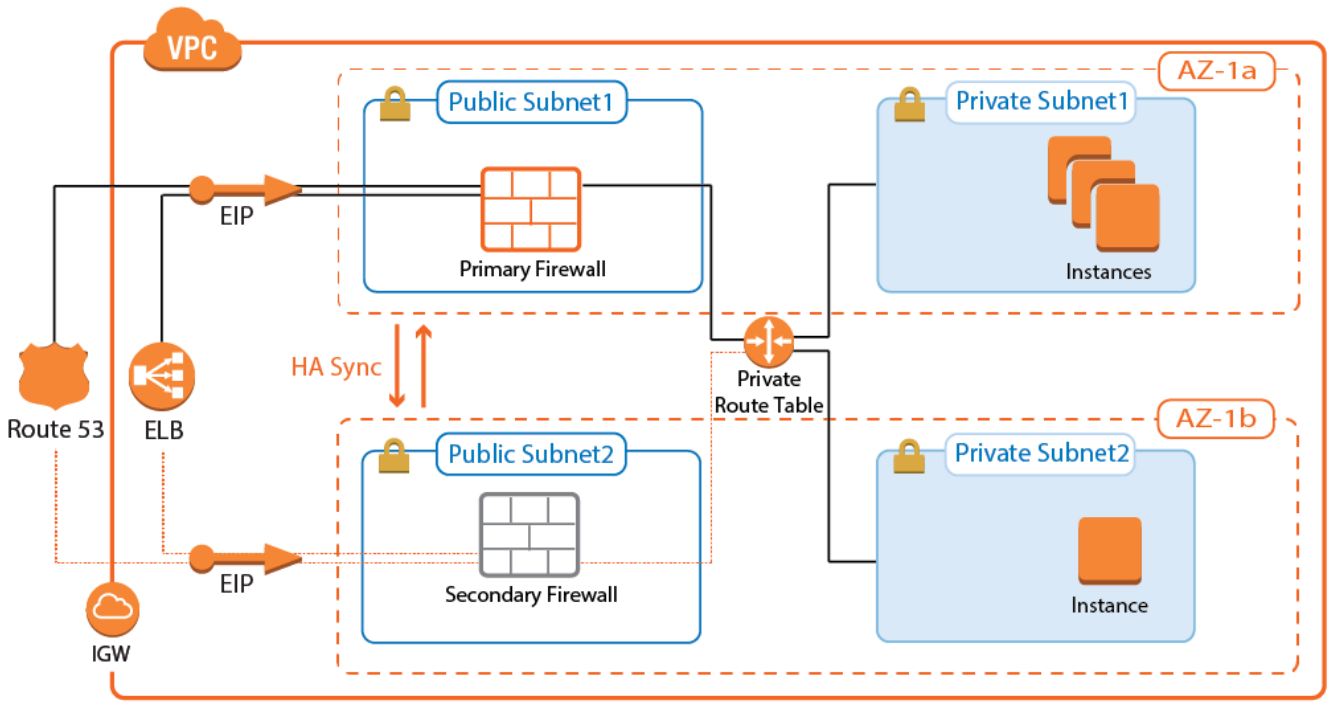
How to Configure a Multi-AZ High Availability Cluster in AWS Using the AWS Console

https://campus.barracuda.com/doc/73719420/

To ensure that at least one firewall is always active, deploy two firewalls into an active-passive high availability cluster. Each firewall is deployed into a different Availability Zone. The active firewall is used as the default gateway in the route table associated with the private networks. When the virtual server fails over from the primary to the secondary firewall, the AWS route table is rewritten to use the now-active secondary firewall as the default gateway.

Depending on the network protocol of the incoming connections, use an Elastic Load Balancer or AWS Route 53. The Elastic Load Balancer only supports TCP-based traffic, whereas Route 53 can be used with both TCP and UDP traffic.

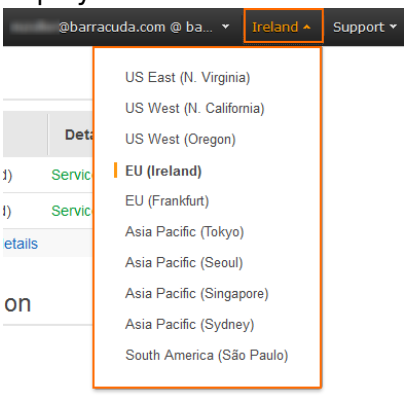


Before You Begin

- (BYOL only) Licenses matching the desired instance size are required when using BYOL images.
- Prepare an IAM role for the firewall instance. For more information, see How to Create an IAM Role for a CloudGen Firewall in AWS.

Step 1. Select the AWS Datacenter

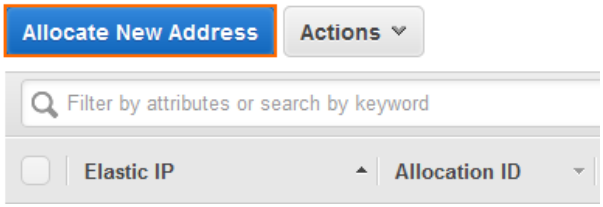
1. Log into the AWS console.
2. In the upper right, click on the datacenter location, and select the datacenter you want to deploy to from the list.



The selected datacenter location is now displayed in the AWS console.

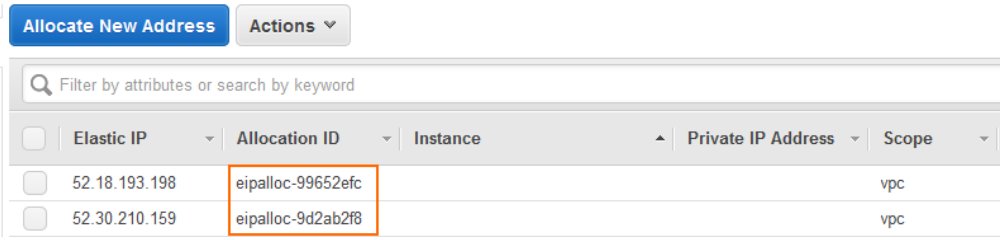
Step 2. Create an Elastic IP for Each Firewall

1. Log into the AWS console.
2. Click **Services** and select **EC2**.
3. In the **Network & Security** section of the left menu, click on **Elastic IPs**.
4. For the primary and secondary firewall:
    1. Click **Allocate New Address**.



    2. Click **Yes, Allocate**.

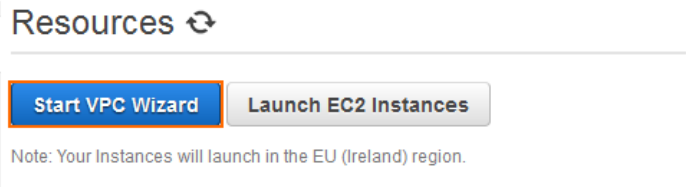
Two unassigned elastic IPs are now added to the list. Copy the **Allocation ID** for future use.



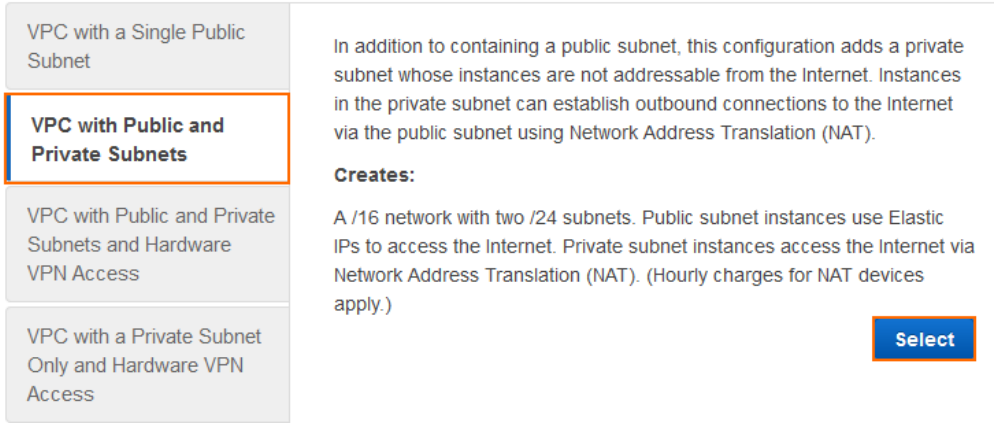
Step 3. Create a VPC with the VPC Wizard

Use the VPC wizard to create a VPC with two subnets. Each subnet must be created in a different availability zone. Additional subnets for the backend instances are added after the wizard.

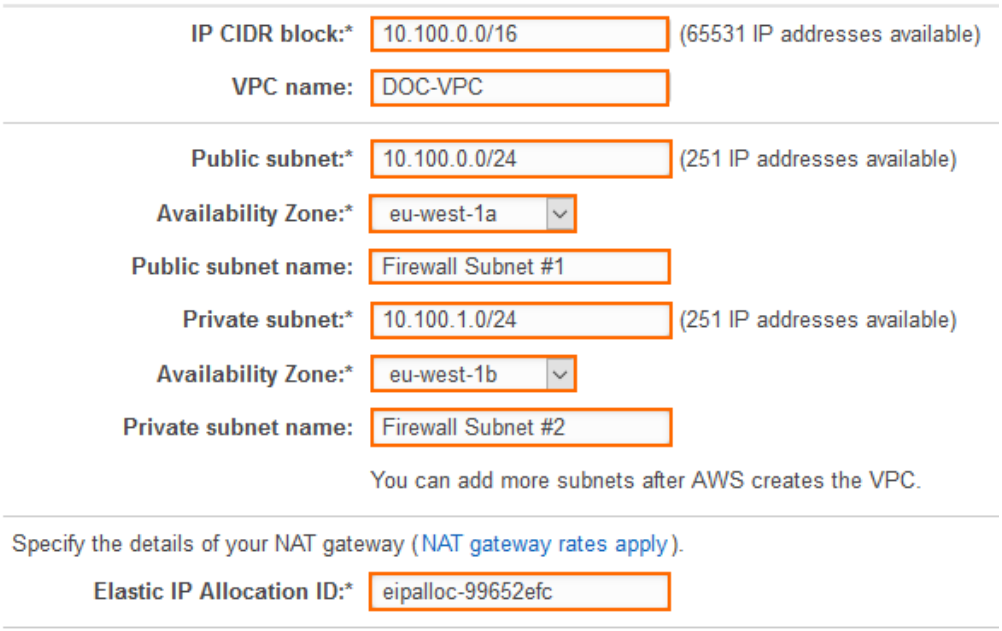
1. Log into the AWS console.
2. Click **Services** and select **VPC**.
3. Click **Start VPC Wizard**. The VPC wizard opens.



4. Select **VPC with Public and Private Subnets** and click **Select**.

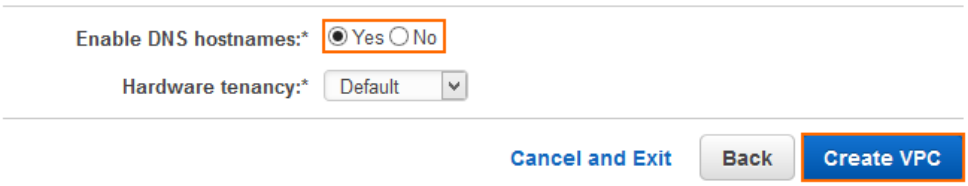


5. Configure the following settings:
   - **IP CIDR block** – Enter a /16 CIDR block that does not overlap with any of your other networks.
   - **VPC Name** – Enter the name.
   - **Public subnet** – Enter the /24 subnet used for the primary firewall.
   - **Public subnet name** – Enter a name for the primary firewall subnet.
   - **Availability Zone** – Select an availability zone.
   - **Private subnet** – Enter the /24 subnet used for the secondary firewall.
   - **Private subnet name** – Enter a name for the secondary firewall subnet.
   - **Availability Zone** – Select a different subnet for the second subnet because the primary and secondary firewalls must be in different Availability Zones. E.g, Select **eu-west-1b** if the you selected **eu-west-1a** as the public subnet Availability Zone.
   - **Elastic IP Allocation ID** – Enter the Allocation ID for the elastic IP address created in step 1.

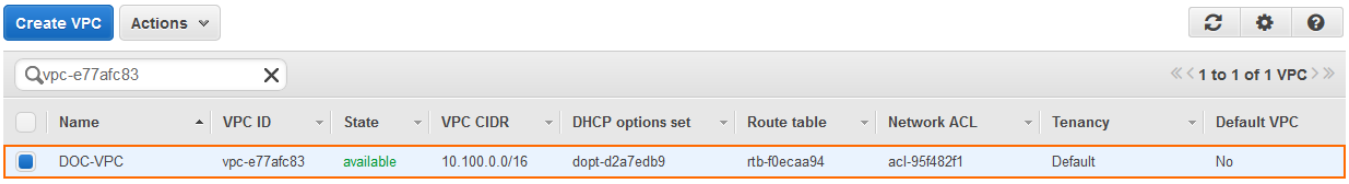


- **Enable DNS hostnames** (optional) – Set to **NO** to use only IP addresses to access your VPC.

6. Click **Create VPC**.



The VPC is now listed in the **Your VPCs** list.

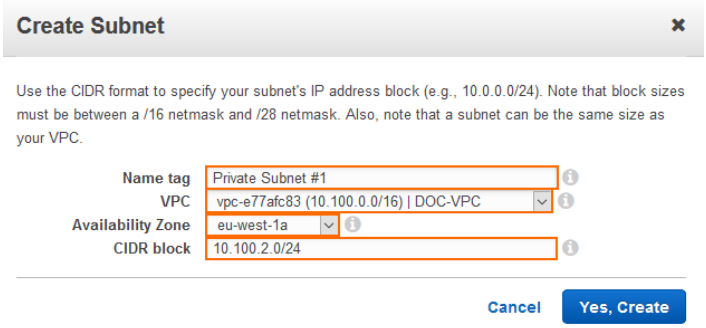


Step 4. Add a Subnet to the VPC

Add a private subnet for instances that use the firewall.

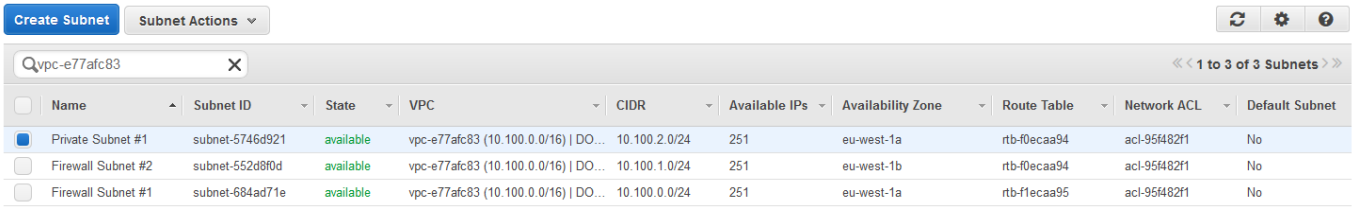
1. Log into the AWS console.
2. Click **Services** and select **VPC**.
3. Click **Subnets** in the left menu.
4. Click **Create Subnet**.
5. Create a subnet:

- **Name tag** – Enter a name for the subnet.
- **VPC** – Select the VPN created in step 3.
- **Availability Zone** – Select an availability zone from the list.
- **CIDR block** – Enter a free subnet in the scope of the network defined for the VPC.



6. Click **Yes, Create**.

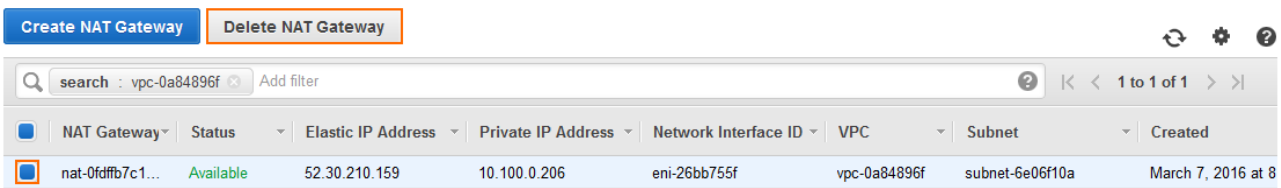
You now have three subnets in the VPC:



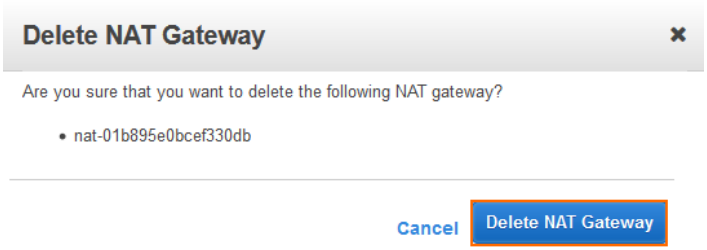
Step 5. Delete the NAT Gateway Instance

The VPC wizard automatically creates a NAT gateway instance. But since the firewall already includes this functionality, the NAT gateway instance must be deleted.

1. Log into the AWS console.
2. Click **Services** and select **VPC**.
3. In the **Virtual Private Cloud** section of the left menu, click on **NAT Gateways**.
4. (optional) Enter the VPC ID in the search bar.
5. Select the NAT gateway created for your VPC and click **Delete NAT Gateway**. The **Delete NAT Gateway** window opens.



6. Click **Delete NAT Gateway**.

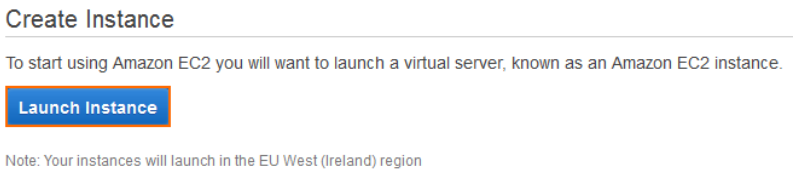


The elastic IP address associated with the NAT gateway is released automatically and is now free to use for one of the firewall instances.

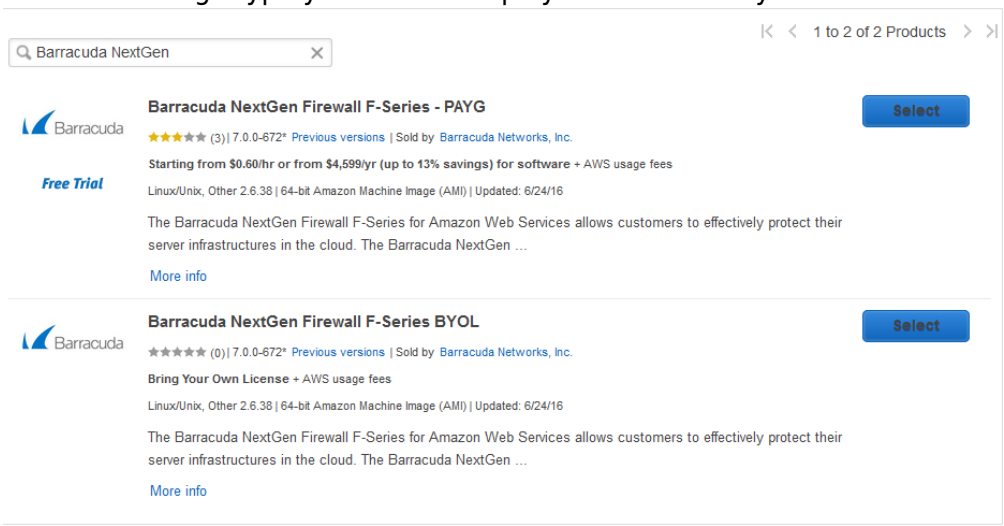
Step 6. Deploy the Primary Firewall

The primary firewall is deployed into the first firewall subnet of the VPC. Two image types are available in the AWS Marketplace: BYOL and hourly.

1. Log into the AWS console.
2. Click **Services** and select **EC2**.
3. Click **Launch Instance** in the **Create Instance** section. The **VPC wizard** starts.



4. In the left menu, click **AWS Marketplace**.
5. Enter `Barracuda NextGen` in the search box**.**
6. Select the image type you want to deploy: BYOL or hourly.

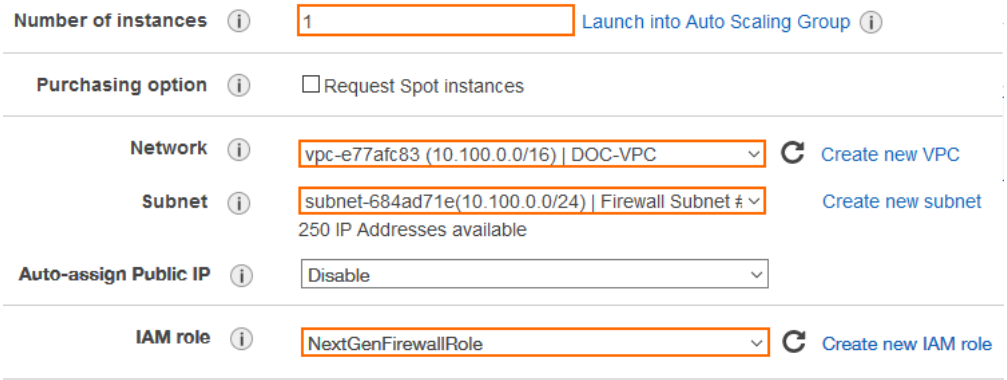


7. Select the **Instance Type**. If you are deploying a BYOL image, verify that the number of CPU cores of the instance matches your license.

Step 2: Choose an Instance Type

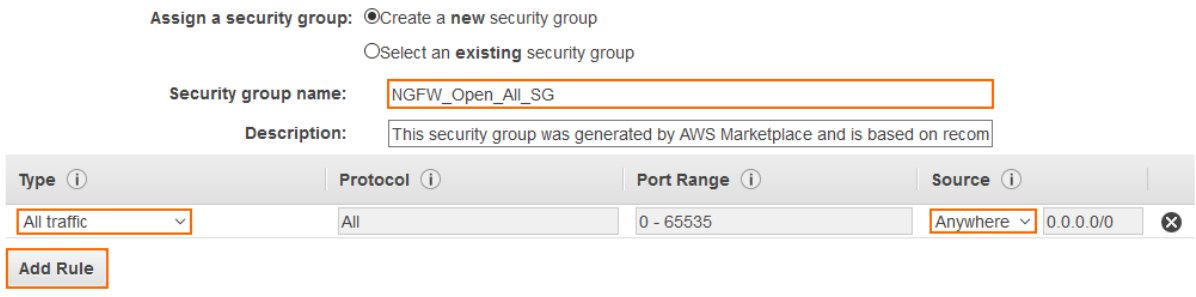
☐	General purpose	m3.2xlarge	8	30	2 x 80 (SSD)	Yes	High
☑	Compute optimized	c4.large	2	3.75	EBS only	Yes	Moderate
☐	Compute optimized	c4.xlarge	4	7.5	EBS only	Yes	High
☐	Compute optimized	c4.2xlarge	8	15	EBS only	Yes	High

8. Click **Next: Configure Instance Details**.
9. Configure the **Instance Details**:
   - **Number of instances** – Enter 1
   - **Network** – Select the VPC created in step 3.
   - **Subnet** – Select the subnet for the primary firewall.
   - **IAM role** – Select the IAM role created for the firewall instance. Verify that all required IAM policies for the route-shifting High Availability cluster are attached.



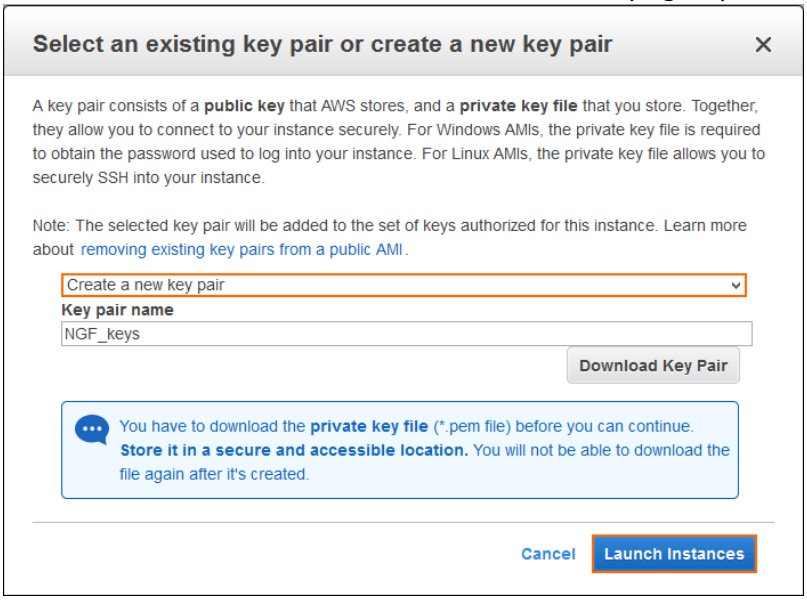
10. In **Network Interfaces**, enter the **Primary IP** address. The IP address must be in the subnet selected above.

11. Click **Next: Add Storage**.
12. Click **Next: Tag Instance**.
13. Click **Next: Configure Security Group**.
14. (optional) Enter a **Security group name**.
15. (optional) Remove the preconfigured rules in the security group.
16. Click **Add Rule** and open up the security group for all traffic.
    - **Type** – Select **All Traffic**.
    - **Source** – Select **Anywhere**.



17. Click **Review and Launch**.

18. Click **Launch**. The **Select and existing key pair or create a new key pair** pop-over window opens.
19. From the drop-down list, select your desired option. The certificate is valid only for root SSH logins. For Barracuda Firewall Admin, the Instance ID is the default password.
20. Click the checkbox to verify that you have access to the selected key, or, to download a new key pair, click **Download Key Pair**.
21. Click **Launch Instances**. The **Launch Status** page opens.

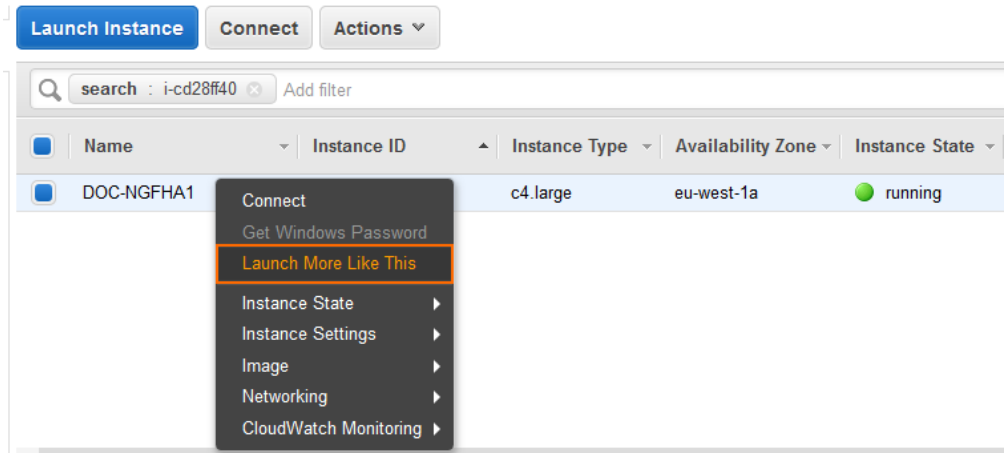


Locate and copy the **Instance IDs**. This is the default password used to log into the primary firewall via Barracuda Firewall Admin.

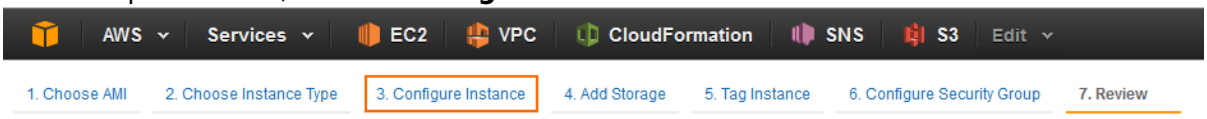
Step 7. Deploy the Secondary Firewall

The secondary firewall instance is deployed into the secondary firewall subnet of the VPC. The configuration of the primary firewall is used as a starting point.

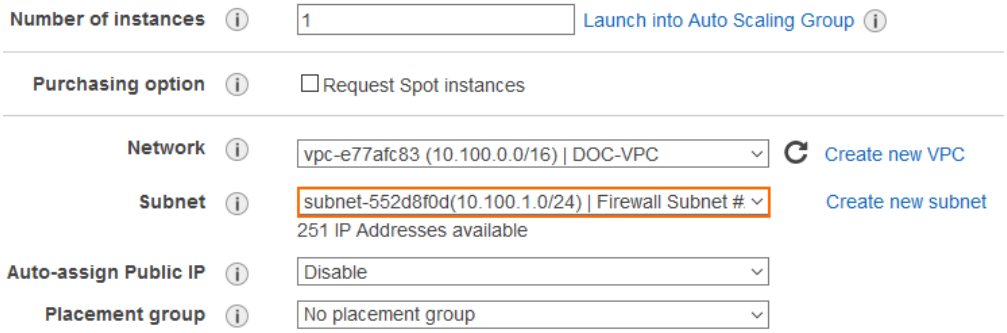
1. Log into the AWS console.
2. Click **Services** and select **EC2**.
3. Right-click on the primary firewall instance created in step 6 and click **Launch More Like This**.



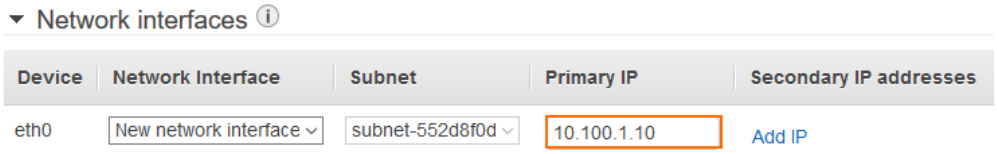
4. On the top menu bar, click **3. Configure Instance**.



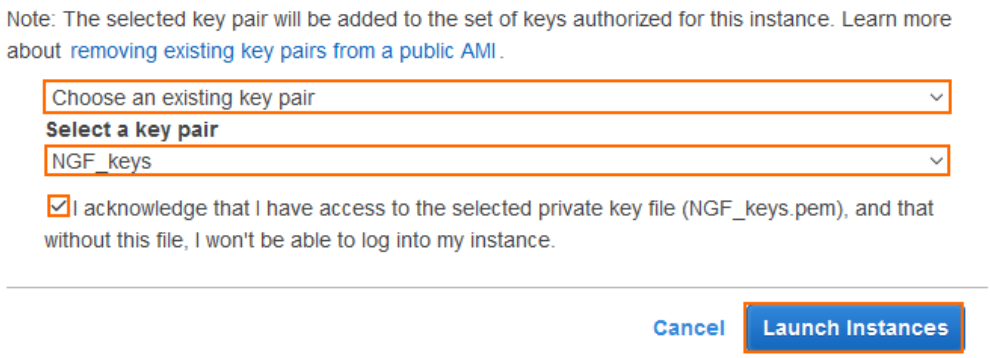
5. Change the subnet in the **Instance Details** section:
   - **Subnet** – Select the subnet for the secondary firewall.



6. Enter the **Primary IP** address in the **Network Interfaces** section. The IP address must be in the subnet selected above.



7. Click **Review and Launch**.
8. Click **Launch**. The **Select an existing key pair or create a new key pair** window opens.
9. Select **Choose an existing key pair** from the drop-down list.
10. Select the key pair used for the first firewall.
11. Click **Launch Instances**. The **Launch Status** page opens.

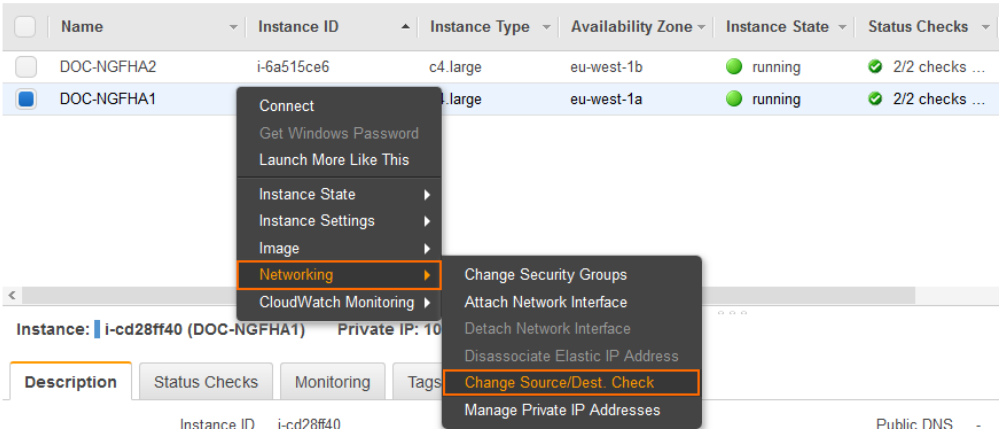


Locate and copy the **Instance IDs**. This is the default password used to log into the secondary firewall via Barracuda Firewall Admin.

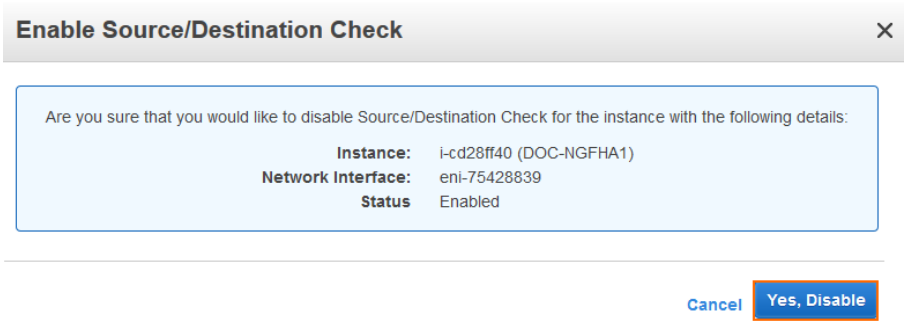
Step 8. Disable the Source/Destination Check for Both Firewalls

To allow the firewall to perform NAT operations, you must disable the source/destination check for the firewall network interfaces.

1. Log into the AWS console.
2. Click **Services** and select **EC2**.
3. Right-click on the primary firewall created in step 6, click **Networking**, and select **Change Source/Dest. Check**.



4. Click **Yes, Disable**.

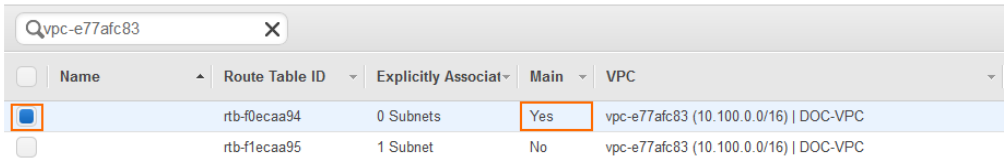


5. Right-click on the secondary firewall created in step 7, click **Networking**, and select **Change Source/Dest. Check**.
6. Click **Yes, Disable**.

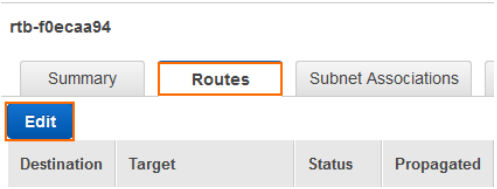
Step 9. Configure an AWS Route Table for Private Subnets

Configure the default route of the main routing table to use the primary firewall instance as the default gateway. Since this is the main route table, it is automatically applied to any subnets not specifically assigned to another route table.

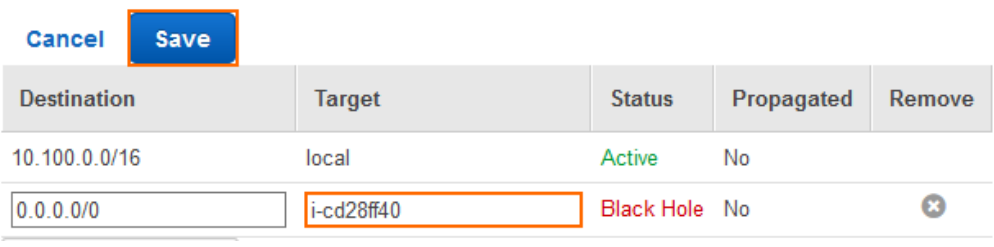
1. Log into the AWS console.
2. Click **Services** and select **VPC**.
3. In the left menu, click **Route Tables**.
4. Click on the main route table for your VPC.



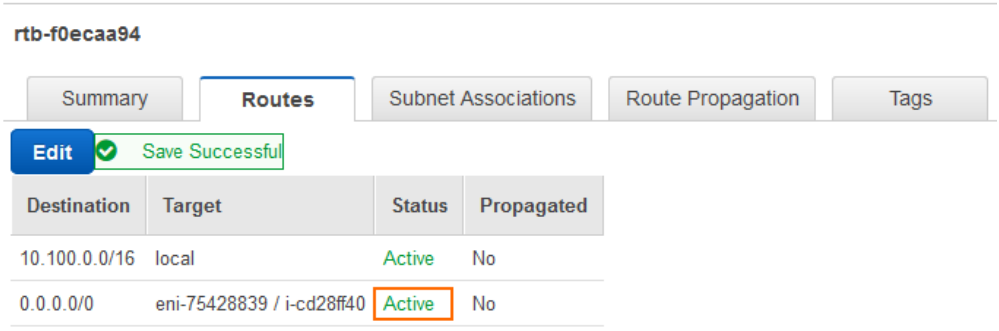
5. On the bottom, click on the **Routes** tab.
6. Click **Edit**.



7. In the **Target** column of the default route (0.0.0.0/0), enter the instance ID of the primary firewall.
8. Click **Save**.



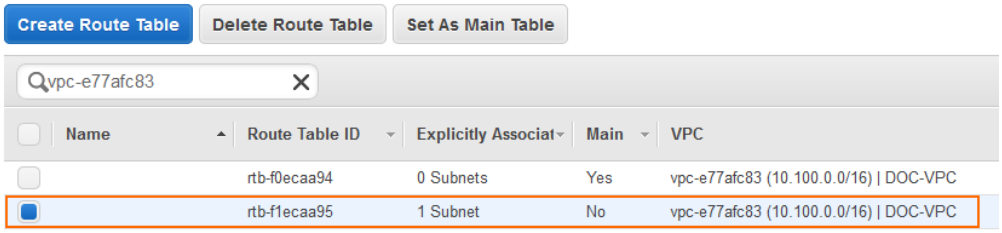
The default route now shows an **Active** state in the **Status** column:



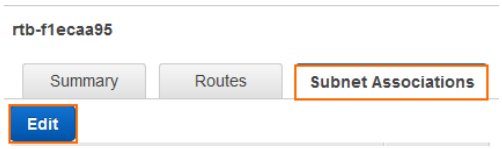
Step 10. Configure an AWS Route Table for the Firewall Subnets

The route table for the firewall subnet routes incoming and outgoing connections through the Internet gateway created by the VPC wizard in step 3.

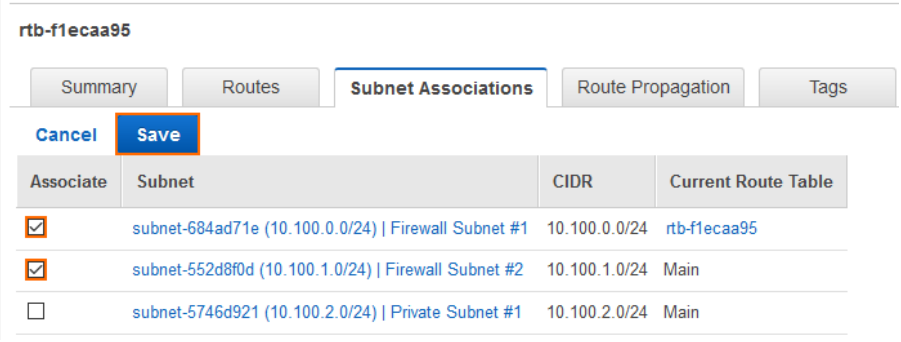
1. Log into the AWS console.
2. Click **Services** and select **VPC**.
3. In the left menu, click **Route Tables**.
4. Click on the second route table, which is currently associated with the subnet for the primary firewall.



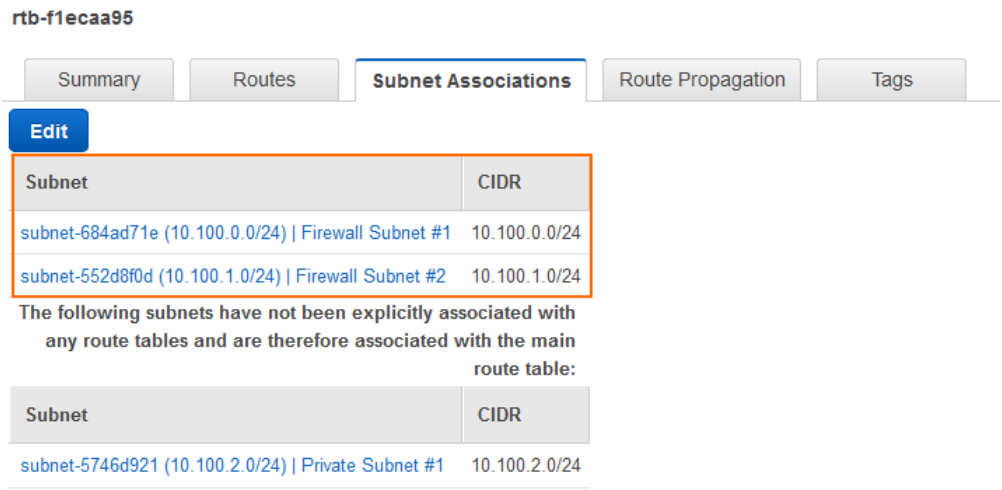
5. On the bottom, click on the **Subnet Associations** tab.
6. Click **Edit**.



7. Select both firewall subnets.
8. Click **Save**.



The firewall subnets are now associated with the AWS route table routing connections over the Internet gateway.

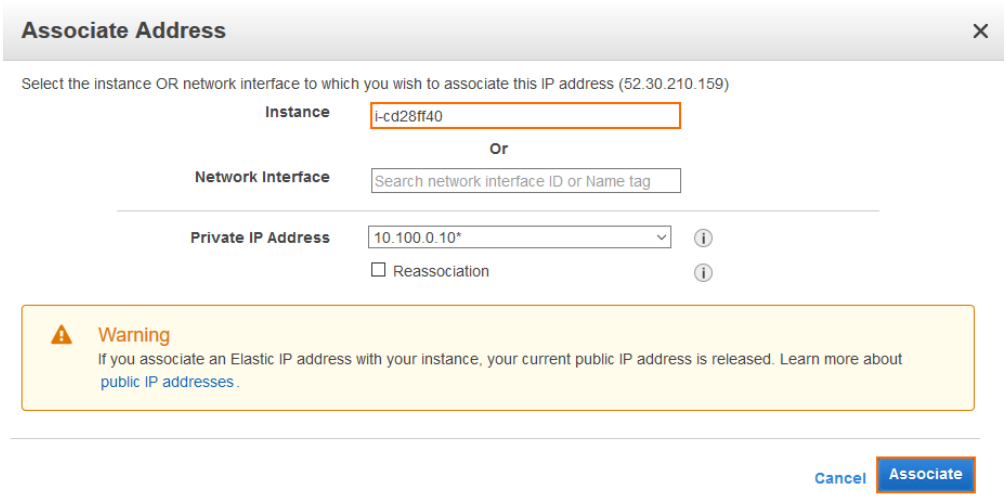


Step 11. Associate the Elastic IPs

Associate the elastic IPs created in step 2 with the firewall network interfaces.

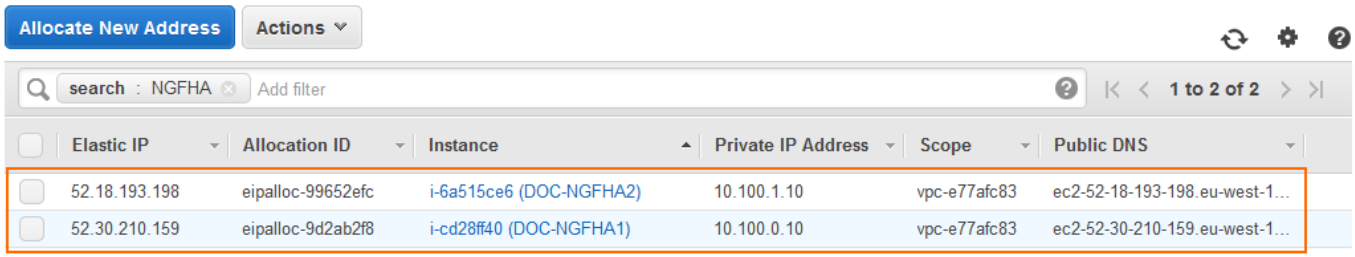
1. Log into the AWS console.
2. Click **Services** and select **EC2**.
3. In the **Network & Security** section of the left menu, click on **Elastic IPs**.
4. Right-click the first Elastic IP created in step 2 and click **Associate Address**.

5. Enter the **Instance** ID of the primary firewall and click **Associate**.



6. Right-click the second Elastic IP created in step 2 and click **Associate Address**.
7. Enter the **Instance** ID of the secondary firewall and click **Associate**.

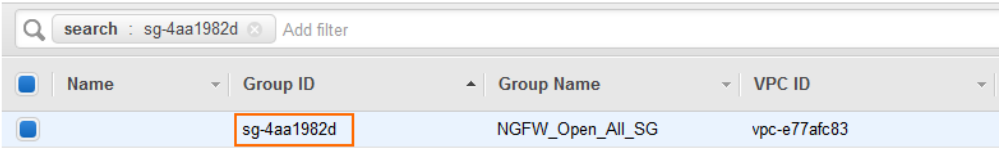
Traffic to the two Elastic IPs is now automatically forwarded to the network interface of the primary and secondary firewalls.



Step 12. Security Groups

Create a security group for the private networks that allow all traffic from the security group assigned to the firewall.

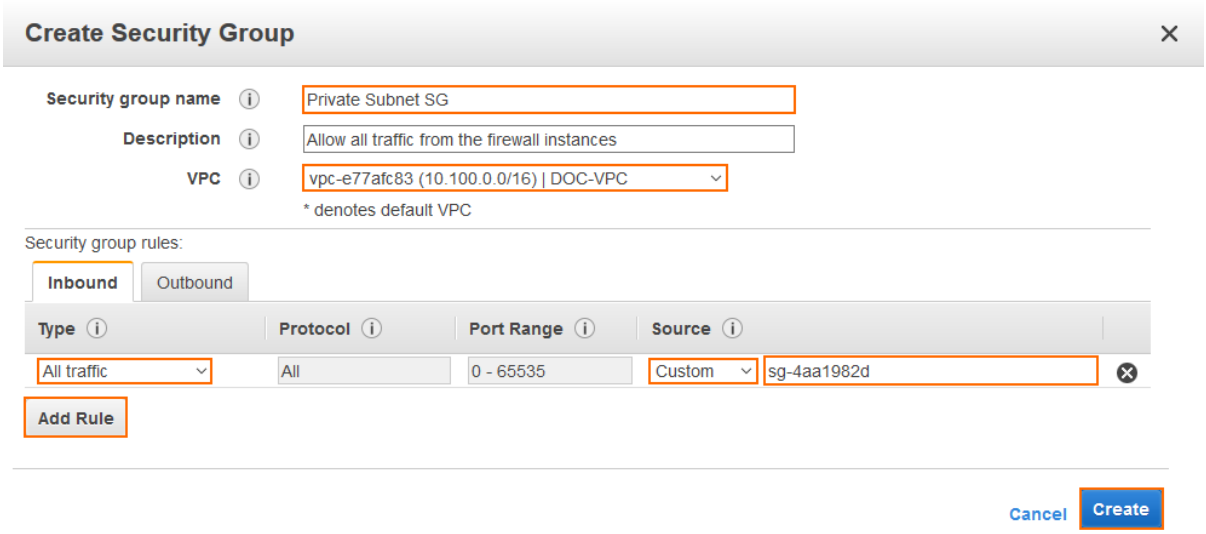
1. Log into the AWS console.
2. Click **Services** and select **VPC**
3. In the **Security** section of the left menu, click on **Security Groups**.
4. Use the VPC ID to filter the security groups, and copy the **Group ID** of the security group assigned to the firewall instances.



5. Click **Create Security Group**.
   - **Group name** – Enter a name for the security group.

- **Description** – Enter a description for the security group.
- **VPC** – Select the VPC you created in Step 3.

6. In the lower half of the page, click on the **Inbound** tab.
7. Create a rule to allow traffic from the firewall security group:
   - **Type** – Select **All Traffic**.
   - **Protocol** – Select **ALL**.
   - **Source** – Enter the group ID of the security group assigned to your firewalls.
8. Click **Add Rule**.



9. Click **Create**.

Assign this security group to all instances in one of the private networks that are routed through the firewall.

Step 13. (optional) Create Network ACLs

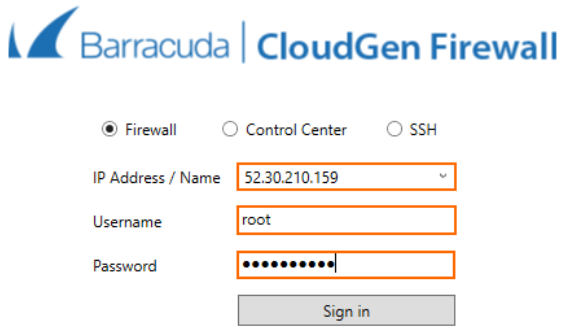
The Network ACLs created by the VPC wizard are configured by default to allow traffic through. If required, go **Network ACLs** to edit the network ACL assigned to your VPC.

Step 14. Change the Primary Firewall Network Configuration from Dynamic to Static

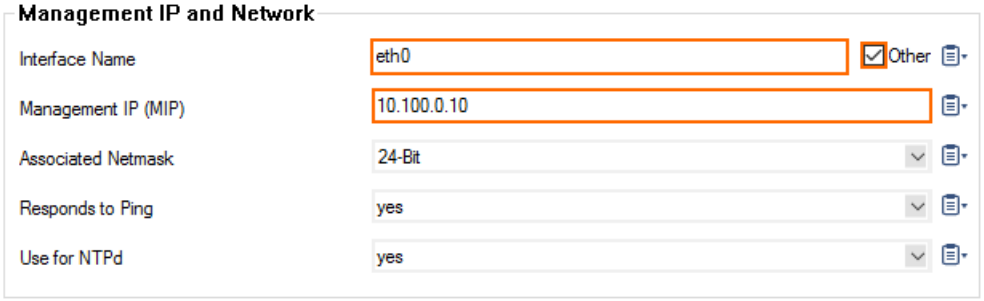
On the primary firewall instance, change the network configuration from the dhcp to a static network interface. Use the static private IP address you assigned during deployment. Always use the first IP address of the subnet as the default gateway.

1. Log into the primary firewall via Barracuda Firewall Admin:

- **IP Address /Name** – Enter the Elastic IP of the primary firewall.
- **Username** – Enter `root`.
- **Password** – Enter the instance ID of the primary firewall.



2. Go to **CONFIGURATION > Configuration Tree > Box > Network**.
3. In the left menu, click on **xDSL/DHCP/ISDN**.
4. Click **Lock**.
5. Delete the **DHCP01** entry in the **DHCP Links** list.
6. Set **DHCP Enabled** to **No**.
7. In the left menu, click on **IP Configuration**.
8. In the **Management IP and Network** section, reconfigure the management IP:
   - **Interface Name** – Select **Other** and enter `eth0`.
   - **Management IP** – Enter the private IP address of the primary firewall. Go to **CONTROL> Network**. The private IP address is assigned to the dhcp interface.
   - **(optional) Netmask** – Change the netmask to match the subnet of the primary firewall subnet.



9. In the left menu, click on **Routing**.
10. Click **+** in the **Routes** table and configure the following settings:
    - **Target Network Address** – Enter `0.0.0.0/0`
    - **Route Type** – Select **gateway**.
    - **Gateway** – Enter the first IP address of the primary firewall subnet. E.g., `10.100.0.1` if the IP address of the firewall is 10.100.0.10.
    - **Trust Level** – Select **Unclassified**.
11. Click **OK**.
12. Click **Send Changes** and **Activate**.
13. Activate the changes to the network configuration:
    1. Go to **CONTROL > Box**.
    2. In the **Network** section of the left menu, click on **Activate new network**

configuration.
3. Click **Activate Now**.

Open the **CONTROL > Network** page. Your interface and IP address are now static.

Step 15. (PAYG only) Import the PAYG License from the Secondary Firewall

Step 15.1 Export the PAYG License from the Secondary Firewall

1. Log into the secondary firewall.
2. Go to **CONFIGURATION > Configuration Tree > Box > Licenses**.
3. Click **Lock**.
4. Select the license file, click the export icon, and select **Export to File**.
5. Click **Unlock**.

Step 15.2 Import the PAYG License on the Primary Firewall

1. Log into the primary firewall.
2. Go to **CONFIGURATION > Configuration Tree > Box > Licenses**.
3. Click **Lock**.
4. Click **+** and select **Import from File**.
5. Select the license file exported from the secondary firewall.

The primary firewall now has both PAYG licenses listed in the **Licenses** list.

Step 16. Create a Stand-Alone HA Cluster

Create a stand-alone high availability cluster between the primary and secondary firewall. The management IP address of the secondary firewall (HA network) must be configured as a static IP address using the private IP address of the secondary firewall. Also, the gateway IP address for the default route of the secondary firewall must be changed to match the subnet the second firewall is running in.

For more information, see How to Set Up a High Availability Cluster.

Step 17. Configure Services to Listen on the Loopback Interface

Because AWS does not support floating IP addresses, you must configure all services on the virtual server to listen on a loopback address (127.0.0.X). Use **Application Redirect** access rules to redirect

incoming traffic from the eth0 interface to the services. Use the private IP addresses of both firewalls as the destination of the rule to ensure that it matches without regard to which firewall VM the virtual server is currently running on.

Step 18. (BYOL only) Activate and License the HA Cluster

Activate the secondary firewall first, then the primary firewall. This ensures that the primary firewall can download the licenses of the secondary firewall.

For more information, see How to Activate and License a Standalone High Availability Cluster.

Step 19. (optional) Configure the Amazon Load Balancer or Amazon Route 53

Depending on the type of traffic, you can use either the AWS Elastic Load Balancer for TCP traffic, or Route 53 for UDP traffic.

Amazon Classic Elastic Load Balancer

The Elastic Load Balancer receives public TCP traffic and forwards it to the active firewall. Protocols other than TCP are not supported. For each TCP port you want to load balance, you must add a Load Balancer rule that maps the external port and protocol to the internal protocol and port. Configure the health checks to check a service on the virtual server, such as TCP 691 for the VPN service. In this way, only the firewall running the virtual server is regarded as healthy by the Load Balancer, and traffic is forwarded only to the active firewall.

For more information, see How to Configure an AWS Elastic Load Balancer for CloudGen Firewalls in AWS

DNS Load Balancing Using Route 53

For services not using TCP connections, Amazon Route 53 can be used to configure a DNS-based Load Balancer. Route 53 is also the preferred load balancing service for geographically distributed cloud resources.

For more information, see How to Configure Route 53 for CloudGen Firewalls in AWS.

Figures

1. multi_AZ_routeshifting_ha_0.png
2. aws_deploy_00.png
3. aws_deploy_01.png
4. awsha_eip_01.png
5. aws_deploy_03.png
6. aws_deploy_04.png
7. aws_deploy_05.png
8. aws_deploy_06.png
9. awsha_vpc_01.png
10. aws_ha_add_subnet01.png
11. aws_ha_add_subnet02.png
12. aws_deploy_08.png
13. aws_deploy_09.png
14. aws_deploy_10.png
15. awsha_primary_fw01.png
16. awsha_primary_fw02.png
17. awsha_primary_fw03.png
18. awsha_primary_fw04.png
19. awsha_primary_fw05.png
20. aws_deploy_15.png
21. awsha_primary_fw06.png
22. awsha_secondary_fw01.png
23. awsha_secondary_fw02.png
24. awsha_secondary_fw03.png
25. awsha_secondary_fw04.png
26. awsha_secondary_fw05.png
27. awsha_secondary_fw06.png
28. awsha_srcdst_01.png
29. awsha_srcdst_02.png
30. awsha_main_route_table_01.png
31. awsha_main_route_table_02.png
32. awsha_main_route_table_03.png
33. awsha_main_route_table_05.png
34. awsha_fw_route_table_01.png
35. awsha_fw_route_table_02.png
36. awsha_fw_route_table_03.png
37. awsha_fw_route_table_04.png
38. awsha_eip01.png
39. awsha_eip02.png
40. awsha_eip03.png
41. awsha_private_security_group01.png
42. awsha_private_security_group02.png
43. awsha_static_NIC_01.png
44. awsha_static_NIC_02.png