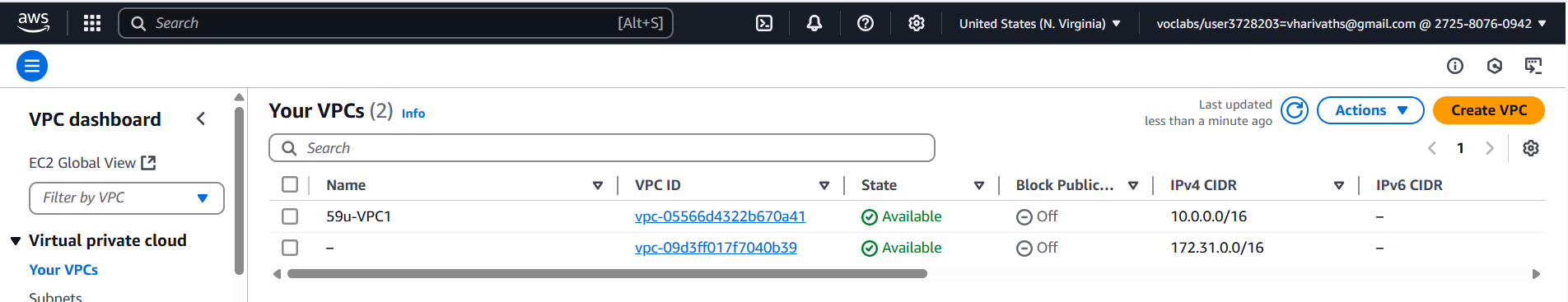
**Step 1: Select N. Virginia Region**

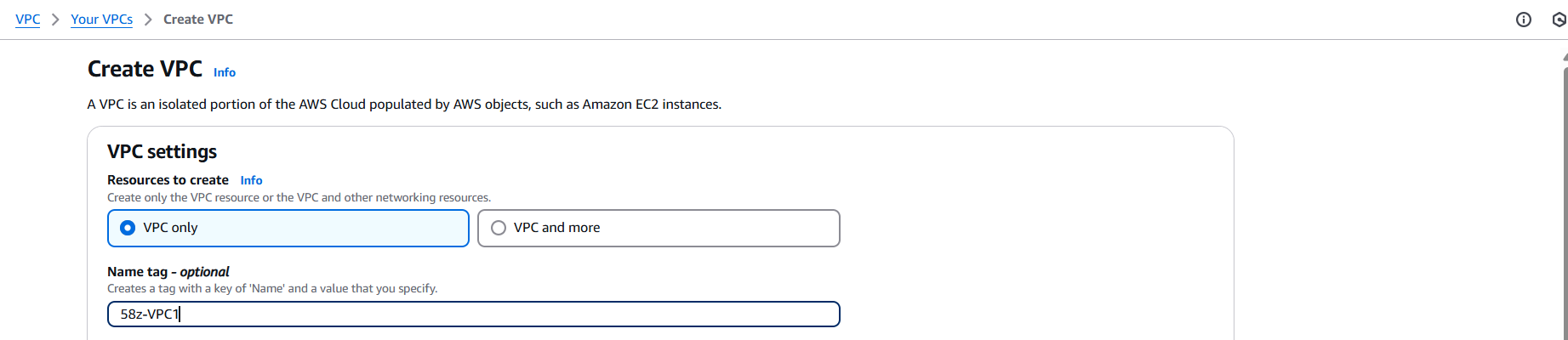
1. Go to the [AWS Management Console](https://aws.amazon.com/console).
2. From the **top right corner**, select the region: US East (N. Virginia).

**Step 2: Create VPC**

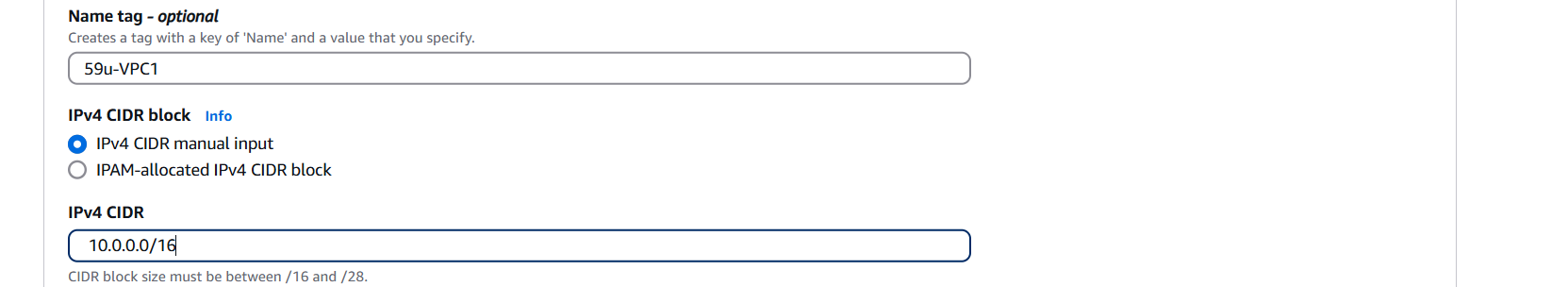
1. Go to **VPC → Your VPCs → Create VPC**.



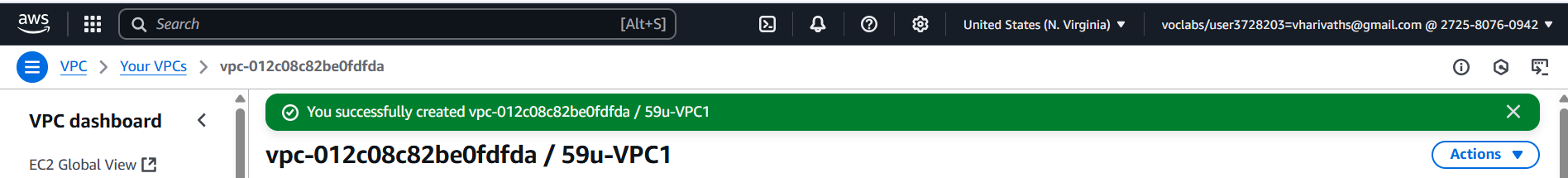
1. **VPC name:** 59u-VPC1



1. **IPv4 CIDR block:** 10.0.0.0/16

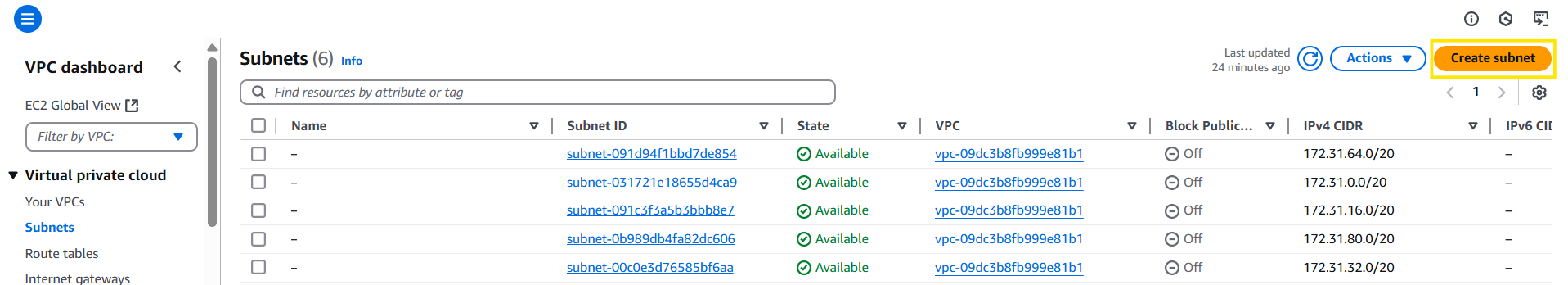


1. Leave **IPv6 CIDR block** empty.
2. **Tenancy:** Default
3. Click on **Create VPC**.

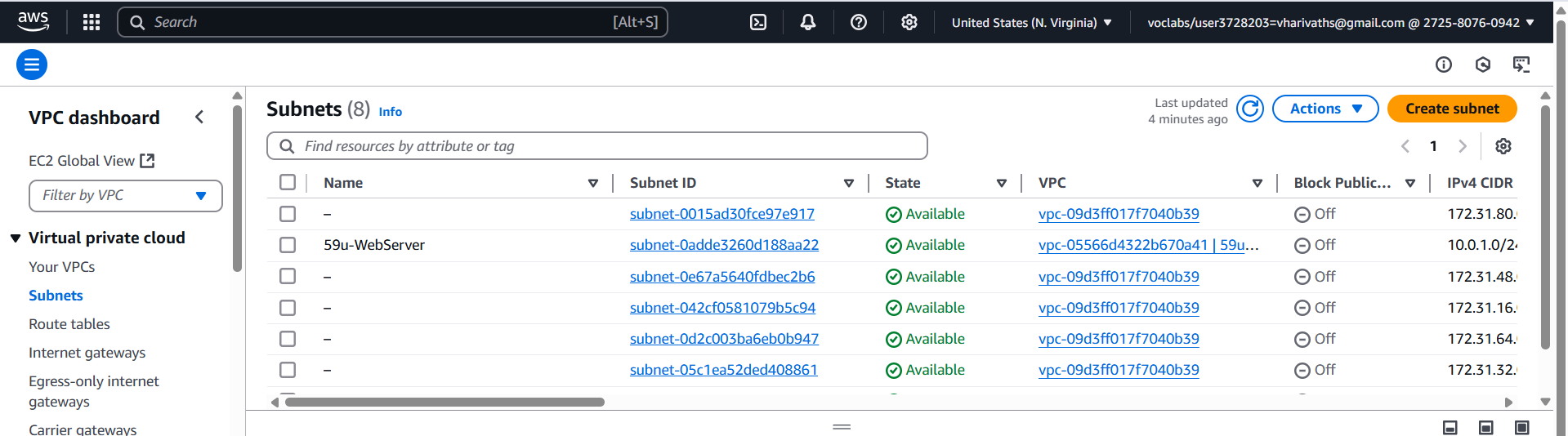


**Step 3: Create Two Subnets**

1. **Go to:** VPC → Subnets → Create Subnet



1. Select **VPC:** 59u-VPC1

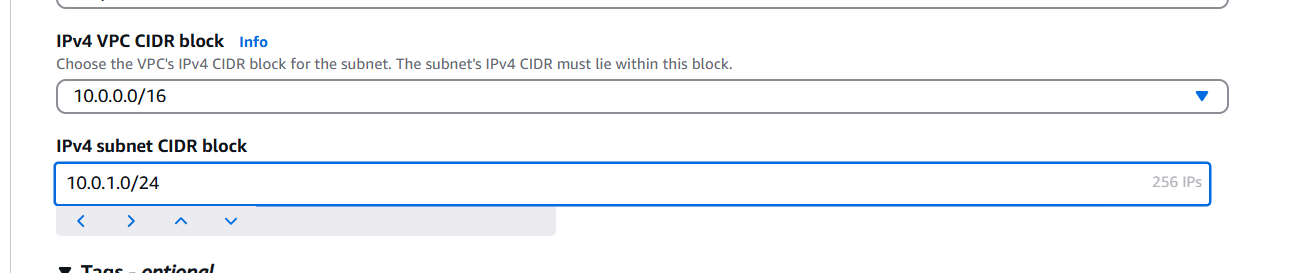


**Create Public Subnet**

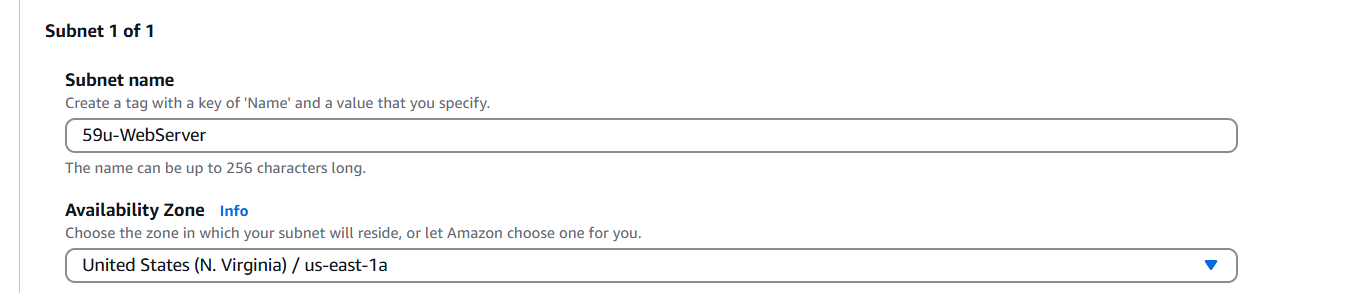
* **Subnet name:** 59u-WebServer



* **CIDR block:** 10.0.1.0/24



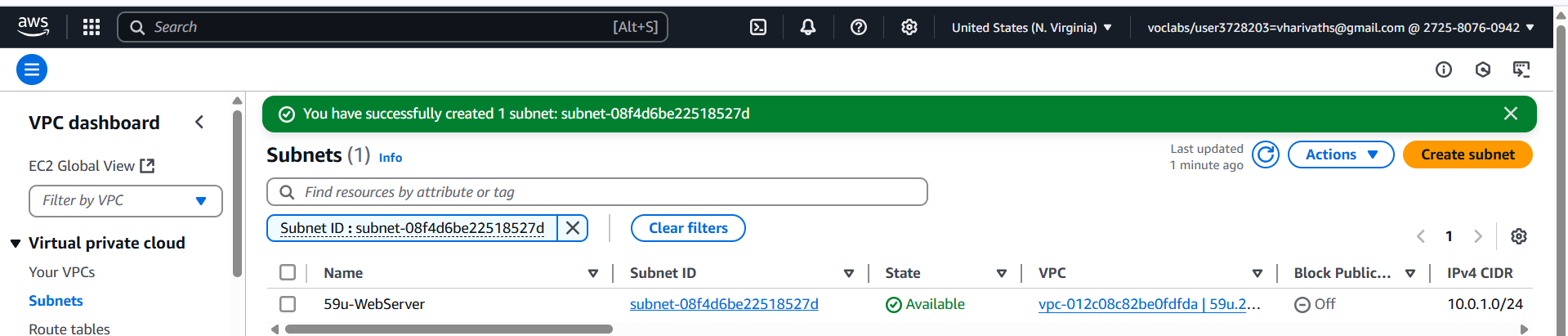
* **Availability Zone:** Select us-east-1a



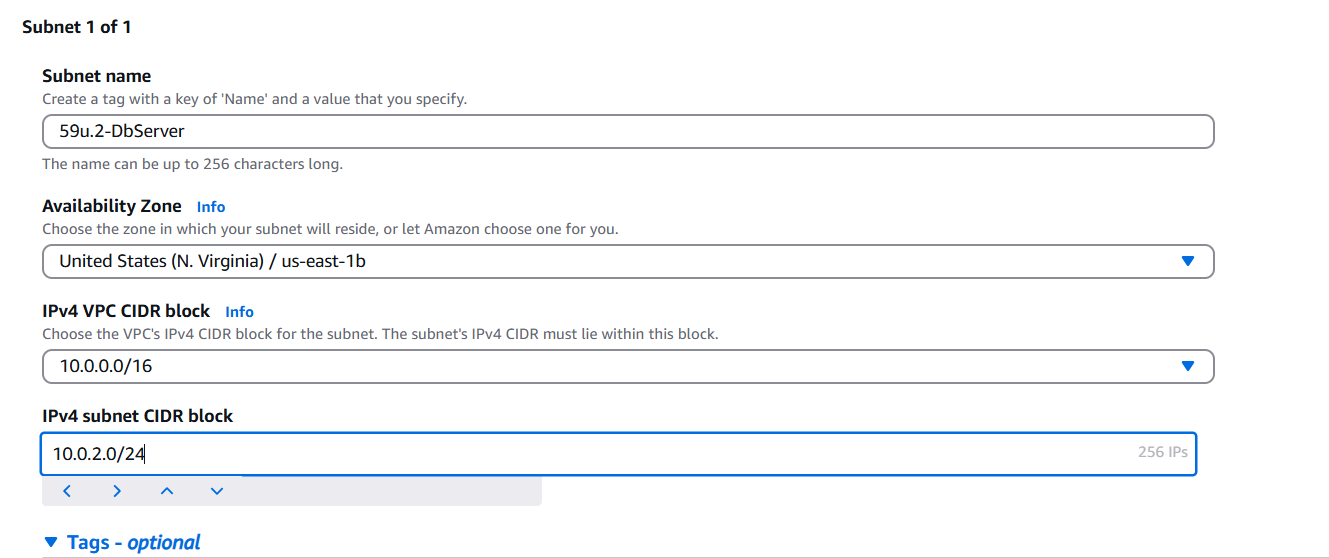
* **Enable auto-assign public IPv4 address:** ✅ Check this option

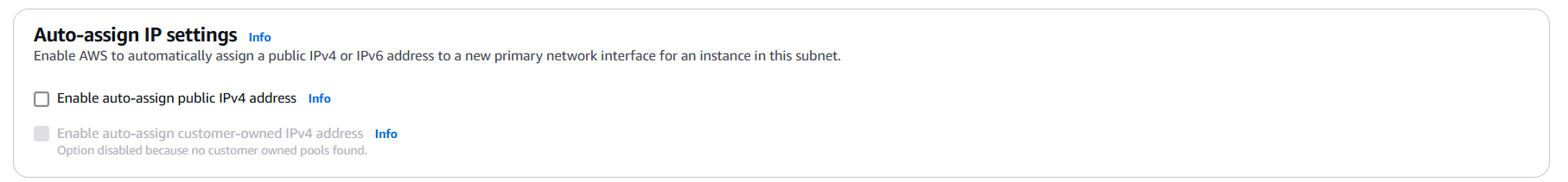


* Click on **Create Subnet**

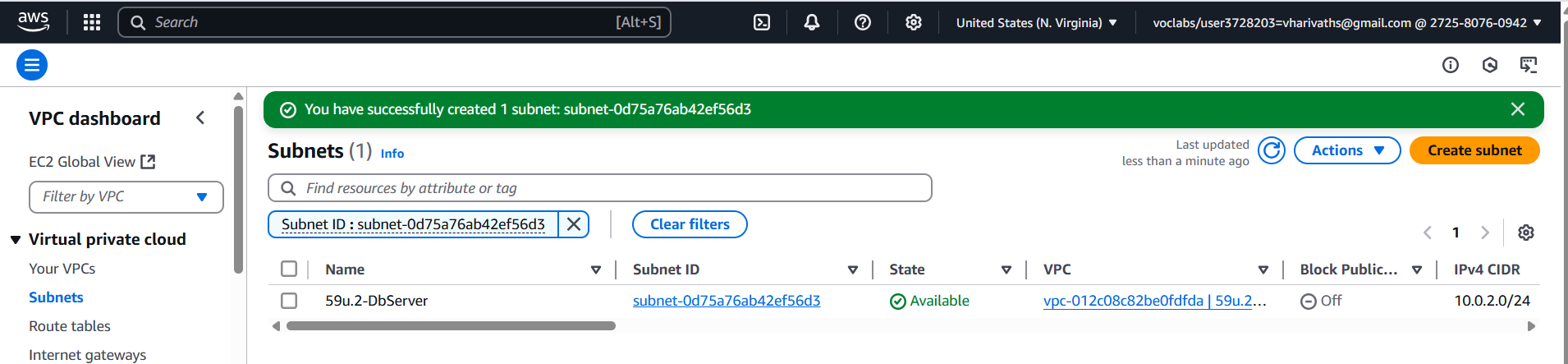


**Create Private Subnet**

* **Subnet name:** 59u-DbServer
* **CIDR block:** 10.0.2.0/24
* **Availability Zone:** Select us-east-1b
* 
* **Enable auto-assign public IPv4 address:** ❌ Unchecked (private subnet)

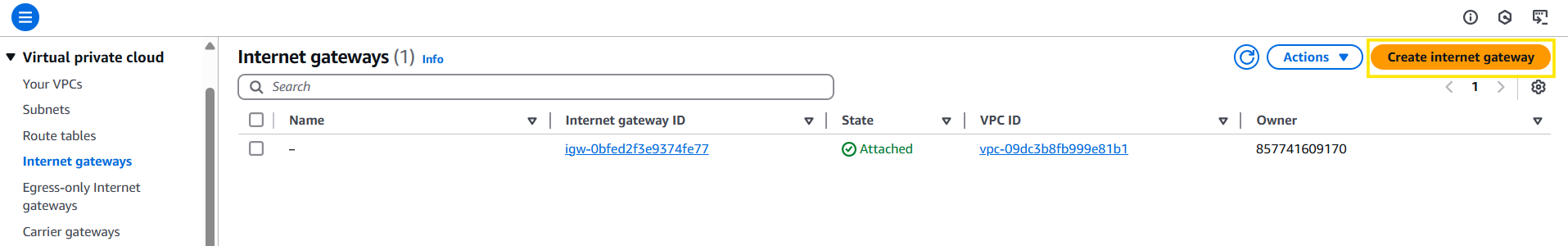


* Click on **Create Subnet**

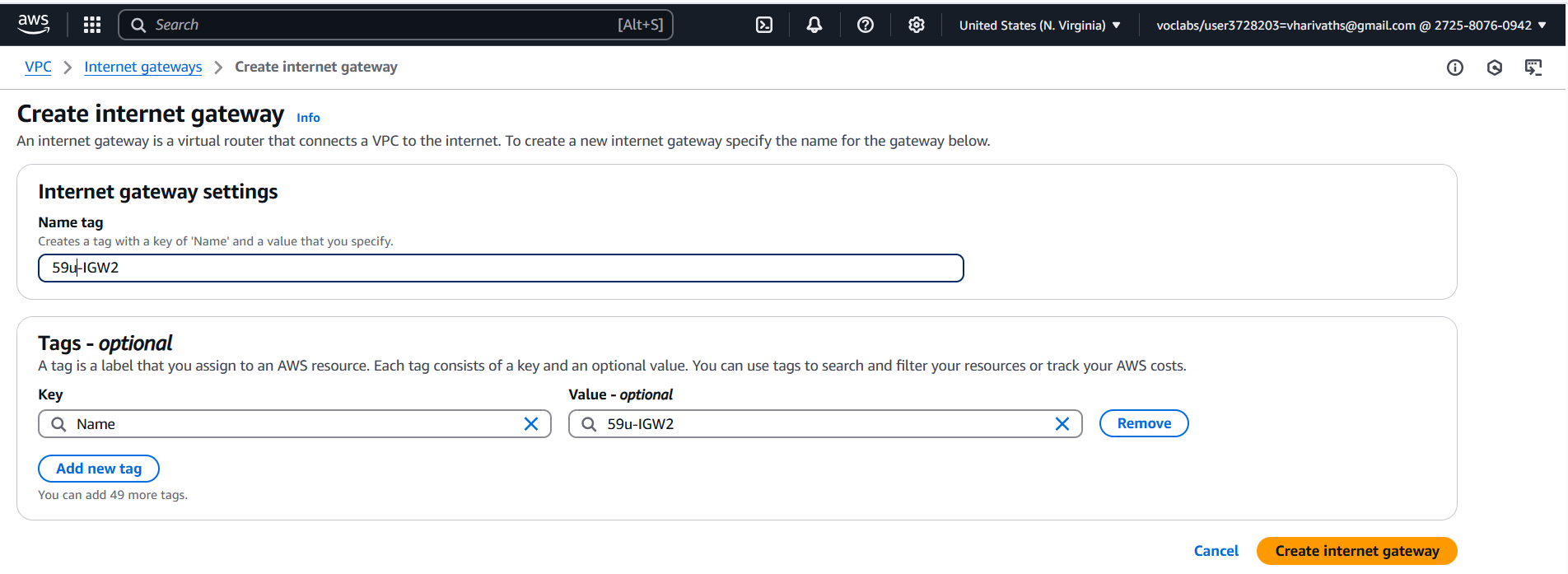


**Step 4: Create and Attach Internet Gateway**

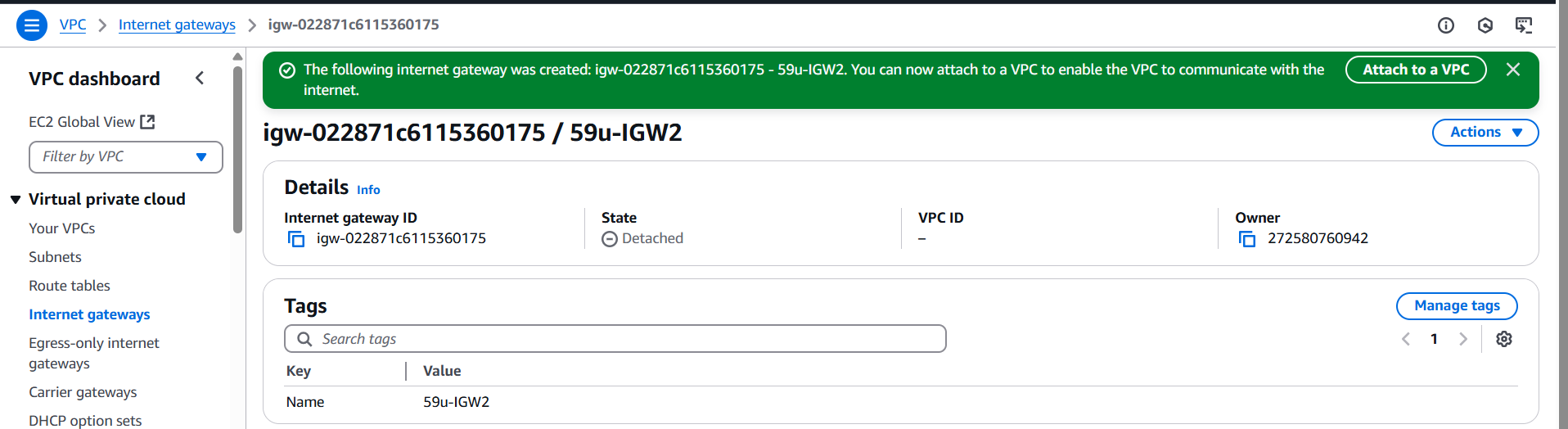
1. **Go to:** VPC → Internet Gateways → Create Internet Gateway



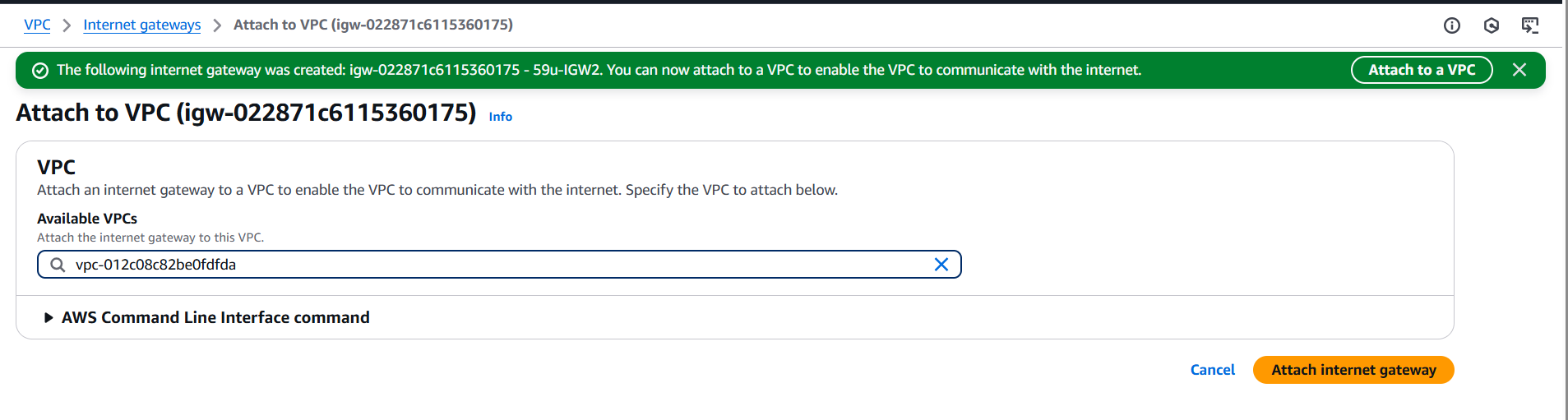
1. **Name:** 59u-IGW2



1. Click on **Create Internet Gateway**

****

1. **Attach to VPC:** Select 59u-VPC1 → Attach.

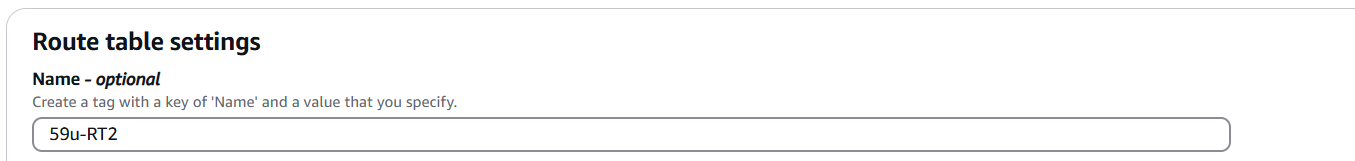


**Step 5: Create Route Table**

1. **Go to:** VPC → Route Tables → Create Route Table



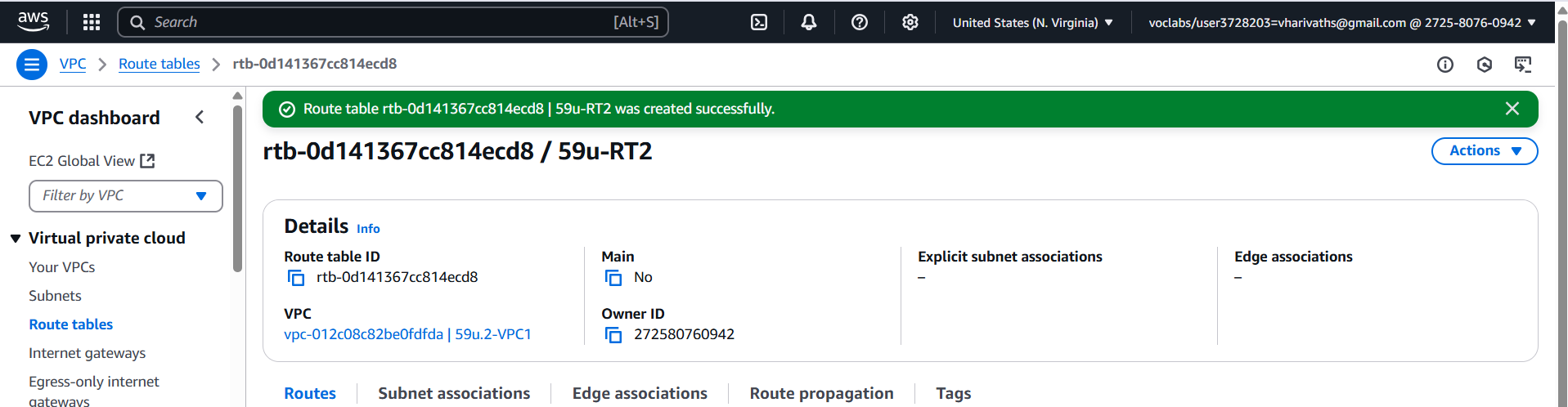
1. **Name:** 59u-RT2



1. **VPC:** 59u-VPC2

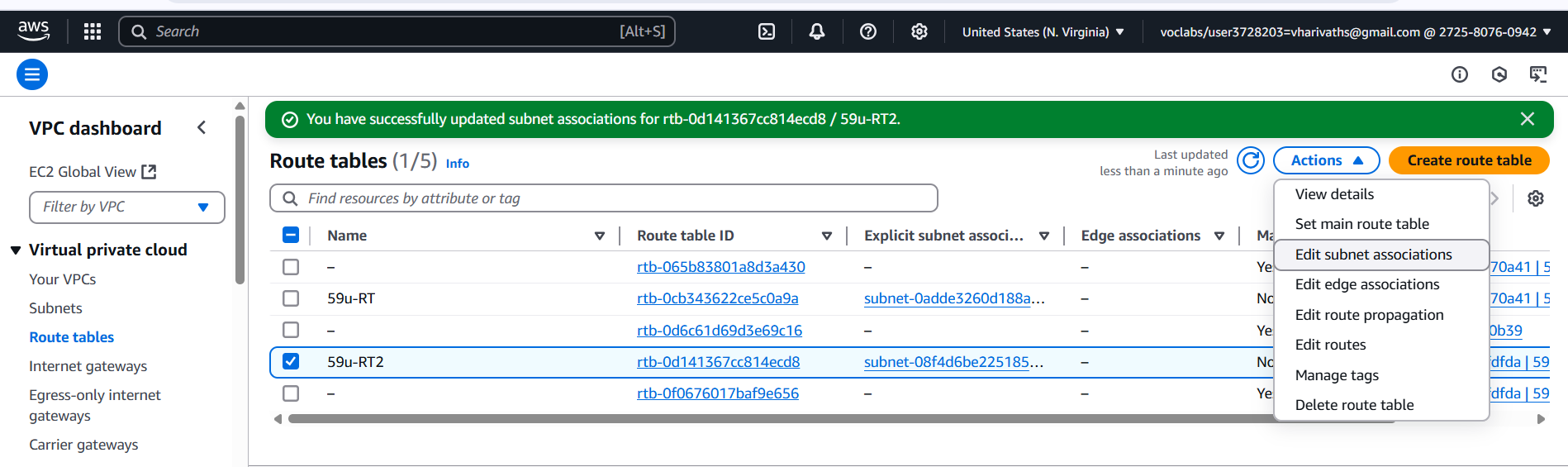


1. Click **Create**

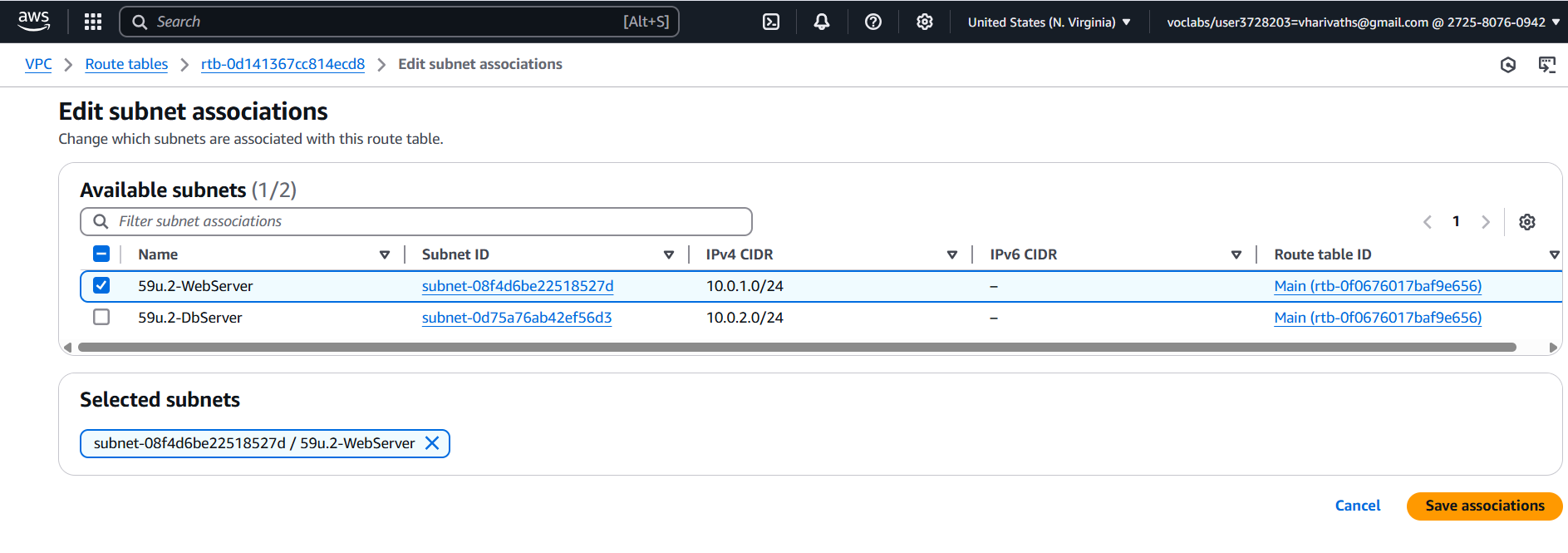


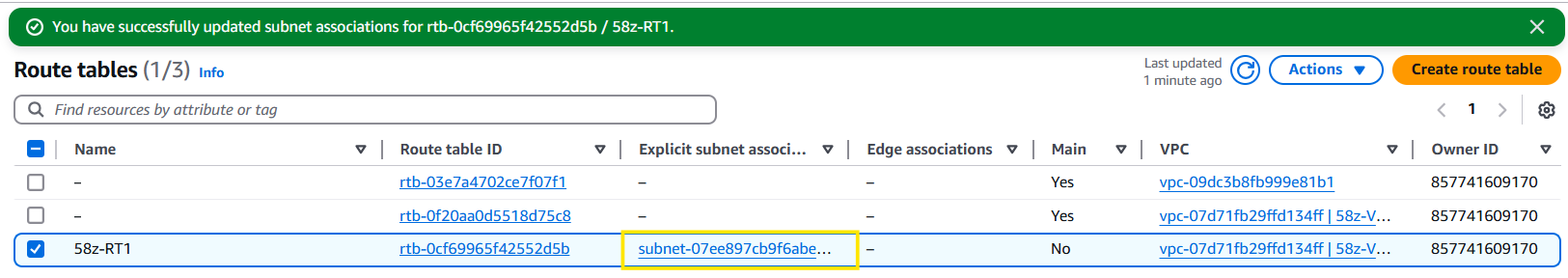
**Step 6: Associate Route Table with Public Subnet**

1. Select the 59u-RT2 route table.
2. **Go to:** Subnet Associations → Edit Subnet Associations.



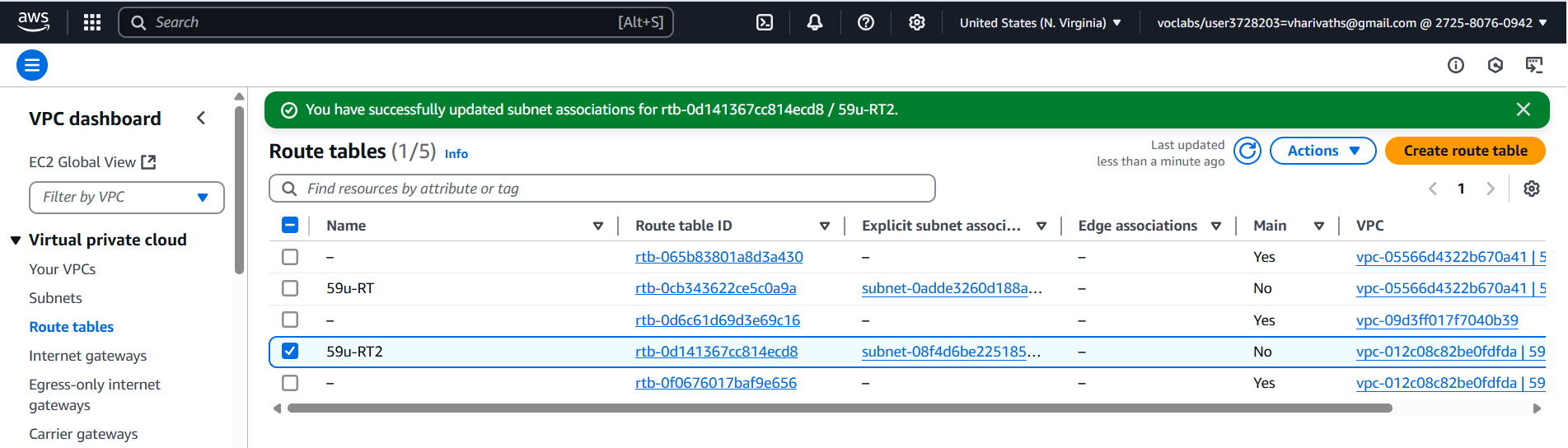
1. Select the **59u-WebServer subnet** (10.0.1.0/24) → Save.



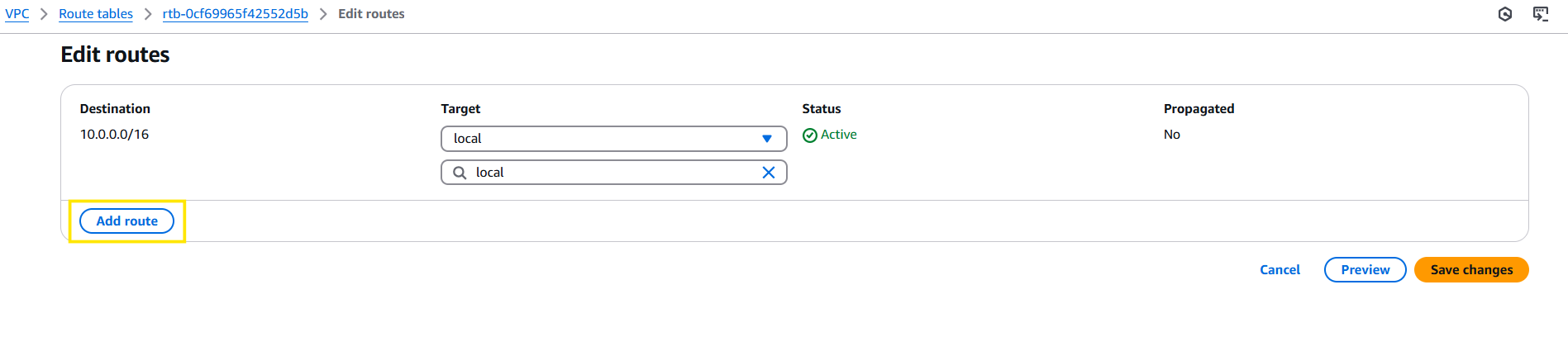


**Step 7: Add Internet Gateway Route**

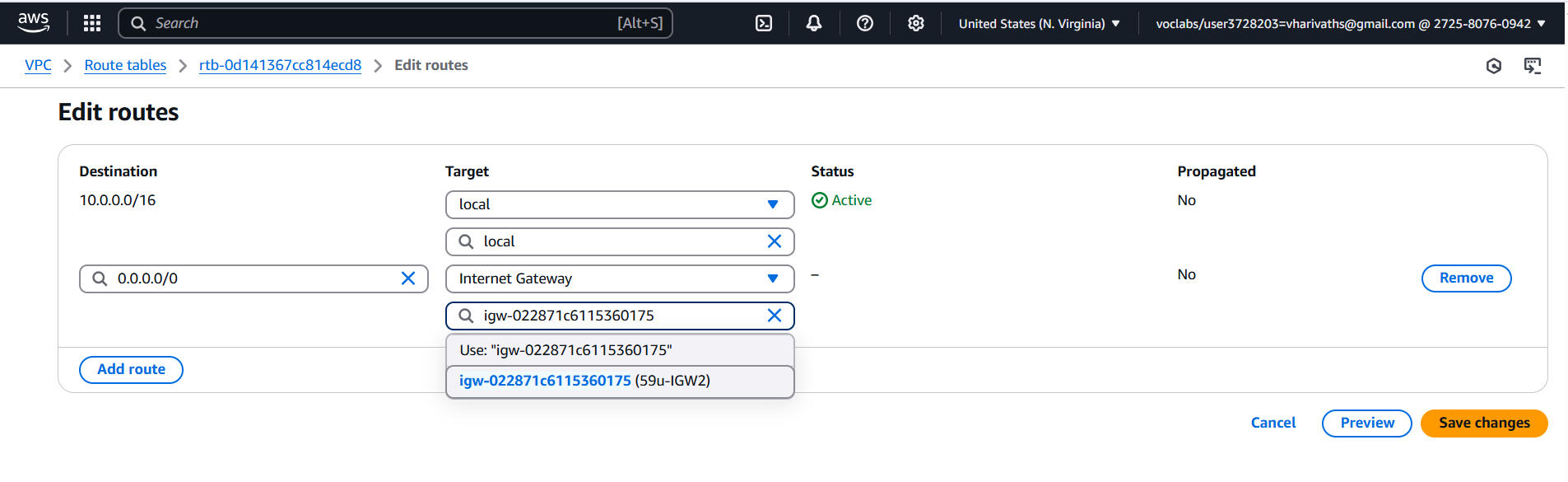
1. Select 59u-RT1.
2. Go to **Routes → Edit Routes**.



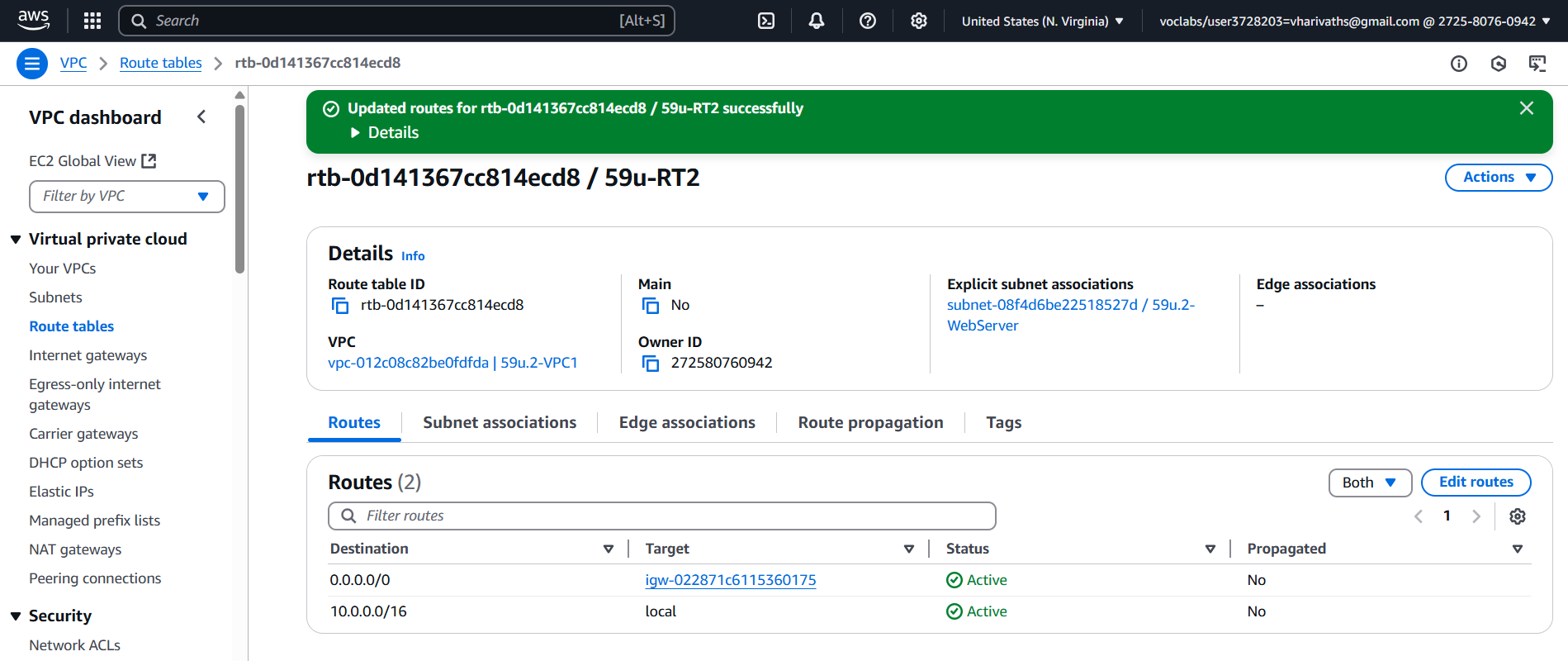
1. Click **Add Route**:



* + **Destination:** 0.0.0.0/0
  + **Target:** 59u-IGW2



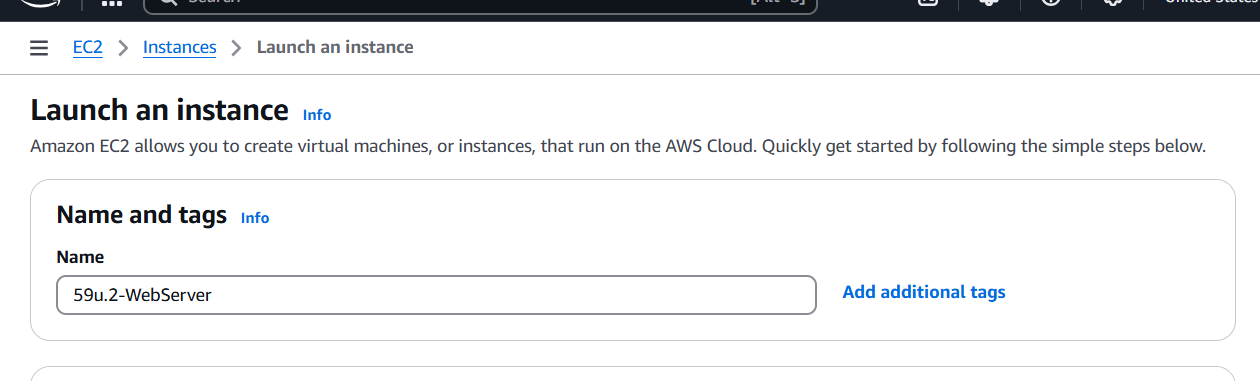
1. Save the route table.



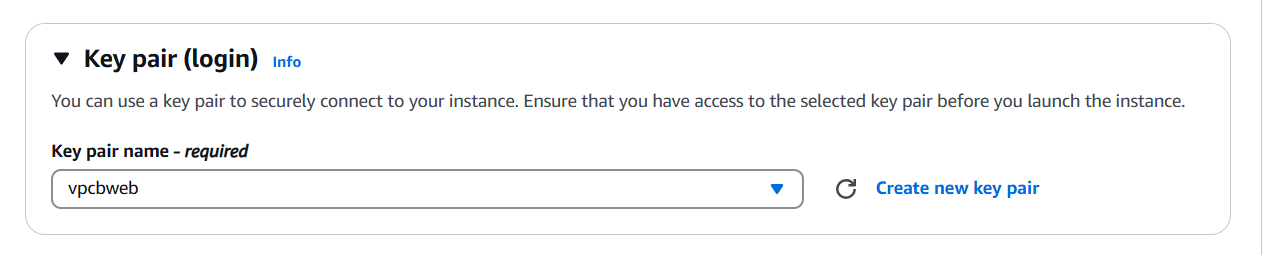
✅ Now, 59u-WebServer is public and has internet connectivity.

**Step 8: Launch Web Server (Public Subnet)**

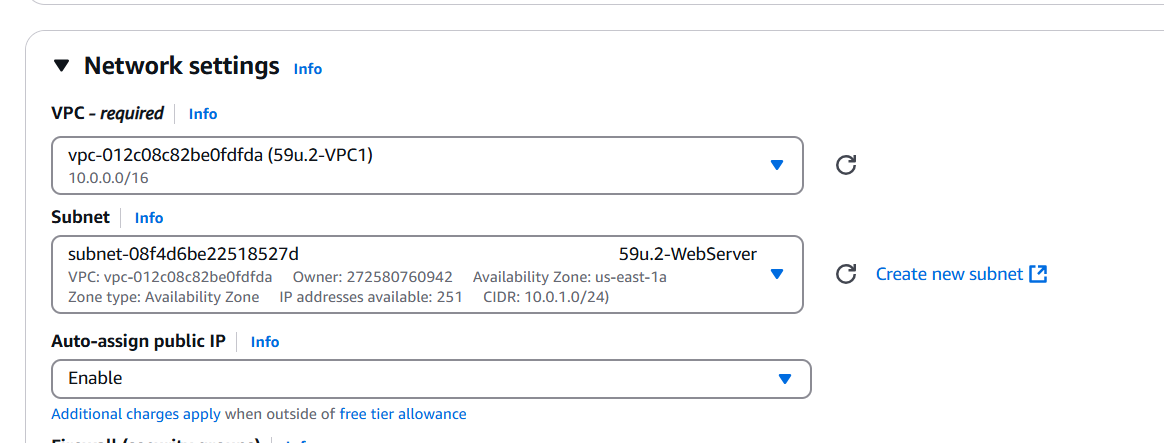
1. **Go to:** EC2 → Instances → Launch Instance.
2. **Name:** 59u-WebServer
3. **AMI:** Amazon Linux



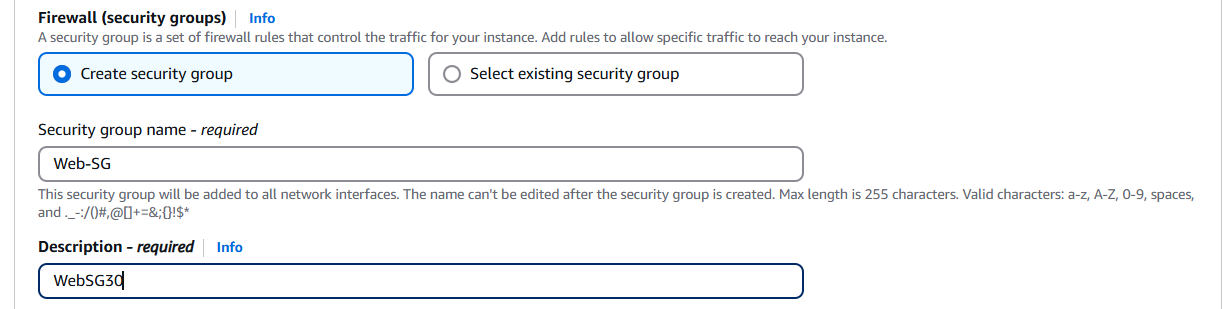
1. **Create New Keypair:**
   * **Name:** vpcbweb
   * **Key Pair Type:** RSA
   * **Private Key File Format:** .pem
   * Click **Create Key Pair** → Download the .pem file and **save it**.



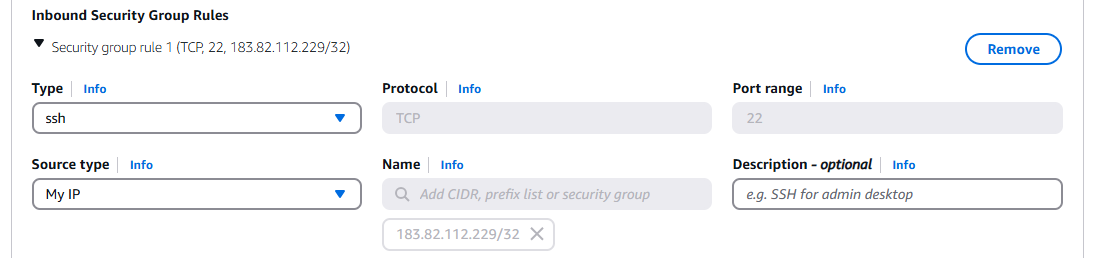
1. **Network:** 59u-VPC2
2. **Subnet:** 10.0.1.0/24 (Public Subnet)



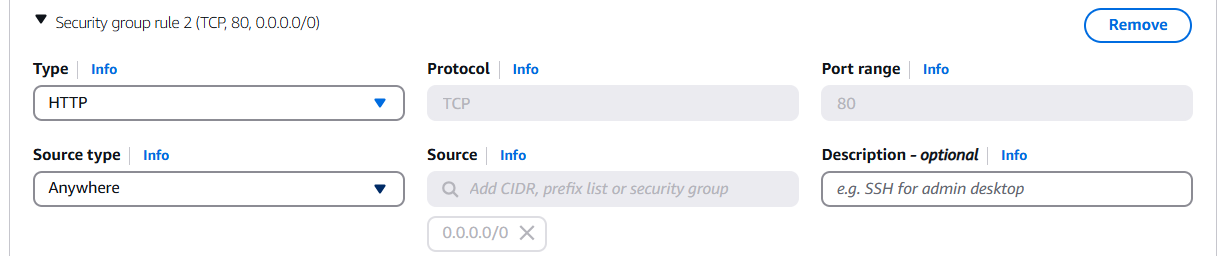
1. **Security Group:**
   * **Name:** Web-SG



* + **Inbound Rules:**
    - **Type:** SSH → **Source:** My Ip



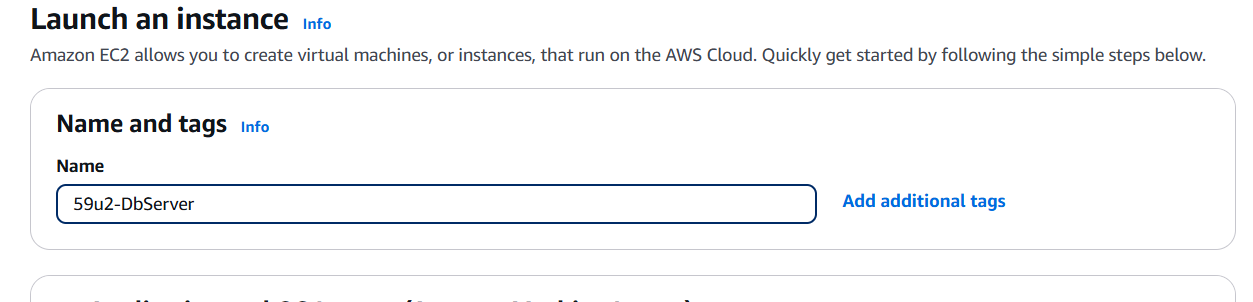
* + - **Type:** HTTP → **Source:** Anywhere



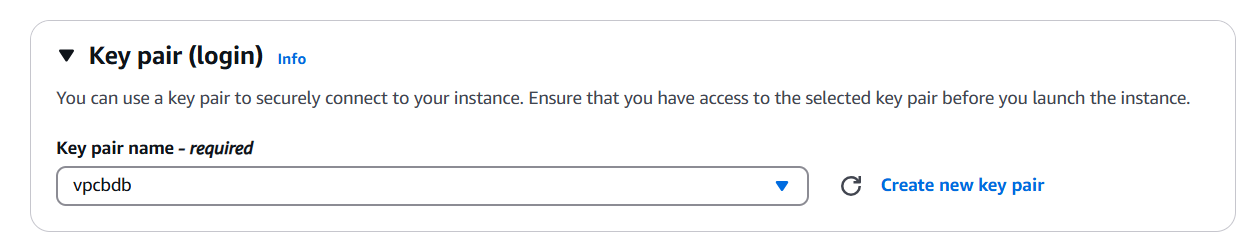
1. Click **Review and Launch → Launch → View Instance**

**✅ Step 9: Launch Database Server (Private Subnet)**

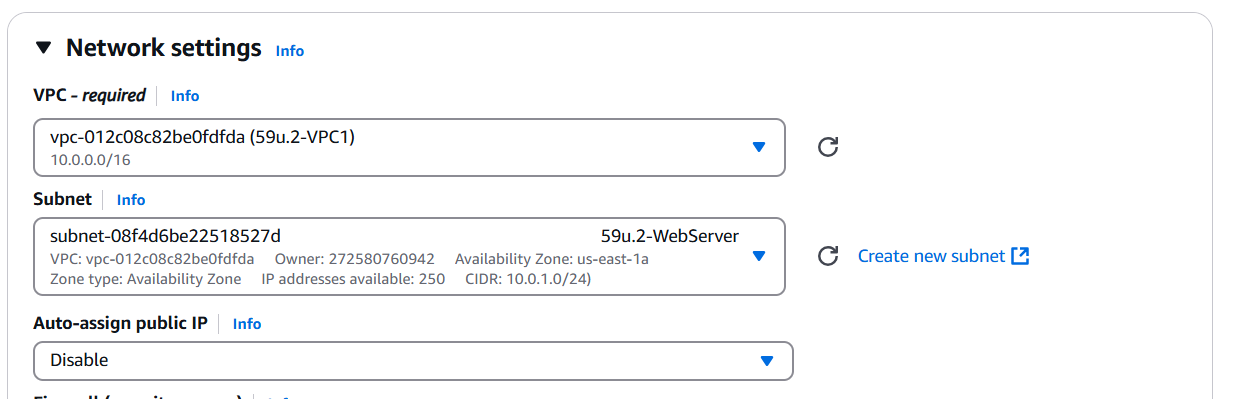
1. **Go to:** EC2 → Instances → Launch Instance.
2. **Name:** 59u2-DbServer
3. **AMI:** Amazon Linux



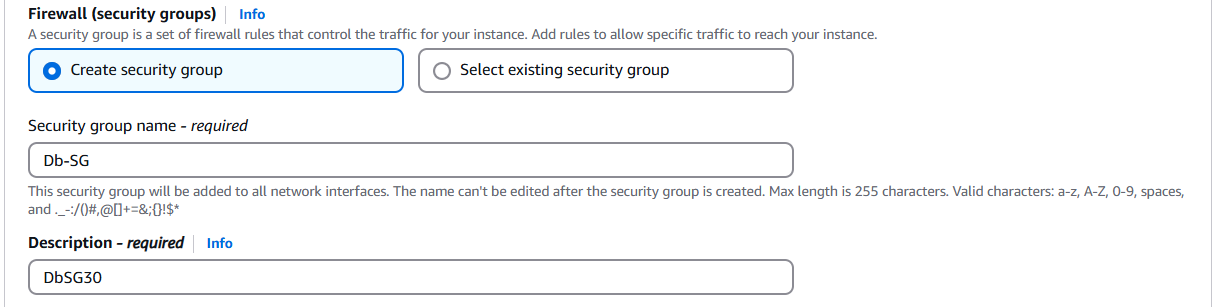
1. **Create New Keypair:**
   * **Name:** dbKP30
   * **Key Pair Type:** RSA
   * **Private Key File Format:** .pem
   * Click **Create Key Pair** → Download the .pem file and **save it**.



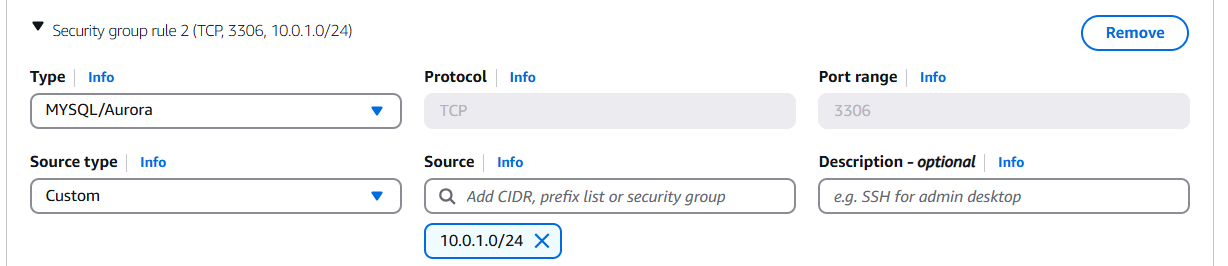
1. **Network:** 59u-VPC1
2. **Subnet:** 10.0.2.0/24 (Private Subnet)



1. **Security Group:**
   * **Name:** Db-SG



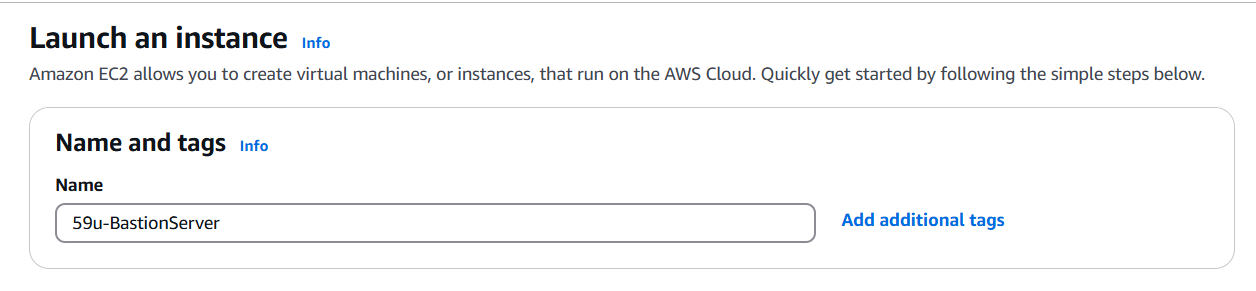
* + **Inbound Rule:**
    - **Type:** MYSQL/Aurora → **Source:** Custom → 10.0.1.0/24



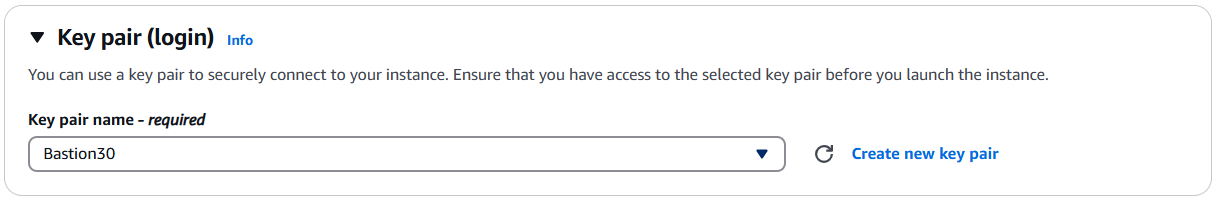
1. Click **Review and Launch → Launch → View Instances**

**✅ Step 10: Create Bastion (Jump) Server**

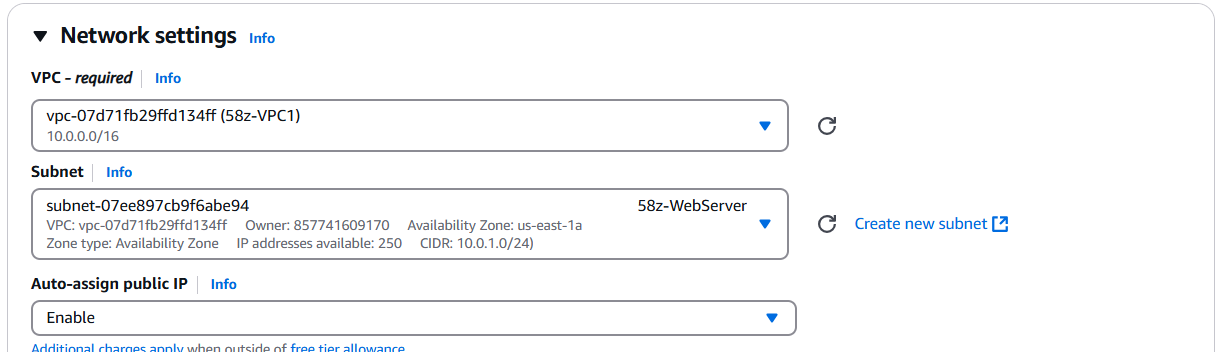
1. **Go to:** EC2 → Instances → Launch Instance.
2. **Name:** 59u-BastionServer
3. **AMI:** Amazon Linux



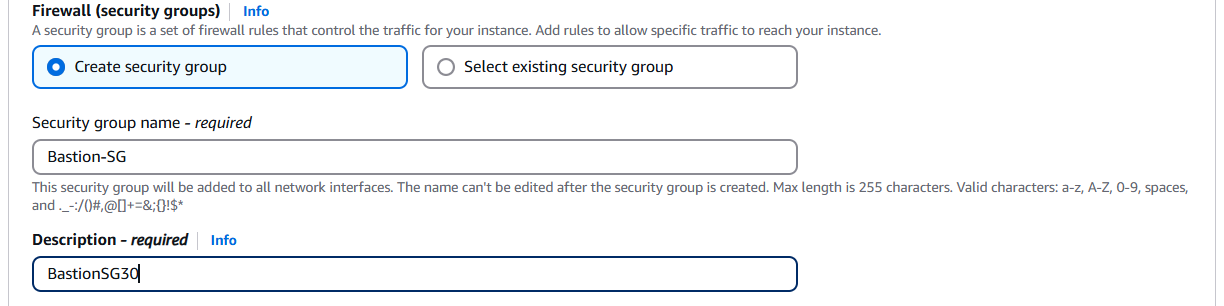
1. **Create New Keypair:**
   * **Name:** Bastion30
   * **Key Pair Type:** RSA
   * **Private Key File Format:** .ppk
   * Click **Create Key Pair** → Download the .ppk file.



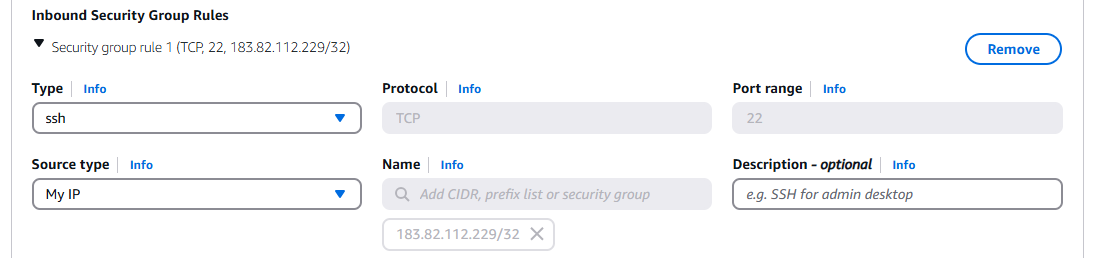
1. **Network:** 59u2-VPC1
2. **Subnet:** 10.0.1.0/24 (Public Subnet)



1. **Security Group:** Bastion-SG



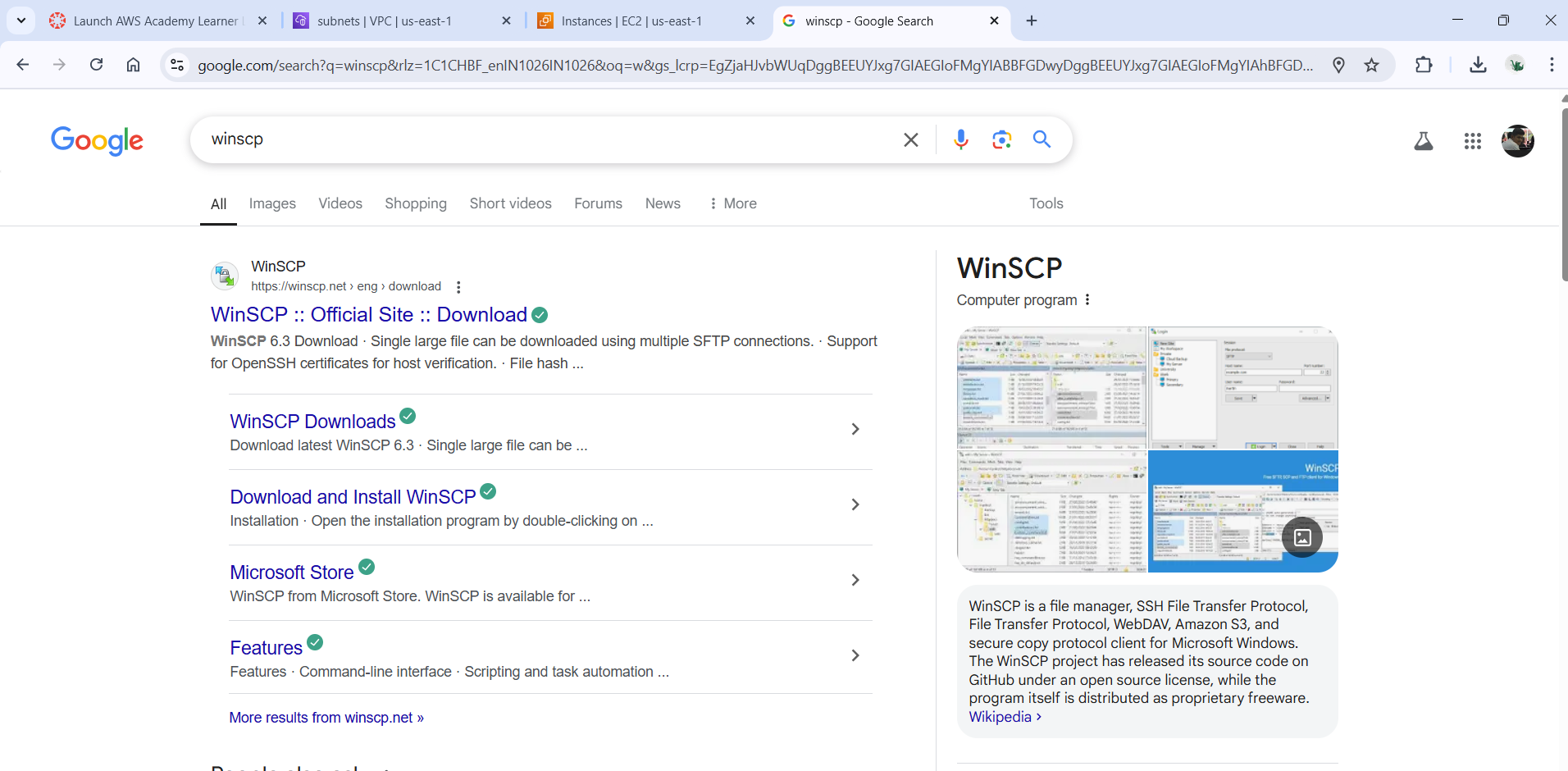
* + **Inbound Rule:**
    - **Type:** SSH
    - **Source:** My IP



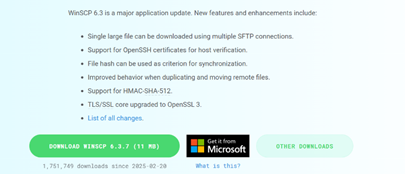
1. Click **Review and Launch → Launch → View Instances**

**Step 11: Connect to Bastion Server**

1. Open **WinSCP** (download if you don’t have it).
   * Search for “Download WinSCP” in any browser, and click on the first link



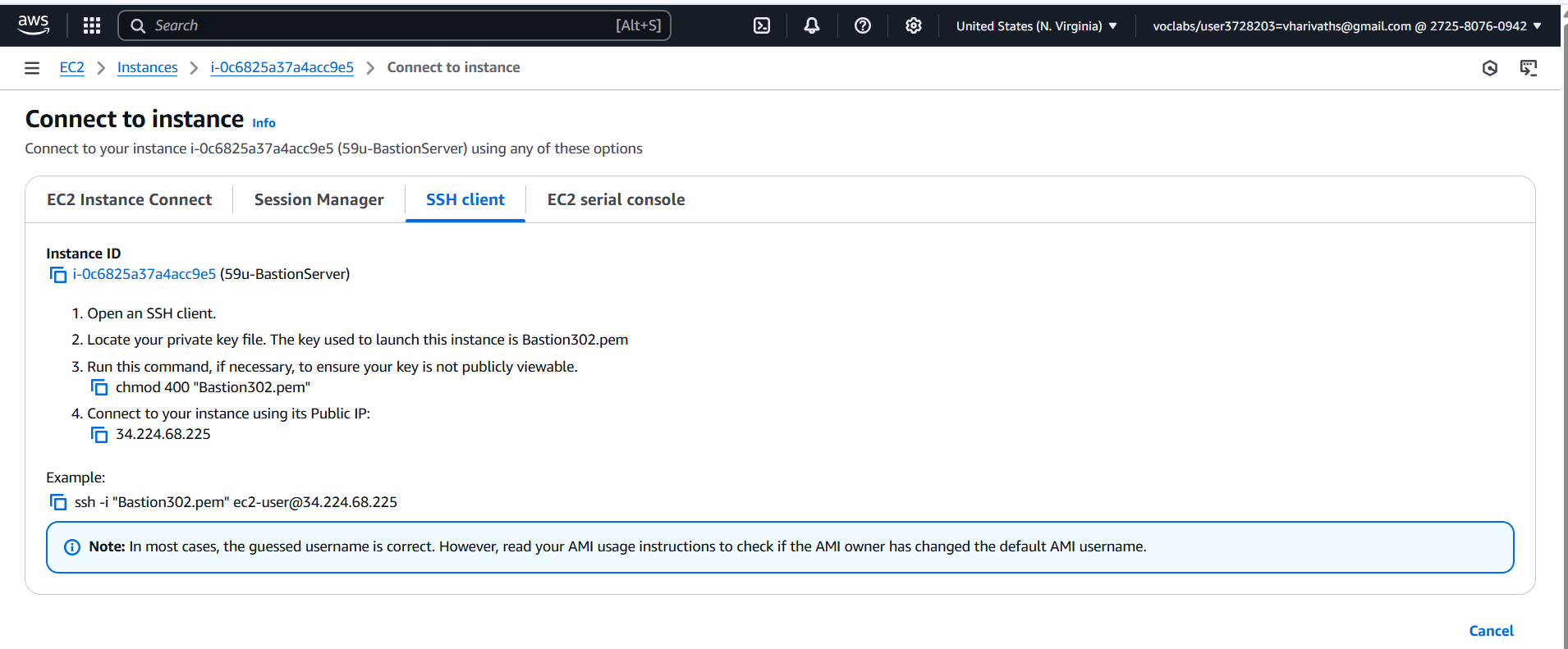
* + Scroll down to download WinSCP

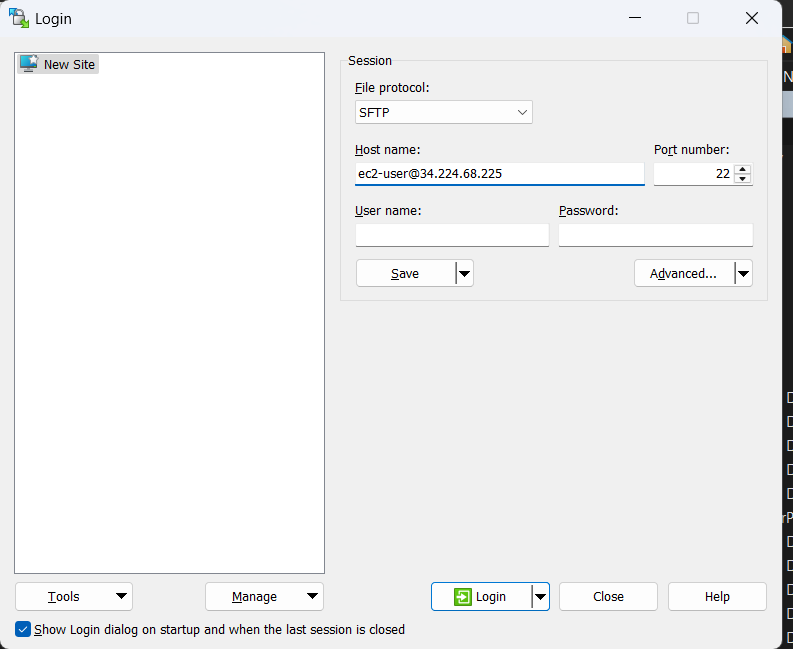


* + Set up on your PC with the necessary (default) settings

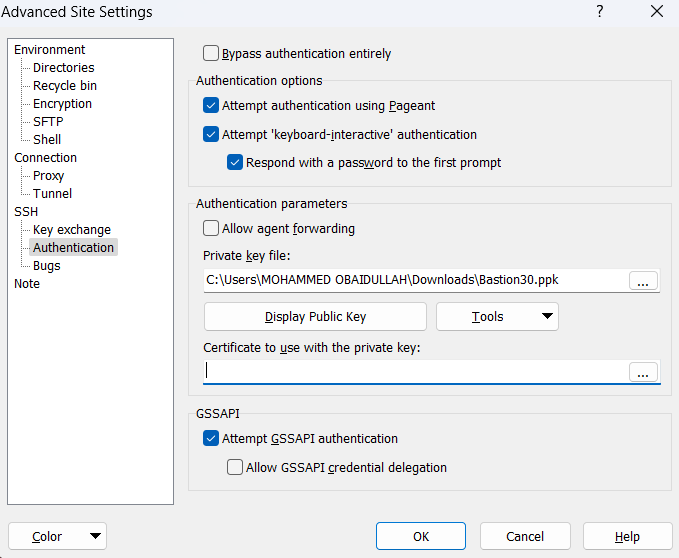


1. **Host:** ec2-user@<bastion\_public\_ip> i.e, ec2-user@54.166.158.199

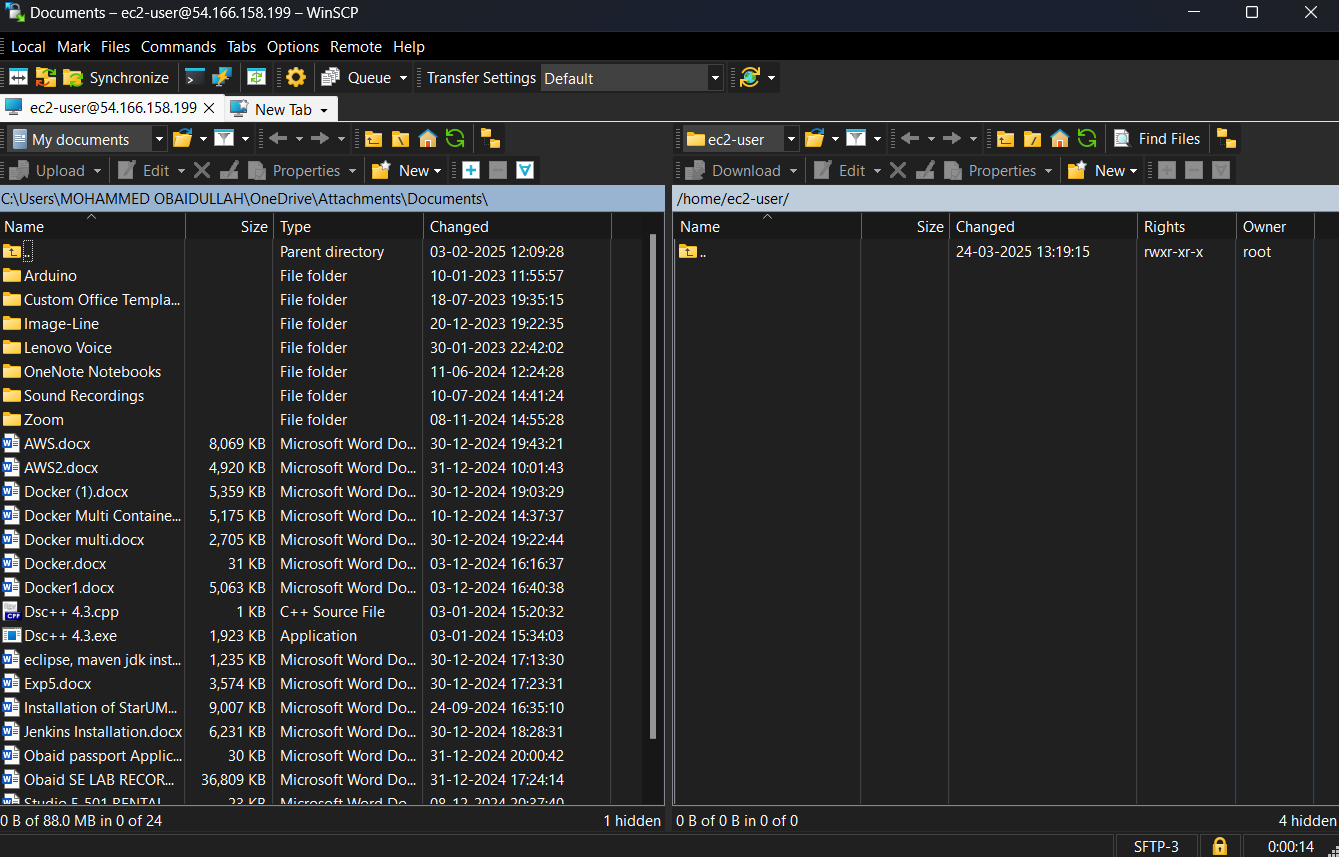




1. **Authentication:**
   * Go to **Advanced → SSH → Authentication**
   * Select the .ppk file (Bastion30.ppk)

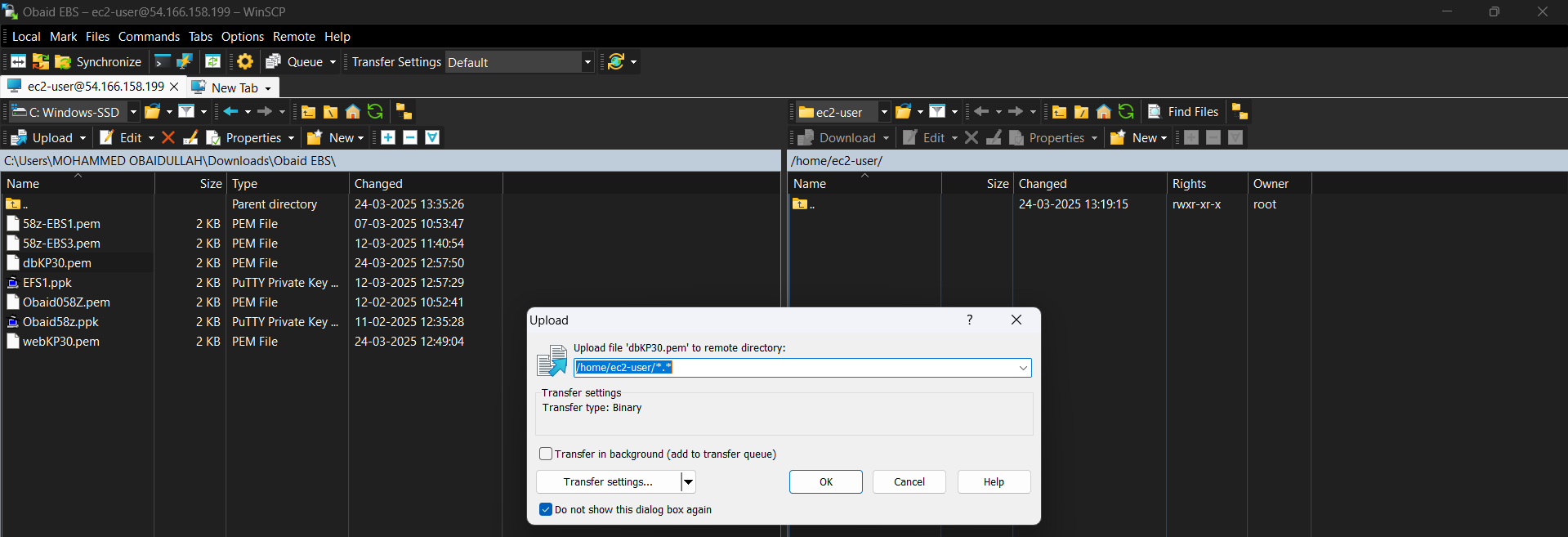


1. Click **Open → Login**

****

**✅ Step 12: Transfer the .pem Key Pair to Bastion**

1. In WinSCP, **drag and drop** the dbKP30.pem file from your local system to the Bastion server.



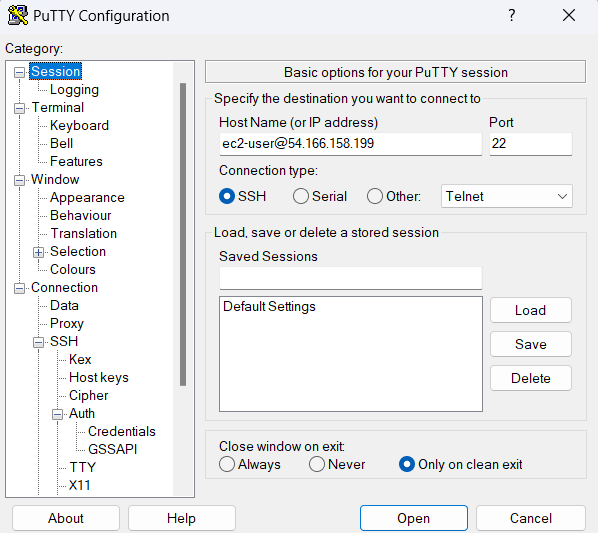
1. On the Bastion server, run: ls

* **Open PuTTY to Access the Bastion Server’s Terminal**

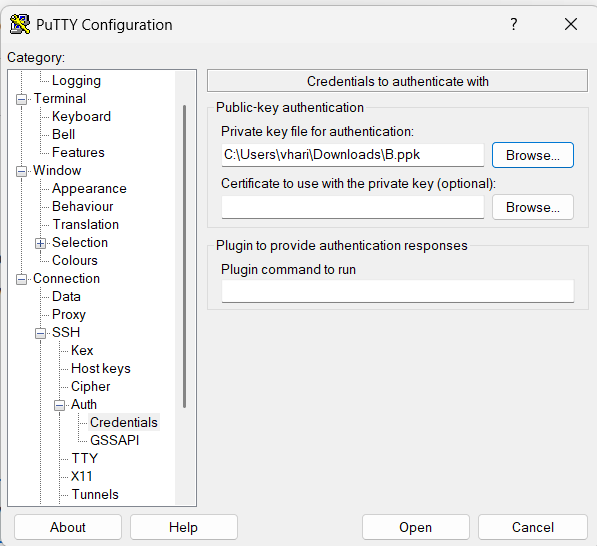
You cannot run ls directly in WinSCP—it is a file transfer tool, not a terminal.  
 To run the ls command, you need to **SSH into the Bastion server using PuTTY**.

* Steps to SSH into Bastion Using PuTTY:

1. **Open PuTTY**.
2. **Host Name:**ec2-user@<bastion\_public\_ip> i.e, ec2-user@54.166.158.199

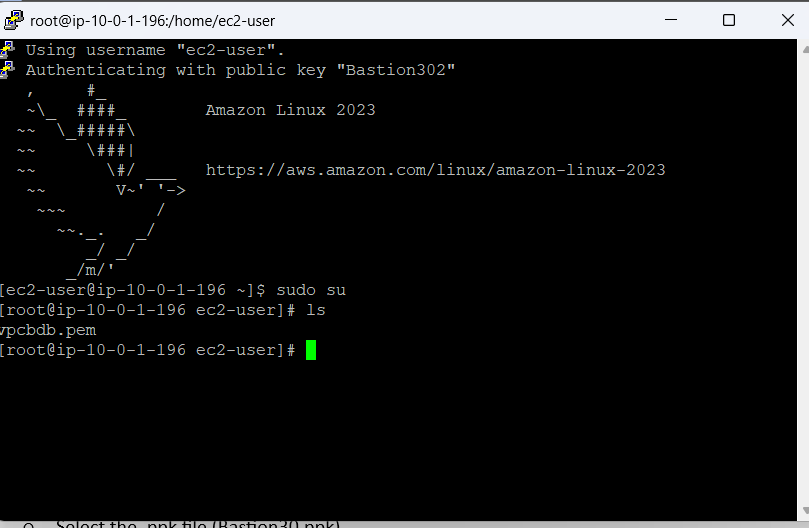


1. **Authentication:**
   * Go to **Connection → SSH → Auth**.
   * Select the .ppk file (Bastion30.ppk).



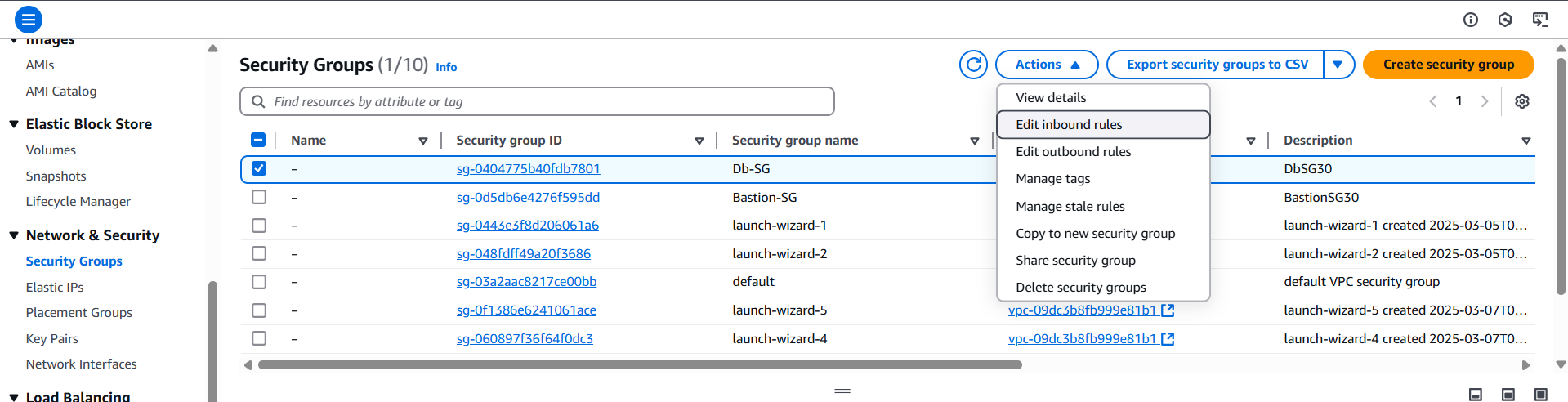
1. Click **Open → Connect**

✅ You should see the dbKP30.pem file.

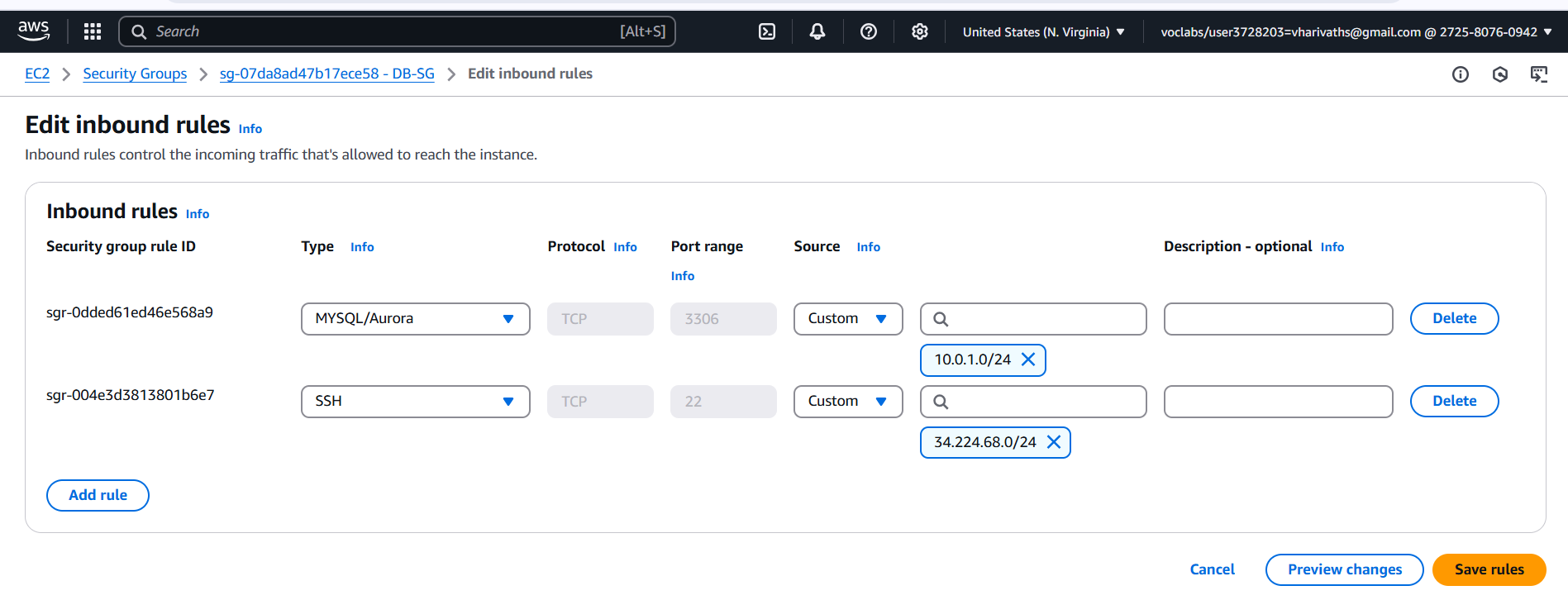


**✅ Step 13: Enable SSH Access to DB Server from Bastion**

1. Go to **EC2 → Security Groups**.
2. Select Db-SG → **Inbound Rules → Edit**.



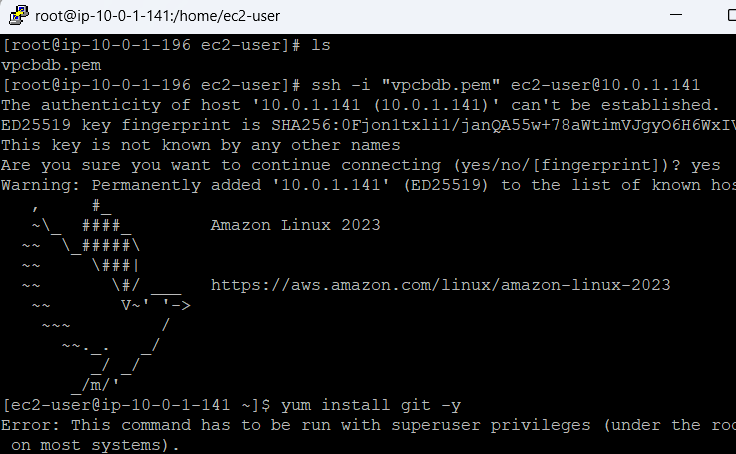
1. **Add Rule:**
   * **Type:** SSH
   * **Source:** Custom → 10.0.1.0/24(we should actually give IP address of Bastion server, so only it can access it but as we are not able to do that we gave the range of the public subnet which allows communication to all the instances in Public Subnet)



1. Click **Save Rules**

**✅ Step 14: Connect to DB Server from Bastion**

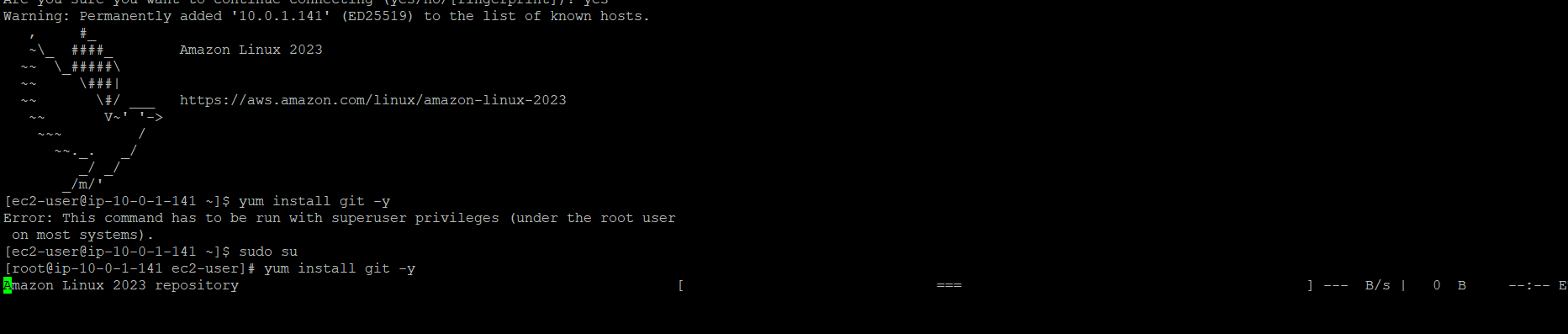
1. On Bastion, run: ssh -i "vpcbdb.pem" ec2-user@10.0.1.141



✅ You are now connected to the DB server!

**✅ Step 15: Install MySQL (Fail)**

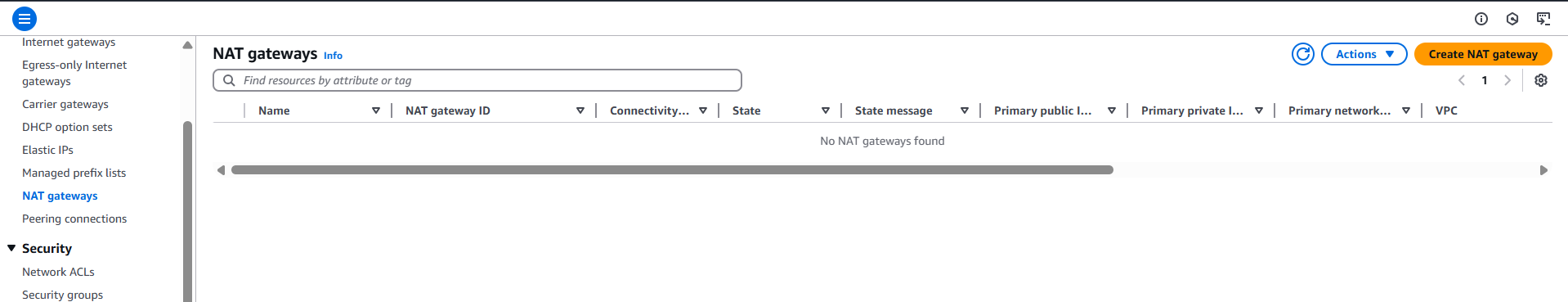
1. Switch to root: $ sudo su
2. Run:# yum install mysql -y



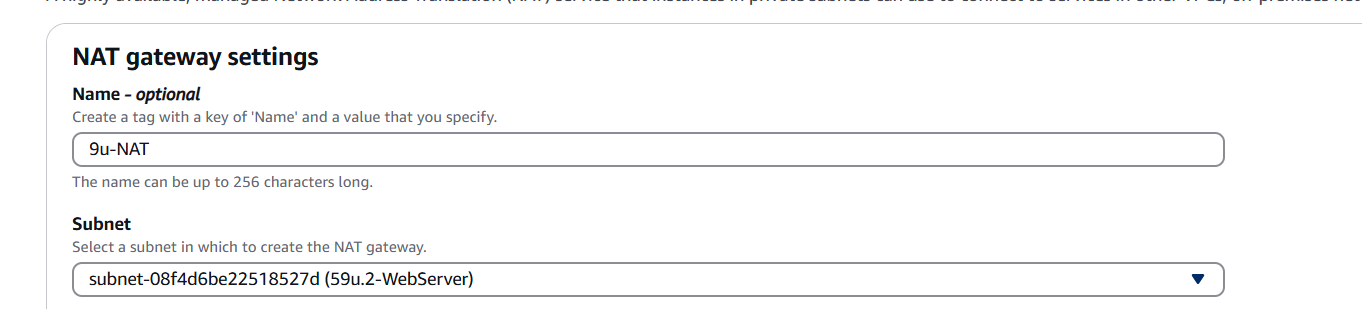
**Fails:** No internet access in the private subnet.

**Step 16: Create a NAT Gateway**

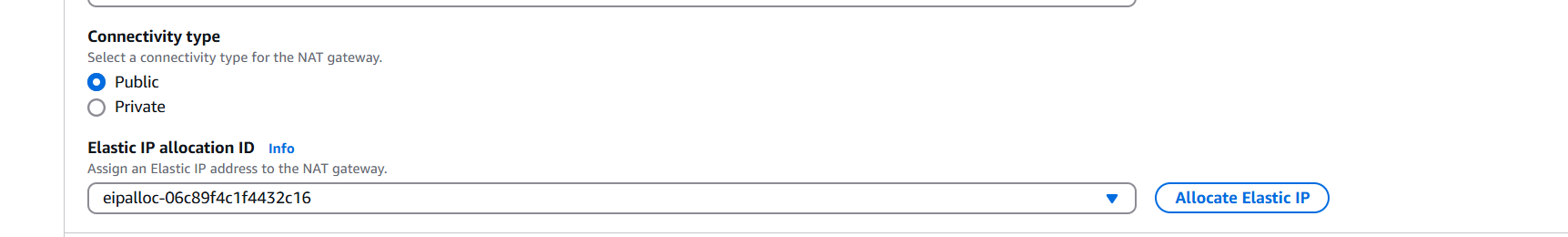
1. **Go to VPC → NAT Gateways → Create NAT Gateway.**

****

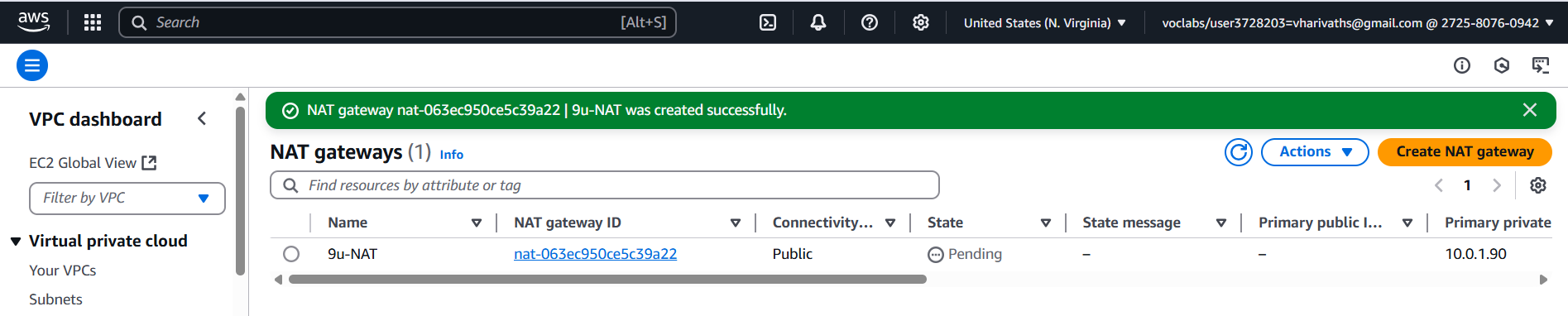
1. **Choose the public subnet:**
   * **Subnet: 10.0.1.0/24 → The public subnet (Web server subnet).**

****

1. **Elastic IP:**
   * **Click on Allocate Elastic IP or select an existing one.**

****

1. **Click Create NAT Gateway.**
2. **Wait for the status to become Available.**

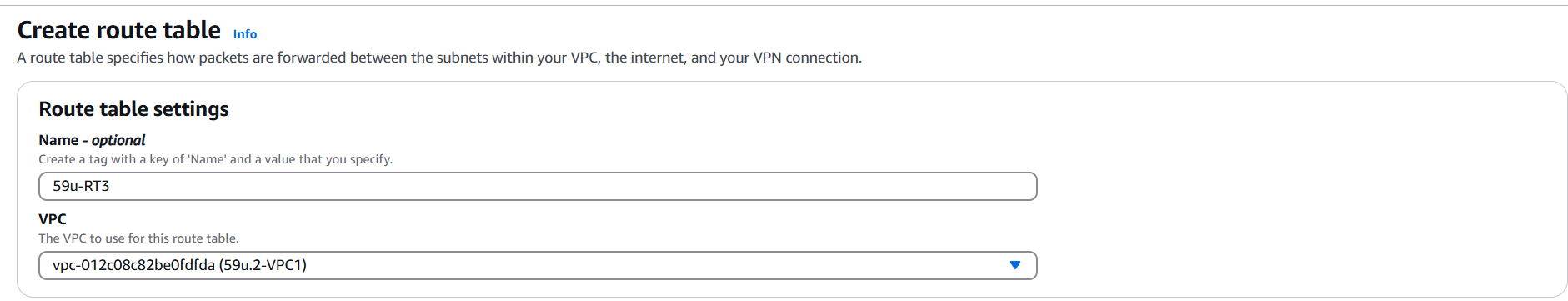
****

**✅ You now have a NAT Gateway ready to provide internet access.**

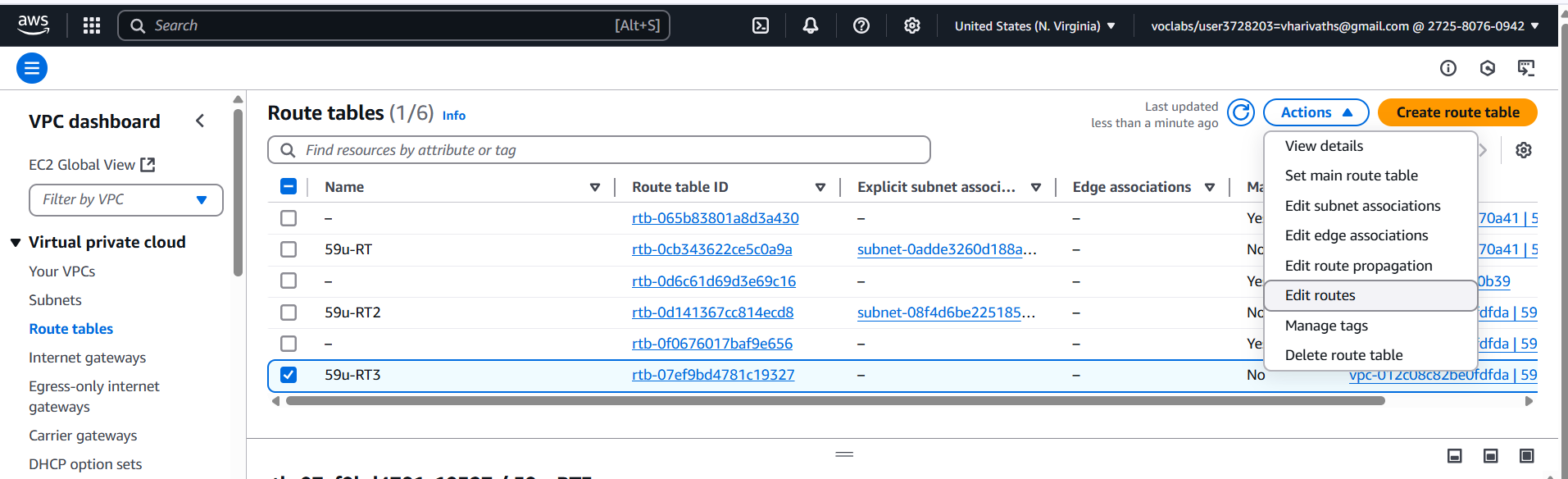
**✅ Step 17: Modify the Route Table for the Private Subnet**

**You need to replace the NAT server in the route table with the NAT gateway.**

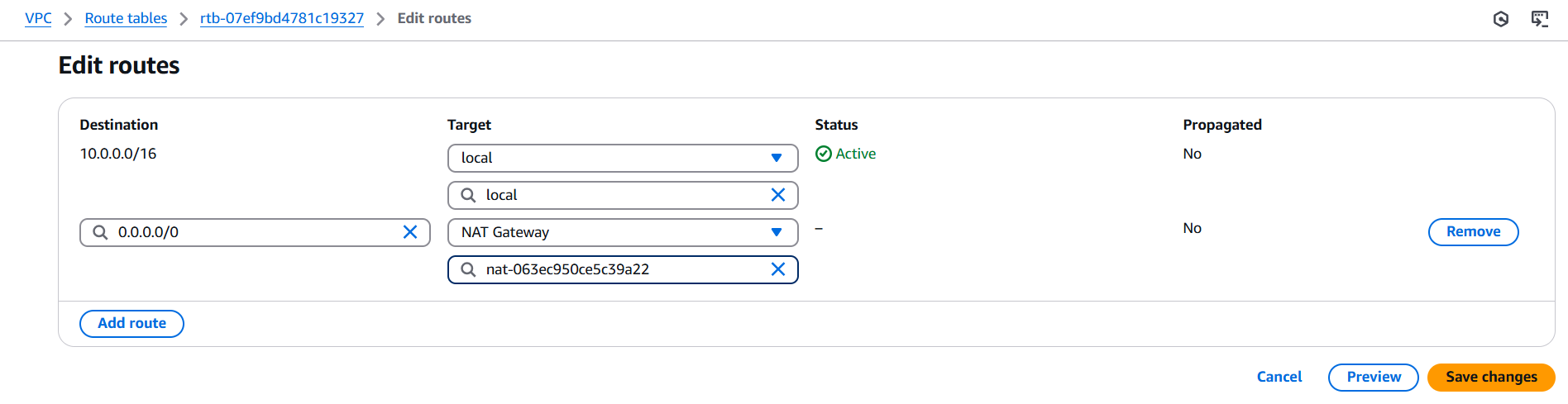
1. **Go to VPC → Route Tables → Click Create Route Table →Name: 59u-RT3 → VPC:** 59u-VPC1



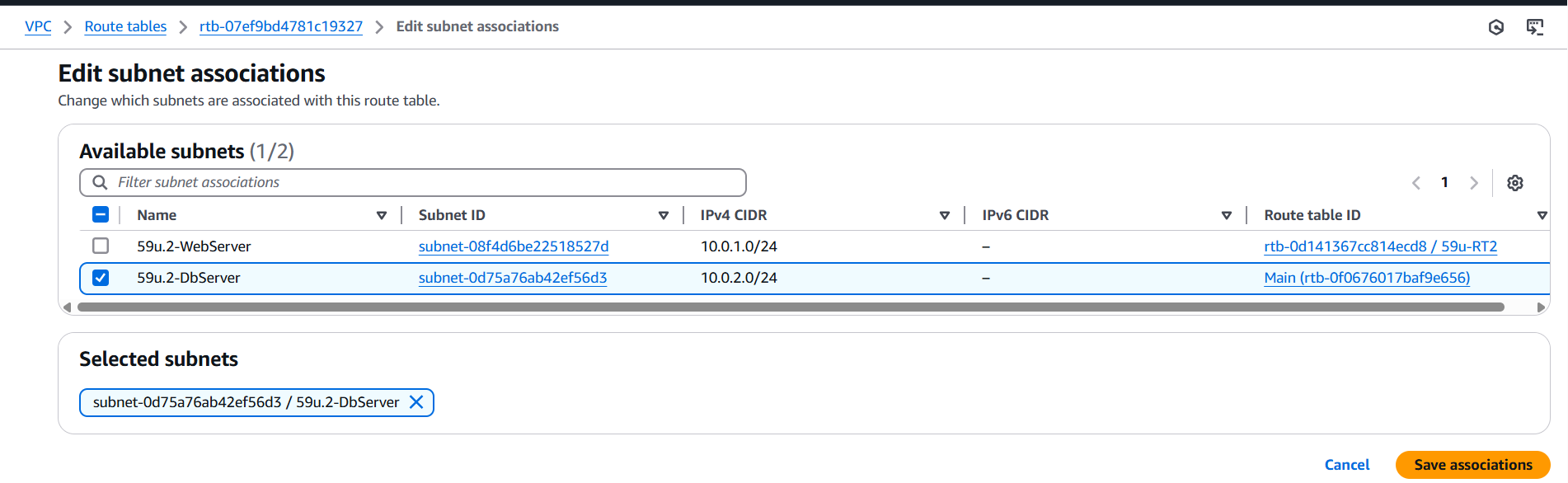
1. **Go to the Routes tab → Select** 59u-RT3 **Click Actions → Edit Routes.**

****

1. **Add a new route:**
   * **Destination: 0.0.0.0/0**
   * **Target: Select the NAT Gateway →**59u-NATGW **from the dropdown.**

****

1. **Save changes.**
2. **Edit Subnet Associations:** 
   * **Check 59u-DbServer(private Subnet)**

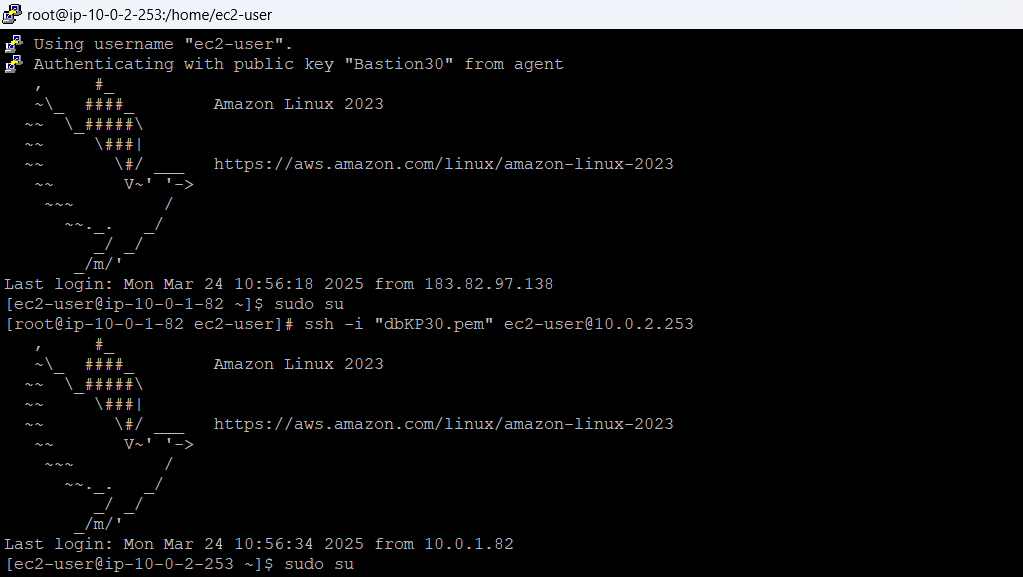
****

1. **Save associations.**

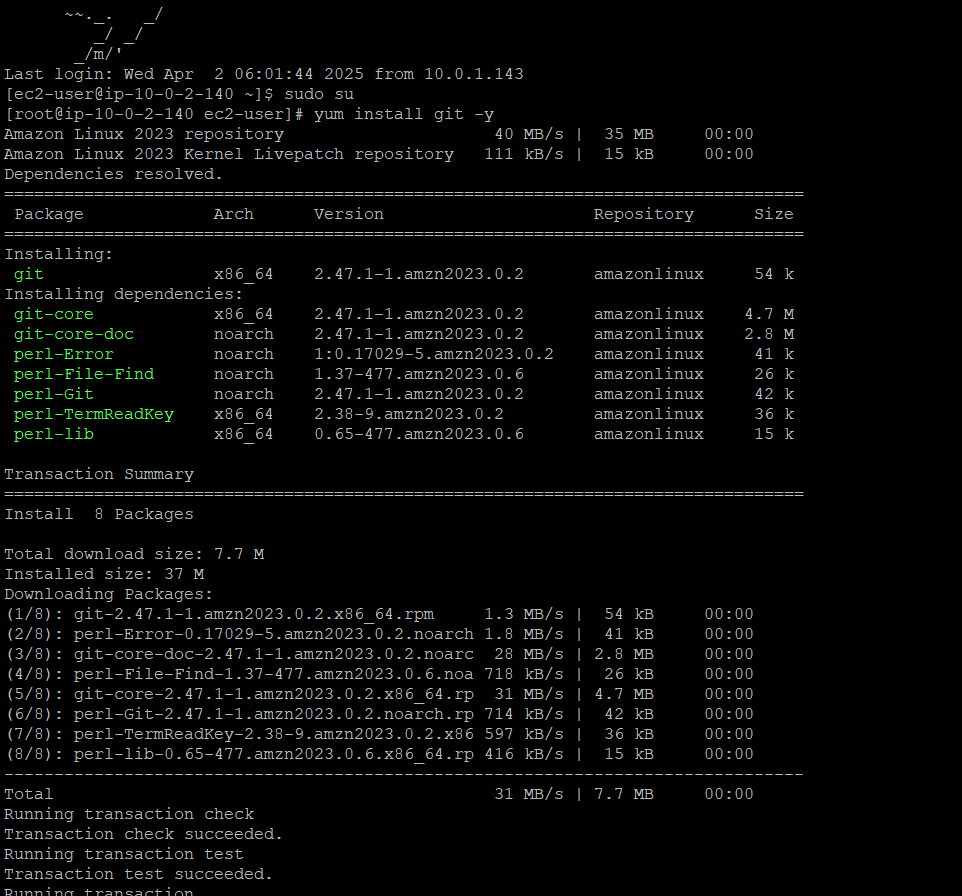
**✅ Your private subnet is now routed through the NAT Gateway.**

**✅ Step 18: Final Verification**

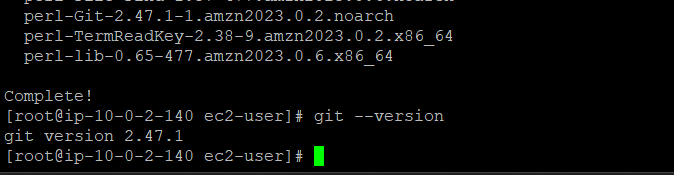
1. Connect to Bastion → Connect to DB Server



1. Run:# yum install git -y



1. Run: #git –version to verify installation of git



**Success:** Git installs successfully on the private server with internet access through NAT. 🚀