

### Overview:

In the Amazon Web Services console, open the Virtual Private Cloud (VPC) console which contains all of the default security groups and custom security groups. Set restrictions on all the inbound traffic, the restrictions can control the traffic based on type, the protocol used, and the port numbers. All the ports are allowed inbound, although that isn't necessarily the best practice. A better way would be to support a limited number of ports and deploy the window system. The user **should know what ports they need** when they are building the instance.

Below is an image of the VPC Dashboard

The screenshot shows the AWS VPC Dashboard. On the left is a sidebar with navigation links: VPC Dashboard, Filter by VPC: (set to None), Virtual Private Cloud, Your VPCs, Subnets, Route Tables, Internet Gateways, Egress Only Internet Gateways, DHCP Options Sets, and Elastic IPs. The main area is titled 'Resources' and contains two buttons: 'Start VPC Wizard' and 'Launch EC2 Instances'. A note states: 'Note: Your Instances will launch in the US West (Oregon) region.' Below this, it says 'You are using the following Amazon VPC resources in the US West (Oregon) region:' followed by a list of resources in two columns: 1 VPC, 0 Egress-only Internet Gateways, 1 Route Table, 0 Elastic IPs, 0 Endpoints, 1 Security Group, 0 VPN Connections, 0 Customer Gateways, 1 Internet Gateway, 3 Subnets, 1 Network ACL, 0 VPC Peering Connections, 0 Nat Gateways, 0 Running Instances, and 0 Virtual Private Gateways.

### Discovery:

In the Virtual Private Cloud (VPC) Console, make sure to view the security groups and determine if the instances are correctly being restricted. Rules can be added to each security group to distinguish the differences for each instance and allow traffic to and from its associated instances. These rules can be modified at anytime.

### Connection Tracking

- Security groups use connection tracking to track all the information to and from the instances. The rules are applied depending on the state of the connection of the traffic. The rules allow and deny the traffic from entering and leaving the instance.

Below is a list of the Inbound Rules

| Inbound |                  |          |            |           |            |
|---------|------------------|----------|------------|-----------|------------|
| Rule #  | Type             | Protocol | Port Range | Source    | Allow/Deny |
| 100     | All IPv4 traffic | All      | All        | 0.0.0.0/0 | ALLOW      |
| *       | All IPv4 traffic | All      | All        | 0.0.0.0/0 | DENY       |

For more information about VPC:

<http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-network-security.html>

For more information about Connection Tracking:

<http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-network-security.html#security-group-connection-tracking>

### **Remediation:**

To remediate open inbound port access attacks, go to the VPC Console in the Amazon Web Services Console. Visit all of the security groups and assign strict rules using the inbound rules tab. Set the access rules that will disallow the instance from being attacked again.

[Go to the Security Group page through the VPC Dashboard](#)

- Change the inbound and outbound rules for each of the individual security groups

The screenshot shows the AWS VPC Dashboard. On the left, the 'VPC Dashboard' link is highlighted with a red box. Below it is a 'Filter by VPC:' dropdown menu set to 'None'. A list of VPC resources is shown: Your VPCs, Subnets, Route Tables, Internet Gateways, Egress Only Internet Gateways, DHCP Options Sets, Elastic IPs, and Endpoints. The main area is titled 'Resources' and contains two buttons: 'Start VPC Wizard' and 'Launch EC2 Instances'. A note states: 'Note: Your Instances will launch in the US West (Oregon) region.' Below this, it says 'You are using the following Amazon VPC resources in the US West (Oregon) region:'. A list of resources is displayed in two columns. In the first column, '1 Security Group' is highlighted with a red box. The second column lists: 1 Internet Gateway, 3 Subnets, 1 Network ACL, 0 VPC Peering Connections, 0 Nat Gateways, 0 Running Instances, and 0 Virtual Private Gateways.

|                                 |                            |
|---------------------------------|----------------------------|
| 1 VPC                           | 1 Internet Gateway         |
| 0 Egress-only Internet Gateways | 3 Subnets                  |
| 1 Route Table                   | 1 Network ACL              |
| 0 Elastic IPs                   | 0 VPC Peering Connections  |
| 0 Endpoints                     | 0 Nat Gateways             |
| 1 Security Group                | 0 Running Instances        |
| 0 VPN Connections               | 0 Virtual Private Gateways |
| 0 Customer Gateways             |                            |

### Prevention:

Amazon S3 Access Control Lists (ACLs) enables the user to manage access to the buckets and objects. Each bucket and object has an ACL attached to its subresources. When a request is received against a resource, Amazon S3 checks the corresponding ACL to verify the requester has the necessary access permissions. ACLs limit the amount of people with capabilities to edit security groups and stand up instances on a network. The security groups will restrict permissions and capabilities of each of the instances.

### [Go to the Network ACL Page](#)

The screenshot shows the AWS VPC Dashboard. On the left, the 'VPC Dashboard' link is highlighted with a red box. The main area shows a list of VPC resources in the US West (Oregon) region. The '1 Network ACL' link is highlighted with a red box.

| Resource Type                 | Count |
|-------------------------------|-------|
| VPC                           | 1     |
| Internet Gateway              | 1     |
| Egress-only Internet Gateways | 0     |
| Subnets                       | 3     |
| Route Table                   | 1     |
| Network ACL                   | 1     |
| Elastic IPs                   | 0     |
| VPC Peering Connections       | 0     |
| Endpoints                     | 0     |
| Nat Gateways                  | 0     |
| Security Group                | 1     |
| Running Instances             | 0     |
| VPN Connections               | 0     |
| Virtual Private Gateways      | 0     |
| Customer Gateways             | 0     |

Change the inbound and outbound rules to make them more secure

The screenshot shows the AWS Network ACL console. The 'Create Network ACL' button is highlighted with a blue box. The 'Subnet Associations' tab is selected. The table below shows the subnet associations for the Network ACL.

| Subnet                          | IPv4 CIDR      | IPv6 CIDR |
|---------------------------------|----------------|-----------|
| <a href="#">subnet-0173ee59</a> | 172.31.0.0/20  | -         |
| <a href="#">subnet-1b73707f</a> | 172.31.16.0/20 | -         |
| <a href="#">subnet-69db811f</a> | 172.31.32.0/20 | -         |