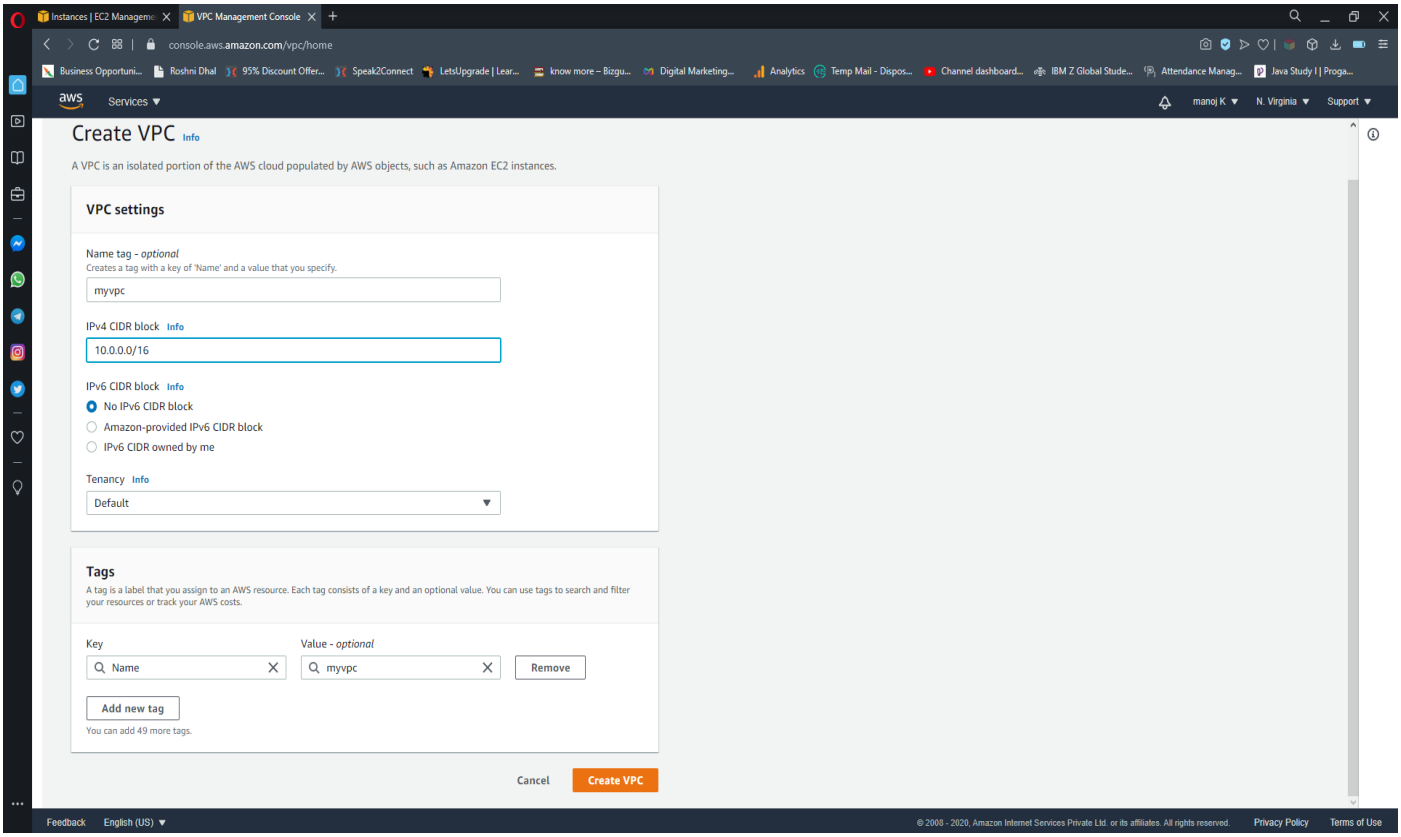
AWS Final Project 1

Deployment of a Highly Available Web Application

-Manoj K

Started with creating a VPC:

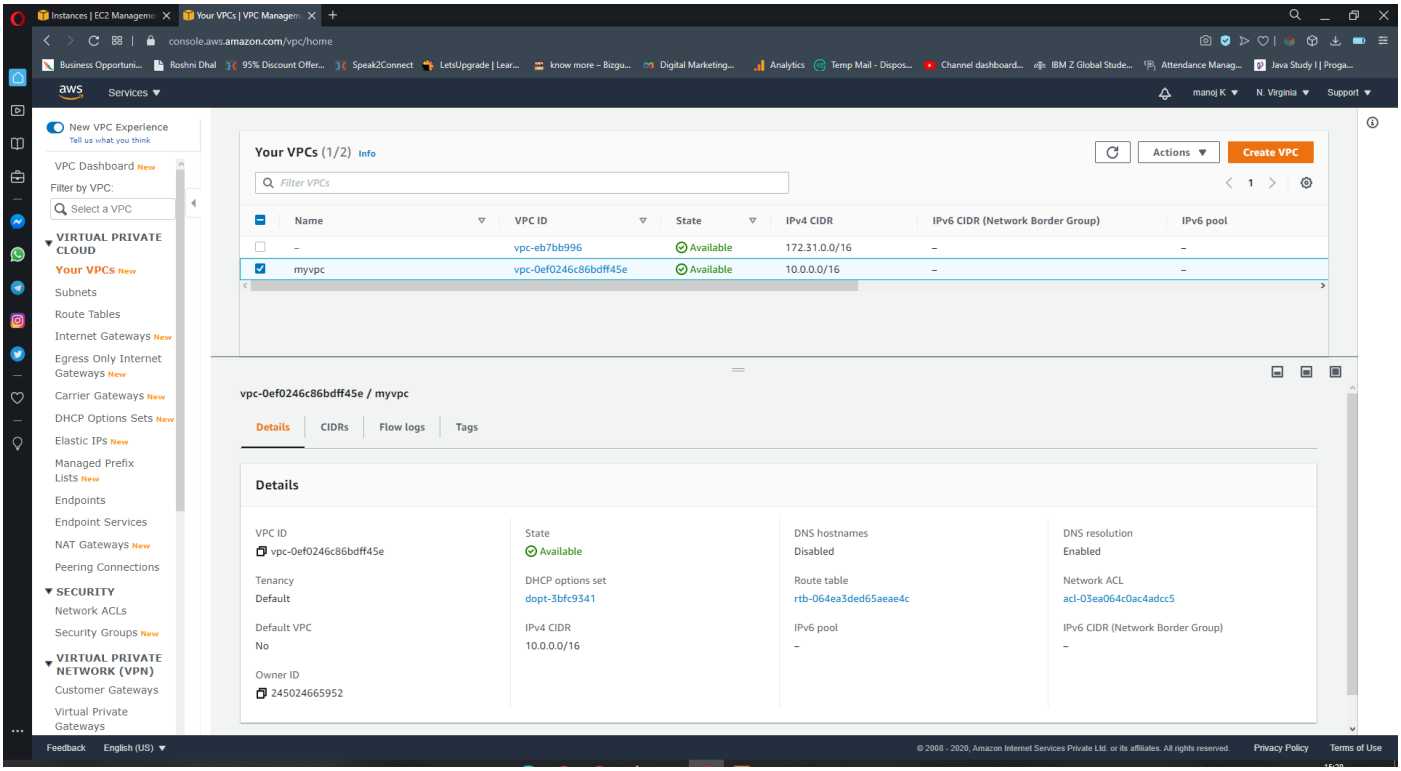


VPC Name: myvpc

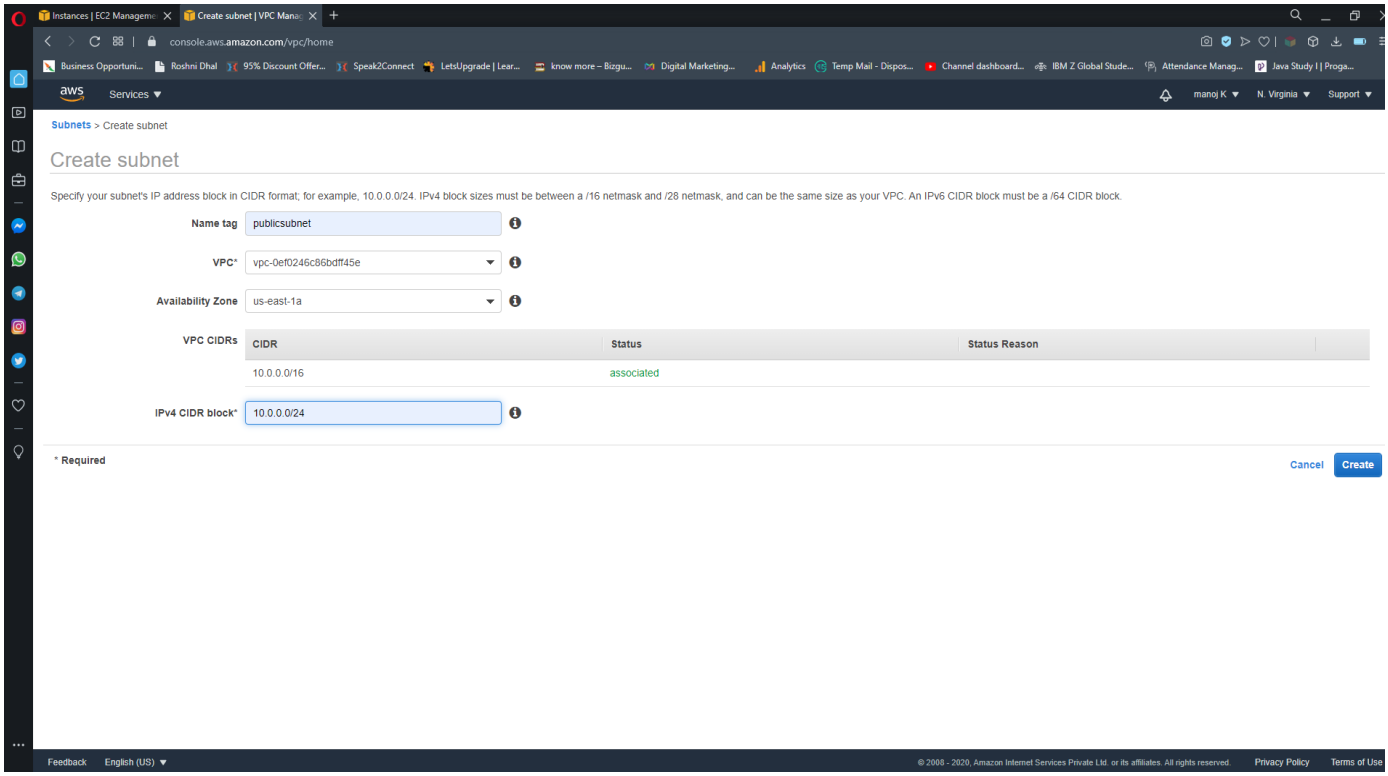
CIDR block:10.0.0.0/16

Tenancy: Default

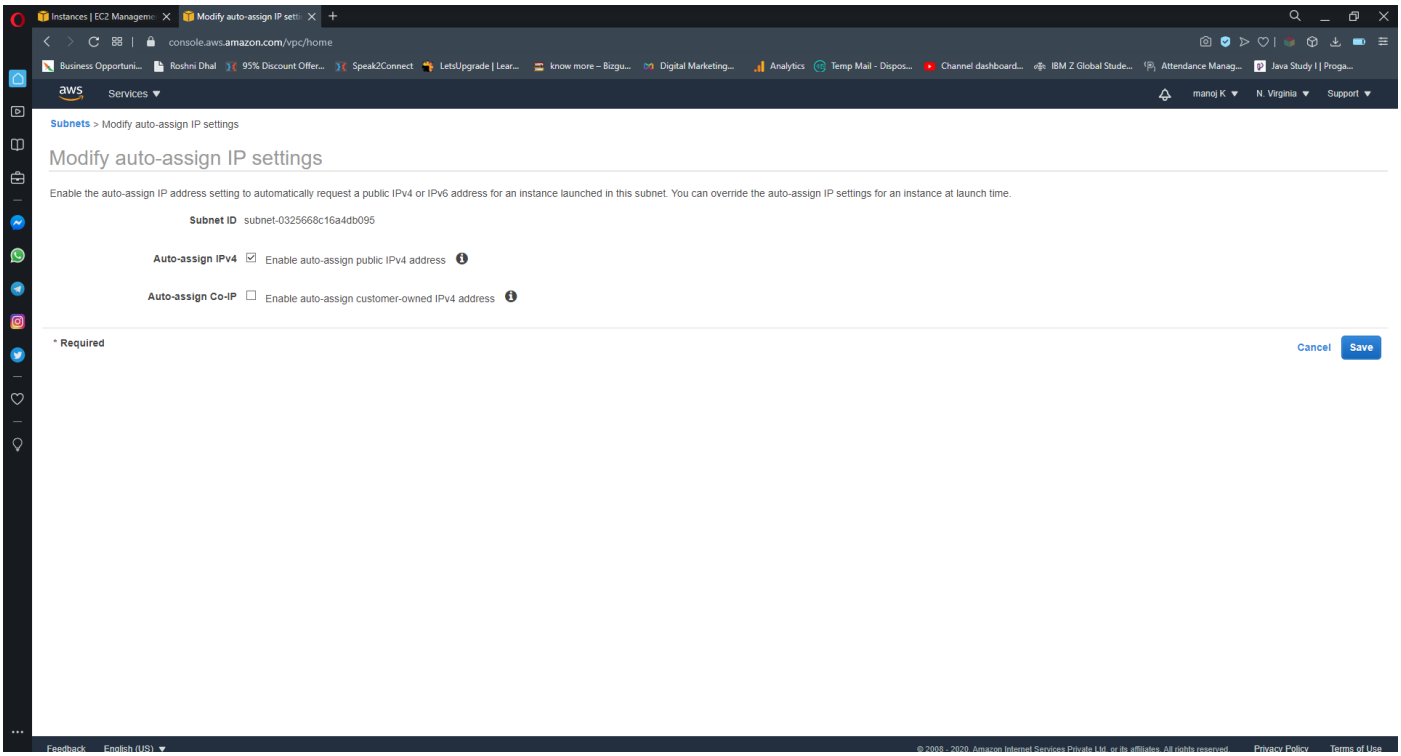
VPC Details:



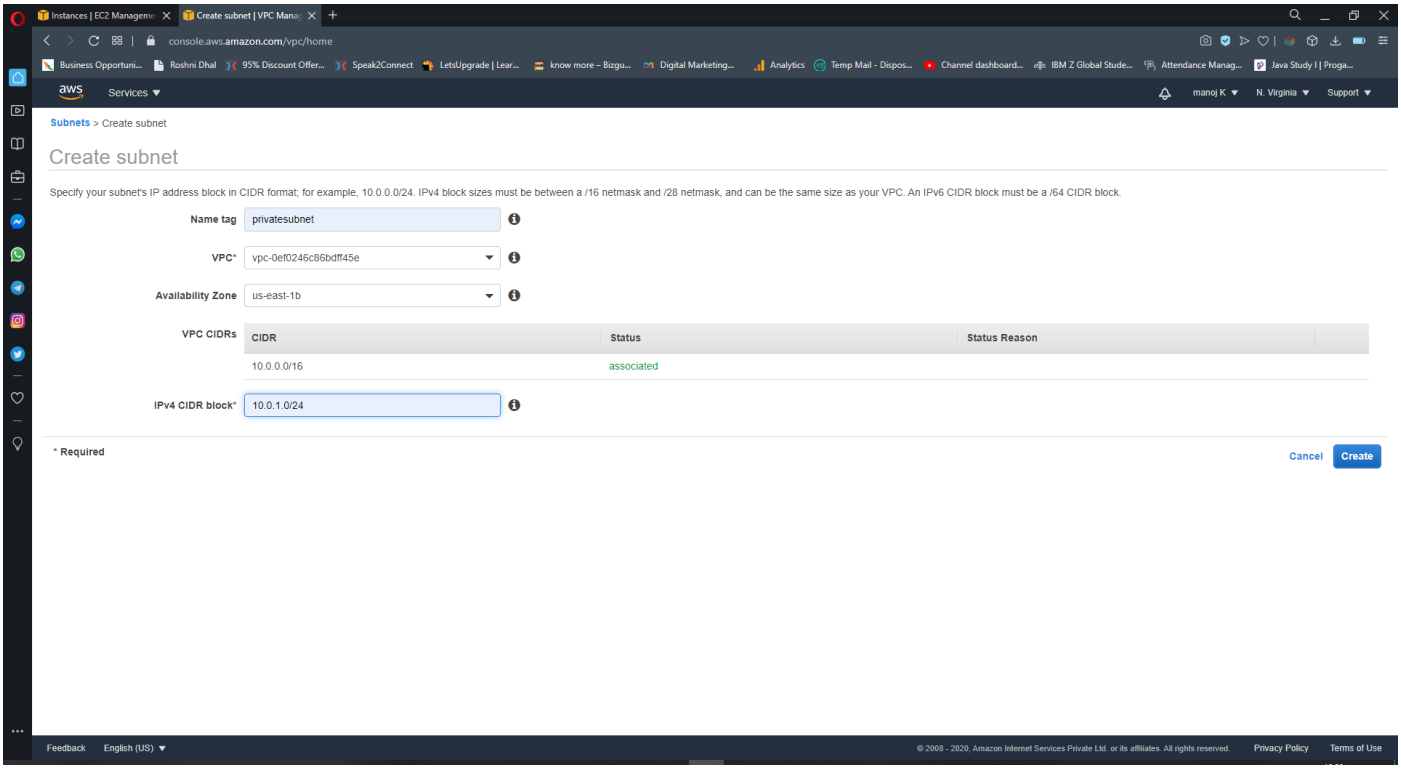
Creation of public subnet:



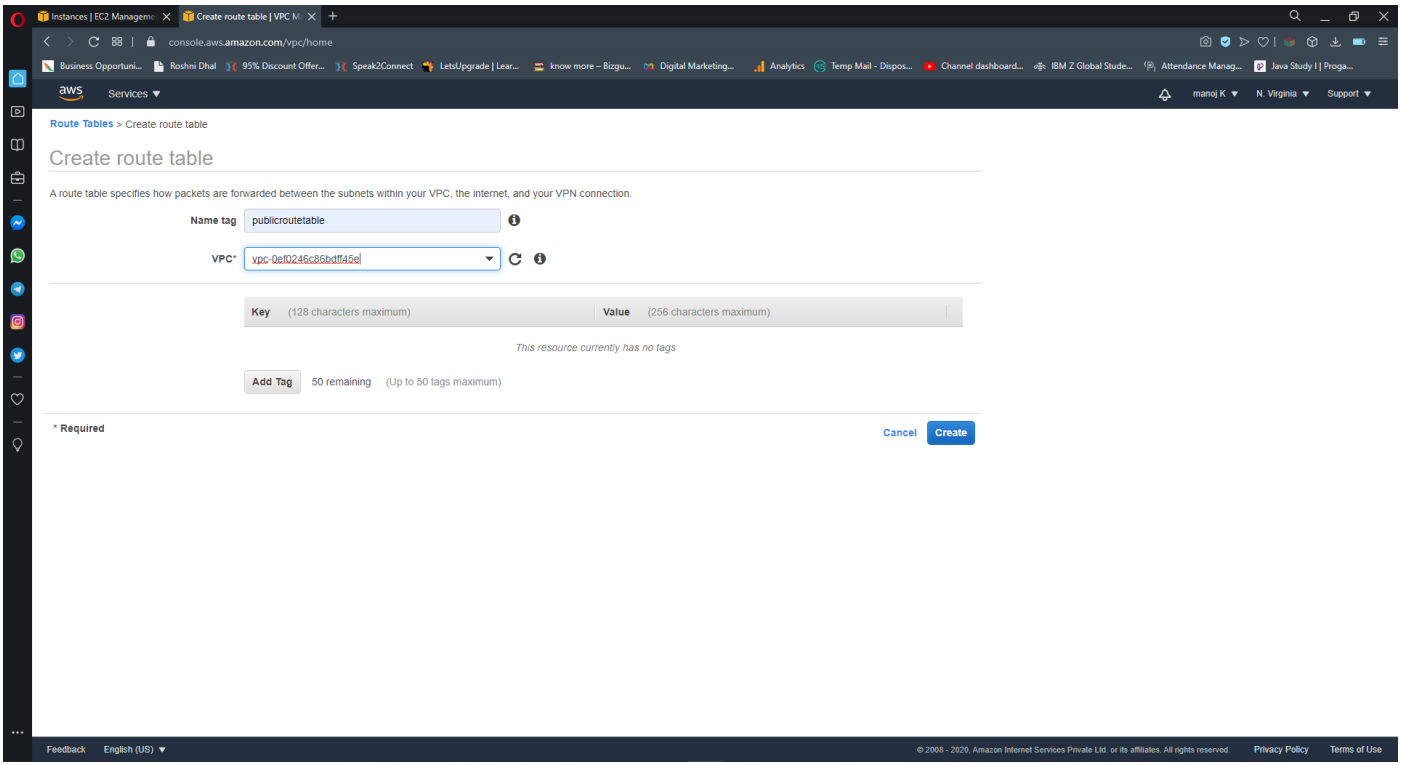
Checking the auto-assign public iPv4 address for public subnet:



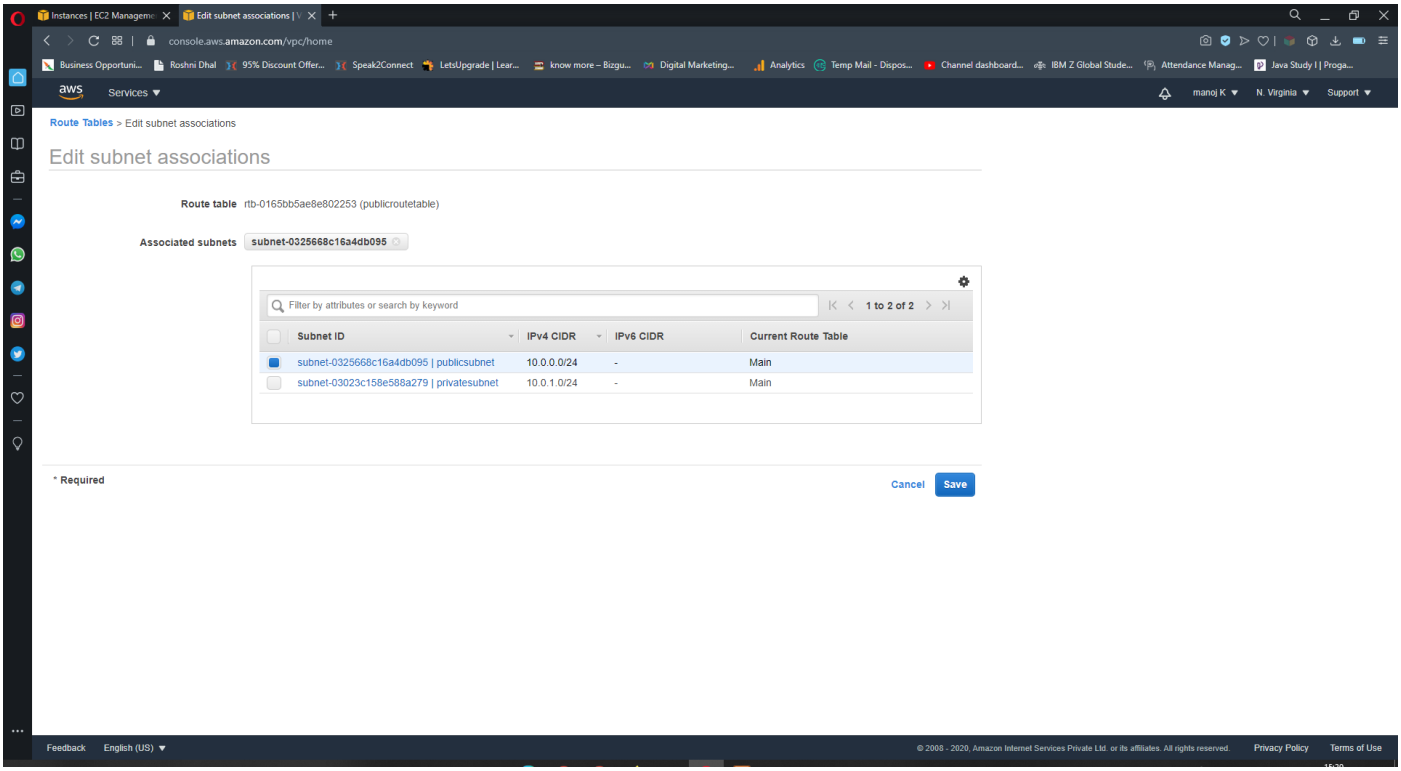
Creation of private subnet:



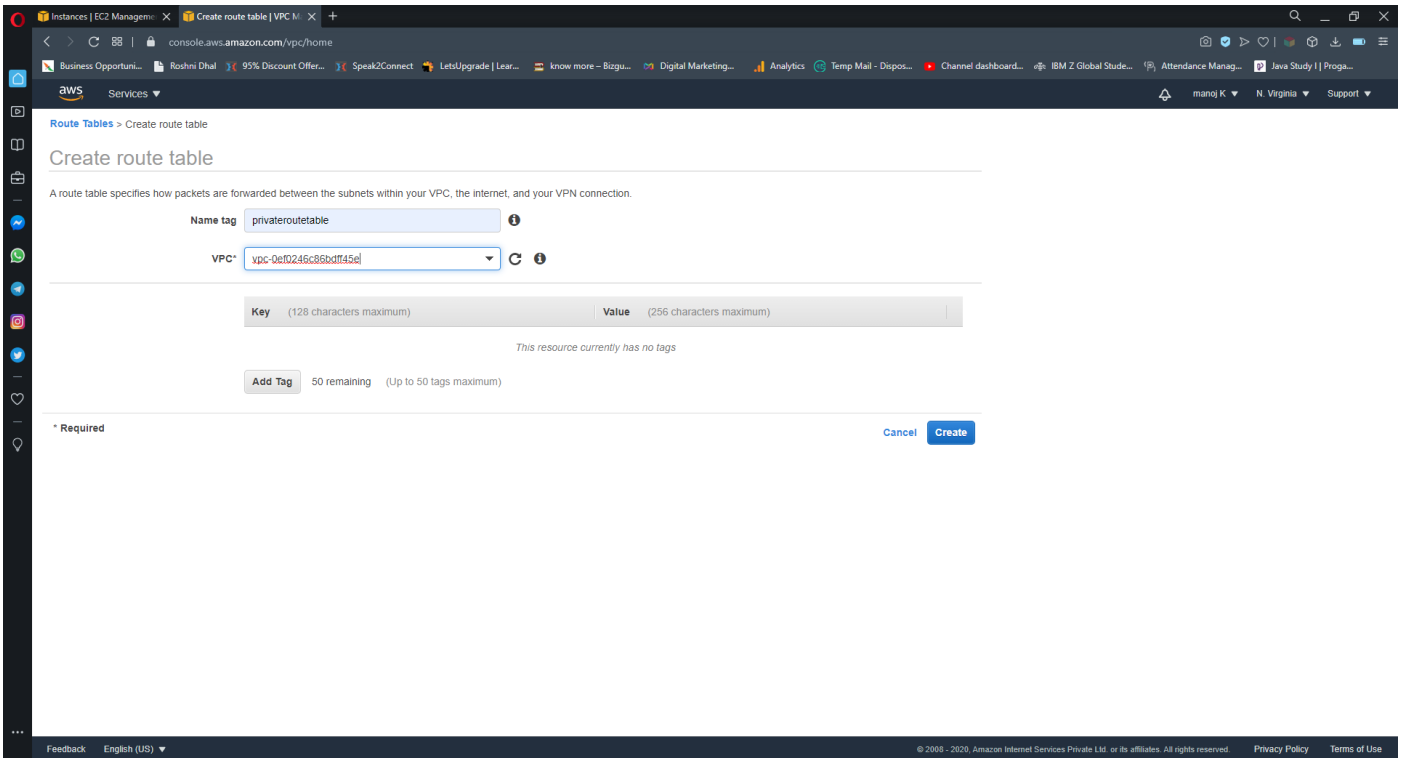
Creating public route table:



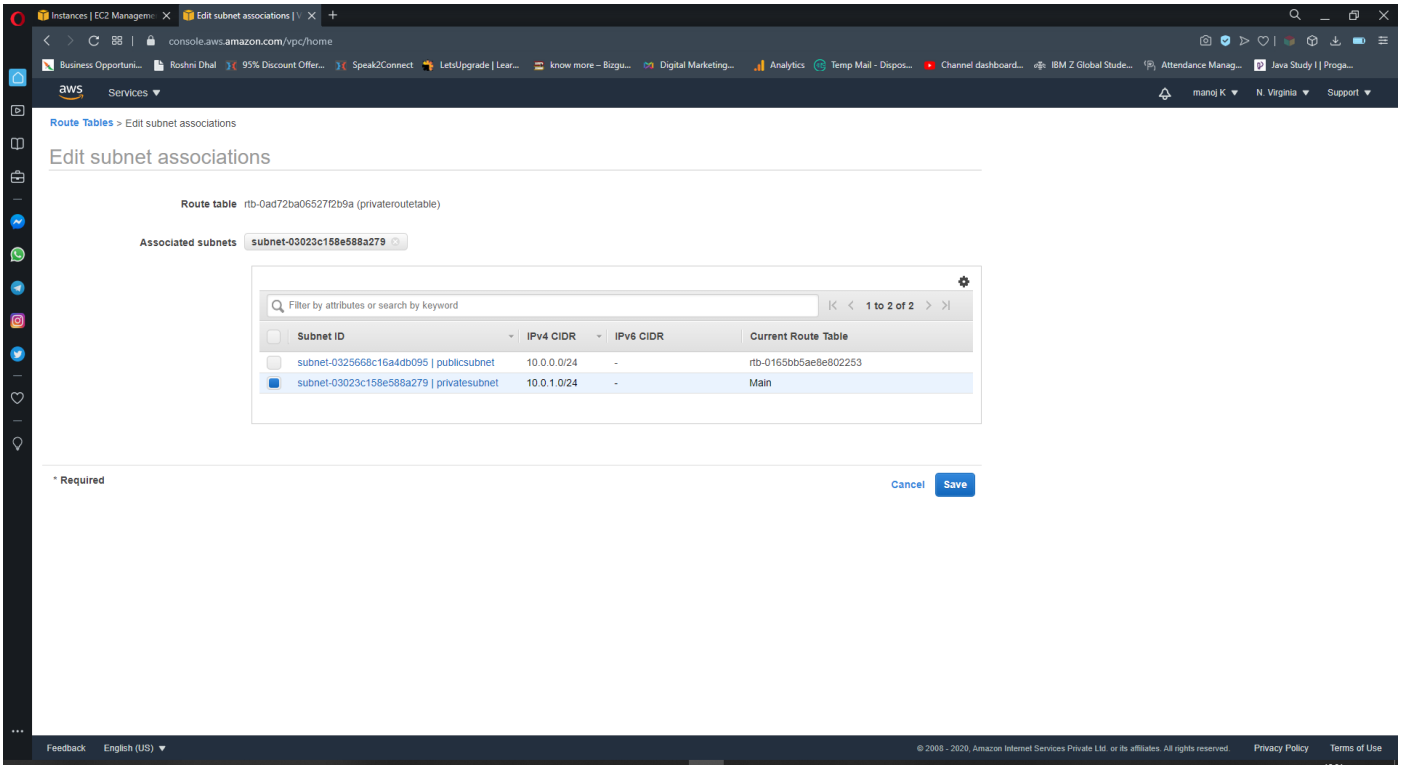
Associating public route table with public subnet:



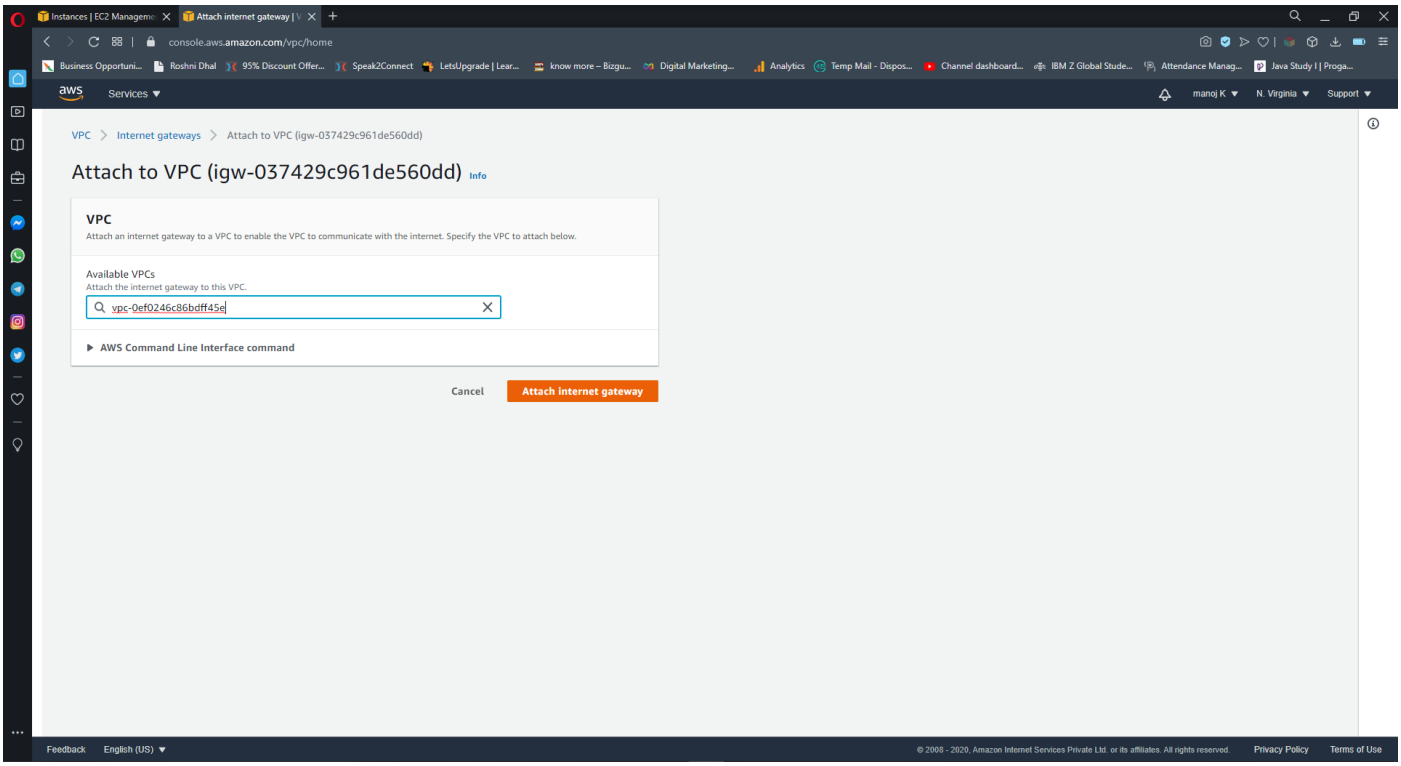
Creating private route table :



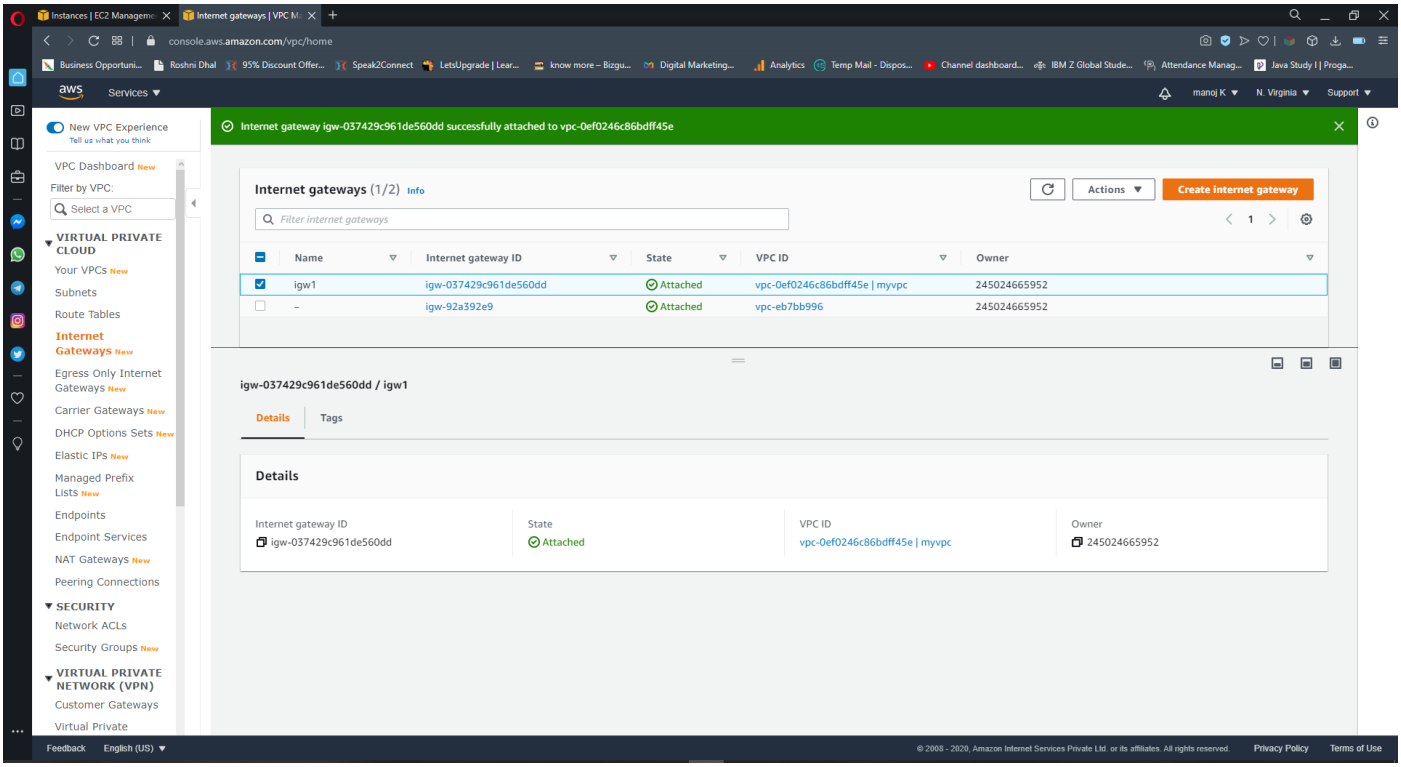
Creating private route table with private subnet:



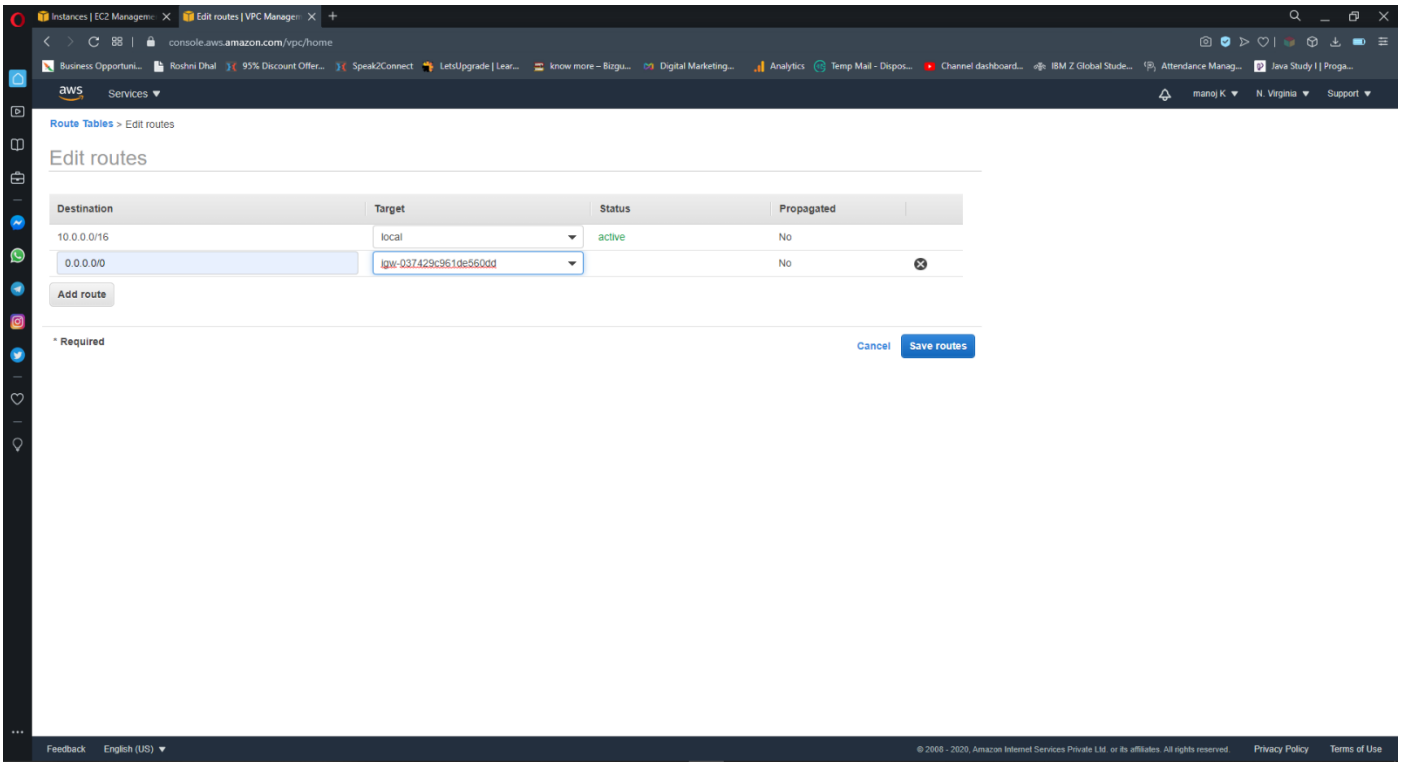
Attaching Internet Gateway with the VPC:



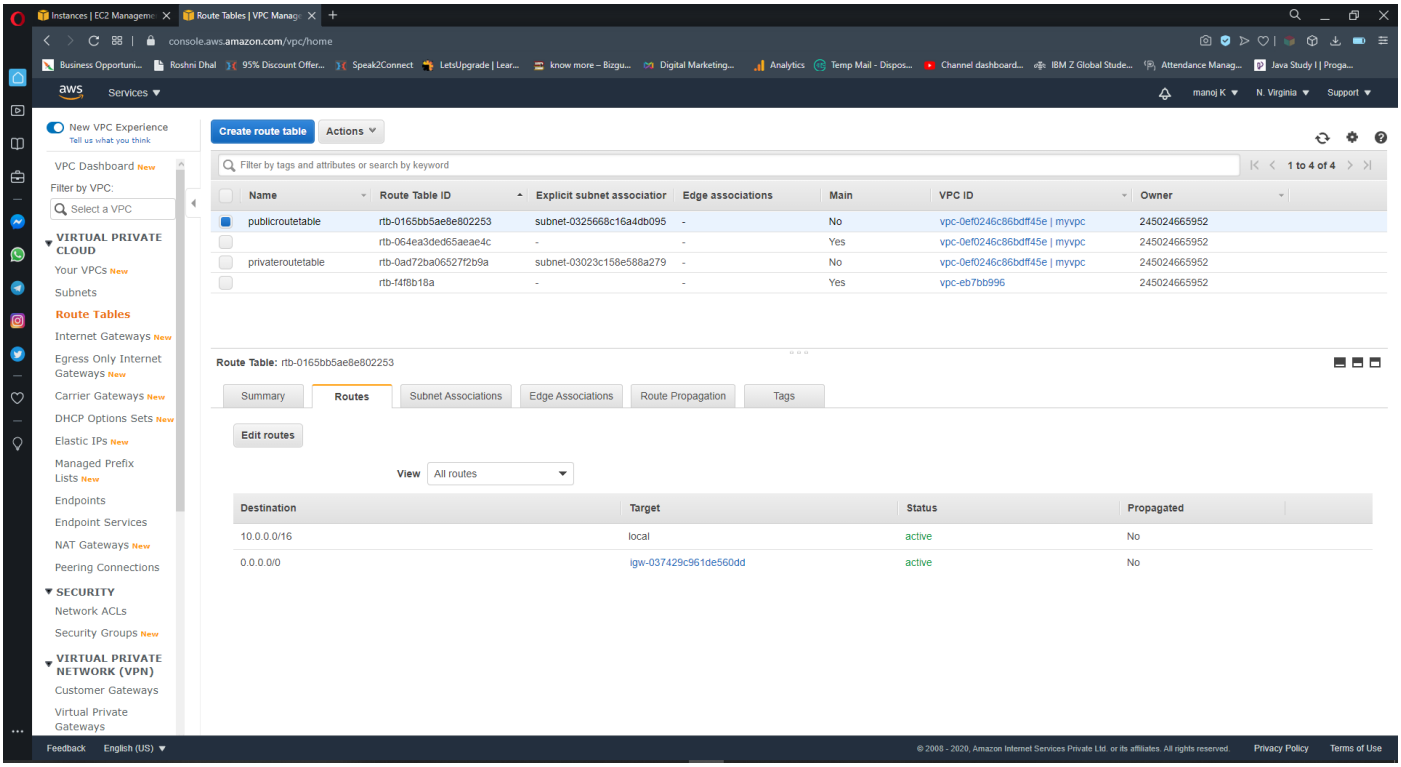
Created Internet Gateway Details:



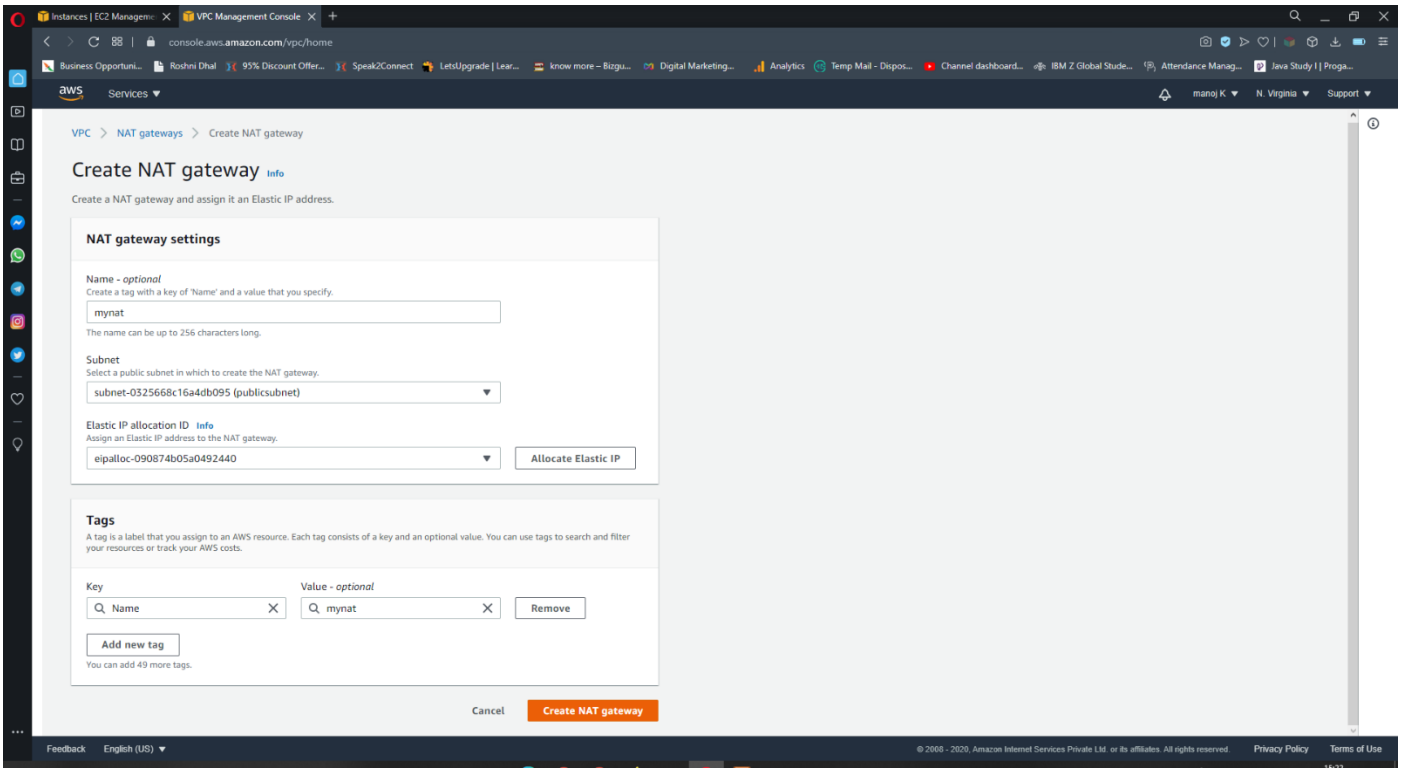
Editing routes in public route table i.e adding internet gateway:



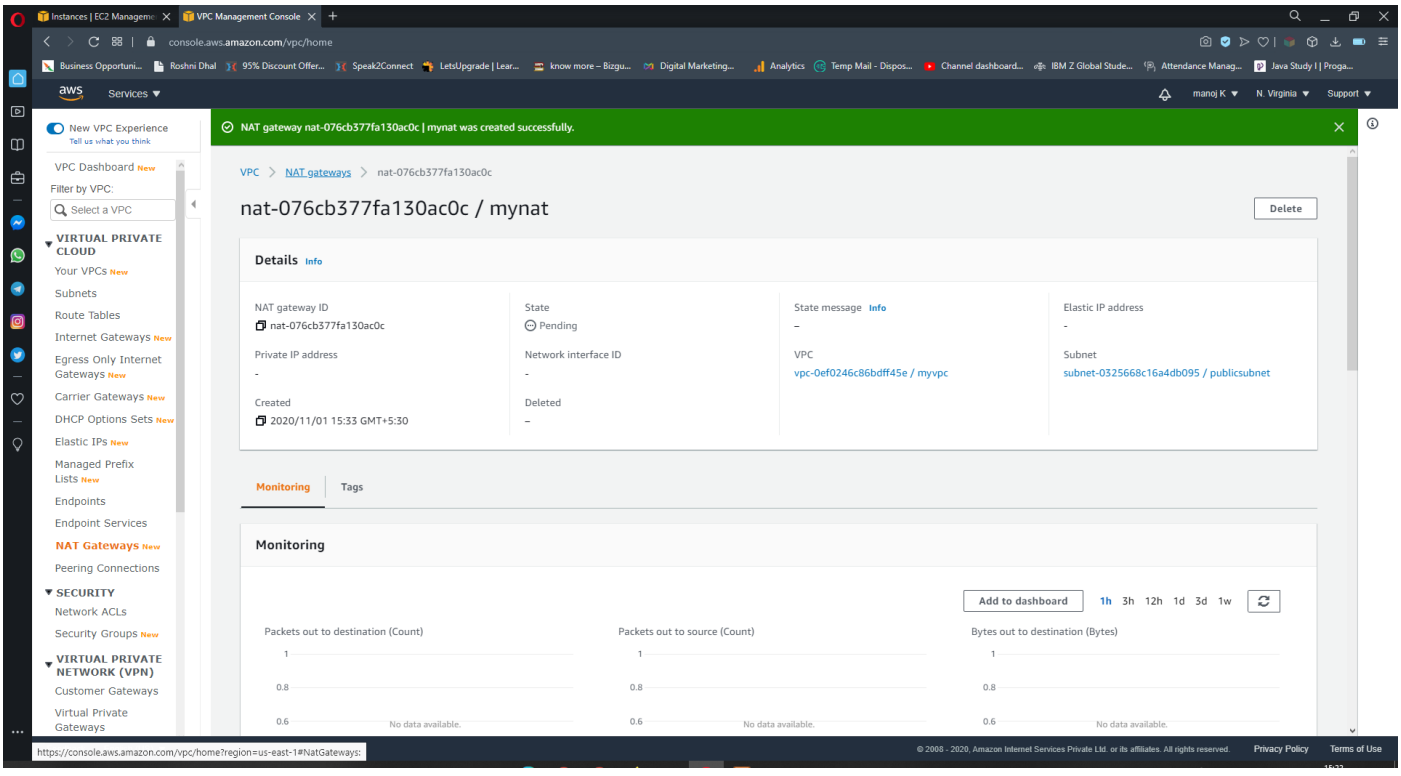
Public route table details:



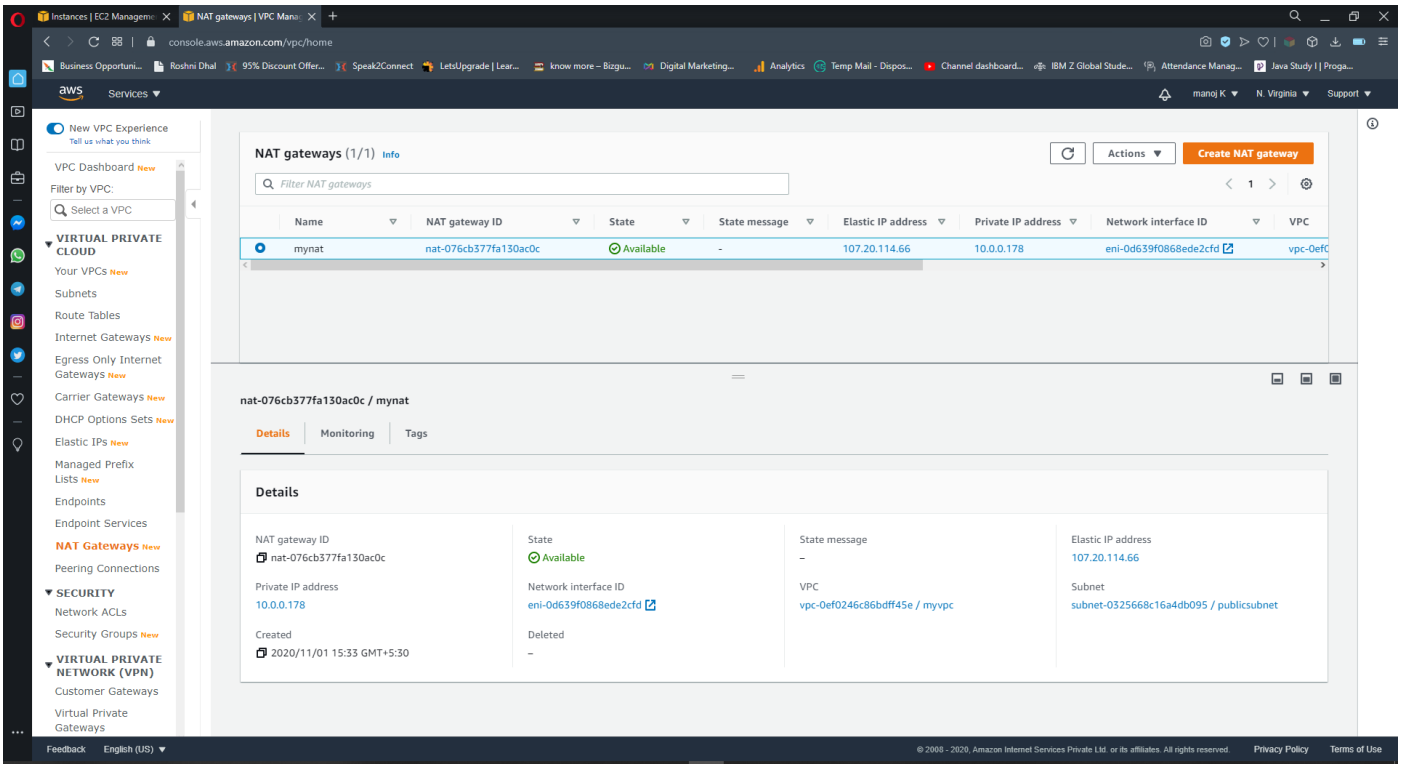
Creating NAT gateway:



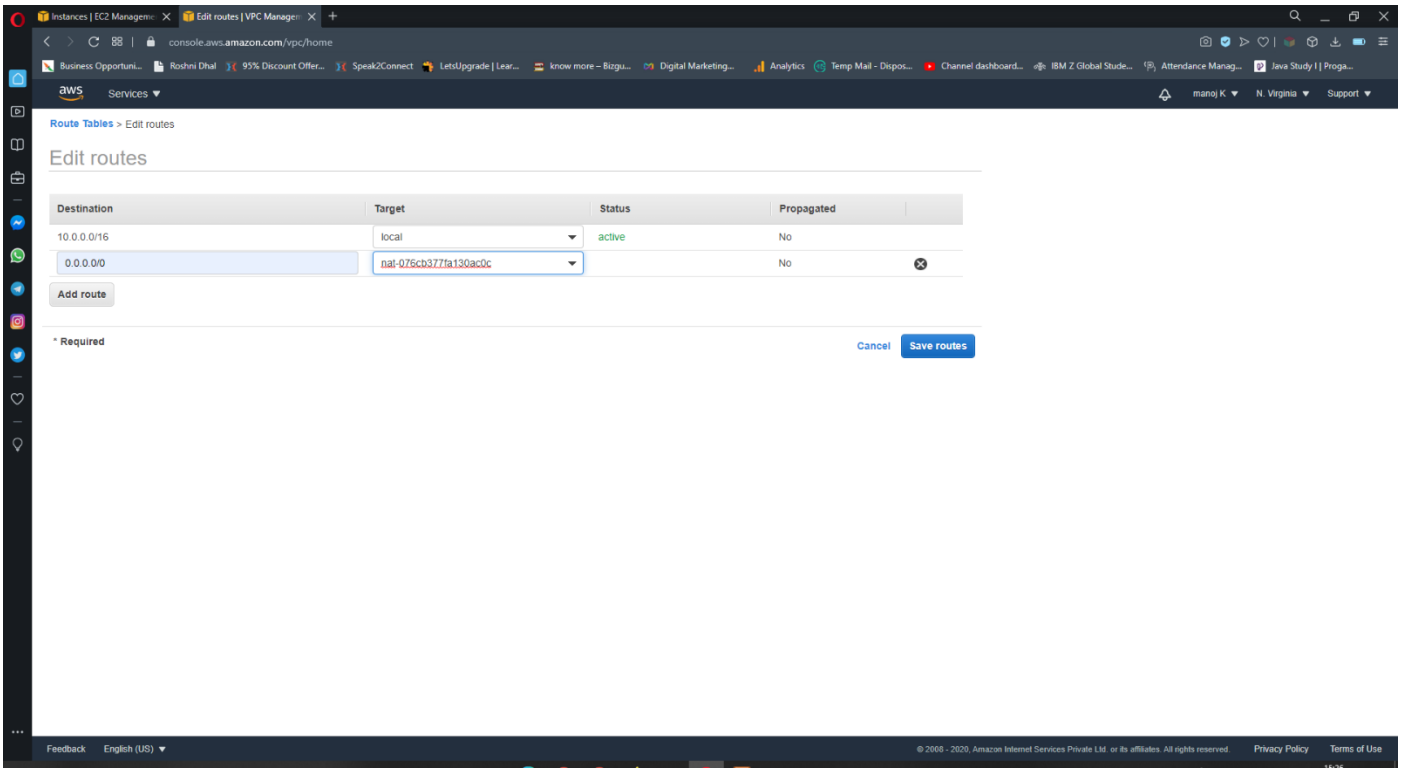
Successful Creation of NAT:



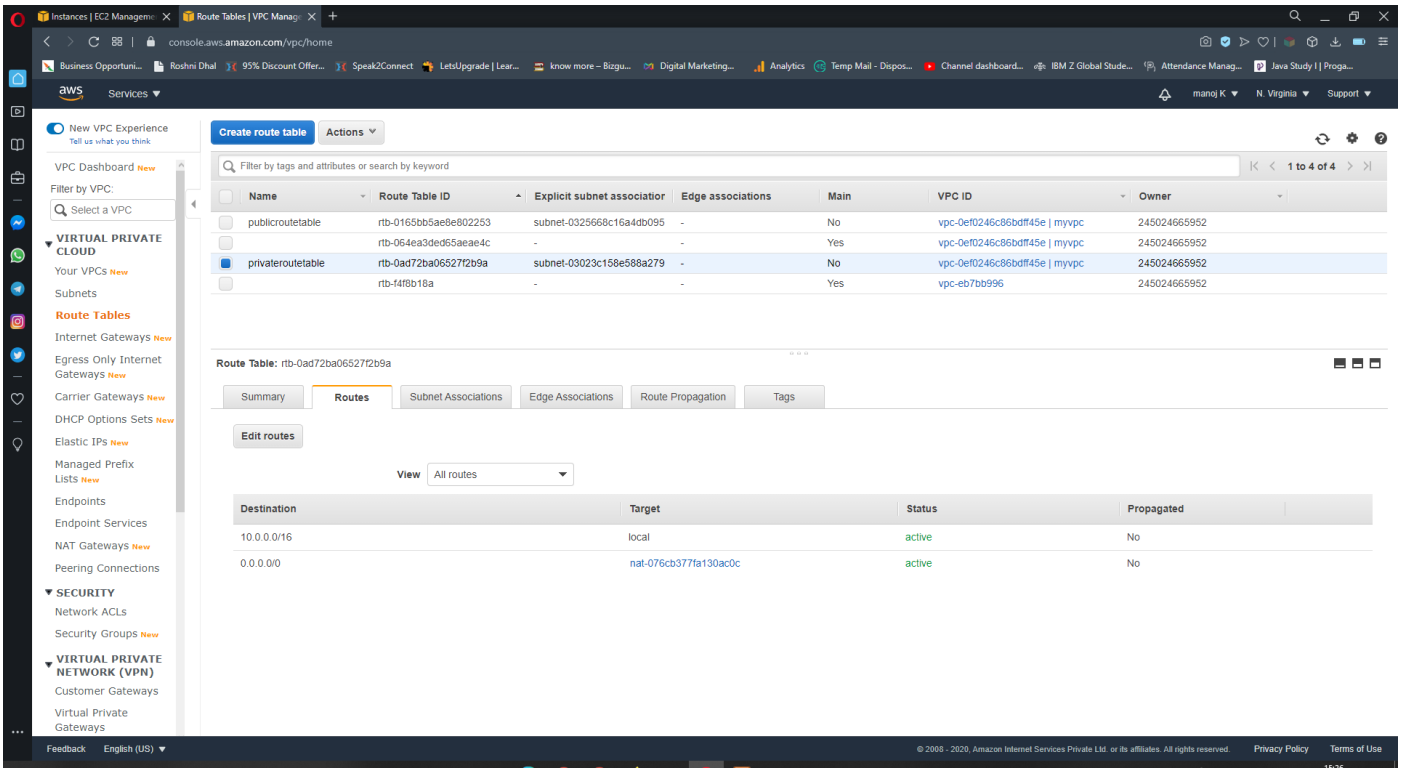
NAT Gateway Details:



Editing Routes in private route Table i.e allowing NAT Gateway:



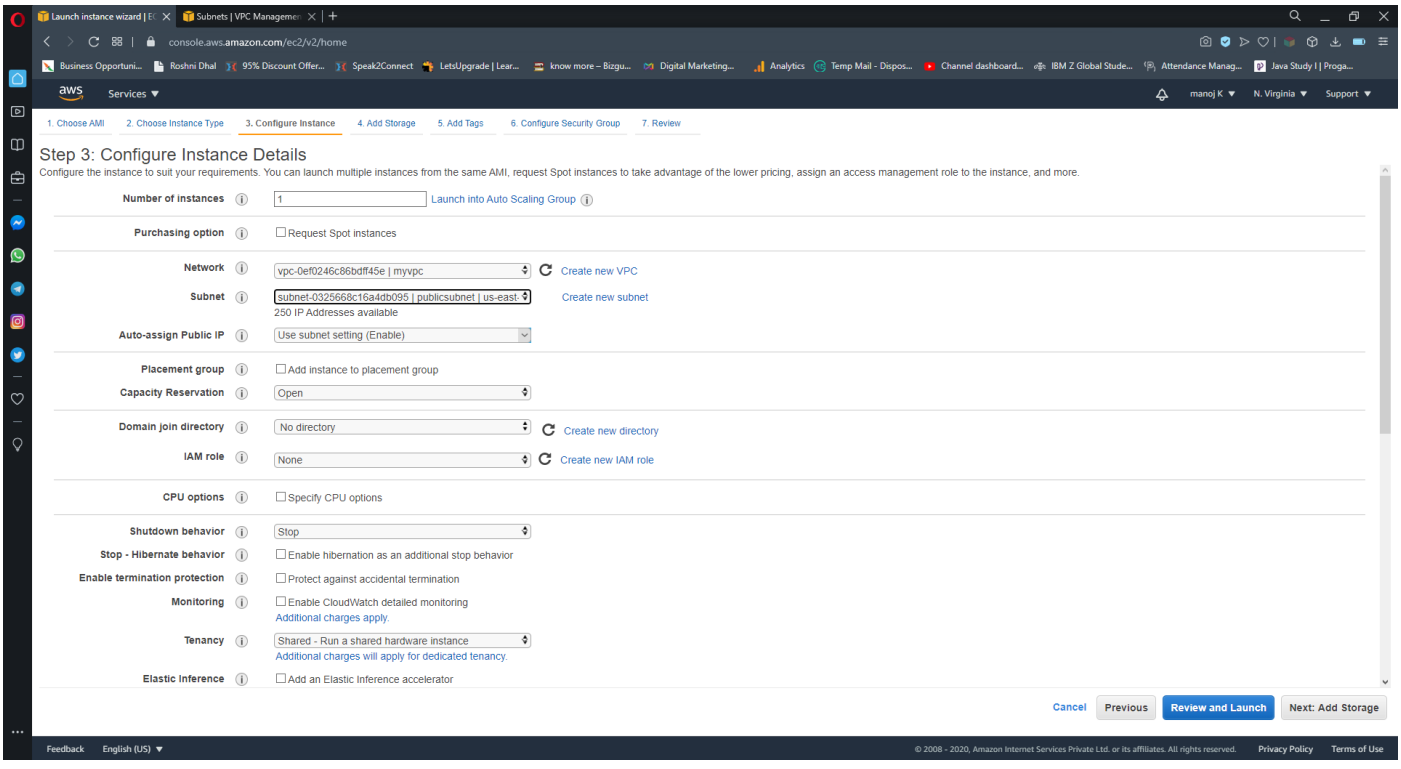
Private Route table Details:



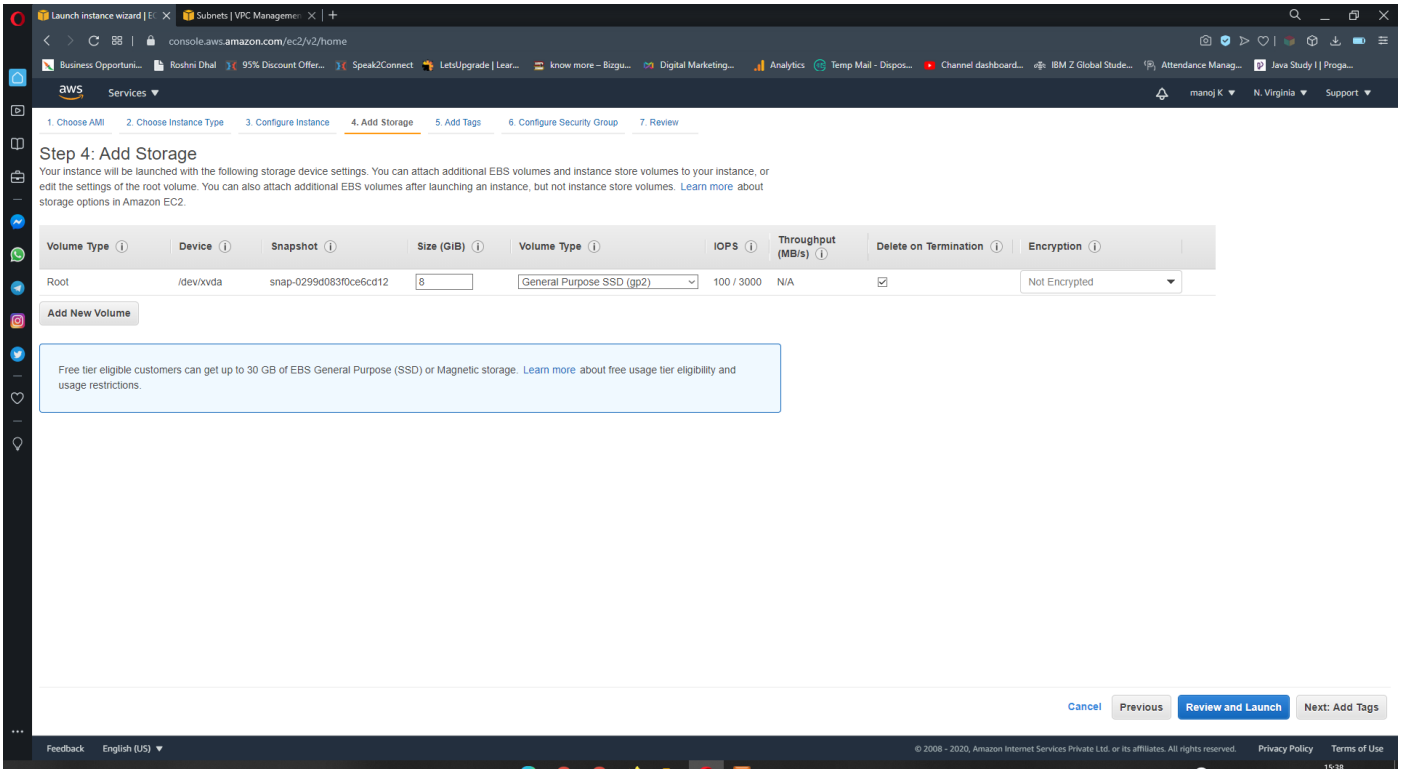
Now Let’s create instance for Public Subnet.

Creating instance in Public Subnet which is called Bastion-Server:

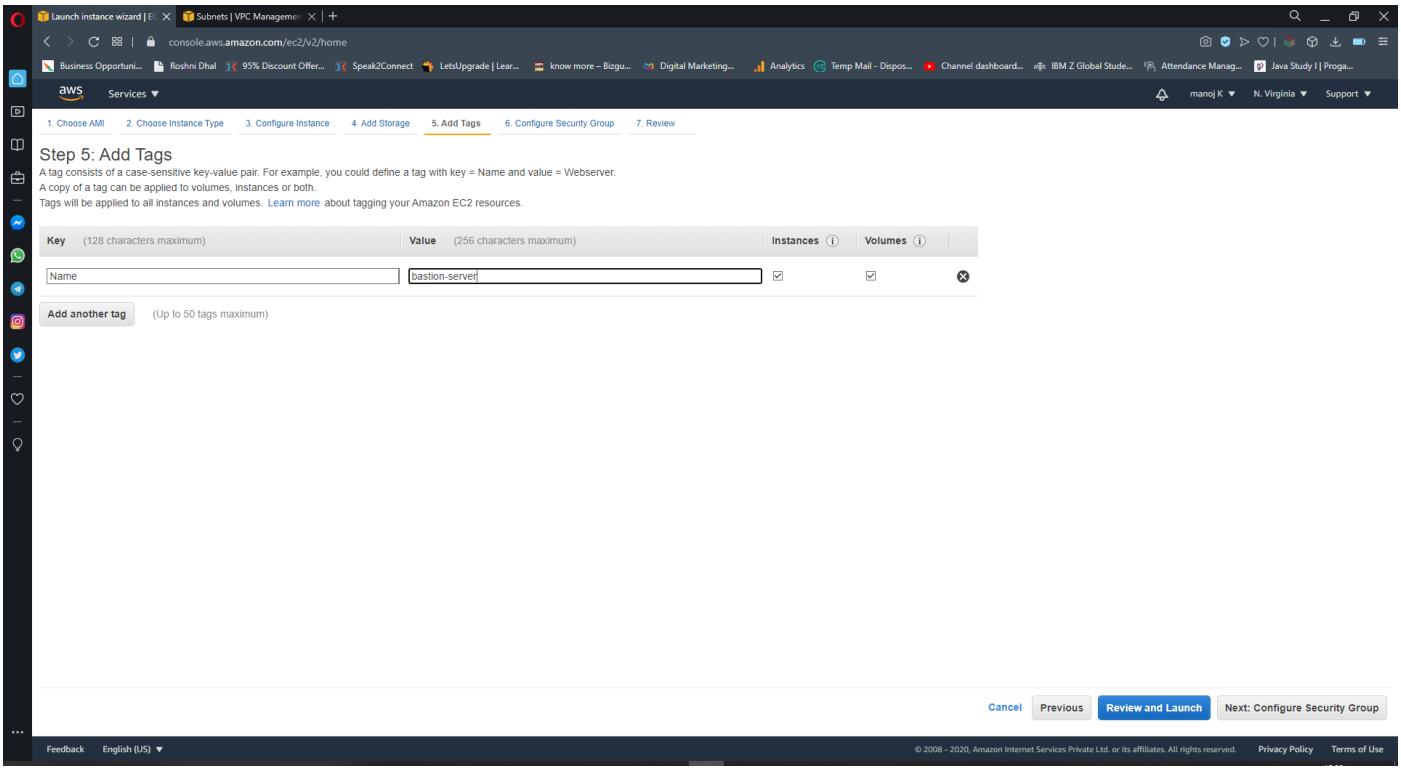
Through which private instance can be accessed.



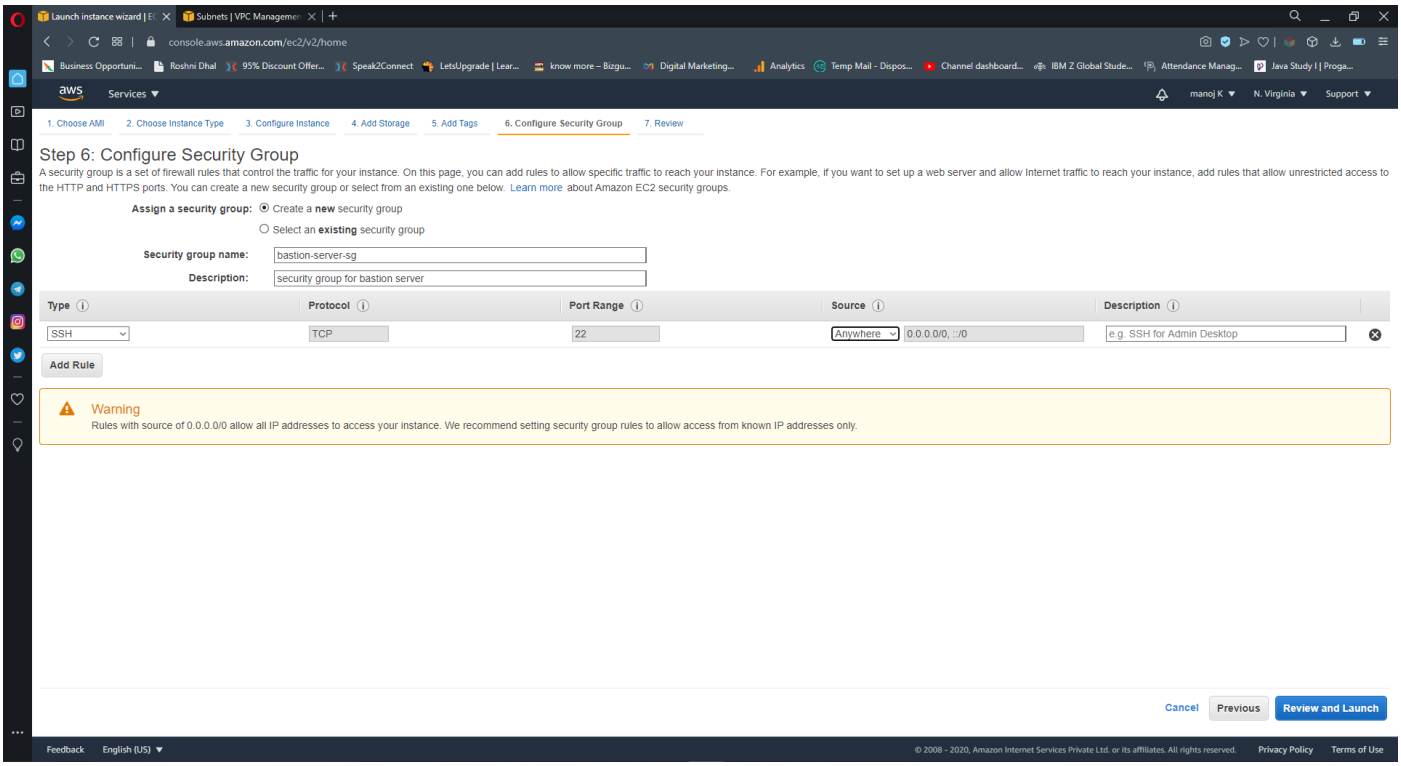
Storage for Bastion-Server:



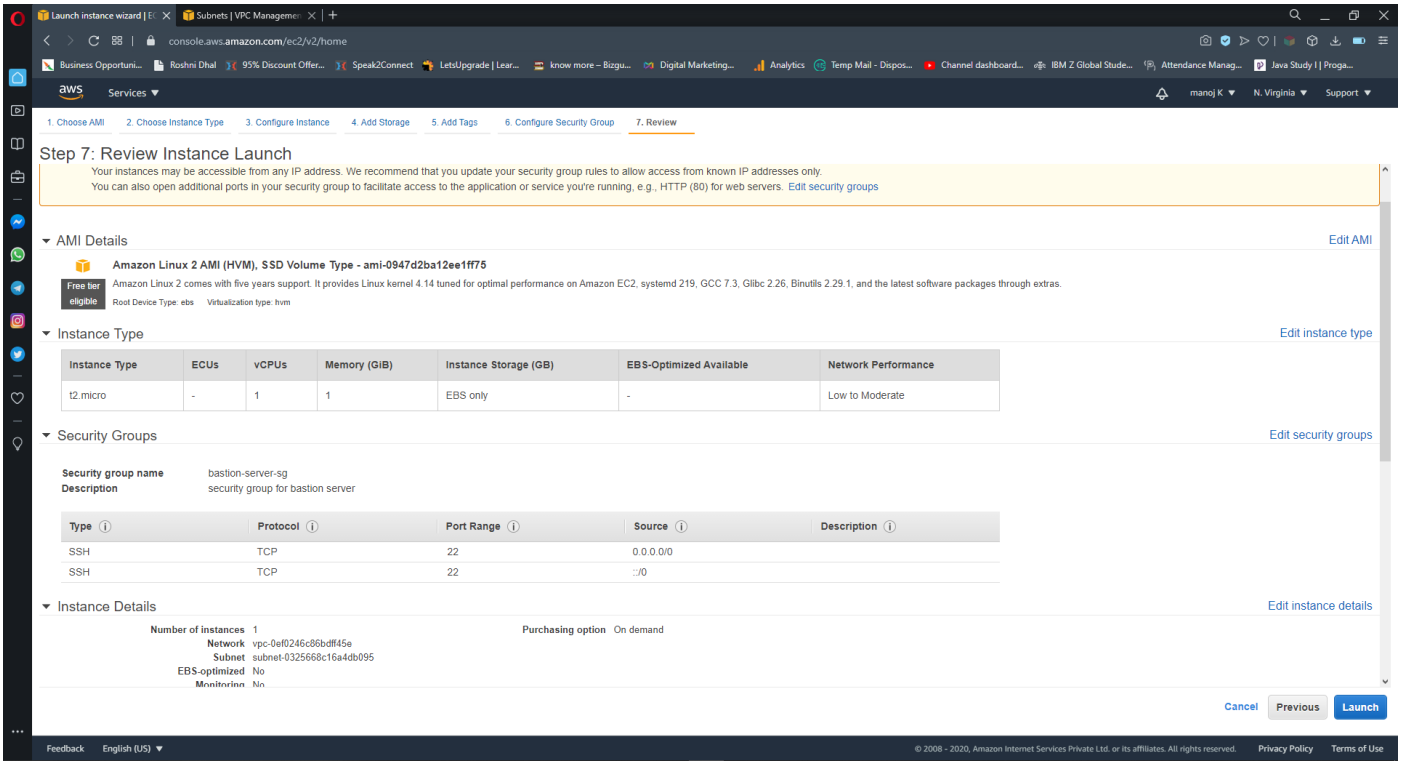
Adding tags for Bastion-server:



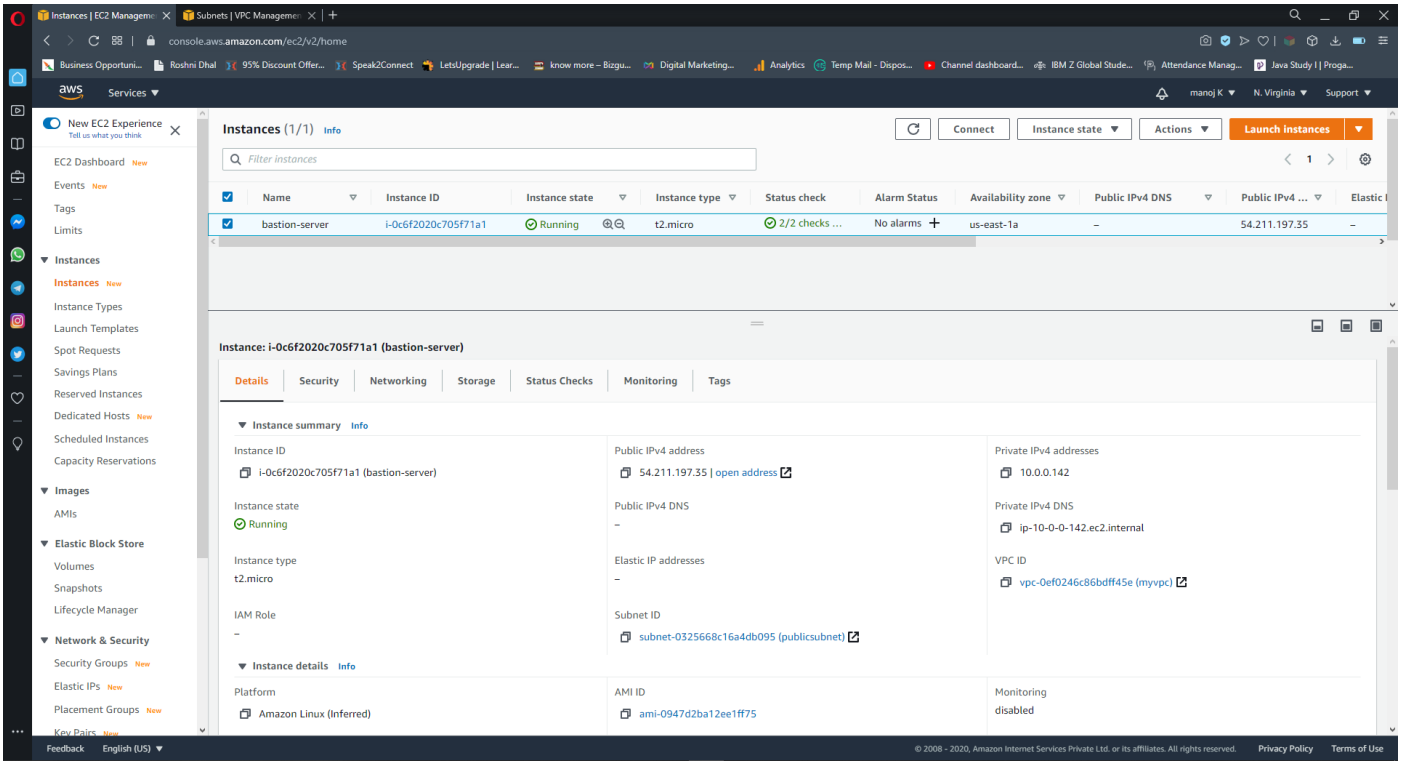
Configuring Security Group for Bastion Server:



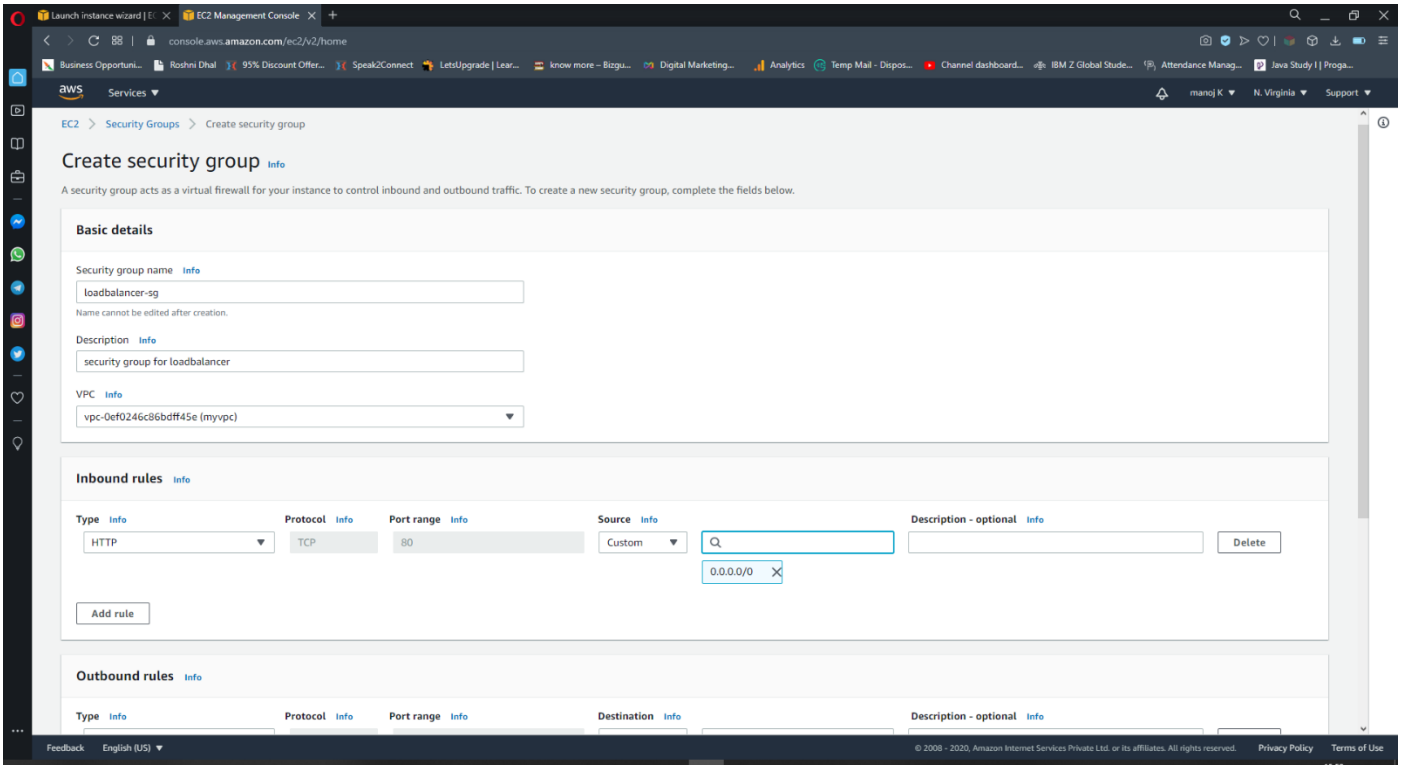
Review of instance for Bastion Server:



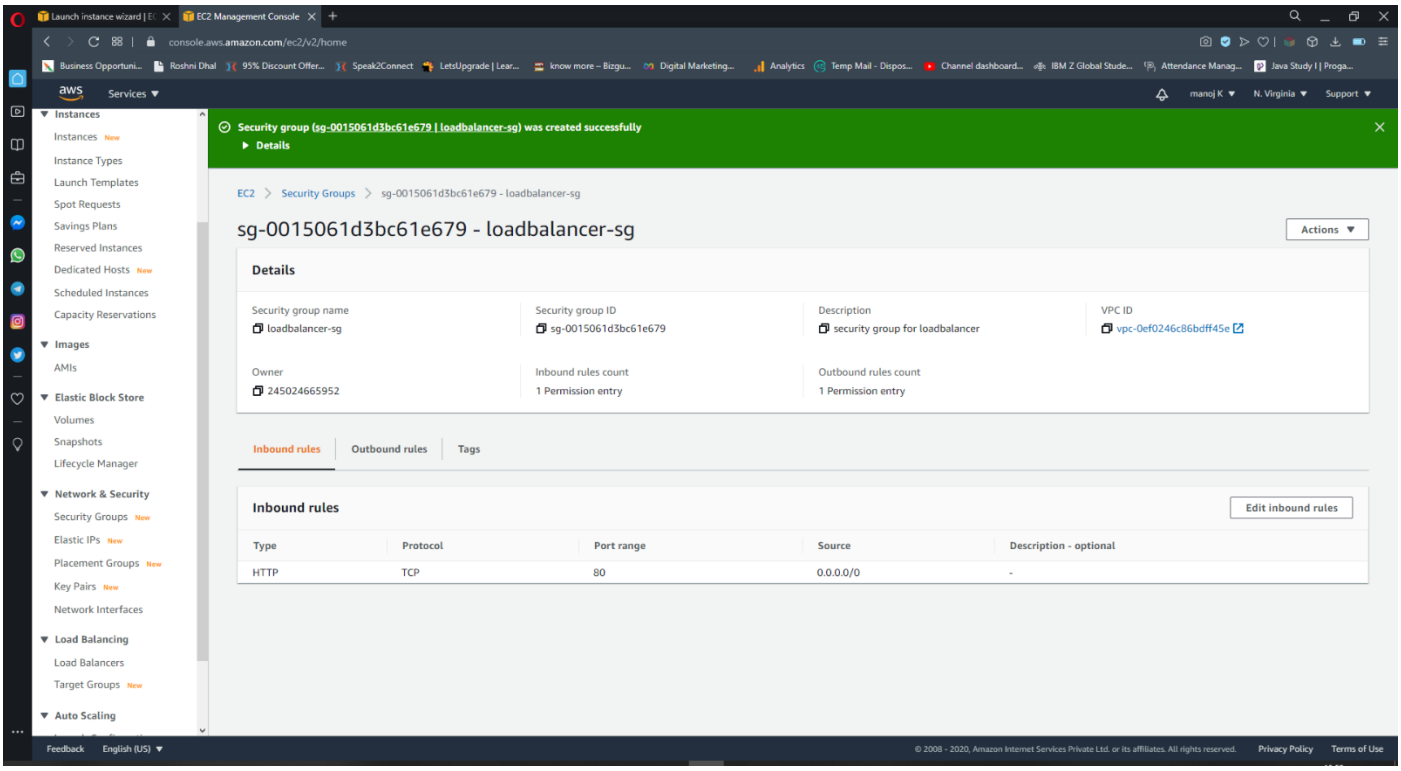
Bastion Server details:



Creating Security Group for loadbalancer:

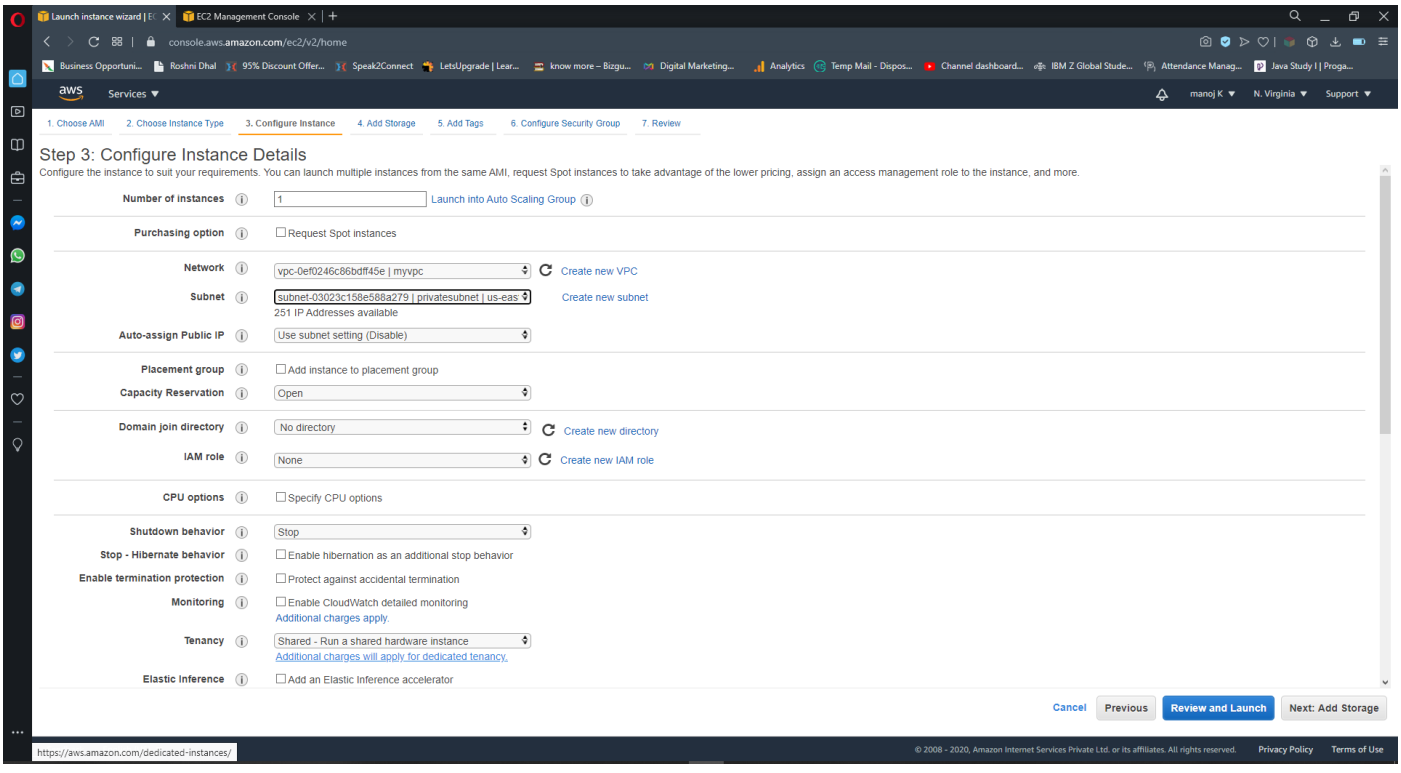


Load balancer security group creation:

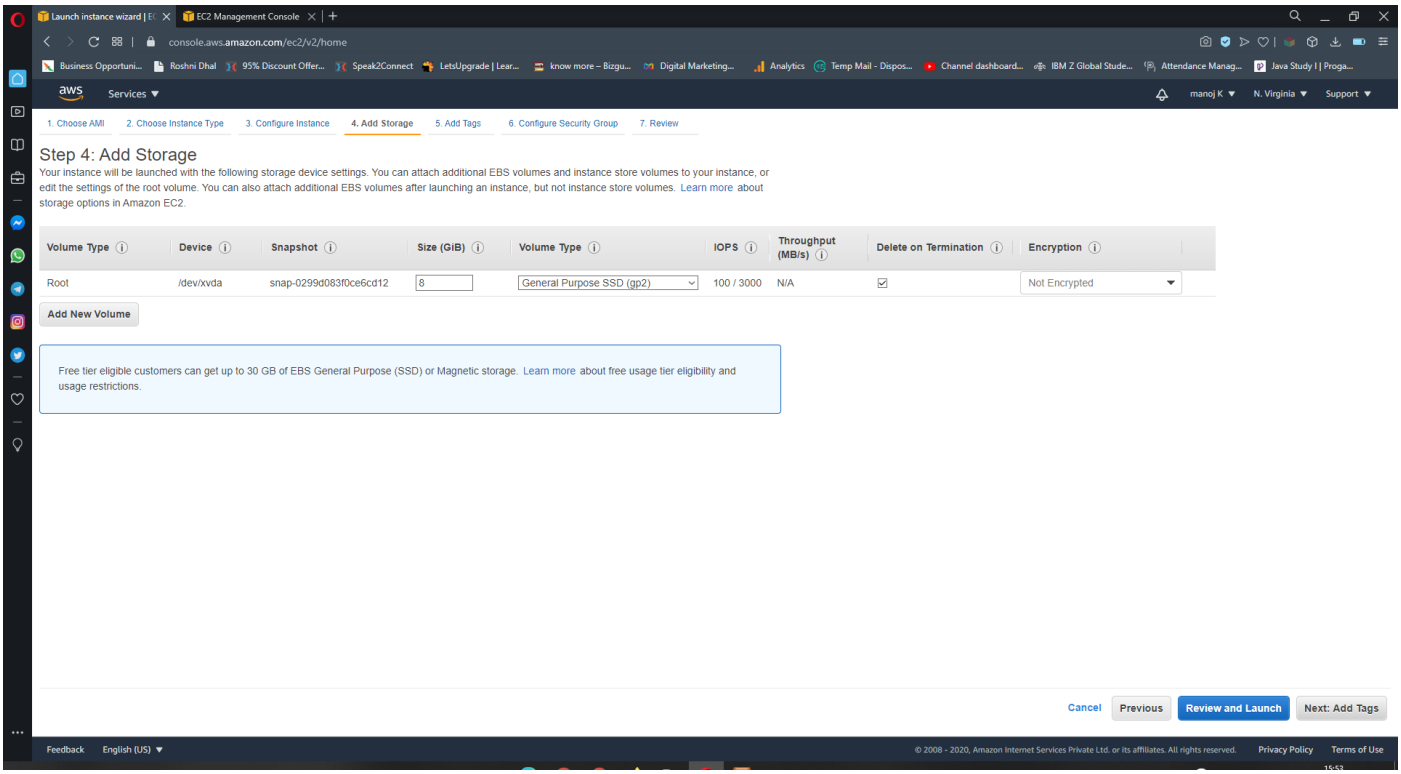


Now it’s time create web-servers in private network

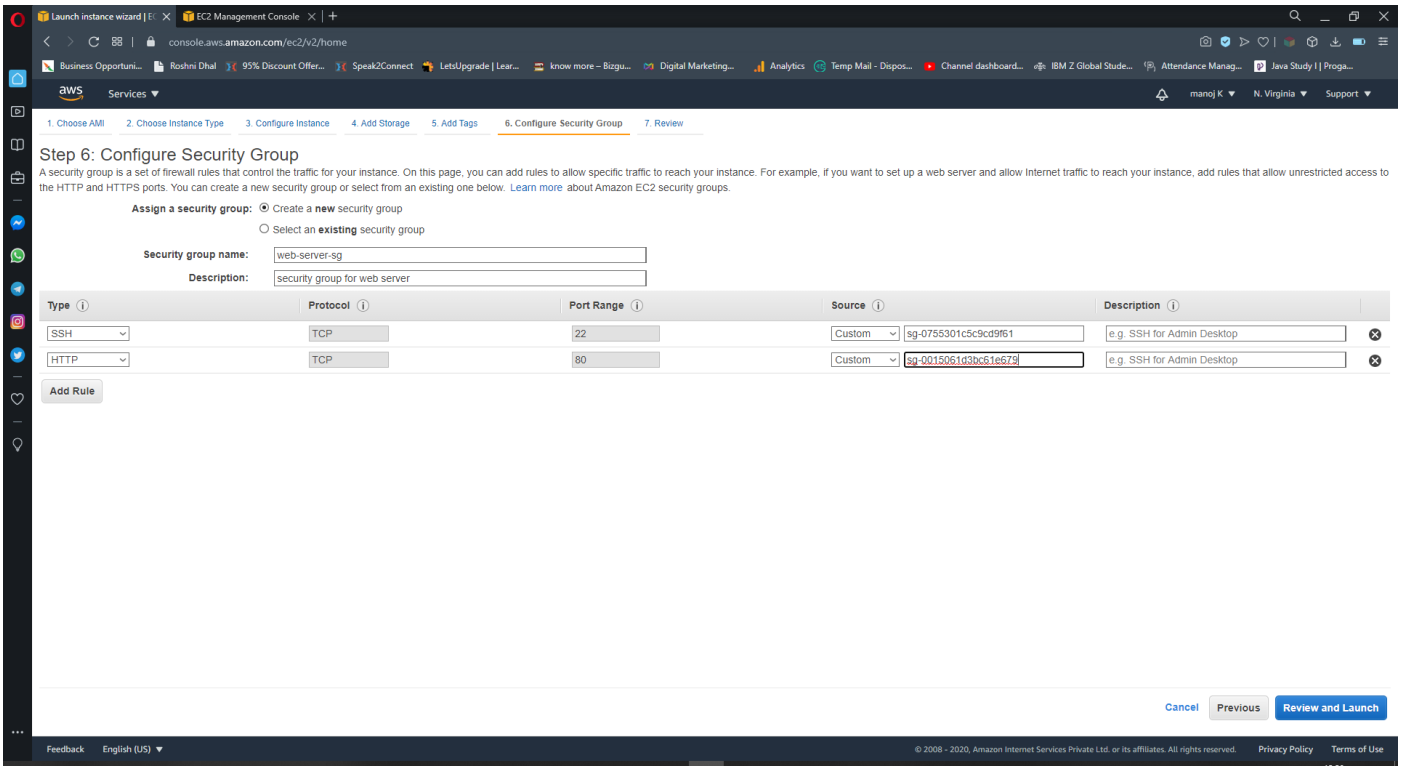
Creation of web-server 1 in private subnet:



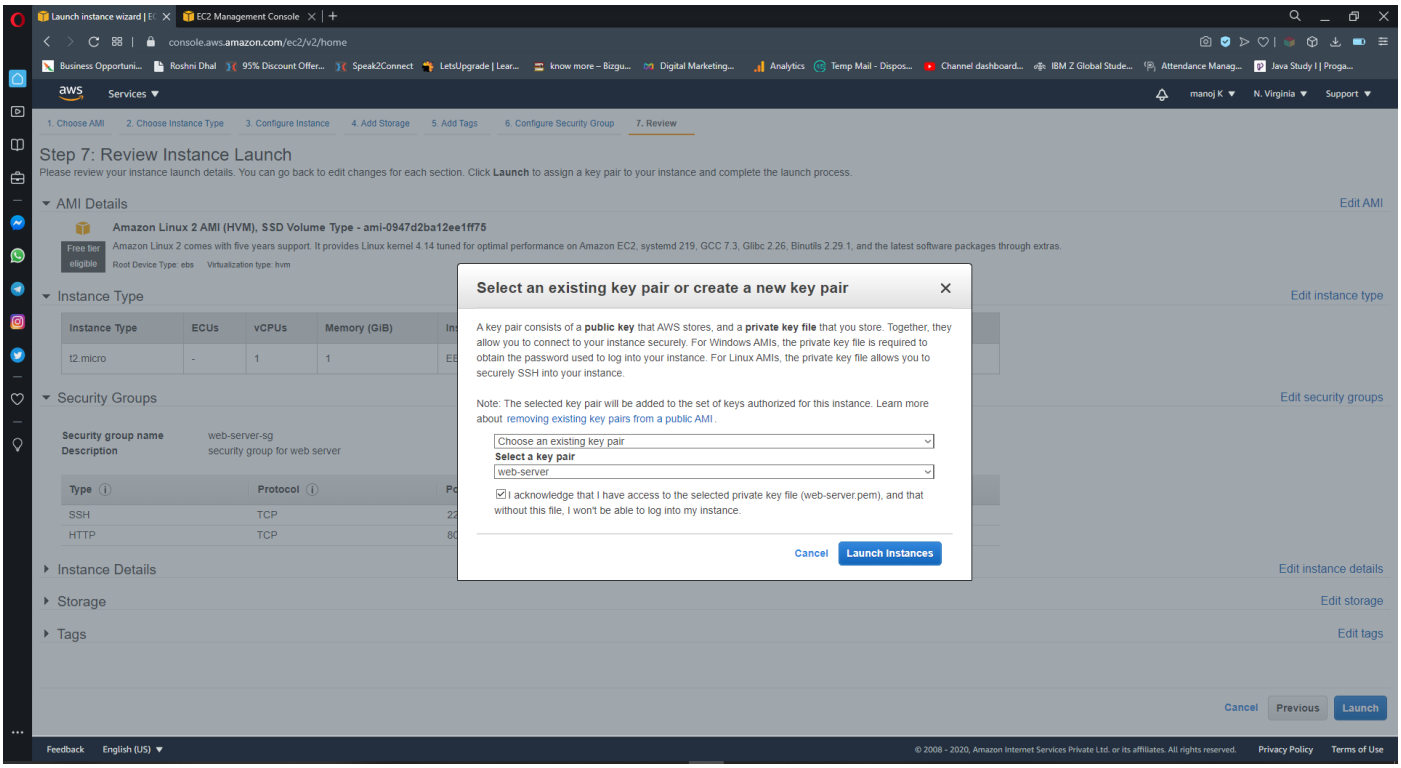
Adding storage to web-server 1:



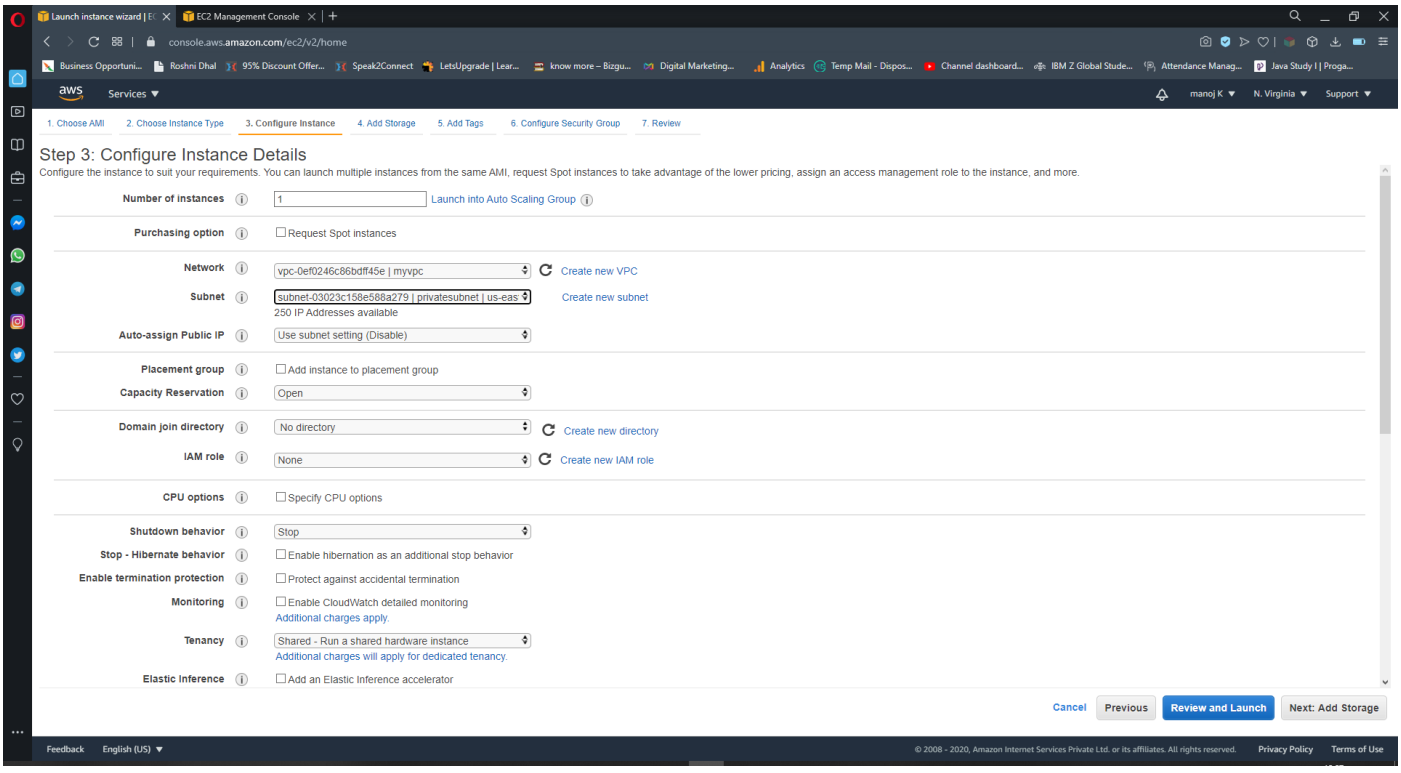
Configuring security group for web-server:



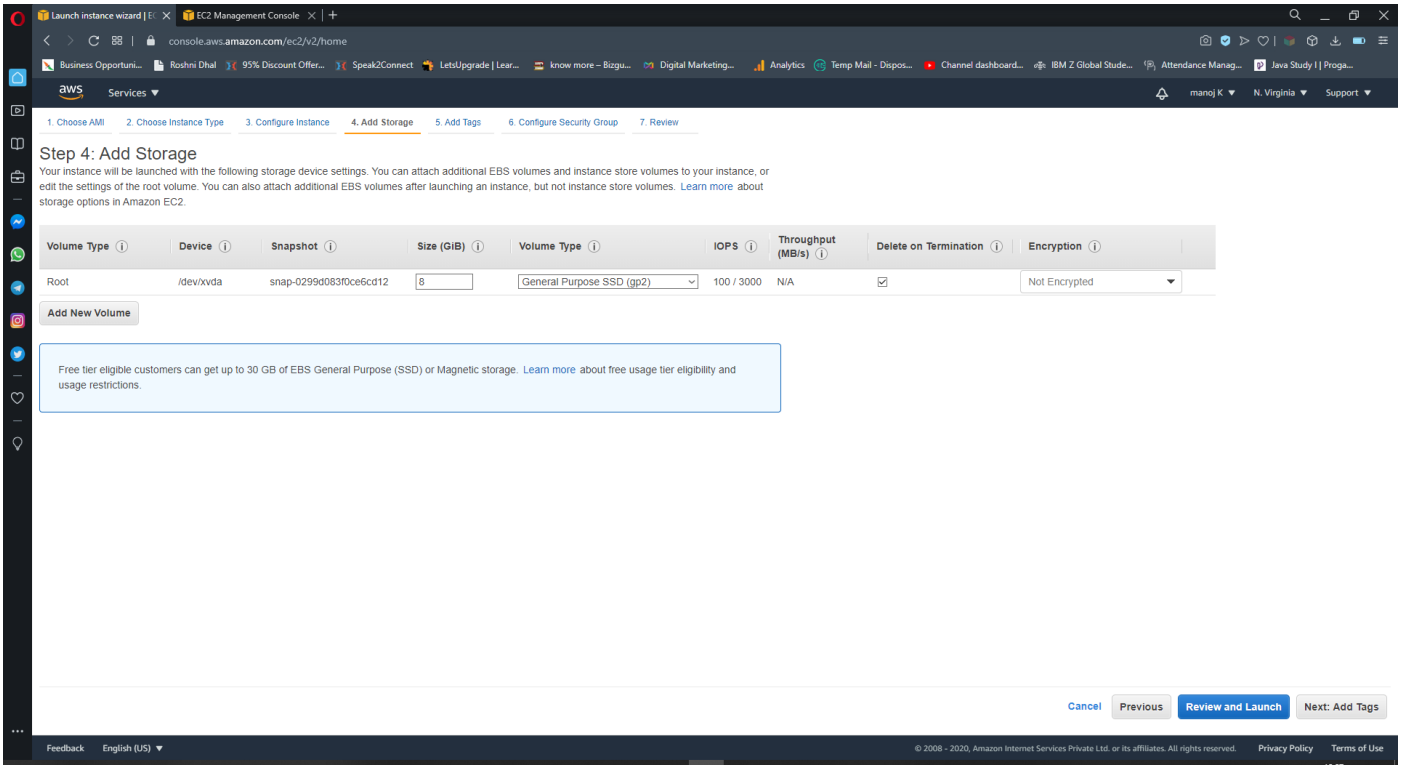
Launching the web-server 1:



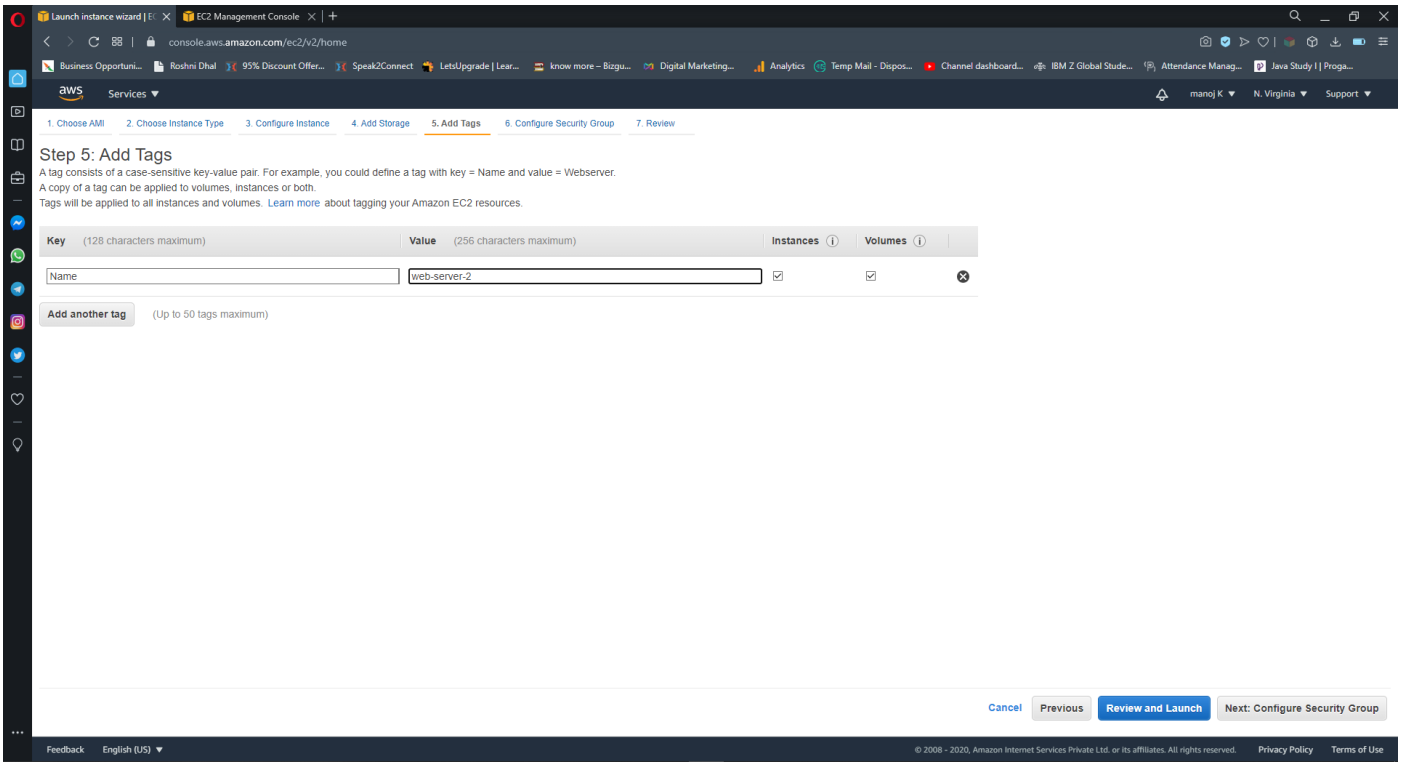
Creation of web-server 2:



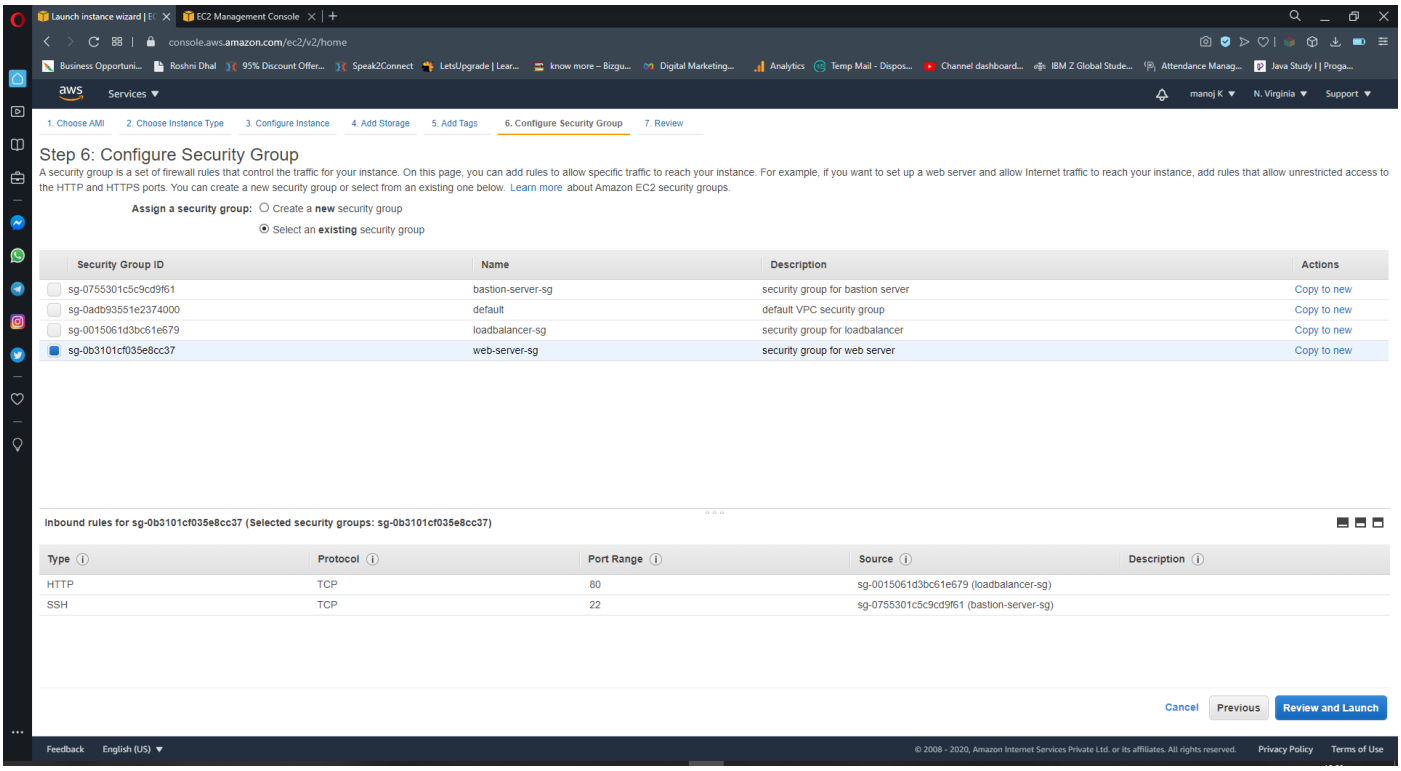
Adding storage to web-server 2:



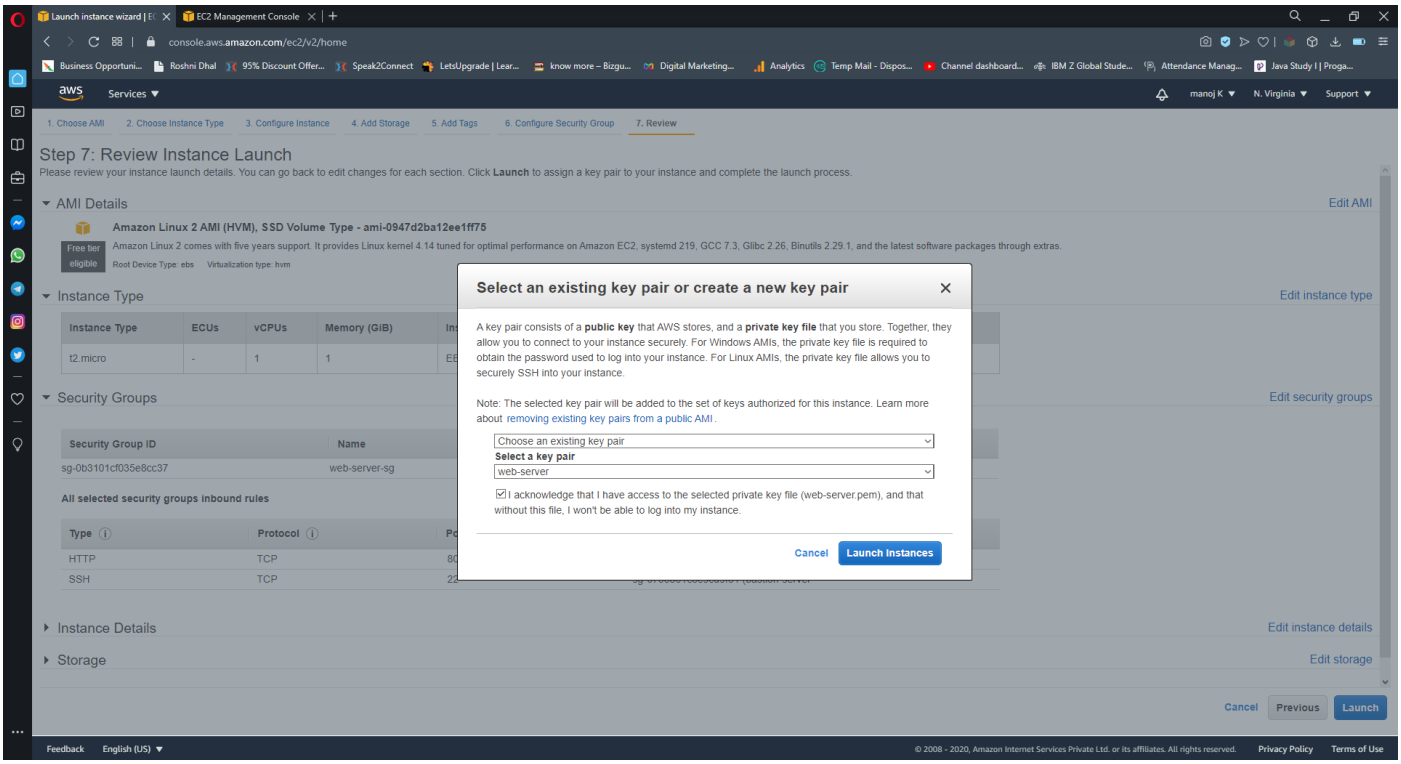
Adding tags for web-server 2:



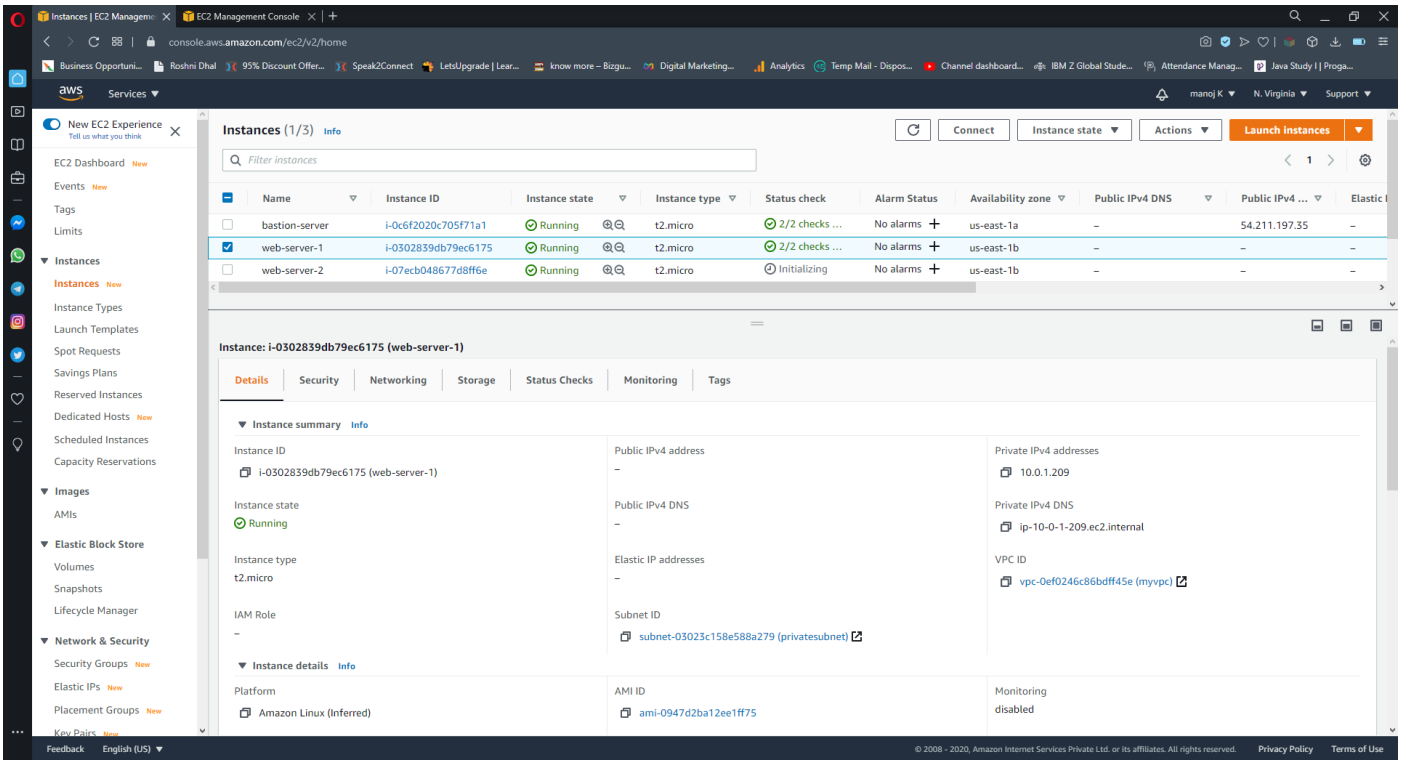
Selecting Security group from previous web-server:



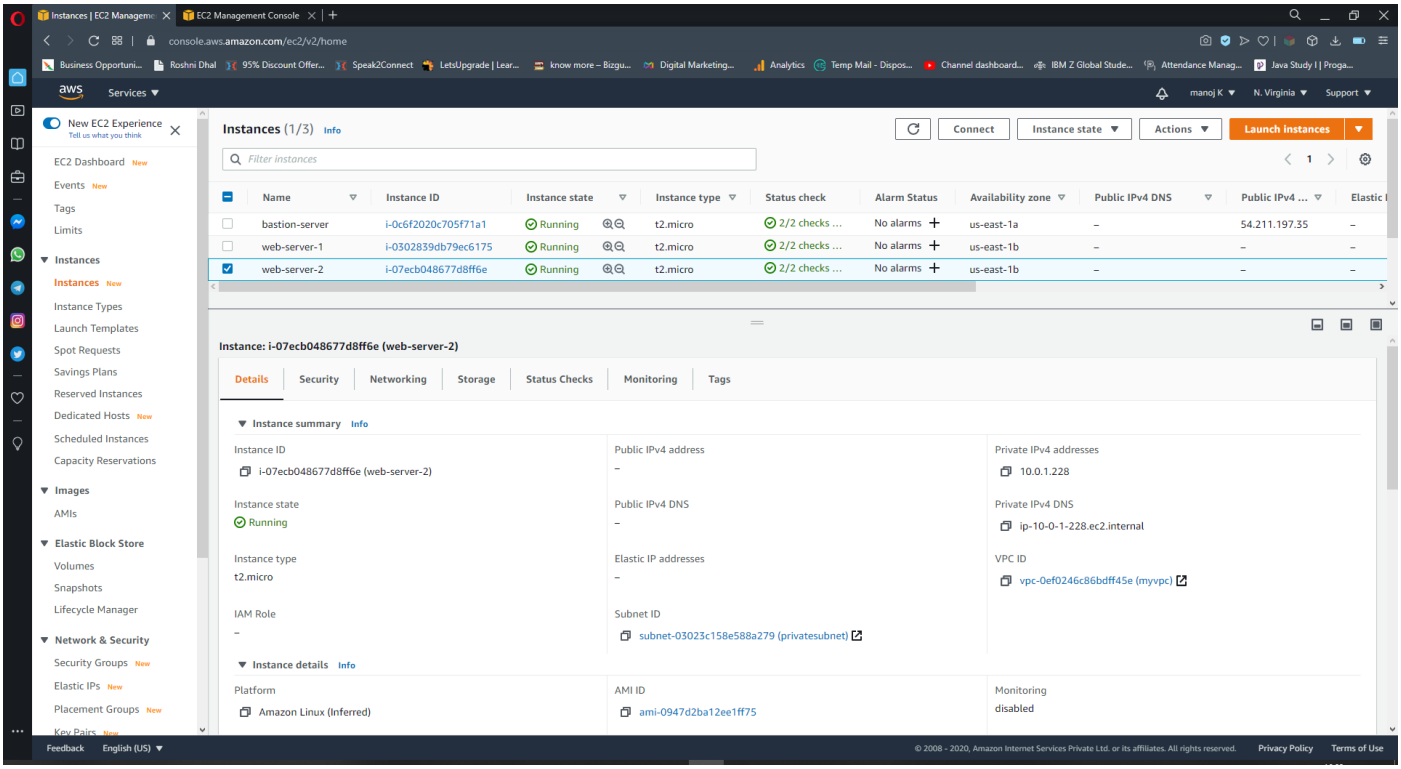
Launching the web-server 2:



Web-server 1 Details:

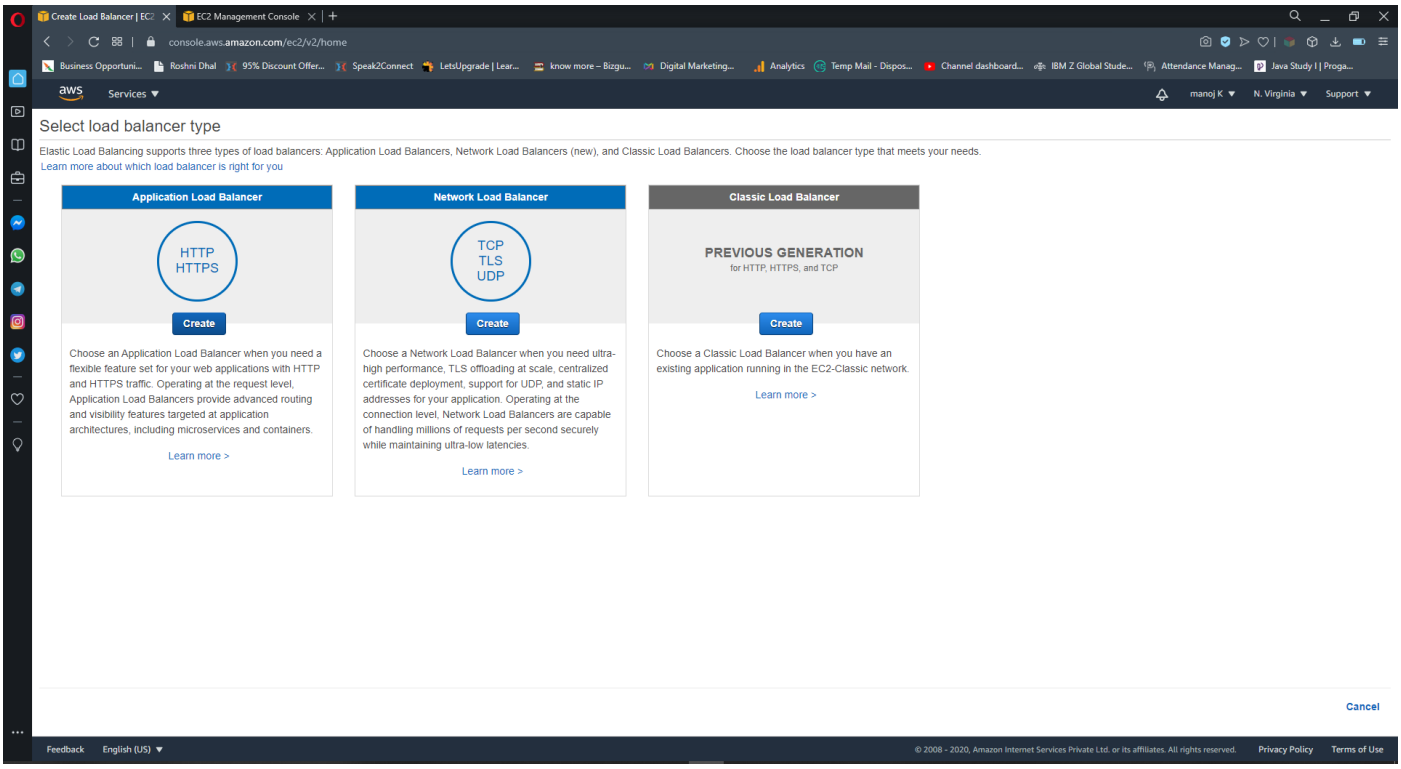


Web-server 2 Details:

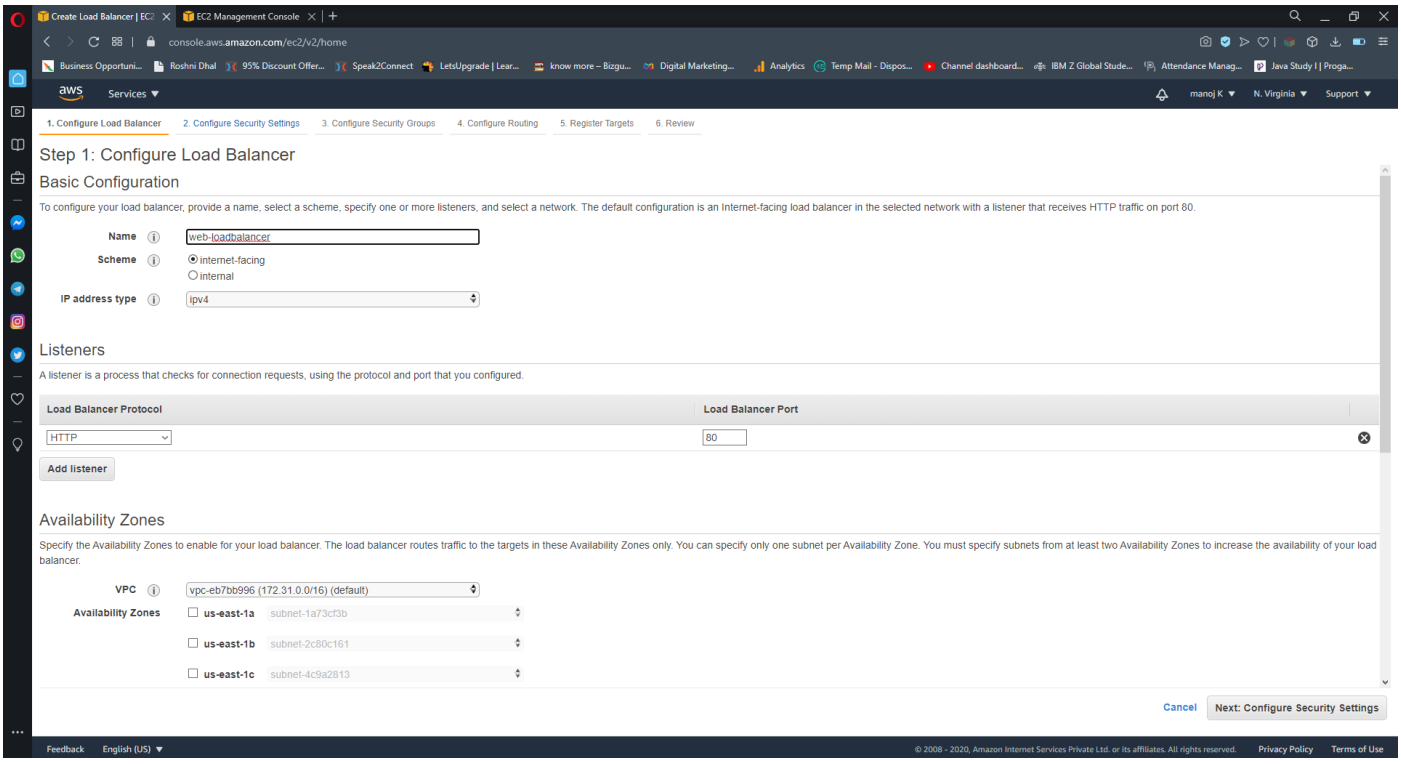


After creating web server

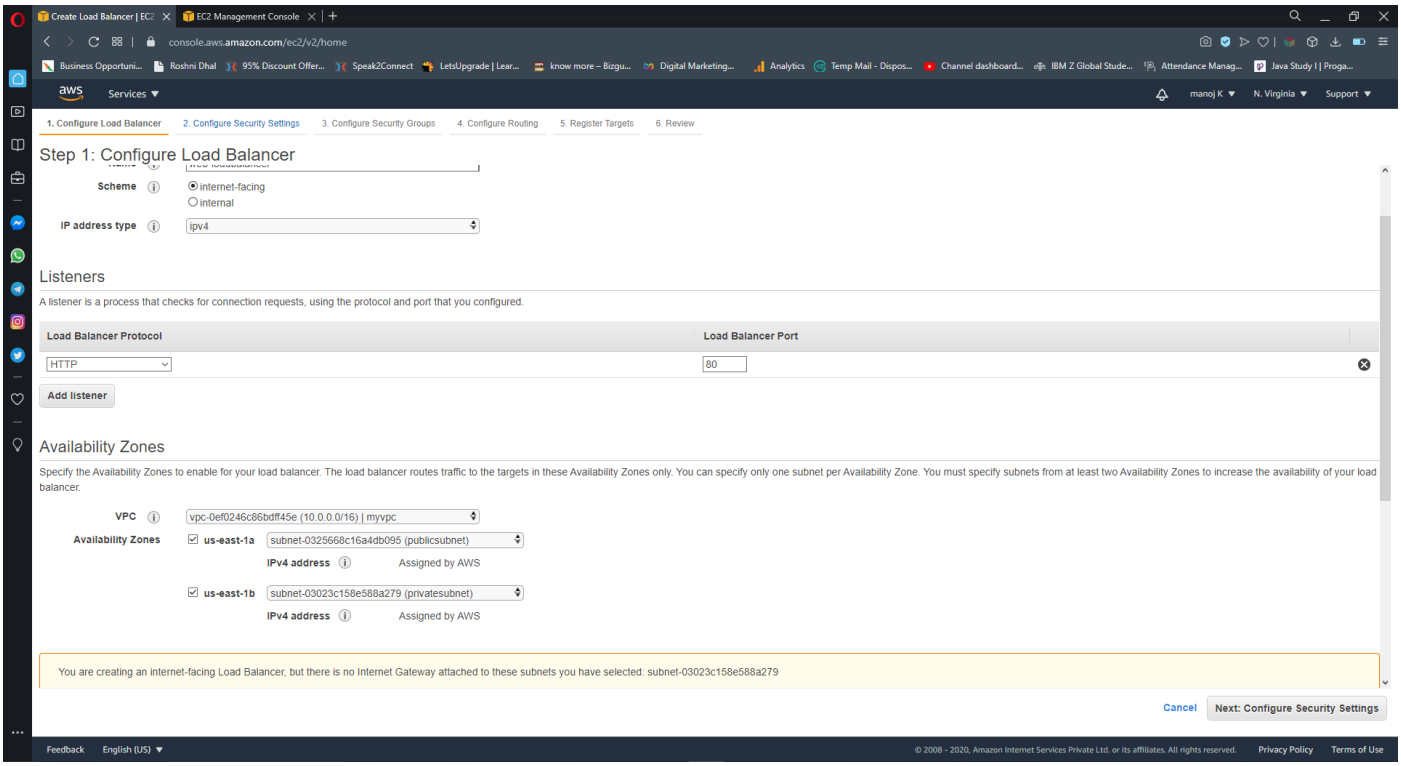
Let us creating a load balancer for web servers:



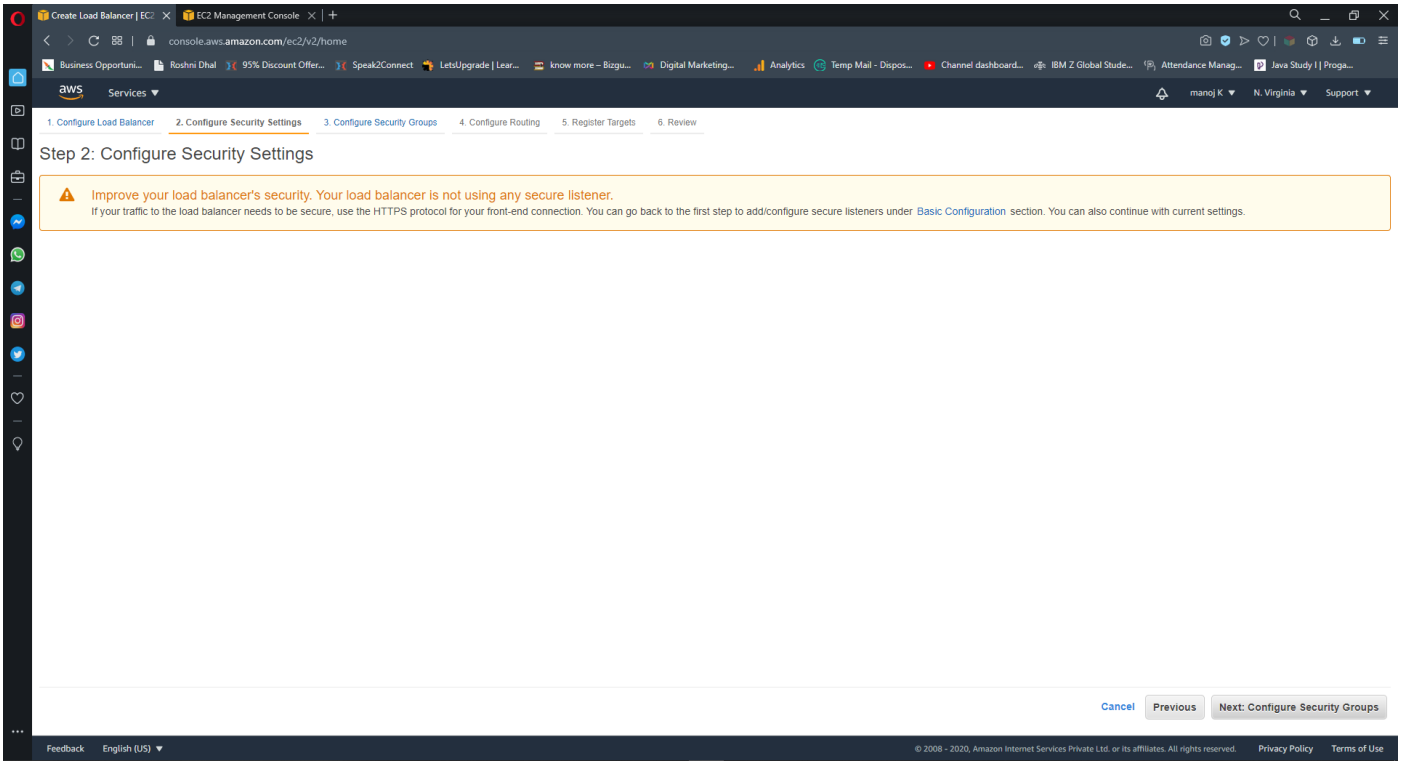
Configuration of load-balancer:



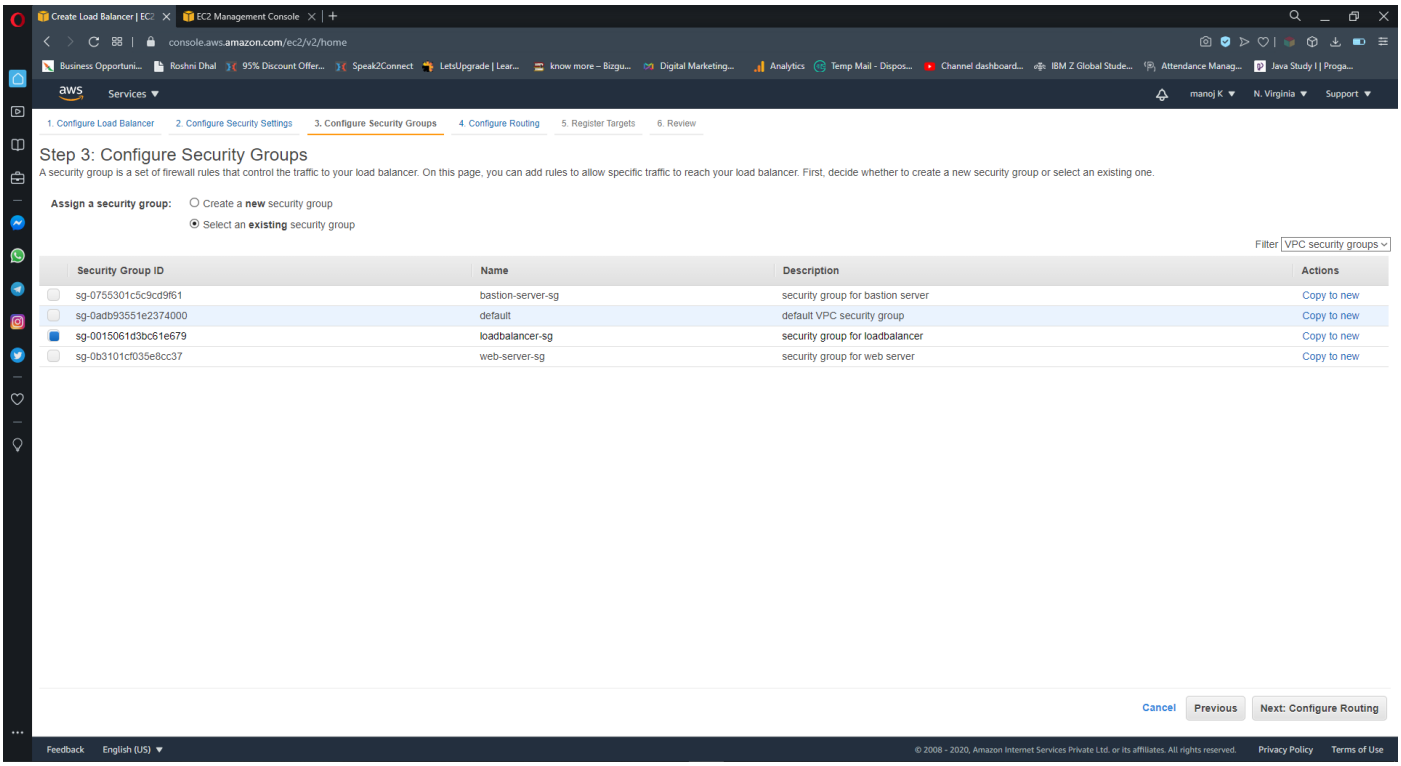
Configuration of load-balancer:



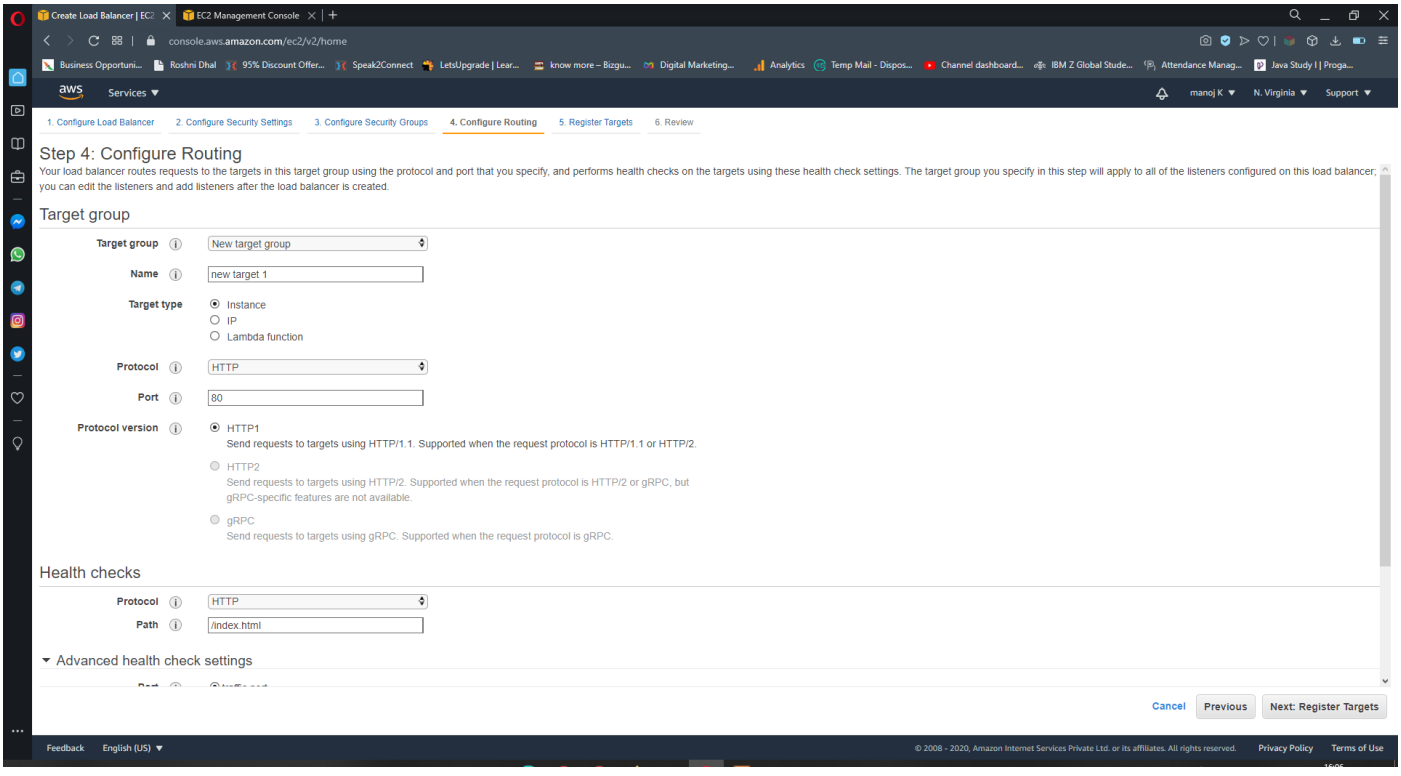
Security setting of load-balancer:



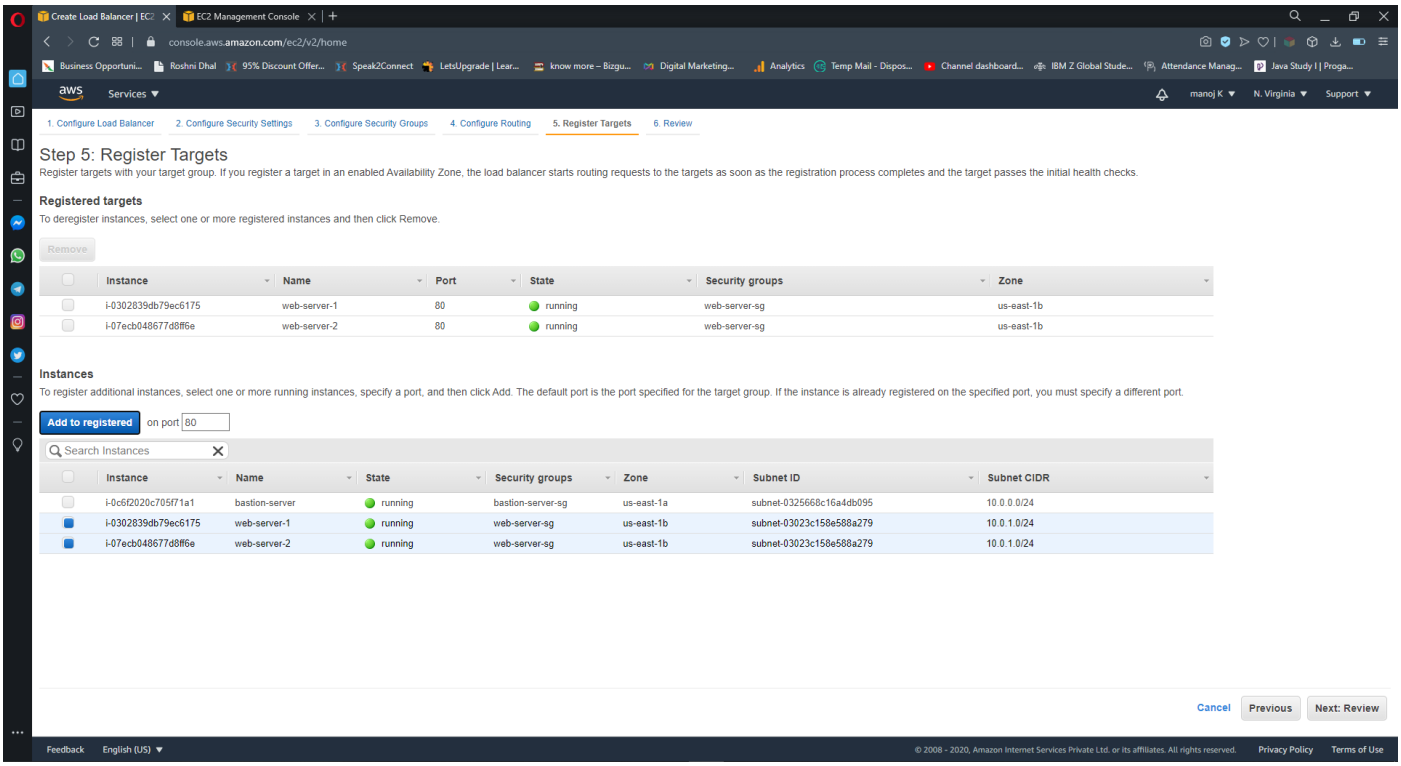
Selecting Security group for load balancer:



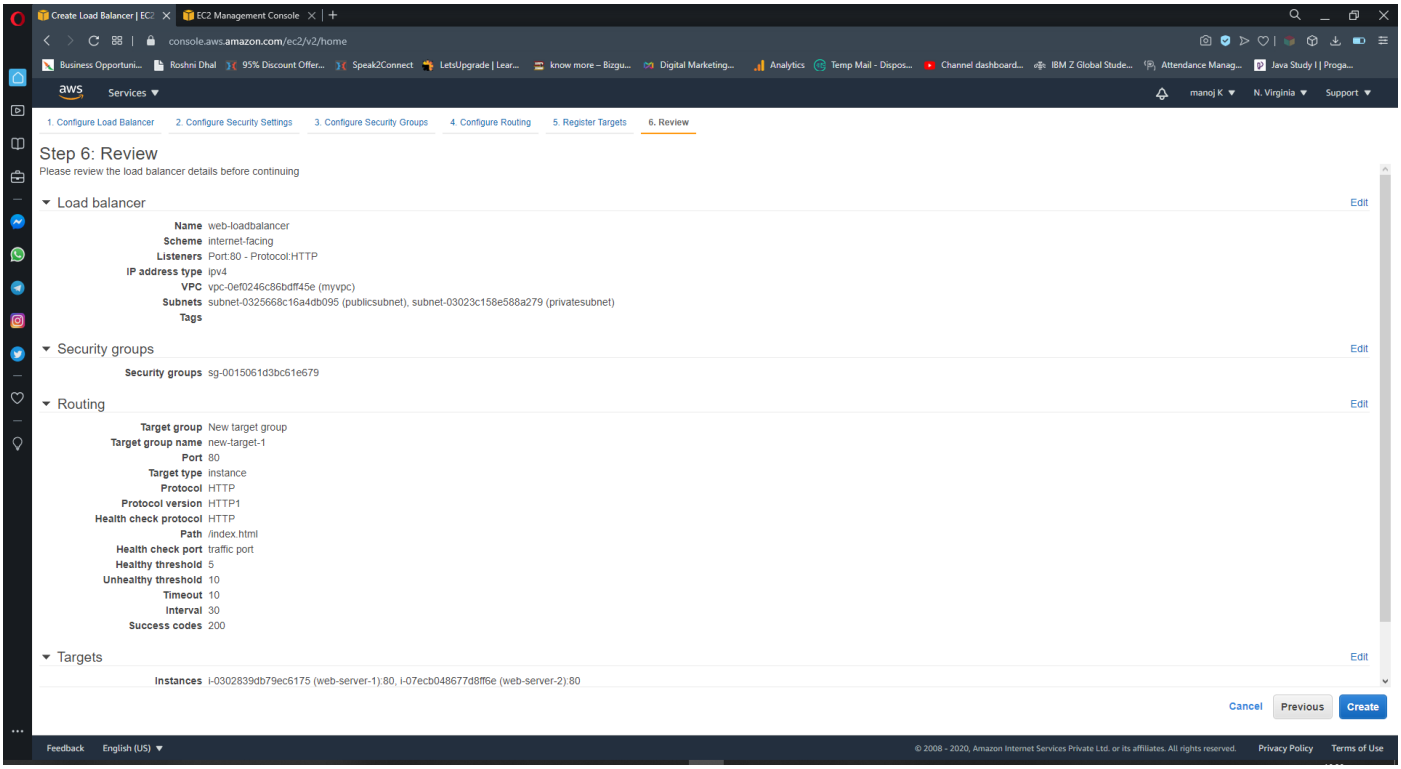
Configuring routing for load balancer:



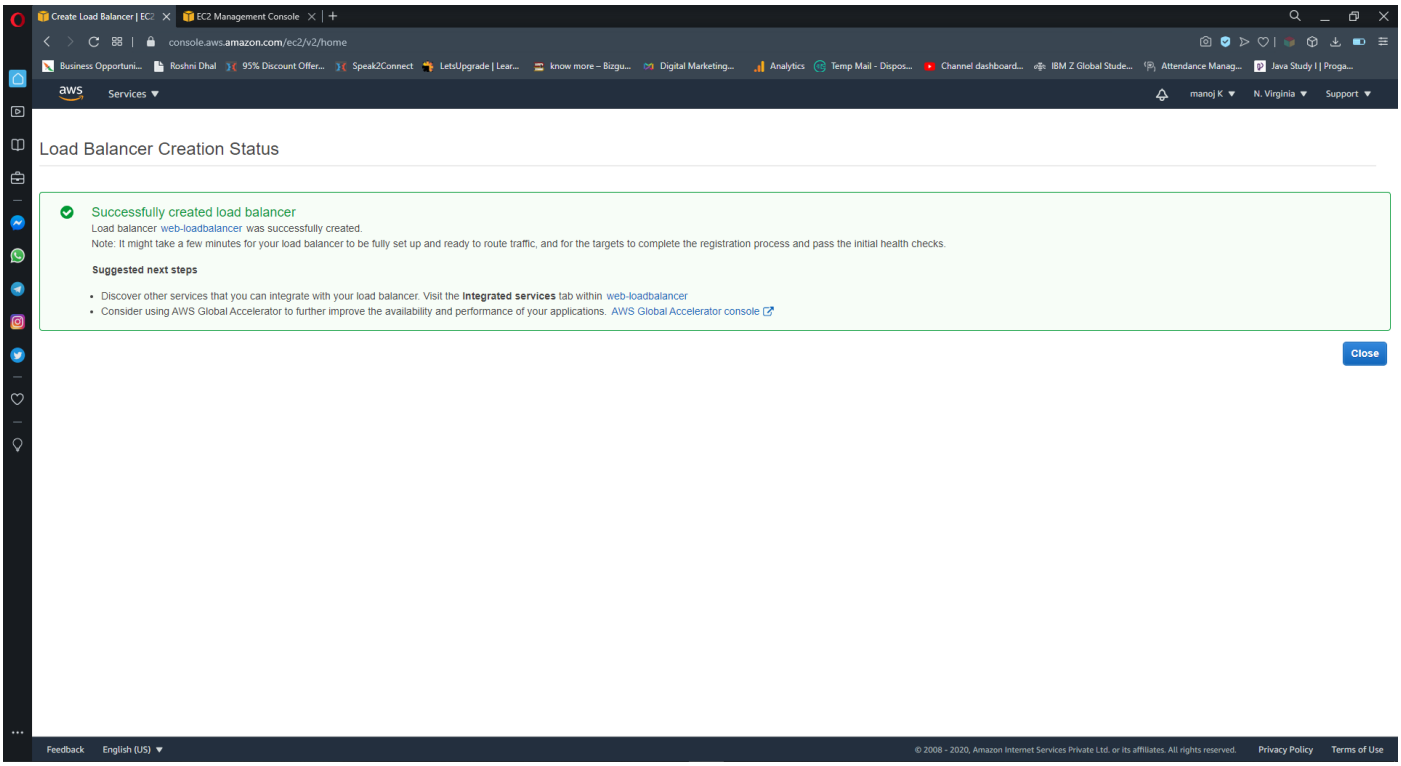
Registering targets for load-balancer:



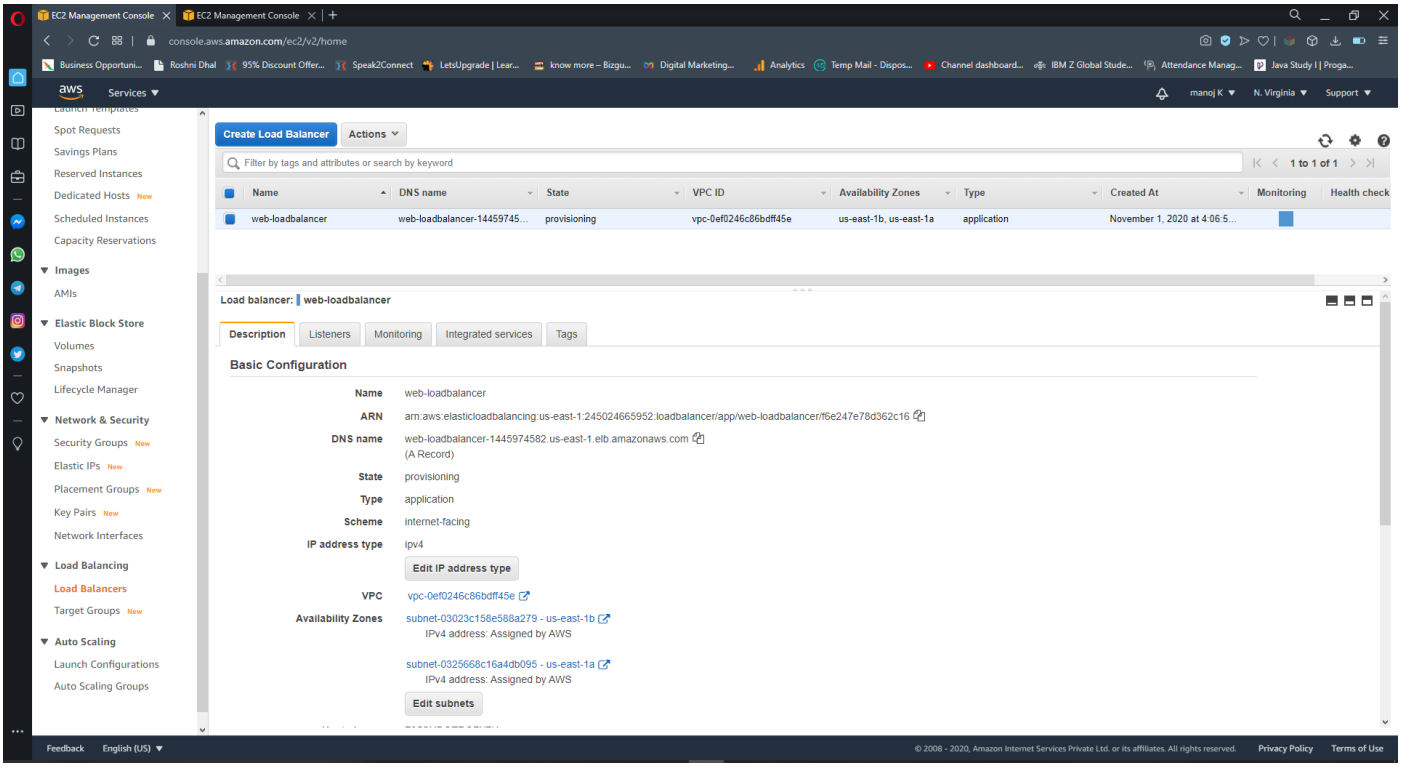
Reviewing Load balancer:



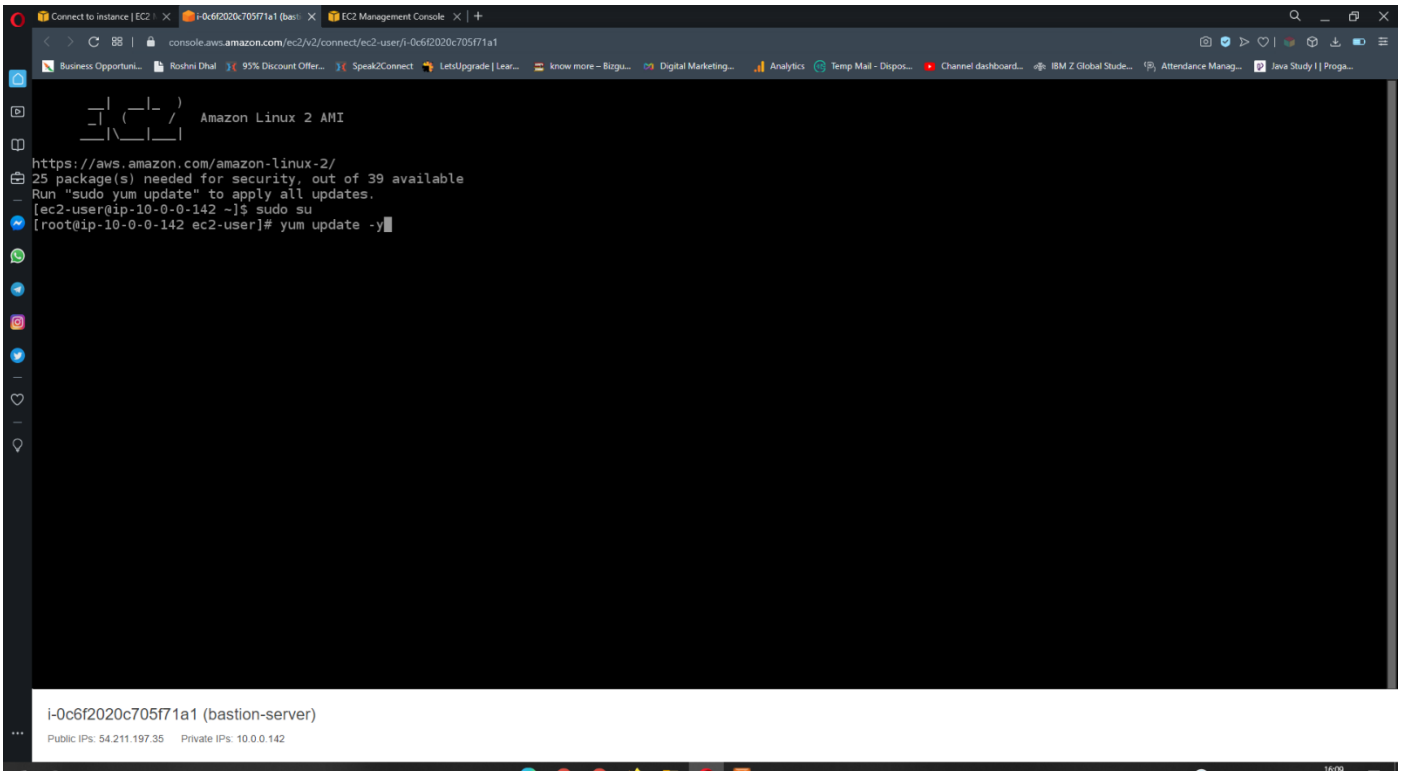
Successful Creation of Load Balancer:



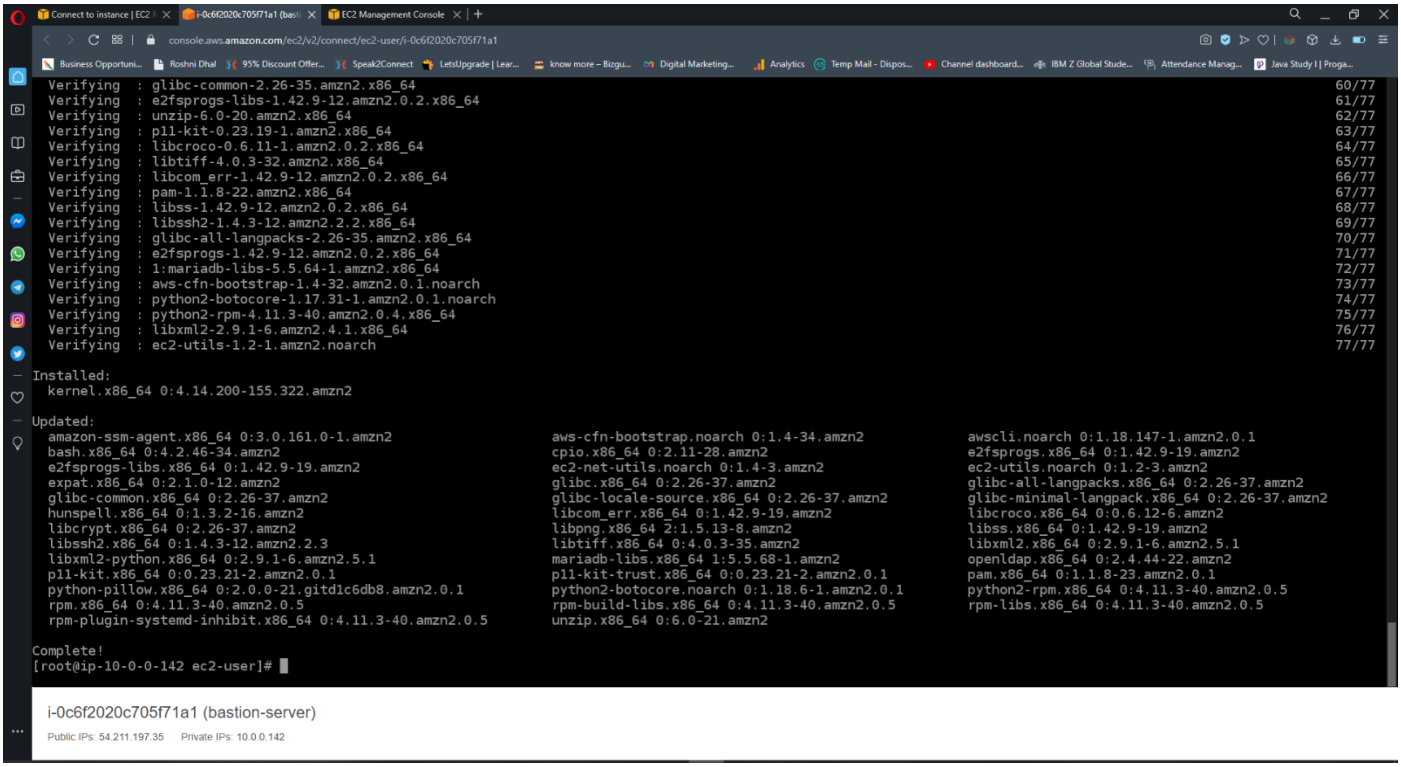
Load Balancer details:



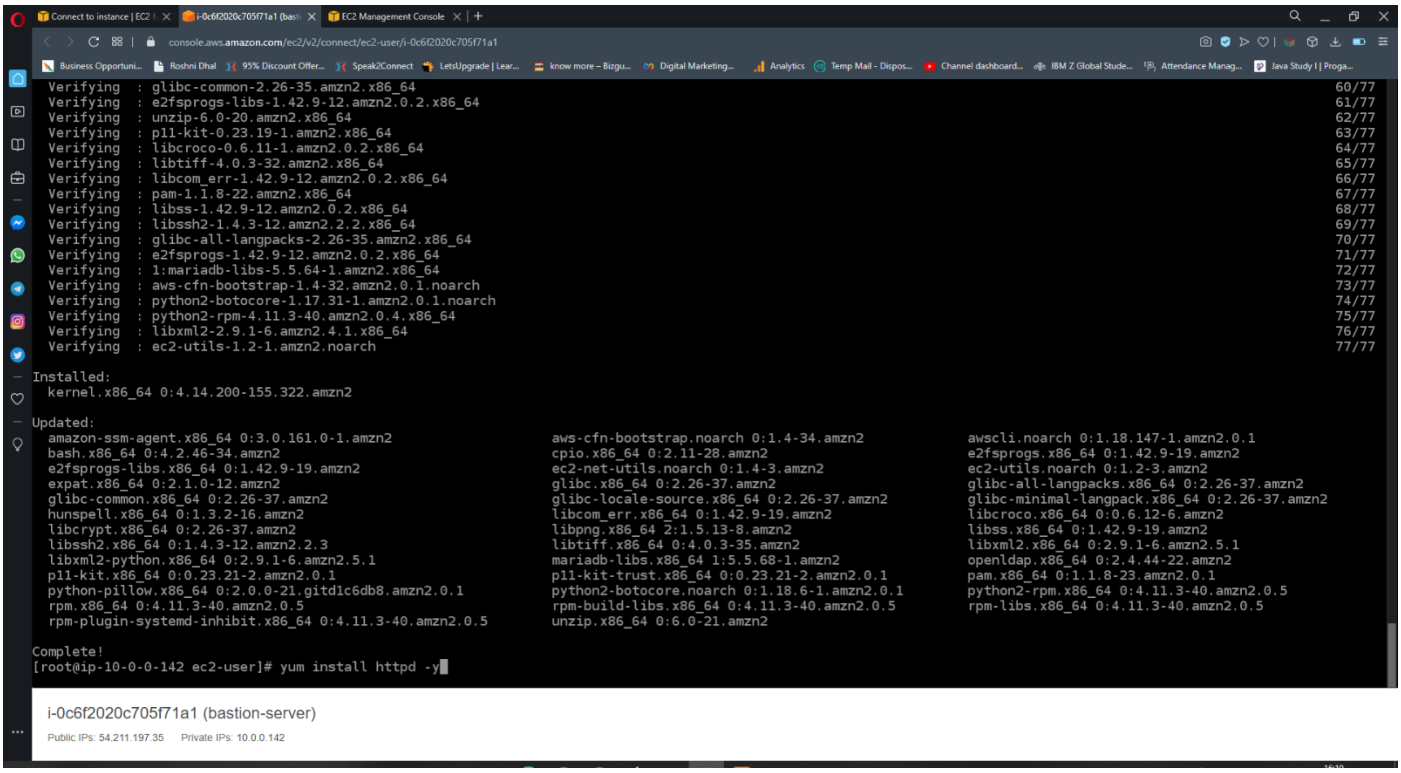
SSH into Public and Private EC2 Instance and Test Internet Connectivity:



Updating linux:

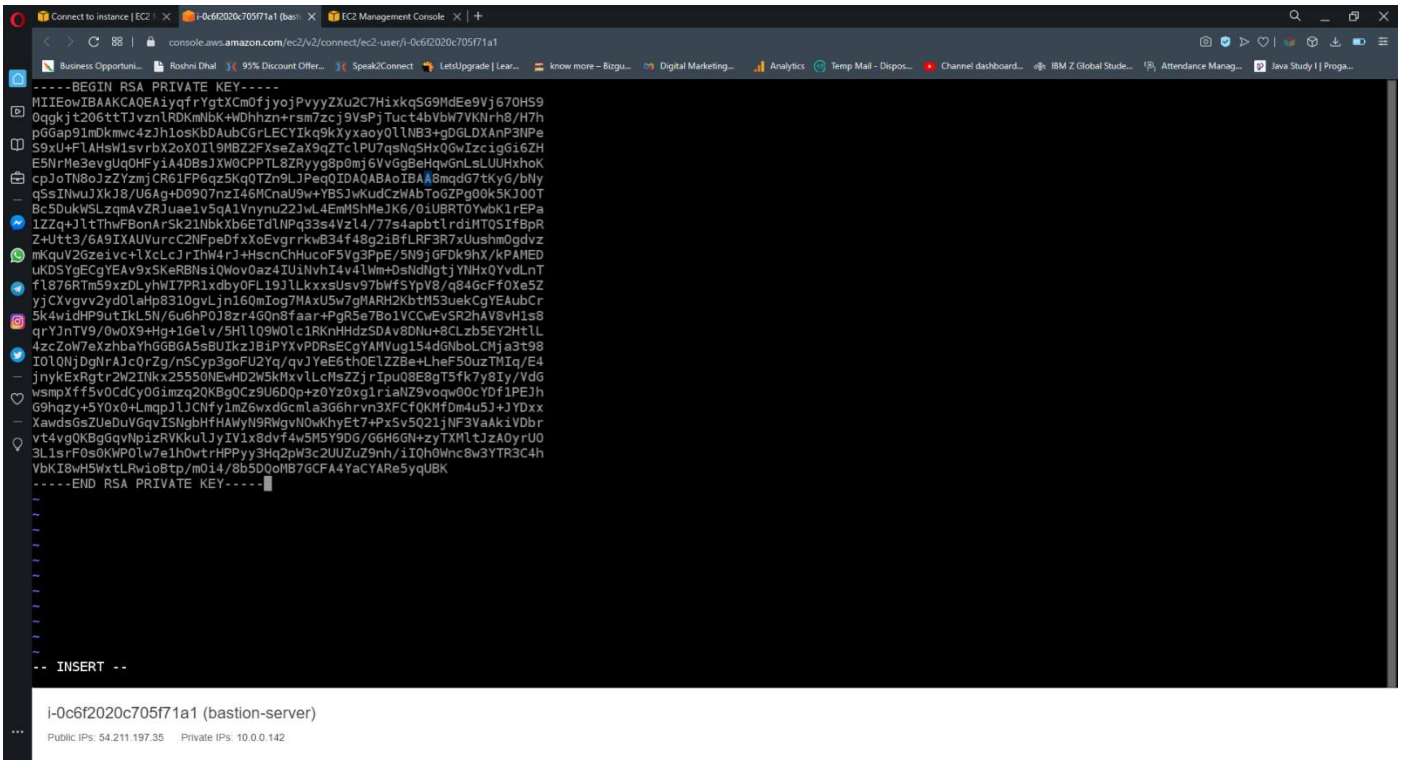


Installing HTTPD:

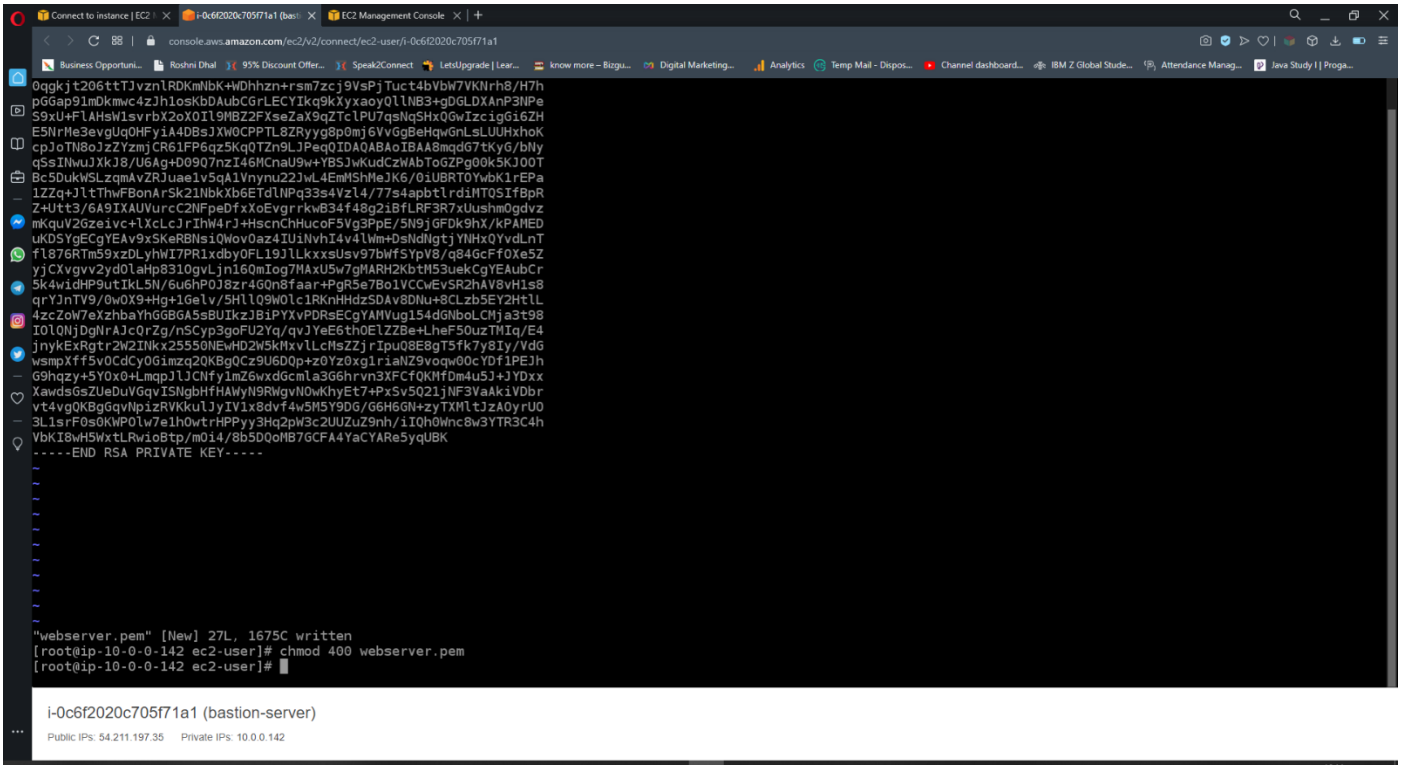


Created file using “vi webserver.pem” command

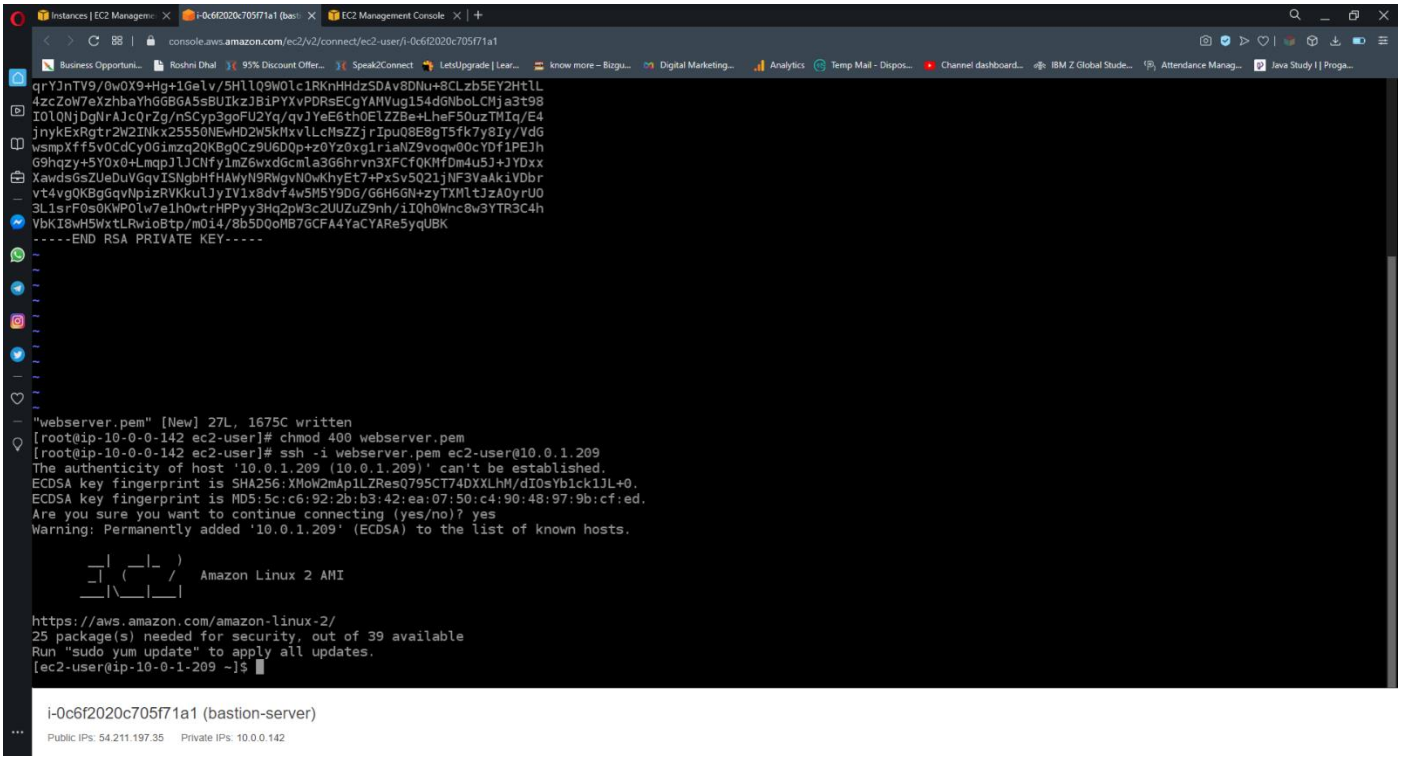
And copy pasted pem file into it:



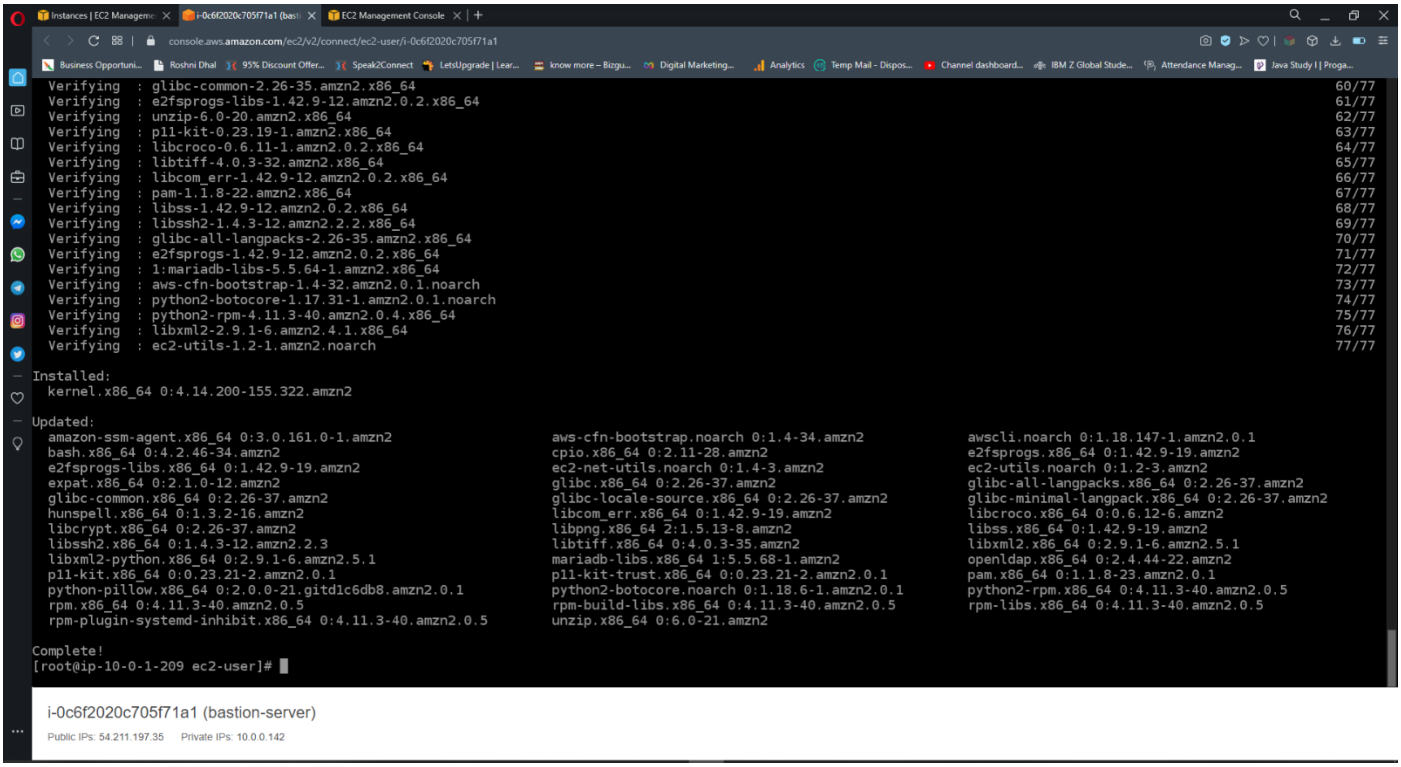
Changing permission of webserver.pem file:



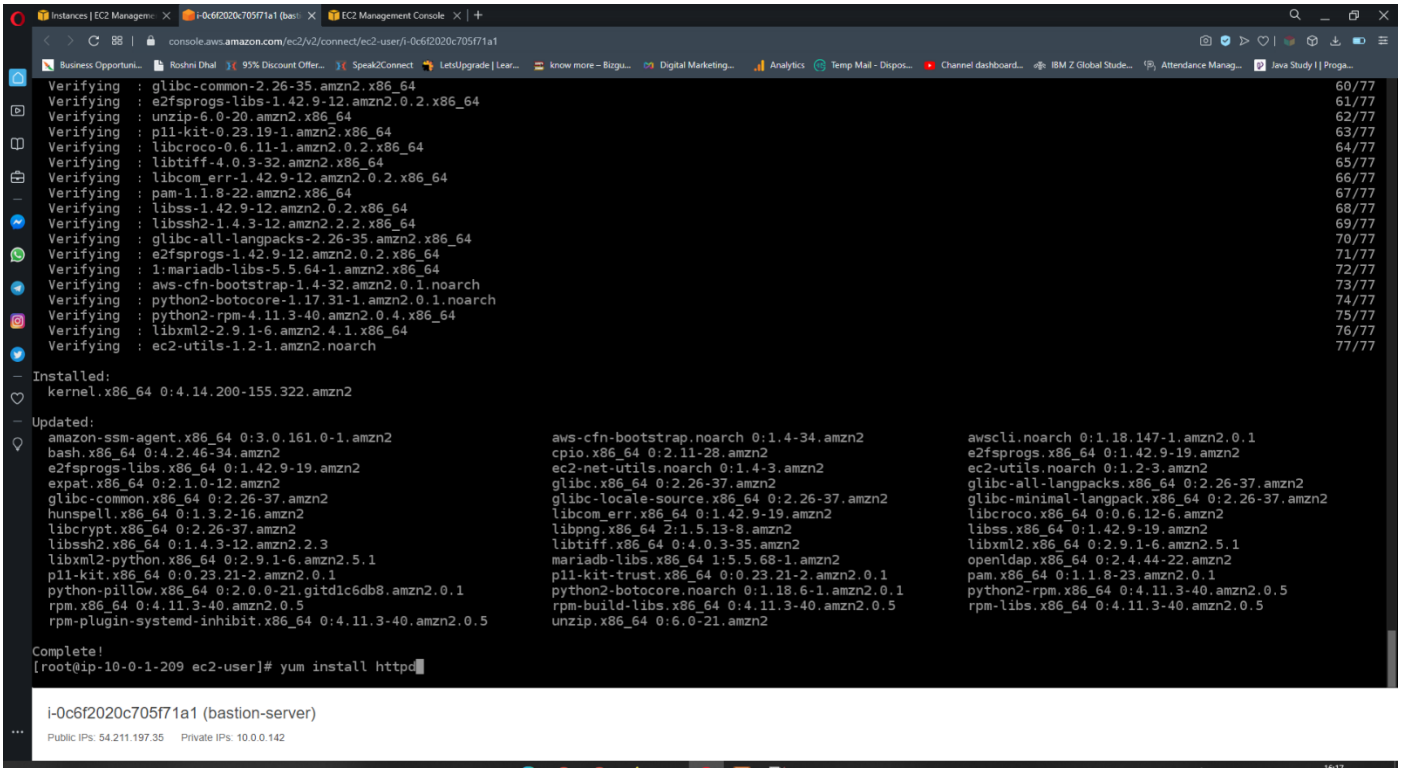
SSH into to private subnet using public subnet:



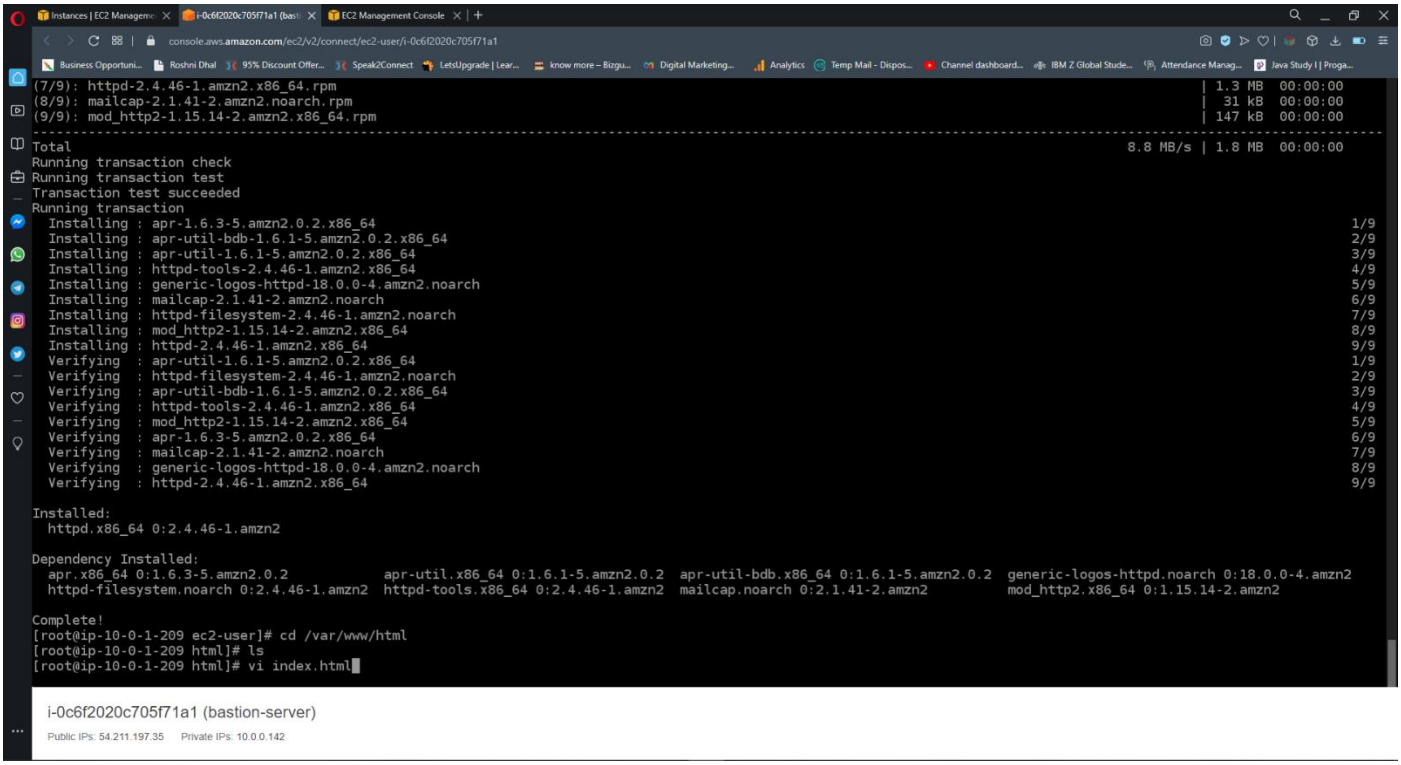
Updated to linux in webser 1:



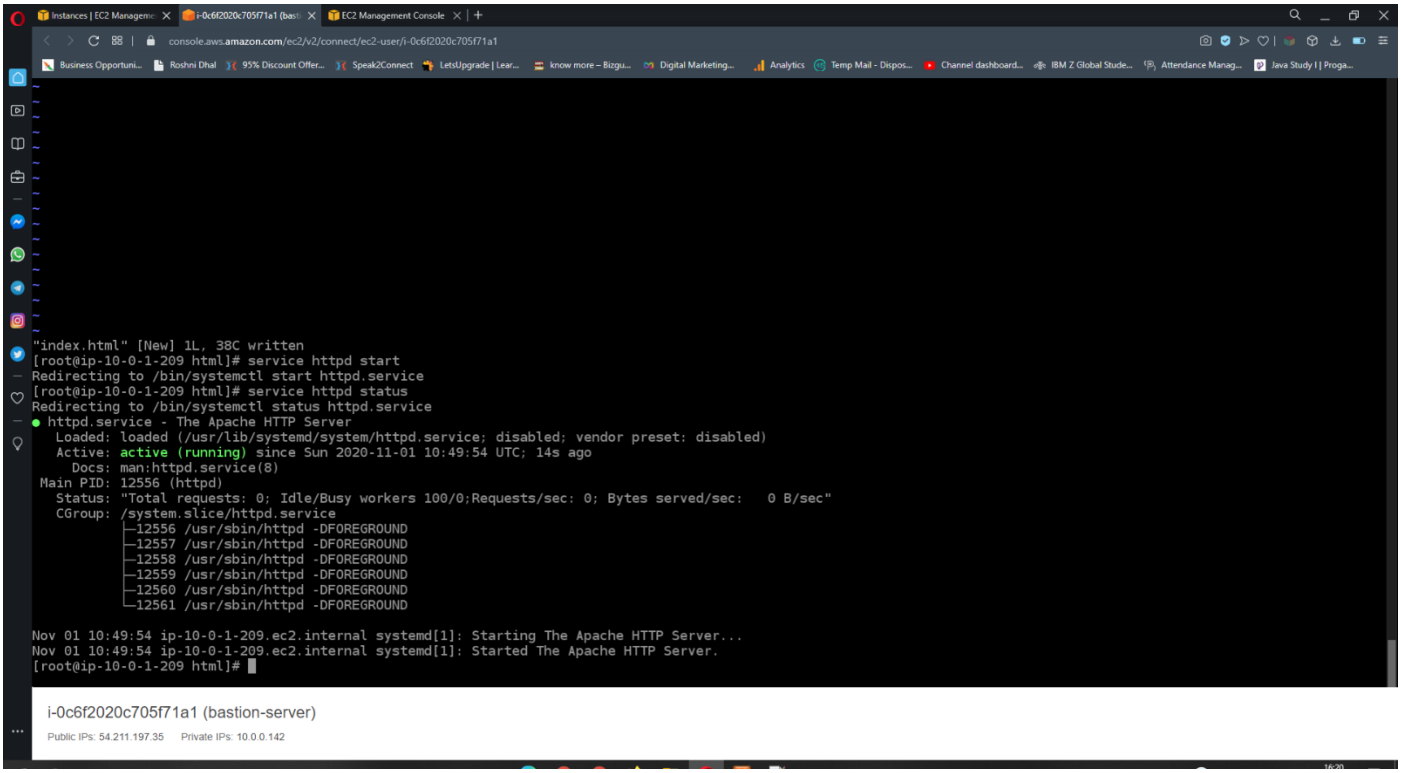
Installing HTTPD in web-server 1:



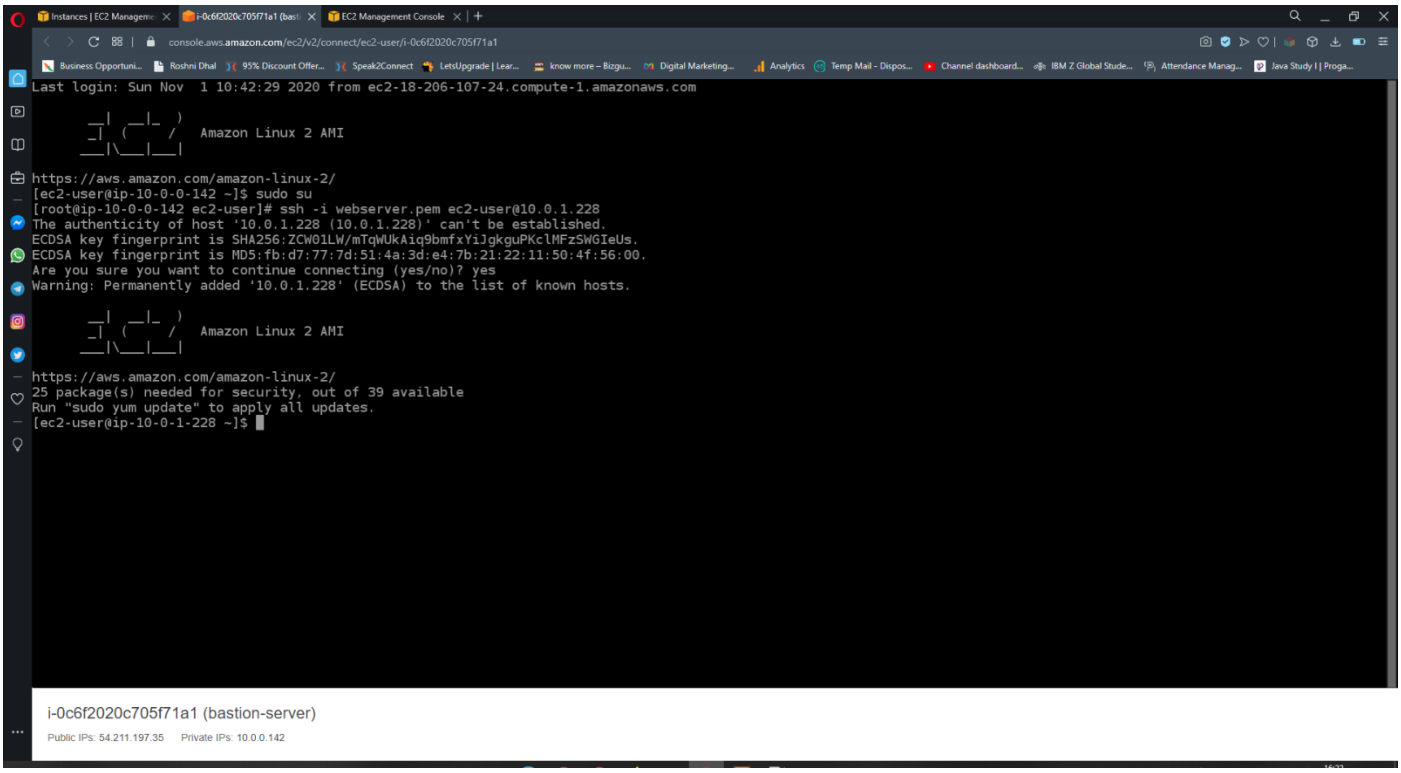
Creating “index.html” file in webserver 1:



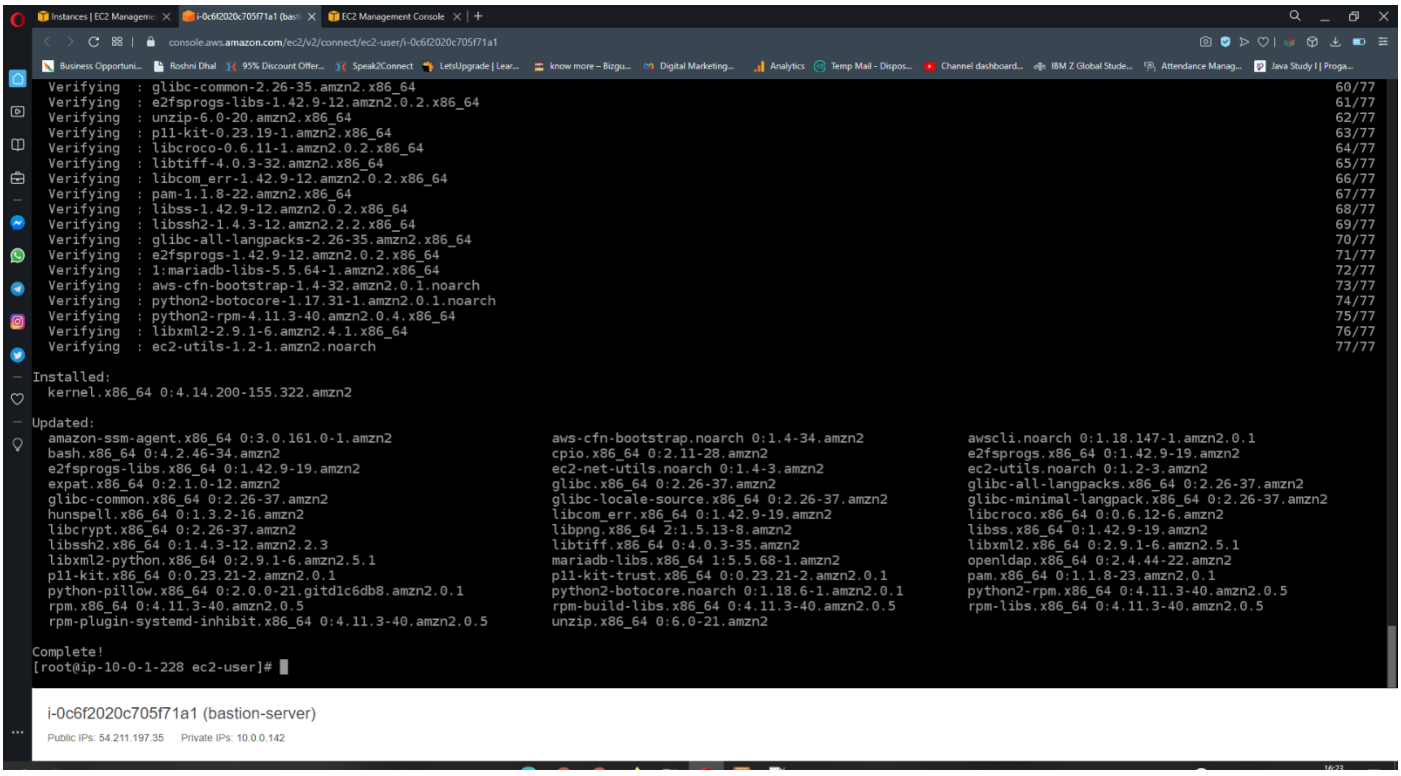
Started httpd service and checking status in web server 1:



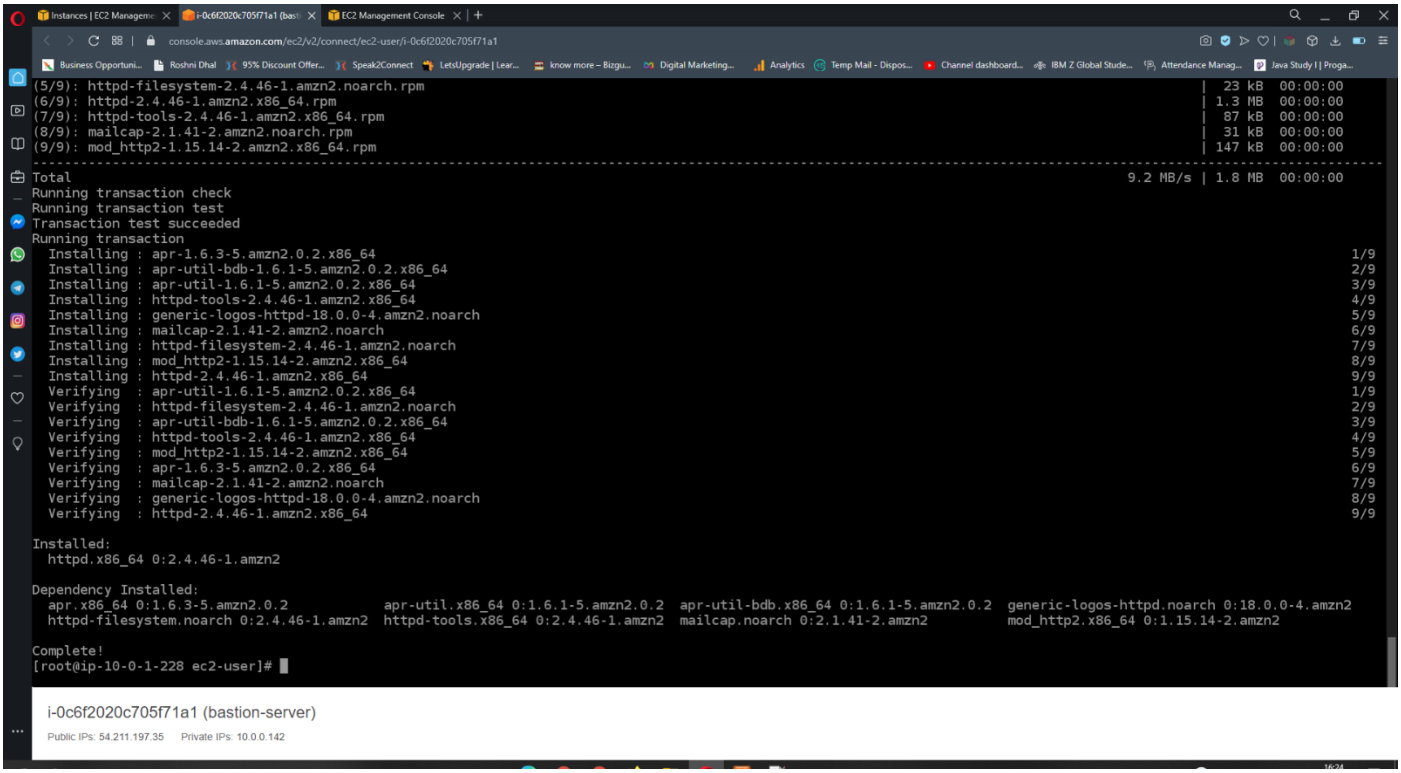
SSH into web server 2:



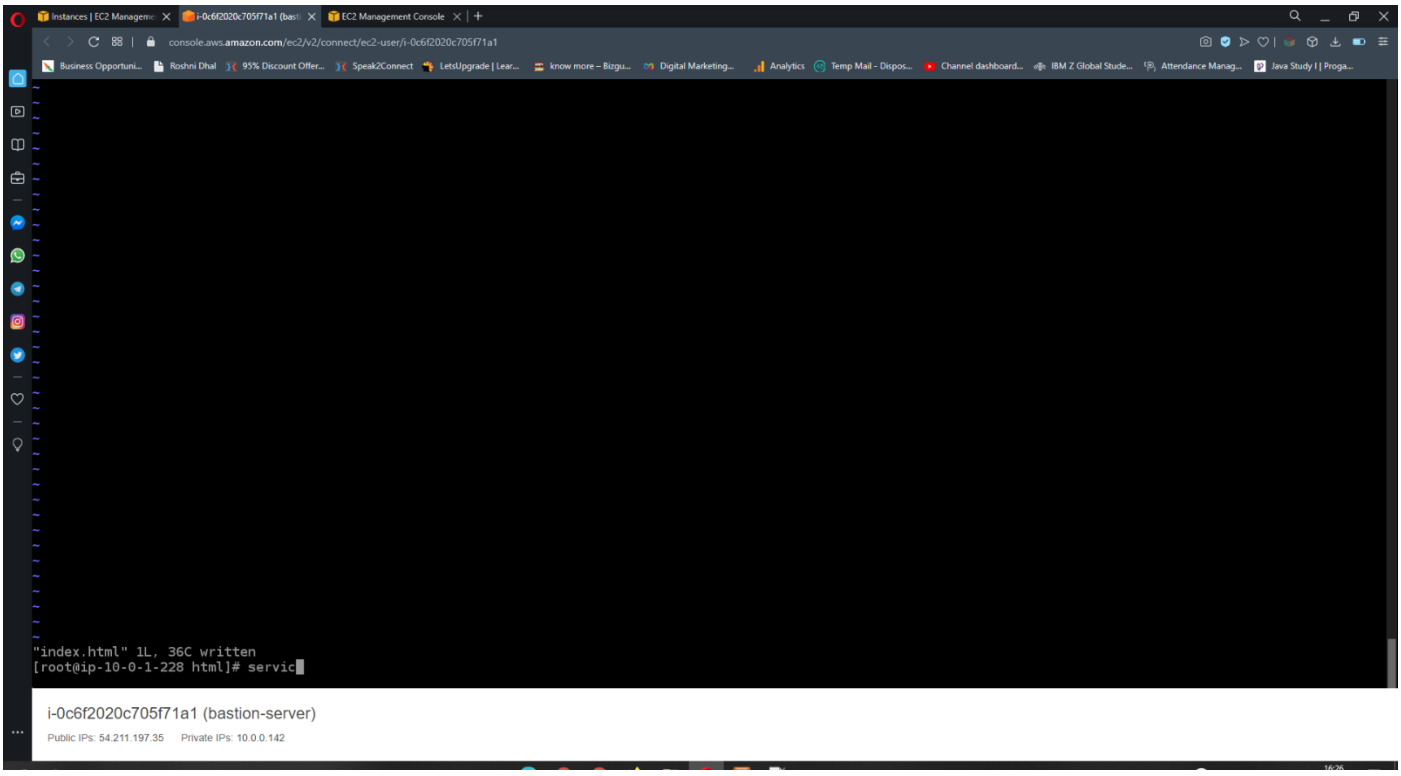
Updating the web server 2:



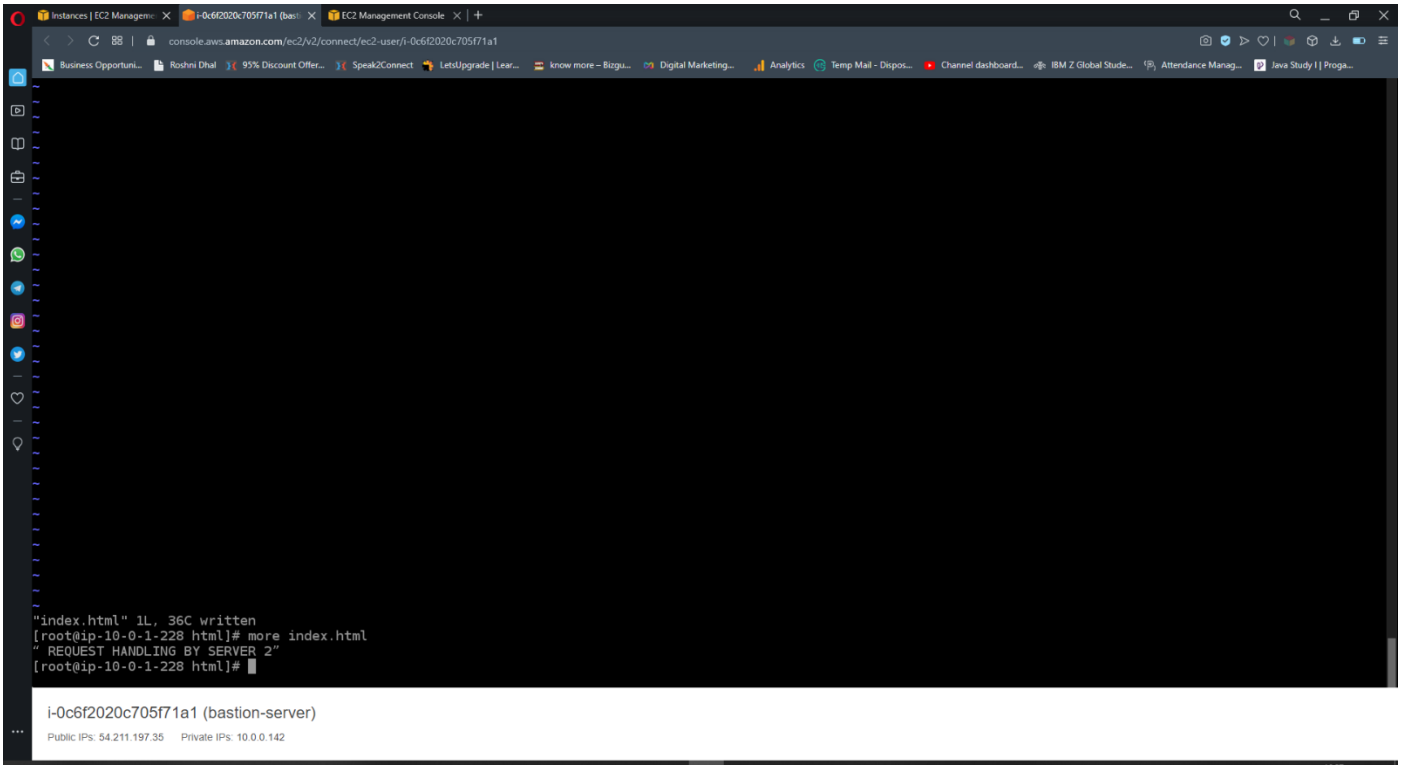
Installing HTTPD in web server 2:



Creating “index.html” file in webserver 2:



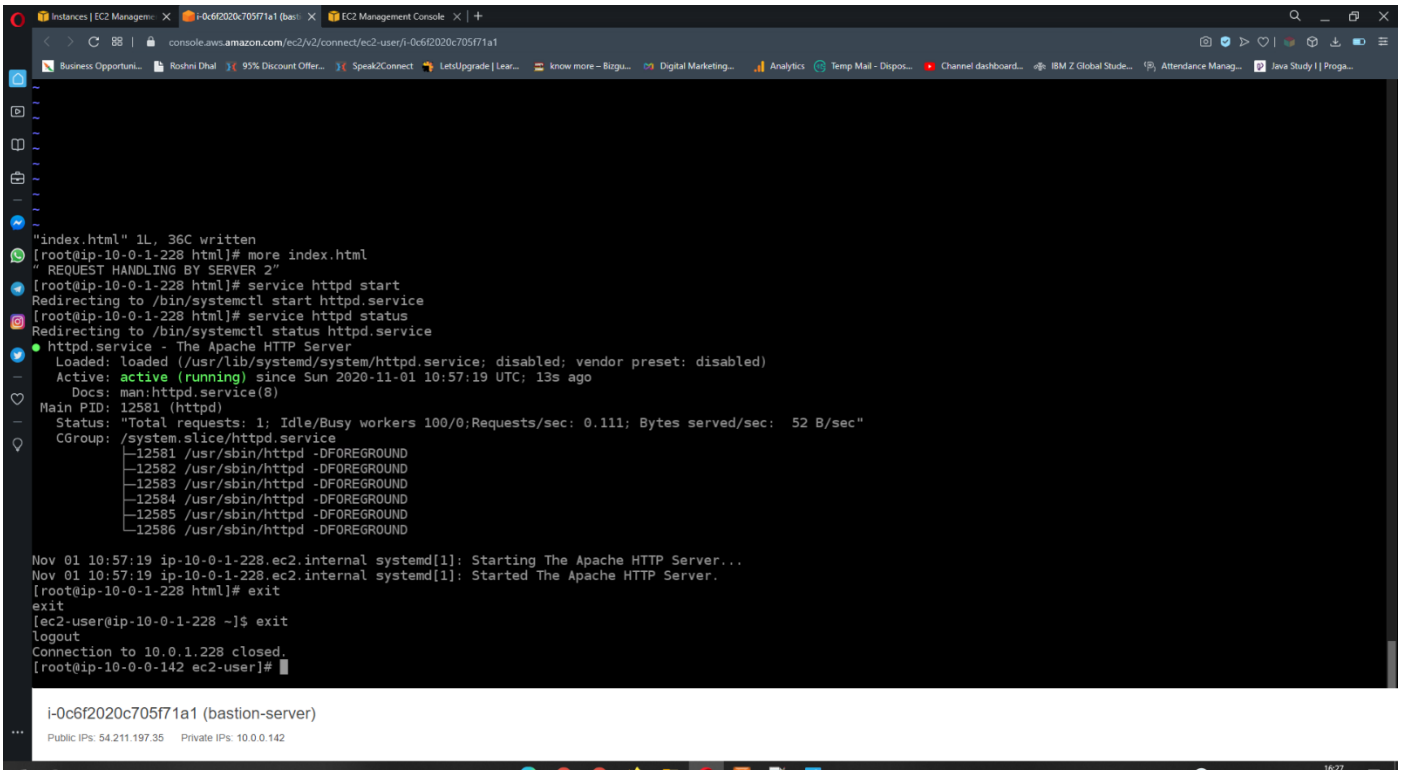
Message when web-server 2 is accessed:



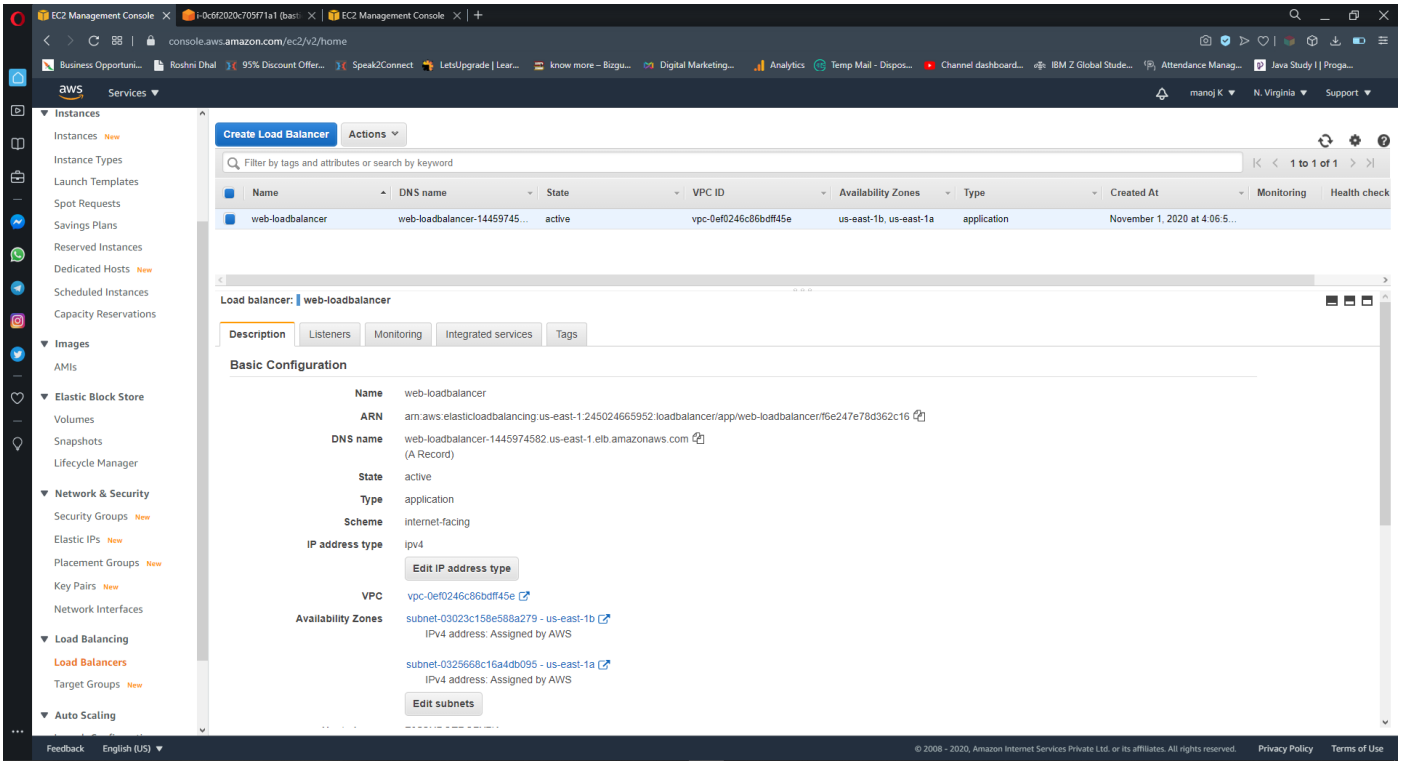
Started httpd service and checked status in web-server 2:

Finally exited from sudo user and also log out from webserver 2

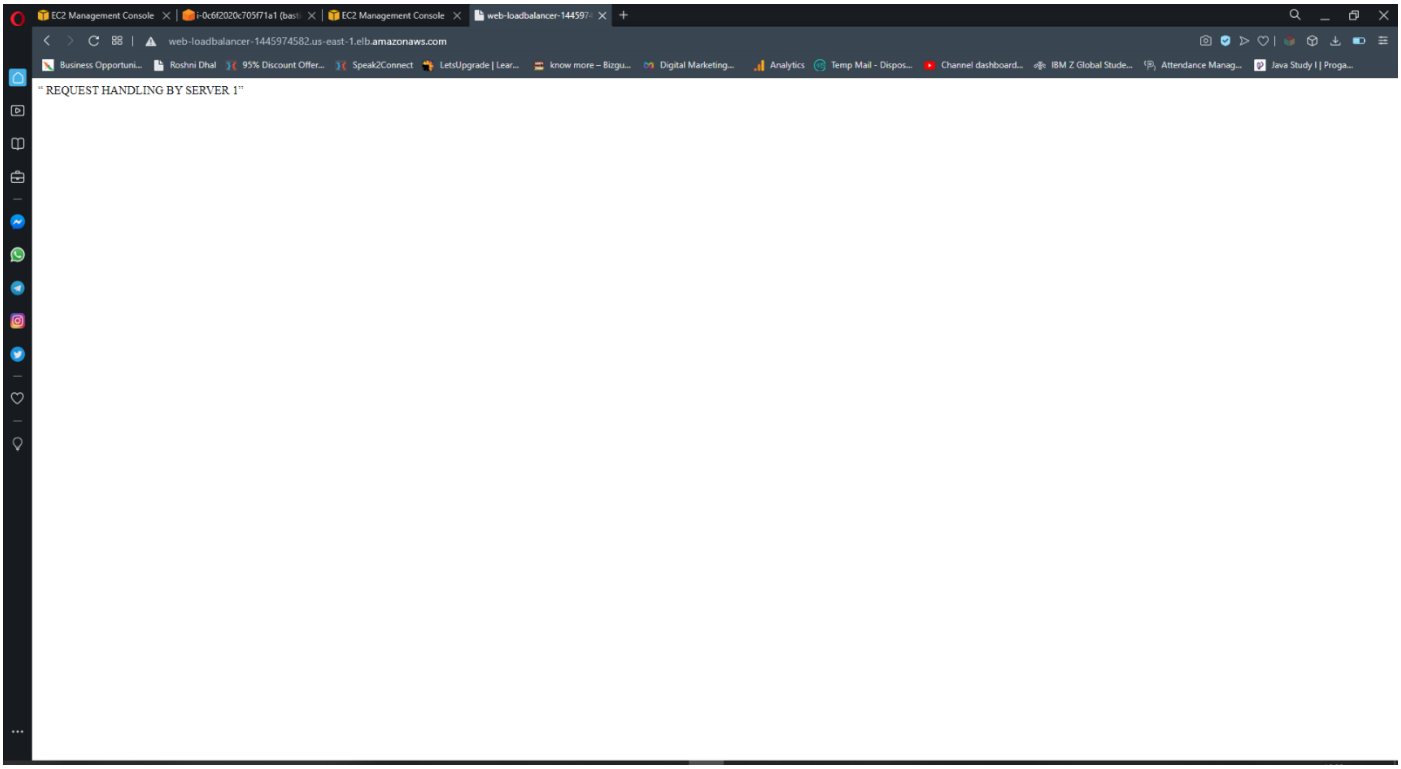
Using exit command.



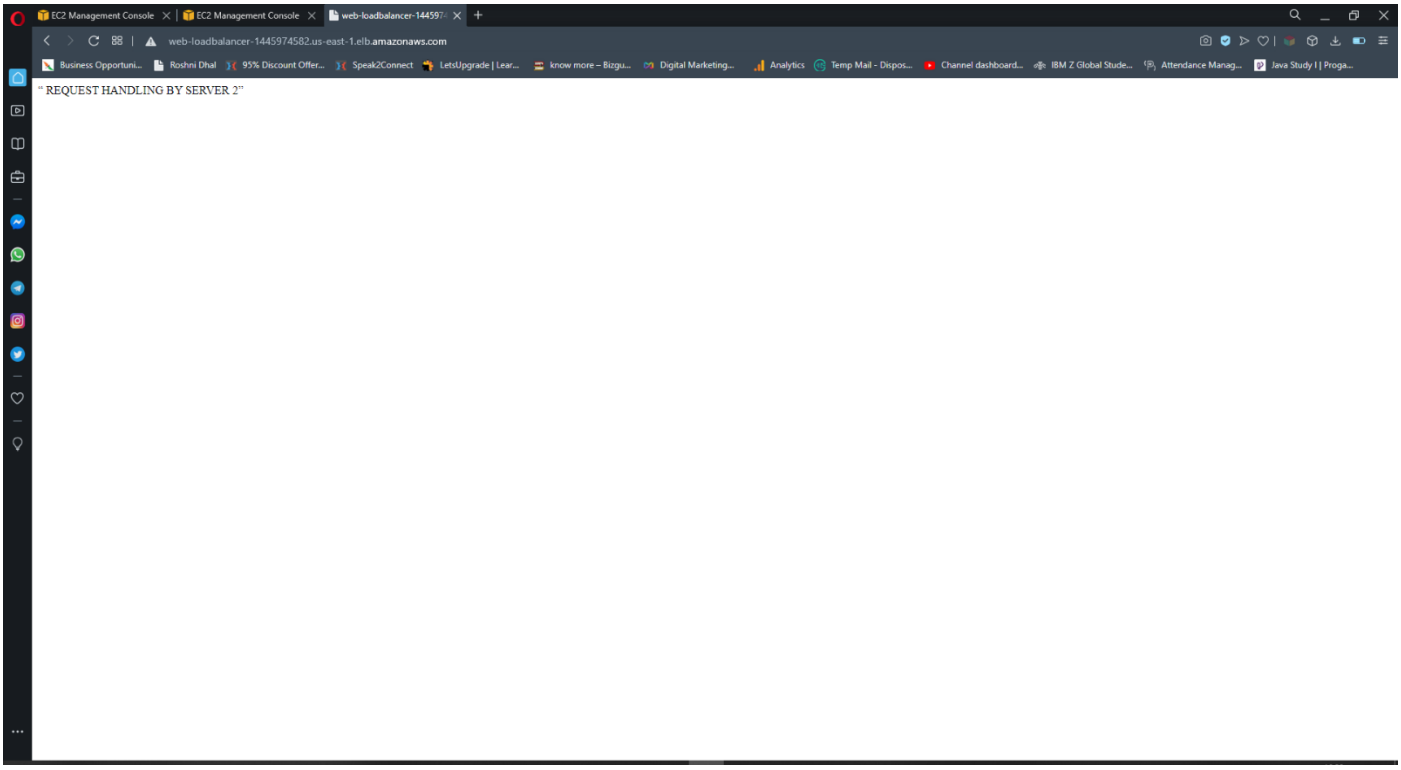
Load-balancer Details:



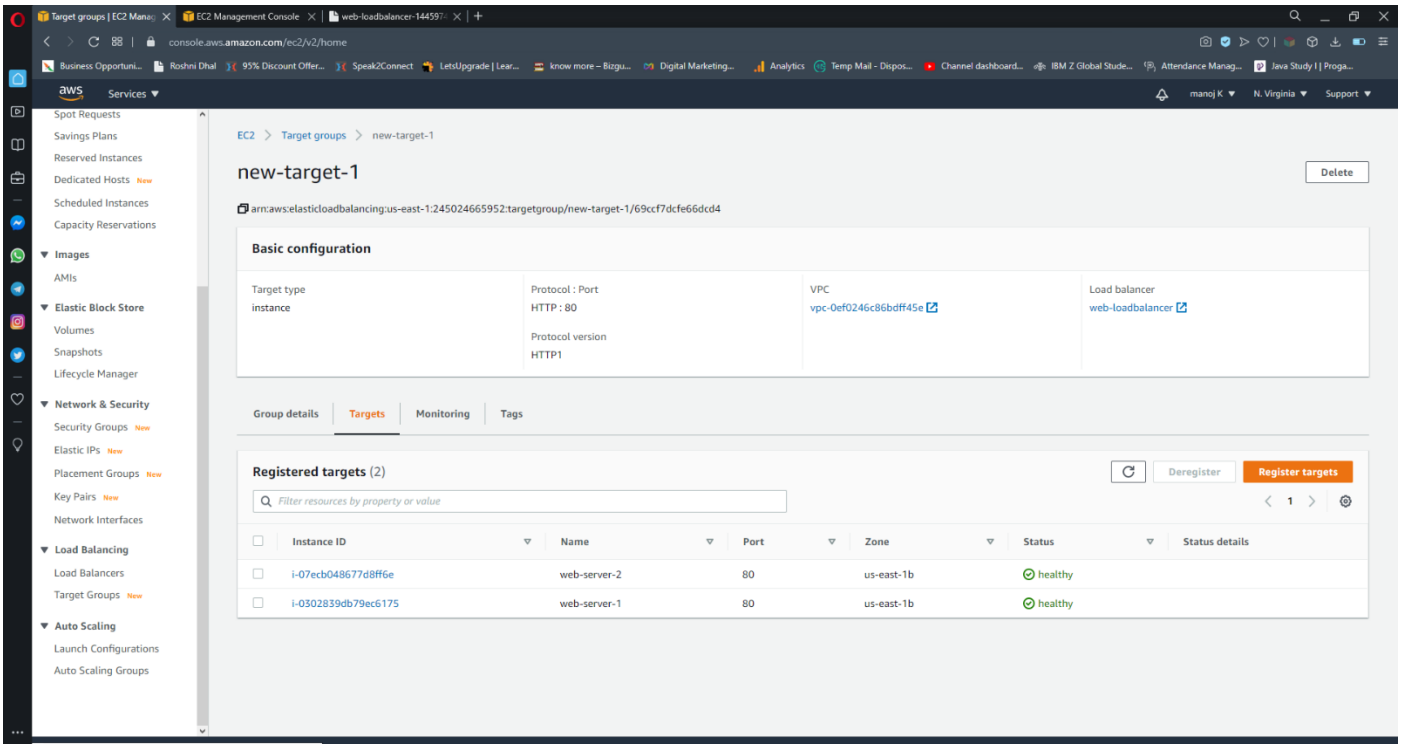
Request from web server 1 when copy pasted DNS of Load balancer:



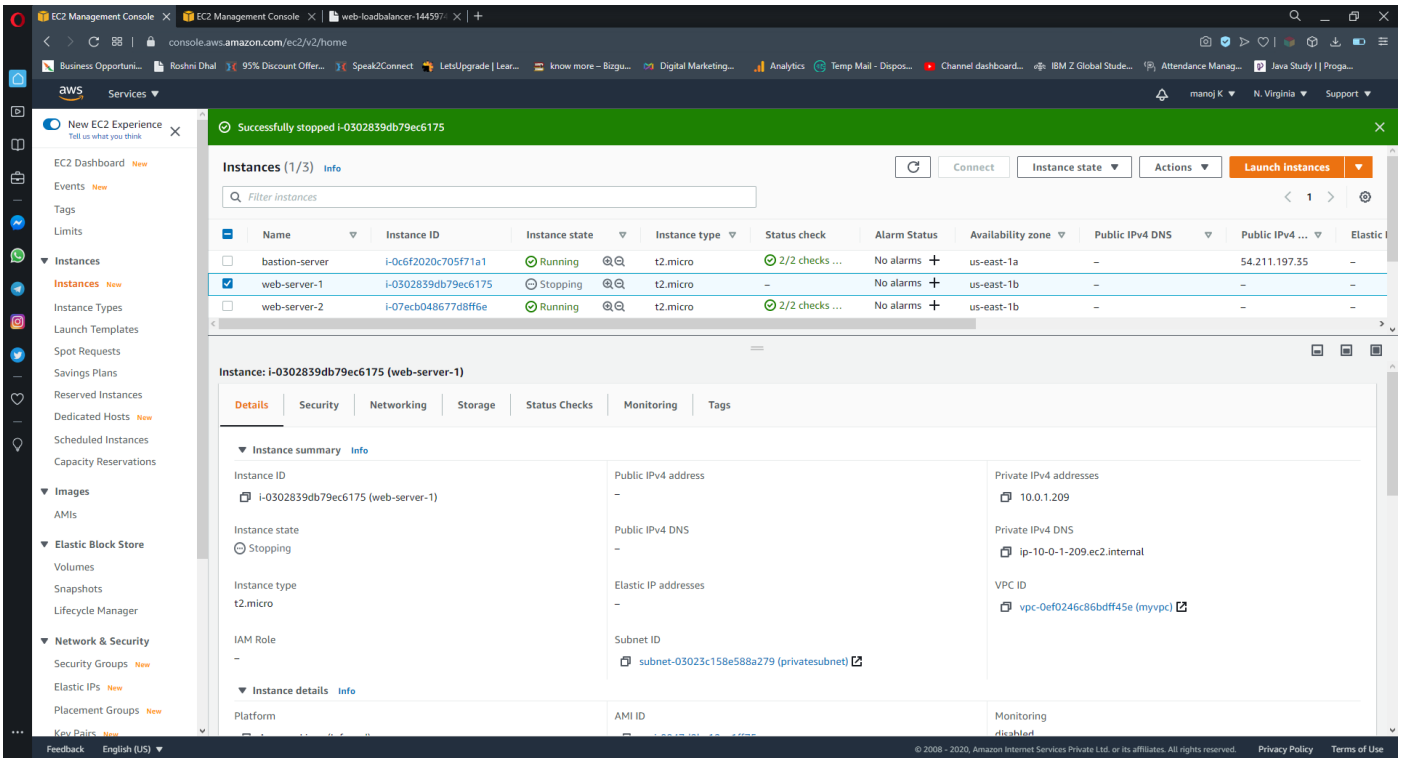
Request from web server 2 when copy pasted DNS of Load balancer:



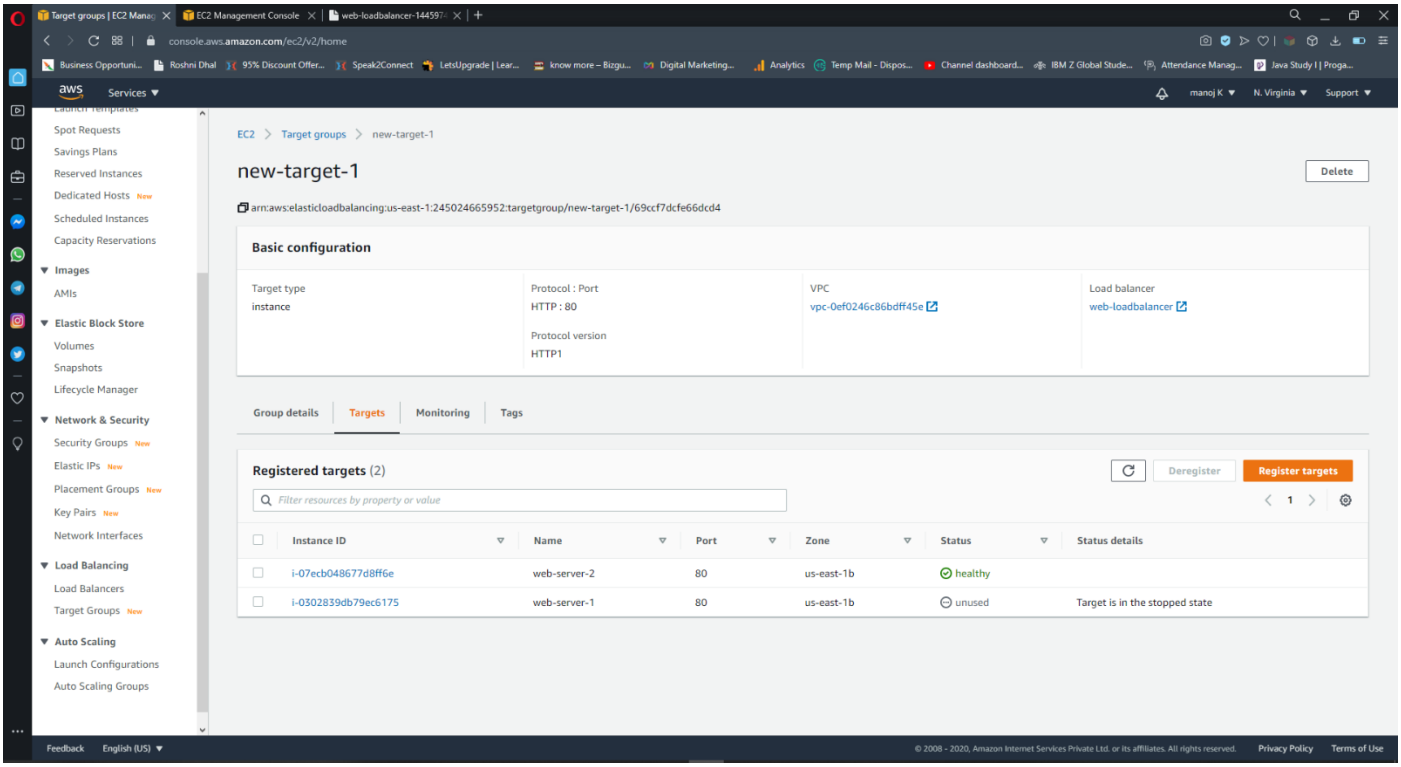
Health of web-server before being stopped:



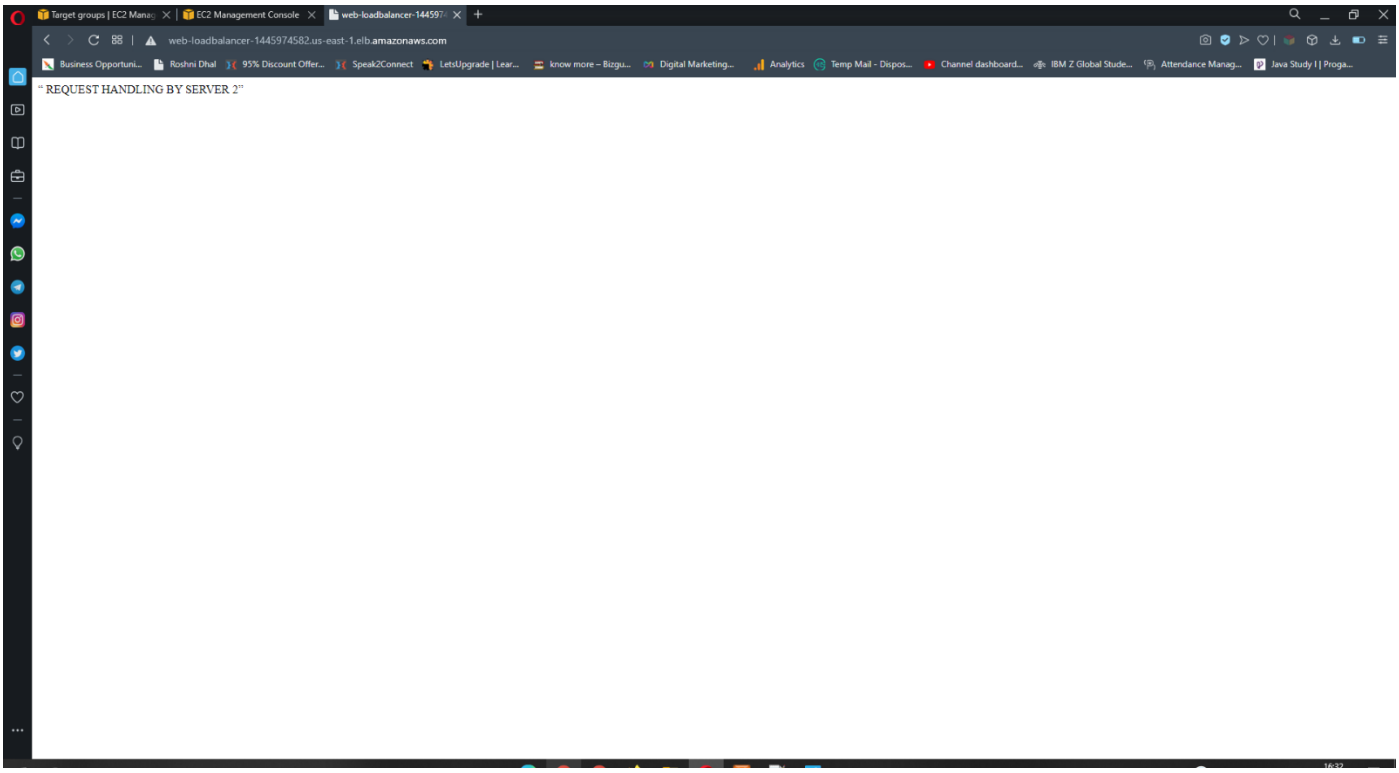
Stopping web server 1:



Health of webserver after being stopped:



When copy pasted DNS of Load balancer after Stopping of webserver 1:



Conclusion:

- Successfully created a new VPC from scratch and created both public and private subnets.
- Created an Internet Gateway and configured a new route table.
- Launched 1 EC2 instance each of the Public and Private subnets and tested Internet access from them.
- Provided Internet access to the EC2 instance in the Private subnet, you created a NAT Gateway and configured a Route table.
- Confirmed that the instance in the private subnet is able to connect to the internet.