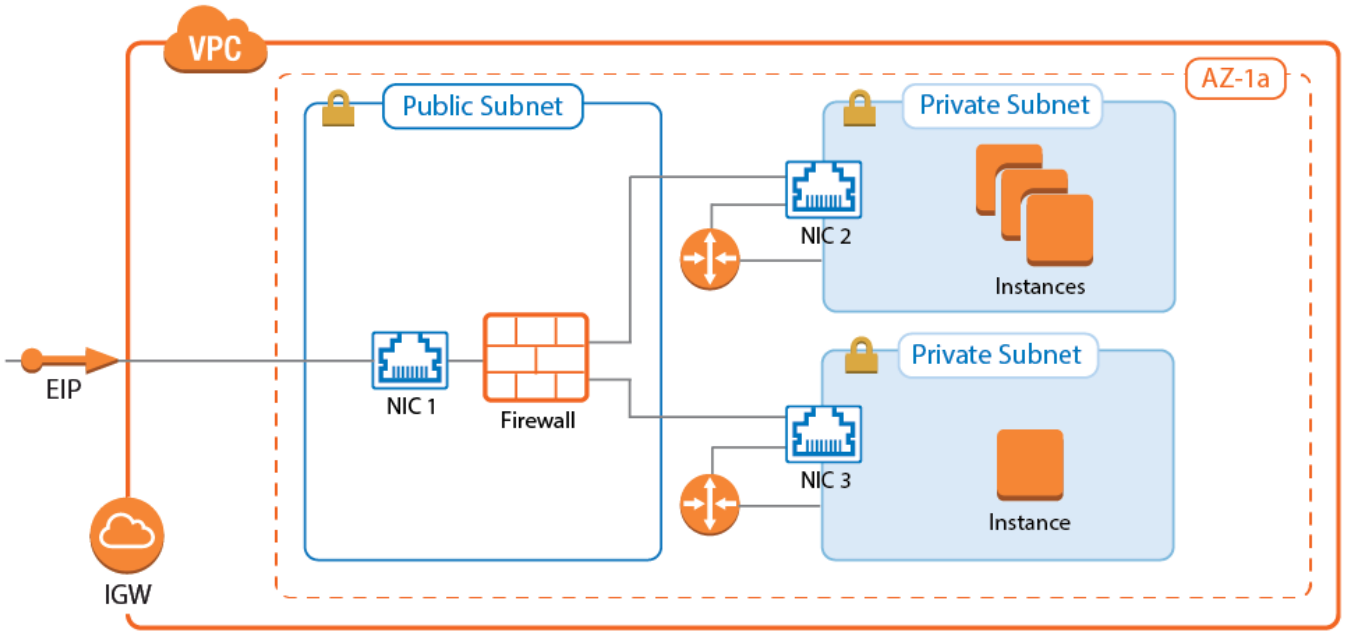
AWS Reference Architecture - Segmentation Firewall for Single AZ VPCs

https://campus.barracuda.com/doc/79462689/

A CloudGen Firewall 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 CloudGen 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 CloudGen Firewall is vastly superior regarding the depth at which both traffic can be inspected and security policies applied. In addition, Barracuda Firewall Admin also provides real-time traffic visibility, and the Firewall Live and History pages allow quick, fine-grained access to all the traffic currently passing through the firewall.

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.



Use Cases for a Multi-NIC Segmentation Firewall

A CloudGen Firewall Segmentation is deployed like an internal firewall for applications moved to AWS using lift-and-shift migrations.

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.

(Alternative) Deploying a Segmentation Firewall via AWS Console

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

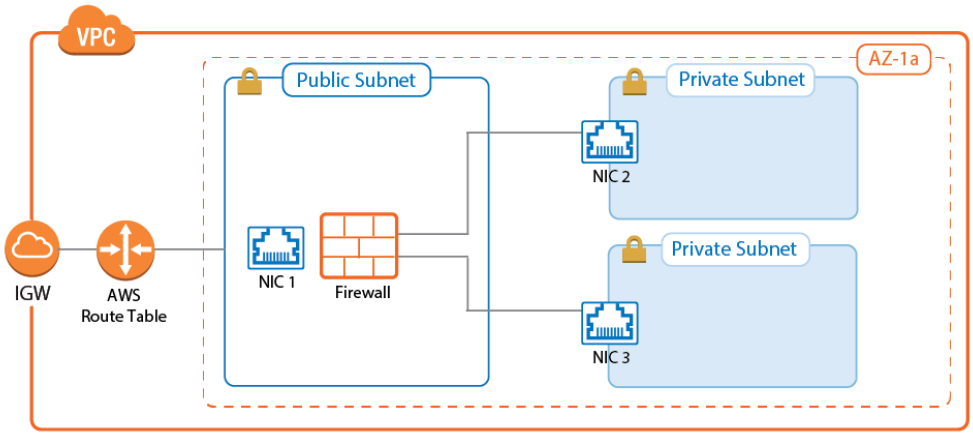
- Create a VPC with the public and private subnets all in one Availability Zone.
- Launch a CloudGen Firewall instance into the public subnet.
- Add an additional ENI per private subnet.

For step-by-step instruction, see How to Deploy a CloudGen Firewall in AWS via AWS Console.

Adding Additional Network Interfaces for Each 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.

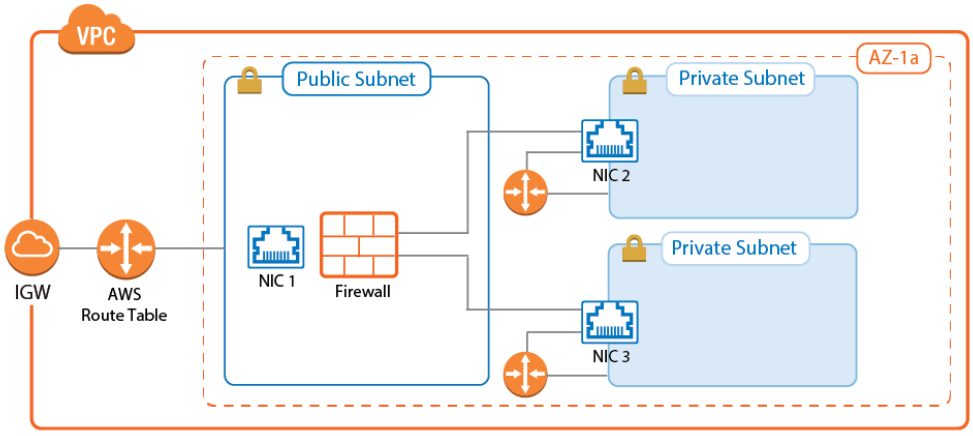
Attach the network interfaces. After starting the firewall, configure the new network interfaces and add the required direct attached routes and shared IP addresses.



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.

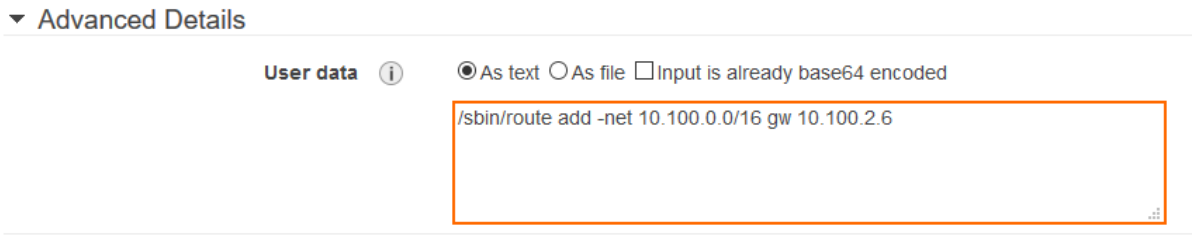
Deploying Instances to Use the Firewall as Default Gateway

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.

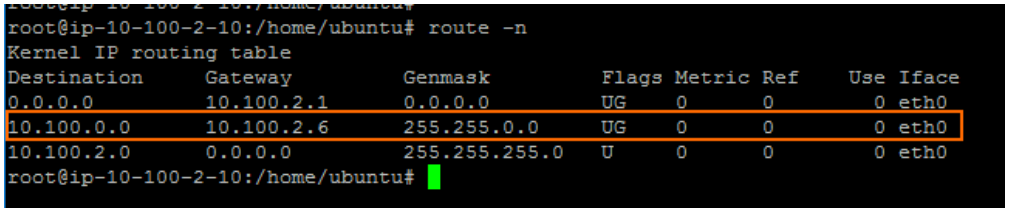
```
"UserData": {
                    "Fn::Base64": {
                        "Fn::Join": [
                            "", [
                                "#!/bin/bash\n\n",
                                "/sbin/route add -net 10.100.1.0/16 gw
10.100.2.6",
                                       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

1. segmentation.png
2. segmentation_4.png
3. segmentation_5.png
4. segmentation_client_userdata_01.png
5. segmentation_client_userdata_02.png
6. segmentation_client_userdata_03.png