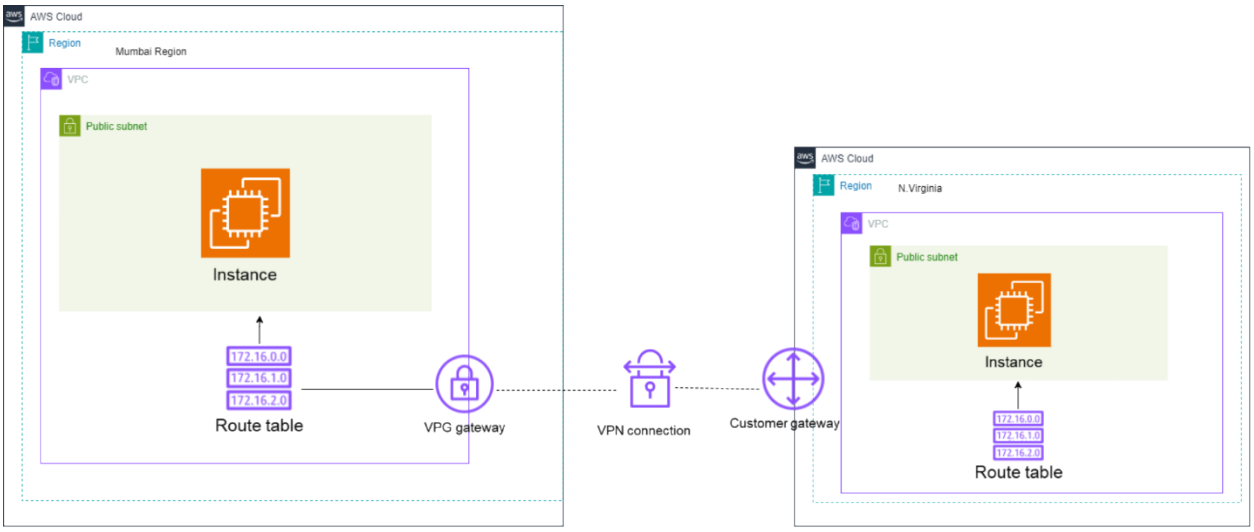
AWS SITE TO SITE VPN PROJECT

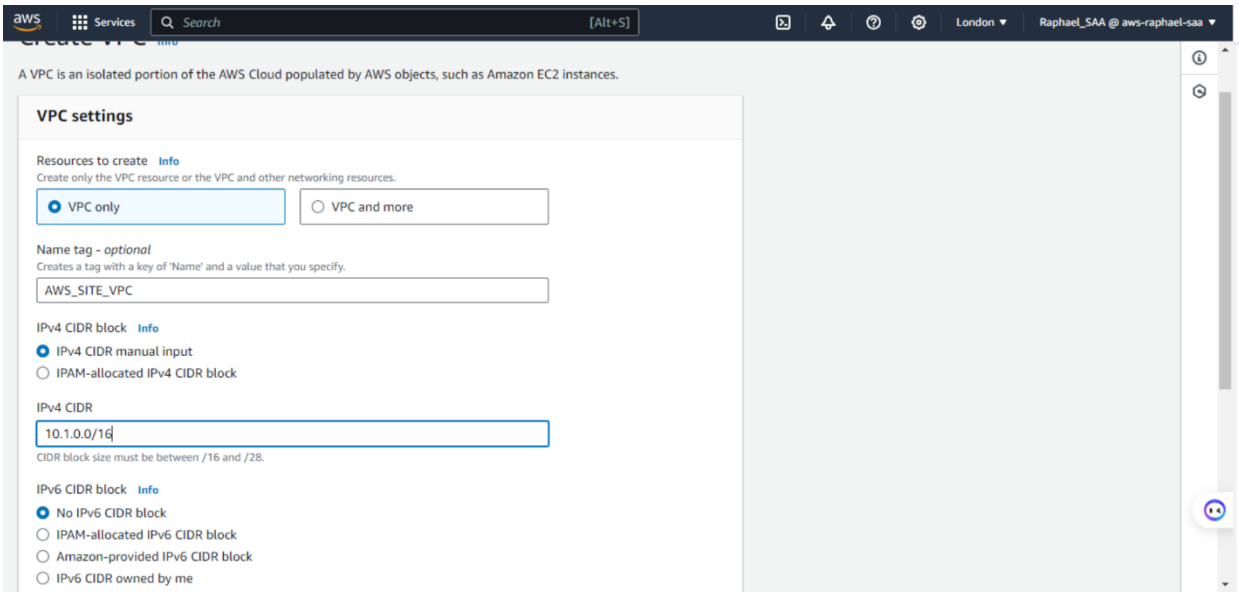
Architecture of Site-to-Site VPN

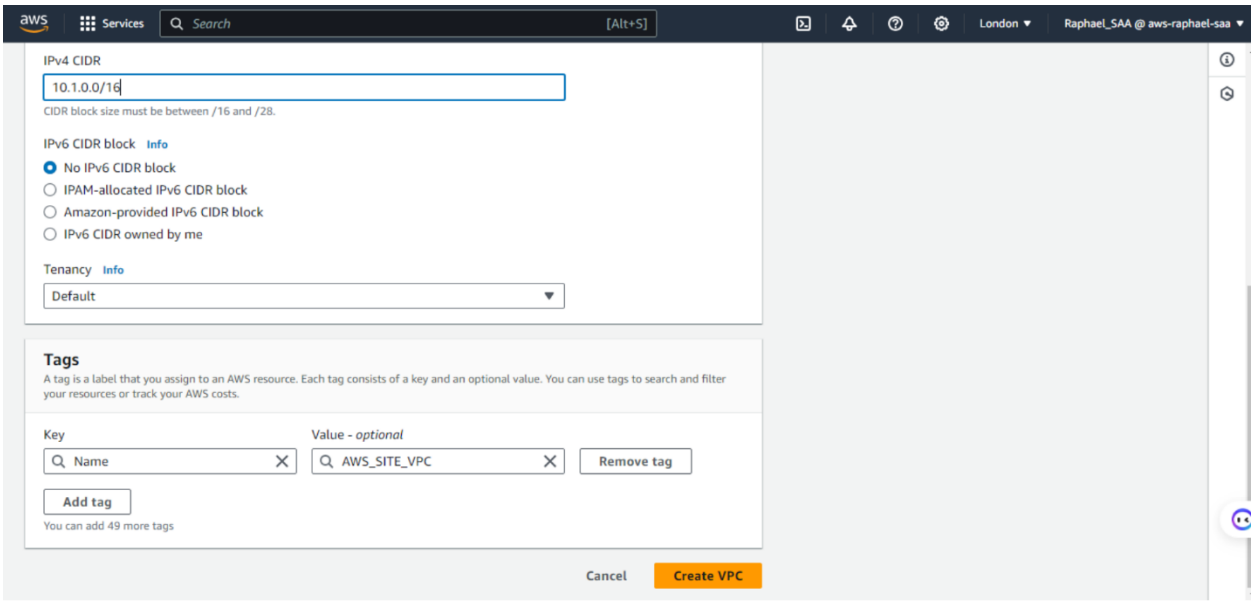


Steps for Setting Up an AWS Site-to-Site VPN

Step 1: Create a VPC in London Region

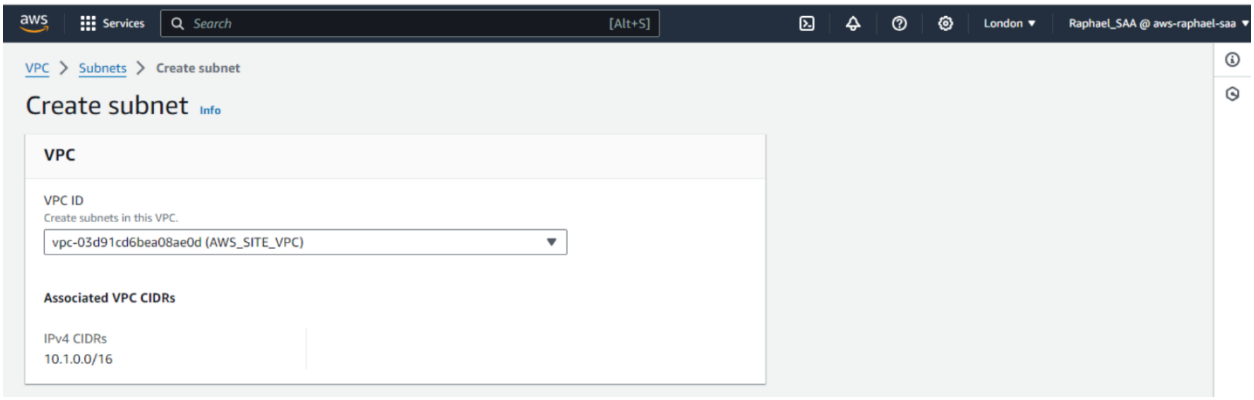
Create a Virtual Private Cloud named ‘AWS_SITE_VPC’ with CIDR block ’10.1.0.0/16’.

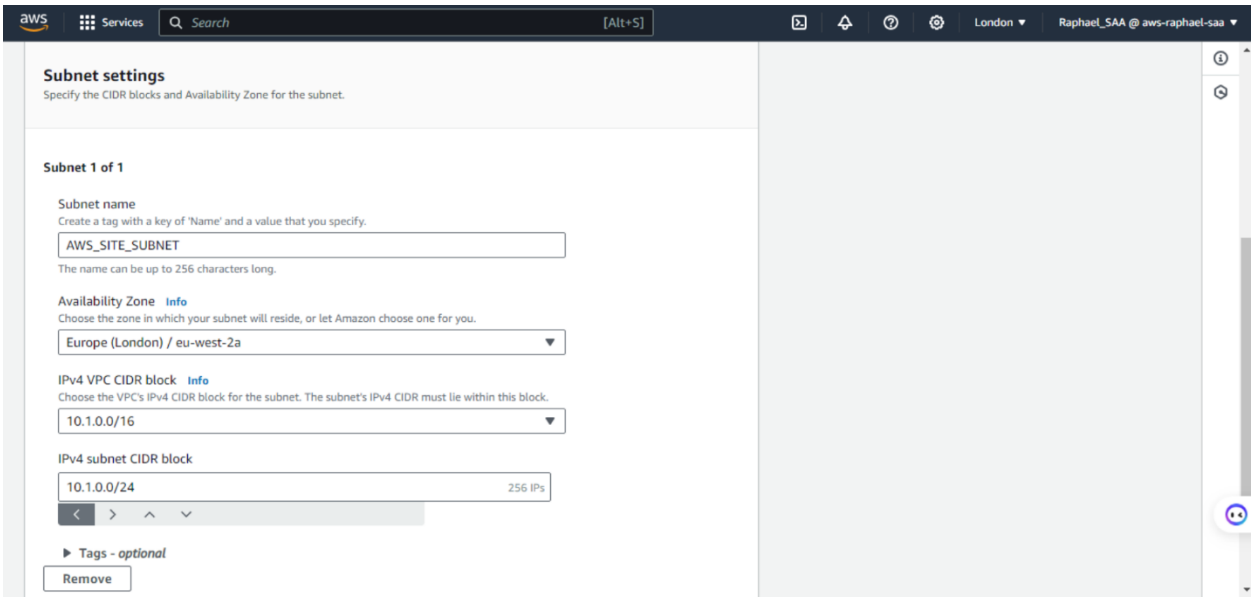




To create a subnet within our VPC 'AWS_SITE_VPC' in the London region (eu-west-2), follow these steps:

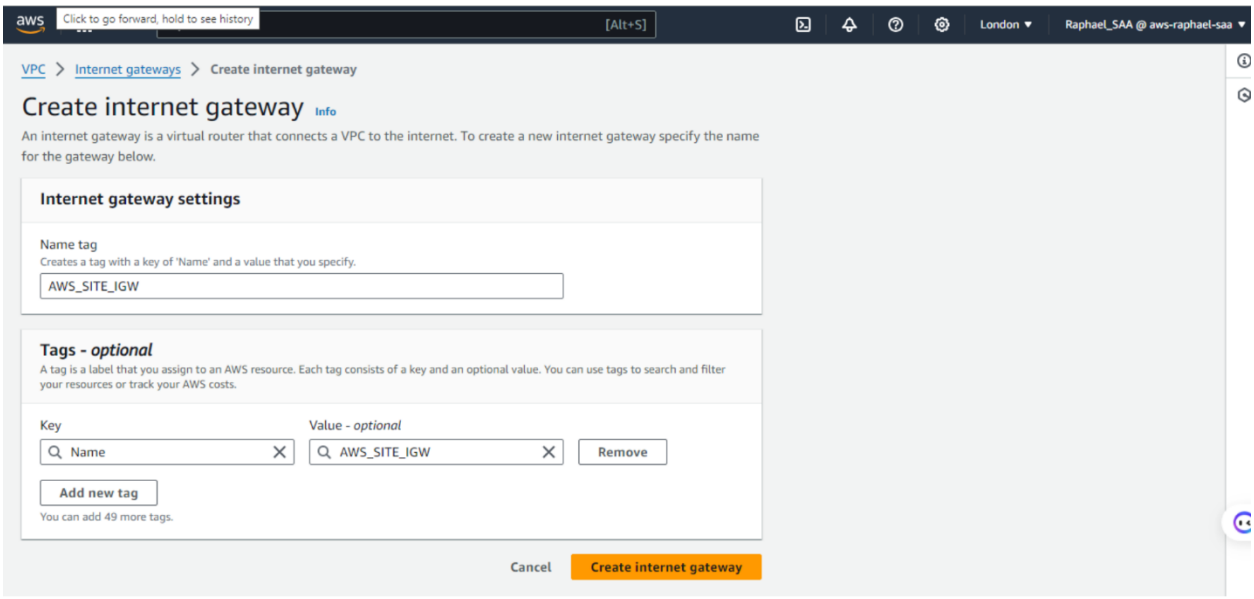
1. Name of the Subnet: AWS_SITE_SUBNET
2. VPC Name: AWS_SITE_VPC
3. Availability Zone: eu-west-2a
4. IPv4 CIDR Block: 10.1.0.0/24
5. VPC CIDR Block: **10.1.0.0/16"**

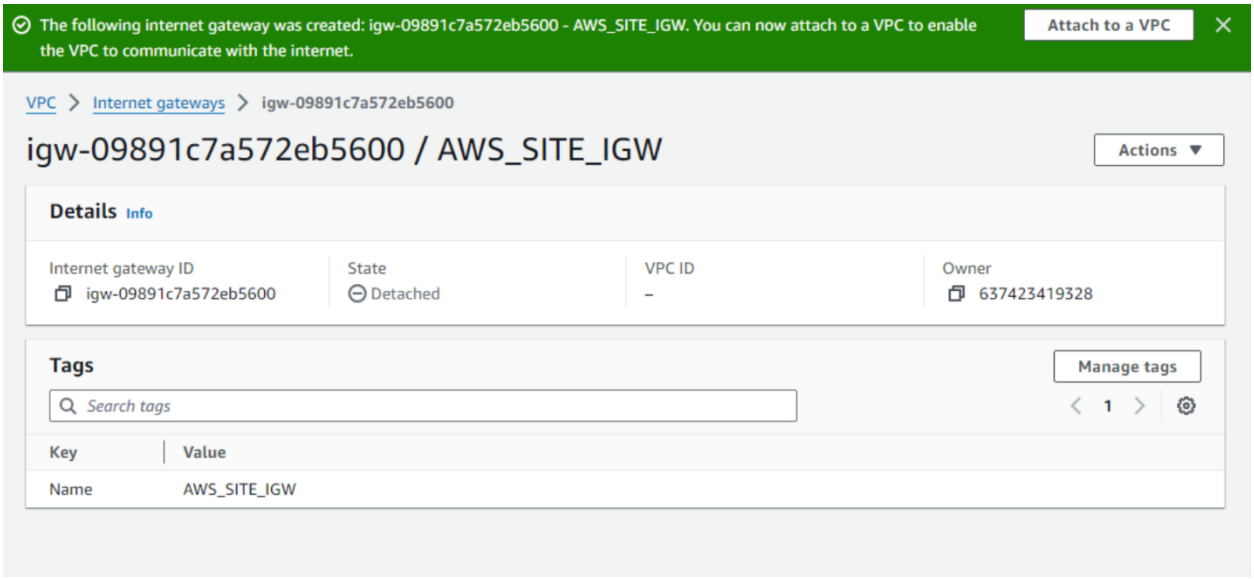


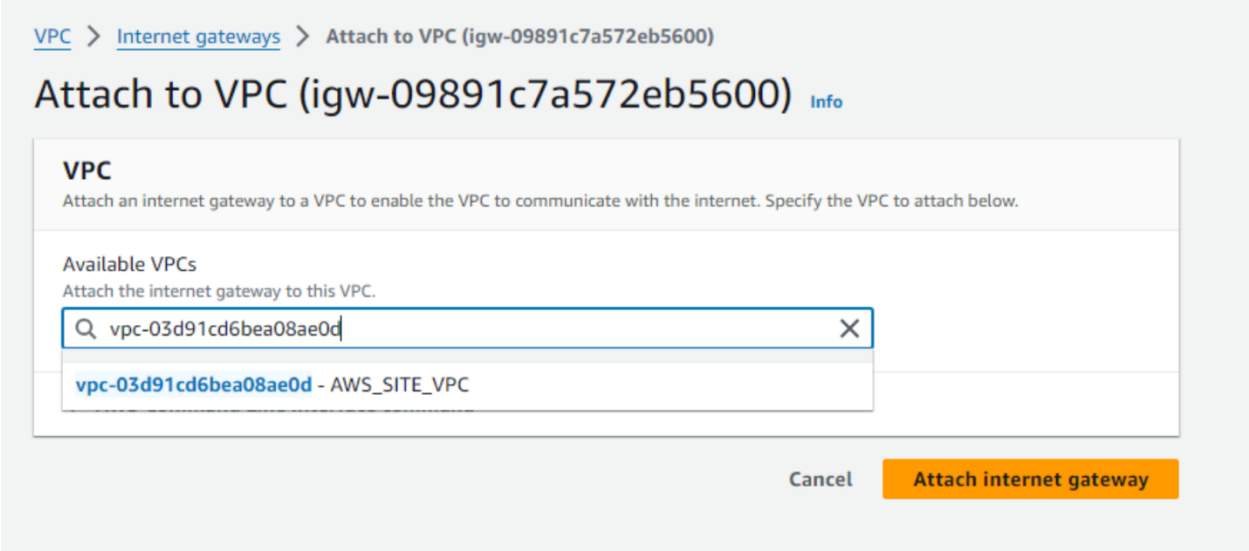


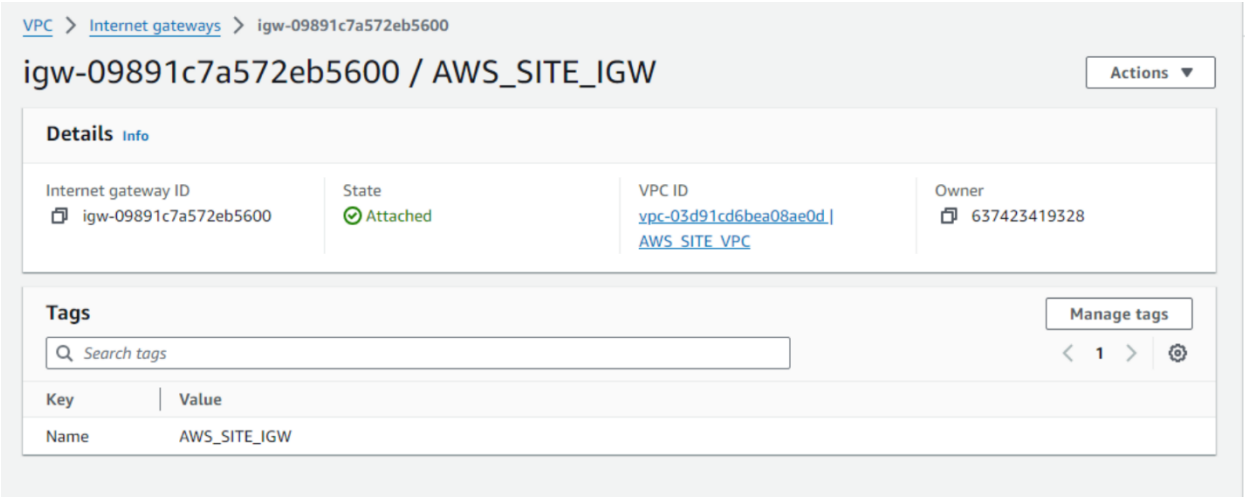
To enable internet connectivity for our VPC ‘AWS_SITE_VPC‘ in the London region, follow these steps:

1. Select ‘Internet Gateway‘from the VPC dashboard and click on ‘Create Internet Gateway‘.
2. Enter the following details:
    - Name: AWS_SITE_IGW
3. Click on ‘Create Internet Gateway’.
4. Next, attach the Internet Gateway to the VPC:
    - Select the Internet Gateway ‘AWS_SITE_IGW‘.
    - Click on ‘Actions’ and choose ‘Attach to VPC‘.
    - Select ‘AWS_SITE_VPC‘ as the VPC to attach the Internet Gateway to.
5. Confirm the attachment.


aws
Click to go forward, hold to see history
[Alt+S]
London
Raphael_SAA @ aws-raphael-saa
VPC > Internet gateways > Create internet gateway
Create internet gateway Info
An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.
Internet gateway settings
Name tag
Creates a tag with a key of 'Name' and a value that you specify.
AWS_SITE_IGW
Tags - optional
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.
Key
Value - optional
Name
AWS_SITE_IGW
Remove
Add new tag
You can add 49 more tags.
Cancel
Create internet gateway

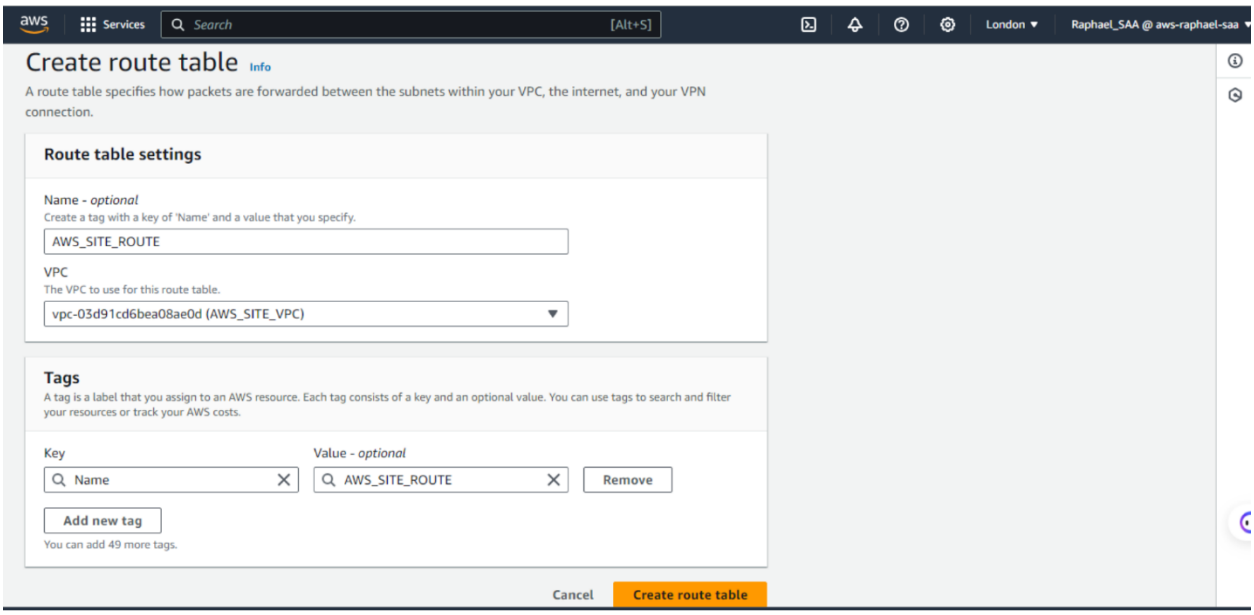

The following internet gateway was created: igw-09891c7a572eb5600 - AWS_SITE_IGW. You can now attach to a VPC to enable the VPC to communicate with the internet.
Attach to a VPC
VPC > Internet gateways > igw-09891c7a572eb5600
igw-09891c7a572eb5600 / AWS_SITE_IGW
Actions
Details Info
Internet gateway ID
igw-09891c7a572eb5600
State
Detached
VPC ID
–
Owner
637423419328
Tags
Manage tags
Search tags
1
Key
Value
Name
AWS_SITE_IGW





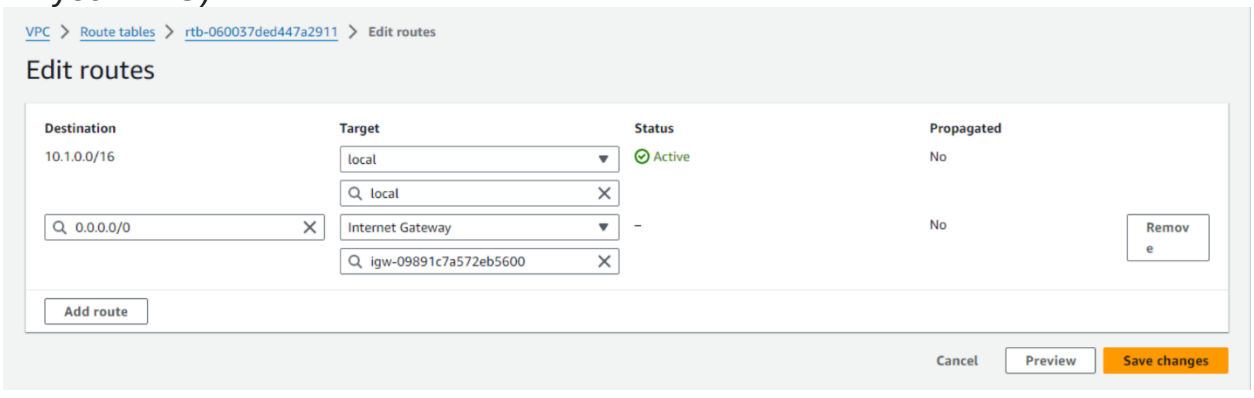
To manage routing within our VPC ‘AWS_SITE_VPC‘ in the London region, let’s create and configure a route table:

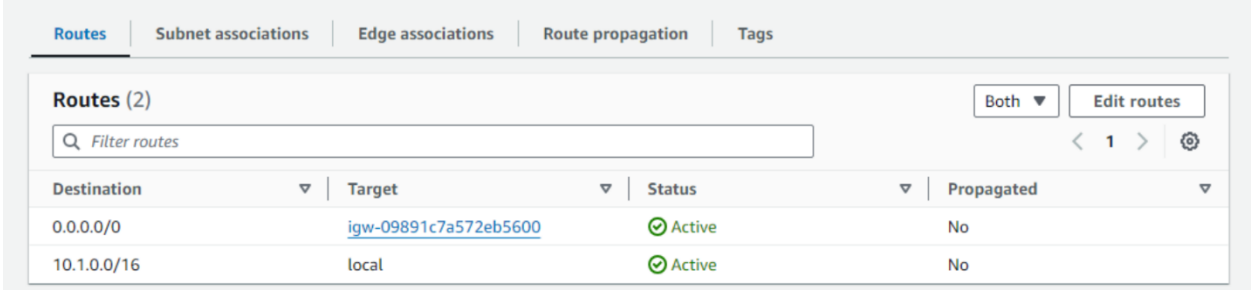
1. Navigate to the Route Tables section in the VPC dashboard.
2. Click on ‘Create Route Table‘.
3. Enter the following details:
    - Name: AWS_SITE_ROUTE
    - VPC: AWS_SITE_VPC
4. Click on ‘Create Route Table‘.



Now, let's edit the routes in the route table to direct traffic:

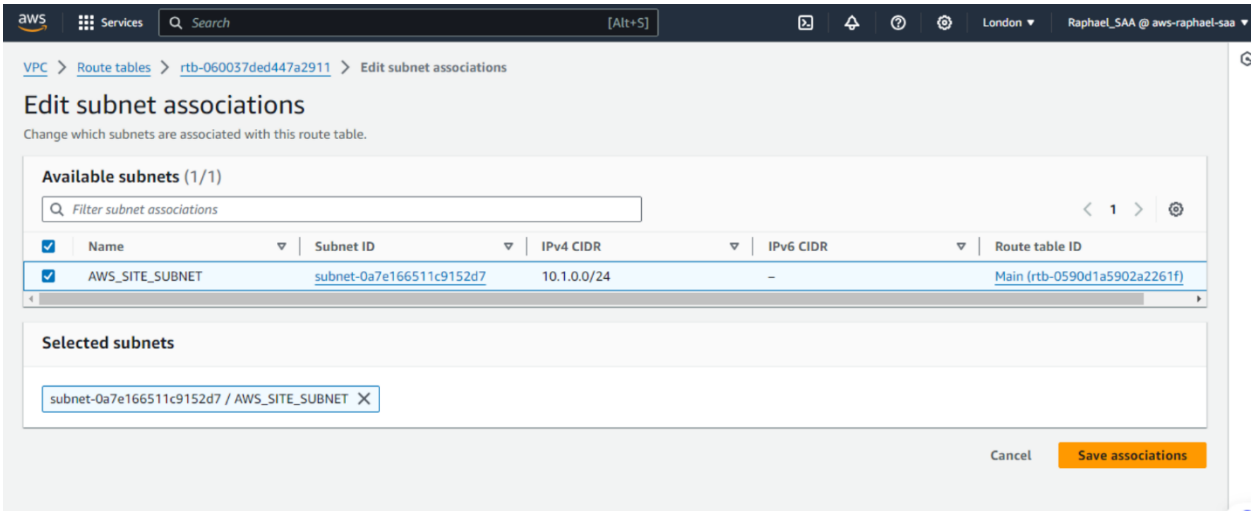
1. Select the route table '**AWS_SITE_ROUTE**'.
2. Click on '**Routes**' tab.
3. Click on '**Edit routes**'.
4. Add a new route with the following details:
   - Destination: **0.0.0.0/0**
   - Target: I**nternet Gateway** (select the Internet Gateway associated with your VPC).





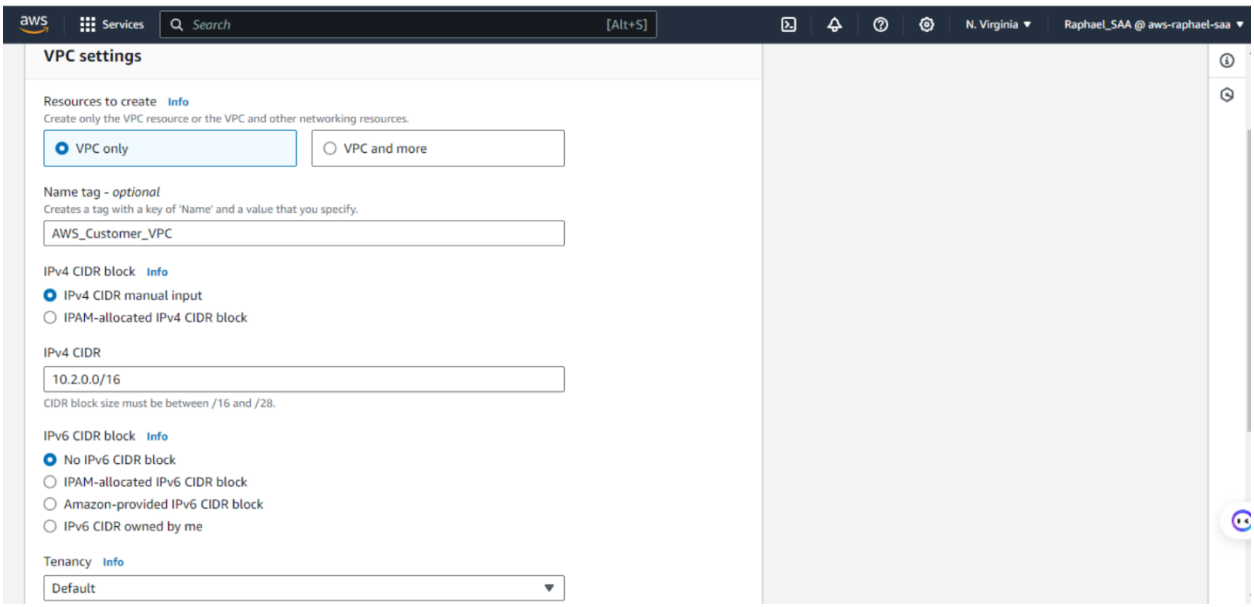
Next, associate the route table with a subnet and ensure proper association:

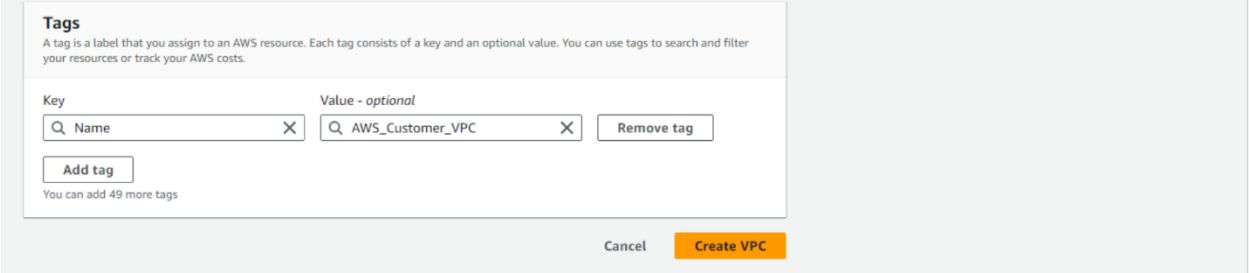
1. Click on '**Subnet Associations**' tab.
2. Click on '**Edit Subnet Associations**'.
3. Select the subnet(s) within '**AWS_SITE_VPC**' that you want to associate with this route table.
4. Click on 'Save'.



Step 2: Create a VPC in N. Virginia Region

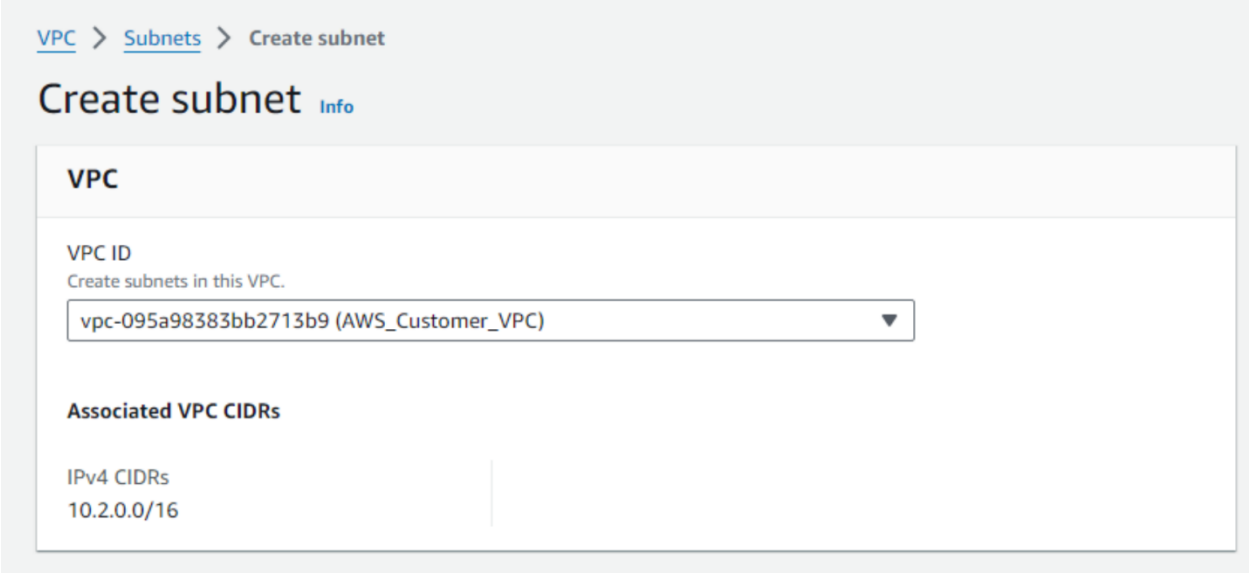
In the N.Virginia region, we have set up a Virtual Private Cloud (VPC) named "AWS_Customer_VPC' with the CIDR block '10.2.0.0/16'."

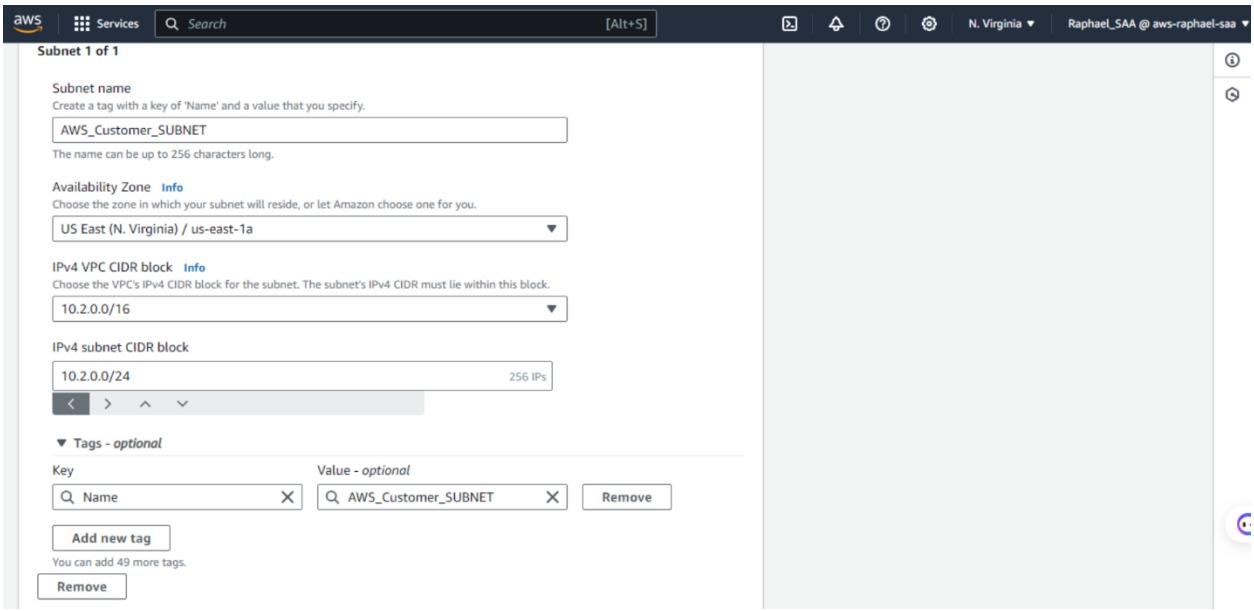




To create a subnet within our VPC '**AWS_Customer_VPC**' in the N.Virginia region (us-east-1a), follow these steps:

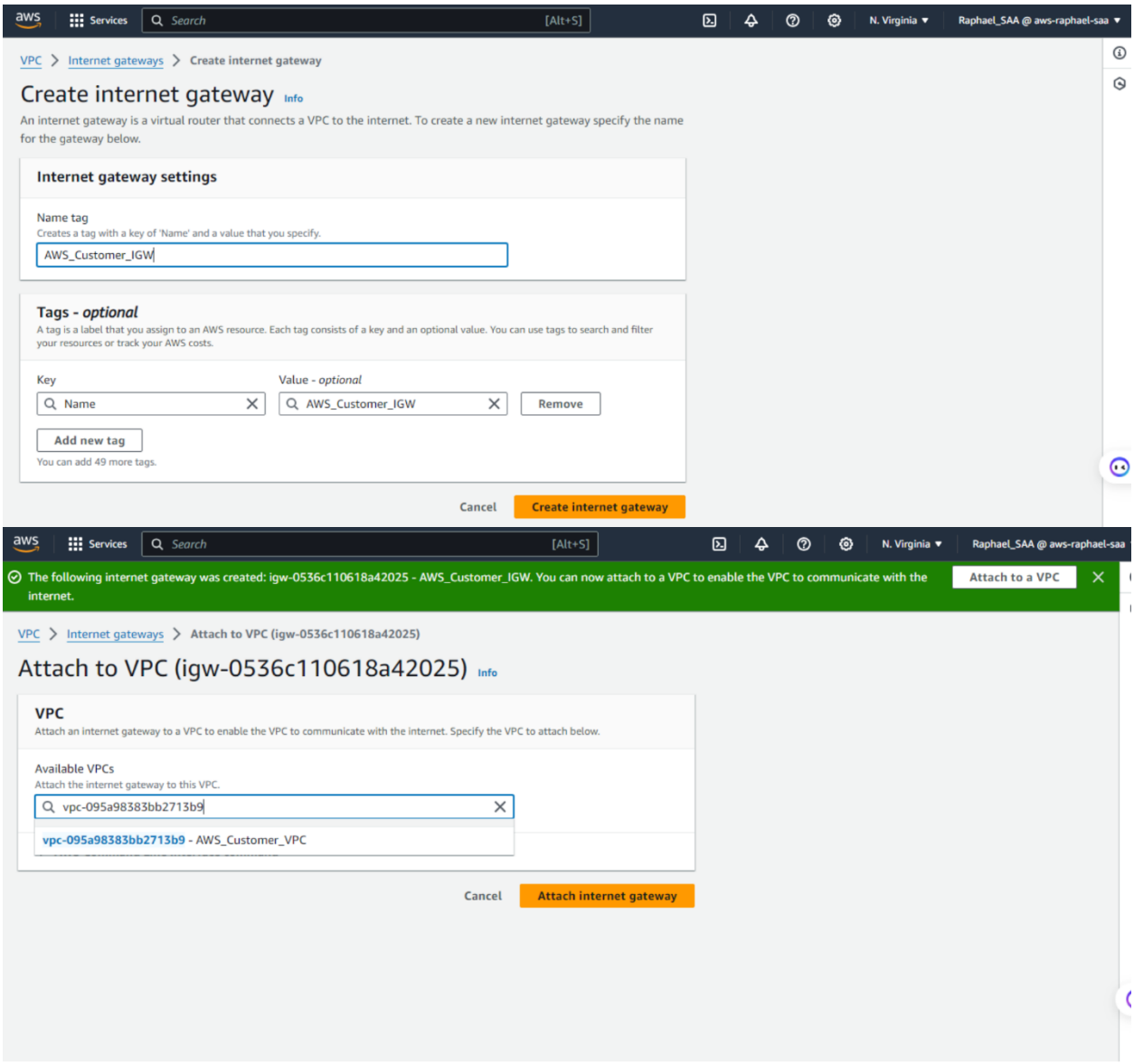
1. Name of the Subnet: **AWS_Customer_SUBNET**
2. VPC Name: **AWS_Customer_VPC**
3. Availability Zone: **us-east-1a**
4. IPv4 CIDR Block: **10.2.0.0/24**
5. VPC CIDR Block: **10.2.0.0/16"**



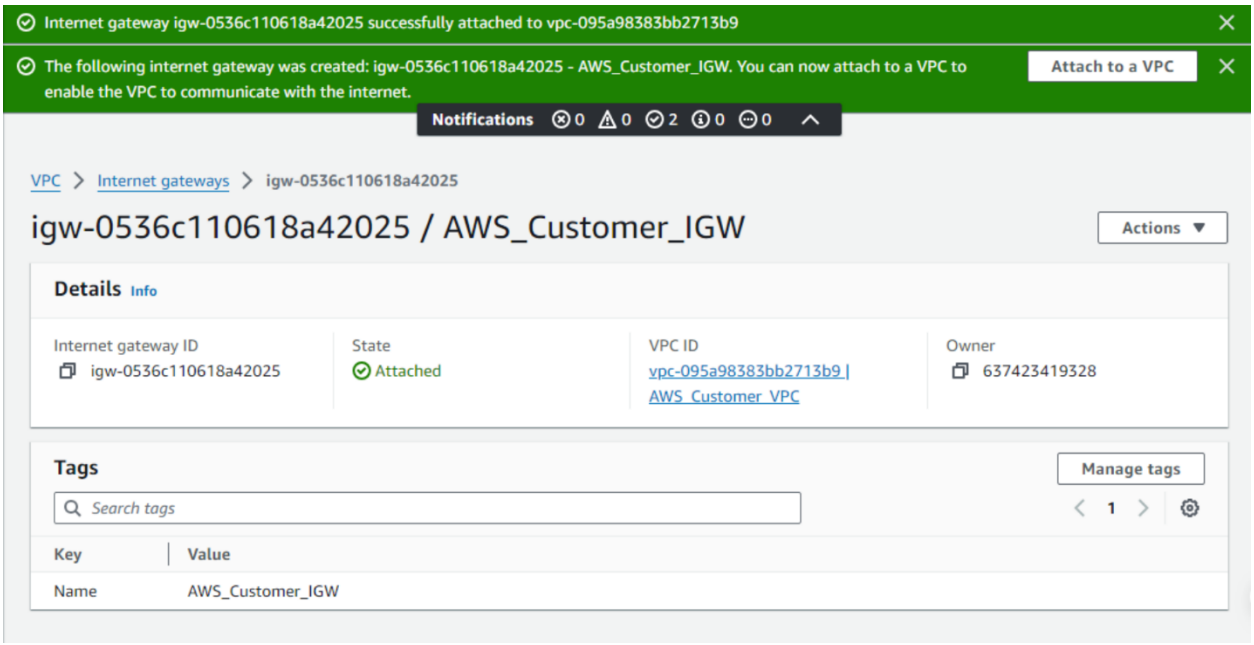


To enable internet connectivity for our VPC '**AWS_Customer_VPC**' in the N.Virginia region, follow these steps:

1. Select '**Internet Gateway**' from the VPC dashboard and click on '**Create Internet Gateway**'.
2. Enter the following details:
   - Name: **AWS_Customer_IGW**
3. Click on 'Create Internet Gateway'.
4. Next, attach the Internet Gateway to the VPC:
   - Select the Internet Gateway '**AWS_Customer_IGW**'.
   - Click on 'Actions' and choose '**Attach to VPC**'.
   - Select '**AWS_Customer_VPC**' as the VPC to attach the Internet Gateway to.
5. Confirm the attachment.

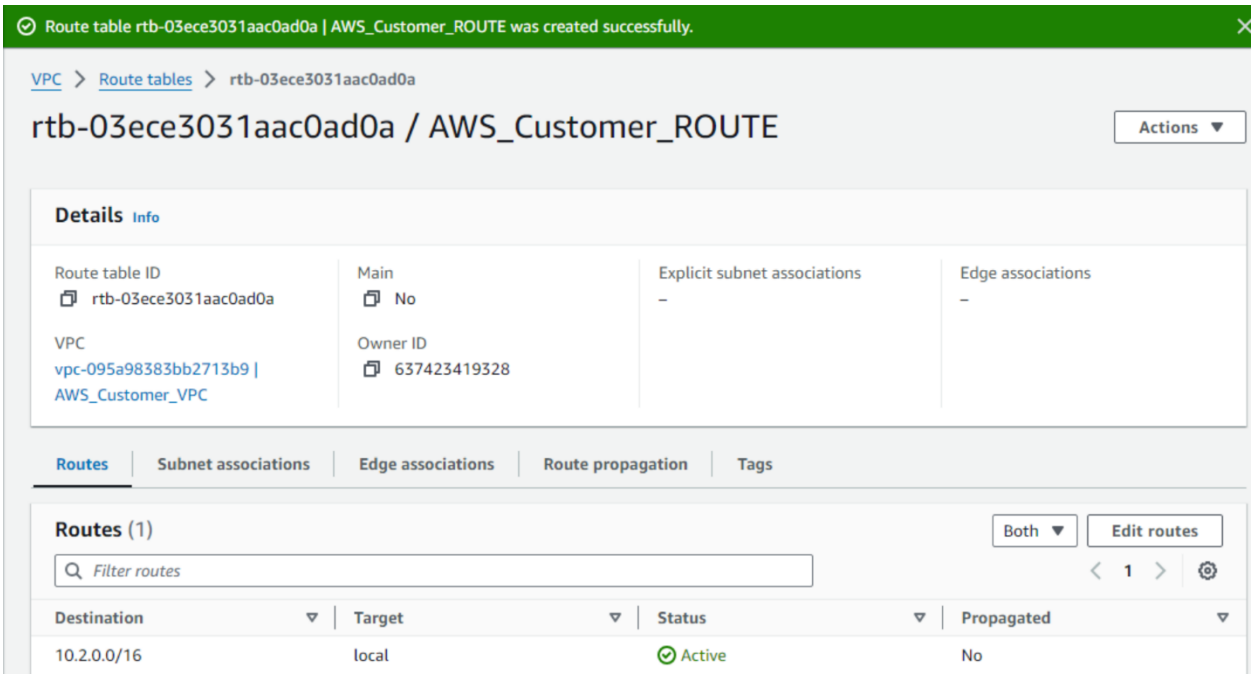

aws
Services
Search
[Alt+S]
N. Virginia
Raphael_SAA @ aws-raphael-saa
VPC > Internet gateways > Create internet gateway
Create internet gateway Info
An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.
Internet gateway settings
Name tag
Creates a tag with a key of 'Name' and a value that you specify.
AWS_Customer_IGW
Tags - optional
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.
Key
Value - optional
Name
AWS_Customer_IGW
Remove
Add new tag
You can add 49 more tags.
Cancel
Create internet gateway

aws
Services
Search
[Alt+S]
N. Virginia
Raphael_SAA @ aws-raphael-saa
The following internet gateway was created: igw-0536c110618a42025 - AWS_Customer_IGW. You can now attach to a VPC to enable the VPC to communicate with the internet.
Attach to a VPC
VPC > Internet gateways > Attach to VPC (igw-0536c110618a42025)
Attach to VPC (igw-0536c110618a42025) Info
VPC
Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.
Available VPCs
Attach the internet gateway to this VPC.
vpc-095a98383bb2713b9
vpc-095a98383bb2713b9 - AWS_Customer_VPC
Cancel
Attach internet gateway



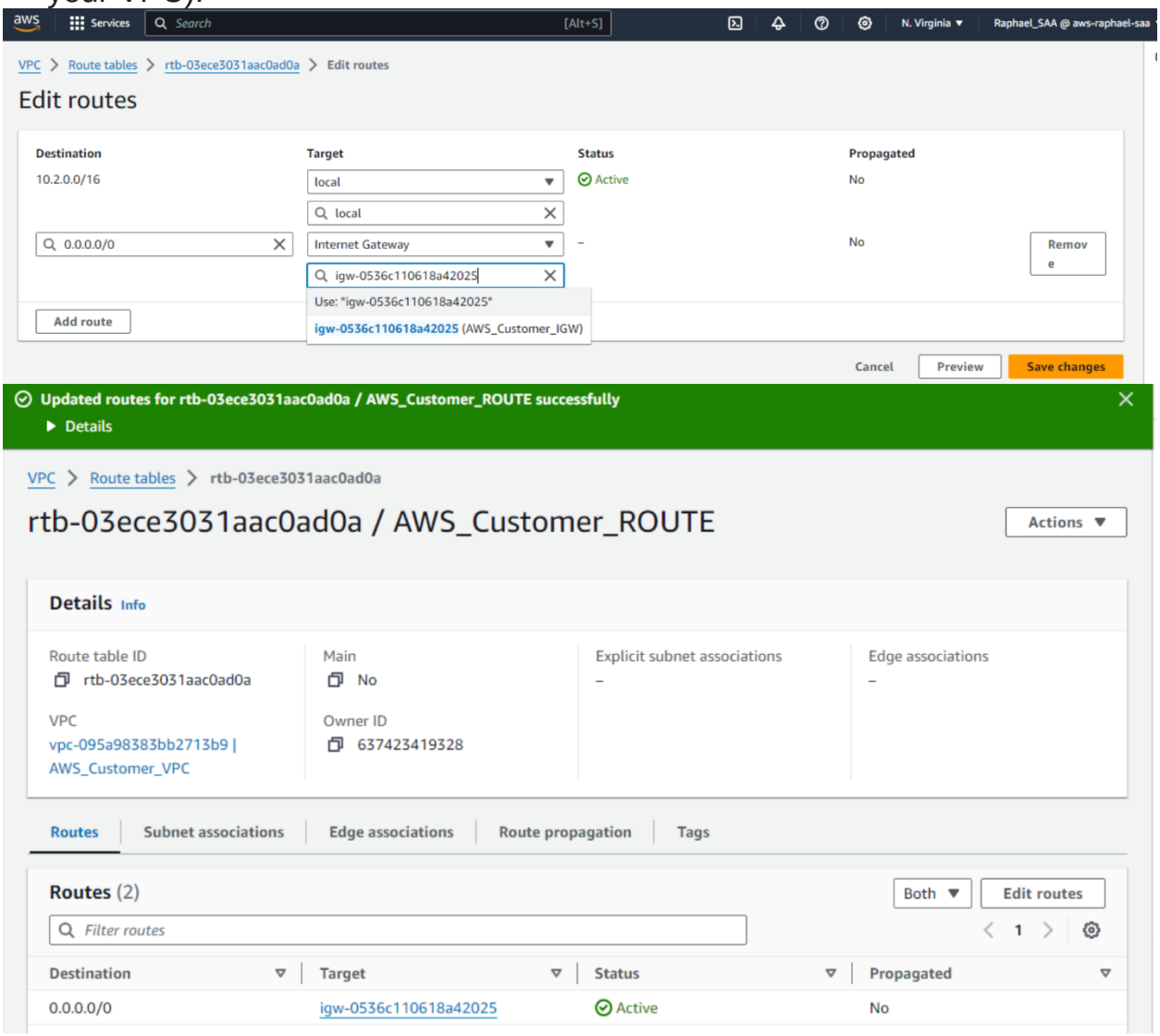
To manage routing within our VPC '**AWS_Customer_VPC**' in the N.Virginia region, let's create and configure a route table:

1. Navigate to the **Route Tables** section in the VPC dashboard.
2. Click on '**Create Route Table**'.
3. Enter the following details:
   - Name: **AWS_Customer_ROUTE**
   - VPC: **AWS_Customer_VPC**
4. Click on '**Create Route Table**'.



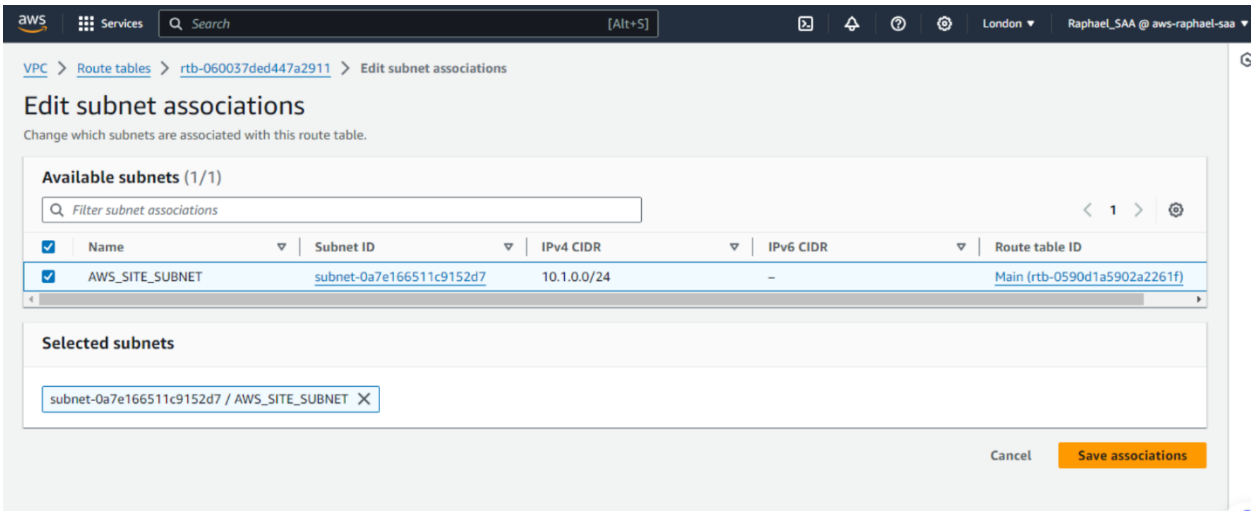
Now, let's edit the routes in the route table to direct traffic:

1. Select the route table '**AWS_Customer_ROUTE**'.
2. Click on '**Routes**' tab.
3. Click on '**Edit routes**'.
4. Add a new route with the following details:
   - Destination: **0.0.0.0/0**
   - Target: I**nternet Gateway** (select the Internet Gateway associated with your VPC).



Next, associate the route table with a subnet and ensure proper association:

1. Click on '**Subnet Associations**' tab.
2. Click on '**Edit Subnet Associations**'.
3. Select the subnet(s) within '**AWS_Customer_VPC**' that you want to associate with this route table.
4. Click on 'Save'.



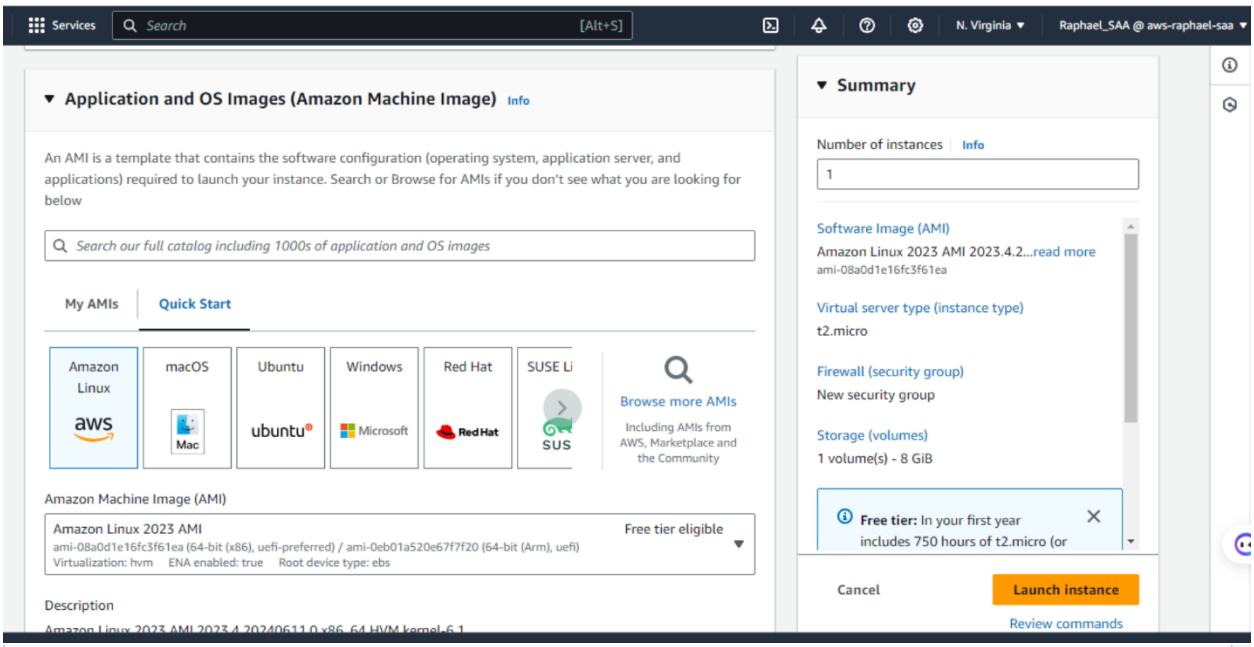
Step 3: Launch Amazon Linux Machine in N. Virginia Region

Deploy Amazon Linux instance in N.Virginia region and configure security groups for SSH, TCP, and ICMP access.

To deploy an EC2 instance in our VPC ‘AWS_Customer_VPC‘ with Amazon Linux as the machine image and t2.micro as the instance type, follow these steps:

1. Navigate to the EC2 dashboard and click on ‘Launch Instance’.
2. Choose ‘Amazon Linux‘ as the machine image and ‘t2.micro‘ as the instance type.
3. Select ‘AWS_Customer_VPC‘ as the VPC for the instance.
4. Enable ‘Auto-assign Public IP‘ for the instance.
5. Create a key pair for SSH access.
6. Create a security group named ‘customer-securitygrp‘ with the description ‘customer-securitygrp‘.
7. Configure the security group with the following rules:
    - SSH (port 22) from anywhere
    - All TCP from anywhere
    - All ICMP (IPv4) from anywhere

Proceed with launching the EC2 instance to set up the desired configuration for your environment.



Instance type Info | Get advice

Instance type

t2.micro — Free tier eligible

Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand RHEL base pricing: 0.0716 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

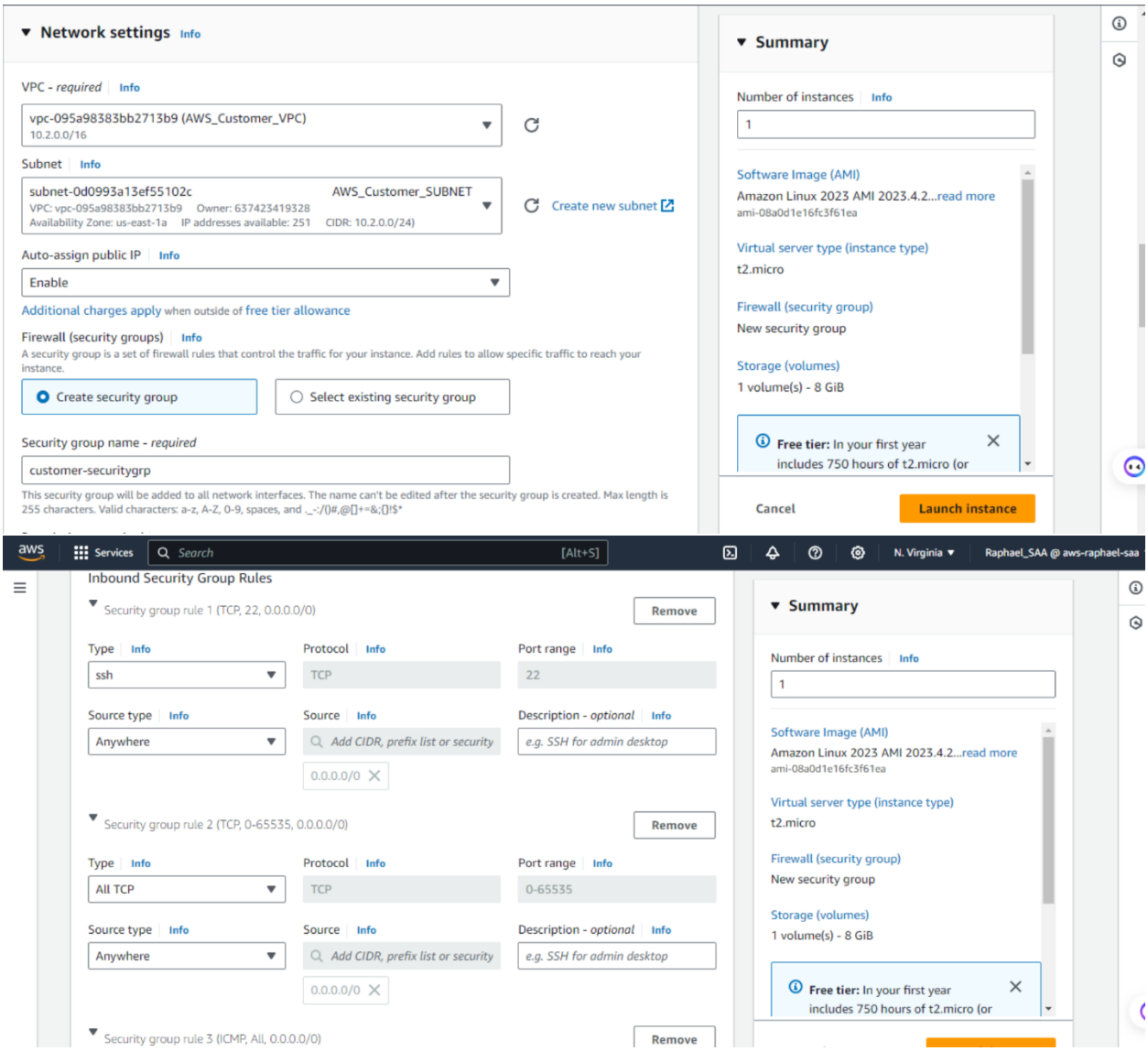
Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

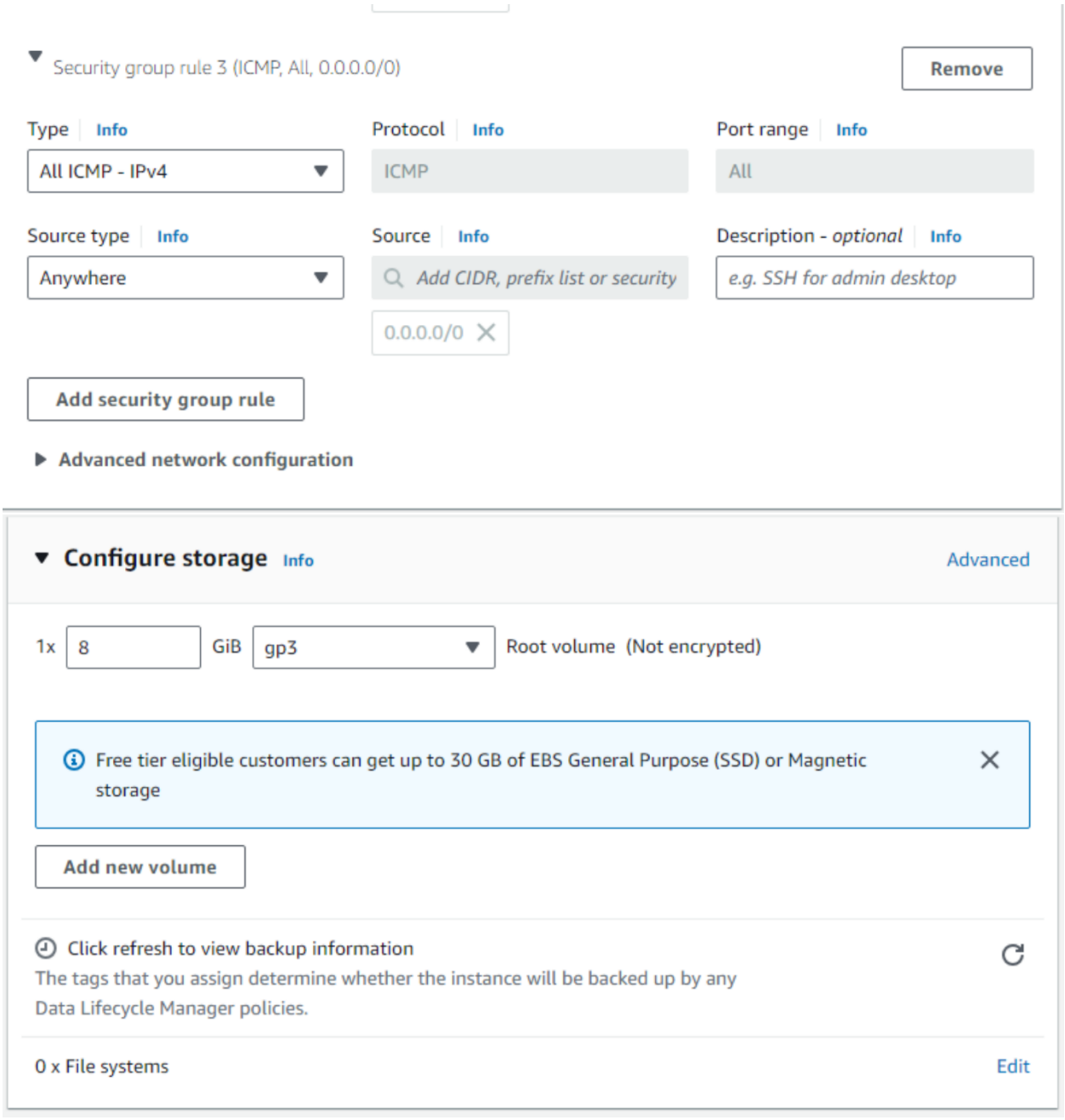
Key pair name - *required*

bit_node

Create new key pair


Network settings Info
VPC - required Info
vpc-095a98383bb2713b9 (AWS_Customer_VPC)
10.2.0.0/16
Subnet Info
subnet-0d0993a13ef55102c AWS_Customer_SUBNET
VPC: vpc-095a98383bb2713b9 Owner: 637423419328
Availability Zone: us-east-1a IP addresses available: 251 CIDR: 10.2.0.0/24)
Create new subnet
Auto-assign public IP Info
Enable
Additional charges apply when outside of free tier allowance
Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
Create security group
Select existing security group
Security group name - required
customer-securitygrp
This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=&;{}!$*
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.4.2...read more
ami-08a0d1e16fc3f61ea
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or
Cancel
Launch instance

aws
Services
Search
[Alt+S]
N. Virginia
Raphael_SAA @ aws-raphael-saa
Inbound Security Group Rules
Security group rule 1 (TCP, 22, 0.0.0.0/0)
Remove
Type Info
ssh
Protocol Info
TCP
Port range Info
22
Source type Info
Anywhere
Source Info
Add CIDR, prefix list or security
Description - optional Info
e.g. SSH for admin desktop
0.0.0.0/0
Security group rule 2 (TCP, 0-65535, 0.0.0.0/0)
Remove
Type Info
All TCP
Protocol Info
TCP
Port range Info
0-65535
Source type Info
Anywhere
Source Info
Add CIDR, prefix list or security
Description - optional Info
e.g. SSH for admin desktop
0.0.0.0/0
Security group rule 3 (ICMP, All, 0.0.0.0/0)
Remove
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.4.2...read more
ami-08a0d1e16fc3f61ea
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or



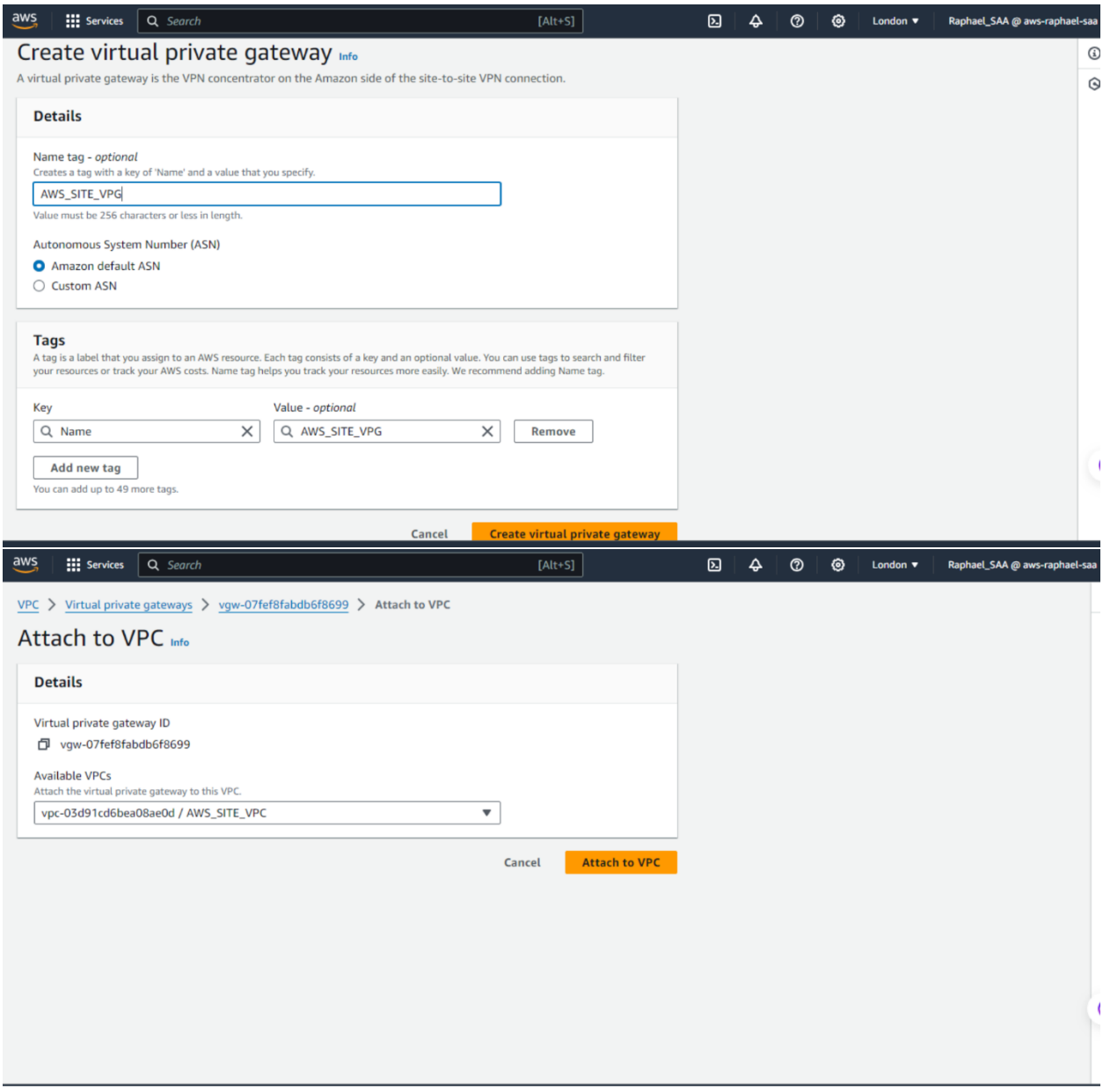
Step 5: Set Up Virtual Private Gateway(VPG) in London Region

Establish a VPG in the Mumbai region to connect to the AWS VPN network.

To establish a connection between our VPC 'AWS_SITE_VPC' and the AWS VPN network, follow these steps:

1. Click on 'Create Virtual Private Gateway (VPG)'.
2. Give the VPG the name 'AWS_SITE_VPG'.

3. Click on ‘Create’.
4. Next, attach the VPG to the VPC ‘AWS_SITE_VPC‘:
    - Select the VPG ‘AWS_SITE_VPG‘.
    - Click on ‘Actions’ and choose ‘Attach to VPC’.
    - Select ‘AWS_SITE_VPC‘ as the VPC to attach the VPG to.
5. Confirm the attachment.

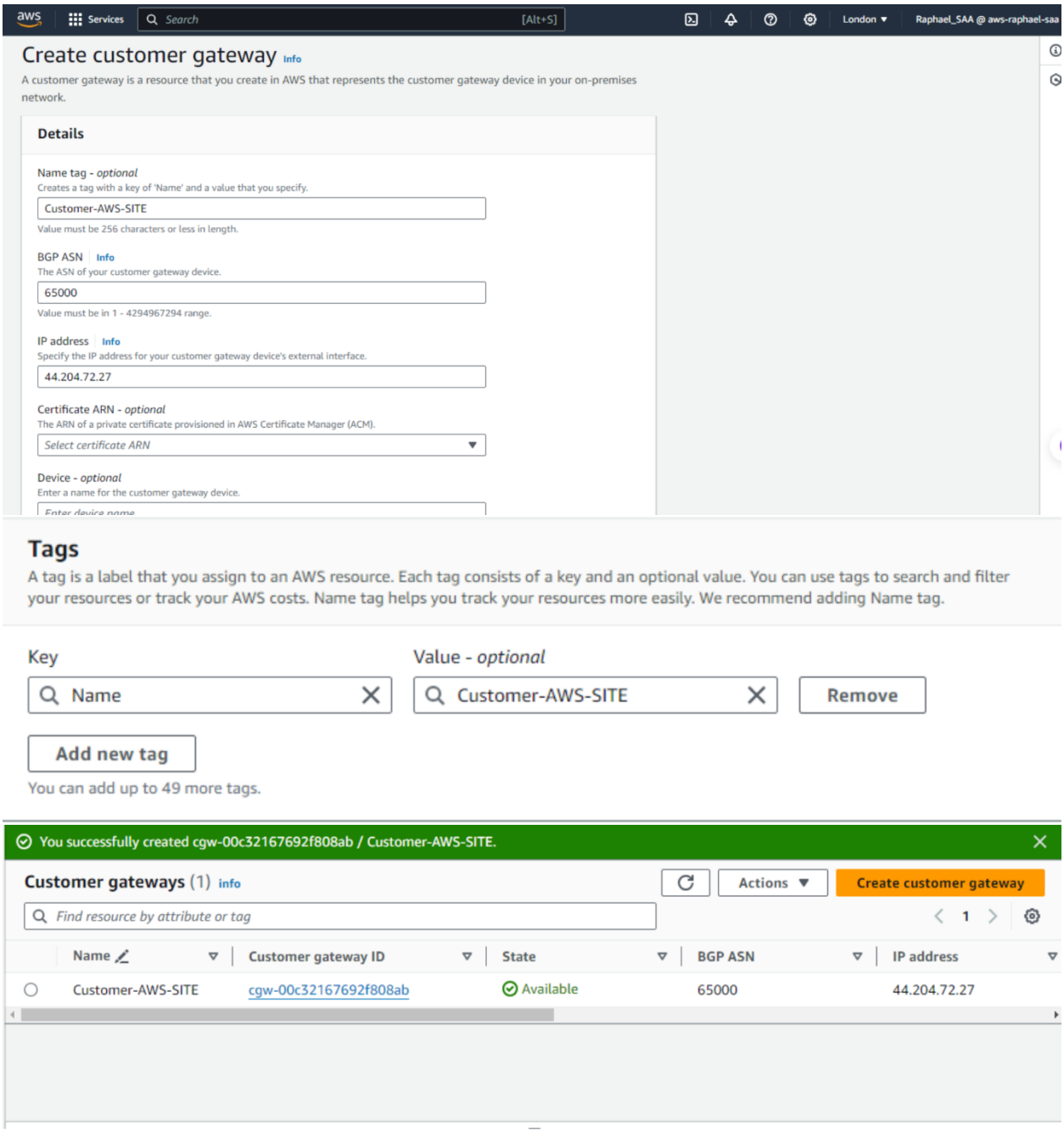


Step 6: Create Customer Gateway (CGW) in London Region

Configure a CGW in the Mumbai region to represent your on-premises network.

To establish connectivity between our network and AWS, we create a customer gateway ‘Customer-AWS-SITE’ with static routing:

1. Navigate to the VPN Connections section in the AWS Console.
2. Click on ‘Create Customer Gateway’.
3. Enter the following details:
    - Name: Customer-AWS-SITE
    - Routing: Static (select this option)
    - IP Address: [Specify the IP address of the instance launched in the N. Virginia region]
4. Keep other settings as default.
5. Click on ‘Create Customer Gateway’.



Step 6: Enable Route Propagation in the London Region

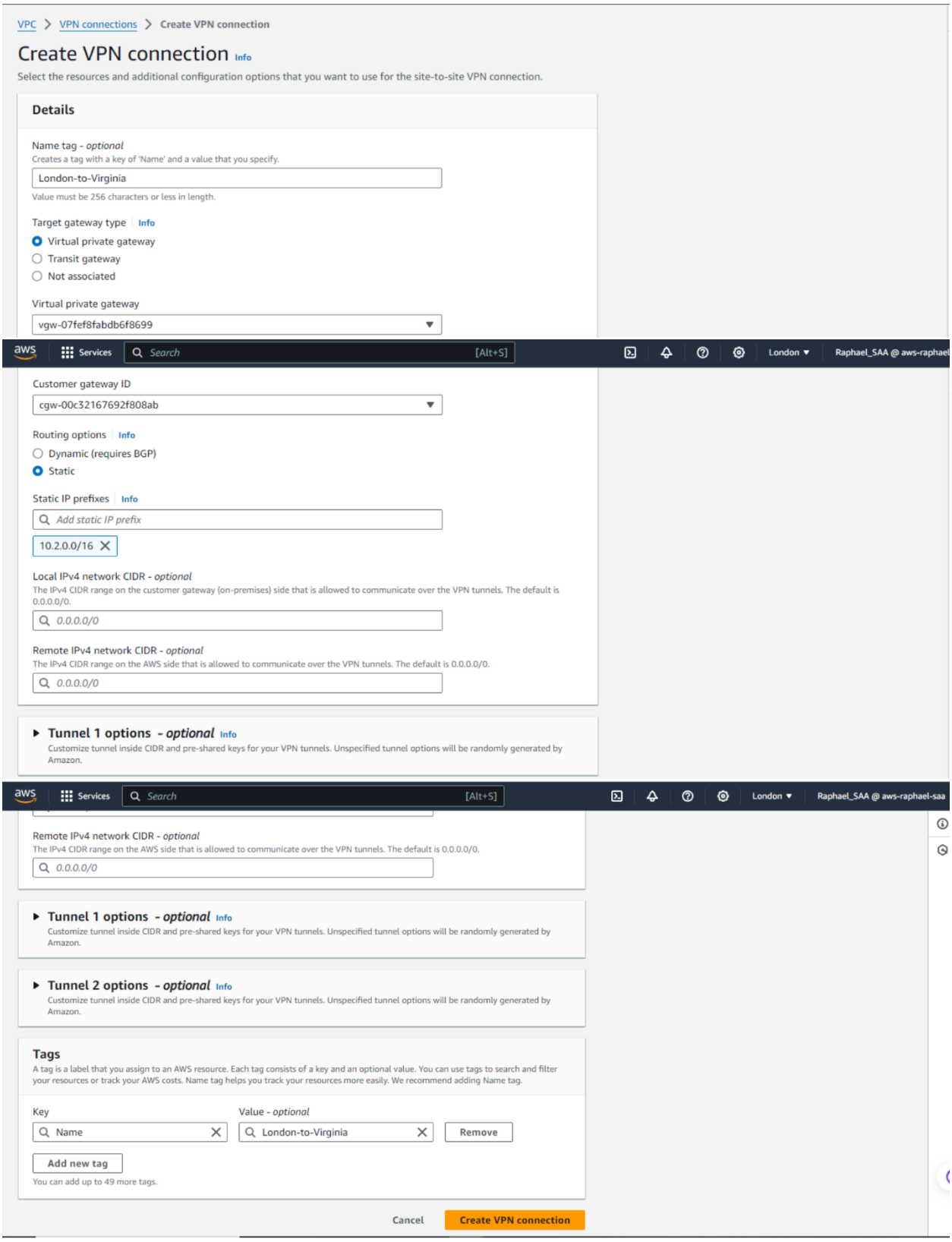
Allow routing between VPCs and on-premises networks by enabling route propagation in Mumbai regions.

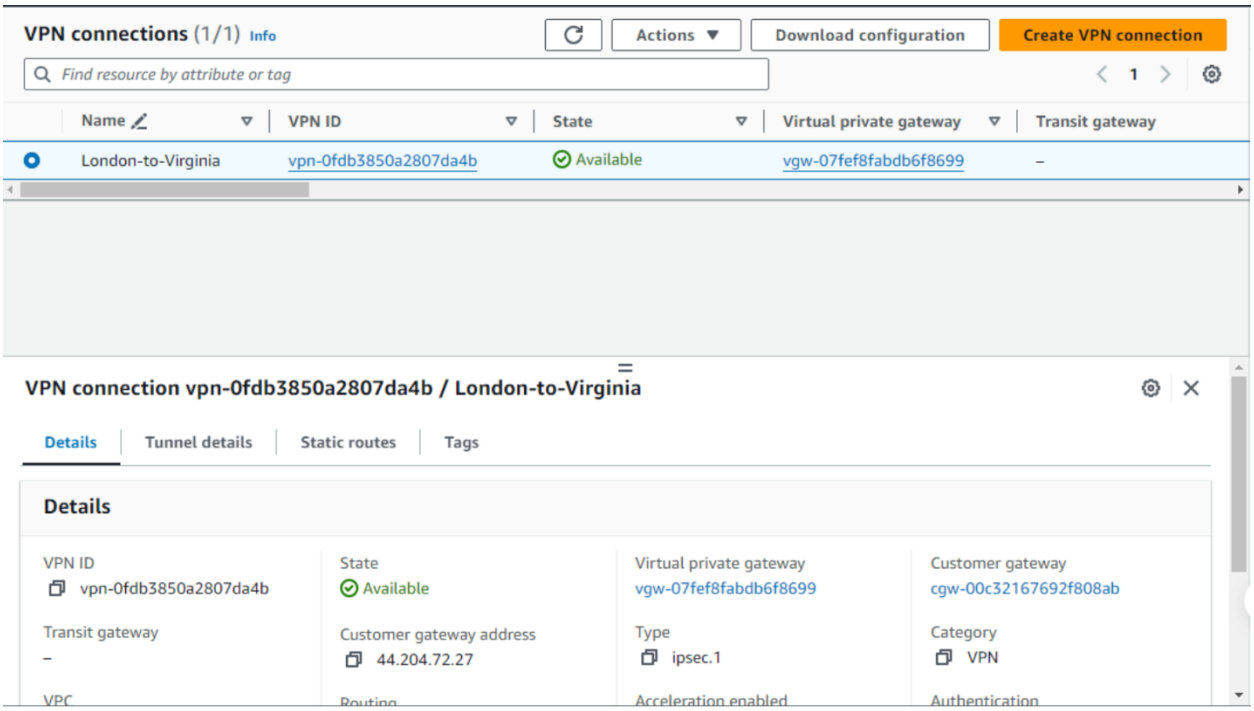
Create a Site-to-Site VPN connection in the London region for secure connectivity between VPCs.

To establish a secure connection between our London and N. Virginia regions, we create a VPN connection named ‘London-to-Virginia’:

1. Navigate to the VPN Connections section in the AWS Console.
2. Click on ‘Create VPN Connection’.
3. Enter the following details:
    - Name: London-to-Virginia
    - Target: Virtual Private Gateway (select the appropriate VPG)
    - Customer Gateway: [Select the previously configured Customer Gateway]
    - Routing Options: Static (select this option)
    - Static IP Prefix: 10.2.0.0/16
4. Click on ‘Create VPN Connection’.

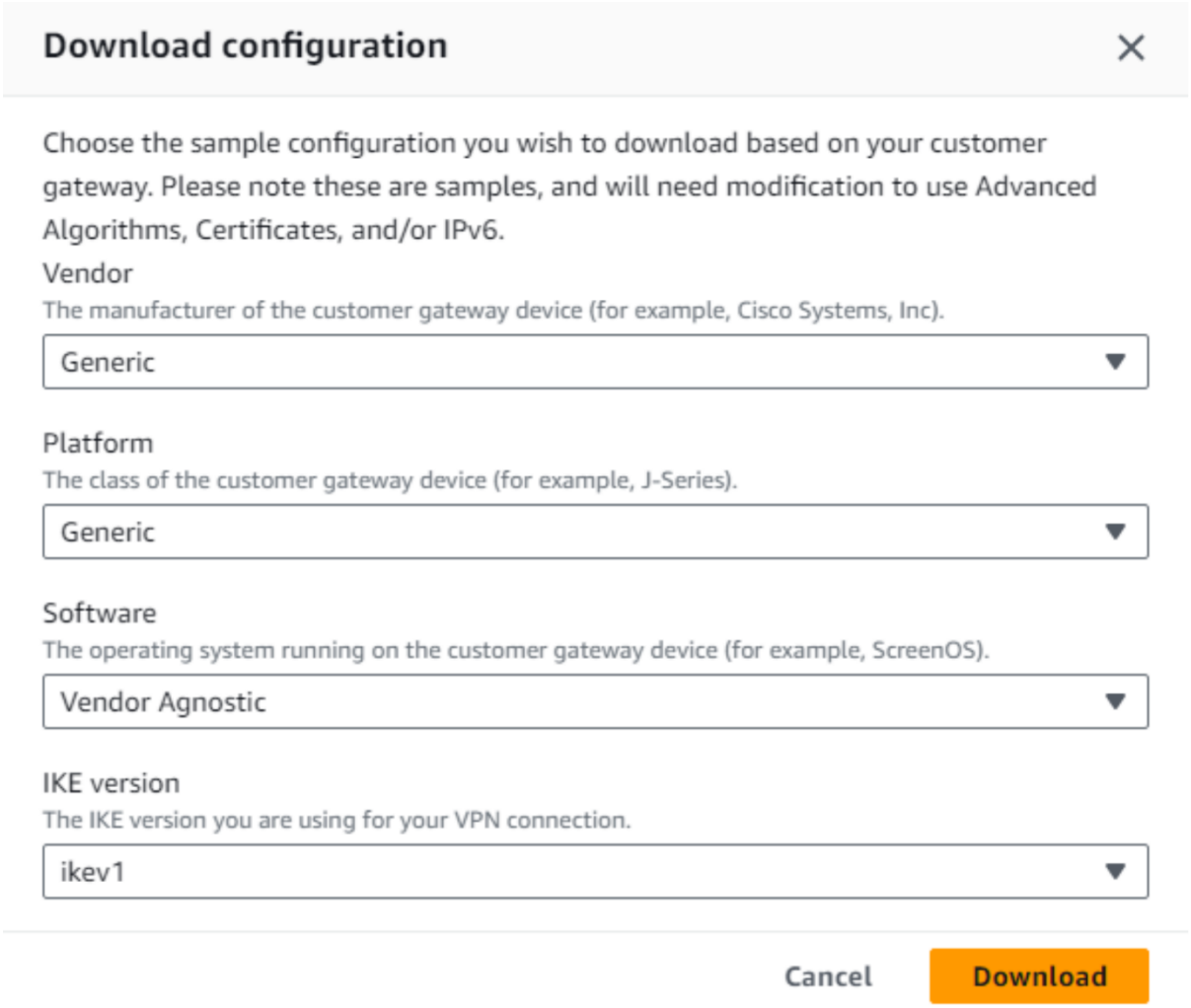
    This setup establishes a VPN connection using static routing, ensuring secure communication between our London and N. Virginia regions.”


VPC > VPN connections > Create VPN connection
Create VPN connection Info
Select the resources and additional configuration options that you want to use for the site-to-site VPN connection.
Details
Name tag - optional
Creates a tag with a key of 'Name' and a value that you specify.
London-to-Virginia
Value must be 256 characters or less in length.
Target gateway type Info
Virtual private gateway
Transit gateway
Not associated
Virtual private gateway
vgw-07fef8fabdb6f8699
aws
Services
Search
[Alt+S]
London
Raphael_SAA @ aws-raphael
Customer gateway ID
cgw-00c32167692f808ab
Routing options Info
Dynamic (requires BGP)
Static
Static IP prefixes Info
Add static IP prefix
10.2.0.0/16
Local IPv4 network CIDR - optional
The IPv4 CIDR range on the customer gateway (on-premises) side that is allowed to communicate over the VPN tunnels. The default is 0.0.0.0/0.
0.0.0.0/0
Remote IPv4 network CIDR - optional
The IPv4 CIDR range on the AWS side that is allowed to communicate over the VPN tunnels. The default is 0.0.0.0/0.
0.0.0.0/0
Tunnel 1 options - optional Info
Customize tunnel inside CIDR and pre-shared keys for your VPN tunnels. Unspecified tunnel options will be randomly generated by Amazon.
Tunnel 2 options - optional Info
Customize tunnel inside CIDR and pre-shared keys for your VPN tunnels. Unspecified tunnel options will be randomly generated by Amazon.
Tags
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs. Name tag helps you track your resources more easily. We recommend adding Name tag.
Key
Value - optional
Name
London-to-Virginia
Remove
Add new tag
You can add up to 49 more tags.
Cancel
Create VPN connection

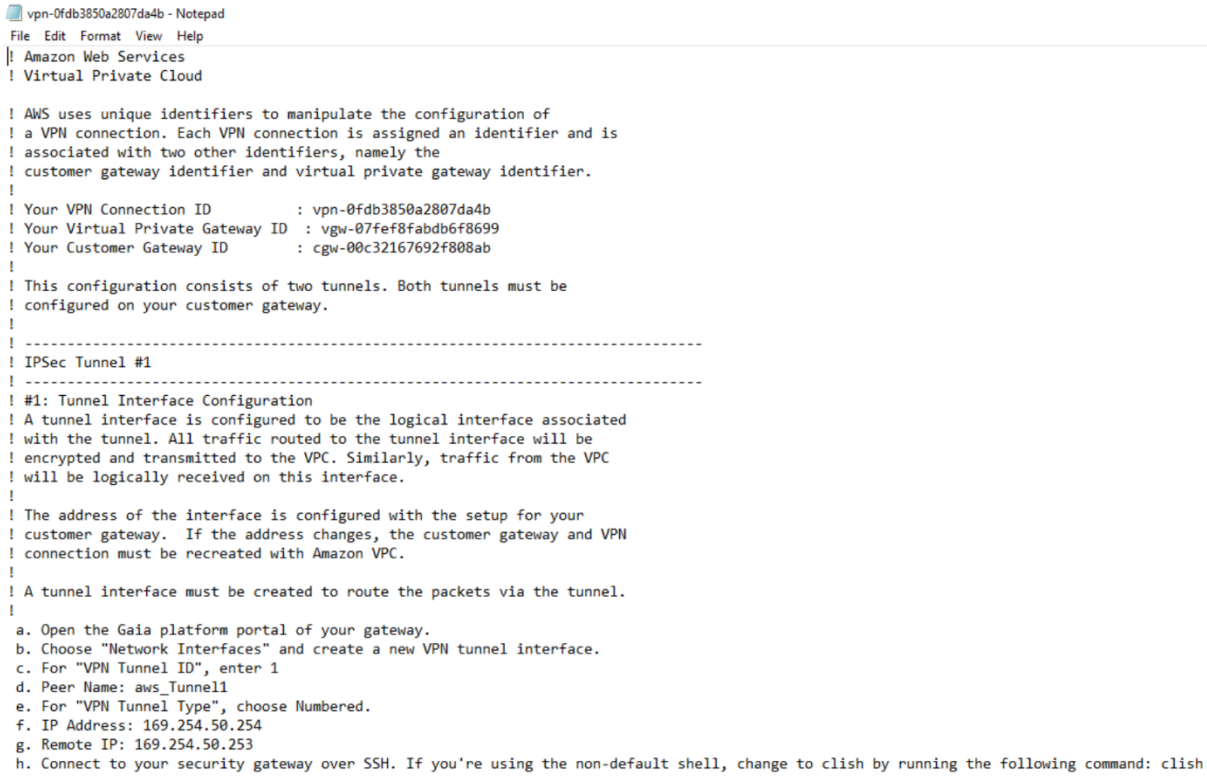


Step 8: Download VPN Configuration

Obtain the Site-to-Site VPN configuration from AWS for setting up your on-premises VPN device.



Configuration is opened on Notepad



Step 9: Access EC2 Instance launched in N. Virginia Region

Connect to the EC2 instance using either EC2 instance connect or Linux terminal using the downloaded Key pair.

What is libreswan?

Libreswan is an open-source implementation of the IPsec (Internet Protocol Security) protocol suite for Linux-based systems. It provides secure communication over IP networks by encrypting and authenticating data packets.

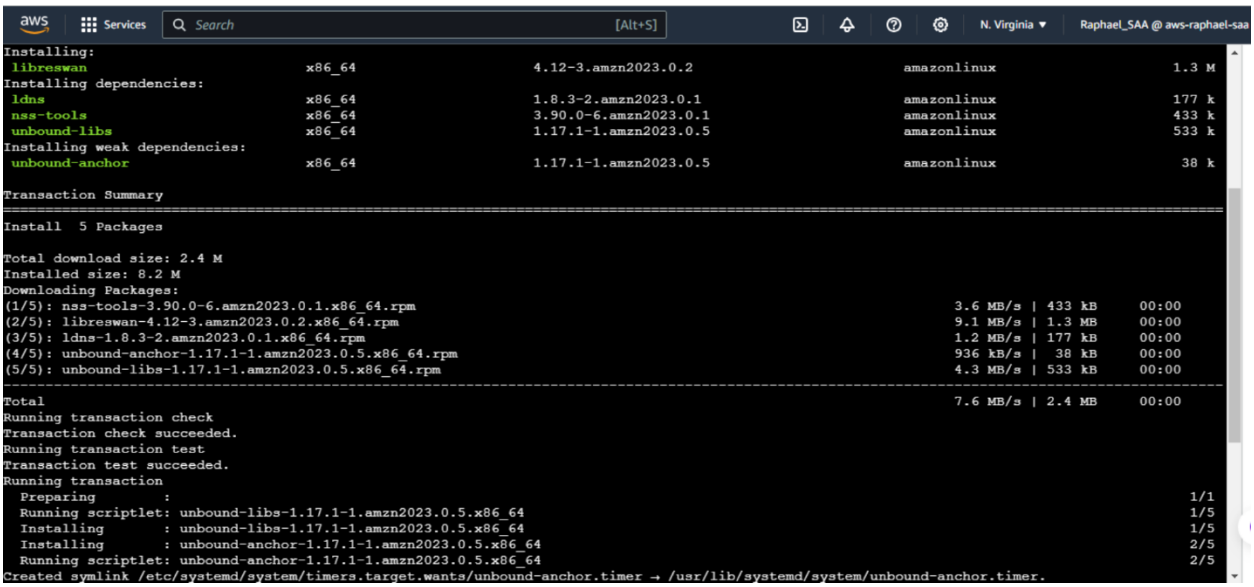
Enter the below commands After Connecting with Ec2 instance: -

- Change to root user:

  sudo su

- Install Libreswan:

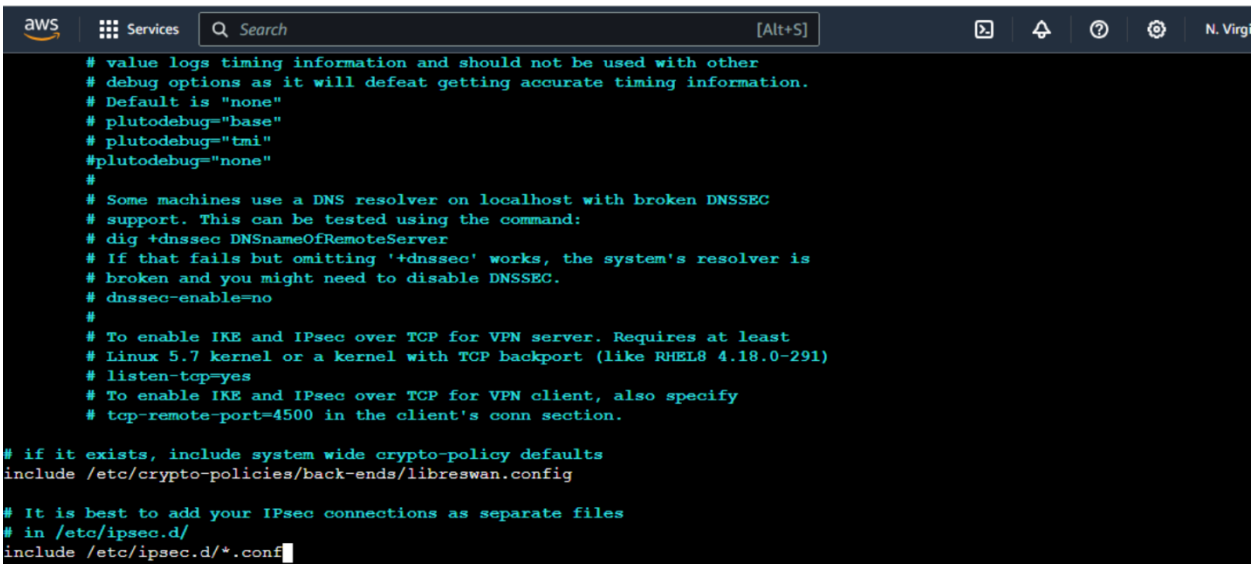
  yum install libreswan -y



- Modify /etc/ipsec.conf:

Ensure the following line is uncommented or add it if not present:

vi /etc/ipsec.conf



- Update /etc/sysctl.conf:

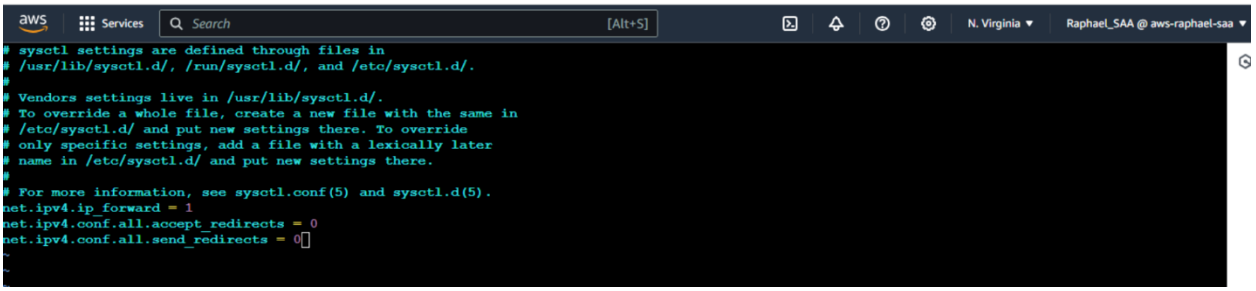
  Edit the /etc/sysctl.conf file and add or modify the following lines:

  vi /etc/sysctl.conf

  net.ipv4.ip_forward = 1

  net.ipv4.conf.all.accept_redirects = 0

  net.ipv4.conf.all.send_redirects = 0



- Restart network service:

  systemctl restart systemd-networkd

- Create/Edit /etc/ipsec.d/aws-vpn.conf:

  vi /etc/ipsec.d/aws-vpn.conf

  conn Tunnel1

  authby=secret

  auto=start

  left=%defaultroute # Assuming this is your local subnet, replace if needed

  leftid= 44.204.72.27

  right= 18.135.148.5

  type=tunnel

  ikelifetime=8h

  keylife=1h

  # Consider stronger options if supported by both sides:

  # phase2alg=aes256-gcm-sha256

  # ike=aes256-gcm-sha256

  # keyingtries=%forever

  keyexchange=ike

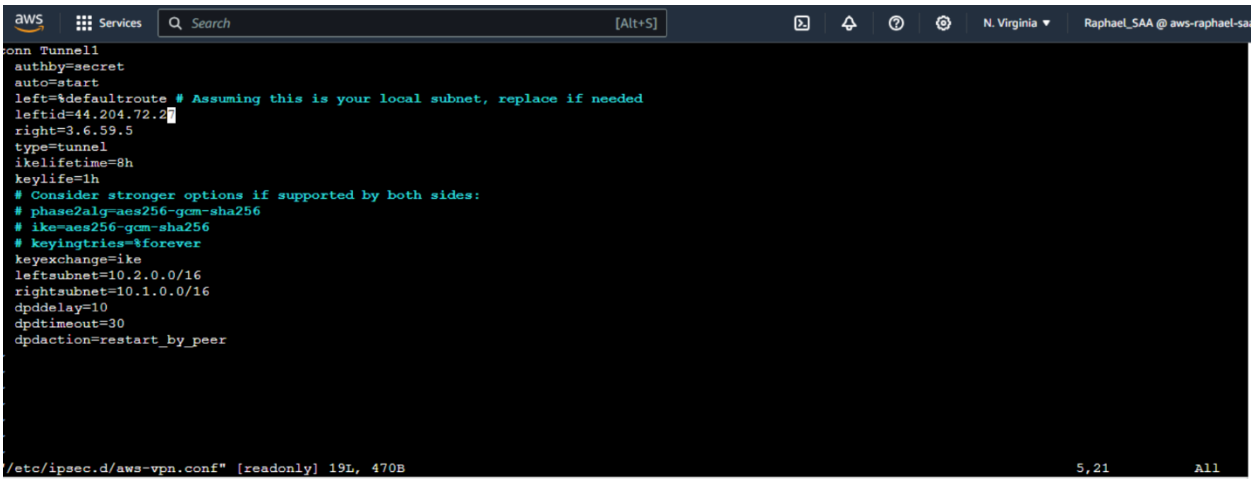
  leftsubnet=10.2.0.0/16

  rightsubnet=10.1.0.0/16

  dpddelay=10

  dpdtimeout=30

  dpdaction=restart_by_peer



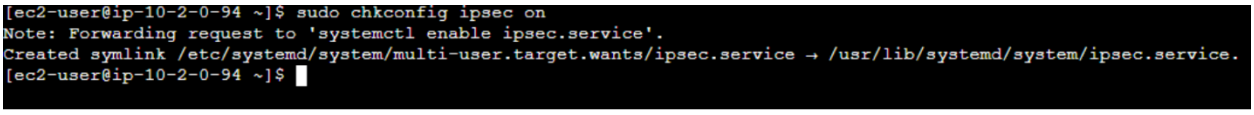
- Create/Edit /etc/ipsec.d/aws-vpn.secrets:

  vi /etc/ipsec.d/aws-vpn.secrets

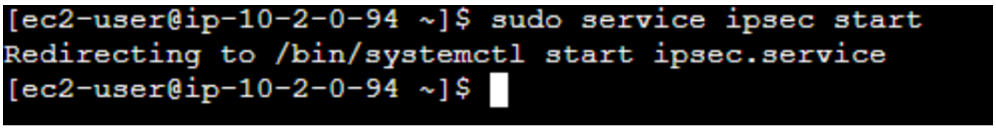
  customer_public_ip aws_vgw_public_ip : PSK "shared secret"

  44.204.72.27 18.135.148.5: PSK " tsEPj.onO1bRBbA2kP2QpzhZO6xdVdhv"

chkconfig ipsec on



service ipsec start



service ipsec status