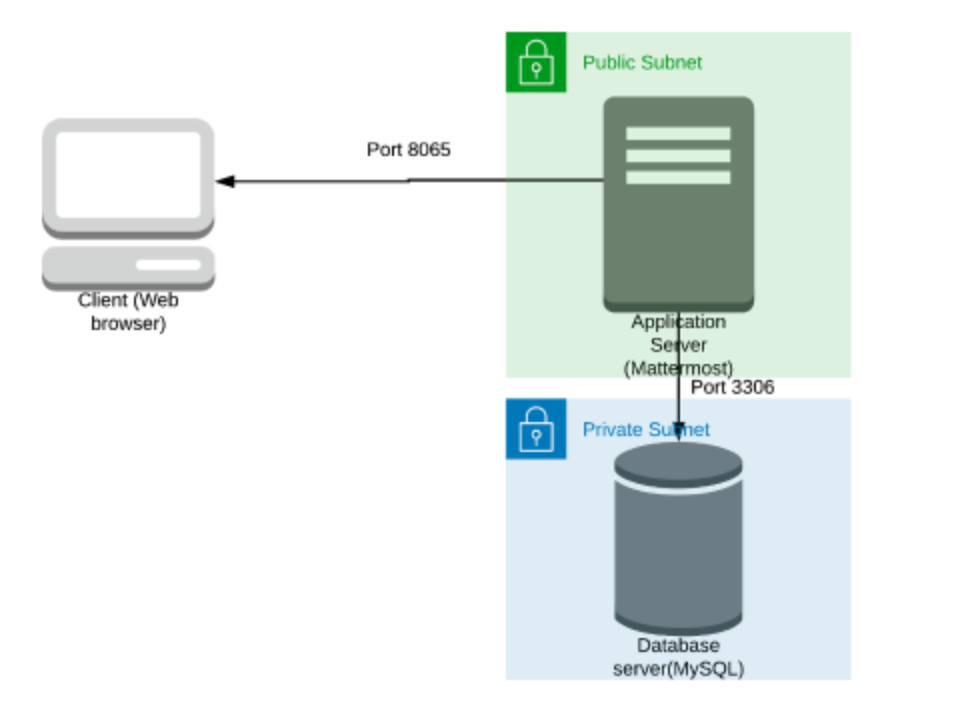
**Scenario**

Team communication and instant messaging solutions are an integral part of any business environment today. As of 2020, the total number of users of Slack and Microsoft Teams exceeded 20 million.

Some organizations might have compliance policies in place which do not allow them to use services managed by third parties. They will prefer solutions that can be managed and hosted on servers controlled by them. The same will extend to communication solutions as well.

**Architecture diagram**

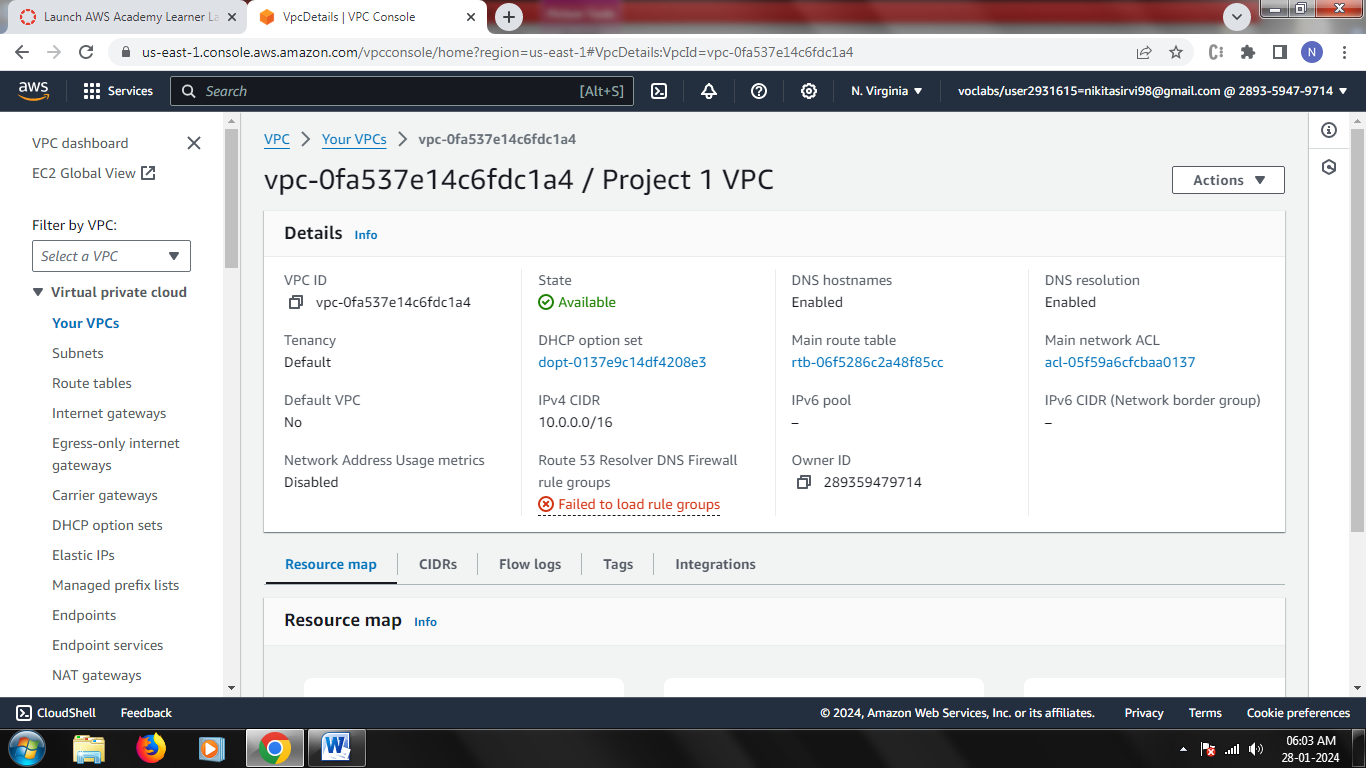
****

|  |  |
| --- | --- |
| **Architecture Implementation** | |
| 1 | Implement 2 different subnets (one public and the other private) in a custom VPC |
| 2 | Install and configure MySQL on an Amazon Linux 2 instance on the private subnet using the instructions provided. (Hint: Use a bastion host and a NAT gateway) |
| 3 | Install and configure Mattermost on an Amazon Linux 2 instance on the public subnet using the provided instructions. |
| 4 | Configure the security groups to allow the ports as shown in the architecture. |
| 5 | Test the installation by accessing the IP of the public instance in a browser via the port 8065. |

**Step 1: VPC and Subnet Creation**

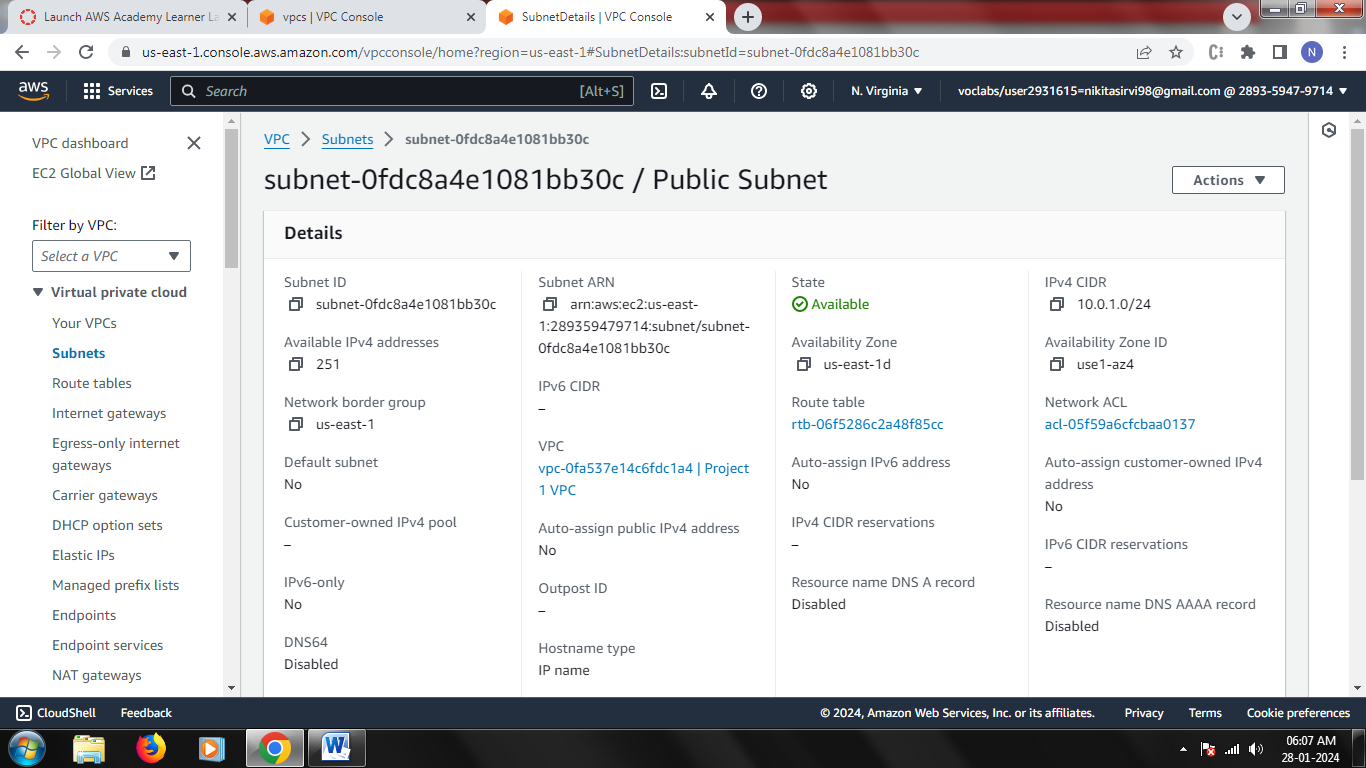
|  |  |  |  |
| --- | --- | --- | --- |
| Step number | A |  |  |
| Step name | Creation of VPC |  |  |
| Instructions | 1) Navigate to VPC using the Services button at the top of the screen  2) Select "Your VPCs" on the left side of the screen  3) Click on "Create VPC"  4) Enter the following fields :  Name: Project 1 VPC  IPv4 CIDR Block : 10.0.0.0/16  The rest of the options can be ignored  5) Select "Create VPC"  6) Select the VPC and click on Actions->Edit DNS hostnames  7) Enable DNS hostnames and click on Save |  |  |
| Expected screenshots | 1. Created VPC with properties visible |  |  |

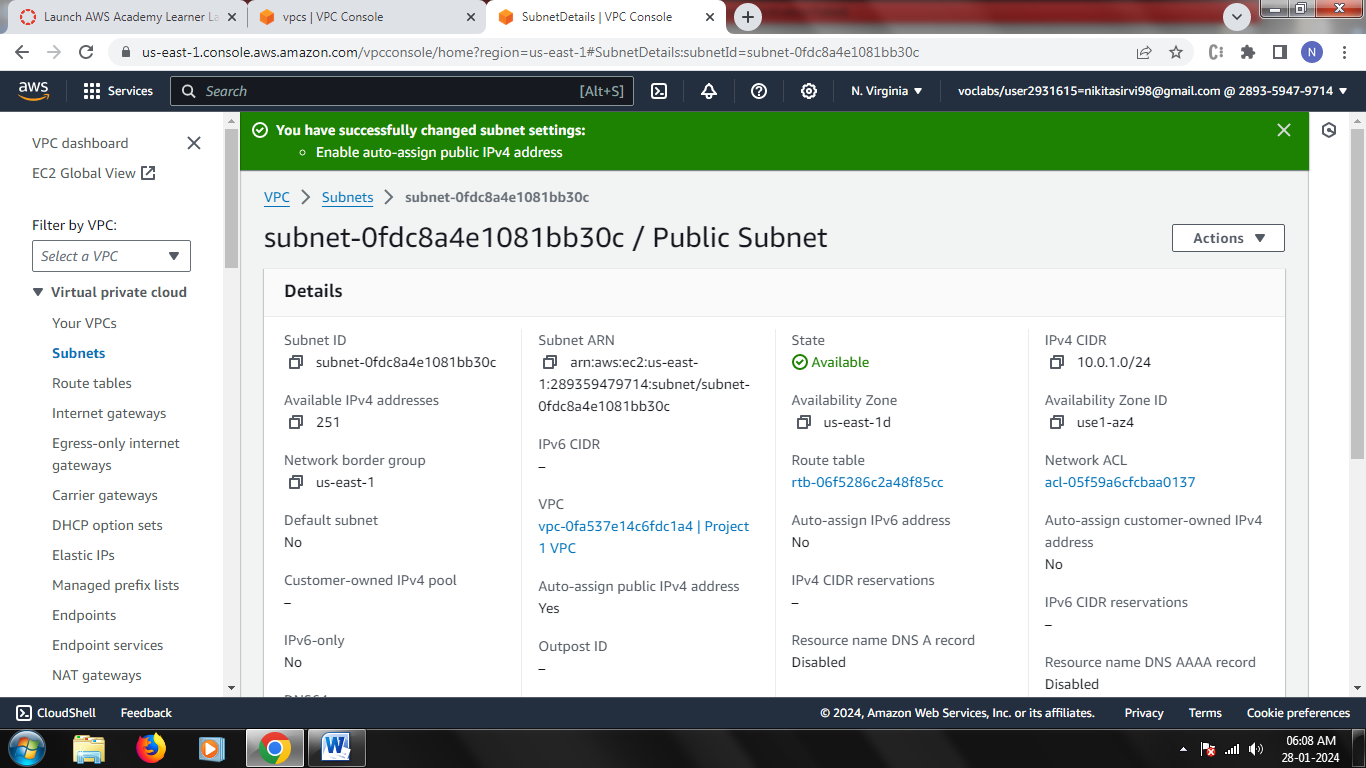
**<Insert Screenshot a(1) here>**



|  |  |  |
| --- | --- | --- |
| Step number | b |  |
| Step name | Creation of public subnet |  |
| Instructions | 1) Navigate to VPC->Subnets  2) Click on "Create Subnet"  3) Enter the following fields  Name tag : Public Subnet  VPC : Select the Project 1 VPC  IPv4 CIDR block : 10.0.1.0/24  The other options can be ignored  4) Click on Create  5) Once the subnet has been created, select the subnet and click on Actions->Modify Auto-assign IP settings  6) Enable the option "Auto assign IPv4" and select Save |  |
| Expected screenshots | 1. Subnet Creation screen |  |

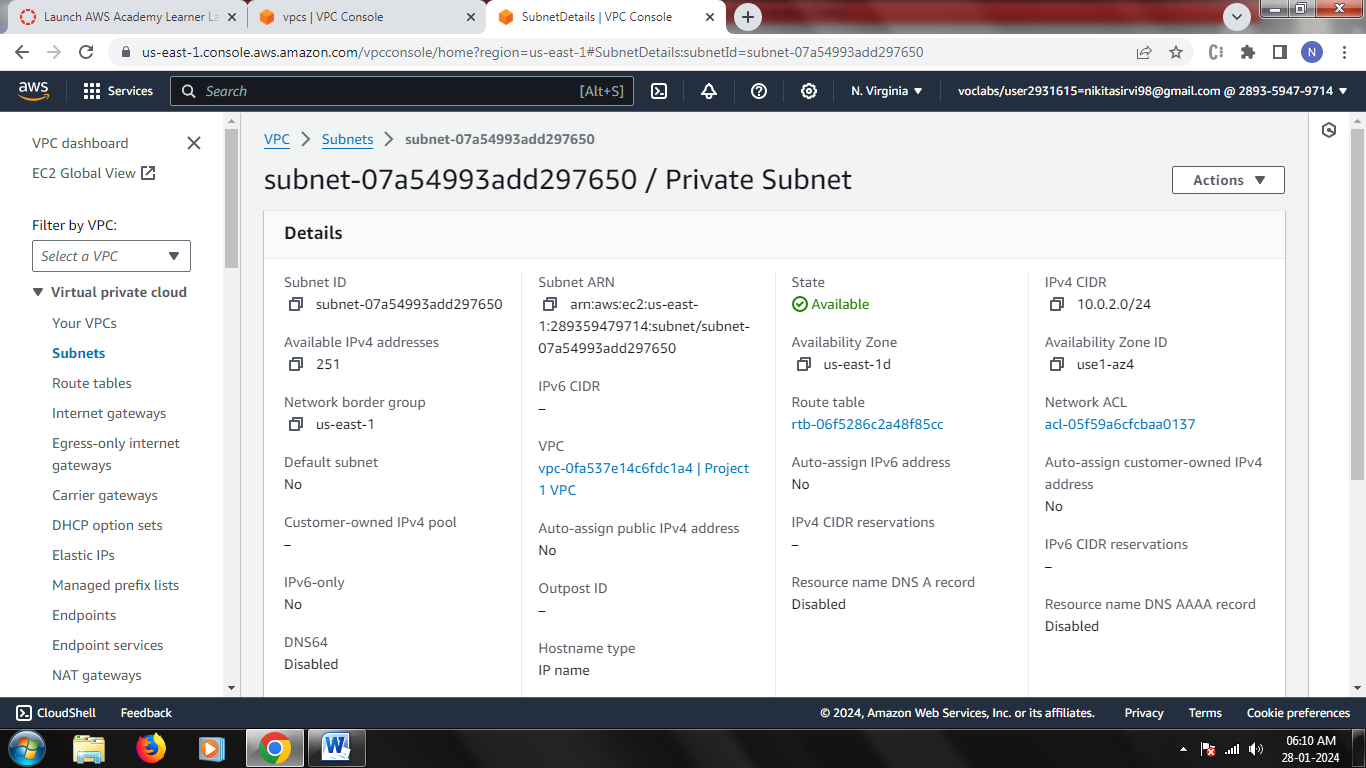
**<Insert Screenshot b(1) here>**





|  |  |
| --- | --- |
| Step number | C |
| Step name | Creation of private subnet |
| Instructions | 1) Navigate to VPC->Subnets  2) Click on "Create Subnet"  3) Enter the following fields  Name tag : Private Subnet  VPC : Select the Project 1 VPC  IPv4 CIDR block : 10.0.2.0/24  The other options can be ignored  4) Click on Create |
| Expected screenshots | 1. Subnet Creation screen |

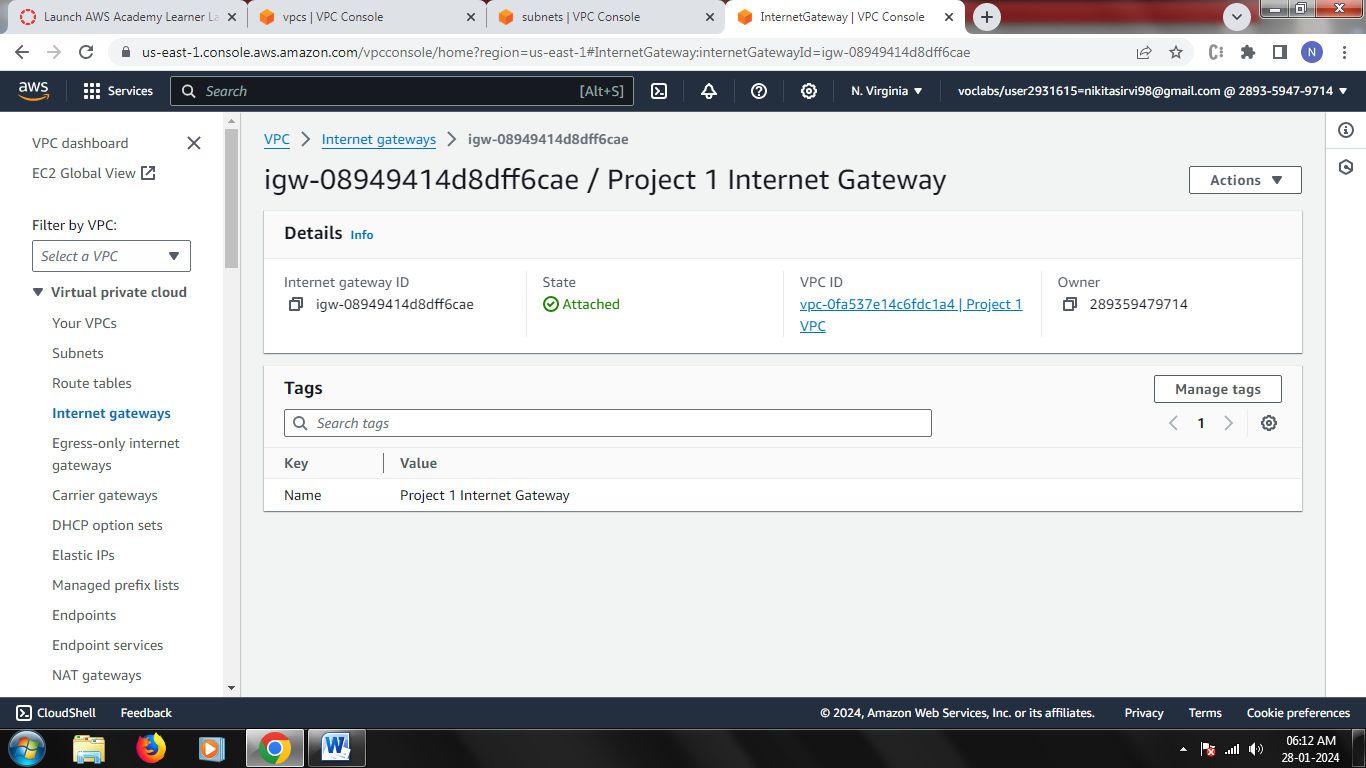
**<Insert Screenshot c(1) here>**



**Step 2 : Internet Gateway and VPC**

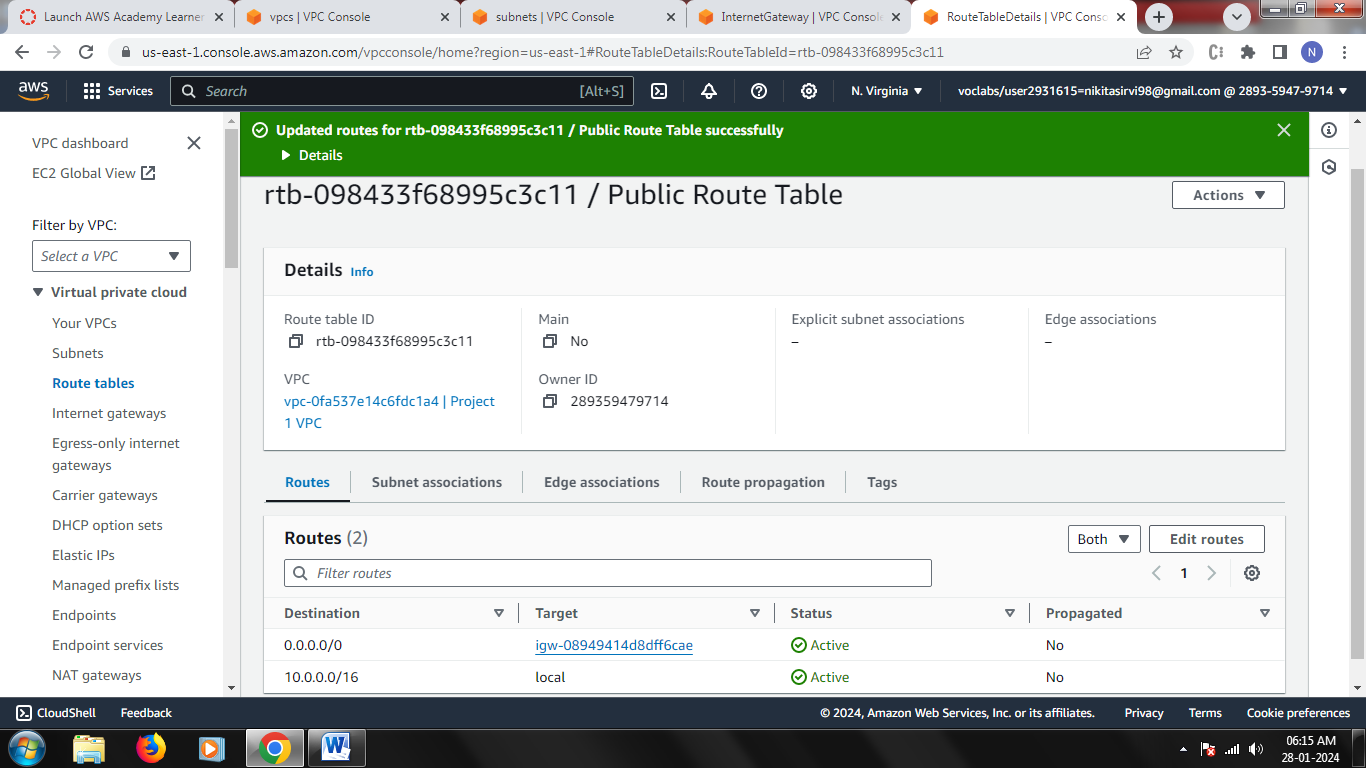
|  |  |  |
| --- | --- | --- |
| Step number | a |  |
| Step name | Creation and Configuration of Internet Gateway |  |
| Instructions | 1) Navigate to VPCs->Internet Gateway  2) Click on "Create Internet Gateway"  3) Enter the name tag "Project 1 Internet Gateway" and click on "Create Internet Gateway"  4) After the gateway is created, select it and click on Actions->Attach to VPC  5) Select the Project 1 VPC and click on "Attach Internet Gateway" |  |
| Expected screenshots | 1. Creation of Internet Gateway |  |

**<Insert Screenshot a(1) here >**

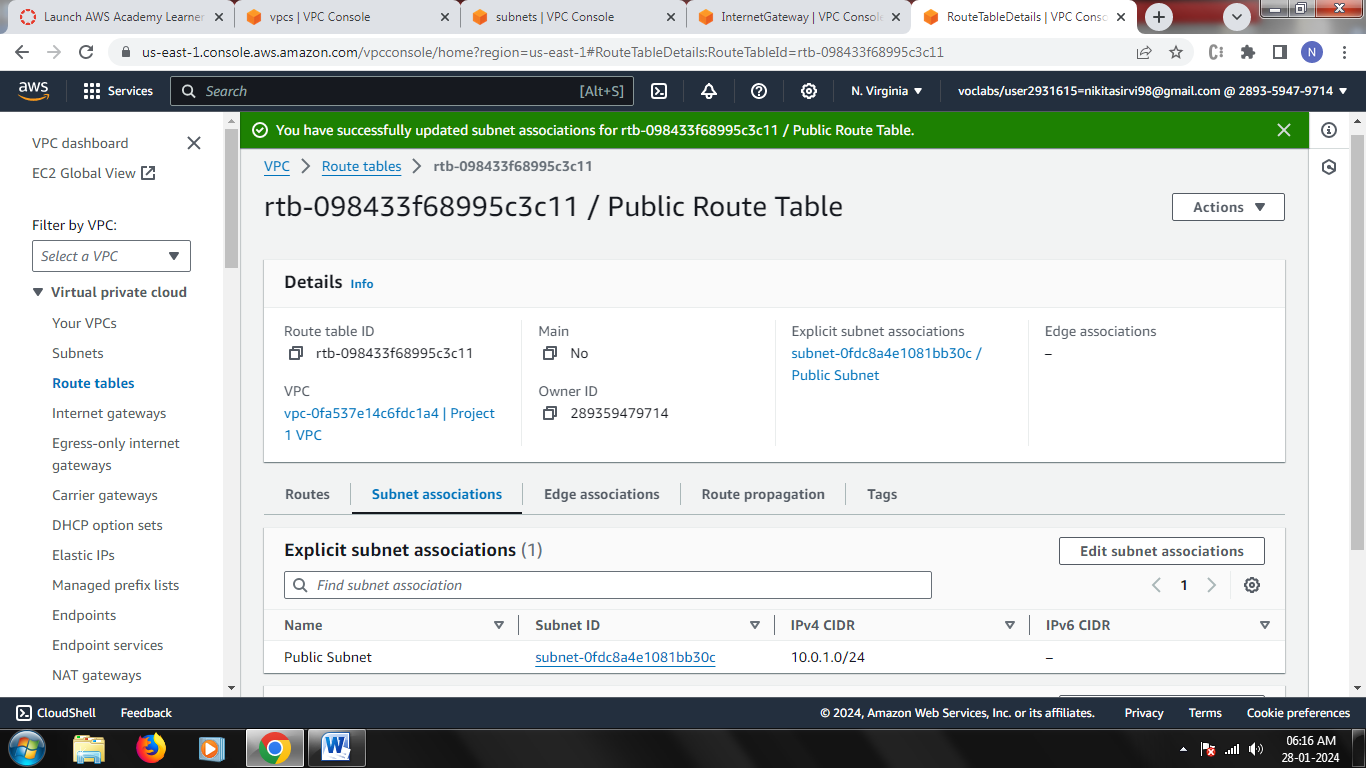


|  |  |  |
| --- | --- | --- |
| Step number | b |  |
| Step name | Creation of public route table |  |
| Instructions | 1) Navigate to VPC -> Route Tables and click on Create Route table  2) Enter the name tag "Public Route Table", select the Project 1 VPC from the dropdown and click on Create  3) Once the route table is created, select it and select the Routes tab below the list of route tables  4) Click in Edit Routes and add the following route (Don't edit the existing one)  - Destination : 0.0.0.0/0  - Target : Select Internet Gateway and the select the Project 1 Internet Gateway  Click on Save Routes  5) Select the Subnet Associations tab and click on Edit Subnet Associations  6) Select the Public Subnet from the list and click on Save |  |
| Expected screenshots | 1. Route list of the route table 2. Subnet Associations of the route table |  |

**<Insert Screenshot b(1) here>**

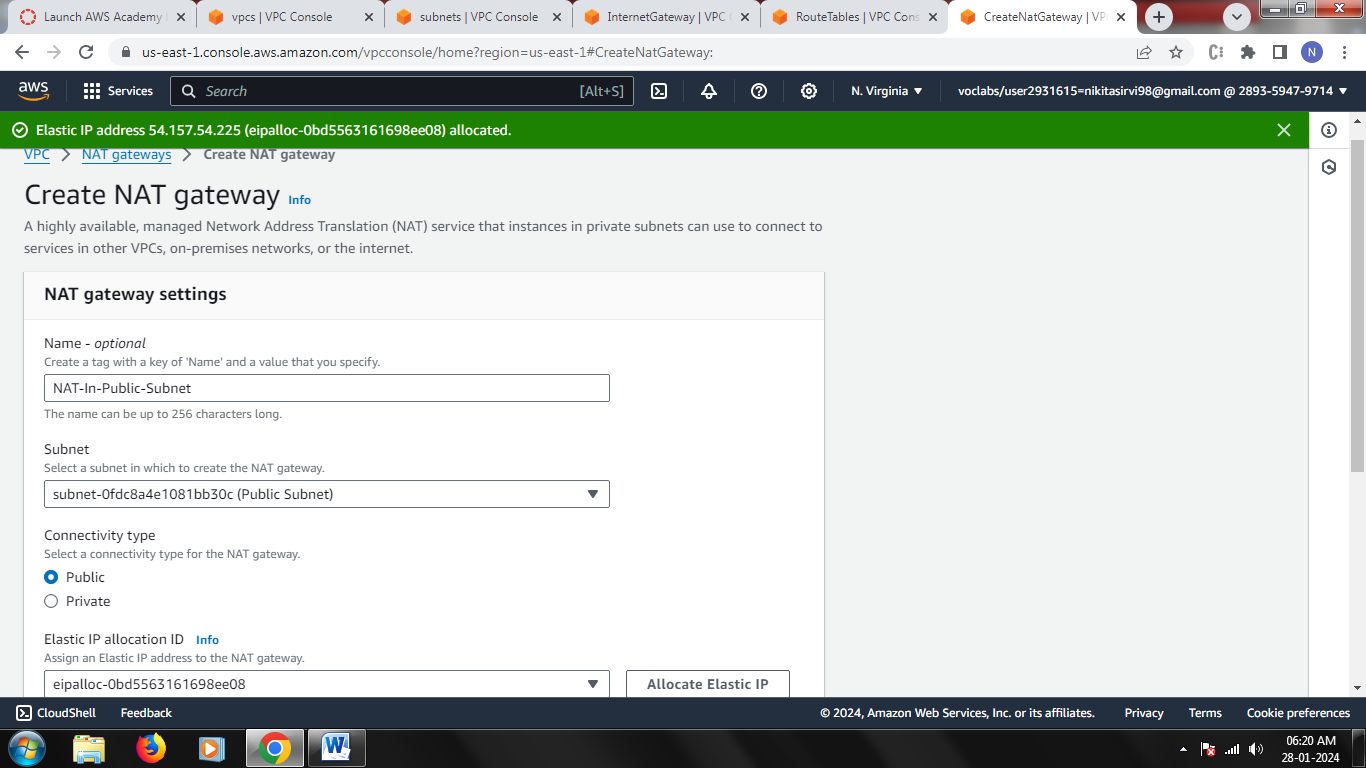


**<Insert Screenshot b(2) here>**

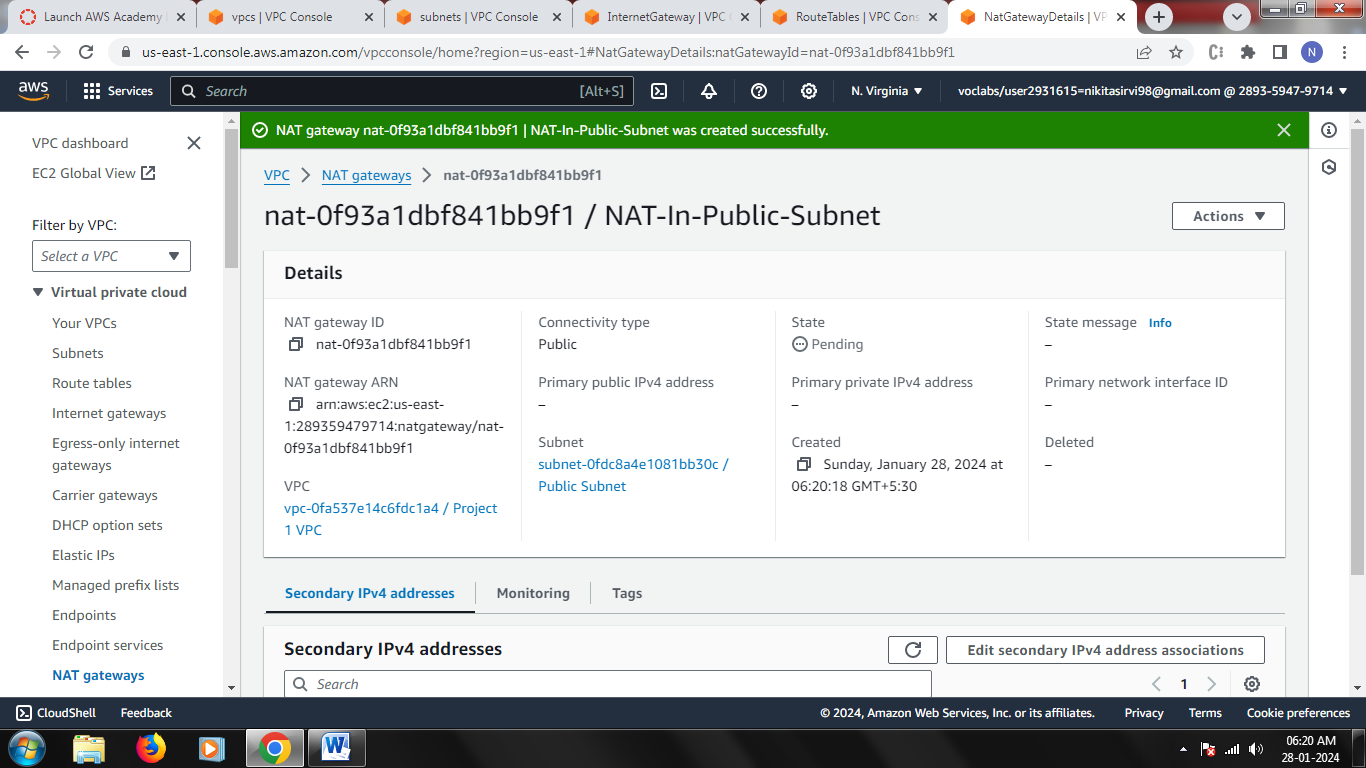


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step number | c |  |  |  |
| Step name | Creation of NAT gateway |  |  |  |
| Instructions | 1) Navigate to VPC using the Services button at the top of the screen  2) Select NAT Gateway at the left side of the screen  3) Click on Create NAT Gateway  - Deploy it in the public subnet  - Connectivity type : Public  - Allocate an elastic IP by clicking on “Allocate Elastic IP”  4) Click on “Create NAT Gateway” to create the gateway |  |  |  |
| Expected screenshots | 1. NAT gateway creation details 2. Gateway after creation |  |  |  |
|  |  |  |  |  |

**<Insert Screenshot c(1) here>**

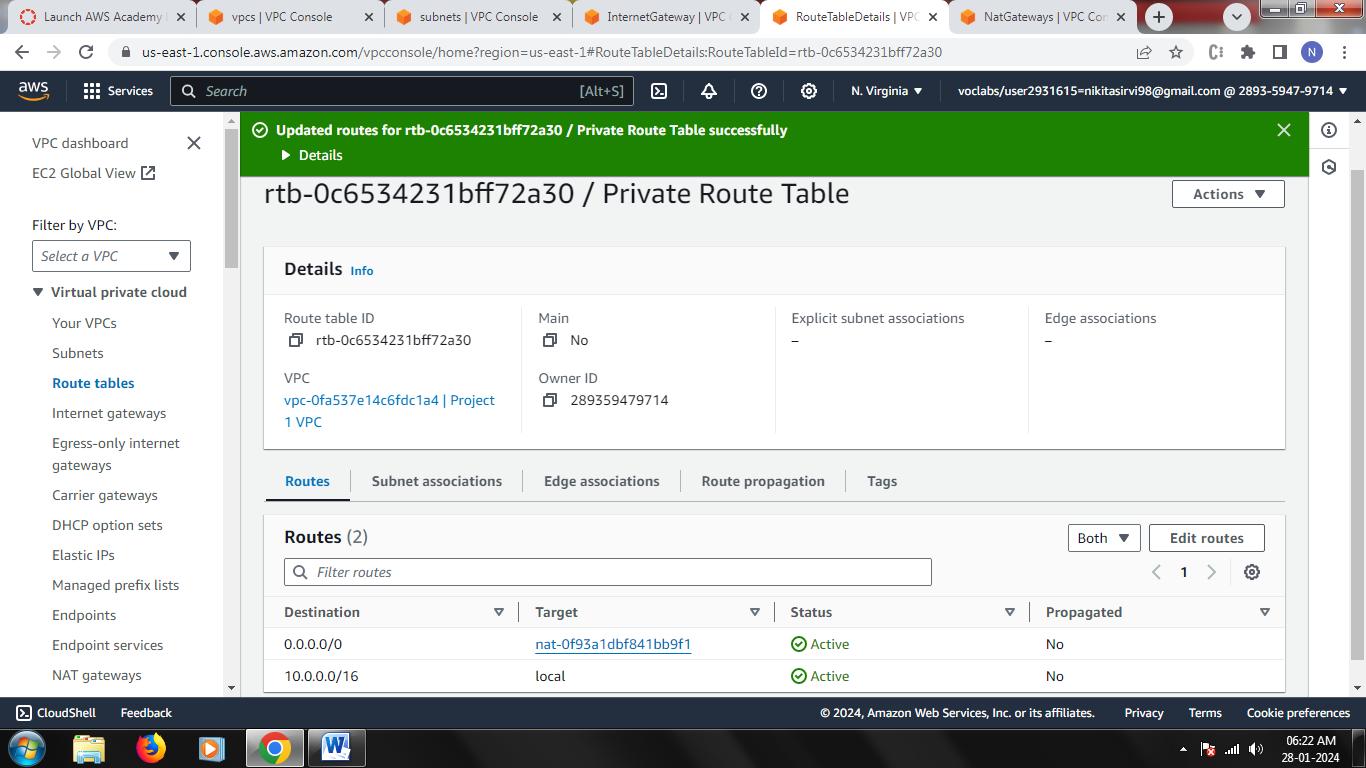


**<Insert Screenshot c(2) here>**

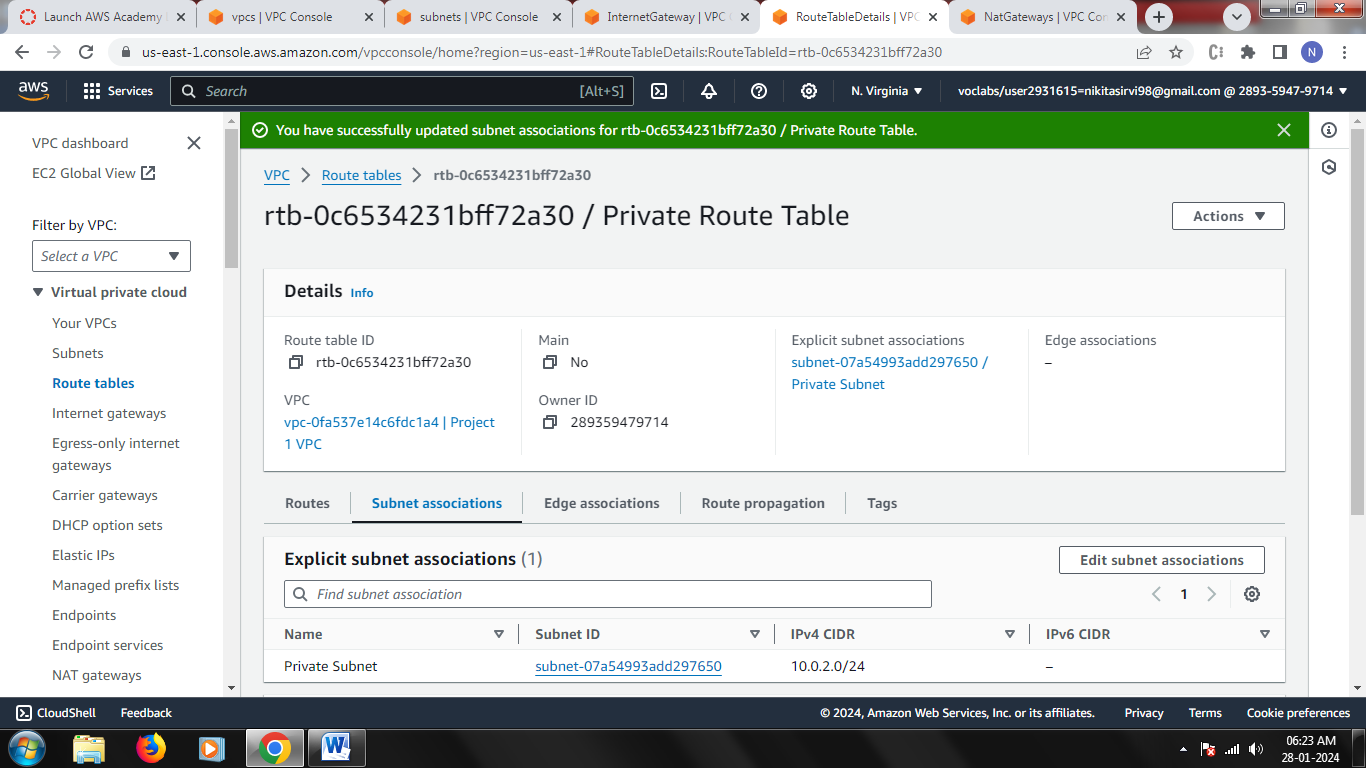


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step number | d |  |  |  |
| Step name | Creation of private route tables |  |  |  |
| Instructions | 1) Navigate to VPC -> Route Tables and click on Create Route table  2) Enter the name tag "Private Route Table", select the Project 1 VPC from the dropdown and click on Create  3) Once the route table is created, select it and select the Routes tab below the list of route tables  4) Click in Edit Routes and add the following route (Don't edit the existing one)  - Destination : 0.0.0.0/0  - Target: Select NAT Gateway and select the NAT Gateway created in the previous step  Click on Save Routes  5) Select the Subnet Associations tab and click on Edit Subnet Associations  6) Select the private Subnet from the list and click on Save |  |  |  |
| Expected screenshots | 1. Route list of the route table 2. Subnet association of the route table |  |  |  |

**<Insert Screenshot for d(1) here >**



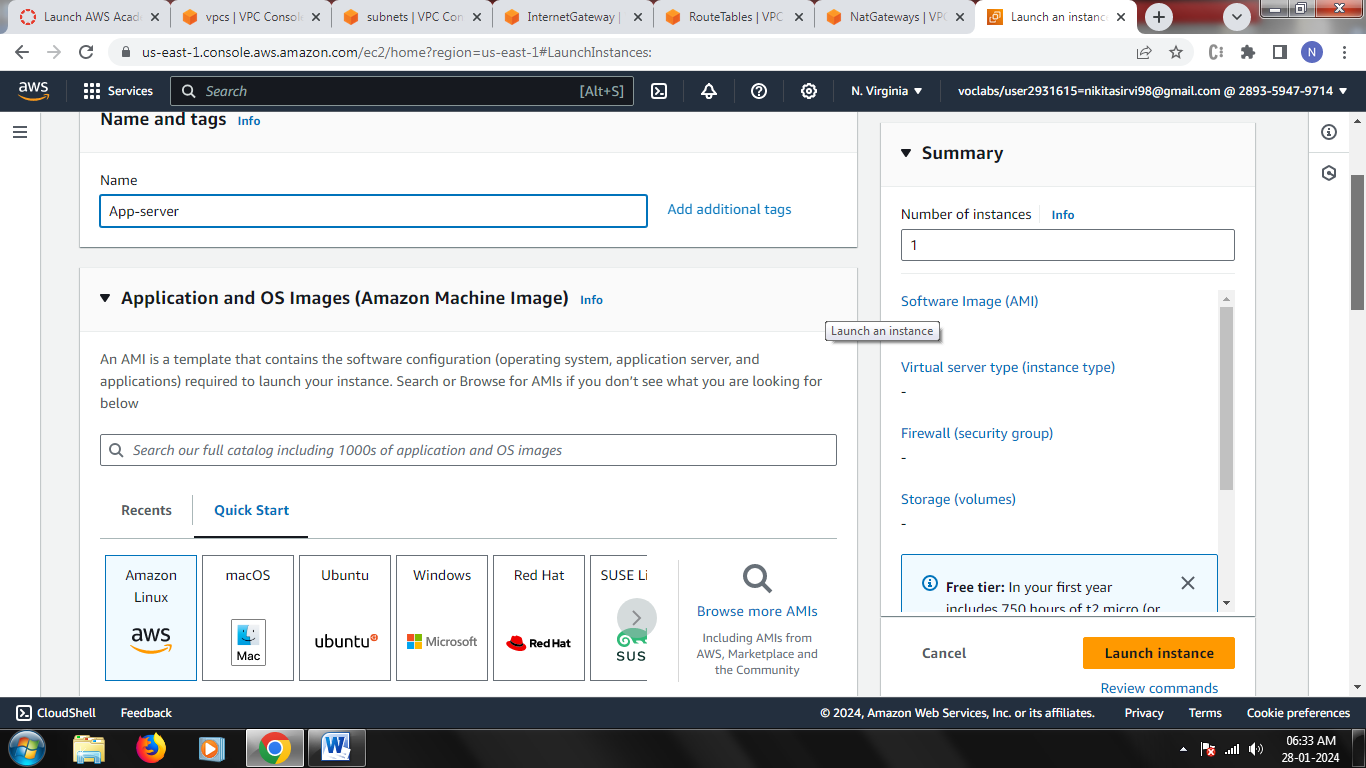
**<Insert Screenshot for d(2) here>**



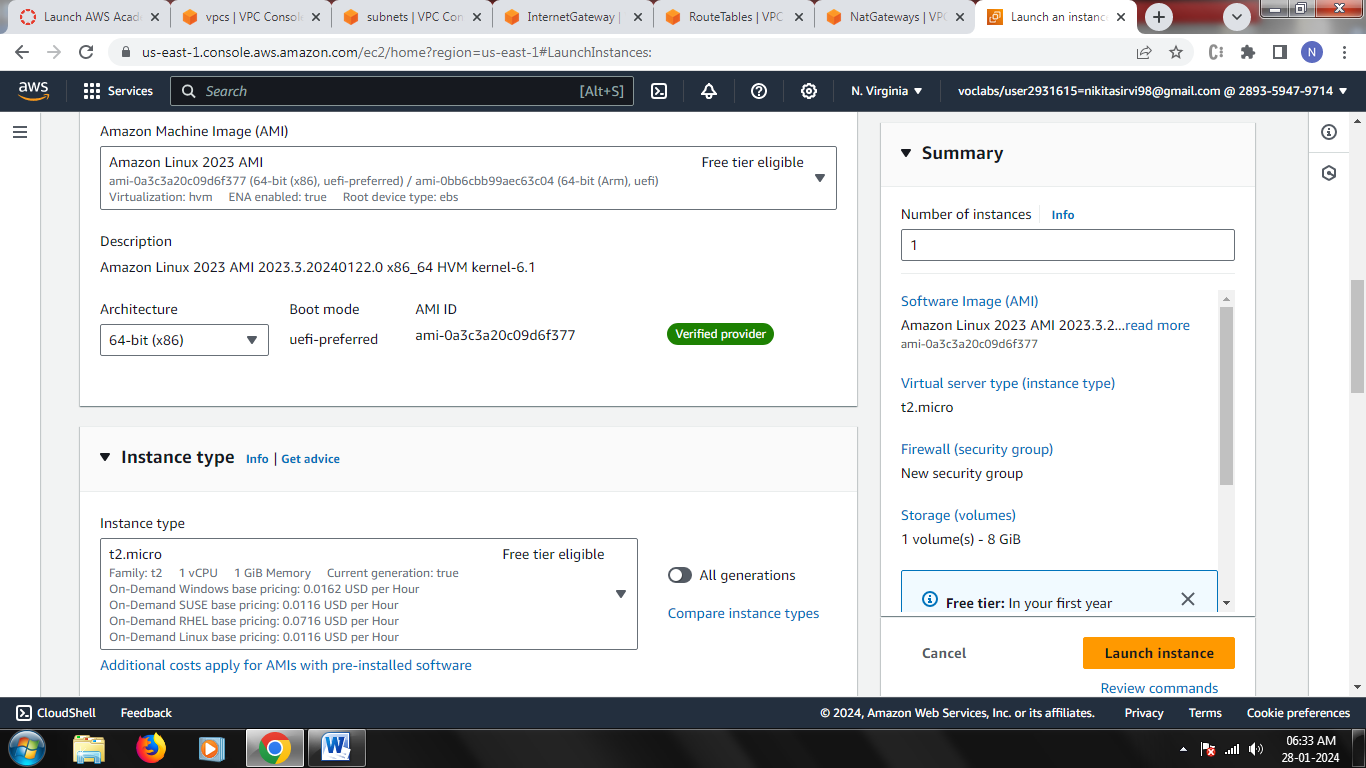
**Step 3 : Creation of database and application servers**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step number | a |  |  |  |
| Step name | Creation of application server |  |  |  |
| Instructions | 1) Navigate to EC2 using the Services button at the top of the screen  2) Select Instances at the left side of the screen  3) Click on Launch Instance  - Select the AMI Amazon 2 Linux  - Select the instance type t2.micro  - Select Network as "Project 1 VPC" and subnet as "Public Subnet"  - For the security group, open the ports 80,443, 22 and 8065 for source set to "Anywhere"  4) Launch the instance after creating a new pem file and downloading it | | | |
| Expected screenshots | 1. AMI used 2. Instance configuration screen 3. Security group rules 4. Instance after creation |  |  |  |

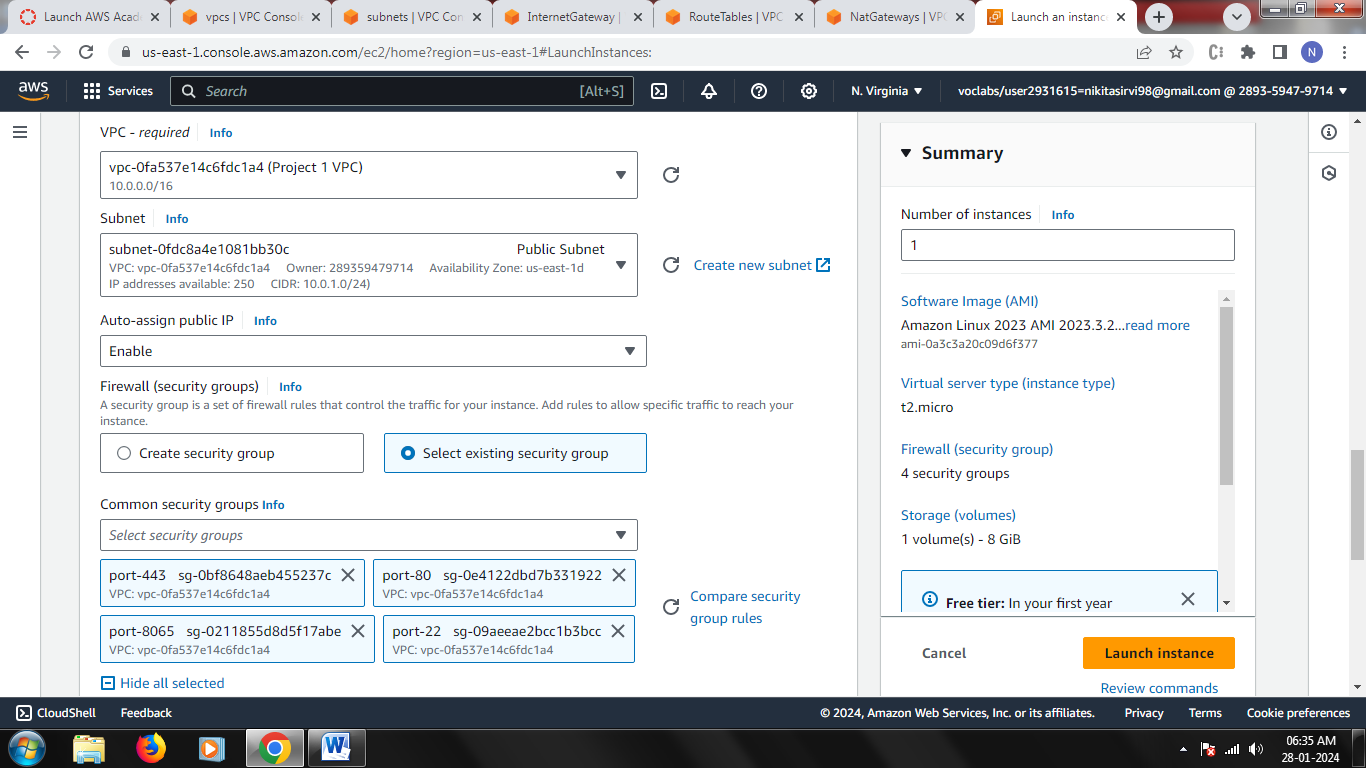
**<Insert screenshot a(1) here>**



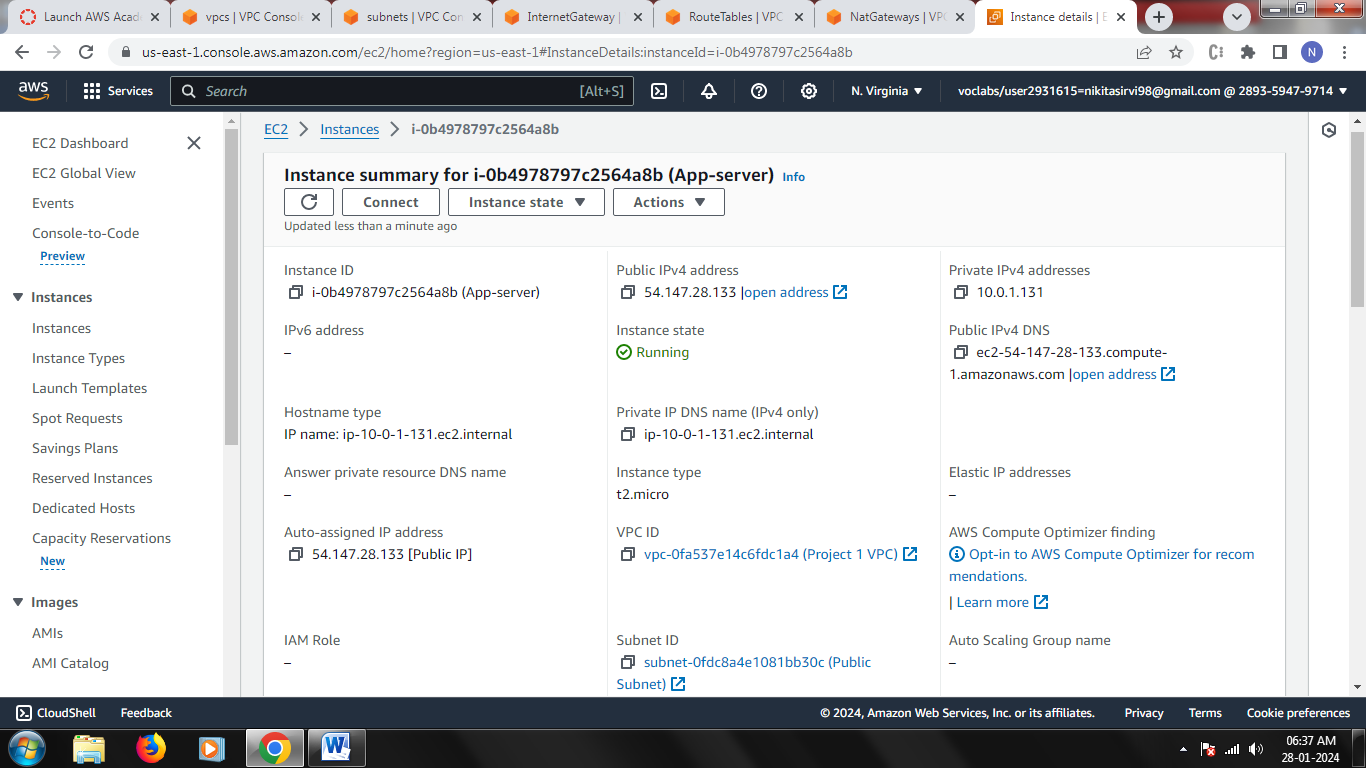
**<Insert screenshot a(2) here>**



**<Insert screenshot a(3) here>**

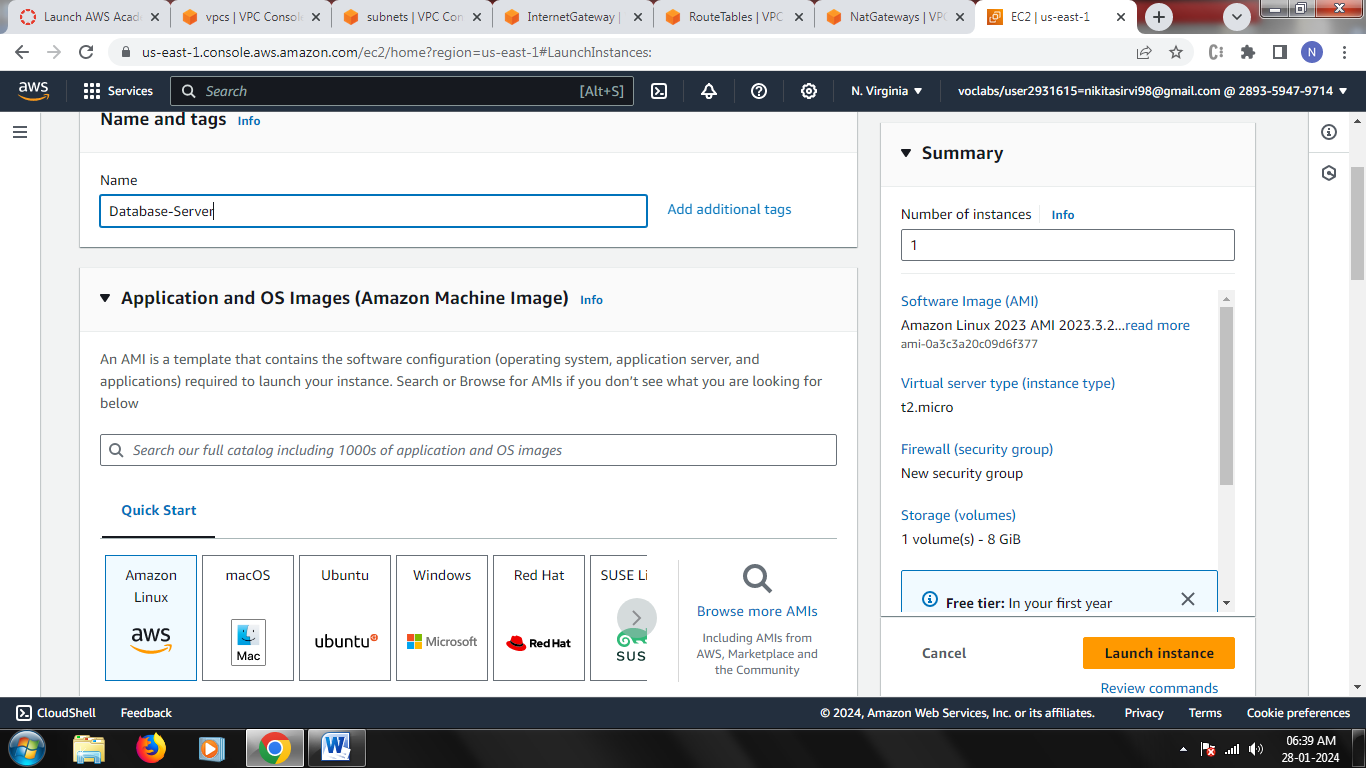


**<Insert screenshot a(4) here>**

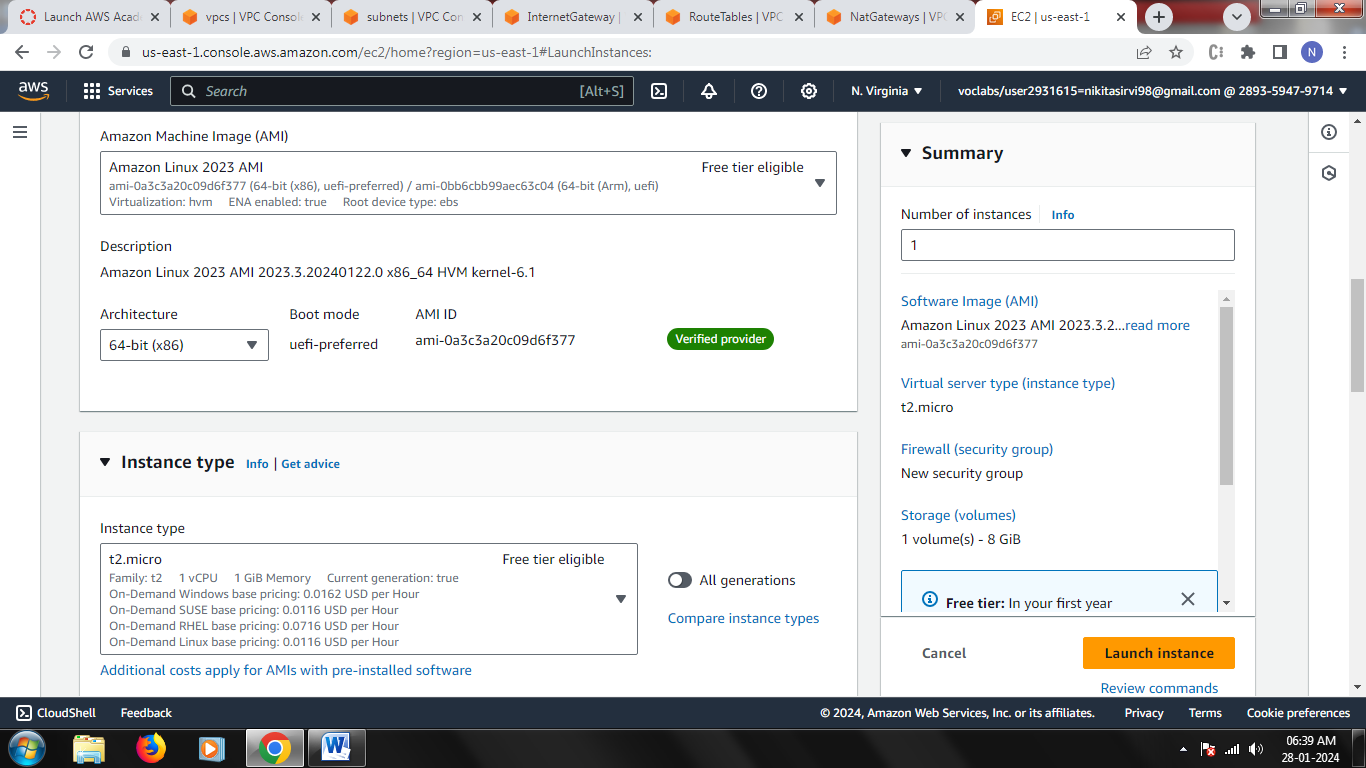


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step number | b |  |  |  |
| Step name | Creation of database server |  |  |  |
| Instructions | 1) Navigate to EC2 using the Services button at the top of the screen  2) Select Instances at the left side of the screen  3) Click on Launch Instance  - Select the AMI Amazon 2 Linux  - Select the instance type t2.micro  - Select Network as "Project 1 VPC" and subnet as "Private Subnet"  - For the security group, open the ports 80, 443,22 and 3306 for source set to "Anywhere"  4) Launch the instance by selecting the same pem file created in the previous step | | | |
| Expected screenshots | 1. AMI used 2. Instance configuration screen 3. Security group rules 4. Instance after creation |  |  |  |

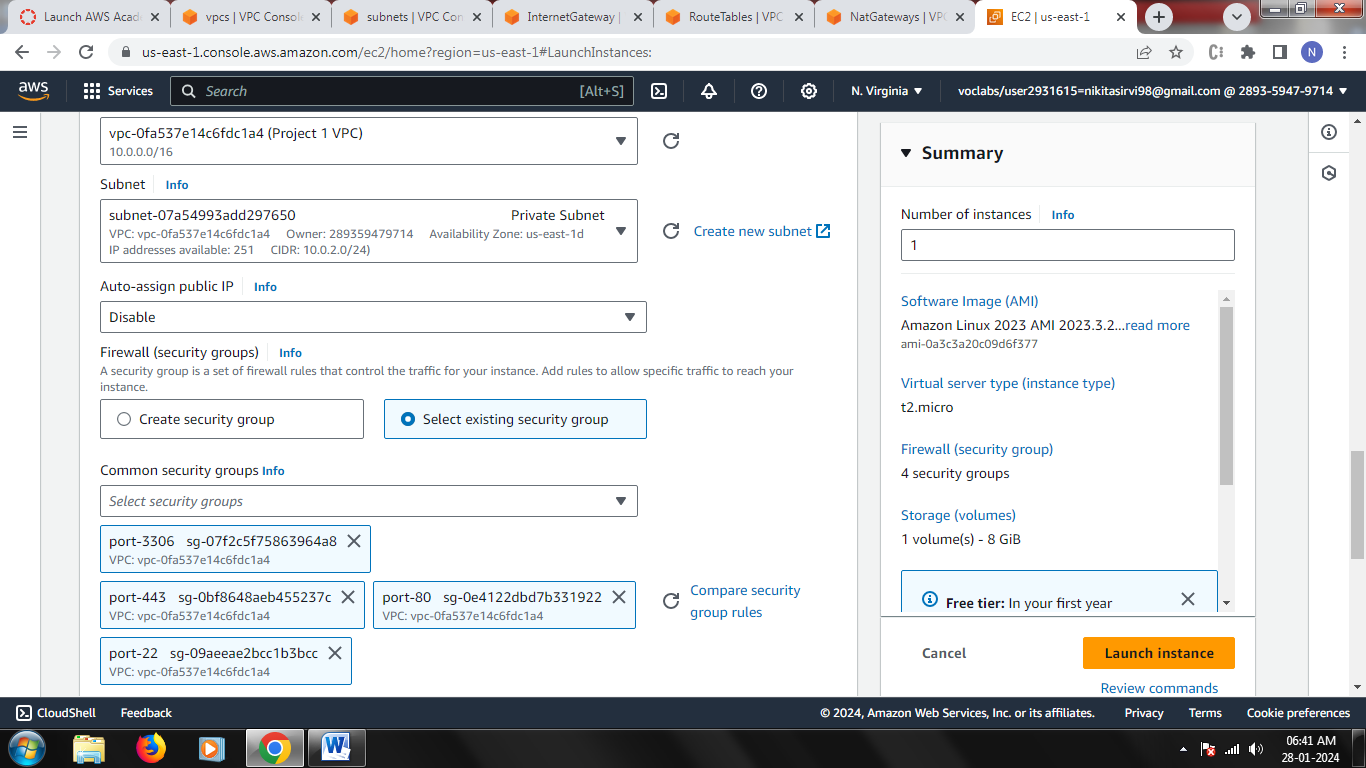
**<Insert screenshot b(1) here>**



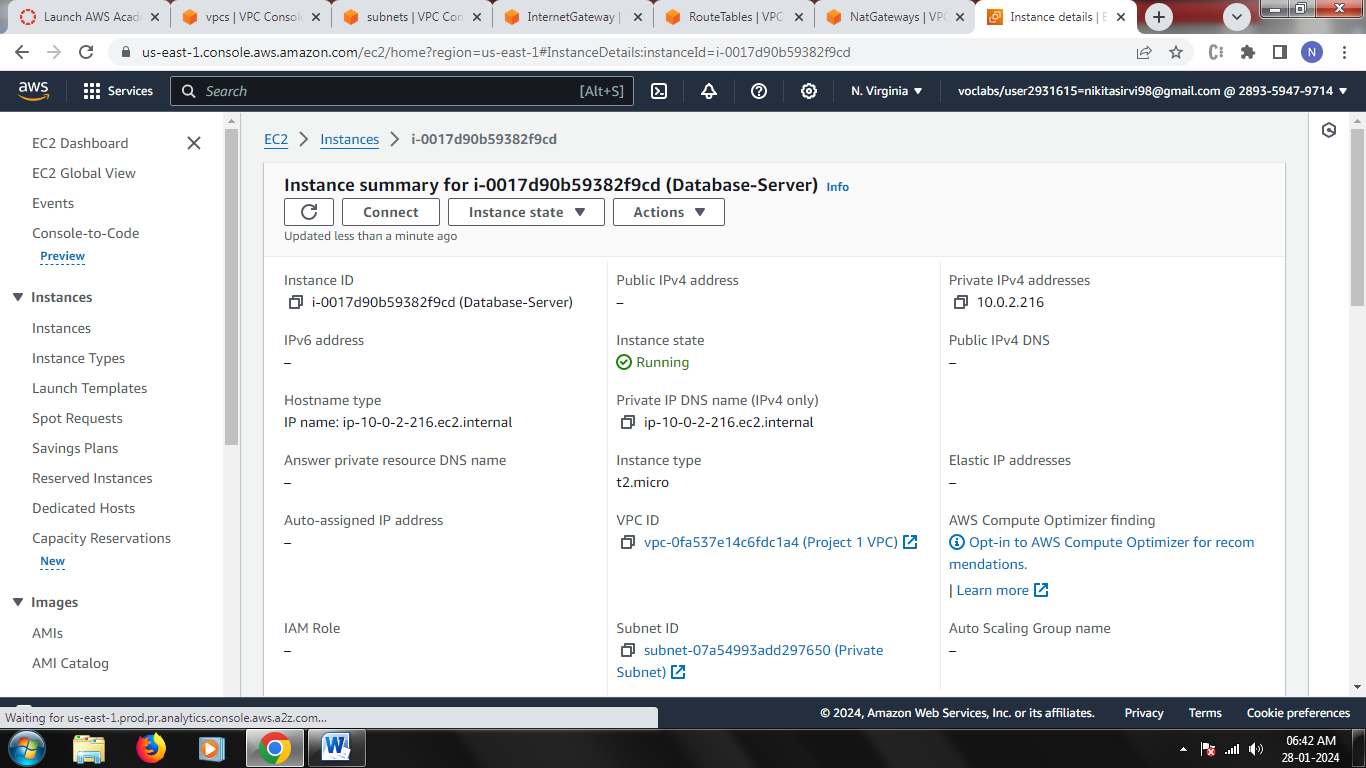
**<Insert screenshot b(2) here>**



**<Insert screenshot b(3) here>**



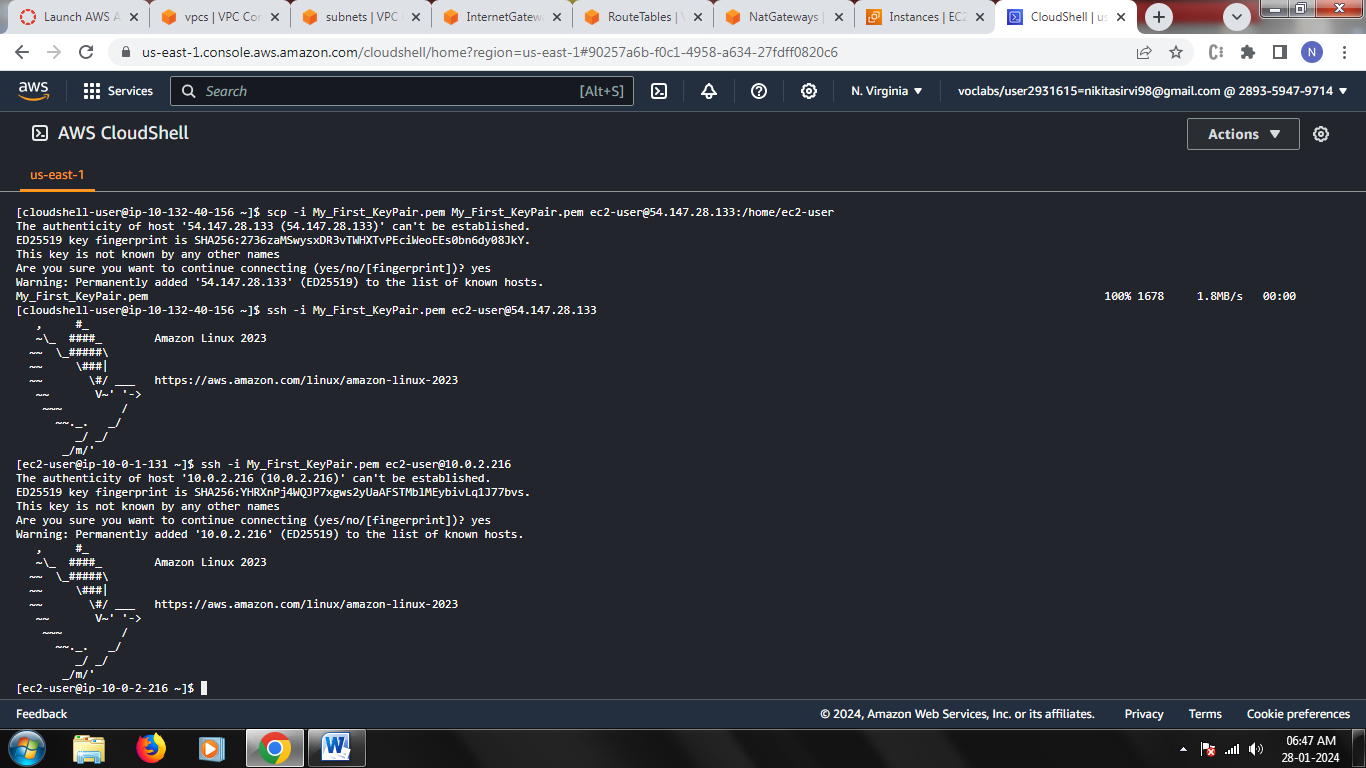
**<Insert screenshot b(4) here>**

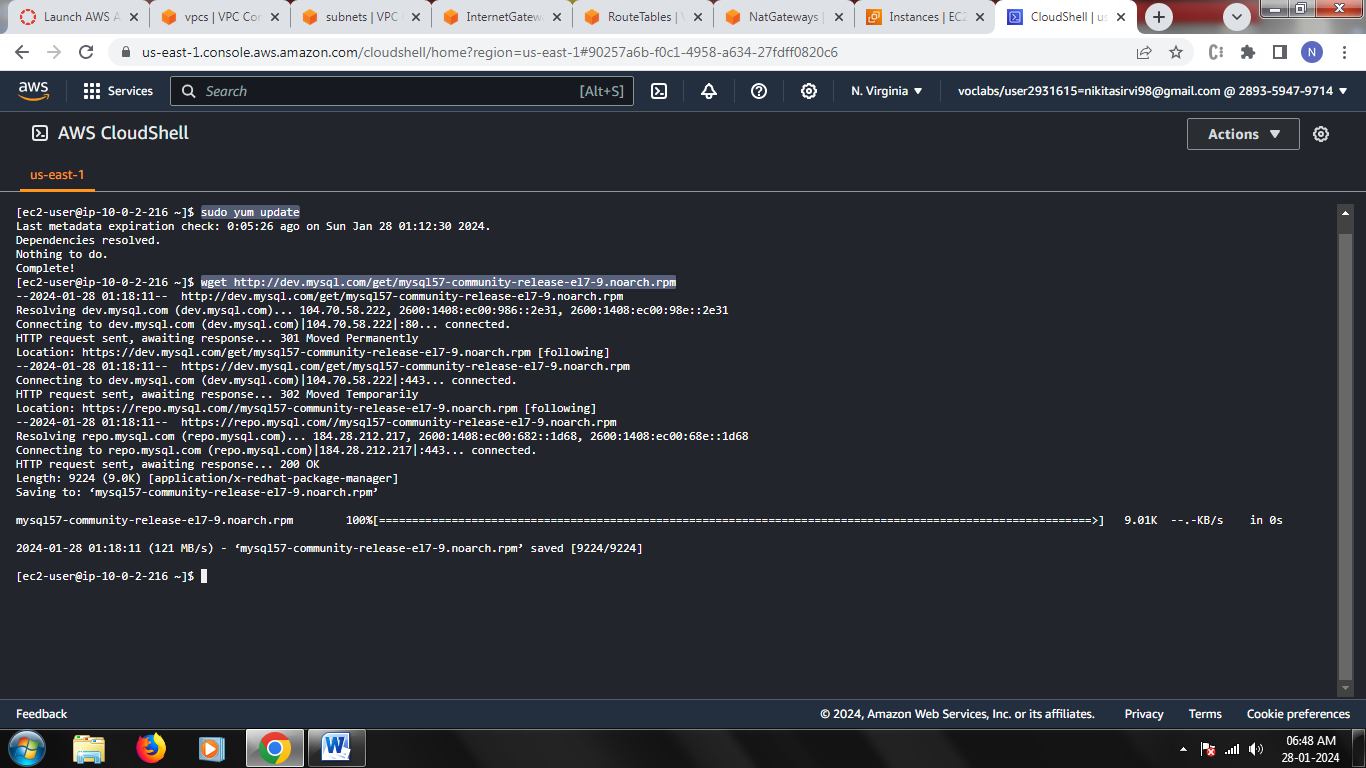


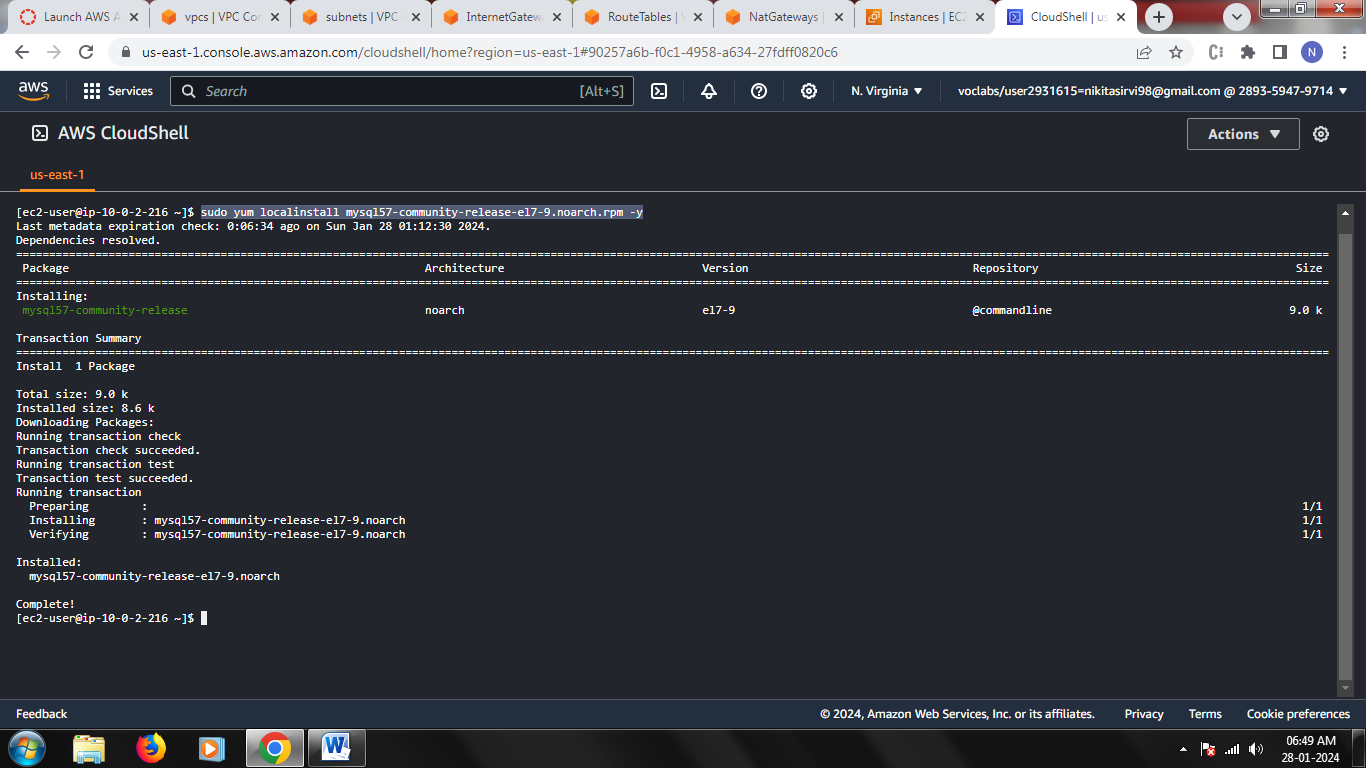
**Step 4: Application and Database Installation and Testing**

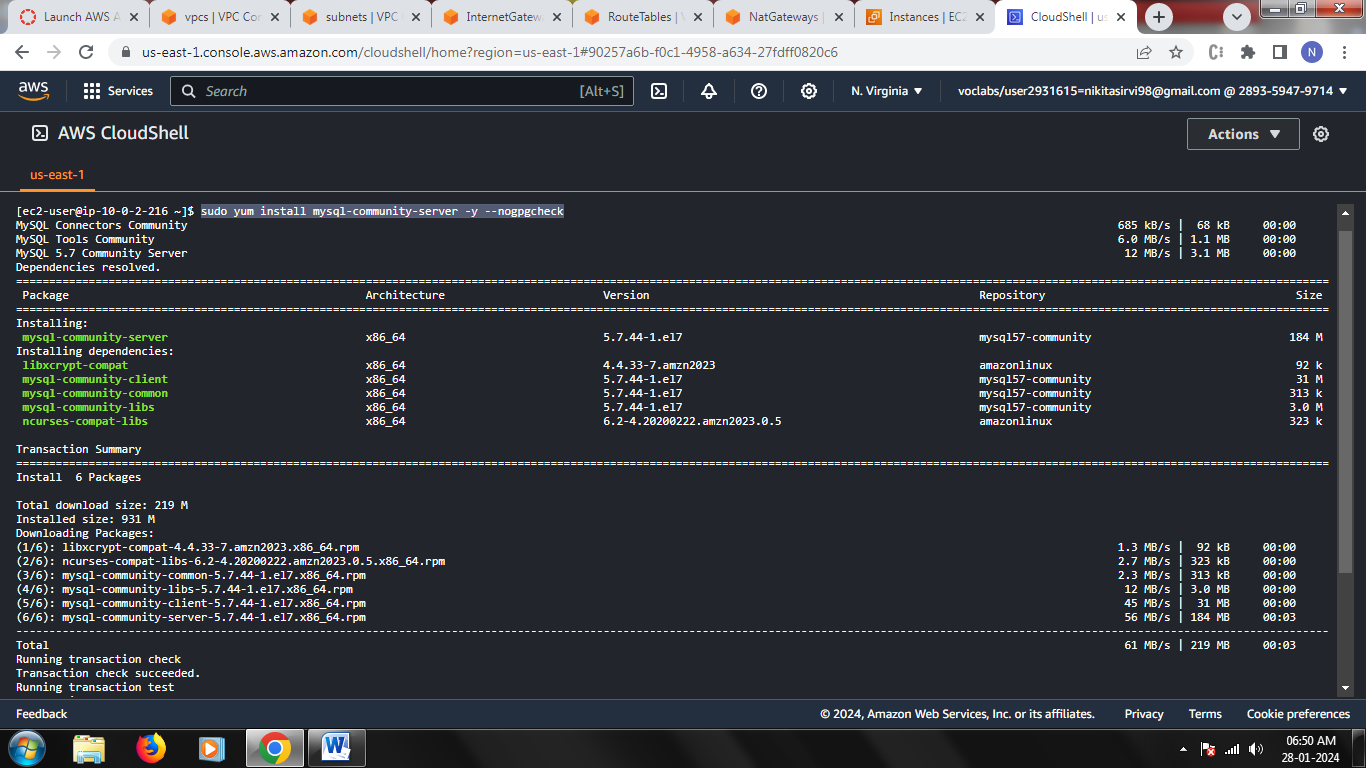
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step number | a |  |  |  |
| Step name | Installation and configuration of MySQL |  |  |  |
| Instructions | 1) Copy the database pem file into the application server using the below command  *scp -i <application server pem file> <database server pem file > ec2-user@<application server public IP>:/home/ec2-user*  2) Log into the application server using SSH/Putty  3) From the application server, log into the database server using the pem file copied in step 1and the private IP address of the database server with the following command  *ssh -i <database server pem file> ec2-user@<private IP of database server>*  4) Enter the following commands to install and configure MySQL on the database server  *sudo yum update* *wget http://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm sudo yum localinstall mysql57-community-release-el7-9.noarch.rpm -y*  *sudo yum install mysql-community-server -y --nogpgcheck*  *sudo systemctl start mysqld.service*  Run the below command to retrieve a temporary password for MySQL *sudo grep 'temporary password' /var/log/mysqld.log | rev | cut -d" " -f1 | rev | tr -d "."*  Log in to MySQL with the below command and enter the above password when prompted *mysql -u root -p*  Enter the below command after you login to MySQL  *ALTER USER 'root'@'localhost' IDENTIFIED BY 'Password42!';*  Type ‘exit’ into the MySQL prompt and press Enter to exit out of the MySQL environment. Enter the below commands to complete the setup. Ignore any warning messages you receive.  *wget https://d6opu47qoi4ee.cloudfront.net/install\_mysql\_linux.sh*  *chmod 777 install\_mysql\_linux.sh*  *sudo ./install\_mysql\_linux.sh*  5) Type *exit* to exit the database server and go back to the application server | | | |
| Expected screenshots | 1. Installation of MySQL 2. Retrieving the temporary password 3. Executing the provided script |  |  |  |

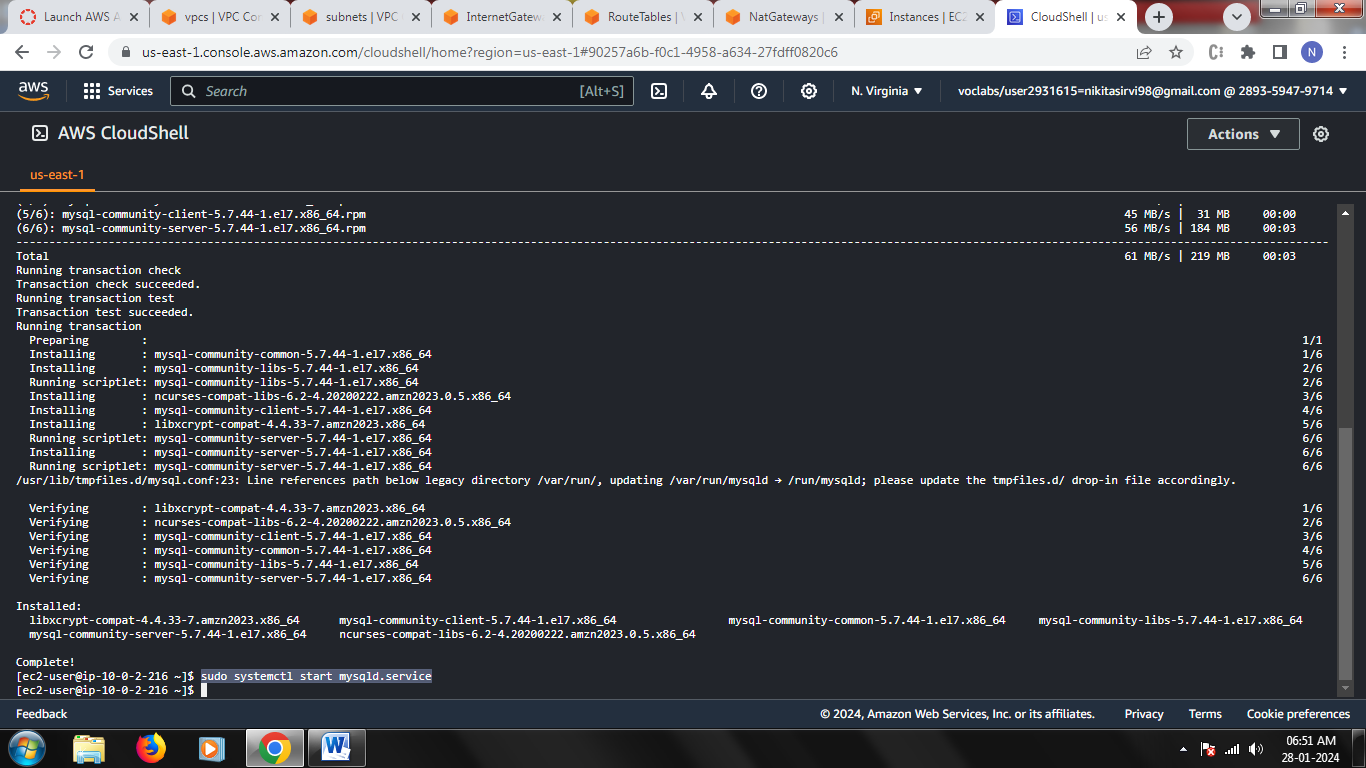
**<Insert screenshot a(1) here>**



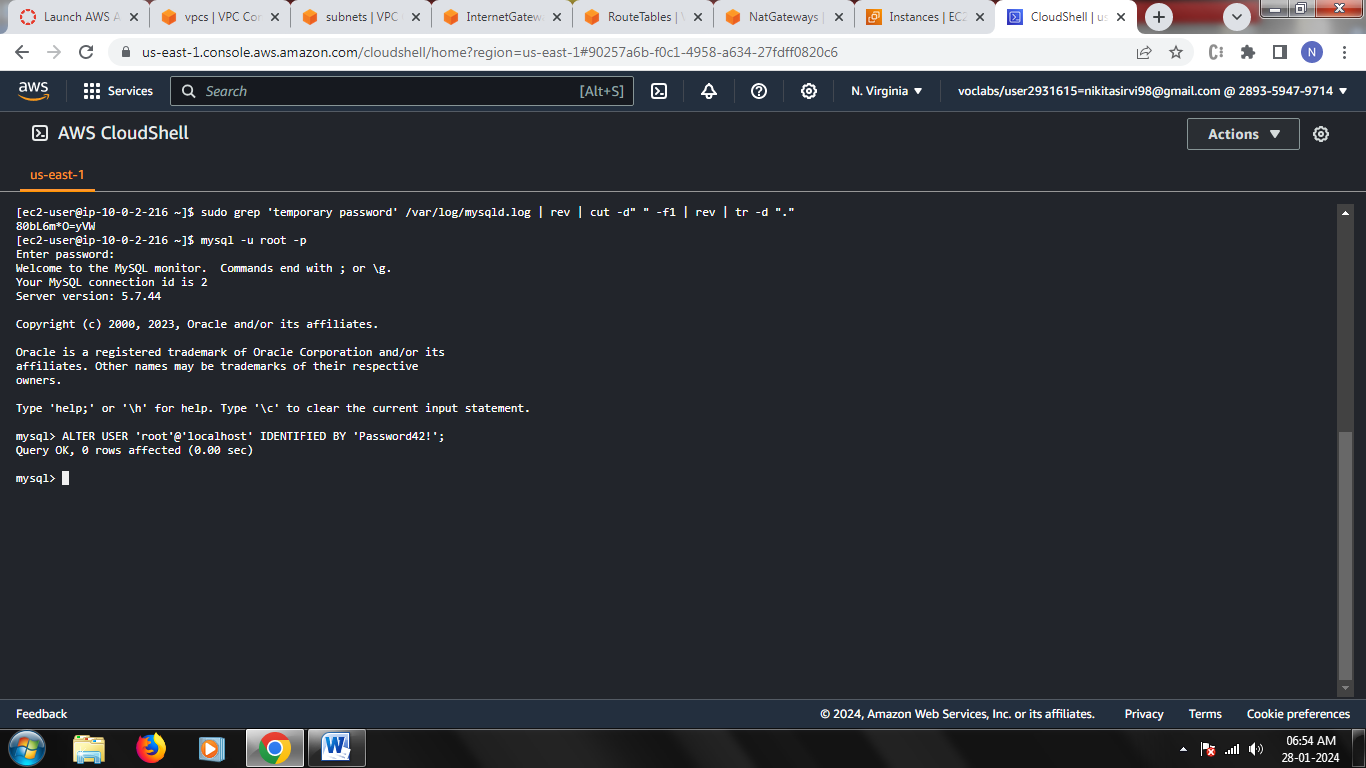




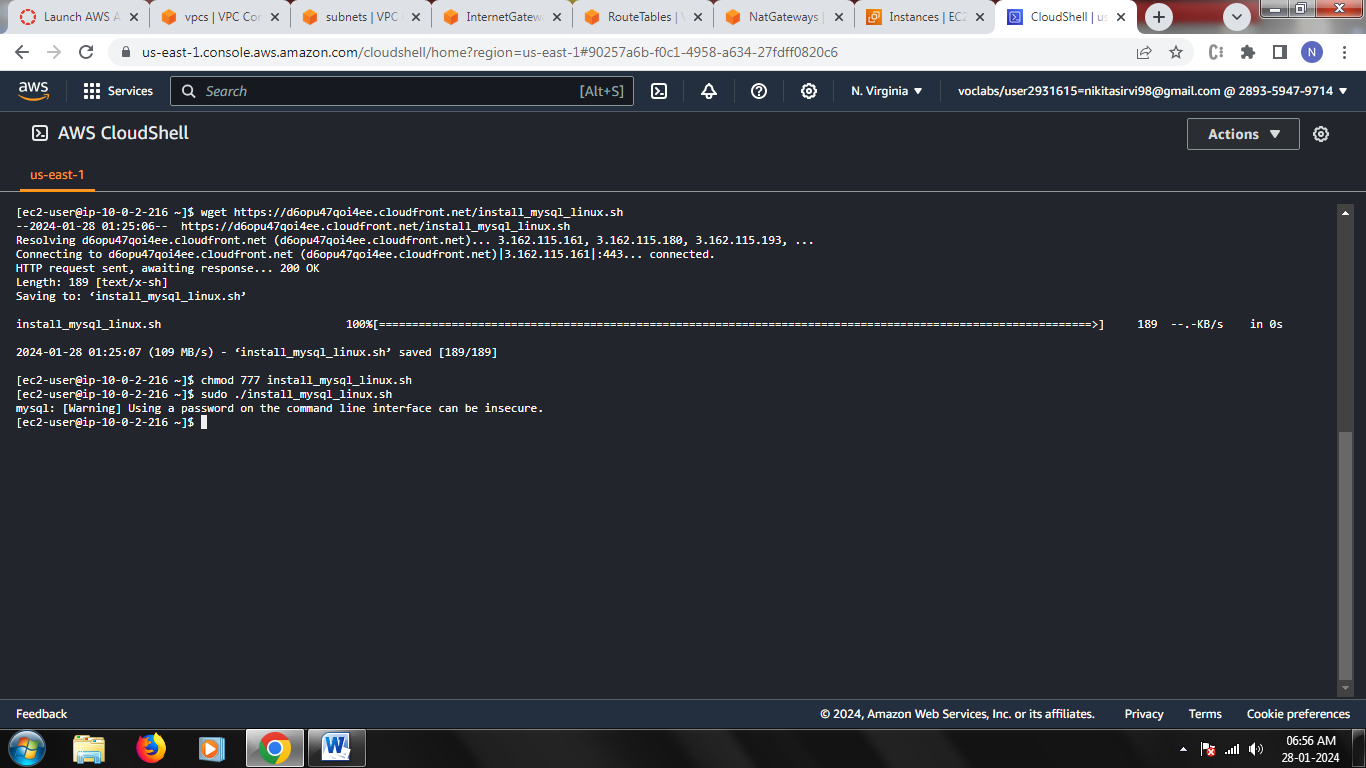




**<Insert screenshot a(2) here>**

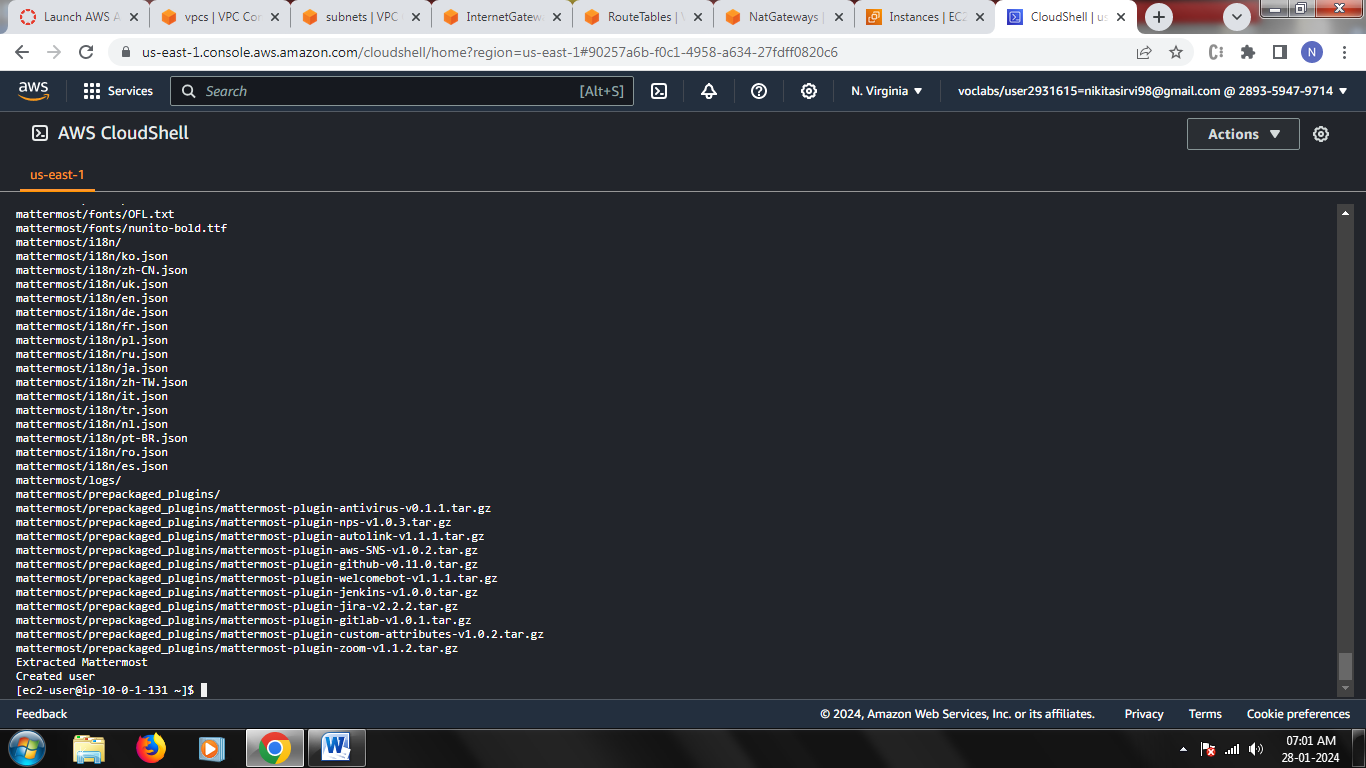


**<Insert screenshot a(3) here>**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step number | b |  |  |  |
| Step name | Installation and configuration of Mattermost |  |  |  |
| Instructions | 1) Enter the following commands after logging into the application server via SSH to install and configure Mattermost  *wget https://d6opu47qoi4ee.cloudfront.net/install\_mattermost\_linux.sh*  *sudo yum install dos2unix -y*  *sudo dos2unix install\_mattermost\_linux.sh*  *chmod 700 install\_mattermost\_linux.sh*  *sudo ./install\_mattermost\_linux.sh <private IP of MySQL server>*  Example : sudo ./*install\_mattermost\_linux* 173.65.34.7  *sudo chown -R mattermost:mattermost /opt/mattermost*  *sudo chmod -R g+w /opt/mattermost*  *cd /opt/mattermost*  *sudo -u mattermost ./bin/mattermost*  2) Check whether the server has been successfully deployed by navigating to the following URL in your web browser. The web page might take a couple of minutes to load.  <public IP of the application server>:8065 | | | |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |  |  |  |  |
| Expected screenshots | 1. Executing the script 2. Starting the Mattermost server 3. Accessing the application via web browser |  |  |  |

**<Insert screenshot b(1) here>**



**<Insert screenshot b(2) here>**



**<Insert screenshot b(3) here>**

