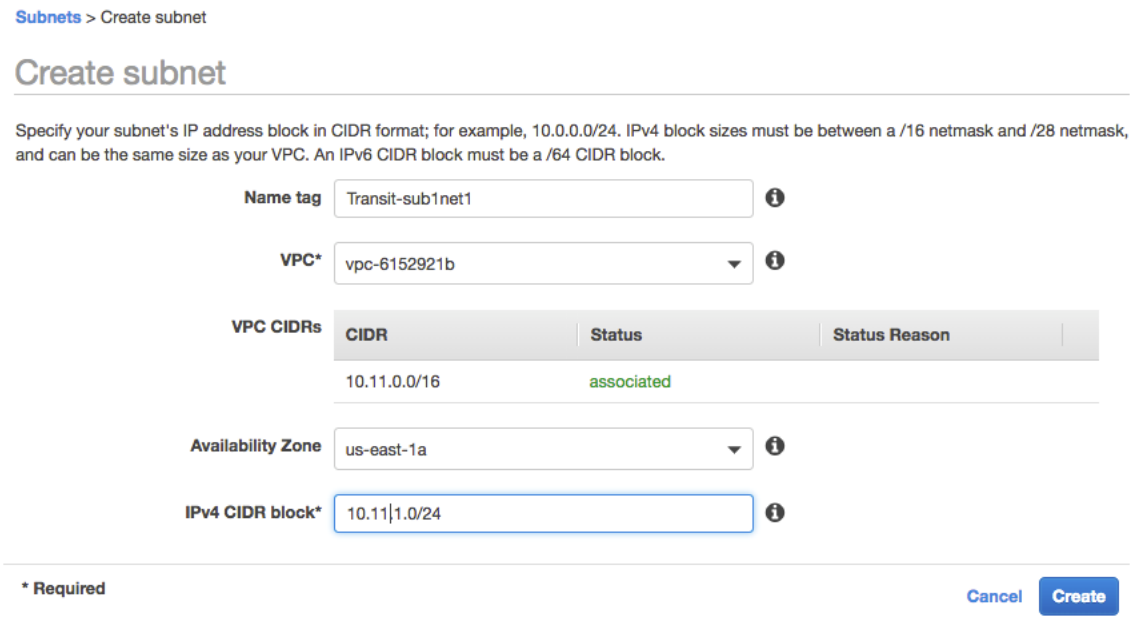
In this section, we will build the (3) subscribing VPC's that will connect back to the Transit VPC via an IPSec connection between the Virtual Gateway (VGW) and the VM-Series firewalls in the Transit VPC. You can configure a second Availability Zone and subnet for each of the subscribing VPC's if you choose.

From the VPC dashboard select the Your VPC's menu, click on the Create VPC button. In the pop up window create a name tag for the VPC and assign a CIDR block. In this example we have used Sub VPC 1- 10.11.0.0/16, Sub VPC 2 10.12.0.0/16 and Sub VPC 3 10.13.0.0/16.



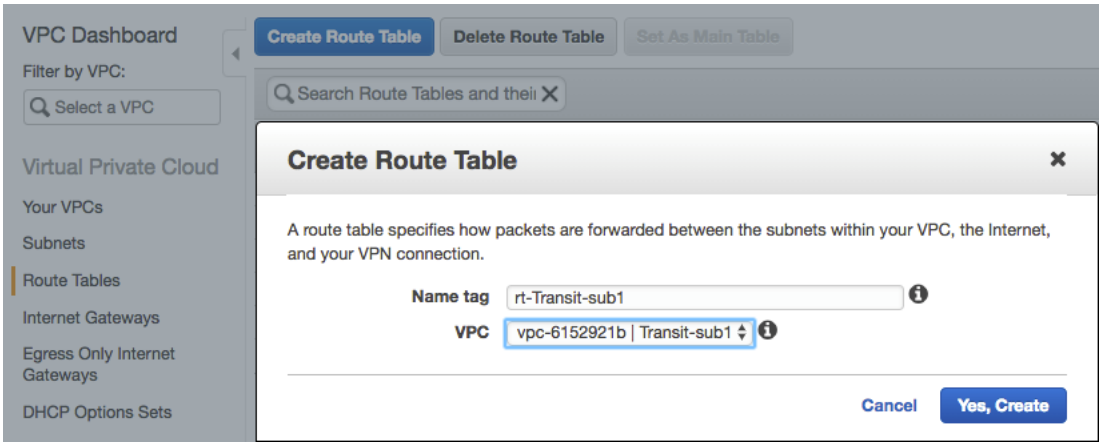
Create the VPC's for Sub VPC 2 and 3 as above with the associated CIDR block.

Next, we will create the Subnets for each of the subscribing VPC's from the VPC Dashboard, select the Subnets menu. From the subnets menu select the Create Subnet button. In the Create Subnet pop up window, create a name for the subnet, select the VPC for the subnet, assign an Availability Zone and assign an IPv4 CIDR block.

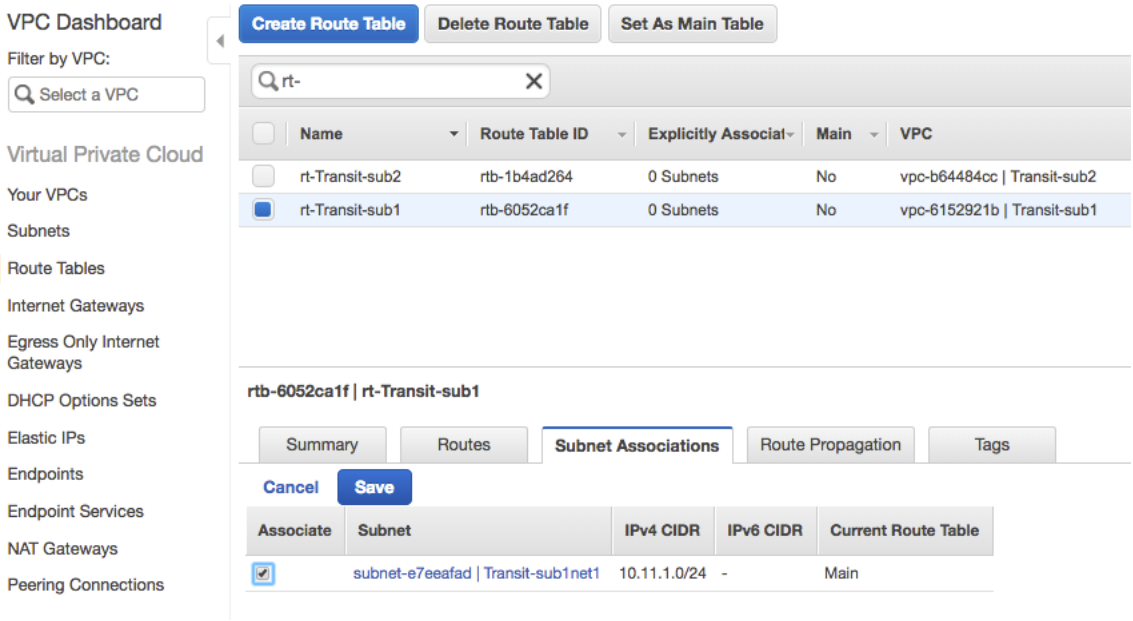


Create (2) more subnets associating them to VPC 2 10.12.1.0/24 and VPC 3 10.13.1.0/24

Next, we will create Route Tables for the new Subscribing VPC's. From the VPC Dashboard, select the Route Tables menu item. From the Route Tables window, select the Create Route Table button. From the Create route table pop up window name your Route table and assign the subscribing VPC from the drop down. Once complete select the Yes, Create button.



Create route table associated to Sub VPC 2 and Sub VPC 3. Once the route tables are created, we want to associate the correct subnets with each of the route tables. From the Route tables menu, select the box next to VPC1 Route Table. From the bottom of the window select the Subnet Associations tab. Click the Edit button where you will click the box next to the subnet created in the last step. Once the subnet has been selected, select the Save button.



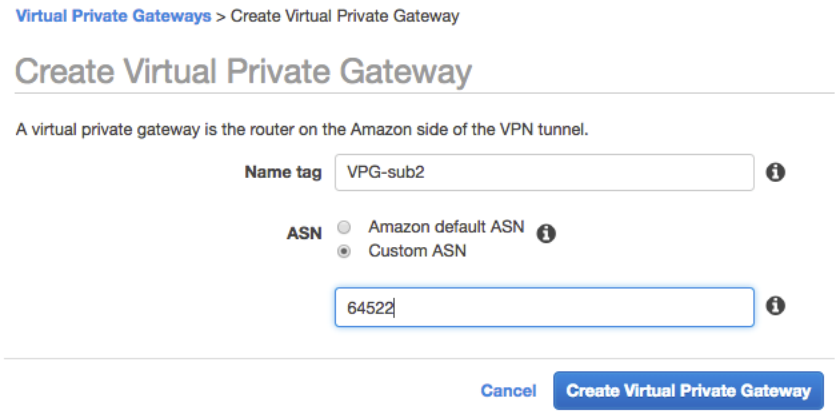
We will revisit the Route tables for the Subscribing VPC's once we complete the VPN configuration.

Next, we are going to create the Virtual Private Gateways for the Subscribing VPC's. From the VPC Dashboard, select the Virtual Private Gateways menu item. From the Virtual Private Gateway window, select the create Virtual Private gateway button. Provide a name for the gateway.

It is recommended to provide a Custom ASN that is unique per subscribing VPC to allow for BGP propagation. The ASN must be in either the 16 bit or 32 bit private range.

- 16 bit: 64512 to 65535
- 32 bit: 4200000000 – 4294967294

Click the Create Virtual Private Gateway button when complete.



Create Gateways for Sub VPC 2 and Sub VPC 3.

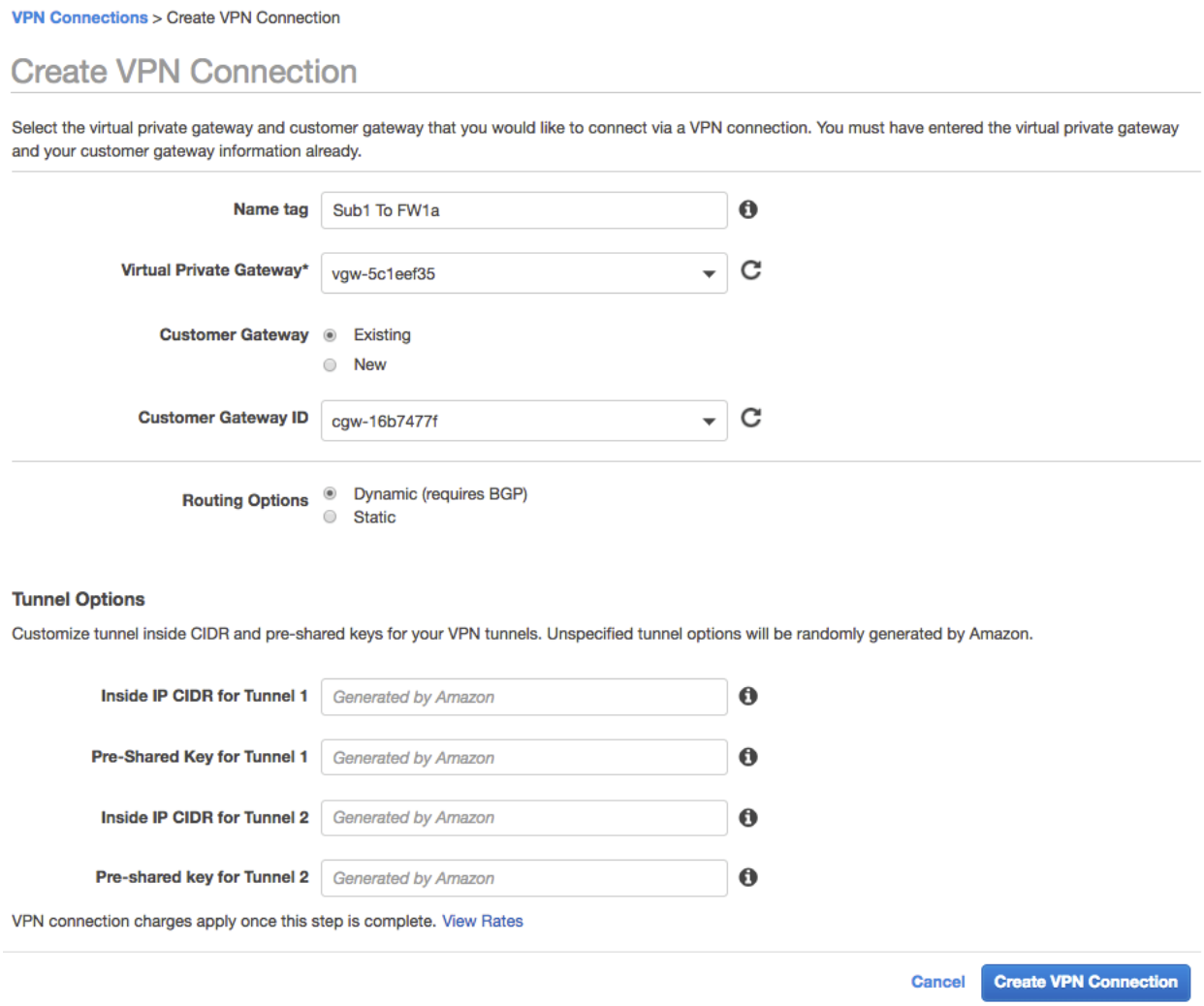
IMPORTANT Ensure to Attach to Virtual Private Gateway to its VPC from the Actions Menu.

Section 3 Create the VPN Connections

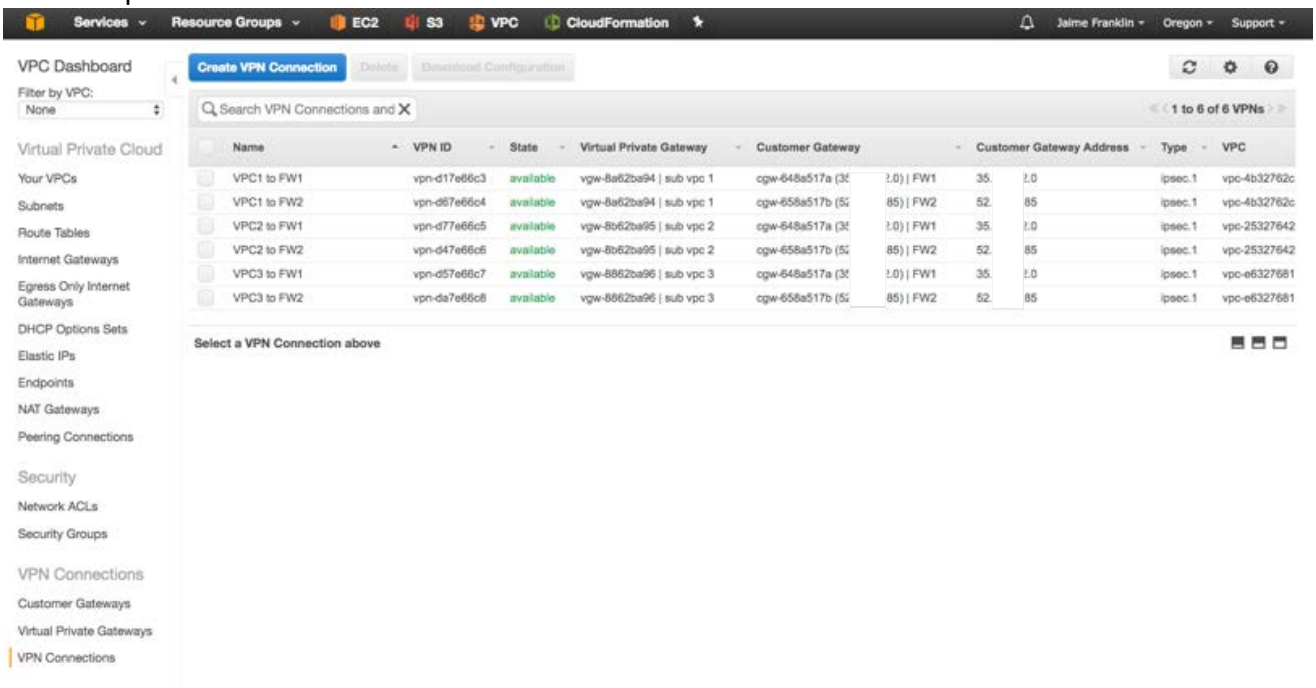
From the VPC Dashboard, select the VPN Connections menu. From the VPN Connections window select the create VPN connection button. In the pop up window, provide a name for the connection. You will need to build a VPN connection from each of the VPC's to both firewalls in the Transit VPC. It is recommended that you use a name descriptor that will be easy to determine which VPC VPN connection belongs to which firewall in the Transit VPC .

NOTE – Amazon now provides the ability the specify the Inside CIDR for the tunnel to prevent overlap. If you will be creating a significant number of spokes, it is recommended to utilize this feature as Amazon has historically had collision issues with more than ~10 spokes.

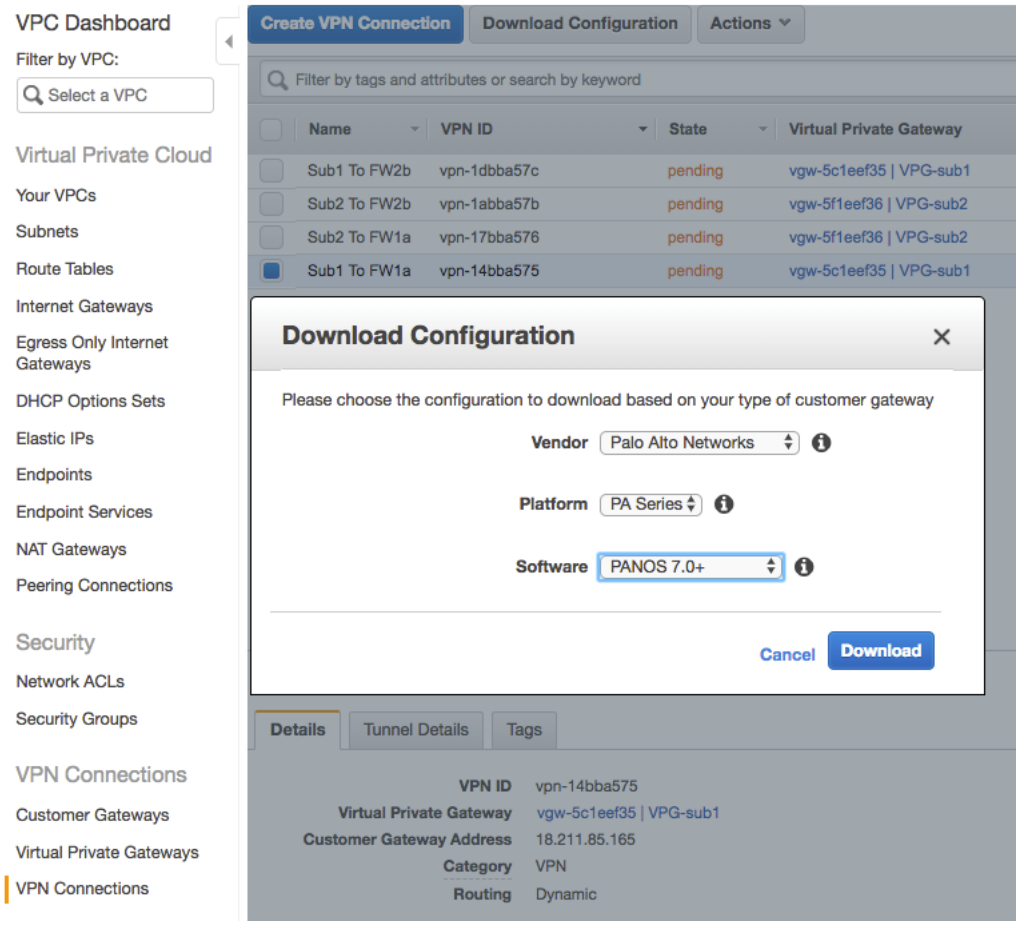
Please see the example below.



You will need to configure a total of 6 VPN Connections for the 3 subscribing VPC's. Please see the example below.



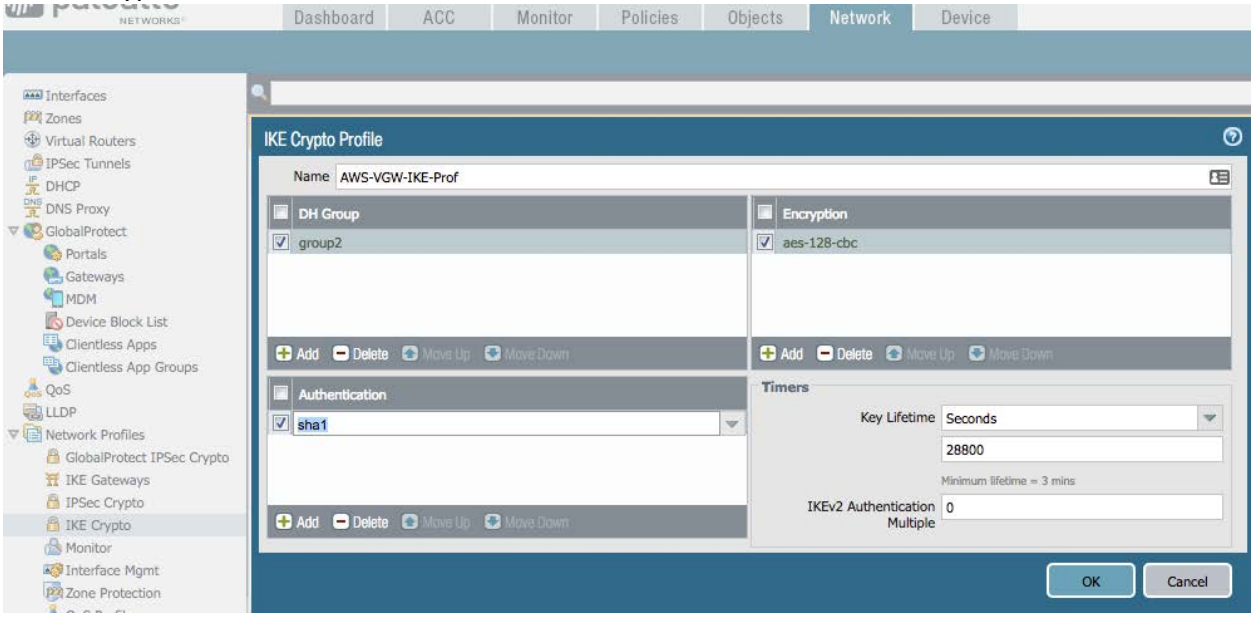
Once the VPN connections have been created, you will need to download the configuration file for each VPN connection. Be very careful to keep each file separate and labeled for which connection it is associated. The configuration file will be used to configure the firewall in PAN-OS. There should be a total of 6 files to download from AWS with configurations for 12 separate tunnels. You will have 6 tunnels per firewall. The files that correspond to firewall 1 needs to be configured on firewall 1. It is recommended that you download the files individually and use them to configure each firewall separately. This will help keep the files and configurations straight as you build the tunnels.


VPC Dashboard
Filter by VPC:
Select a VPC
Virtual Private Cloud
Your VPCs
Subnets
Route Tables
Internet Gateways
Egress Only Internet Gateways
DHCP Options Sets
Elastic IPs
Endpoints
Endpoint Services
NAT Gateways
Peering Connections
Security
Network ACLs
Security Groups
VPN Connections
Customer Gateways
Virtual Private Gateways
VPN Connections
Create VPN Connection
Download Configuration
Actions
Filter by tags and attributes or search by keyword
Name
VPN ID
State
Virtual Private Gateway
Sub1 To FW2b
vpn-1dbba57c
pending
vgw-5c1eef35 | VPG-sub1
Sub2 To FW2b
vpn-1abba57b
pending
vgw-5f1eef36 | VPG-sub2
Sub2 To FW1a
vpn-17bba576
pending
vgw-5f1eef36 | VPG-sub2
Sub1 To FW1a
vpn-14bba575
pending
vgw-5c1eef35 | VPG-sub1
Download Configuration
Please choose the configuration to download based on your type of customer gateway
Vendor
Palo Alto Networks
Platform
PA Series
Software
PANOS 7.0+
Cancel
Download
Details
Tunnel Details
Tags
VPN ID
vpn-14bba575
Virtual Private Gateway
vgw-5c1eef35 | VPG-sub1
Customer Gateway Address
18.211.85.165
Category
VPN
Routing
Dynamic

An example would be naming the above file “VPC 1 firewall1.txt”. Renaming the file will help as you begin to create the IKE Gateways and other steps through this process. You will have 3 files that will provide the VPN configuration for each of the VPC’s connecting to firewall 1 in the Transit VPC. You will have an additional 3 files that will provide the VPN configuration for firewall 2 in the Transit VPC. You will need to keep them separate. The file contains the PAN-OS commands that can be pasted into the console of the firewall from an SSH session or can be configured in the Mgmt Web Interface of the firewall. For this example, we used the Web interface.

Once you have the corresponding VPN configuration file, log into the firewall web interface. From the Network Tab select IKE Crypto and configure as below. These settings can be verified in the downloaded configuration files.

set dh-group group2
set hash sha1
set lifetime seconds 28800
set encryption aes-128-cbc

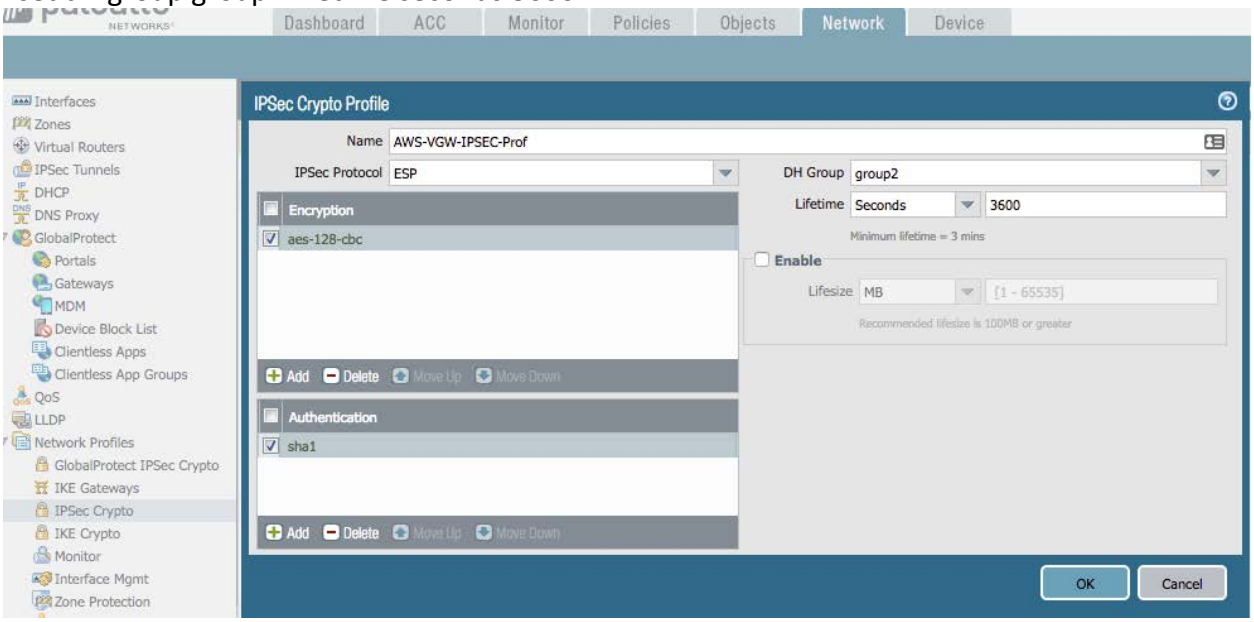


Configure the IPSec Crypto as follows:

set esp authentication sha1

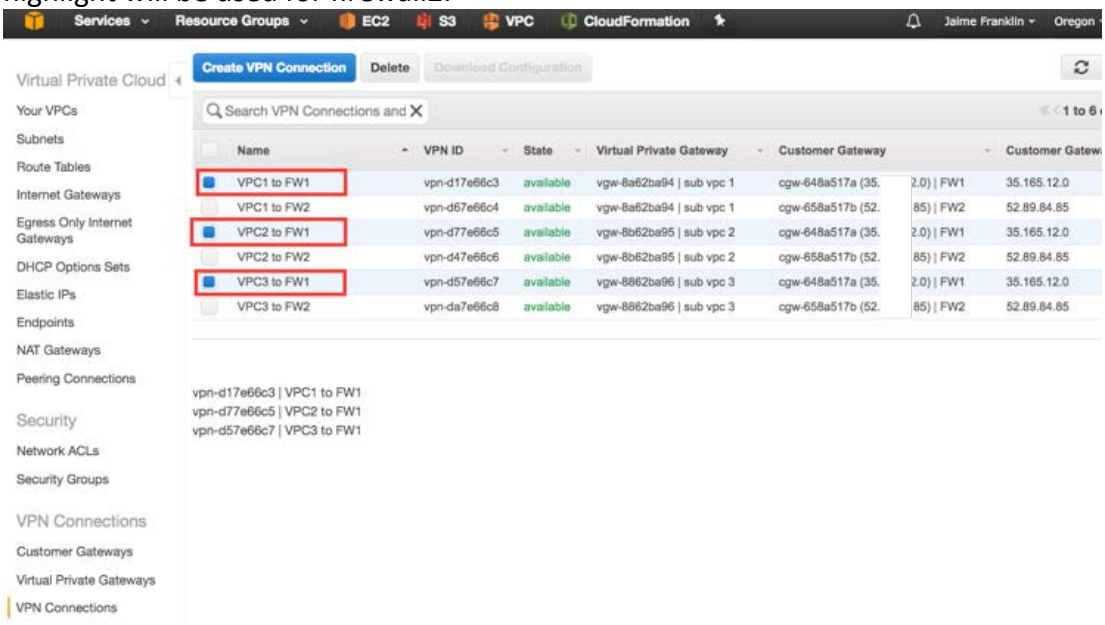
set esp encryption aes-128-cbc

set dh-group group2 lifetime seconds 3600



For the IKE gateway, you will need to build one for each Subscribing VPC Gateway. We used the Name VPC 1-1 and VPC 1-2 as examples to differentiate the gateways. For each of the IKE gateways, you will need to pull the configuration from the VPN file downloaded from the VPN Connection.

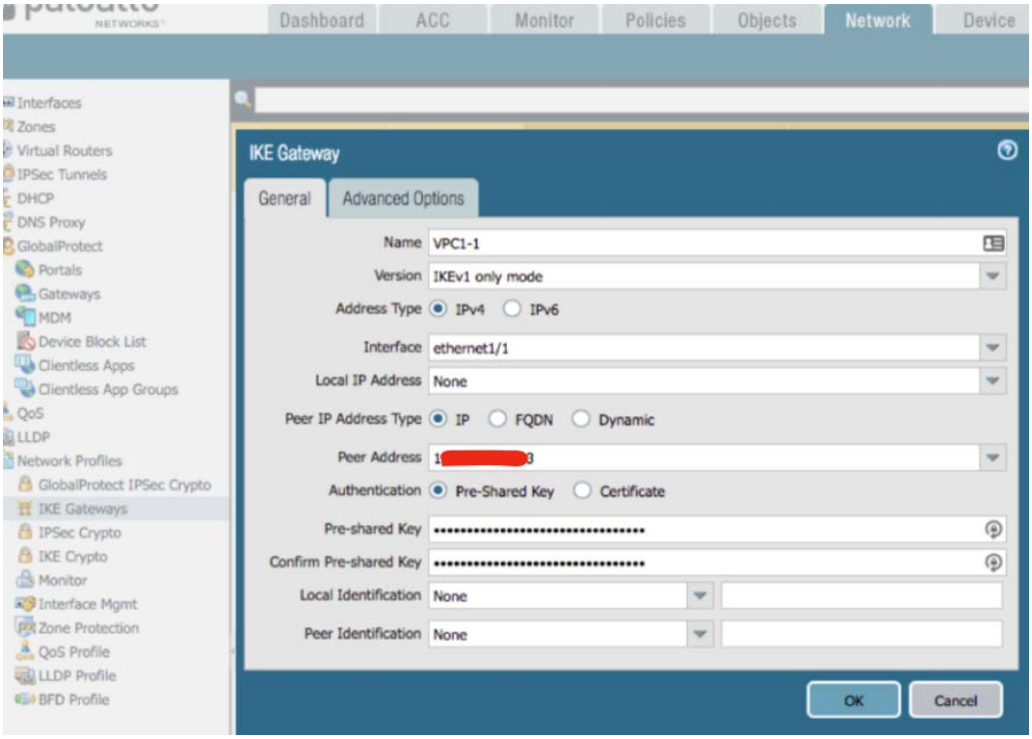
For Gateway 1-1 use the file from the VPN Connection 1. As the screen below shows which files need to be used for firewall 1 and which files will be used for firewall 2. In the file there are configurations for (2) tunnels. You will need to create an IKE gateway for each of the tunnels. The highlighted VPN connection configuration files should be used for firewall1. The 3 files not highlight will be used for firewall2.



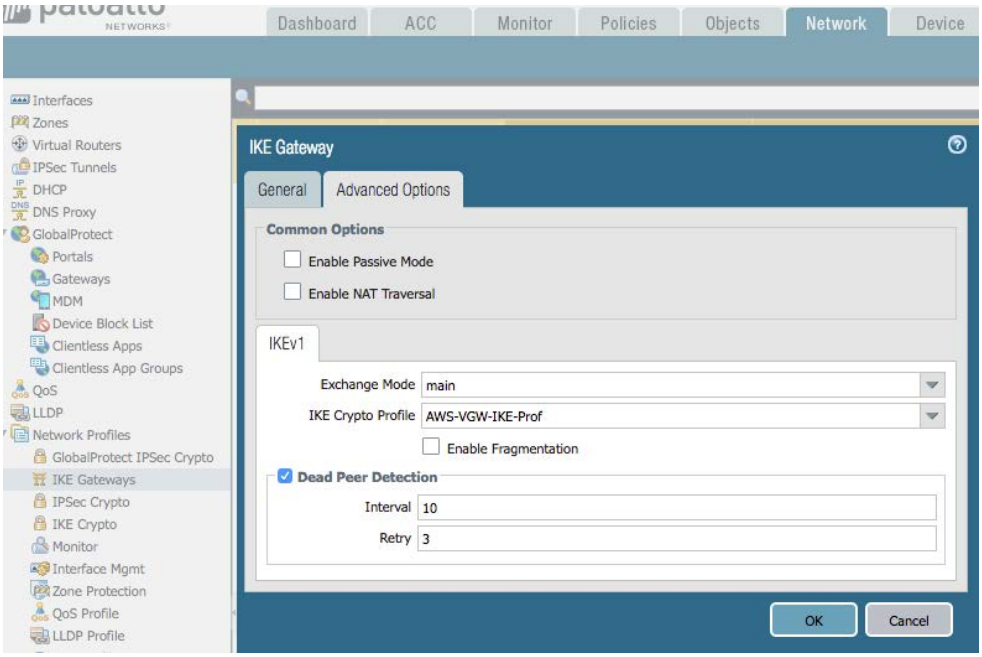
Below is a sample of the configuration file to be used on VPC 1 firewall 1.

```
! ----------------------------------------------------------------------------
! IPSec Tunnel #1
! ----------------------------------------------------------------------------
! #1: Internet Key Exchange (IKE) Configuration
!
! A policy is established for the supported ISAKMP encryption,
! authentication, Diffie-Hellman, lifetime, and key parameters.
! Please note, these sample configurations are for the minimum requirement of AES128,
SHA1, and DH Group 2.
! You will need to modify these sample configuration files to take advantage of AES256,
SHA256, or other DH groups like 2, 14-18, 22, 23, and 24.
! The address of the external interface for your customer gateway must be a static
address.
! Your customer gateway may reside behind a device performing network address translation
(NAT).
! To ensure that NAT traversal (NAT-T) can function, you must adjust your firewall !rules
to unblock UDP port 4500. If not behind NAT, we recommend disabling NAT-T.

!

 configure
 edit network ike crypto-profiles ike-crypto-profiles ike-crypto-vpn-d17e66c3-0
  set dh-group group2
  set hash sha1
  set lifetime seconds  28800
  set encryption aes-128-cbc
  top

! With local-address IP please append the configured subnet mask (i.e, /30) on the VPN
initiating interface (i.e., ethernet 1/1)
! For example if you have /30 as subnet mask the local-address ip should be
35.      0/30

 edit network ike gateway ike-vpn-d17e66c3-0
  set protocol ikev1 ike-crypto-profile ike-crypto-vpn-d17e66c3-0 exchange-mode main
  set protocol ikev1 dpd interval 10 retry 3 enable yes
  set authentication pre-shared-key key N                              T
  set local-address ip 35.      .0
  set local-address interface ethernet1/1
  set peer-address ip 35.      .179
 top
```

IKE Gateway General Tab:



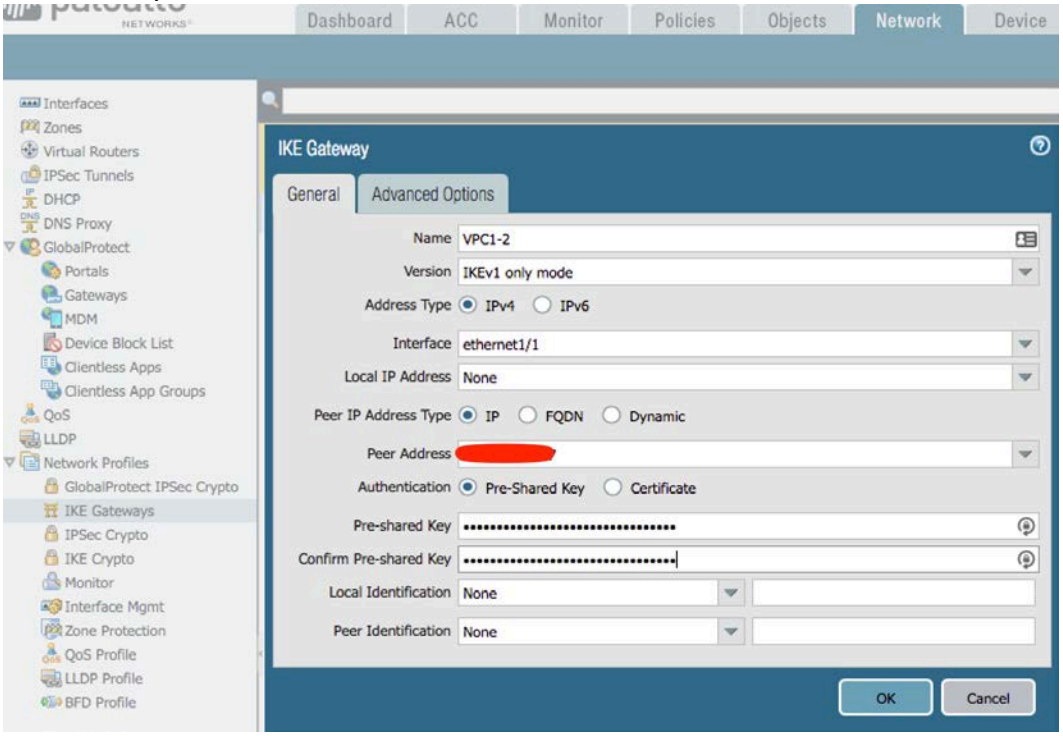
Advanced Options tab:



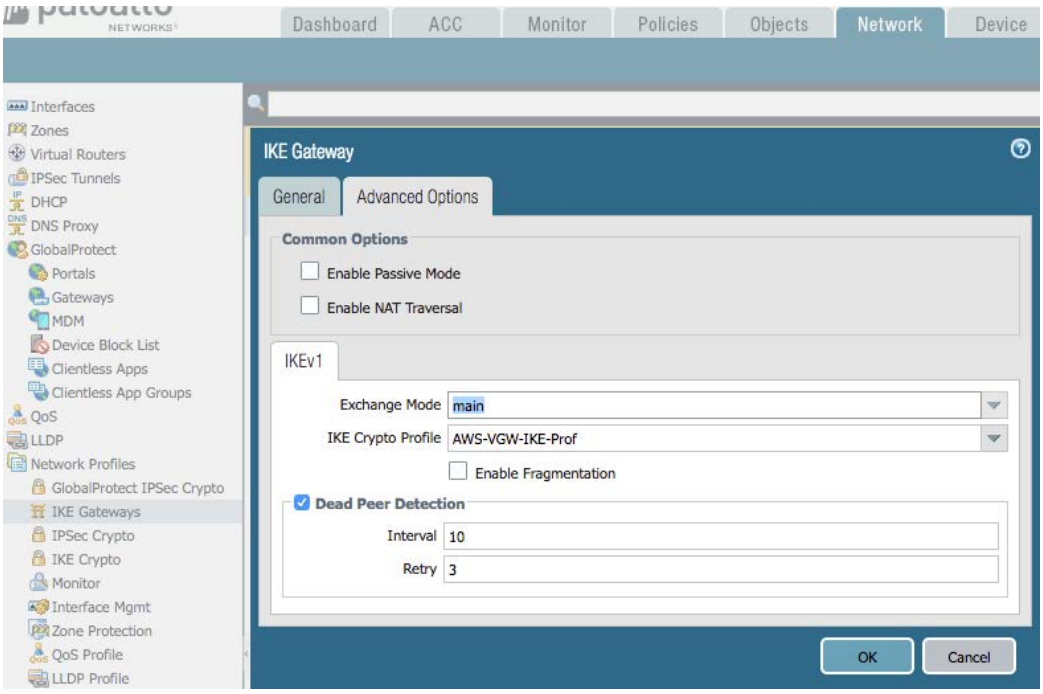
For tunnel 2 you will need to use the same file as above, but create a new IKE Gateway with the tunnel 2 parameters from the bottom portion of the file:

```
!
! ----------------------------------------------------------------------------
! IPSec Tunnel #2
! ----------------------------------------------------------------------------
! #1: Internet Key Exchange (IKE) Configuration
!
! A policy is established for the supported ISAKMP encryption,
! authentication, Diffie-Hellman, lifetime, and key parameters.
! Please note, these sample configurations are for the minimum requirement of AES128,
SHA1, and DH Group 2.
! You will need to modify these sample configuration files to take advantage of AES256,
SHA256, or other DH groups like 2, 14-18, 22, 23, and 24.
! The address of the external interface for your customer gateway must be a static
address.
! Your customer gateway may reside behind a device performing network address translation
(NAT).
! To ensure that NAT traversal (NAT-T) can function, you must adjust your firewall !rules
to unblock UDP port 4500. If not behind NAT, we recommend disabling NAT-T.

!

 configure
 edit network ike crypto-profiles ike-crypto-profiles ike-crypto-vpn-d17e66c3-1
  set dh-group group2
  set hash sha1
  set lifetime seconds  28800
  set encryption aes-128-cbc
  top

! With local-address IP please append the configured subnet mask (i.e, /30) on the VPN
initiating interface (i.e., ethernet 1/1)
! For example if you have /30 as subnet mask the local-address ip should be
35.      .0/30

 edit network ike gateway ike-vpn-d17e66c3-1
  set protocol ikev1 ike-crypto-profile ike-crypto-vpn-d17e66c3-1 exchange-mode main
  set protocol ikev1 dpd interval 10 retry 3 enable yes
  set authentication pre-shared-key key 7                                  5
  set local-address ip 35.      ?.0
  set local-address interface ethernet1/1
  set peer-address ip 35.      .221
 top
```

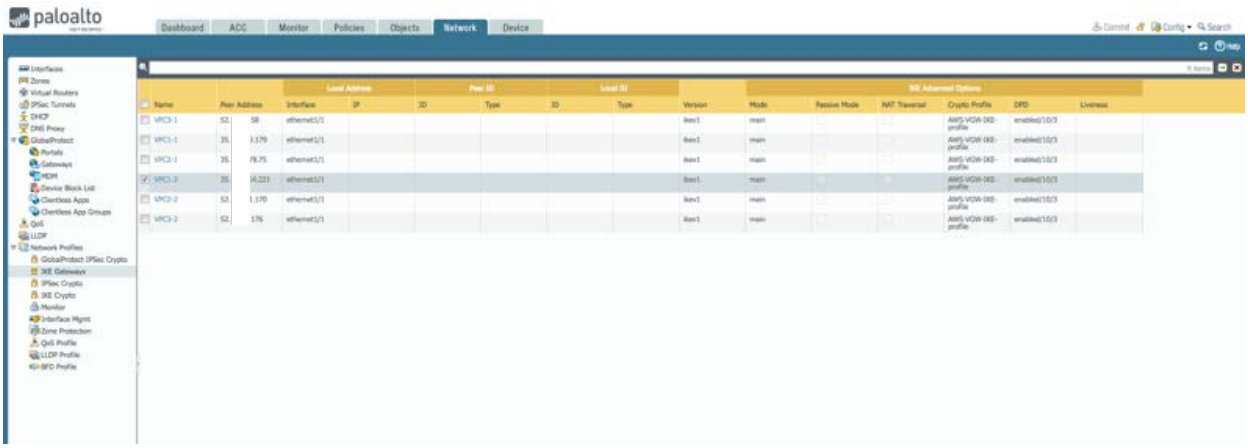
IKE Gateway 2 General Tab



Advanced Tab:



You will need to complete the above steps for Sub VPC 2 and Sub VPC 3 so that your final configuration looks like the example below:

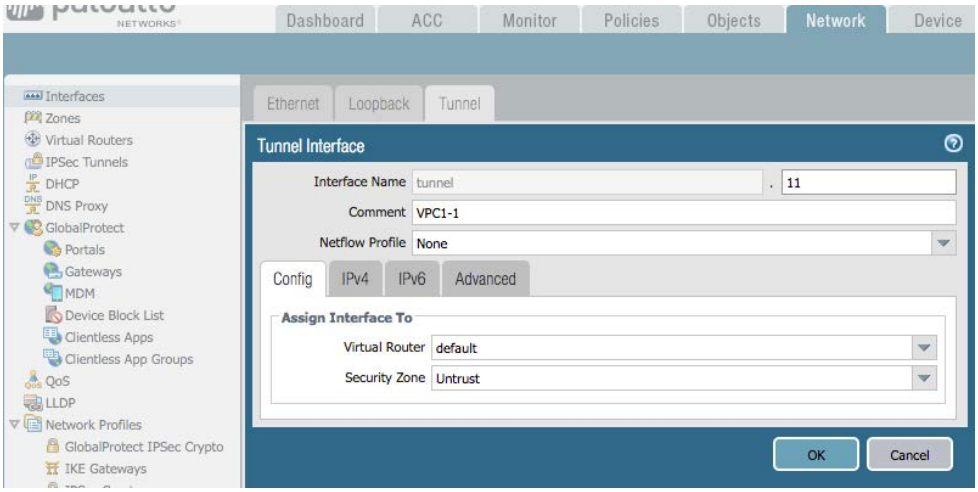


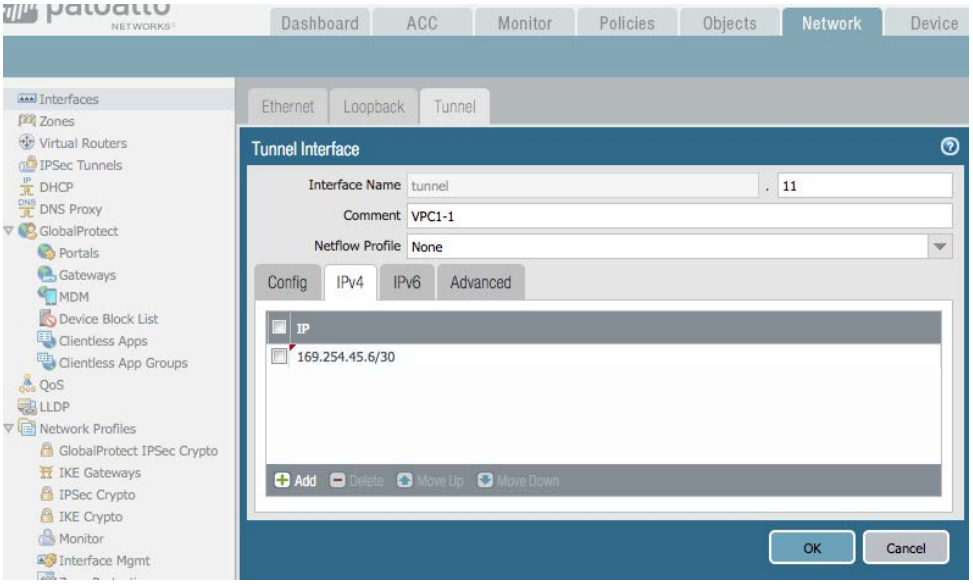
Next, we will create the tunnel interfaces on the VM-Series. From the network tab select the Interface menu. From the tunnel tab select Add from the bottom of the window and configure the tunnels as shown below:

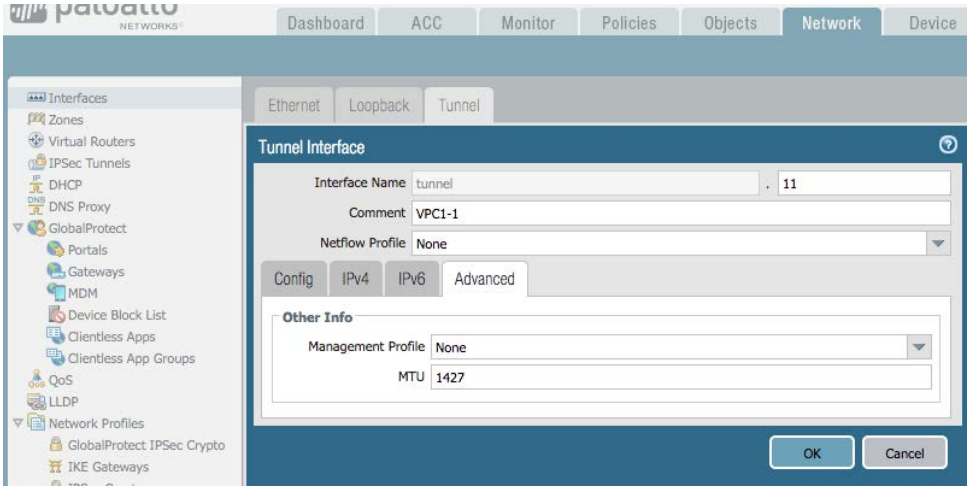
From the VPN config file for the VPN connection use the tunnel interface config section of the document. You will have a tunnel interface for tunnel 1 and a tunnel interface for tunnel 2 from the VPN Document downloaded from AWS.

NOTE – This example shows all tunnel interfaces in the Untrust Zone which will all traffic to be allowed by Virtue of the "intrazone-default" security policy. The reader has the option of controlling traffic with subnet-based policies or mapping the Tunnel interfaces to differing Zones. For more information on Zones, please refer to this guide. https://www.paloaltonetworks.com/documentation/81/pan-os/pan-os/getting-started/segment-your-network-using-interfaces-and-zones

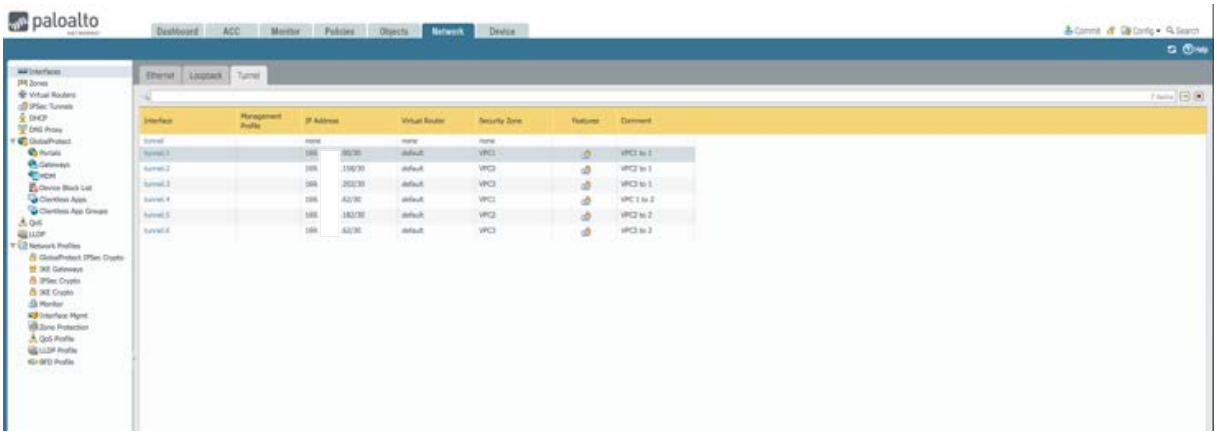
```
! ------------------------------------------------------------------------------
! #3: Tunnel Interface Configuration
!
! A tunnel interface is configured to be the logical interface associated
! with the tunnel. All traffic routed to the tunnel interface will be
! encrypted and transmitted to the VPC. Similarly, traffic from the VPC
! will be logically received on this interface.
!
! Association with the IPSec security association is done through the
! "tunnel protection" command.
!
! The address of the interface is configured with the setup for your
! Customer Gateway.  If the address changes, the Customer Gateway and VPN
! Connection must be recreated with Amazon VPC.
!

 edit network interface tunnel units tunnel.1
  set ip 169.       .90/30
  set mtu 1427
 top

!
! Tunnel interface needs to be associated to Zone, we are using untrust zone as an
example, please adjust according
!

set zone untrust network layer3 tunnel.1
```


Dashboard
ACC
Monitor
Policies
Objects
Network
Device
Interfaces
Zones
Virtual Routers
IPSec Tunnels
DHCP
DNS Proxy
GlobalProtect
Portals
Gateways
MDM
Device Block List
Clientless Apps
Clientless App Groups
QoS
LLDP
Network Profiles
GlobalProtect IPSec Crypto
IKE Gateways
Ethernet
Loopback
Tunnel
Tunnel Interface
Interface Name
tunnel
11
Comment
VPC1-1
Netflow Profile
None
Config
IPv4
IPv6
Advanced
Assign Interface To
Virtual Router
default
Security Zone
Untrust
OK
Cancel

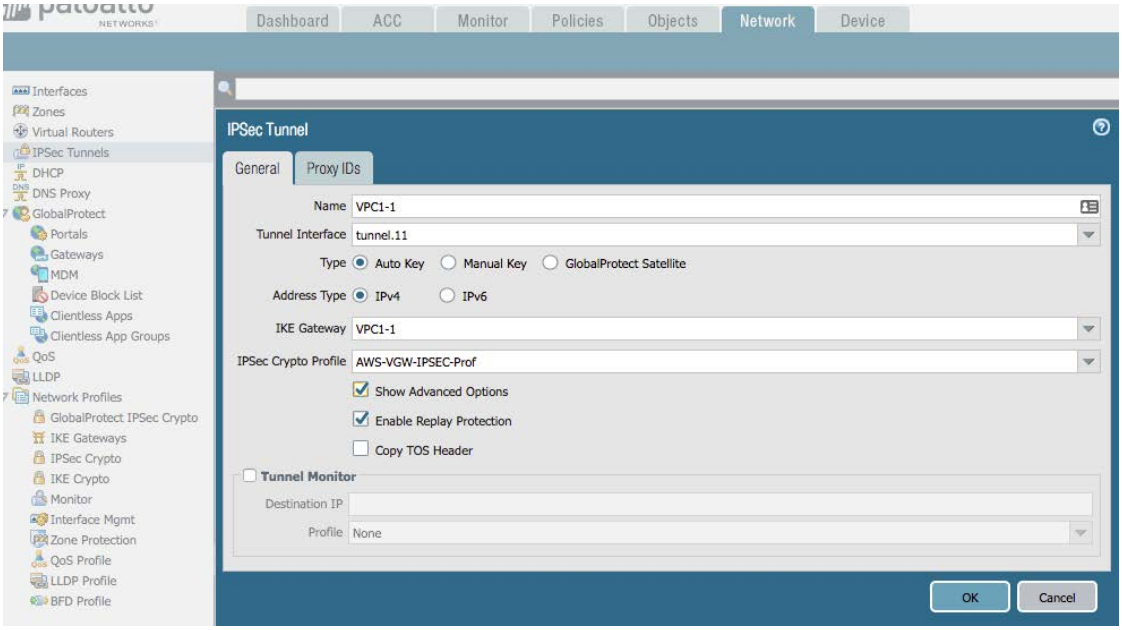

Dashboard
ACC
Monitor
Policies
Objects
Network
Device
Interfaces
Zones
Virtual Routers
IPSec Tunnels
DHCP
DNS Proxy
GlobalProtect
Portals
Gateways
MDM
Device Block List
Clientless Apps
Clientless App Groups
QoS
LLDP
Network Profiles
GlobalProtect IPSec Crypto
IKE Gateways
IPSec Crypto
IKE Crypto
Monitor
Interface Mgmt
Ethernet
Loopback
Tunnel
Tunnel Interface
Interface Name
tunnel
11
Comment
VPC1-1
Netflow Profile
None
Config
IPv4
IPv6
Advanced
IP
169.254.45.6/30
Add
Delete
Move Up
Move Down
OK
Cancel


Dashboard
ACC
Monitor
Policies
Objects
Network
Device
Interfaces
Zones
Virtual Routers
IPSec Tunnels
DHCP
DNS Proxy
GlobalProtect
Portals
Gateways
MDM
Device Block List
Clientless Apps
Clientless App Groups
QoS
LLDP
Network Profiles
GlobalProtect IPSec Crypto
IKE Gateways
Ethernet
Loopback
Tunnel
Tunnel Interface
Interface Name
tunnel
11
Comment
VPC1-1
Netflow Profile
None
Config
IPv4
IPv6
Advanced
Other Info
Management Profile
None
MTU
1427
OK
Cancel

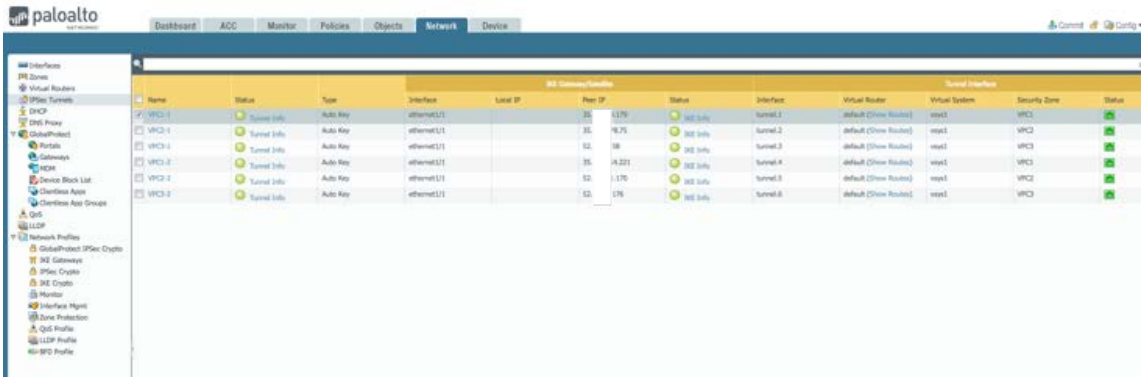
You will create (2) tunnels per Subscribing VPC documented in the VPN connection file. Your configuration should look like the example below.



Now we will create the IPSec tunnels on the VM Series. From the network tab and the IPSec menu, select add at the bottom of the window to create a new IPSec tunnel.



Repeat the steps and match tunnel interfaces to the correct IKE gateways that have been created. Your IPSec Tunnels should look like the following example.



Once you are to this point, commit the configuration to the firewall.

To validate the tunnels, from the CLI of the firewall, issue the following command:
show vpn ipsec-sa

```
admin@PA-VM> show vpn ipsec-sa

GwID/client IP  TnID   Peer-Address          Tunnel(Gateway)                Algorithm         SPI(in)  SPI(out) life(Sec/KB)
--------------  ----   ------------          ---------------                ---------         -------  -------- ------------
2               1      35.      .179         VPC1-1(VPC1-1)                 ESP/A128/SHA1     B70A1906 F73B7FB1 644/0
3               2      35.      78.75        VPC2-1(VPC2-1)                 ESP/A128/SHA1     A462A648 65F30F73 644/0
1               3      52.      158          VPC3-1(VPC3-1)                 ESP/A128/SHA1     9CD3CA26 B9913997 645/0
4               4      35.      64.221       VPC1-2(VPC1-2)                 ESP/A128/SHA1     D00AF805 EFEA72E7 1232/0
5               5      52.      1.170        VPC2-2(VPC2-2)                 ESP/A128/SHA1     A86C9EBA 91C18940 1063/0
6               6      52.      .176         VPC3-2(VPC3-2)                 ESP/A128/SHA1     9F6E427A 10BAB5C1 1064/0

Show IPSec SA: Total 6 tunnels found. 6 ipsec sa found.

admin@PA-VM>
```

You should see 6 tunnels active, if you do not see the above, the tunnels may not come up automatically. If not, they can be manually triggered using the CLI.

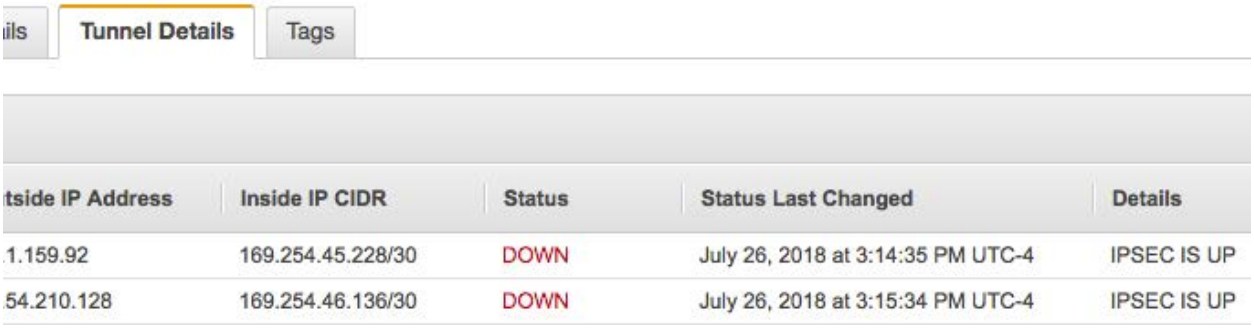
test vpn ipsec-sa

```
admin@SVC-VPC-firewall2> test vpn ipsec-sa

Initiate 6 IPSec SA.
```

If you still do not see the 6 active tunnels after manually triggering the tunnels, validate that you have configured your ike gateways and vpn connections correctly in the above steps.

You will need to complete the same steps on firewall 2 with the VPN files for firewall2.

At this point, the VPN connections will Report IPSEC IP and Status Down in the AWS console. This is due to a lack of traffic on the tunnels. They report up once BGP or other traffic is traversing the tunnels.

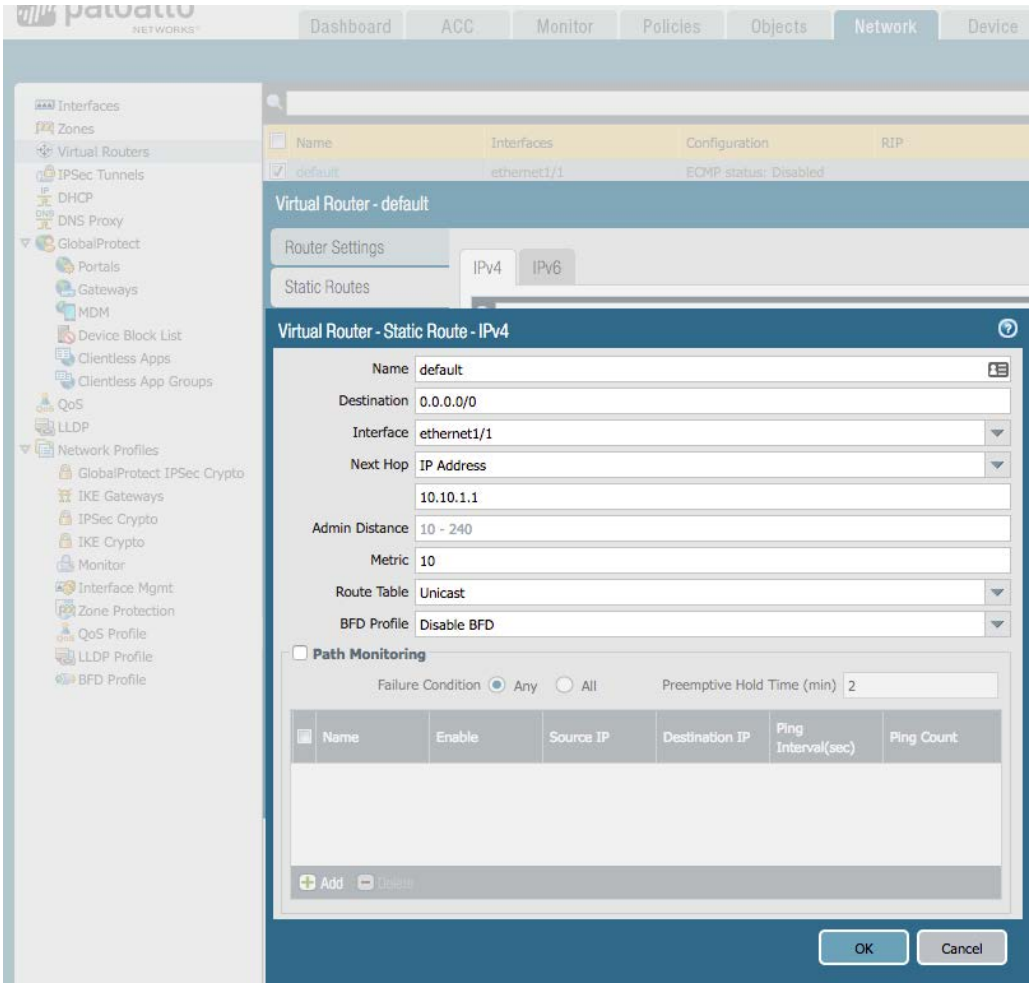


Section 4 Configure BGP

Now that we have our communication between the subscribing VPCs and the Transit VPC established, we will configure a dynamic routing protocol to assist with dynamic route learning for each of the VPC's, fault tolerance and symmetric path.

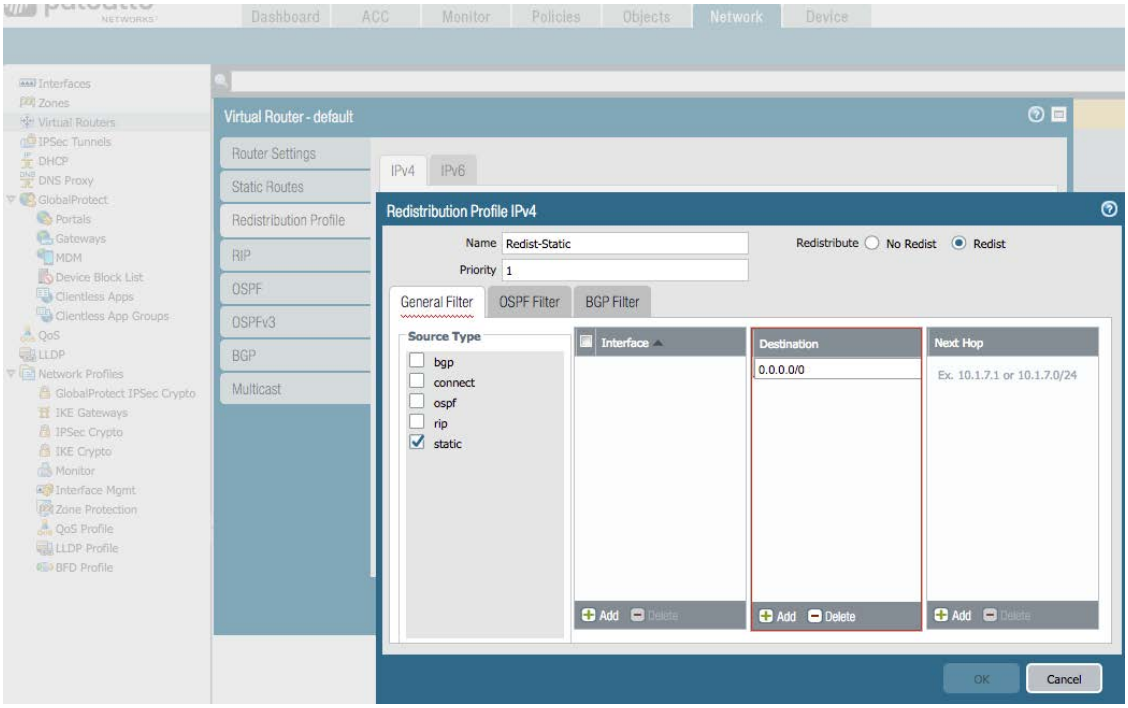
First we will create a static default route. This route will point to the AWS VPC router. From the firewall1 web interface, select the network tab and select the virtual routers menu item. From the window select the default router. You will get a pop-up window for the virtual router. Select the Static Route menu item.

From the pop up window select add at the bottom left of the window to create a new static route. Name the route Default and configure the route as the example below:



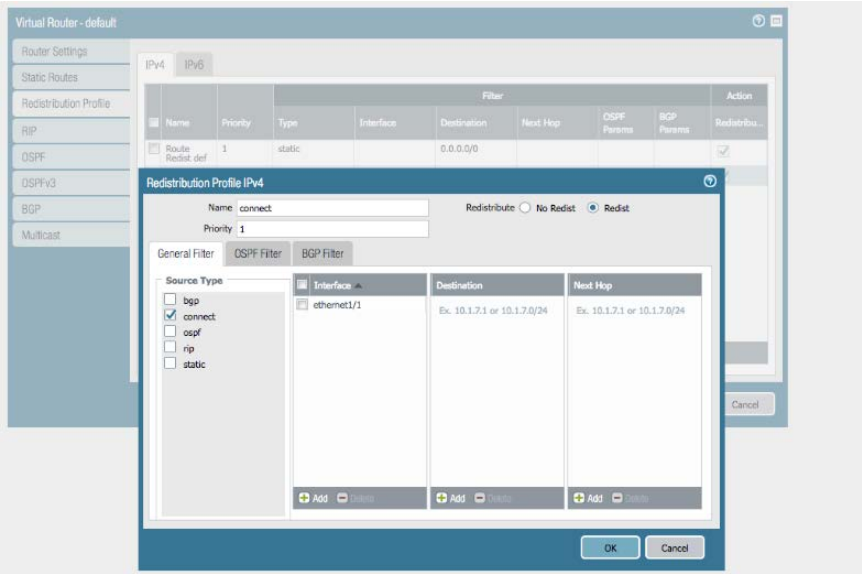
Next, we need to create a Redistribution Profile. From the virtual router, select the Redistribution Profile tab. From the bottom of the virtual router window select add to create a new redistribution profile. The first redistribution profile will create a policy to redistribute the default route. Configure your profile as shown in the example below.

General Tab:

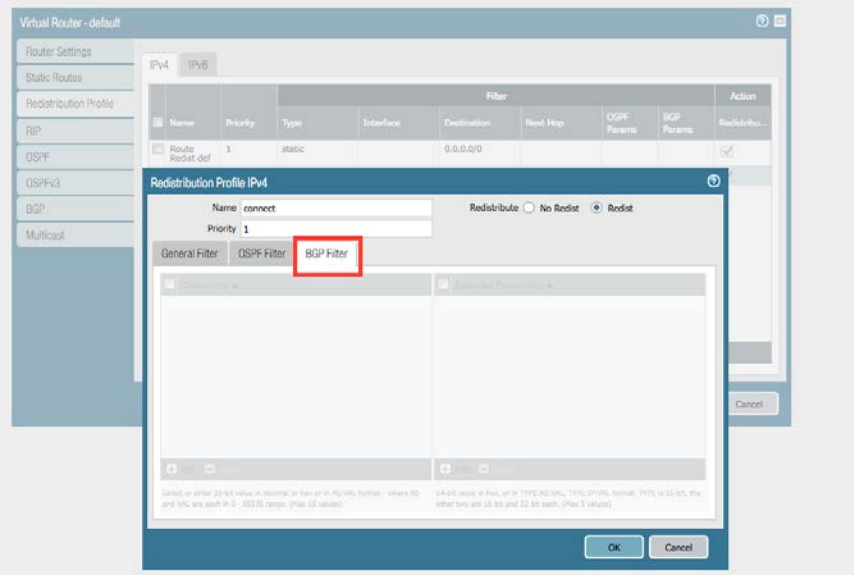


Next we will create a profile to redistribute connected interfaces. From the Redistribution Profile tab select add to create a new profile and configure your connected profile as shown in the example below.

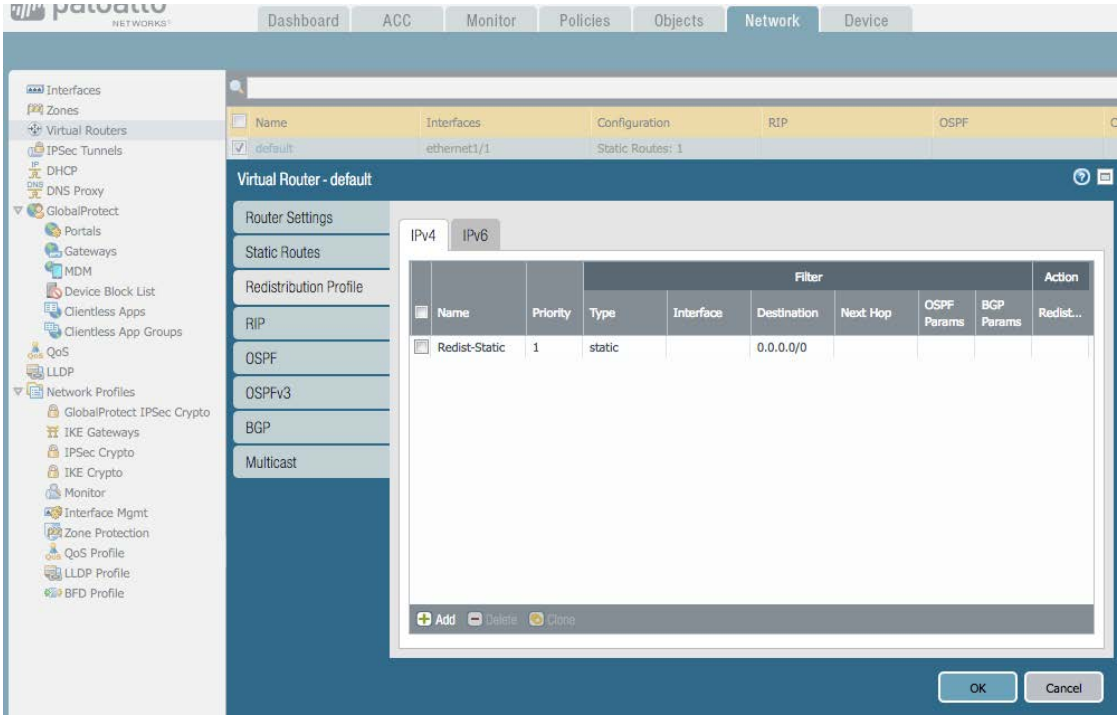
General Filter tab:



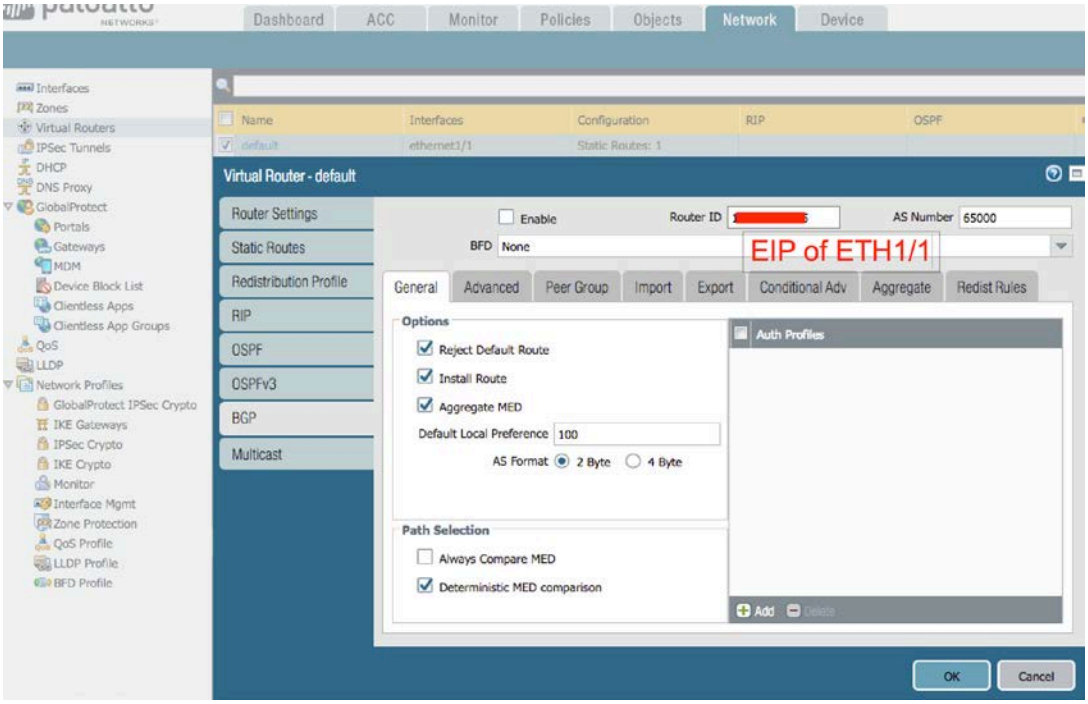
BGP Filter tab:



Your Redistribution Profile window should look like the example below:

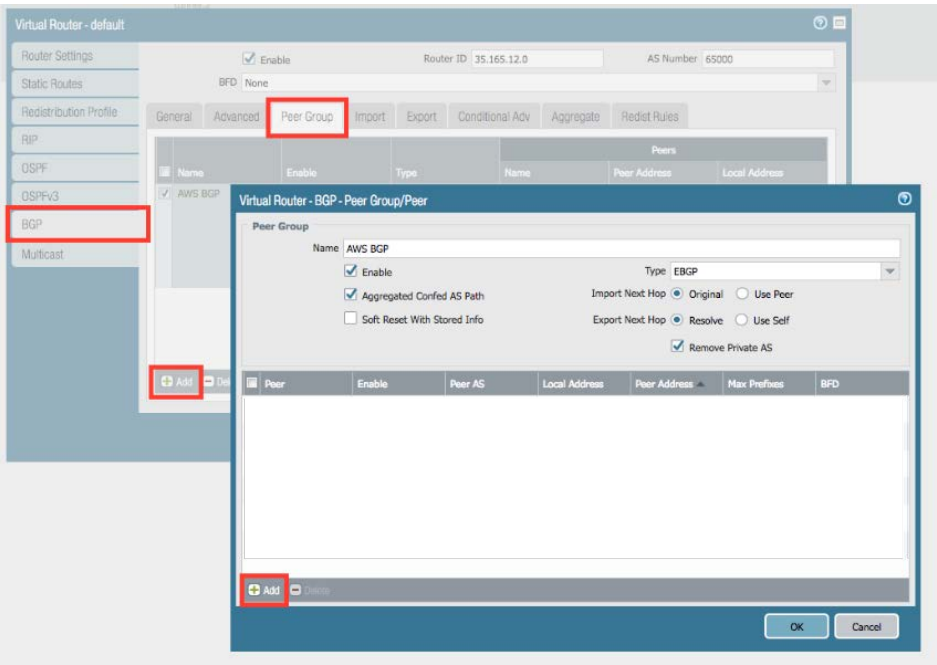


Next, we will enable and configure BGP parameters. From the Virtual router window select the BGP tab. Set the Router IP to the EIP assigned to ETH1 in AWS. Configure the general tab as shown in the example below.



Next, we are going to create a BGP peer group. From the BGP tab select the Peer Group tab. At the bottom on the BGP window, click add to create a new peer group. As you create each peer in the group, you will have to match up the local IP address to the correct peer address. These addresses can be found in the AWS VPN configuration files. As you will recall the AWS VPN configuration file contains configuration information for (2) tunnels. You will need the local address and peer address for each tunnel. Below is an example of the config file:

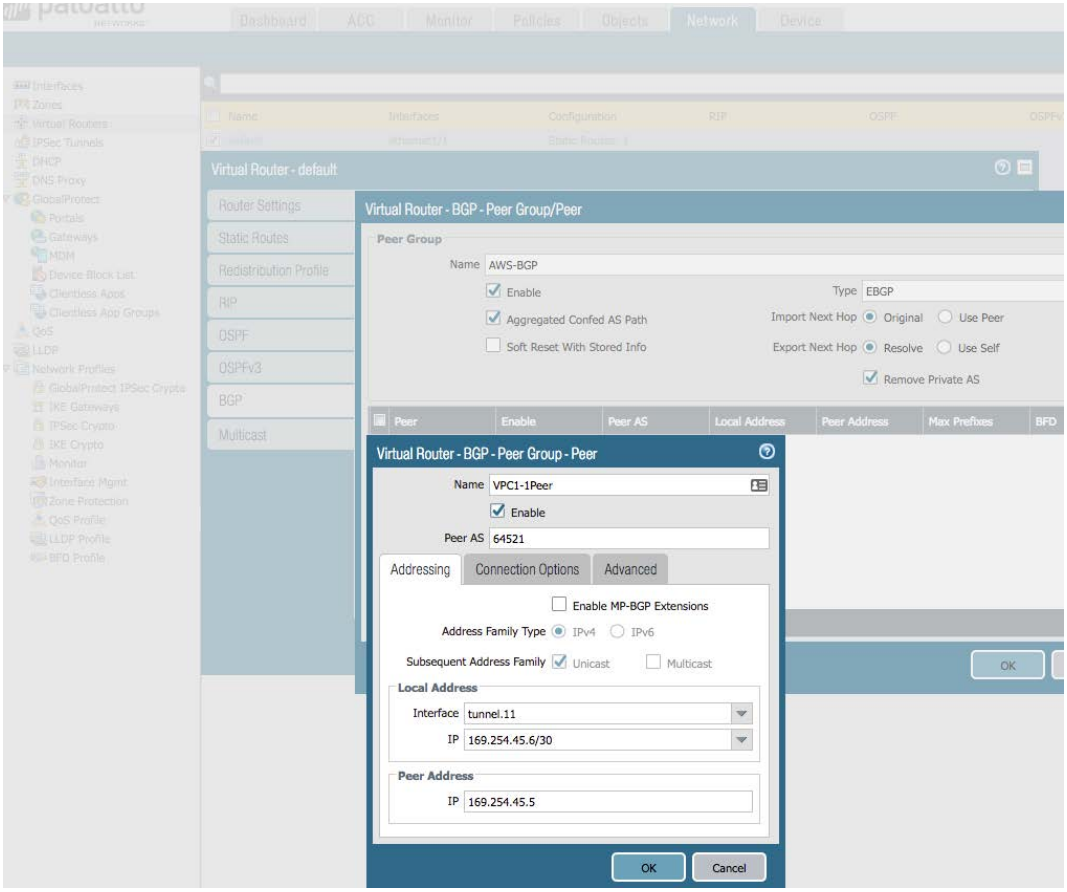
Configure the peer group as shown in the example below.



From the Add pop up window, configure the peer as shown in the example below:

```
! ------------------------------------------------------------------------------
! #3: Tunnel Interface Configuration
!
! A tunnel interface is configured to be the logical interface associated
! with the tunnel. All traffic routed to the tunnel interface will be
! encrypted and transmitted to the VPC. Similarly, traffic from the VPC
! will be logically received on this interface.
!
! Association with the IPSec security association is done through the
! "tunnel protection" command.
!
! The address of the interface is configured with the setup for your
! Customer Gateway.  If the address changes, the Customer Gateway and VPN
! Connection must be recreated with Amazon VPC.
!

edit network interface tunnel units tunnel.1
 set ip 169.      .90/30
 set mtu 1427
top
```

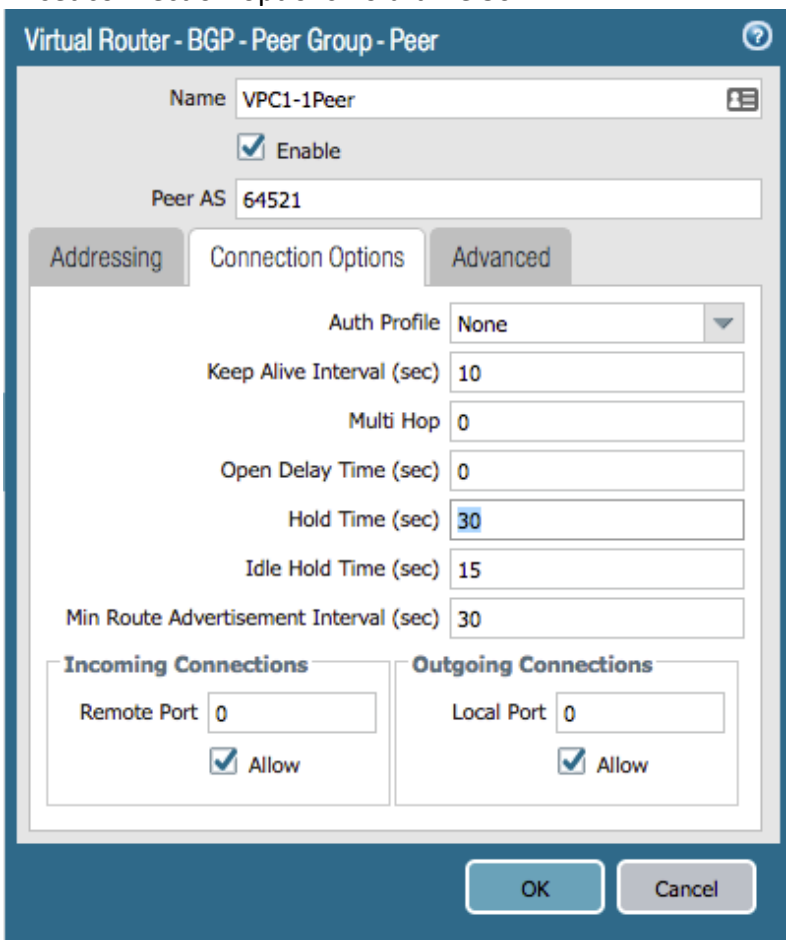
The example abovedisplays information for VPC 1, firewall 1, Tunnel 1. The local address of the firewall is 169.x.x.90/30 and the peer address will be 169.x.x.89. You will need to scroll down the file to get the local and peer address for the second tunnel for VPC 1, firewall1, tunnel 2. See the example below



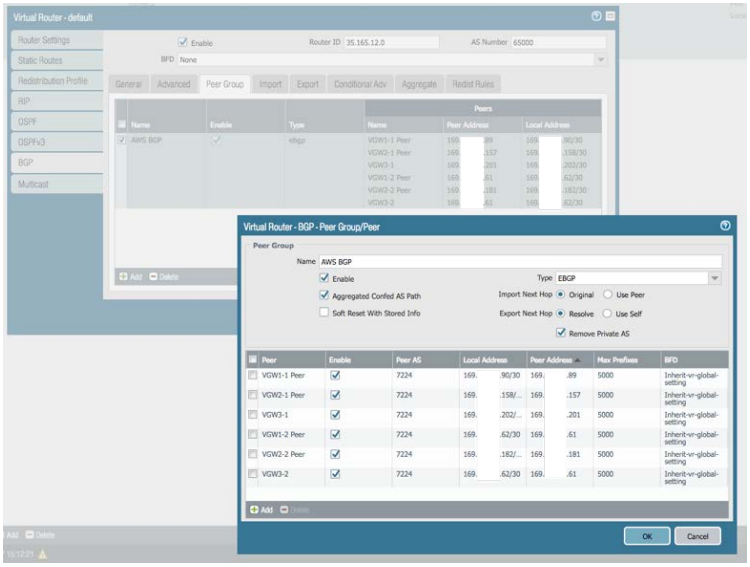
On the Connections Options Tab:

set connection-options keep-alive-interval 10

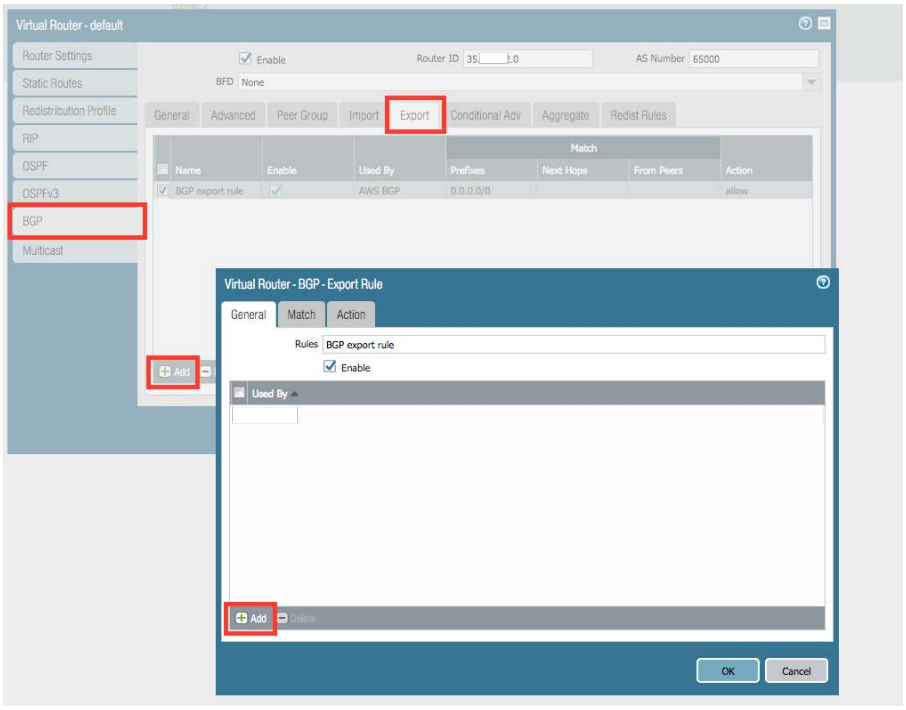
set connection-options hold-time 30



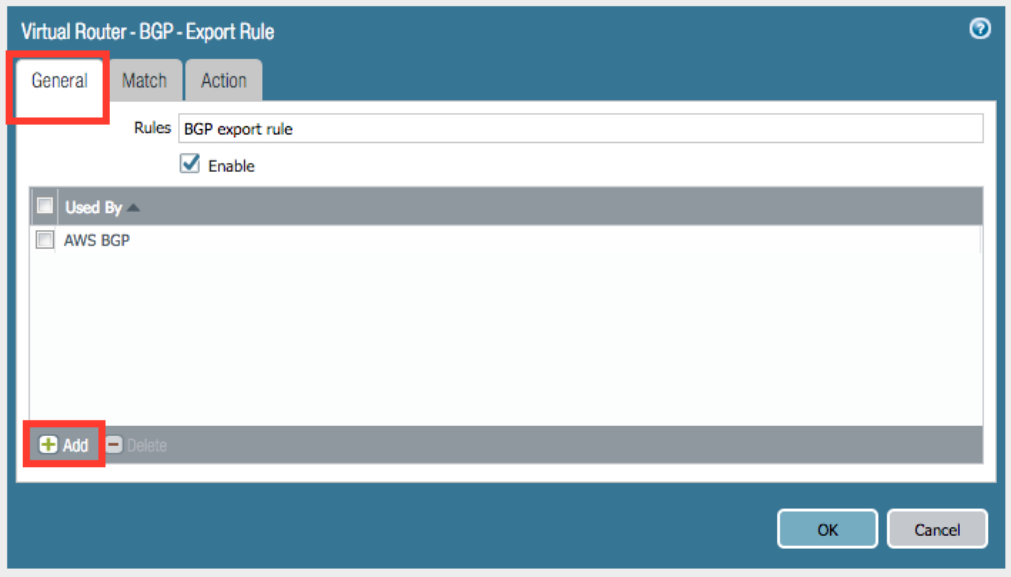
Once you have created the (6) peer connections(?) in the peer group your window should like the following example:



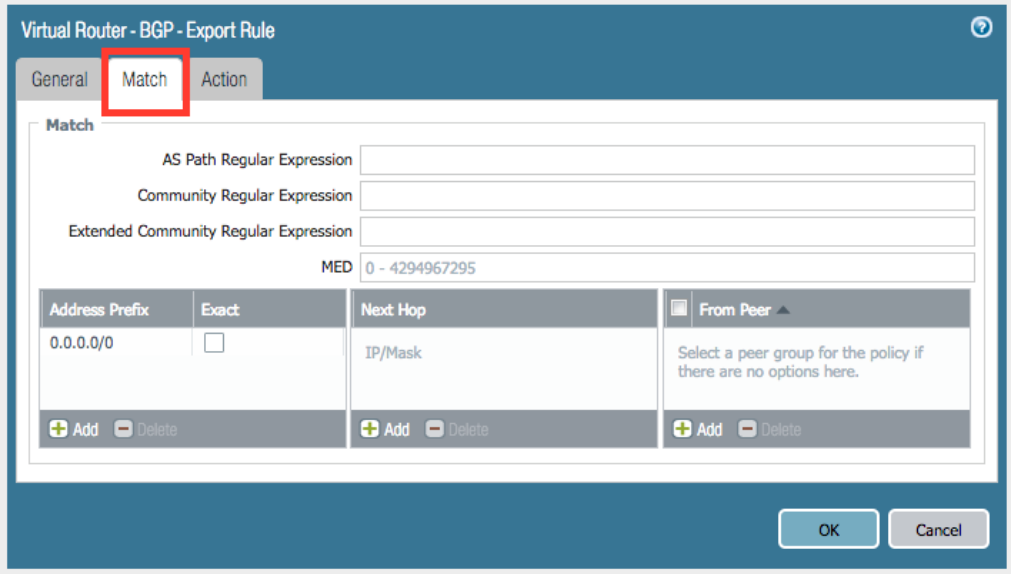
Next, we are going to create a BGP Export rule. From the virtual router BGP menu, select the Export tab. From the bottom of the virtual router window select add to create a new export rule.



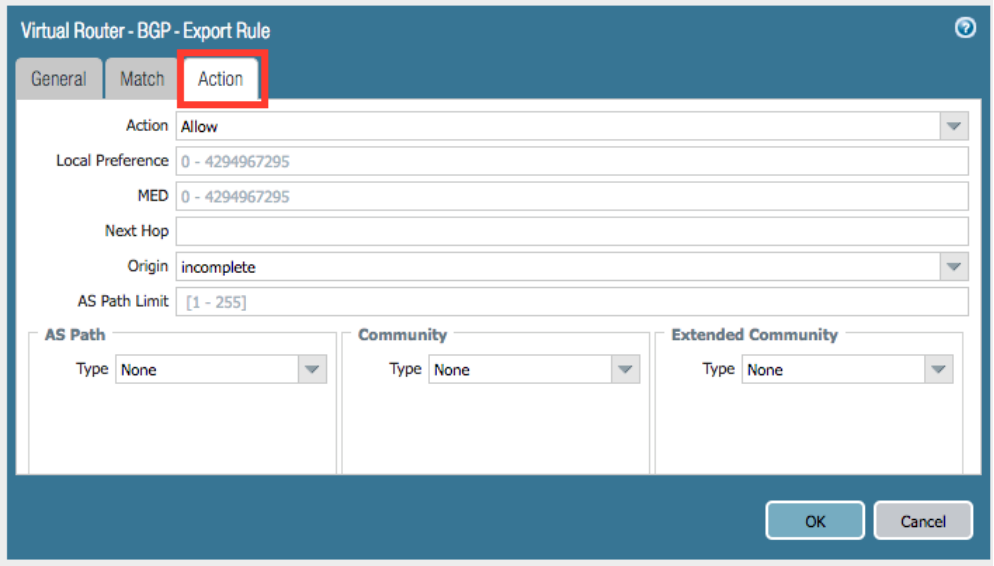
From the pop-up window, click on the add button to create a new peer group configured like the example below.



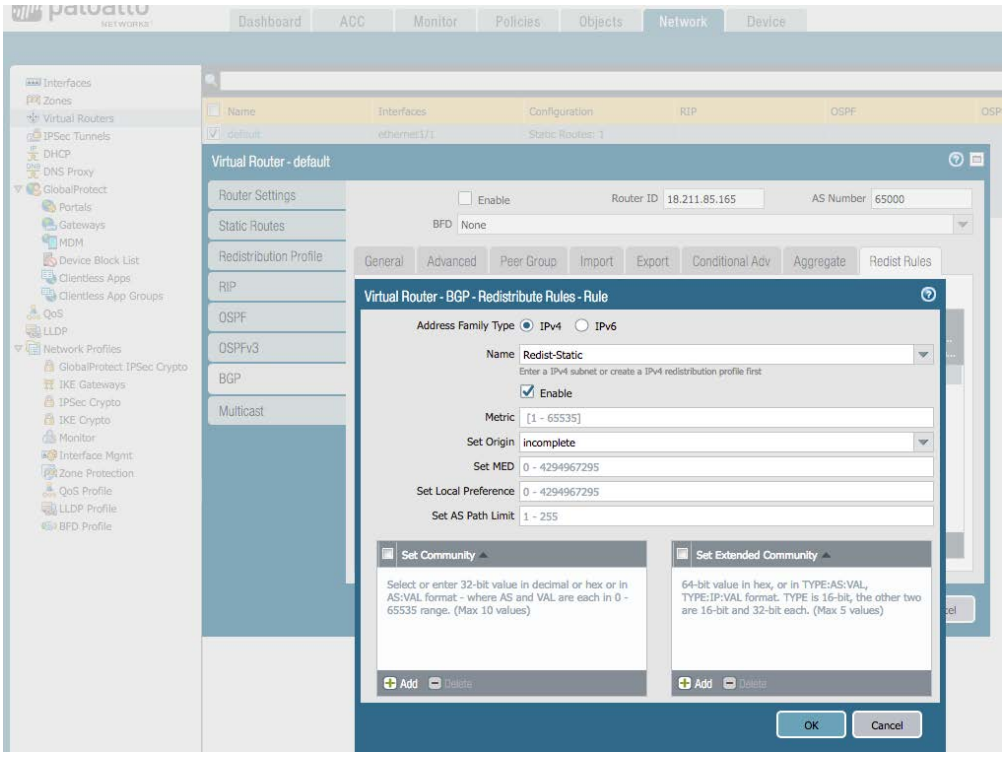
The match tab:



The Action Tab:



Next, we will configure the BGP Redistribution Rules. From the virtual router select the BGP tab. From the BGP window select the Redist Rules Tab. From the bottom of the Redist Rule window select the Add button. You will get a pop-up window to configure a rule. From the name drop down select the Route Redist def that was created above. Configure the rule as shown below.



From the main BGP window, select the ok button to finish your BGP configuration.

At this point you will want to commit the routing configuration changes to the firewall.

Next, we will validate the routing configuration. Access firewall 1 from an SSH session to access the cli of the firewall.

From the cli, issue the following commands:

show routing protocol bgp summary

```
admin@PA-VM> show routing protocol bgp summary

  ==========
  router id:                     35.165.12.0
  virtual router:                default
  reject default route:          yes
  redist default route:          allow
  Install BGP routes:            yes
  Graceful Restart:              supported
  AS size:                       2
  Local AS:                      65000
  Local member AS:               0
  Cluster id:                    0.0.0.0
  Default local preference:      100
  Always compare MED:            no
  Aggregate regardless MED:      yes
  Deterministic MED processing:  yes
  Accept ORF:                    no
  Accept CISCO style prefix:     yes
  mp-bgp-enable:                 yes
  afi-safi-ipv4-unicast:         yes
  rib-out entries:               current 18, peak 18
    peer VGW1-2 Peer:            AS 7224, Established, IP 169.254.12.61
      bgpAfiIpv4/unicast pfx:    Accepted pfx: 1, Advertised pfx: 3
    peer VGW2-1 Peer:            AS 7224, Established, IP 169.254.12.157
      bgpAfiIpv4/unicast pfx:    Accepted pfx: 1, Advertised pfx: 3
    peer VGW3-1:                 AS 7224, Established, IP 169.254.13.201
      bgpAfiIpv4/unicast pfx:    Accepted pfx: 1, Advertised pfx: 3
    peer VGW1-1 Peer:            AS 7224, Established, IP 169.254.14.89
      bgpAfiIpv4/unicast pfx:    Accepted pfx: 1, Advertised pfx: 3
    peer VGW2-2 Peer:            AS 7224, Established, IP 169.254.14.181
      bgpAfiIpv4/unicast pfx:    Accepted pfx: 1, Advertised pfx: 3
    peer VGW3-2:                 AS 7224, Established, IP 169.254.15.61
      bgpAfiIpv4/unicast pfx:    Accepted pfx: 1, Advertised pfx: 3

admin@PA-VM>
```

Next, we will check to make sure that the default routes are being advertised:

show routing protocol bgp rib-out

```
admin@PA-VM> show routing protocol bgp rib-out


VIRTUAL ROUTER: default (id 1)
  ==========
  Prefix              Nexthop          Peer       Originator      Adv Status  Aggr Status     AS-Path
 0.0.0.0/0            169.254.14.90    VGW1-1 Peer 0.0.0.0          advertised  no aggregation  65000
 10.1.1.1/32          169.254.14.90    VGW1-1 Peer 0.0.0.0          advertised  no aggregation  65000
 10.10.1.0/24         169.254.14.90    VGW1-1 Peer 0.0.0.0          advertised  no aggregation  65000
 0.0.0.0/0            169.254.12.158   VGW2-1 Peer 0.0.0.0          advertised  no aggregation  65000
 10.1.1.1/32          169.254.12.158   VGW2-1 Peer 0.0.0.0          advertised  no aggregation  65000
 10.10.1.0/24         169.254.12.158   VGW2-1 Peer 0.0.0.0          advertised  no aggregation  65000
 0.0.0.0/0            169.254.13.202   VGW3-1      0.0.0.0         advertised  no aggregation  65000
 10.1.1.1/32          169.254.13.202   VGW3-1      0.0.0.0         advertised  no aggregation  65000
 10.10.1.0/24         169.254.13.202   VGW3-1      0.0.0.0         advertised  no aggregation  65000
 0.0.0.0/0            169.254.12.62    VGW1-2 Peer 0.0.0.0          advertised  no aggregation  65000
 10.1.1.1/32          169.254.12.62    VGW1-2 Peer 0.0.0.0          advertised  no aggregation  65000
 10.10.1.0/24         169.254.12.62    VGW1-2 Peer 0.0.0.0          advertised  no aggregation  65000
 0.0.0.0/0            169.254.14.182   VGW2-2 Peer 0.0.0.0          advertised  no aggregation  65000
 10.1.1.1/32          169.254.14.182   VGW2-2 Peer 0.0.0.0          advertised  no aggregation  65000
 10.10.1.0/24         169.254.14.182   VGW2-2 Peer 0.0.0.0          advertised  no aggregation  65000
 0.0.0.0/0            169.254.15.62    VGW3-2      0.0.0.0         advertised  no aggregation  65000
 10.1.1.1/32          169.254.15.62    VGW3-2      0.0.0.0         advertised  no aggregation  65000
 10.10.1.0/24         169.254.15.62    VGW3-2      0.0.0.0         advertised  no aggregation  65000

total routes shown: 18

admin@PA-VM>
```

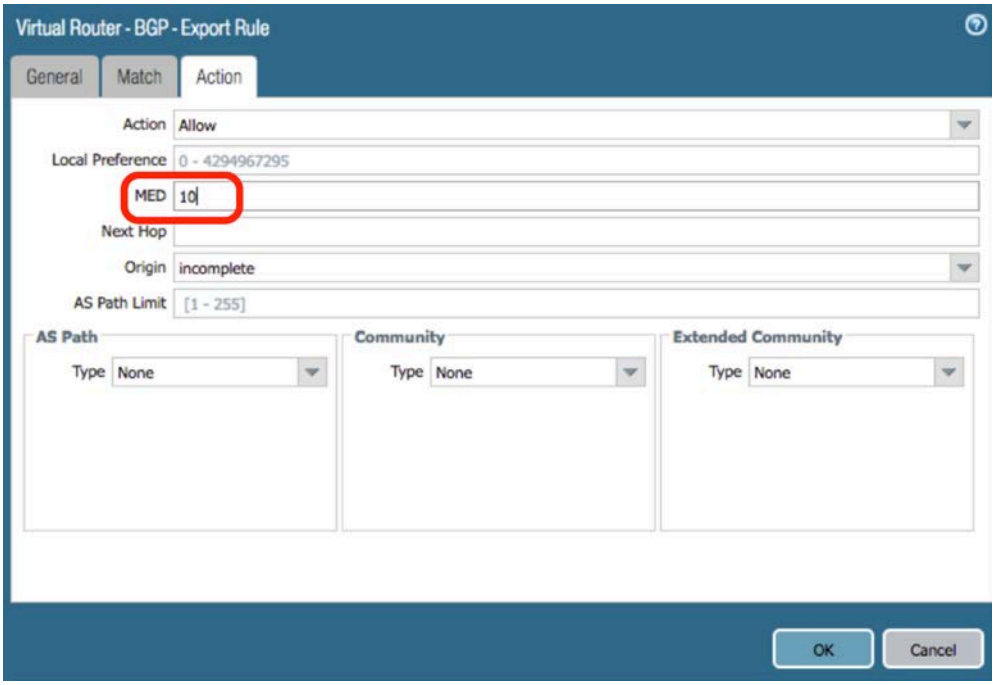
Finally, we will validate learning routes from each VPC in AWS.

show routing protocol bgp loc-rib

```
admin@PA-VM> show routing protocol bgp loc-rib

VIRTUAL ROUTER: default (id 1)
  ==========
  Prefix              Nexthop          Peer       Weight   LocPrf Org      MED flap AS-Path
  10.20.0.0/16        169.254.14.89    VGW1-1 Peer      0      100 igp      200    0 7224
  10.30.0.0/16        169.254.12.157   VGW2-1 Peer      0      100 igp      200    0 7224
  10.40.0.0/16        169.254.13.201   VGW3-1           0      100 igp      200    0 7224
 *10.20.0.0/16        169.254.12.61    VGW1-2 Peer      0      100 igp      100    0 7224
 *10.30.0.0/16        169.254.14.181   VGW2-2 Peer      0      100 igp      100    0 7224
 *10.40.0.0/16        169.254.15.61    VGW3-2           0      100 igp      100    0 7224
 *0.0.0.0/0           10.10.1.1        Local            0      100 i/c        0    0
 *10.1.1.1/32                          Local            0      100 i/c        0    0
 *10.10.1.0/24                         Local            0      100 i/c        0    0

total routes shown: 9
```

You have now completed the BGP configuration for firewall 1. You will need to perform the same steps for the Routing configuration on firewall 2.

In Order to prevent routing asymmetry will configure Firewall 2 with a higher Multi-Exit Discriminator (MED).

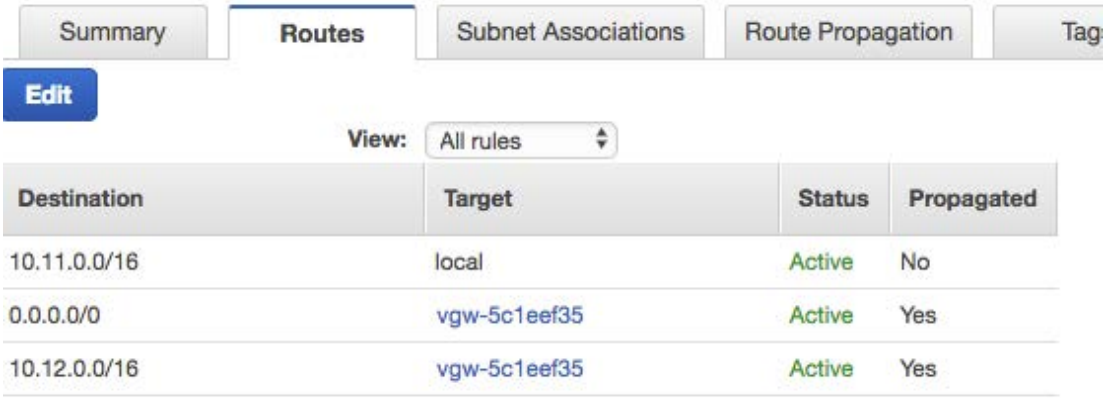


Once routing has been configured, the VPN connections will report the number of Routes learn.

utside IP Address	Inside IP CIDR	Status	Status Last Changed	Details
.233.183.193	169.254.45.4/30	UP	July 26, 2018 at 4:01:55 PM UTC-4	2 BGP ROUTES
.201.48.137	169.254.44.36/30	UP	July 26, 2018 at 4:03:04 PM UTC-4	2 BGP ROUTES

The last step is to configure route propagation or configure a static route pointing to the VGW in the AWS console. In the VPC console go to Route Tables and select the Route Table tied to your spoke Subnets. In the route propagation table, select Edit and Check the box for the attached VGW and hit Save.

Move to the Routes tab and you will now see the propagated routes targeting the VGW.



At this point you should have built a Transit VPC on AWS with the VM-Series. You can setup an ec2 instance in a subscribing VPC to test outbound functionality. This provides the connectivity between the subscribing VPC and the Transit VPC.