AWS Project: A Secure, Scalable, and Highly Available Production VPC Infrastructure.

Summary

In this project, I demonstrate setting up a secure multi-AZ VPC infrastructure with highly-available, auto-scaling Linux servers in private subnets. This involves creating a multi-AZ VPC with public and private subnets, configuring an auto-scaling group with a minimum of two EC2 instances for high availability in private subnets. Access to private subnets is restricted to the public subnets' security group. A bastion host enables secure connections to EC2 instances. An application load balancer evenly distributes traffic to web servers across multiple availability zones. The load balancer, situated in public subnets, securely routes internet traffic to the appropriate web server. Successful access to online resumes hosted on web servers confirms the load balancer's effectiveness in directing traffic to designated EC2 instances across availability zones.

Project Steps:

1. Create VPC

2. Create EC2 Instances

3. Create an Auto Scaling Group for the EC2 instances

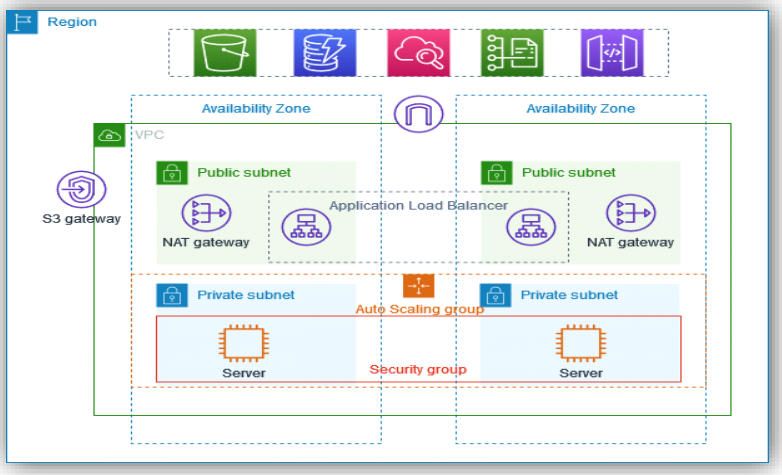
4. Create Bastion Host

5. Connect Bastion Host to EC2 instances

6. Create Application Load Balancer

7. Start Web Server on EC2 instances

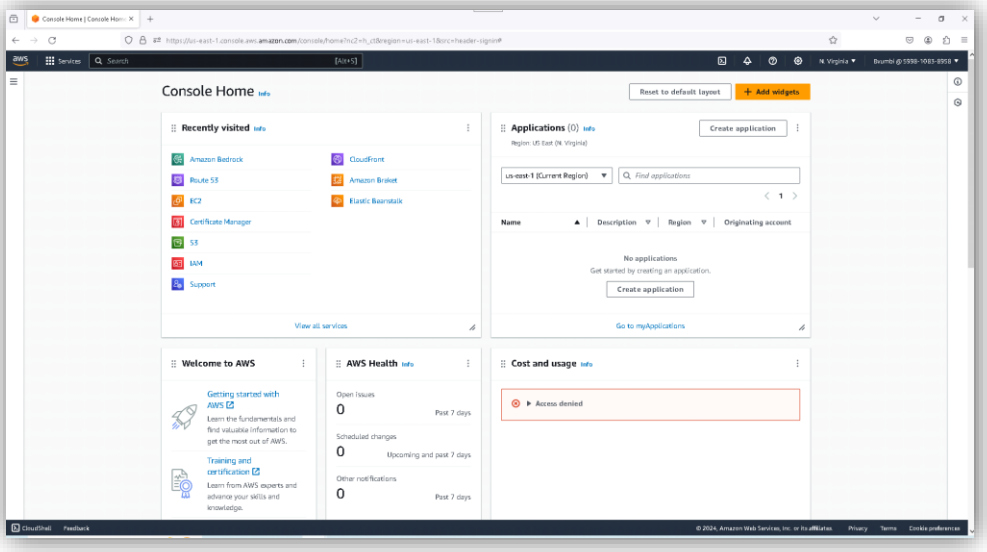
8. Connect to Web Server on EC2 instances



Topology of the Distributed System Architecture.

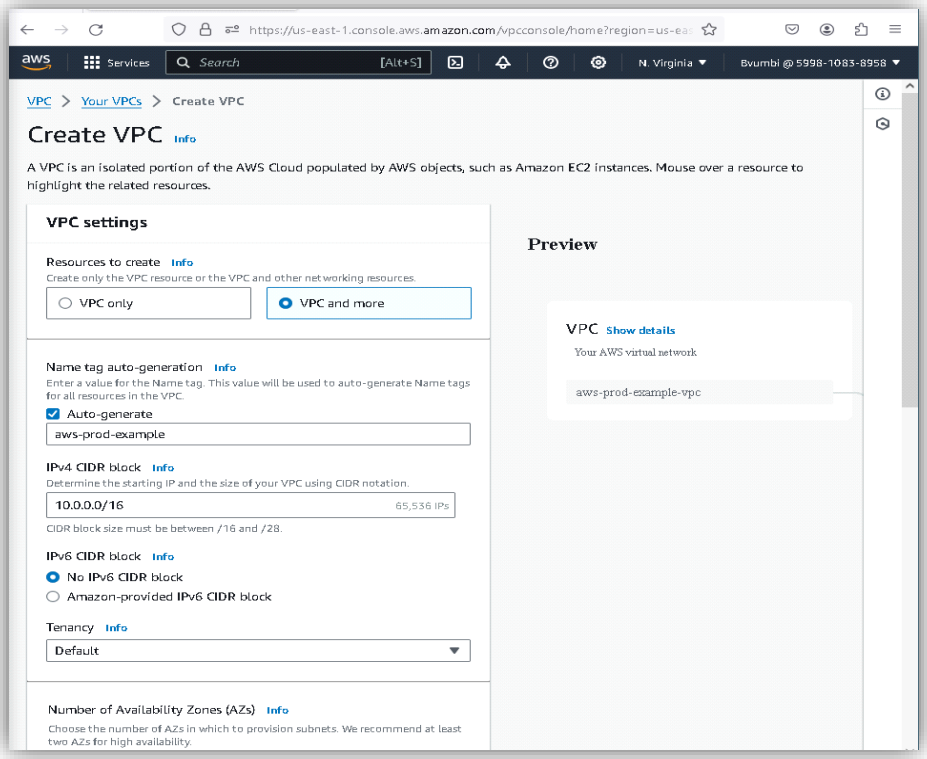
Step by Step implementation:

1.



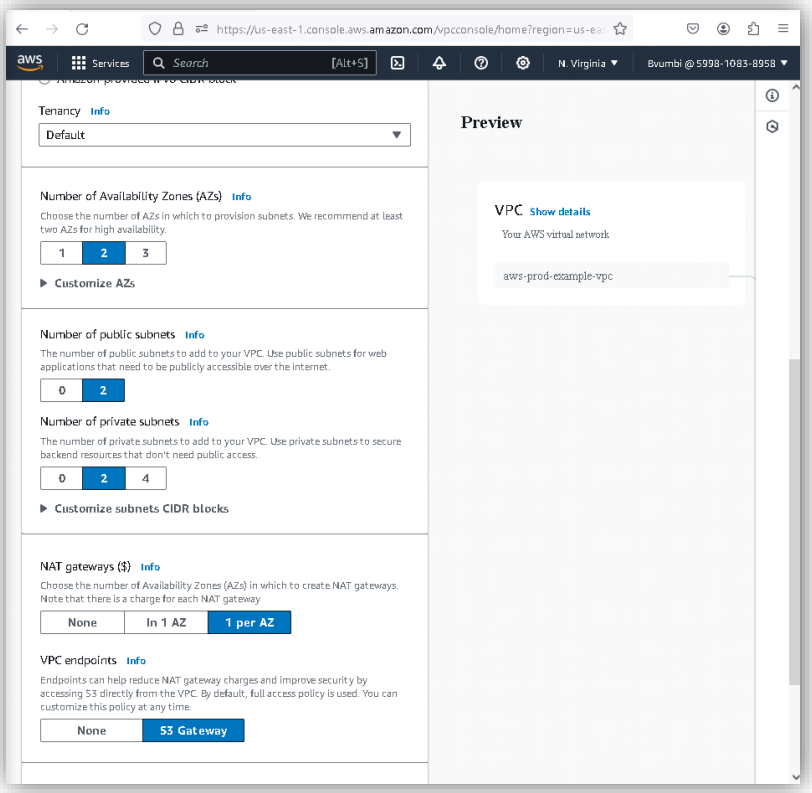
AWS Main Console.

2.



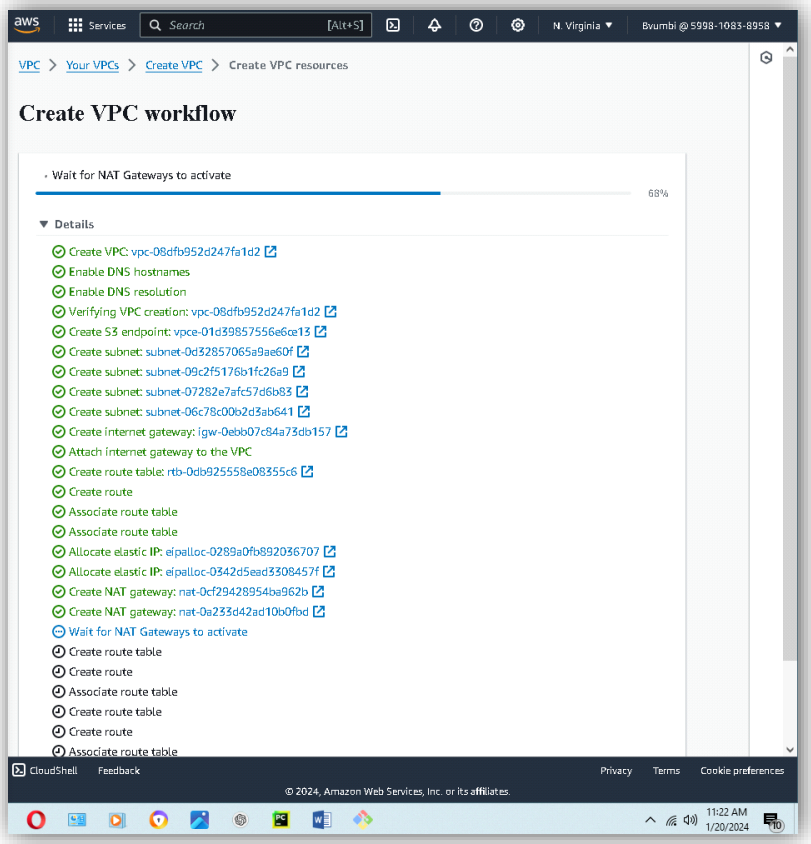
Creating Virtual Private Cloud (VPC).

3.



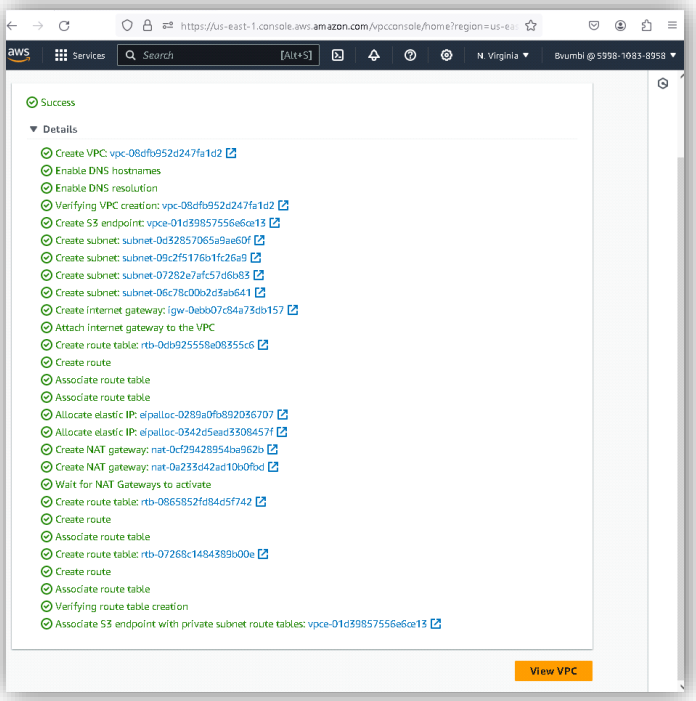
Multi-AZ Settings.

4.



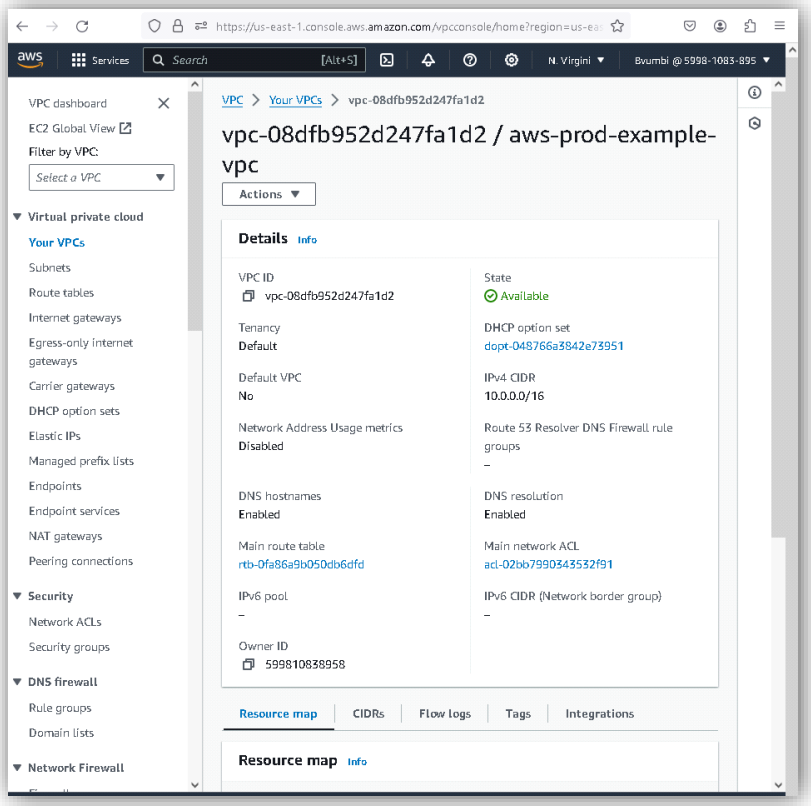
VPC Creating.

5.



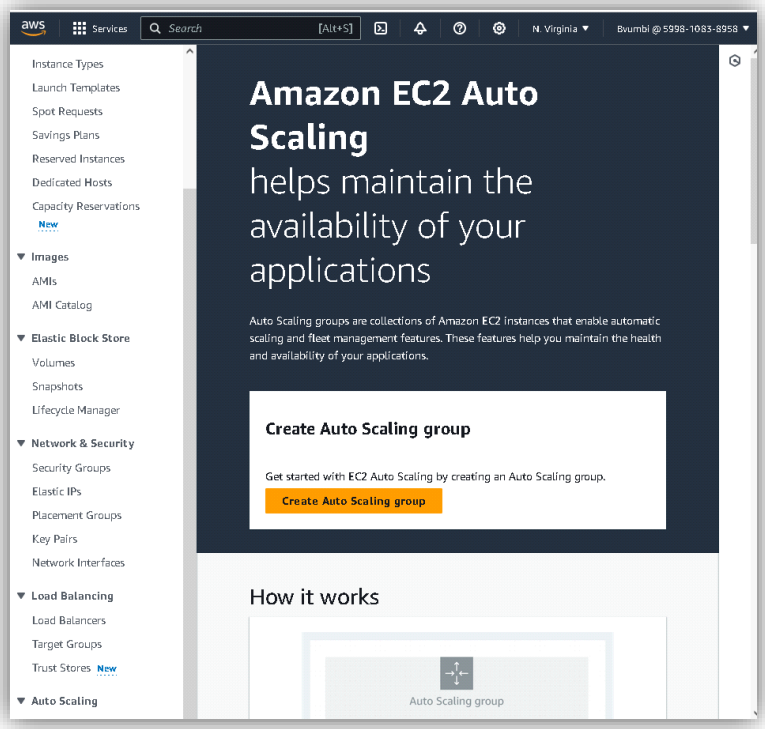
VPC Creation Success.

6.



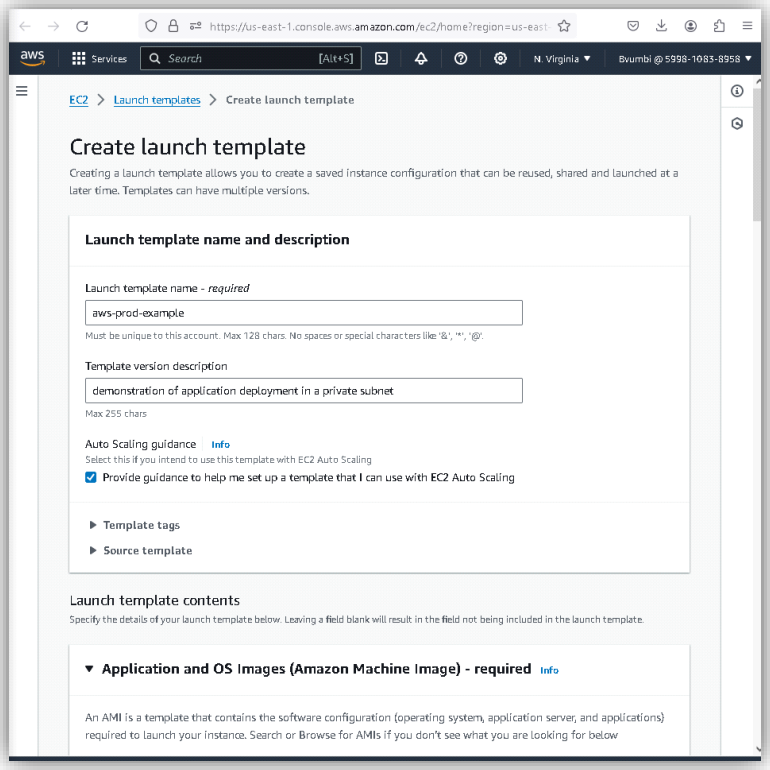
VPC View.

7.



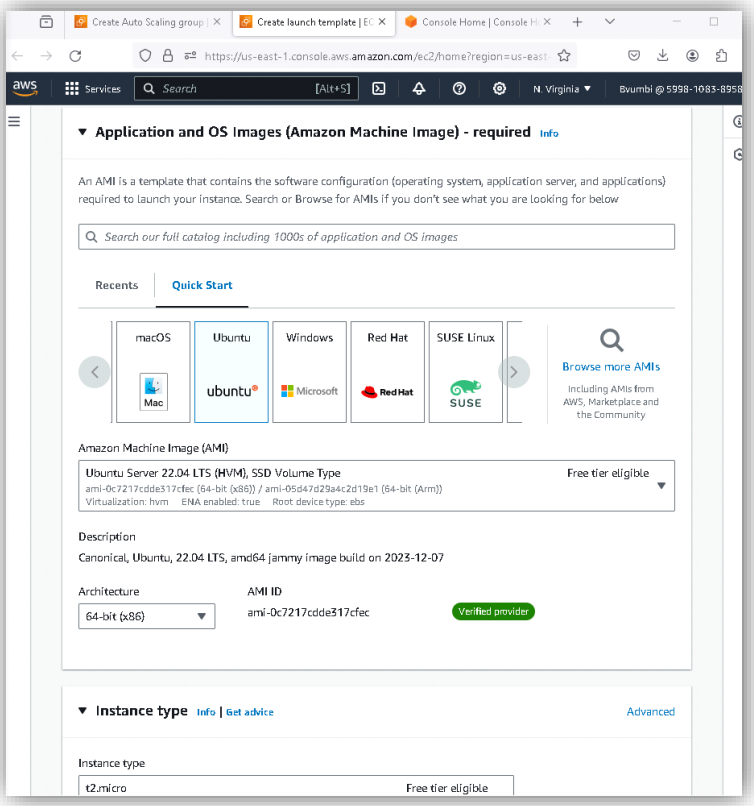
EC2 Auto Scaling Group Creation.

8.



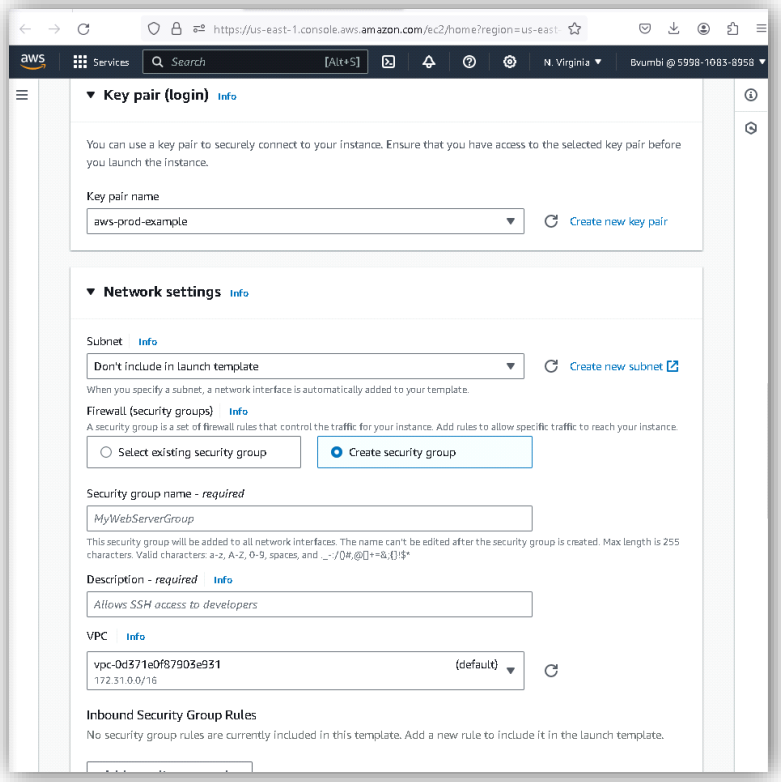
ASG Launch Template.

9



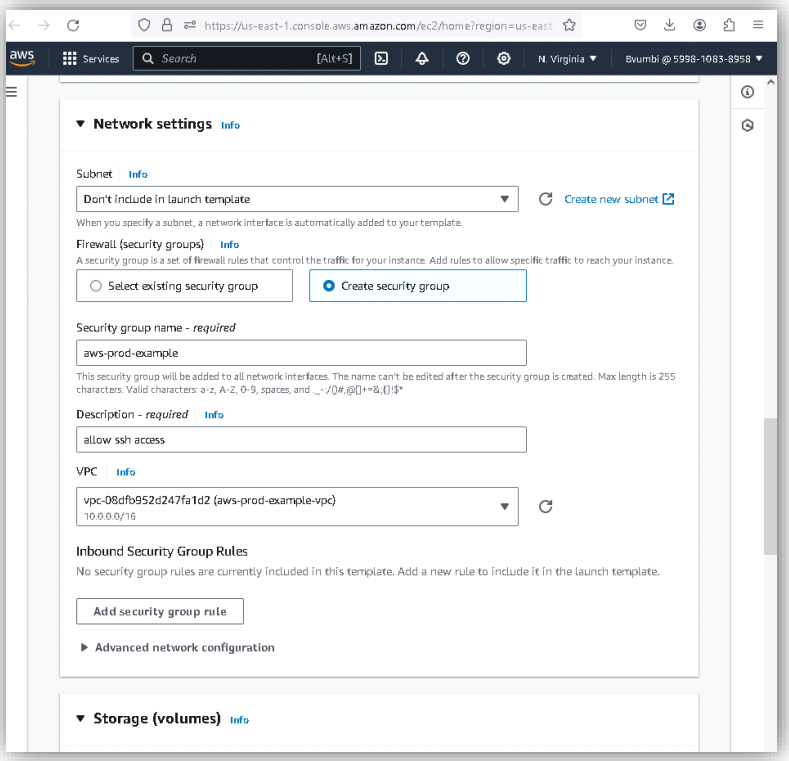
EC2 Instance Creation.

10.



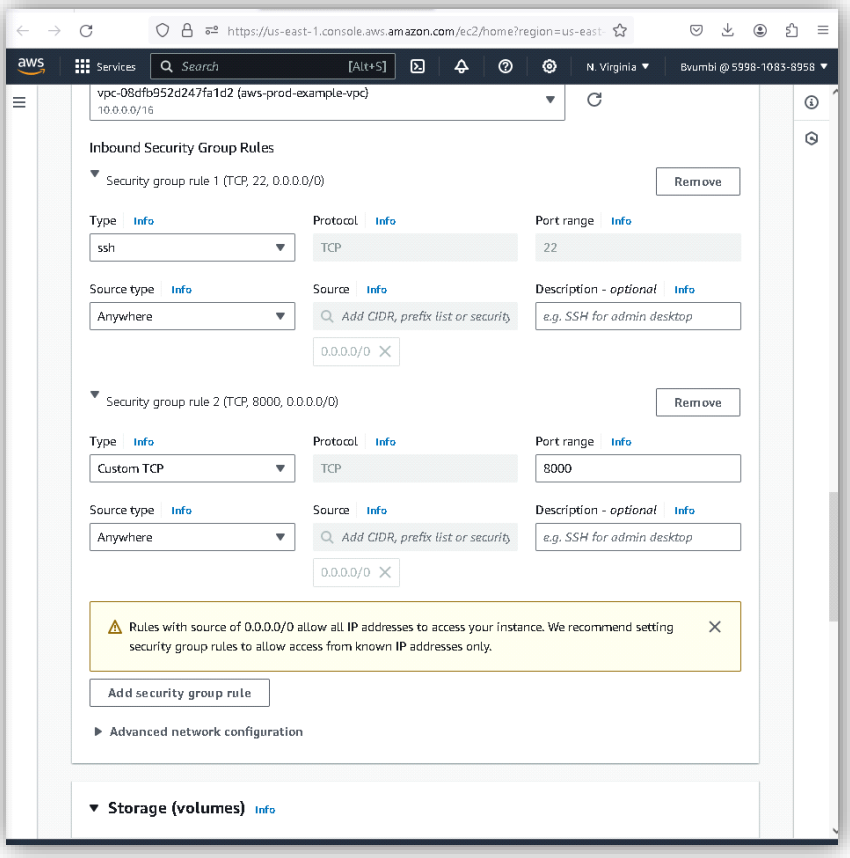
Key Pair Creation.

11.



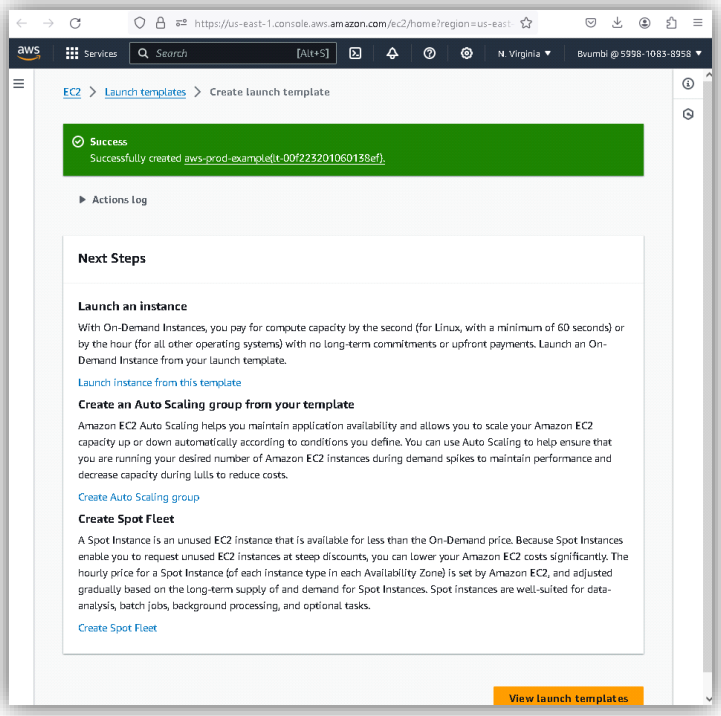
ASG Security Group Setup.

12.



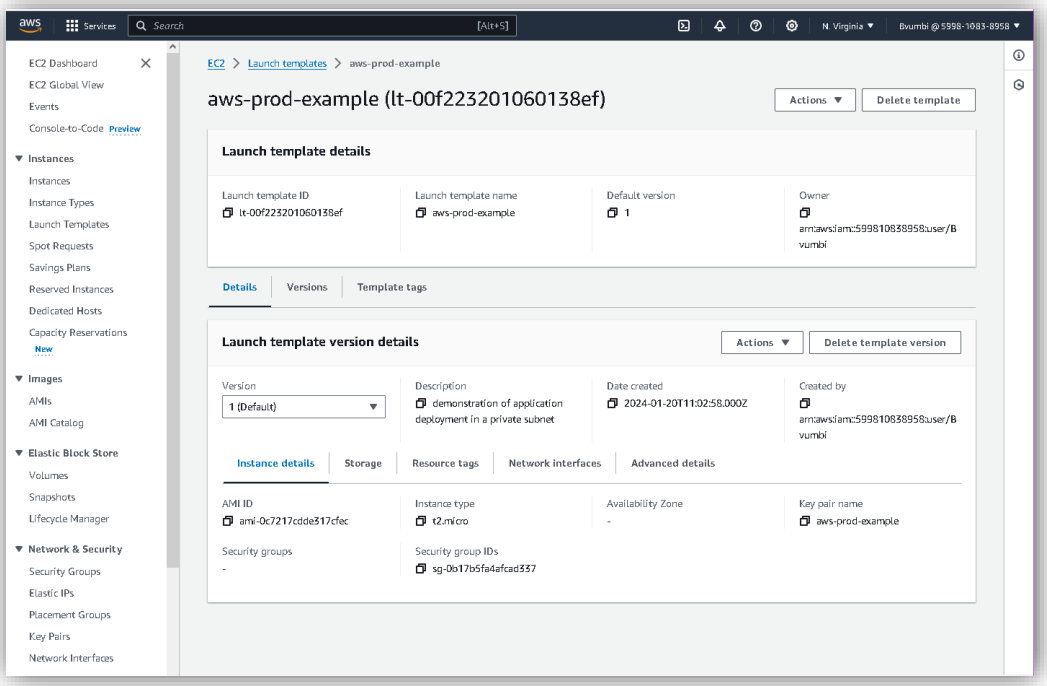
Configuring Security Group Inbound Rules (Firewall).

13.



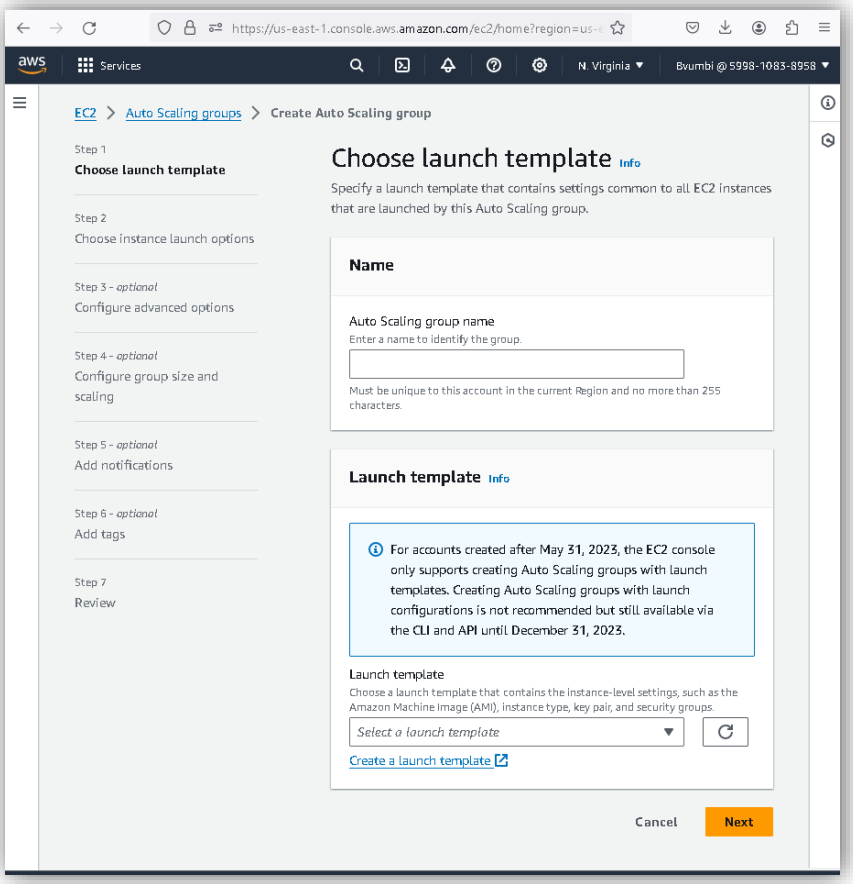
EC2 Launch Template Creation Success.

14.



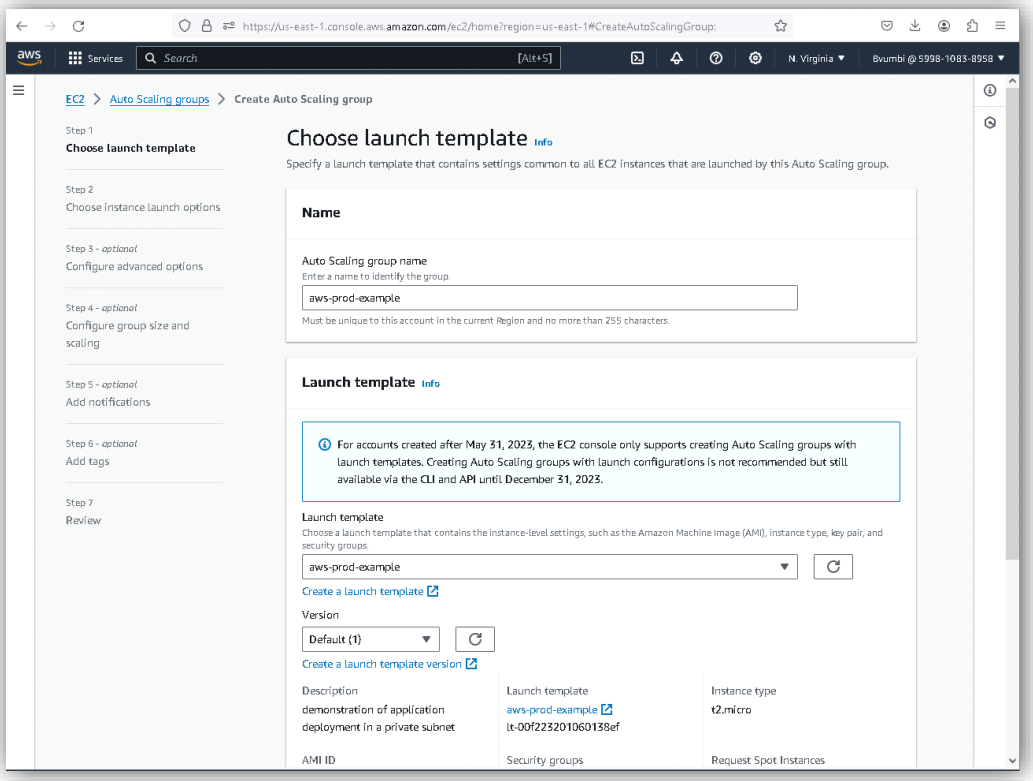
Launch Template Details.

15.



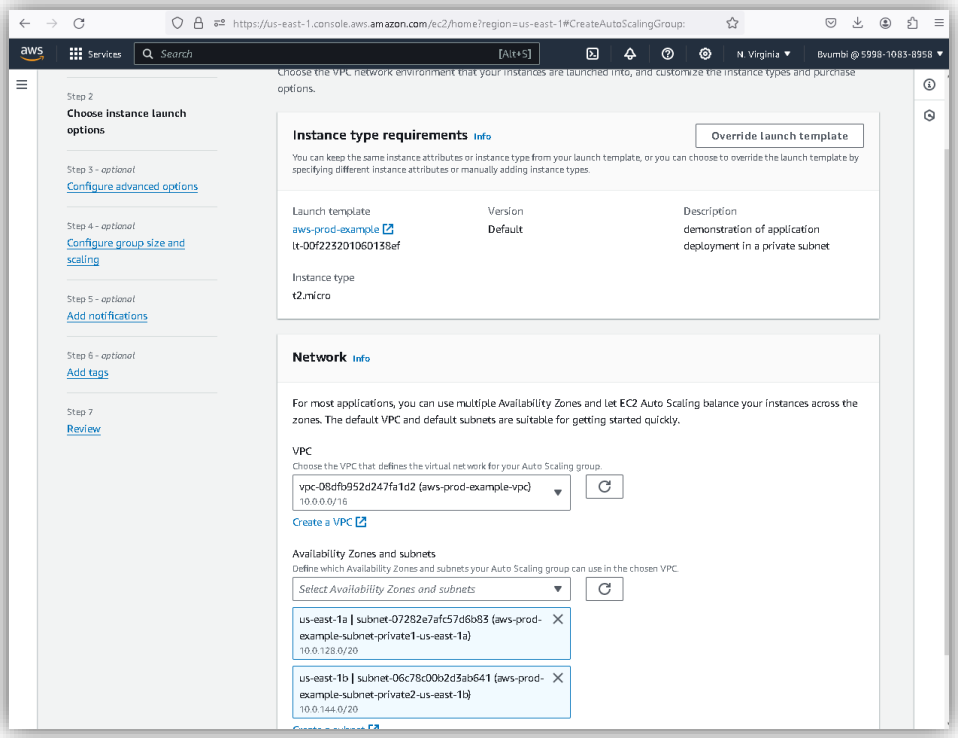
ASG Setup.

16.



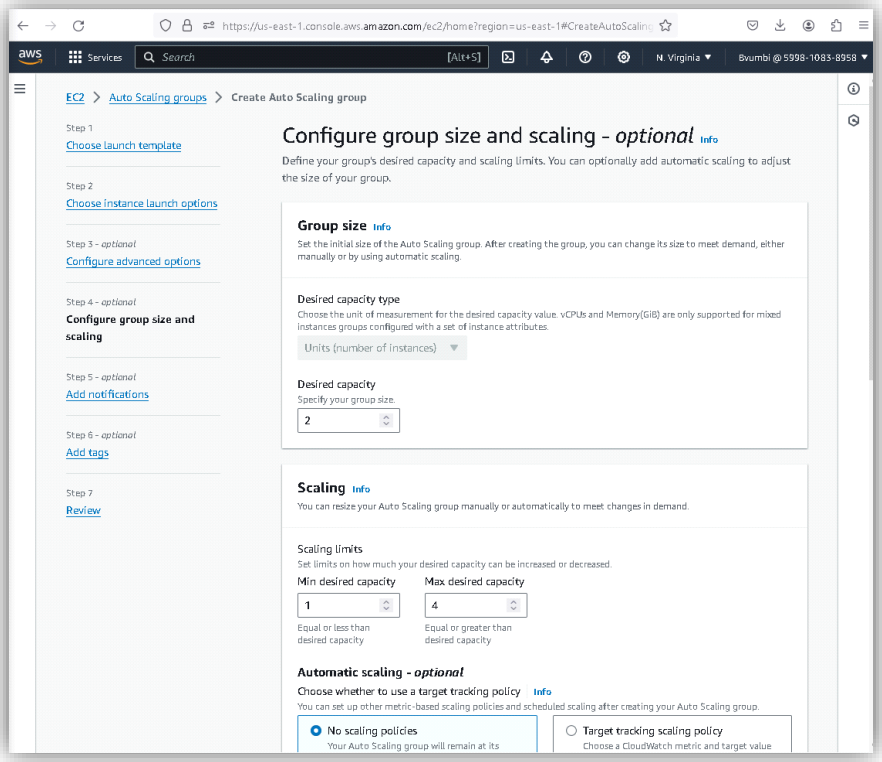
ASG Launch.

17.



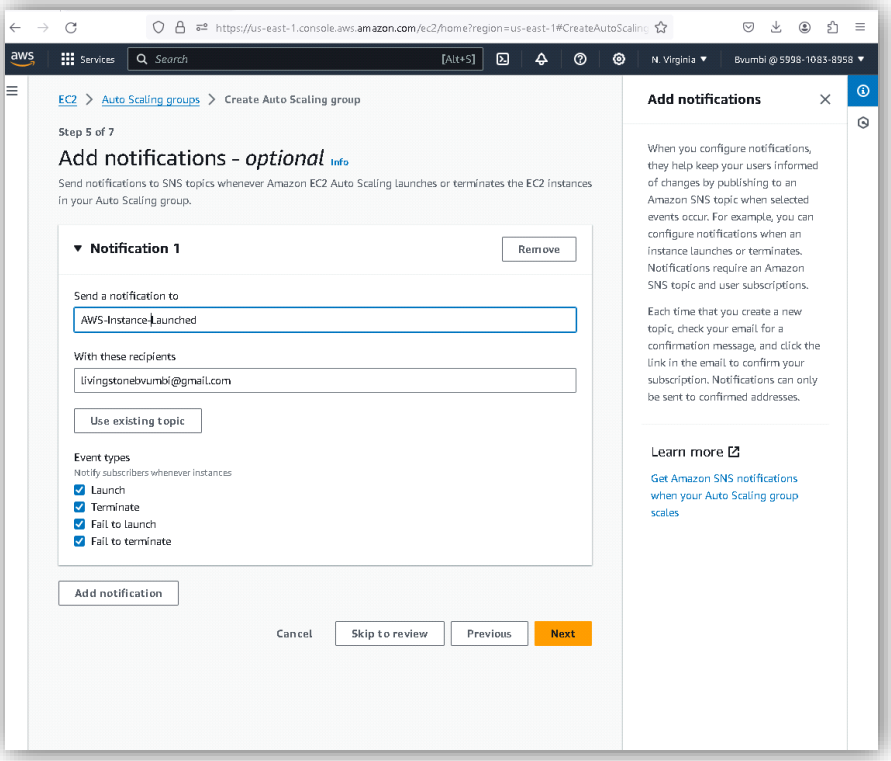
ASG Assigned to Private Subnets.

18.



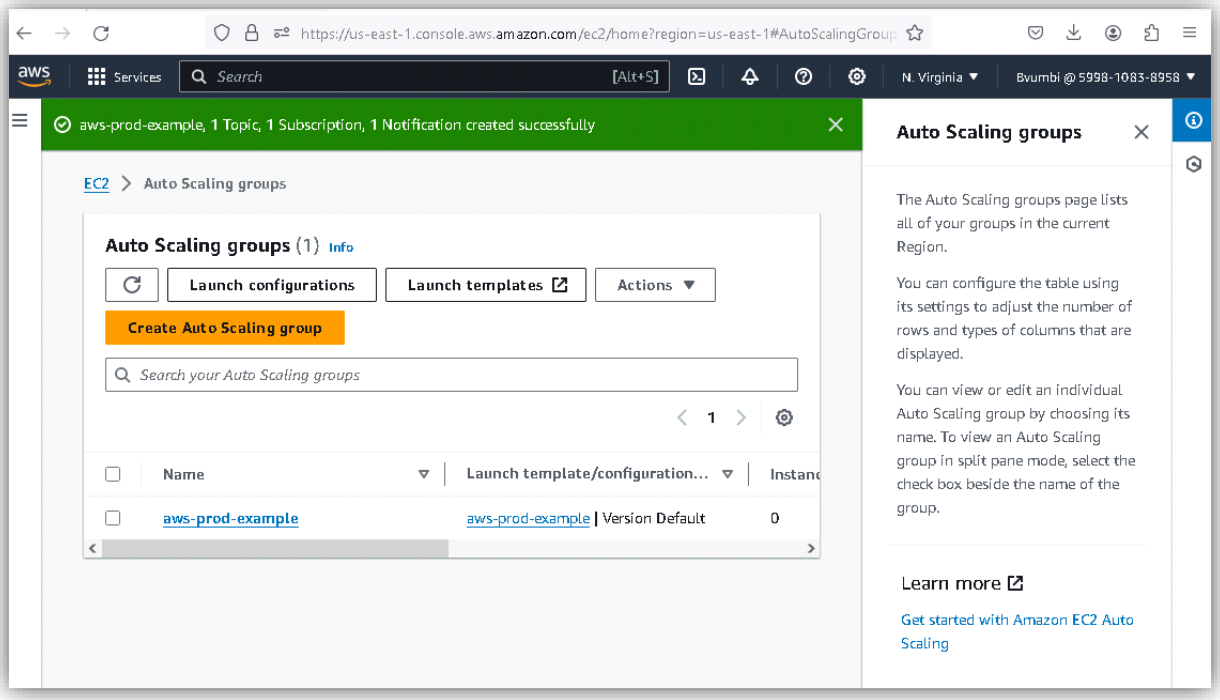
ASG Maximum and Minimum Limit Settings.

19.



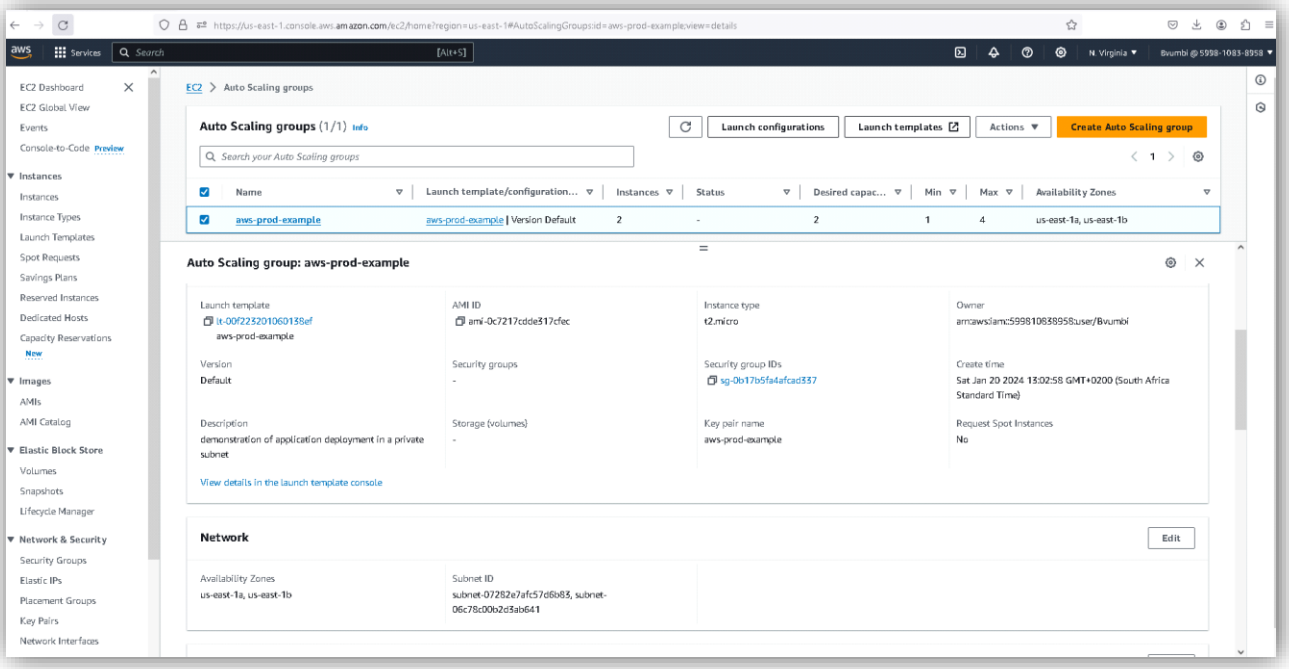
SNS Setup for EC2 Instances.

20.



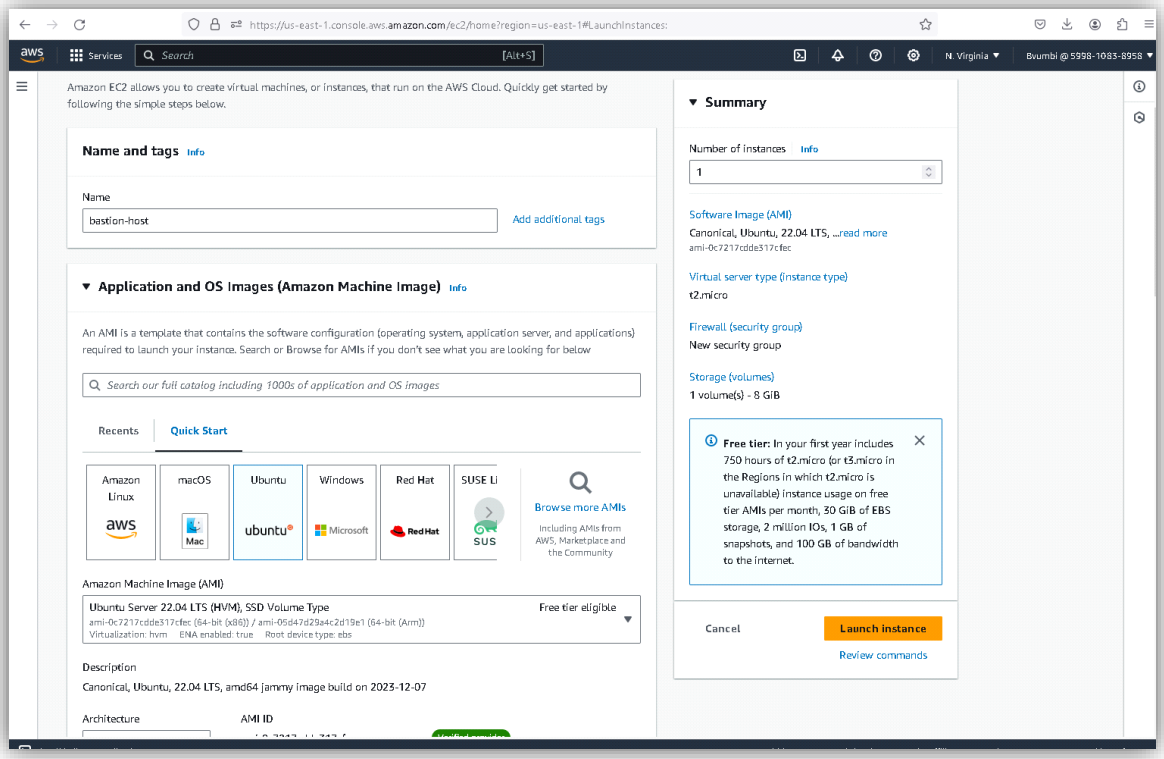
ASG Creation Success.

21.



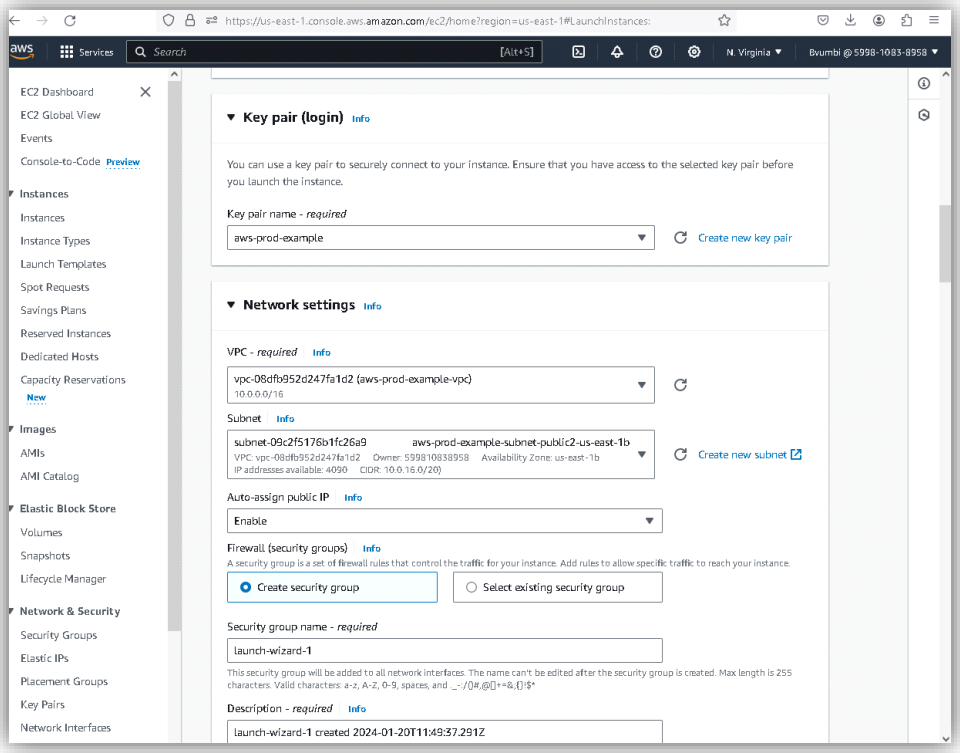
ASG EC2 Instances Running.

22.



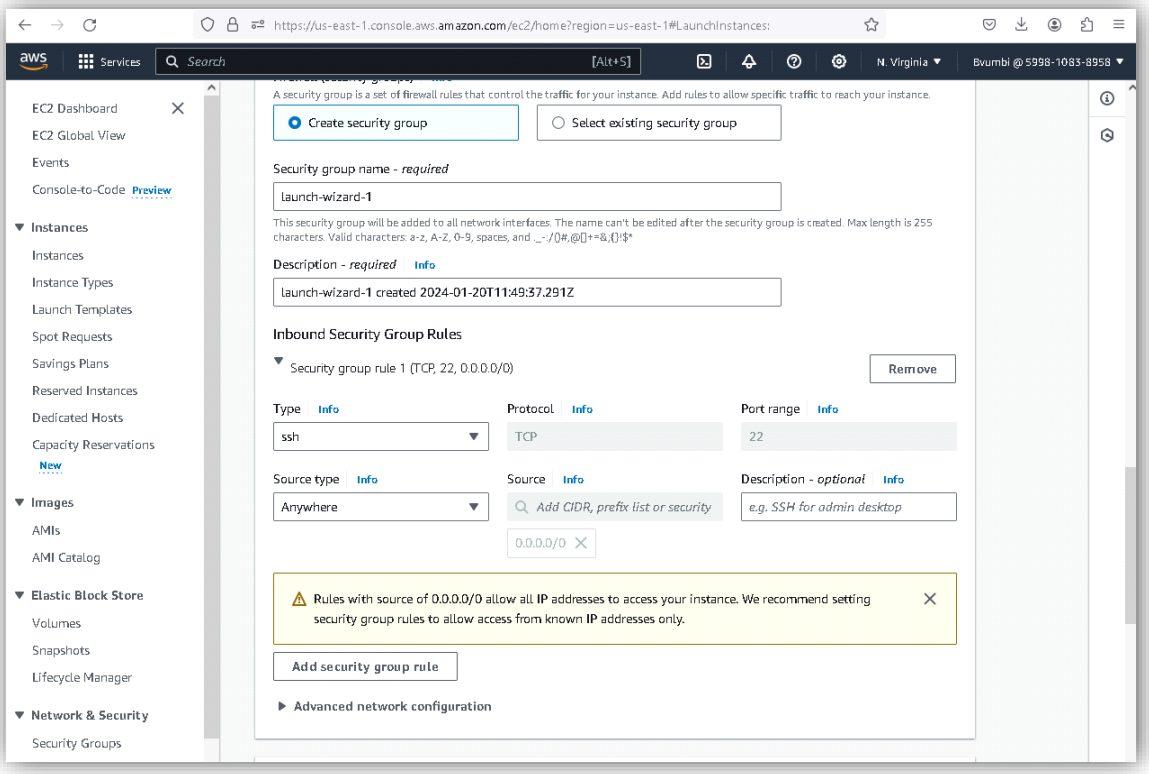
Bastion Host Creation.

23.



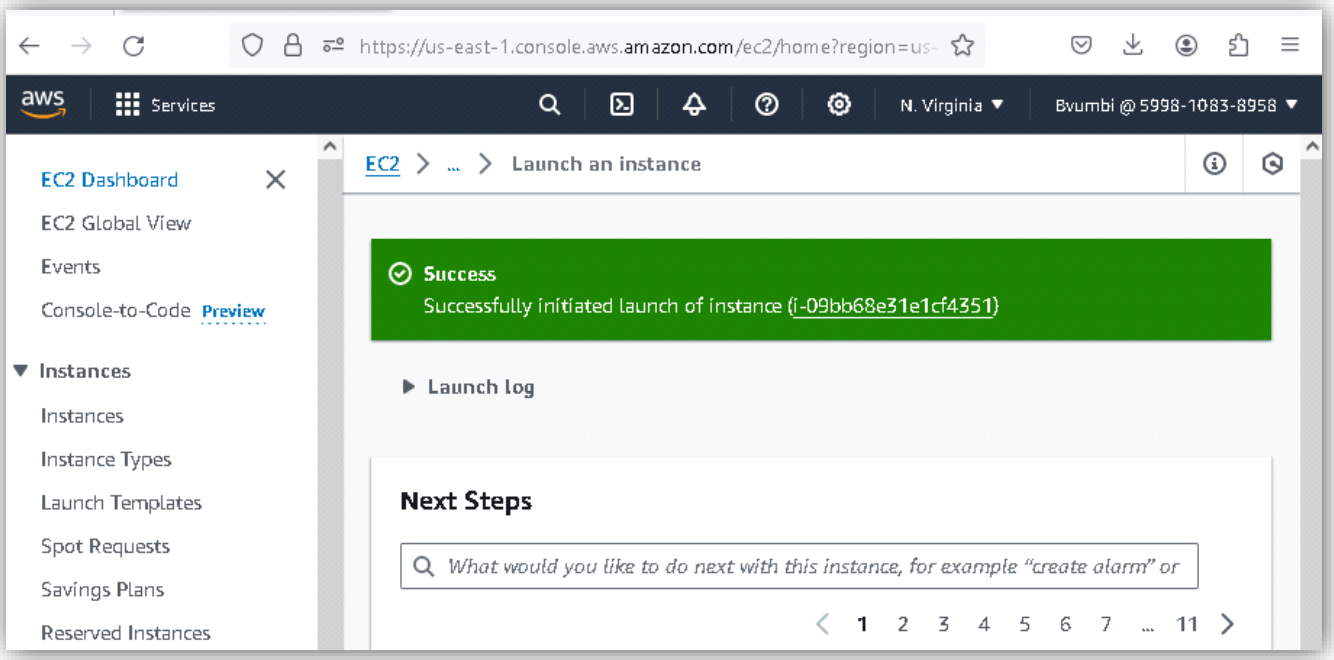
Bastion Host Public IP and Subnet Assigning.

24.



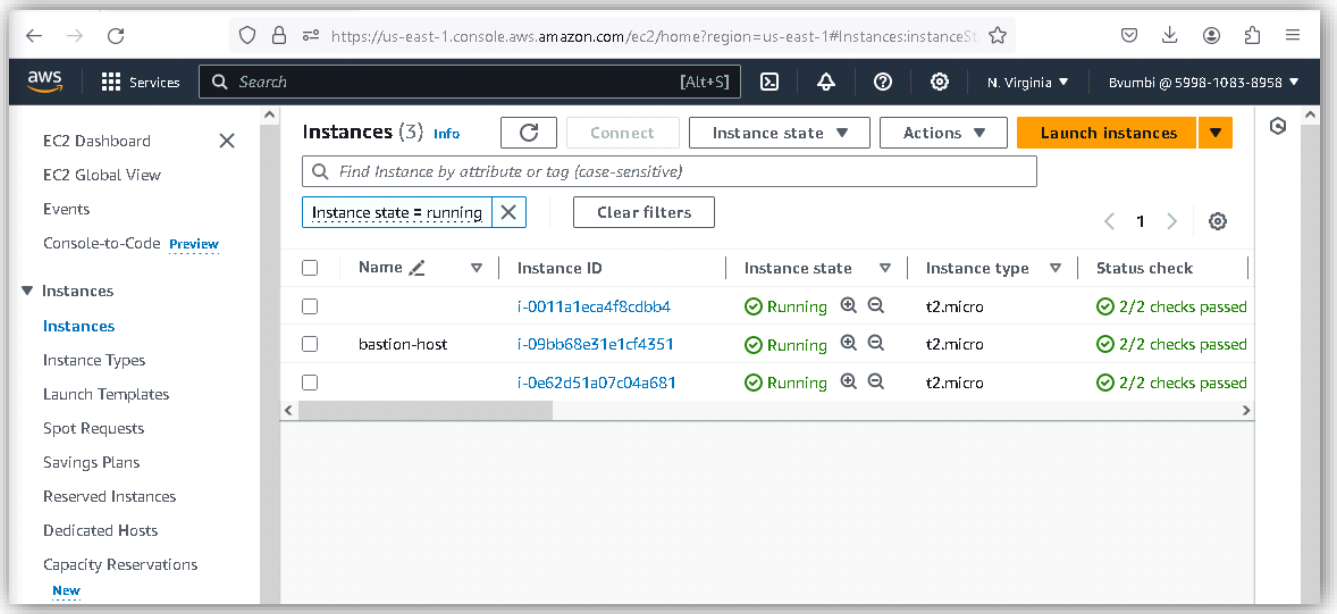
Bastion Host Firewall Setting.

25.



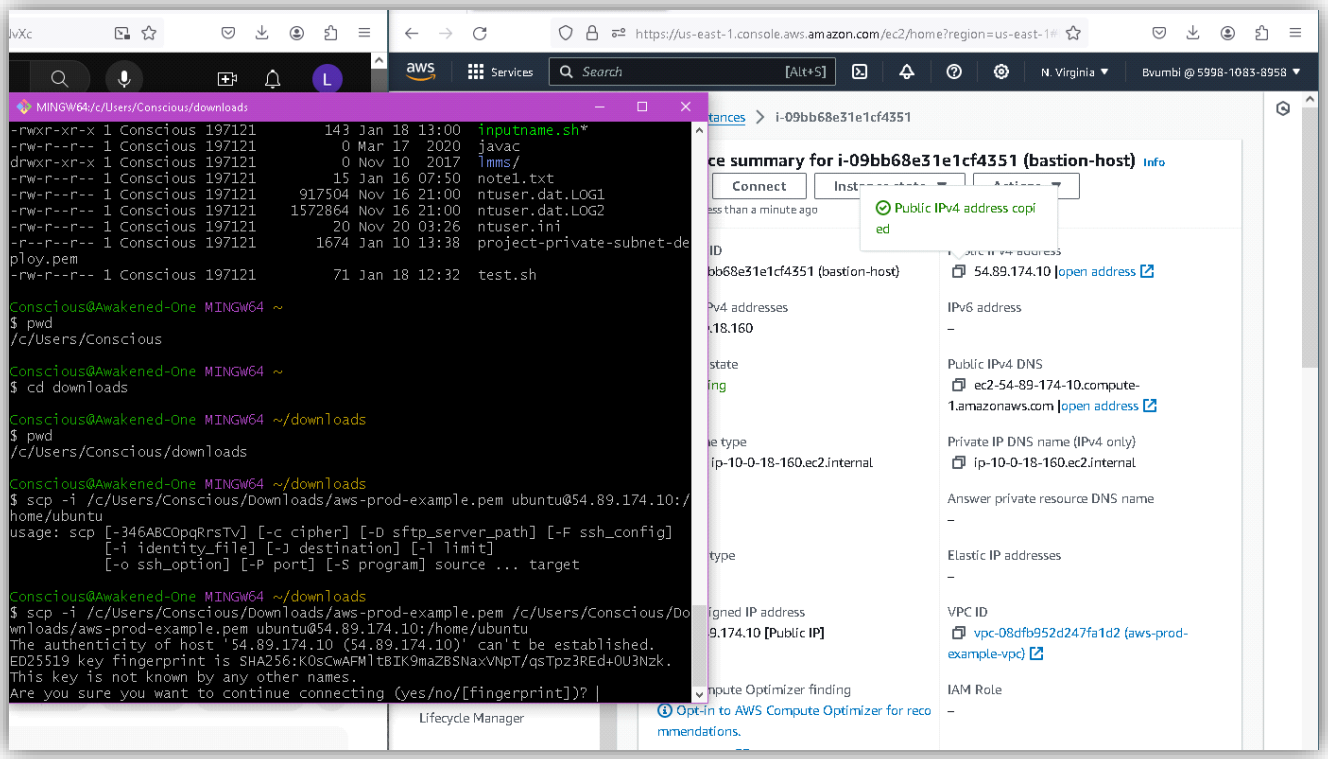
Bastion Host Creation Success.

26.



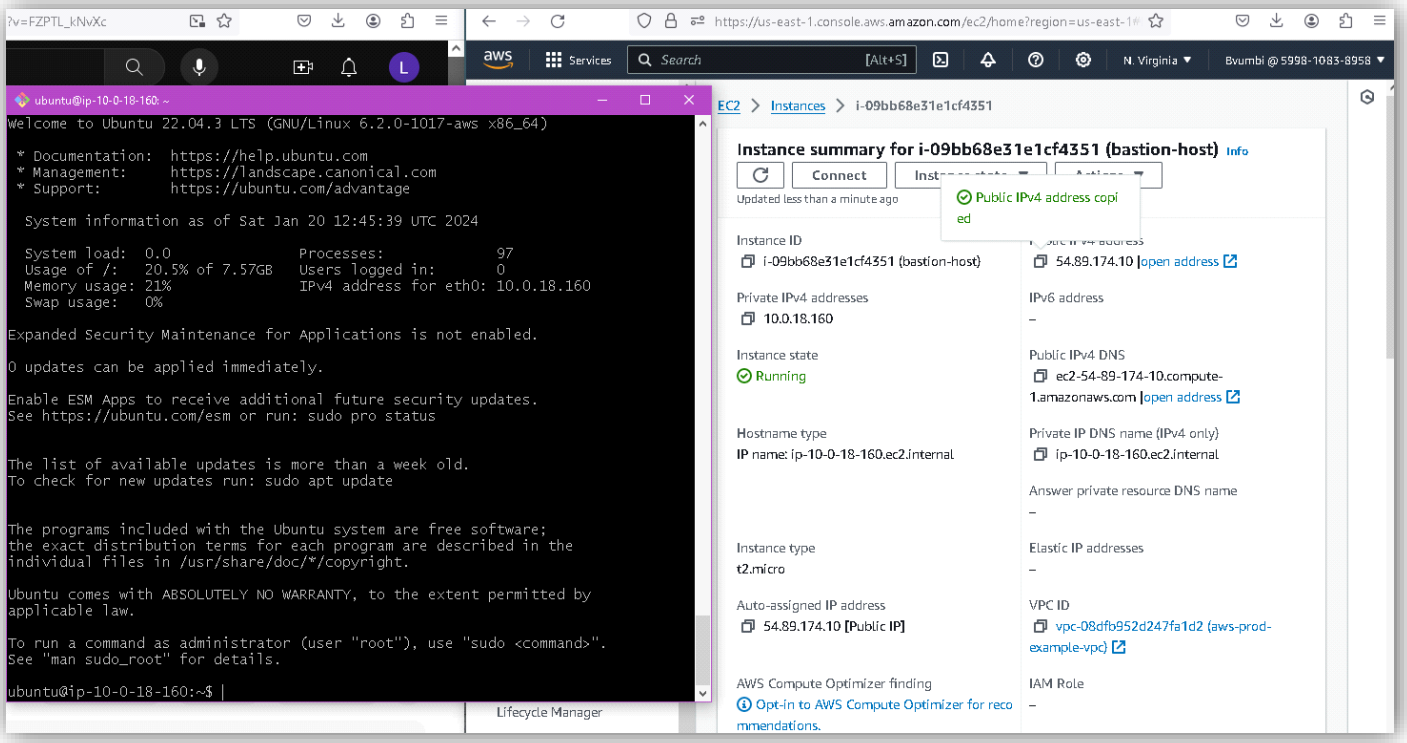
Bastion Host and Two EC2 Instances Running.

27.



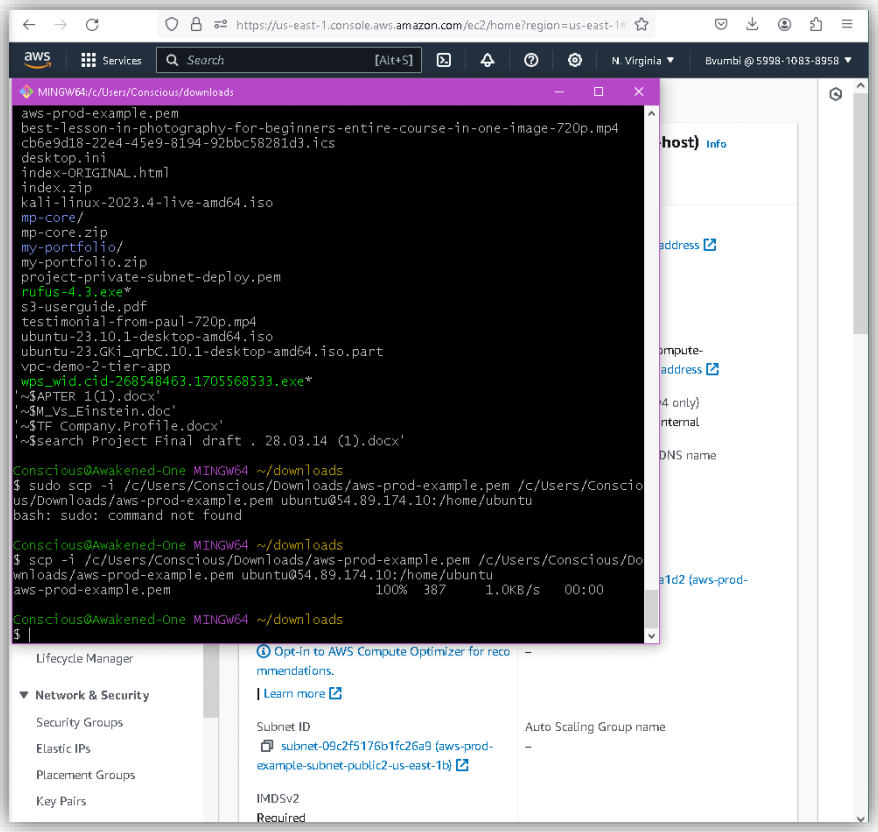
Copying Key Pair from Laptop to Bastion Host.

28.



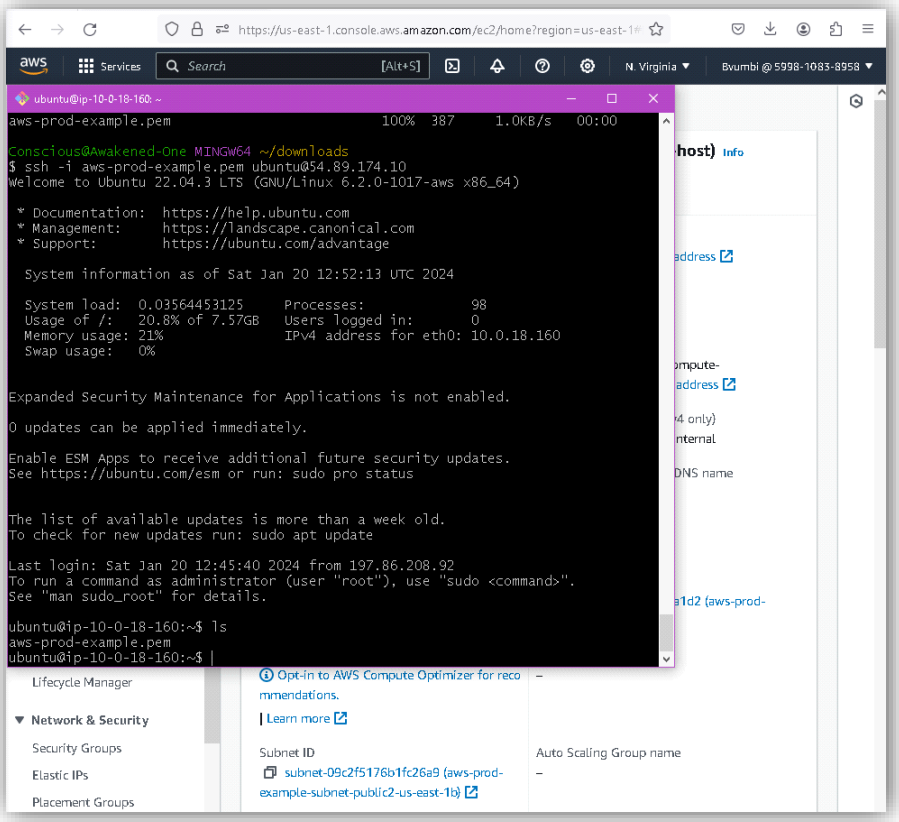
Logged In to Bastion Host from Laptop.

29.



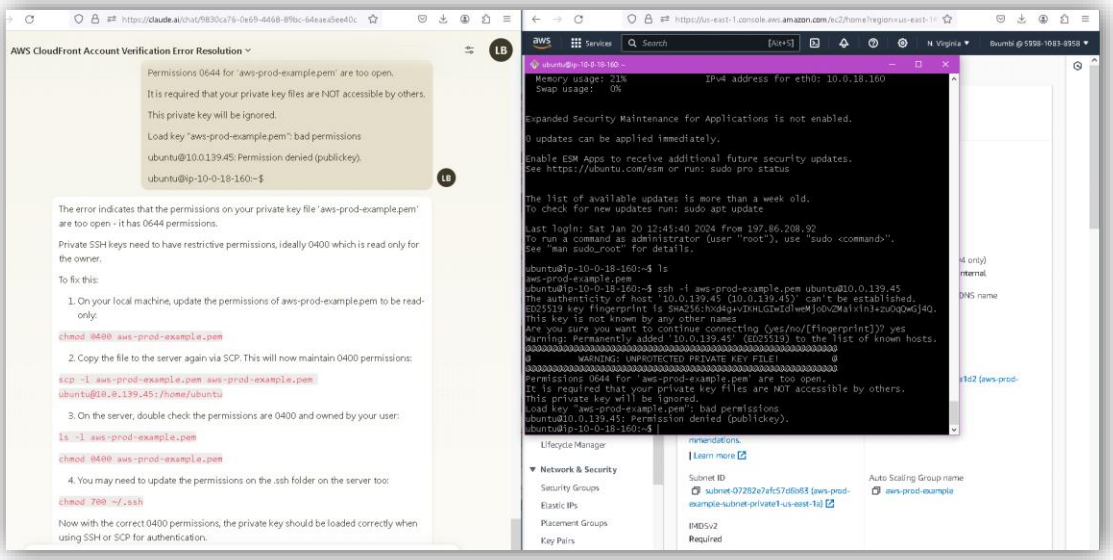
Key Pair Copying to Bastion Host Success.

30.



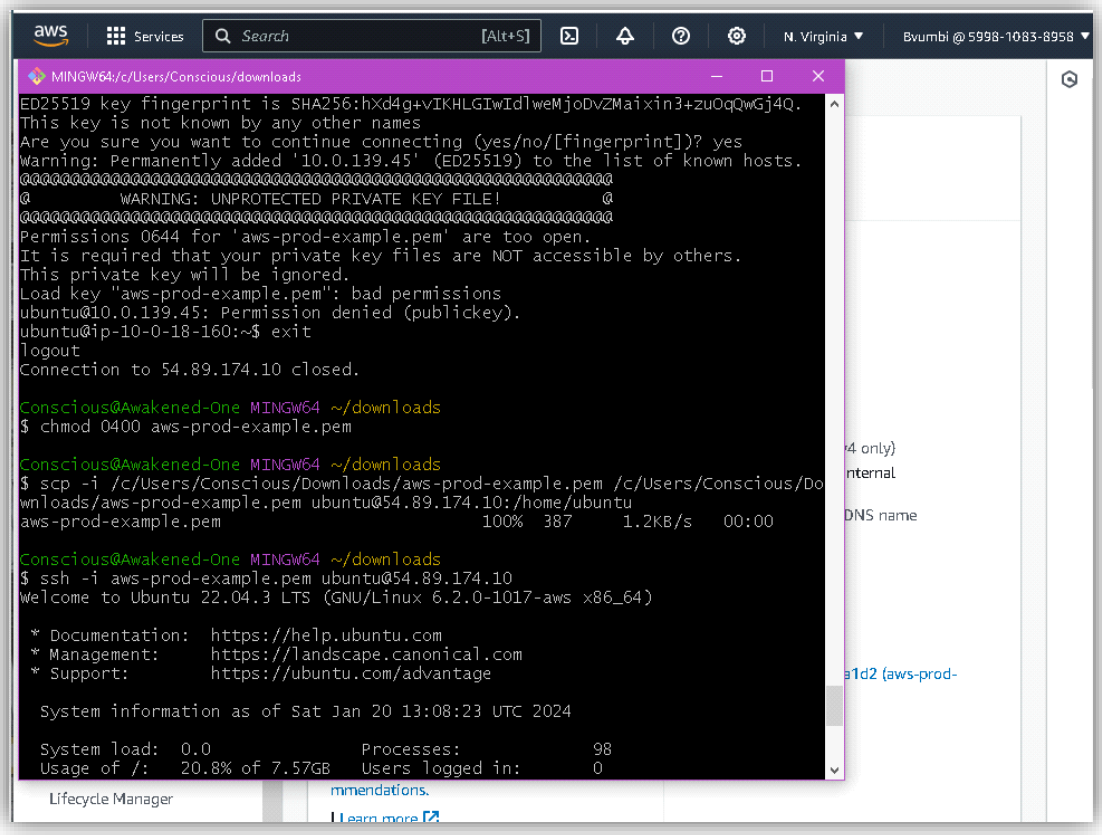
Confirmed Key Pair on Bastion Host.

31.



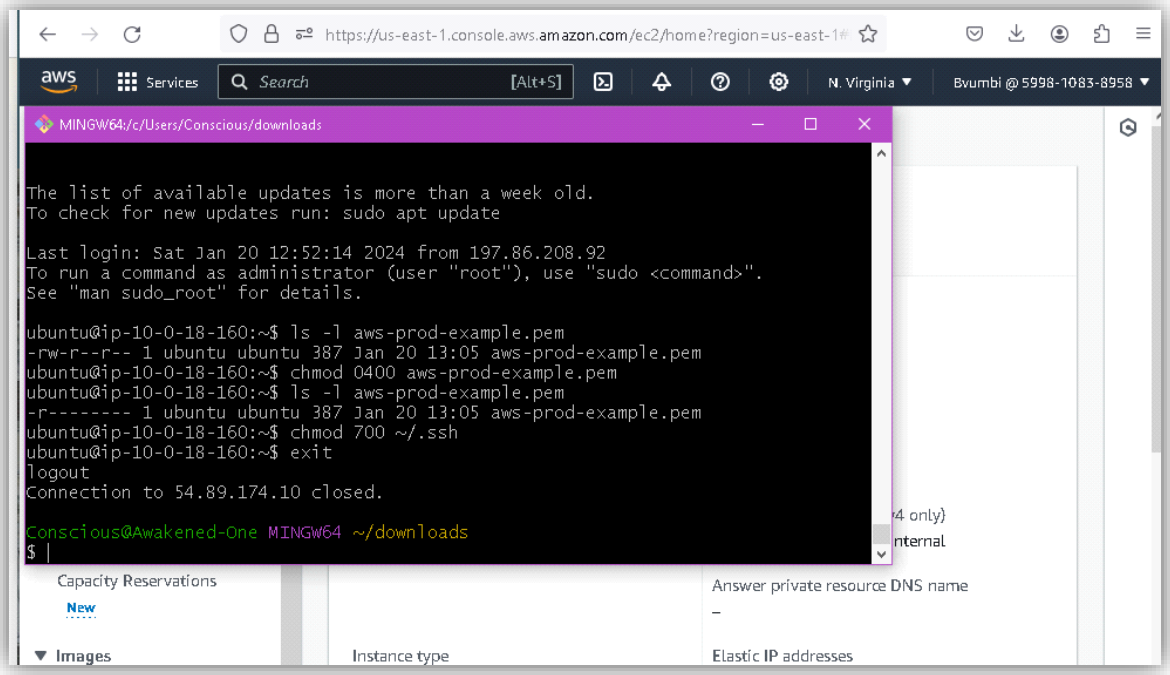
Public Key Permission Denied Error to Private EC2 server.

33.



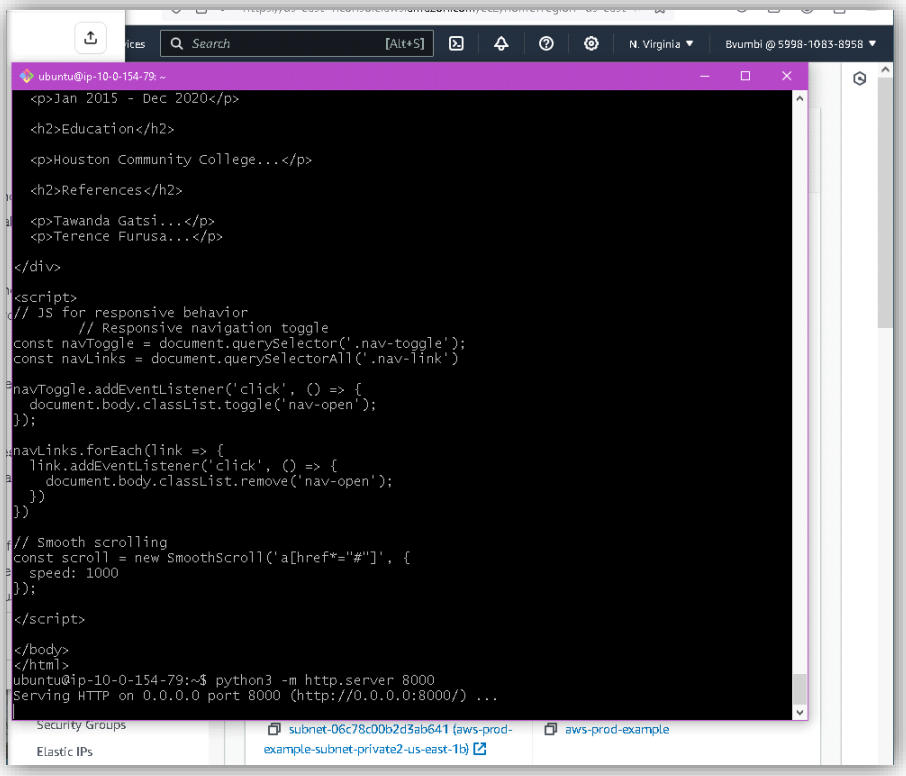
Changed Permissions of Key Pair to Read Only on Laptop.

34.



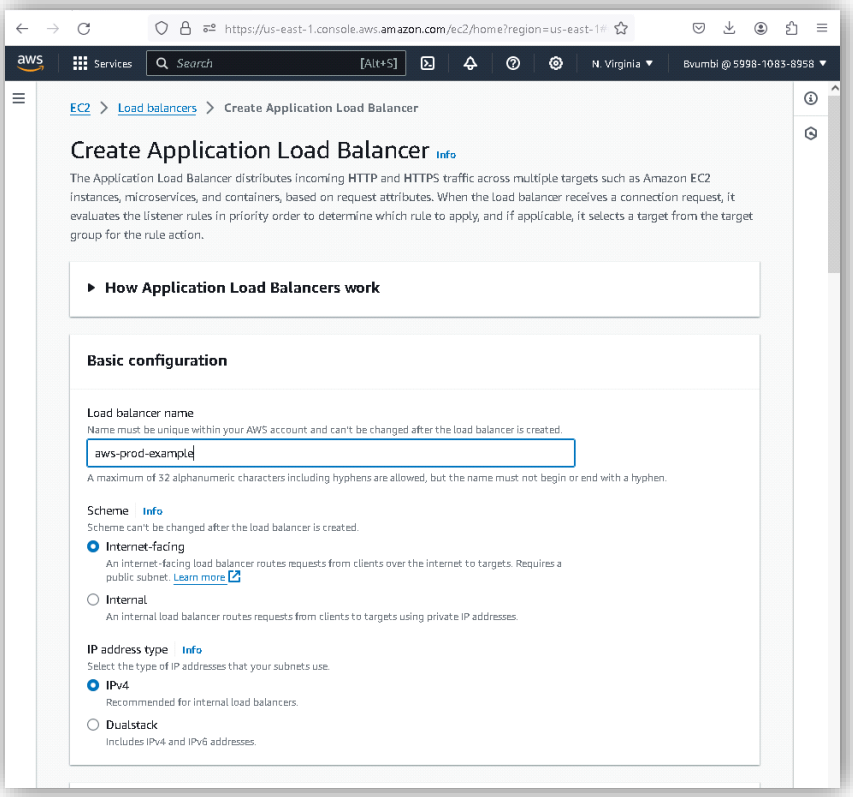
Changed Permissions of Key Pair to Read Only on Bastion Host.

35.



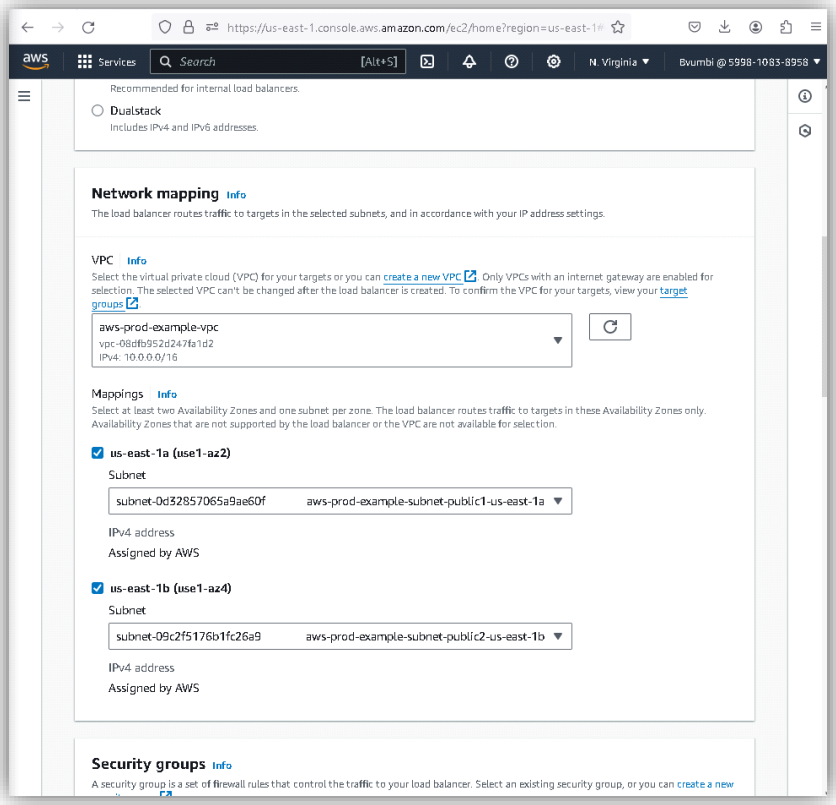
Starting Web Server on Private EC2 instance.

36.



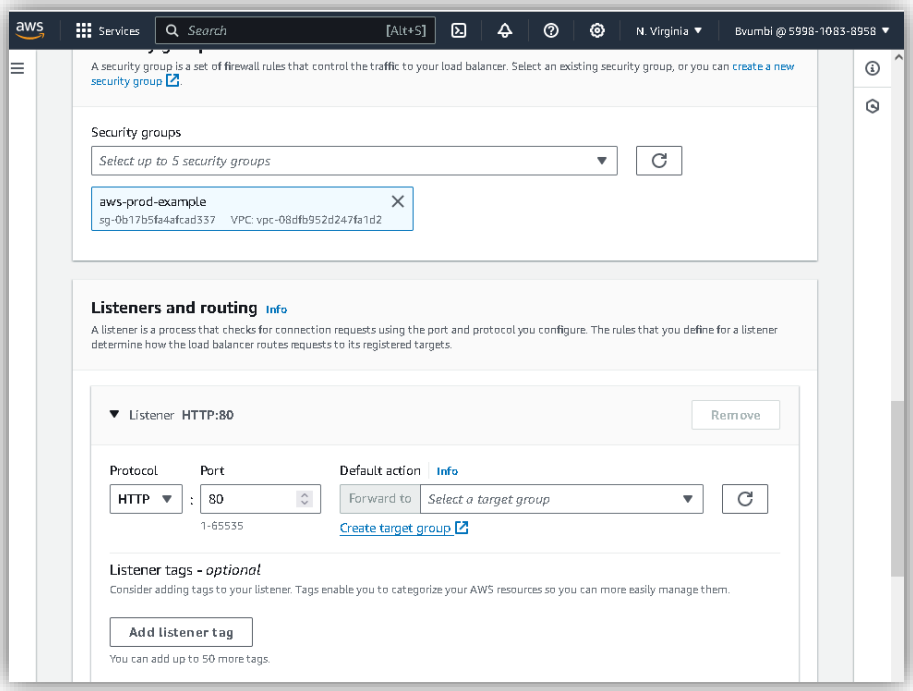
Creating Application Load Balancer (ALB).

37.



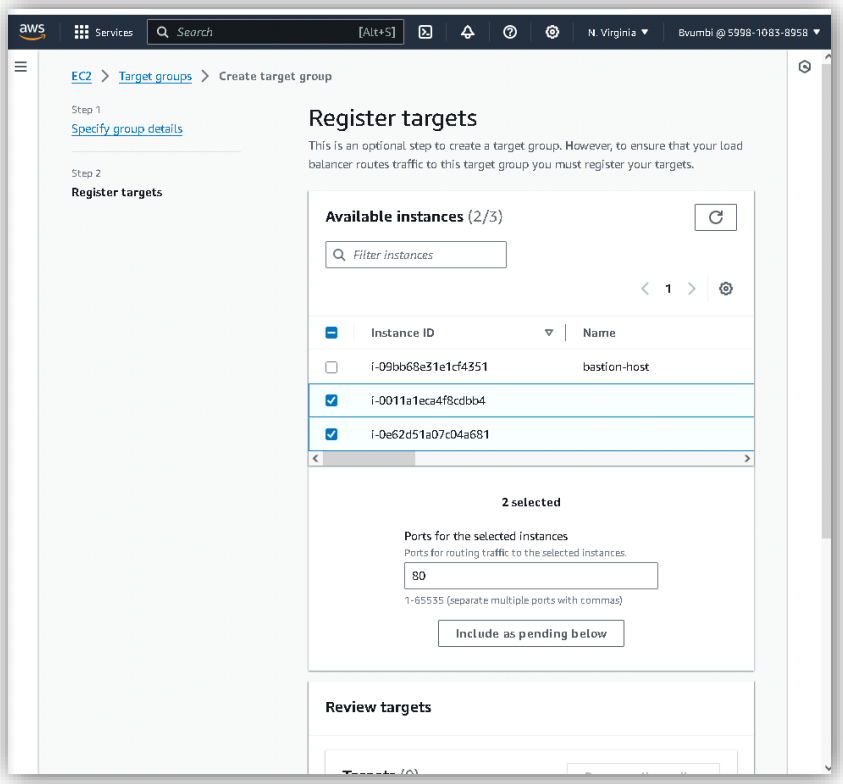
ALB VPC and Public Subnet Settings.

38.



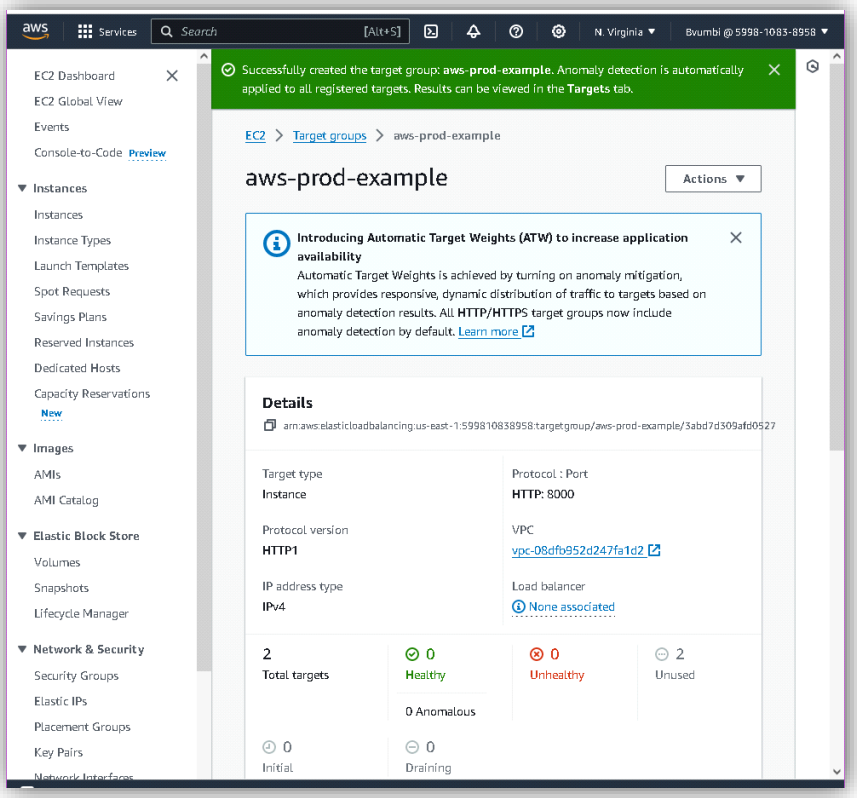
Adding Security Group to ALB.

39.



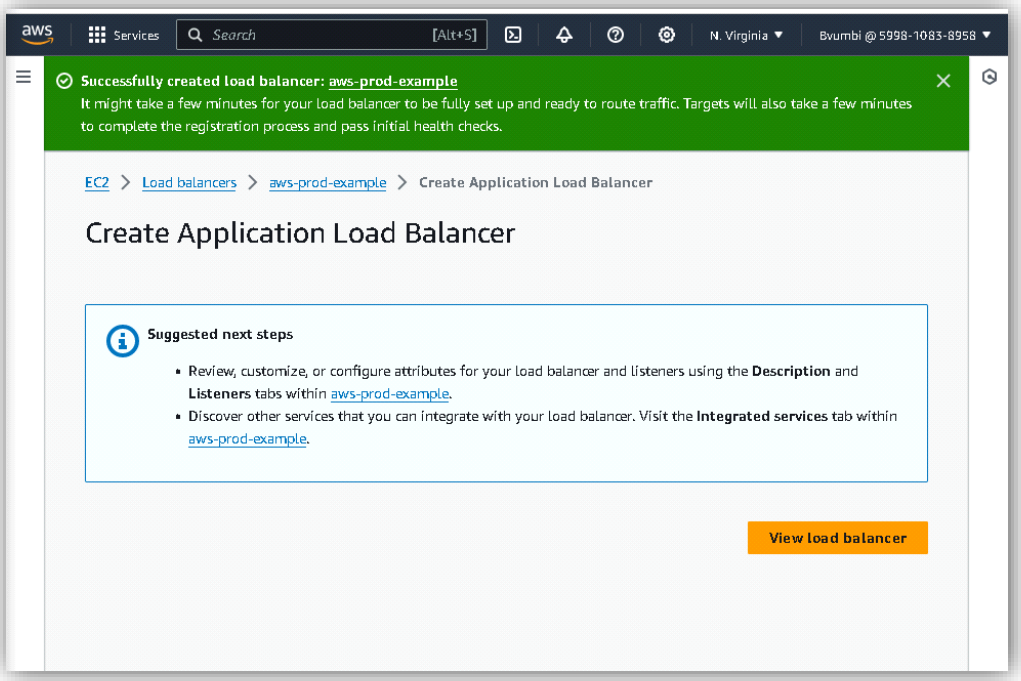
Adding EC2 instances to Target Group for Load Balancer.

40.



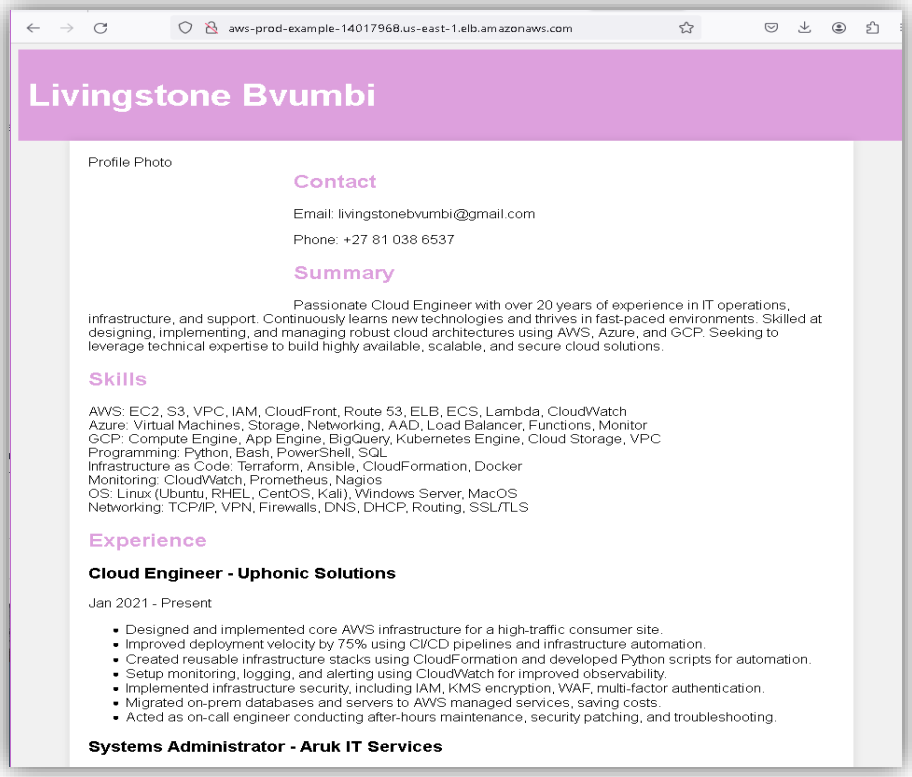
Target Group Setup Success.

41.



ALB Setup Success.

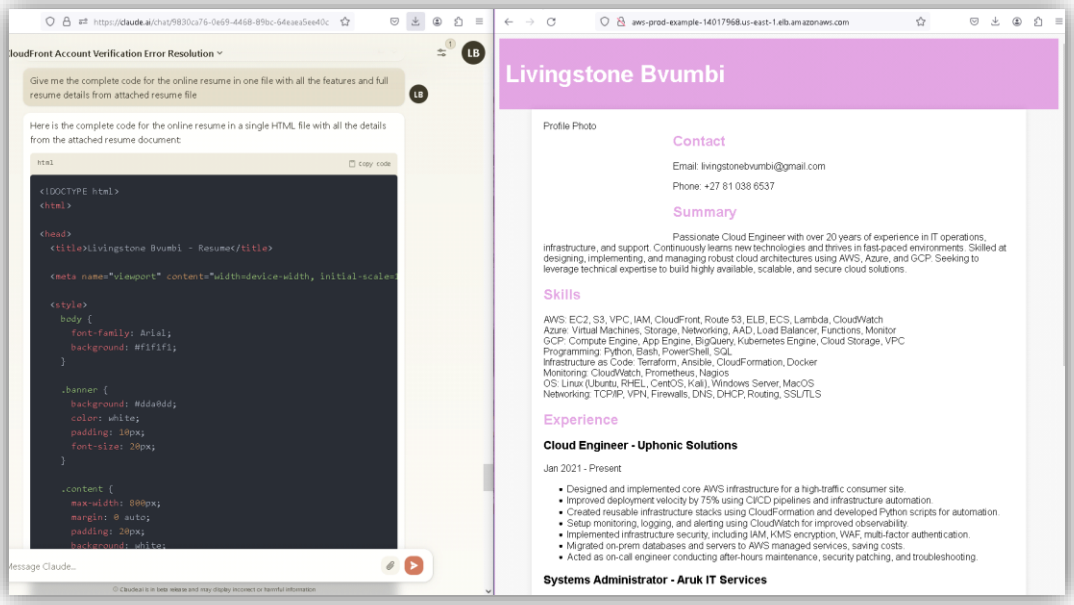
42.



Online Resume hosted on Private Web Server Displaying in Web Browser Proving Successful Setup of Infrastructure.

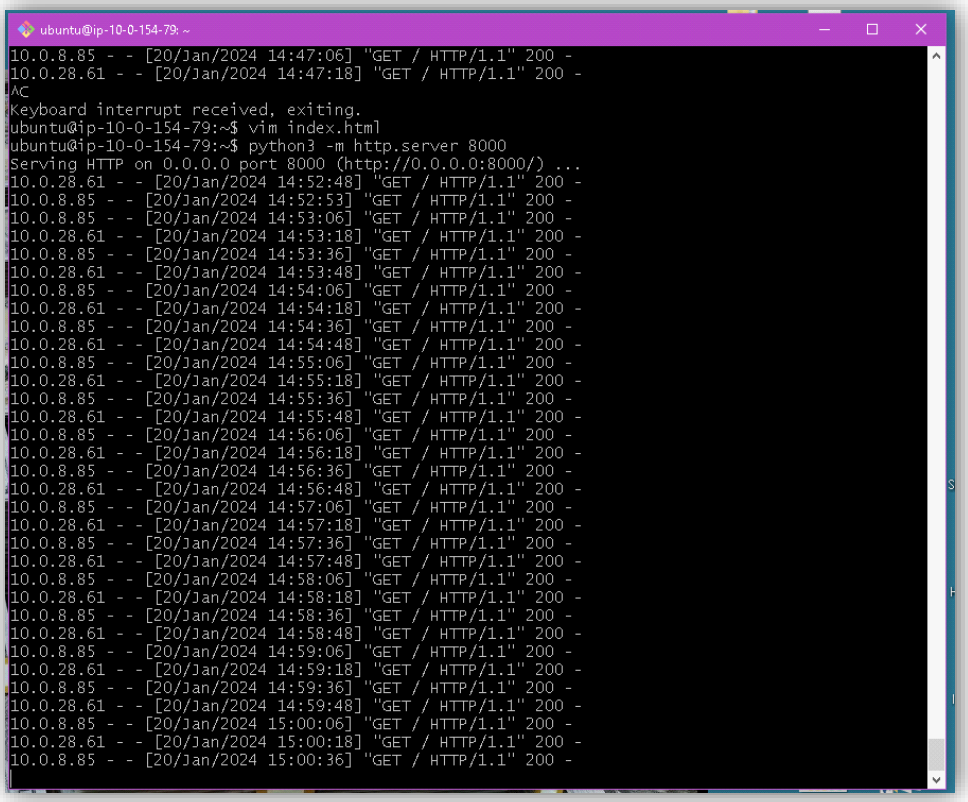
How website files where created and uploaded to the web servers:

43.



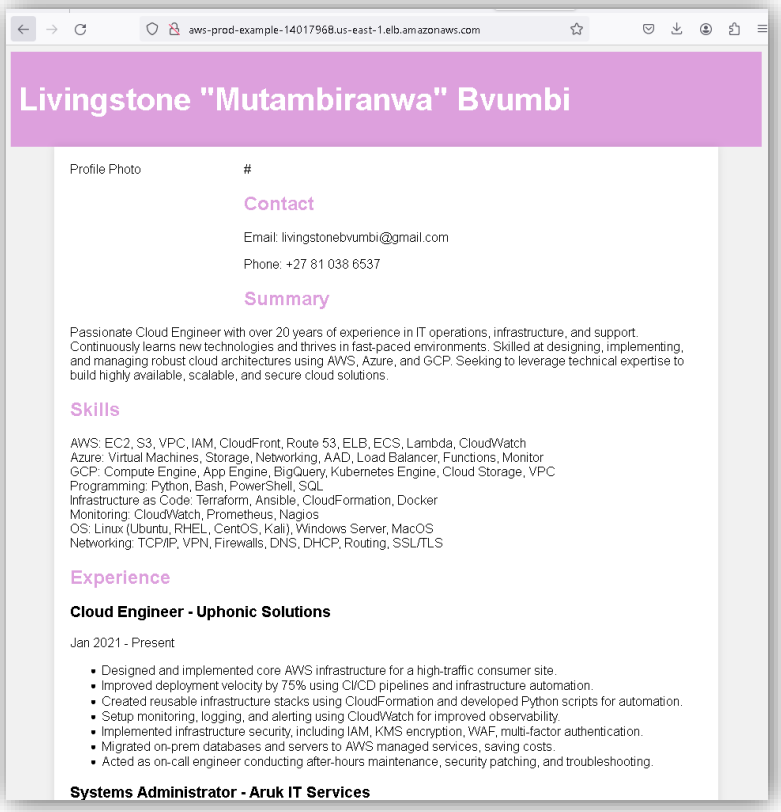
AI Prompts used to generate web files, index.html.

44.



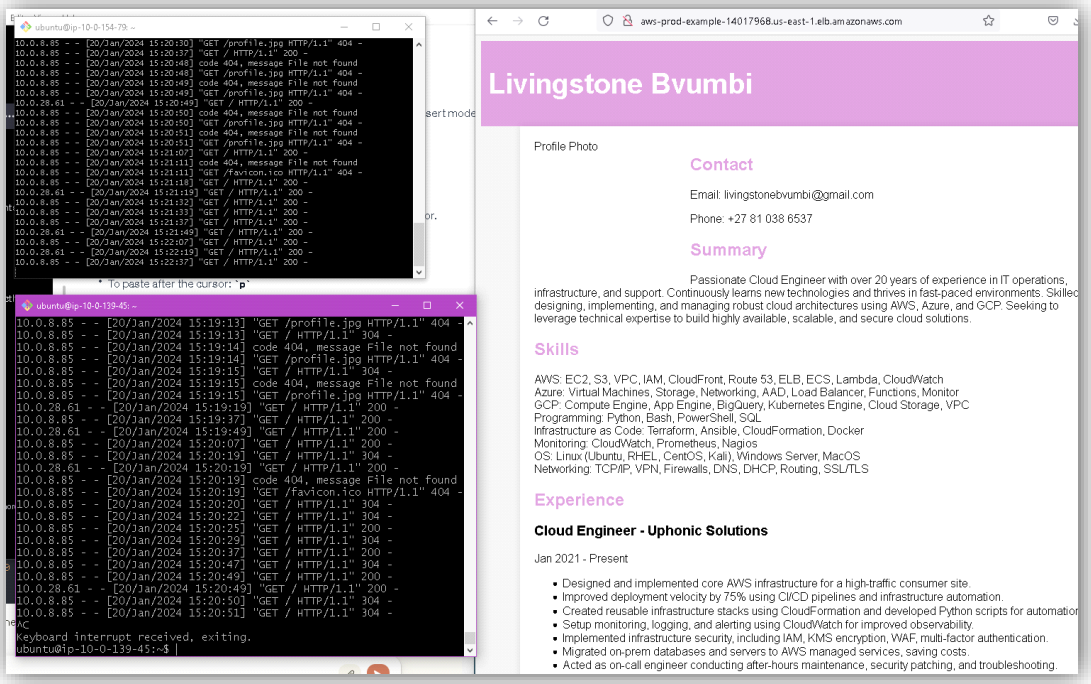
Creating and editing html file on Ubuntu virtual machine using vim editor and starting web server.

45.



Online Resume displaying on web server running on second private EC2 to demonstrate load-balancing, and multi-AZ feature of the distributed system architecture.

46.



ssh connections to two private EC2 web servers.