**GurGroup 4: Marianna Espinoza, Shantericka Greene, Rebecca Silos, Pedro Gomez, Khorey Brown, Jaydin Mohan, Justin Vargas, and Fernanda Somohana**

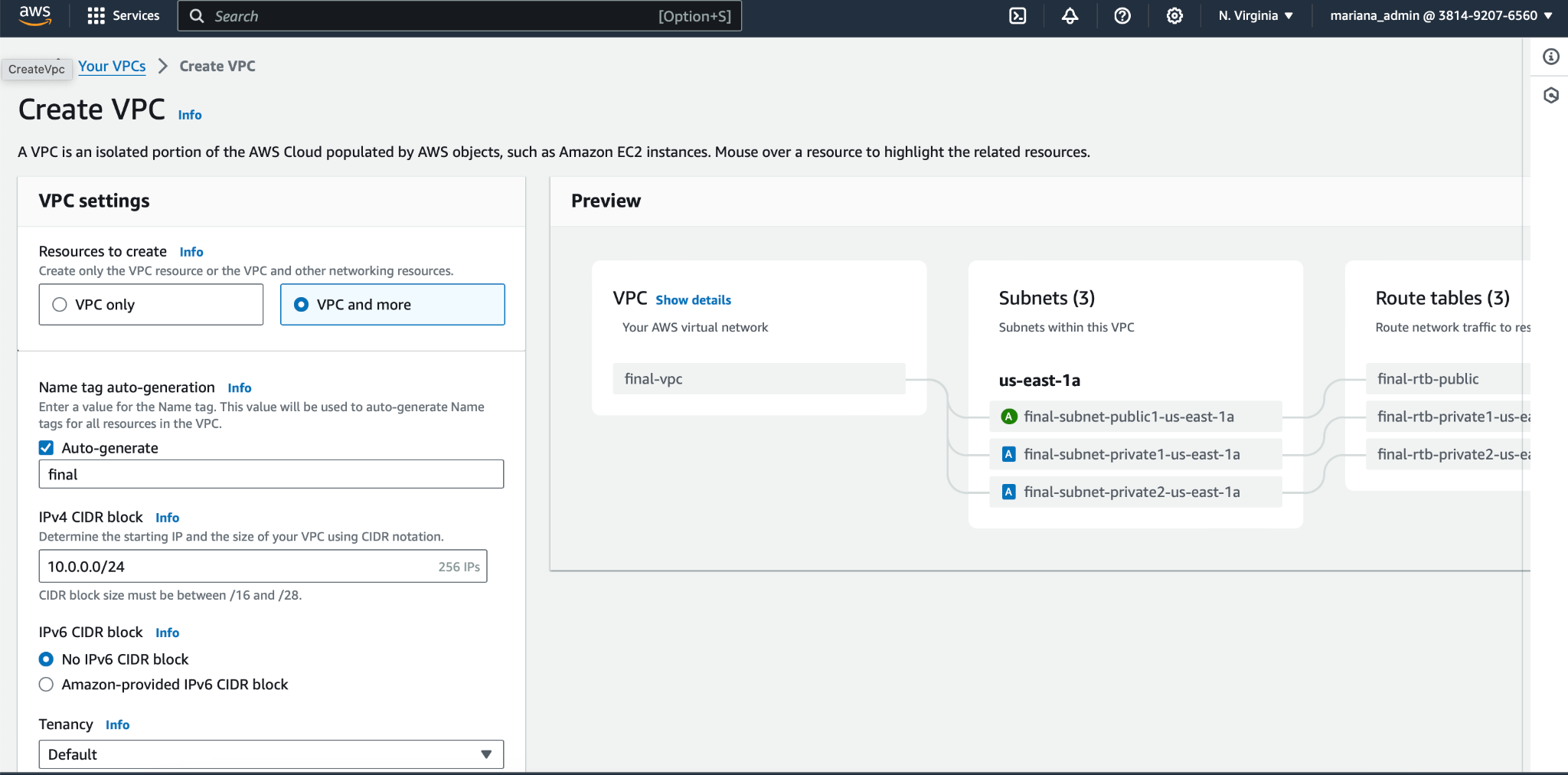
**AWS Technical Documentation**

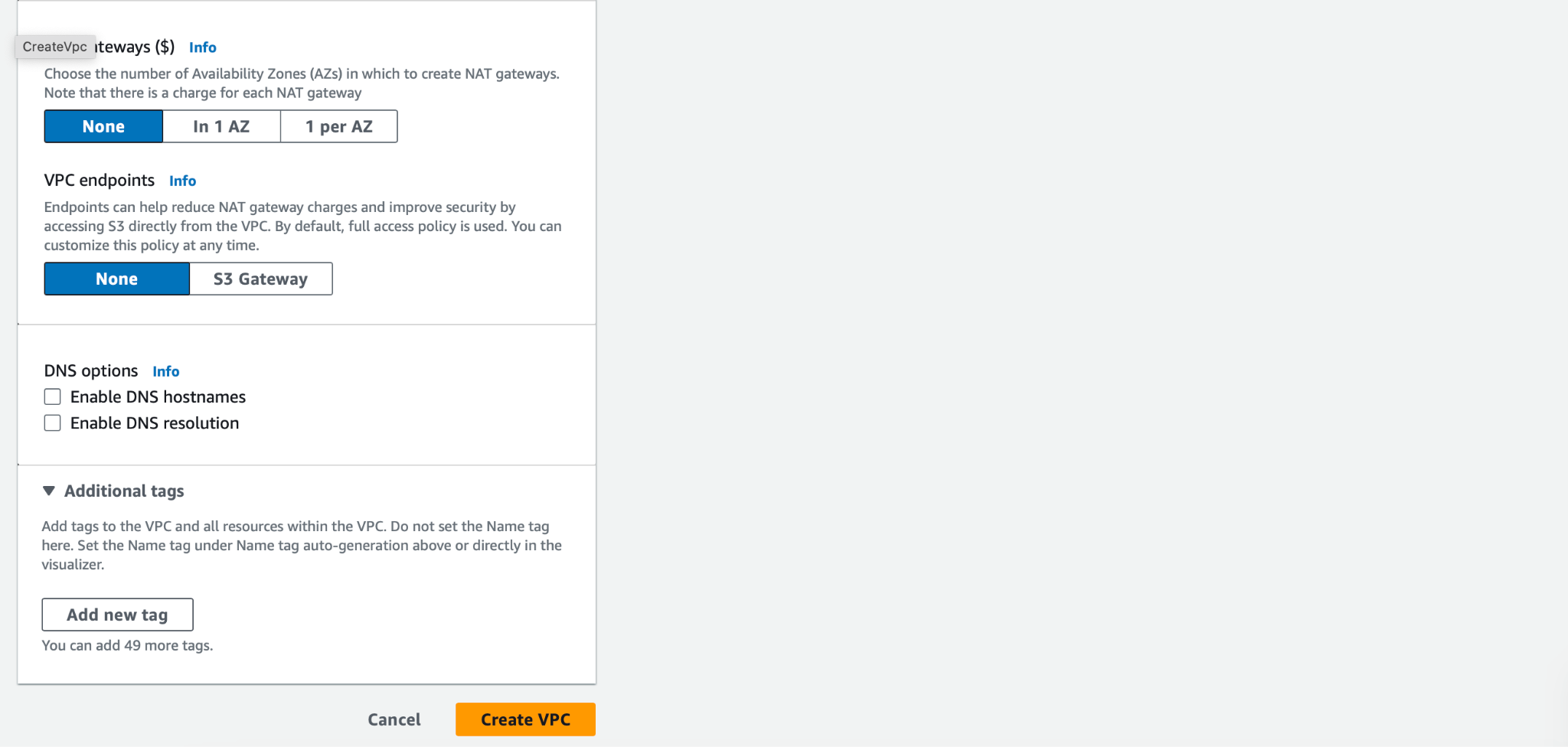
**Amazon Web Services Virtual Private Cloud (AWS VPC)**

Amazon Web Services Virtual Private Cloud is a virtual network service that allows users to launch AWS resources in a logically isolated section of the AWS cloud. With VPC, users can define their own virtual network topology, including IP address ranges, subnets, route tables, and network gateways.

**Steps for creating a VPC**

1. Log in to the AWS Management Console and search for "VPC".
2. Click on "Create VPC".
3. Choose a name for your VPC (e.g., "SecurityTrainingLab").
4. Under "CIDR Block", enter a unique CIDR block (e.g., 10.0.0.0/24). This defines the range of IP addresses for your VPC.
5. Click "Create".

**A visual example of a VPC on AWS**



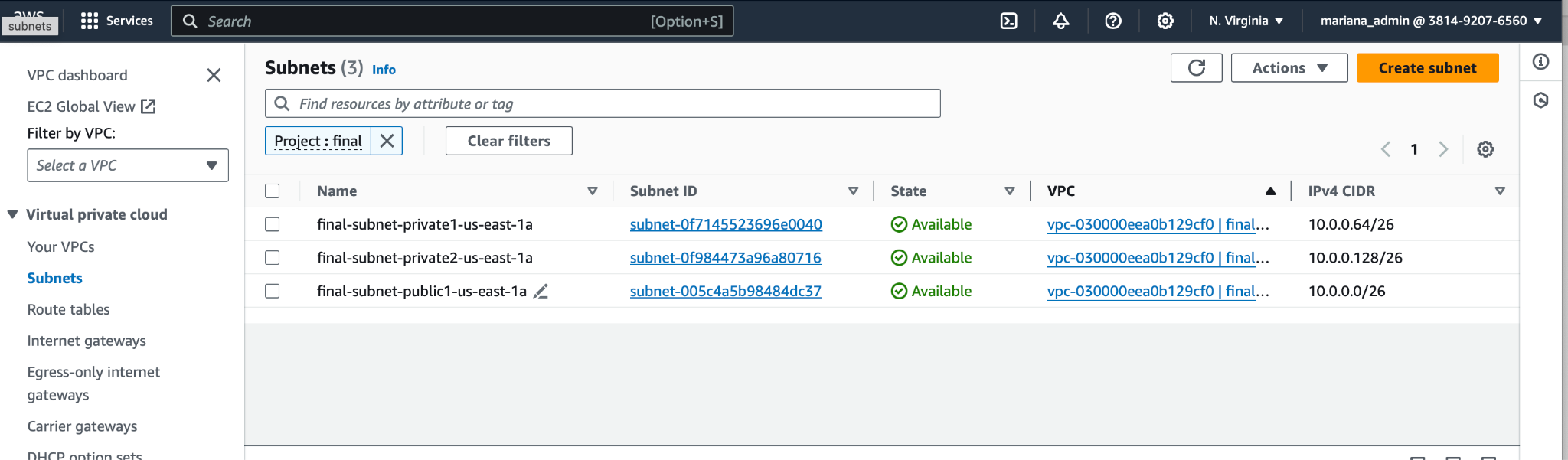
**Segmentation for Subnets**

Subnets are segments of Amazon Web Services VPC that allow you to group resources based on different network requirements such as isolation, availability zones, public and private subnets, route tables, NACLs, and subnet sizing.

* + Public subnets can have public IP addresses and can communicate with the internet directly.
  + Private subnets cannot be accessed from the internet without the use of a NAT gateway or NAT instances are used to enable instances in private subnets to access the internet while keeping them protected from inbound traffic.

**Steps for Creating Subnets**:

1. Click on "Subnets" in the VPC navigation pane.
2. Click on "Create Subnet".
3. Choose your VPC from the dropdown menu.
4. Enter a name for the subnet (e.g., "PublicSubnet").
5. Under "Availability Zone", select a zone (e.g., "us-east-1a"). Availability zones provide redundancy within a region.
6. For "CIDR Block", enter a subnet block within your VPC's CIDR block (e.g., 10.0.1.0/24). This subnet will house resources with public access (like the firewall).
7. Click "Create Subnet".
8. Repeat the process to create another subnet (e.g., "PrivateSubnet") in a different Availability Zone (e.g., "us-east-1b") with a different CIDR block within the VPC (e.g., 10.0.2.0/24). This subnet will house private resources (like the target server and Kali).

**A visual example of a subnet created below**

**Network Access Control Lists**

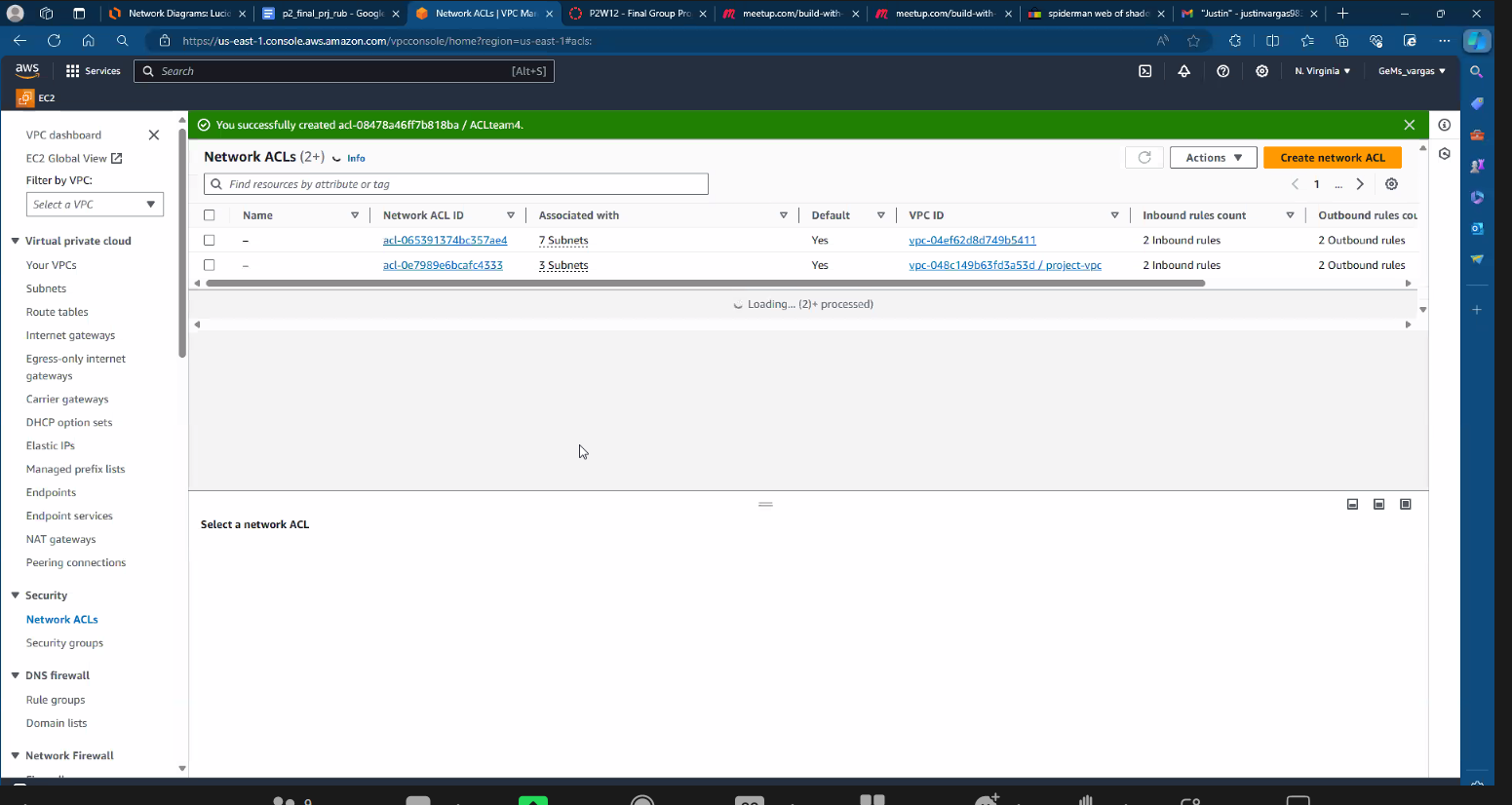
NACLs act as a firewall for controlling traffic to and from subnets in a VPC in AWS. They are stateless, meaning that they do not keep track of the state of connections.

When configuring a NACL for a public subnet you would set up rules to allow inbound traffic from the internet while still maintaining necessary restrictions for security. In contrast, when configuring a NACL for a private subnet you would typically set up rules to restrict inbound and outbound traffic based on specific IP addresses, protocols, and port numbers.

**Steps for Creating a NACLs**

1. Log in to our AWS Management Console
2. Navigate to the VPC dashboard by selecting “Services” and then “VPC” under the Networking & Content Delivery Section.
3. In the VPC dashboard, locate the “Network ACLs” option on the left-hand side menu and click on it.
4. Click on the “Create network ACL” button.
5. Enter a name for your NACL and choose the VPC where you want to create the NACL.
6. Click on the “Create” button to create the NACL.
7. Once the NACL is created, you can define inbound and bound rules for the NACL to control the traffic flow.
8. To add inbound rules, select the newly created NACL and click on the “Inbound Rules” tab. Click on the “Edit inbound rules” button to add new rules.
9. Add a new rule by specifying the rule number, protocol, port range, source IP or CIDR block, and action (allow or deny).
10. Repeat the same process to add outbound rules by selecting “Outbound Rules” tab and clicking on the “Edit outbound rules” button.
11. Once you have added the rules click the “Save” button to apply the changes to the NACL.

**A visual example a NACL created below**

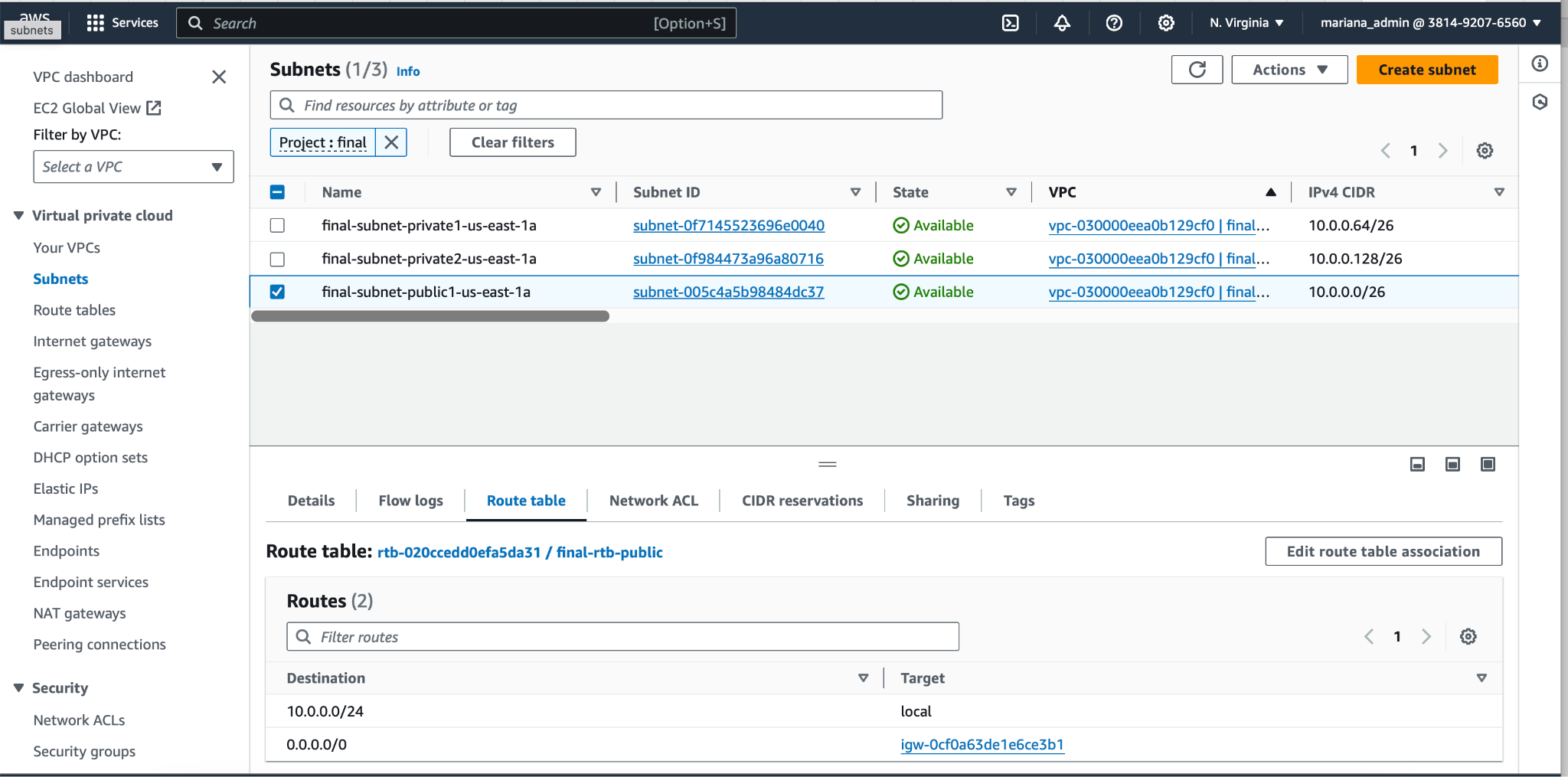


**Route Tables**

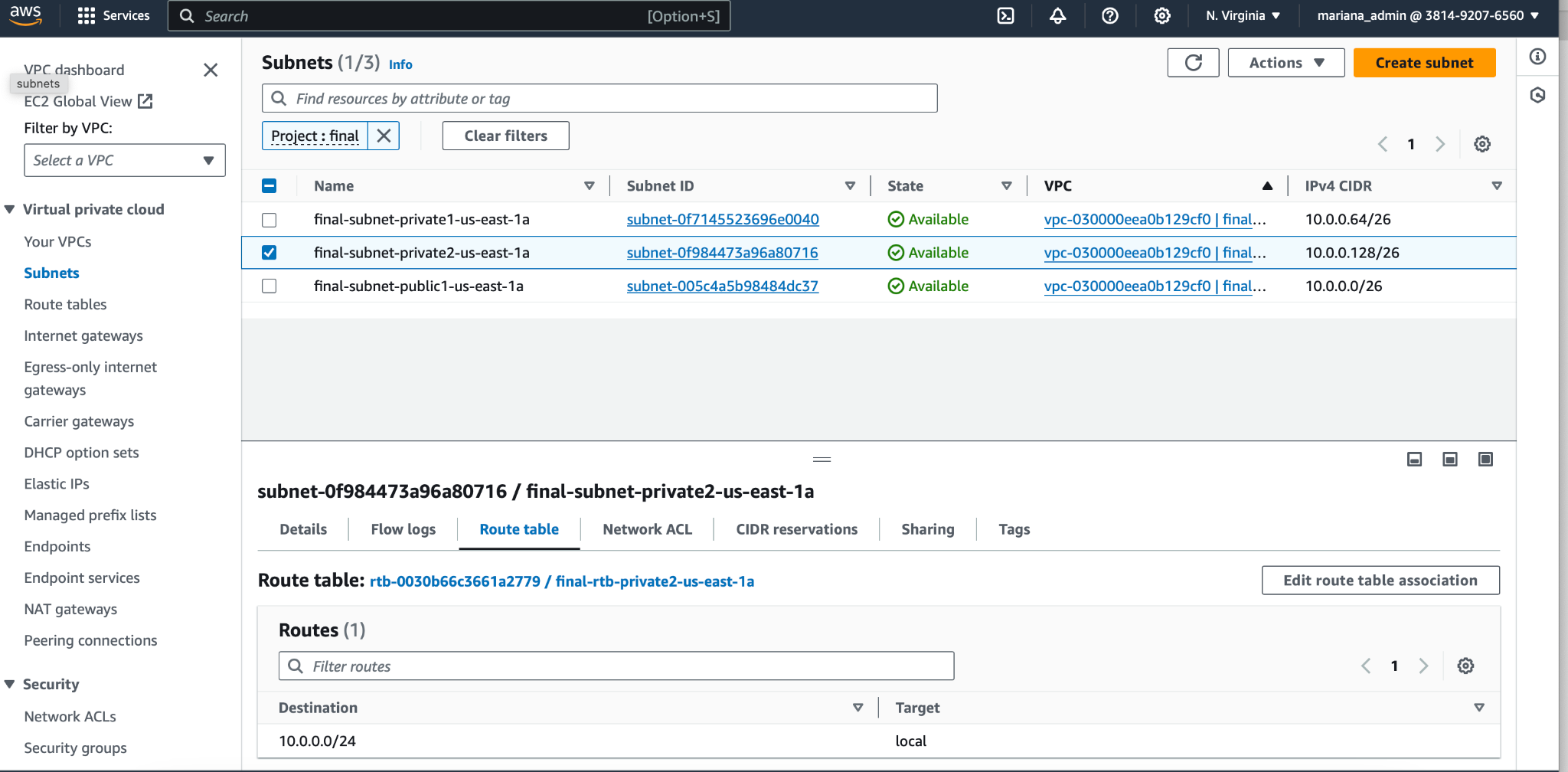
Are used to determine where traffic is directed within a VPC. They consist of a set of route rules that define how incoming and outgoing traffic is routed between subnets, internet gateways, virtual private gateways, and other network interfaces.

**Steps for Creating Route Tables**

1. In the VPC dashboard select “Services” and then “VPC under the Networking and Content Delivery Section”
2. Locate the “Route Tables” option on the left-hand side menu and click it.
3. Click on the “Create a Route Table” button.
4. Enter a name for your route table and choose the VPC where you want to create the route table.
5. Once the table is created, you can now add routes to specify where network traffic should be directed.
6. Click on the “Routes” tab of the newly created route table.
7. Click on the “Edit routes” button to add a new route.
8. Add a new route by specifying the destination CIDR block, target (e.g., an internet gateway or a specific subnet), and then status of the route (active or blackhole).
9. Click on the “Save” button to apply the route table.
10. You can also associate the route table with subnets in your VPC by clicking on the “subnet associations” tab and then clicking on the “edit subnet associations” button.
11. Select the subnet(s) you want to associate with the route table and slick on the “save” button.

**A visual example of a public route table created in the AWS lab below**

**A visual example of a private route table created below**



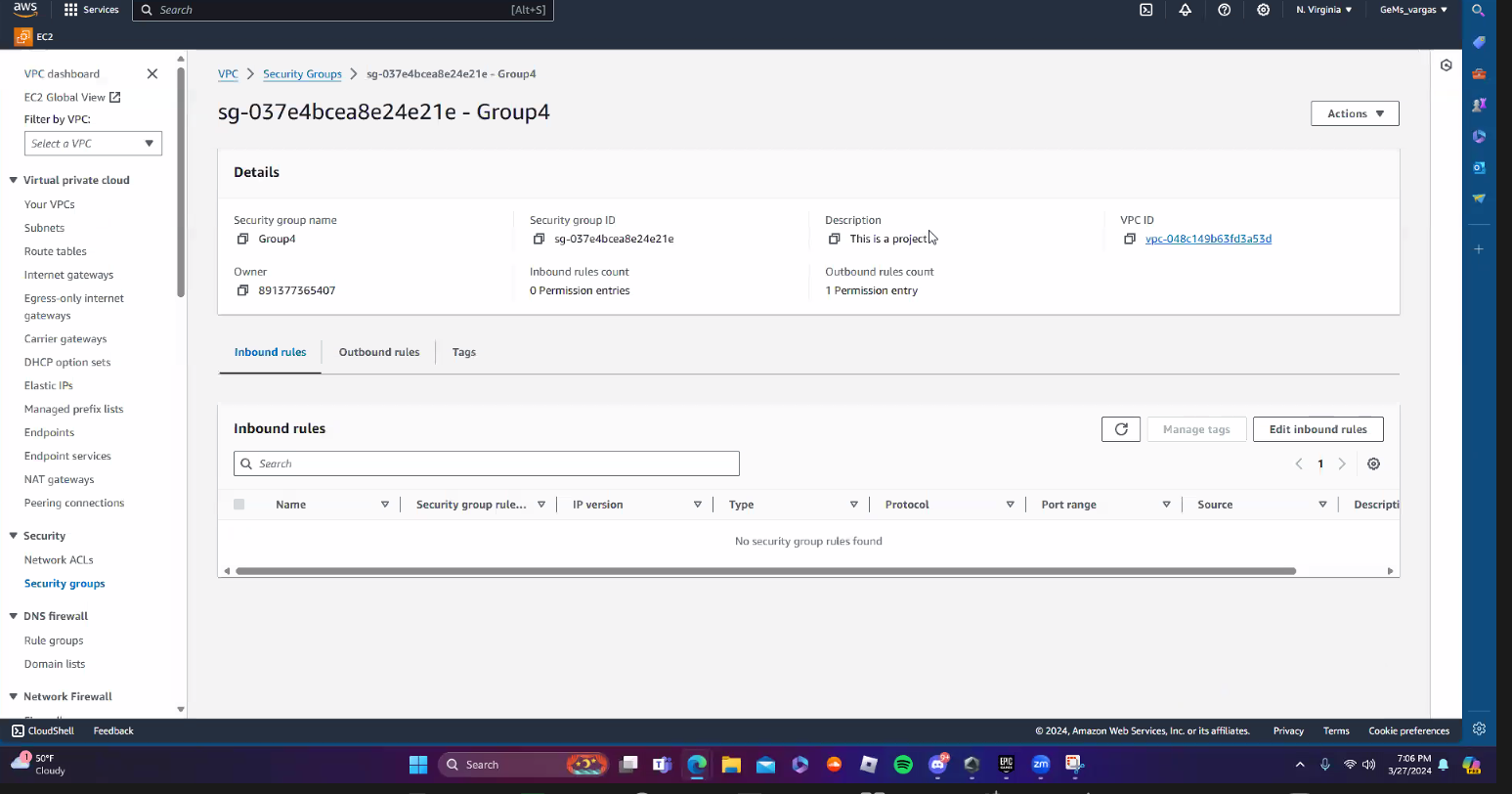
**Security Groups**

Are virtual firewalls that control inbound and outbound traffic to Amazon EC2. They act as an additional layer of security beyond the NACLs at the subnet level. Each security group contains a set of rules that define the allowed traffic based on protocols, ports, and IP addresses.The key points include: stateful, traffic rules, default allow, dynamic updates, instance associations, and granular control.

**Steps to Create a Security Group**

1. Create a Security Group:
2. On the left scroll down to Security, and click “Security Groups”
3. Then click “Create a Security Group”
4. Configure your rules for inbound and outbound traffic
5. One of the inbound rules created was the SSH rule
6. Select **Add rule** in the **Inbound rules** section.
7. In the drop-down menu, find and select **SSH**
8. Select **Source type** field to Custom. In the **Source** section, select the drop-down and select **0.0.0.0/0**
9. By default, Security Groups deny access.
10. Rules are created to allow access

**A visual example of a security group created below**



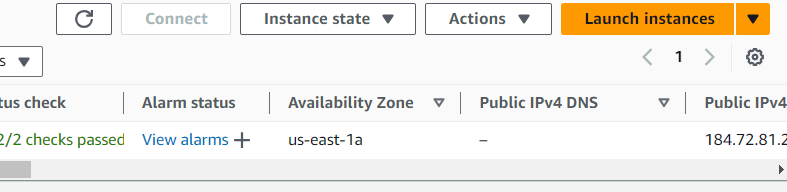
**Amazon Elastic Compute Cloud (EC2)**

EC2 instances are virtual servers that allow users to run various applications and services on the cloud. They can be easily launched, scaled, and managed. Users can choose from a variety of instance types and configurations to suit their specific needs such as CPU, memory, storage, and network capacity.

**Steps to Launch a EC2 Instance**

1. Navigate to the EC2 dashboard and click the “Launch Instance” button.
2. Choose an Amazon Machine Image (AMI) this is used to run the instance
3. Select an instance type that is best suitable for you
4. Configure install details
5. Add storage
6. Configure security groups
7. Review and launch
8. Choose or create a key repair; ensure you connect to an instance using a SSH if needed
9. Launch the instance. Once you are ready click the “Launch” button to start the instance. You will then be able to access the instance.

**A visual example of an EC2 launch below**

****

The training tools used in our technical labs include the following:

* + Kali was used for security testing in our AWS environment
  + Vulnerable target (Ubuntu machine) was used as a security testing tool and to protect from potential threats and attacks.
  + Bastion host was used as a component for remote access to instances in the private subnet and provided a secure way for users to connect to reduce risk.

**Network Monitoring**

In case you suspect unauthorized users gaining access to AWS services, you can use Network Access Analyzer tool to discover source of intrusion. Scans and Network Access Scopes allow for tailoring of analysis to your specific needs. The number of elastic network interfaces investigated determines the cost of a Network Access Analyzer analysis.

**SIEM**

SIEM stands for Security Information Event Management. It is a product that aggregates the event data that is collected by monitoring, assessment, detection and response solutions deployed across the application, network, endpoint and cloud environments. SIEM can be used in both cloud environments and on-premises.

One of the major differences between Network Monitors (such as Network Access Analyzer) and SIEM tools is that SIEM follows data through everything. This includes network and endpoint devices. Network Monitoring tools focus on data passing through networks.

GuardDuty was used as the SIEM tool in our virtual lab. It was enabled into our AWS environment to monitor for security threats. The configuration was sent for security findings and alerts to your SIEM tool. This integration allows the SIEM solution to analyze security data generated by GuardDuty. Next centralized monitoring and analysis was used to accumulate and correlate security data sources, such as log files, network traffic, and endpoint activity to help centralize views and help identify and prioritize security incidents effectively. Rules and alerts were set to trigger notifications and responses based on the security findings received from GuardDuty. This approach enables responses to quickly find security threats. These security incidents were investigated to identify the root cause of anomalies, and take appropriate actions. Lastly continuous reviewing and analyzing the security data in our SIEM solution to identify patterns, trends, and emerging threats to stay ahead of potential security risks. By following these steps we were able to enhance our lab as a group, improve threat detection capabilities, and effectively respond to incidents in our environment.