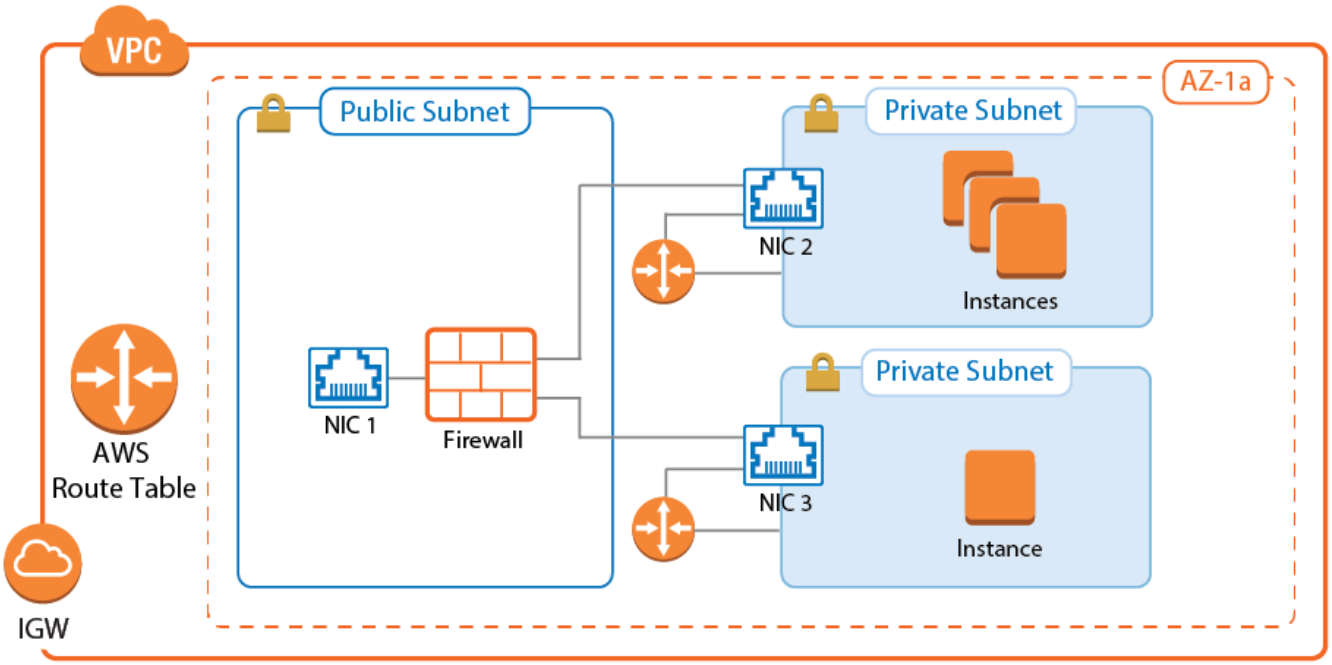
AWS Implementation Guide - Segmentation Firewall for Single AZ VPCs

A NextGen Firewall F with multiple network interfaces can be used as a segmentation firewall for your private subnets in the VPC. Traffic passing between the private subnets is routed through the firewall, where you can apply security policies and visualize traffic in real time between the subnets. To be able to route the traffic over the firewall, the standard route for internal VPC traffic must be circumvented. By default, all traffic within the VPC is routed over the default gateway. This route cannot be overridden by other more specific routes, nor can it be changed to use the firewall as the gateway instead. Using a combination of a firewall instance with multiple network interfaces and adding a route on the client instances allows you to use the F-Series Firewall as a segmentation firewall in AWS.

Use a segmentation firewall to enforce access policies and monitor traffic passing between the subnets. When compared with an AWS native solution, a NextGen Firewall is vastly superior regarding the depth at which both traffic can be inspected and security policies applied. In addition, NextGen Admin also provides real-time traffic visibility, while the Firewall Live and History pages allow quick, fine-grained access to all the traffic currently passing through the firewall.



Limitations

- All resources must be in a single Availability Zone.
- The number of private subnets is limited by the number of network interfaces supported by the instance type. So if the firewall supports three network interfaces, two private subnets can be connected. The primary network interface is used for external connectivity.
- A route must be added to the client instances in the private subnets. The default route over the gateway in the subnet bypasses the firewall. This can be stopped via Security Groups.
- Cannot be deployed as a high availability cluster.
- Connecting to subnets in other Availability Zones requires use of source NAT on the matching access rule.

Example CloudFormation template

To deploy the AWS infrastructure of this architecture quickly, use the CloudFormation template below. This

template only deployed the AWS infrastructure. The NextGen Firewall must be configured manually.

- One Barracuda NextGen Firewall (PAYG) m3.large instance.
- Two t2.micro Linux clients - one in each private subnet.

Download the Example Segmentation CloudFormation Template and the Network Diagram containing the IP addresses.

For step-by-step instructions on how to deploy a CloudFormation template, see How to Deploy an F-Series Firewall in AWS via CloudFormation Template.

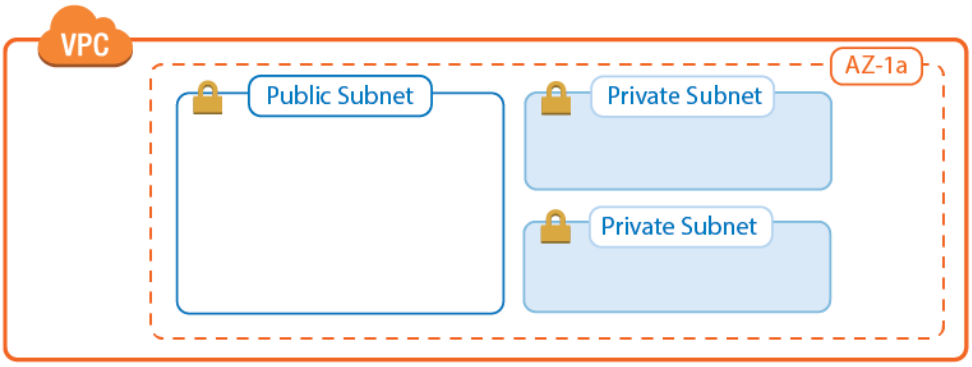
Deploying a segmentation firewall

Complete the following configuration steps to deploy the NextGen Firewall F as a segmentation firewall. For more detailed descriptions, follow the links for step-by-step instructions.

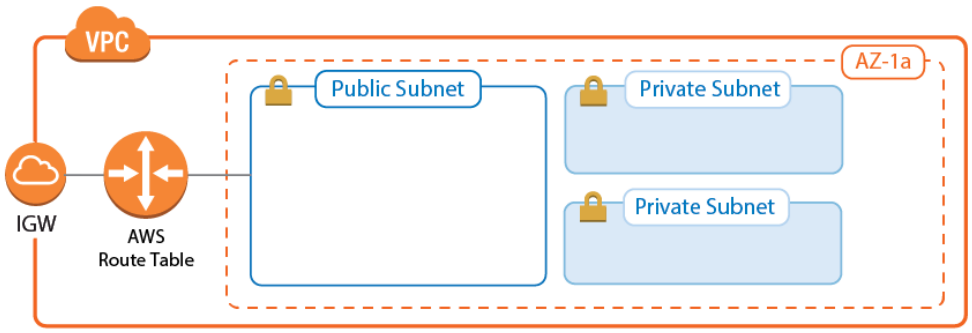
Create a VPC and deploy the firewall instance

Create a VPC in an AWS region of your choice. Since each subnet will contain a firewall network interface, the subnets must be created in the same Availability Zone. A minimum of three subnets are required:

- **Public subnet** – This is the subnet for the firewall instance and all other instances with public IP addresses.
- **Private subnets** – This is the subnet for all instances without external connectivity. All traffic is routed over the firewall. At least two private subnets are required for this architecture.



Add an AWS Internet gateway and modify the AWS routing table for the public subnet to use the Internet gateway as the target for the default route. Verify that the route table is associated with the public subnet. Instances deployed to the public subnet can now connect to the Internet via the AWS Internet gateway.

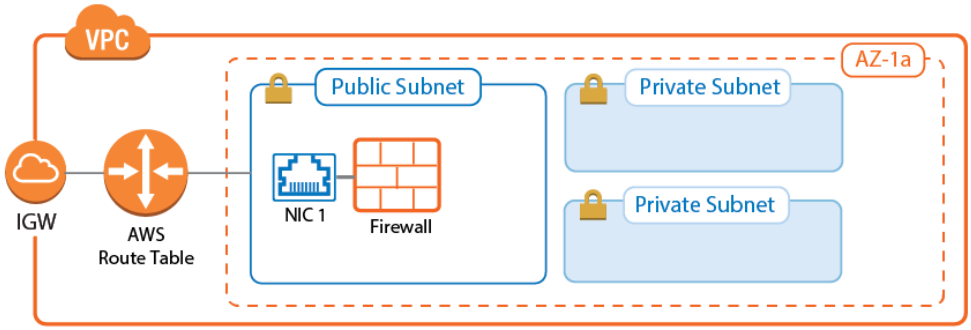


For the firewall, select the instance type according to the number of network interfaces. The number of network interfaces is the number of private subnets plus one for the public subnet. At least three network interfaces are required. The instance type must support at least three network interfaces: one for the public subnet and two for the private subnets. For more information, see AWS documentation.

Launch the firewall instance with the primary network interface in the public subnet. Since the firewall takes over the functionality of the security groups, the security group is configured to allow all traffic to and from the firewall instance. Disable the source/destination check. The default root password is the instance ID.

If you selected the BYOL image type, activate the license. For both PAYG and BYOL, install the available hotfixes as displayed in the **DASHBOARD > Firmware Update** element when logged in via NextGen Admin.

For step-by-step instructions, see How to Deploy a F-Series Firewall in AWS via Web Portal.

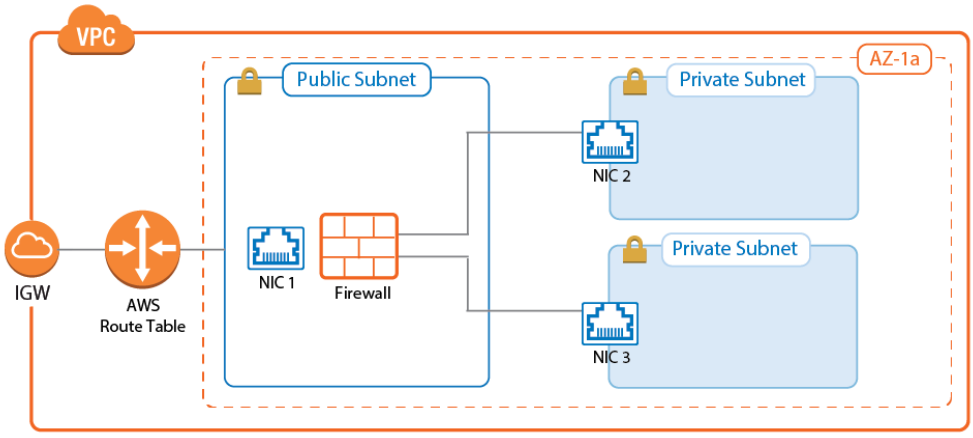


You now have a VPC with one public and multiple private subnets. A firewall instance with one network interface deployed is deployed into the public subnet.

Add one additional network interface per private subnet

The firewall must have a network interface in each private subnet. Create an AWS elastic network interface (ENI) for each private subnet in your VPC. The private IP address must be set explicitly to be able to configure the network interface statically. Also, disable the source/destination check for each interface to be able to process traffic with a destination address not matching the private IP of the network interface. Before attaching the ENIs to the firewall, shut the firewall instance down.

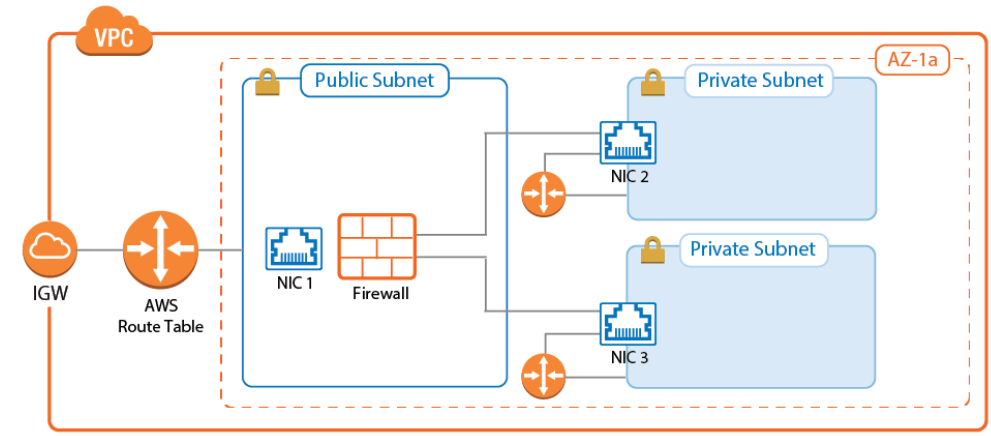
Attach the network interfaces. After starting the firewall, configure the new network interfaces and add the required direct attached routes and virtual server IP addresses.



For step-by-step instructions, see How to Add AWS Elastic Network Interfaces to a Firewall Instance.

Route table for private subnets

For each private subnet, a dedicated AWS route table handles all traffic with destinations outside the VPC. Associate the subnet with the route table and create a default route with the network interface of the firewall in this subnet as the target.



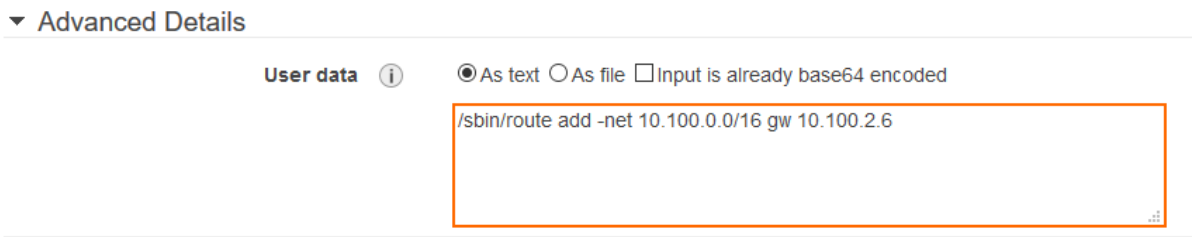
For step-by-step instructions, see How to Configure AWS Route Tables for Firewalls with Multiple Network Interfaces.

Deploying instances to use the firewall as default gateway

It is not currently possible to configure the AWS route table to send traffic between two subnets through the firewall instance. By default, each route table includes a static route for the VPC pointing to the AWS gateway of the subnet. This route cannot be overridden by a more specific route, nor can it be deleted. To send traffic via the firewall, add a route directly on the instance. The route can be added either manually after the instance has been deployed, or automatically in the **User data** section.

AWS console (Linux instances only)

Add the routes to **User data** field of the **Advanced Details** section.



CloudFormation (Linux instances only)

Add the definition for the routes in the **UserData** section of the CloudFormation template. If multiple private subnets are used, more than one route may be required.

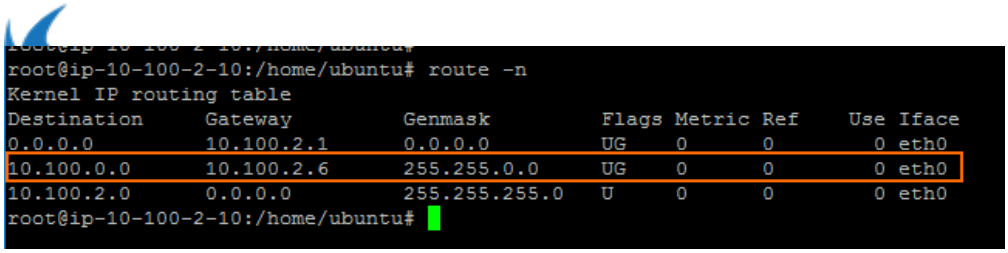
```
"UserData": { "Fn::Base64": { "Fn::Join": [ "", [ "#!/bin/bash\n\n", "/sbin/route add -net 10.100.1.0/16 gw 10.100.2.6", "\n" ] ] } } },
```

Manually (Linux instances only)

Log into the instance via ssh, and with root privileges enter:

```
root@ip-10-100-2-10:/home/ubuntu# route add -net 10.100.0.0/16 gw 10.100.2.6
```

The route is now in the route table. Enter `route -n` to list the routes:



Firewall service configuration

Now that the routing and setup in AWS is complete, access rules must be configured to apply your security policies to the traffic passing between the VPC subnets:

- **Network objects** – Create network objects for the VPC, for each subnet, and for individual instances. For more information, see Network Objects.
- **Access rules** – By default, all connections are blocked. Create access rules for each service the instances are allowed to access. Use the **FIREWALL > Live** and **FIREWALL > History** pages to verify which rule matches and which traffic is blocked. For more information, see Live Page and History Page.

Access rules allowing the backend instances access to the Internet must use the **Dynamic NAT** connection objects to rewrite the source IP of the packets to the IP address of the firewall.

Figures