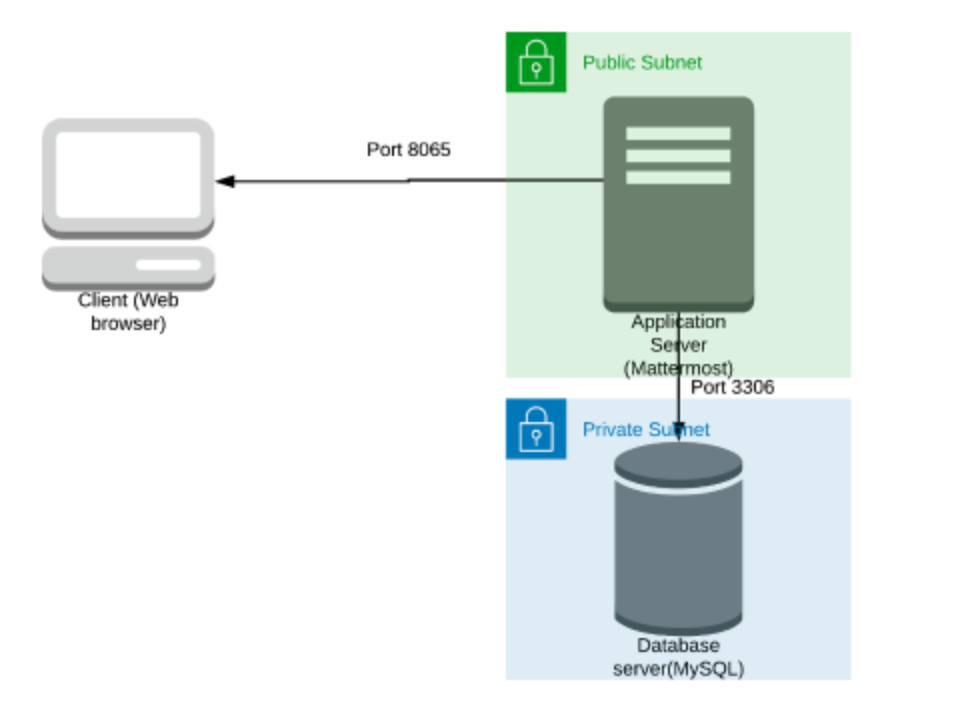
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Declaration** |  | | | | |
| Questions in this exercise are intentionally complex and could be convoluted or confusing. This is by design and to simulate real life situations where customers seldom give crystal clear requirements and ask unambiguous questions. | | | | | |
|
|
| I have read the above statement and agree to these conditions | | | | | |
| I AGREE | RAHUL JAIN | | | | |
| <Enter your name above this line to indicate that you are in agreement> | | | | |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| **Instructions** |  |  |  |  |  |
| Every screenshot requested in this workbook is compulsory and carries 0.5 marks | | | | | |
| Your AWS account ID must be clearly visible in every screenshot using the AWS console; missing id or using someone else's id is not permitted. Such cases will be considered as plagiarism and severe penalty will be imposed. | | | | | |
| All screenshots must be in the order mentioned under "Expected Screenshots" for every step | | | | | |
| DO NOT WAIT UNTIL THE LAST MINUTE. The program office will not extend the project submission deadline under any circumstances. | | | | | |
| The file should be renamed in the format BATCH\_FIRSTNAME\_LASTNAME\_PROJECT1.  For example: PGPCCMAY18\_VIJAY\_DWIVEDI\_PROJECT1.pdf | | | | | |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| **Resource Clean Up** | |  |  |  |  |
| Cloud is always pay per use model and all resources/services that we consume are chargeable. Cleaning up when you’ve completed your lab or project is always necessary. This is true whether you’re doing a lab or implementing a project at your workplace. | | | | | |
| After completing the lab, make sure to delete each resource created in reverse chronological order. | | | | | |

**Architecture diagram**

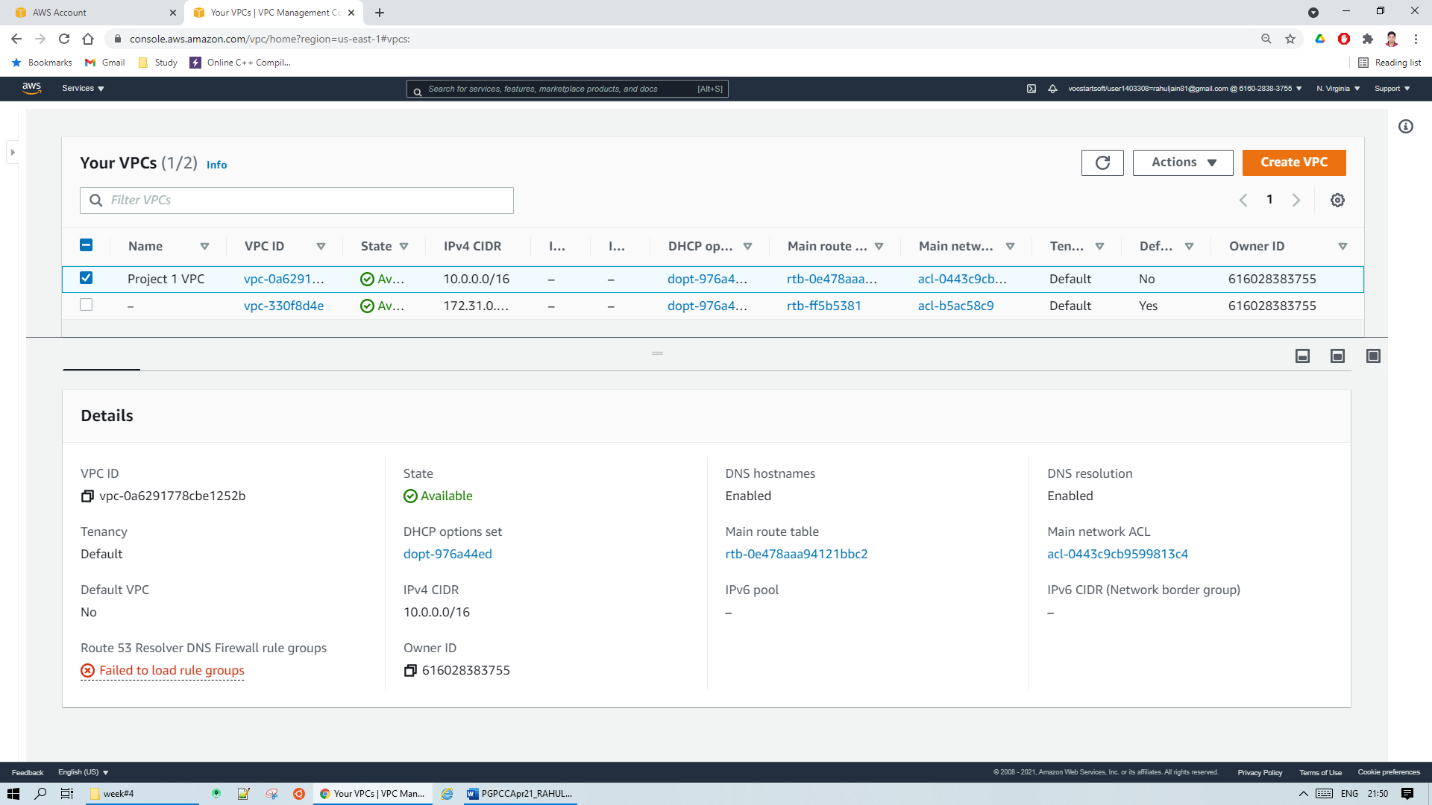
****

|  |  |
| --- | --- |
| **Architecture Implementation** | |
| 1 | Implement 2 different subnets (one public and the other private) in a custom VPC |
| 2 | Install and configure MySQL on an Ubuntu 18.04 instance on the private subnet using the instructions provided. (Hint: Use a bastion host and a NAT instance) |
| 3 | Install and configure Mattermost on an Ubuntu 18.04 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**

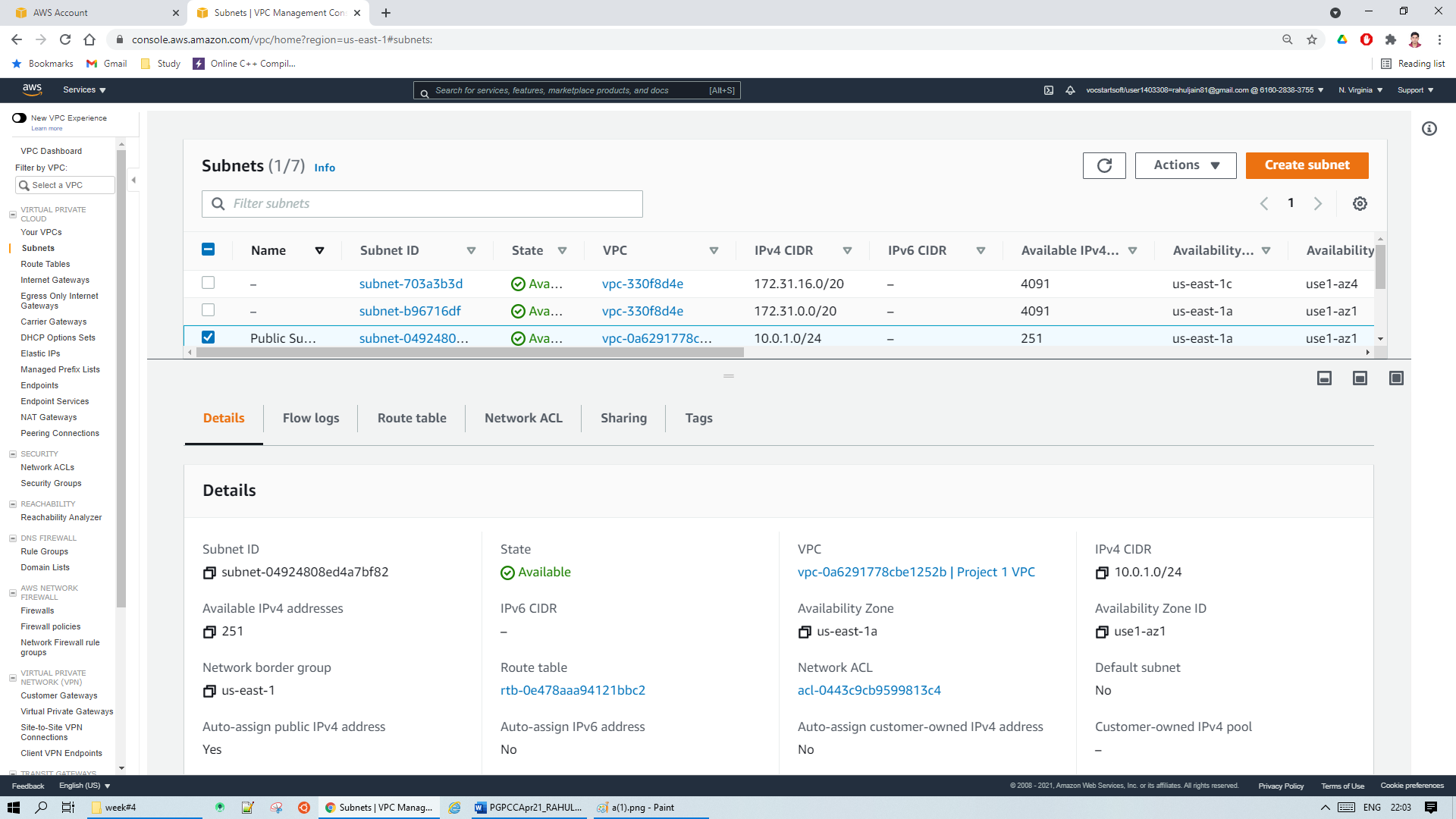
|  |  |  |  |
| --- | --- | --- | --- |
| Step number | a |  |  |
| Step name | Creation of VPC |  |  |
| Instructions | 1) Navigate to VPC using the Services button at the top of the screen  2) Select "Your VPCs" on the left side of the screen  3) Click on "Create VPC"  4) Enter the following fields :  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>**

****

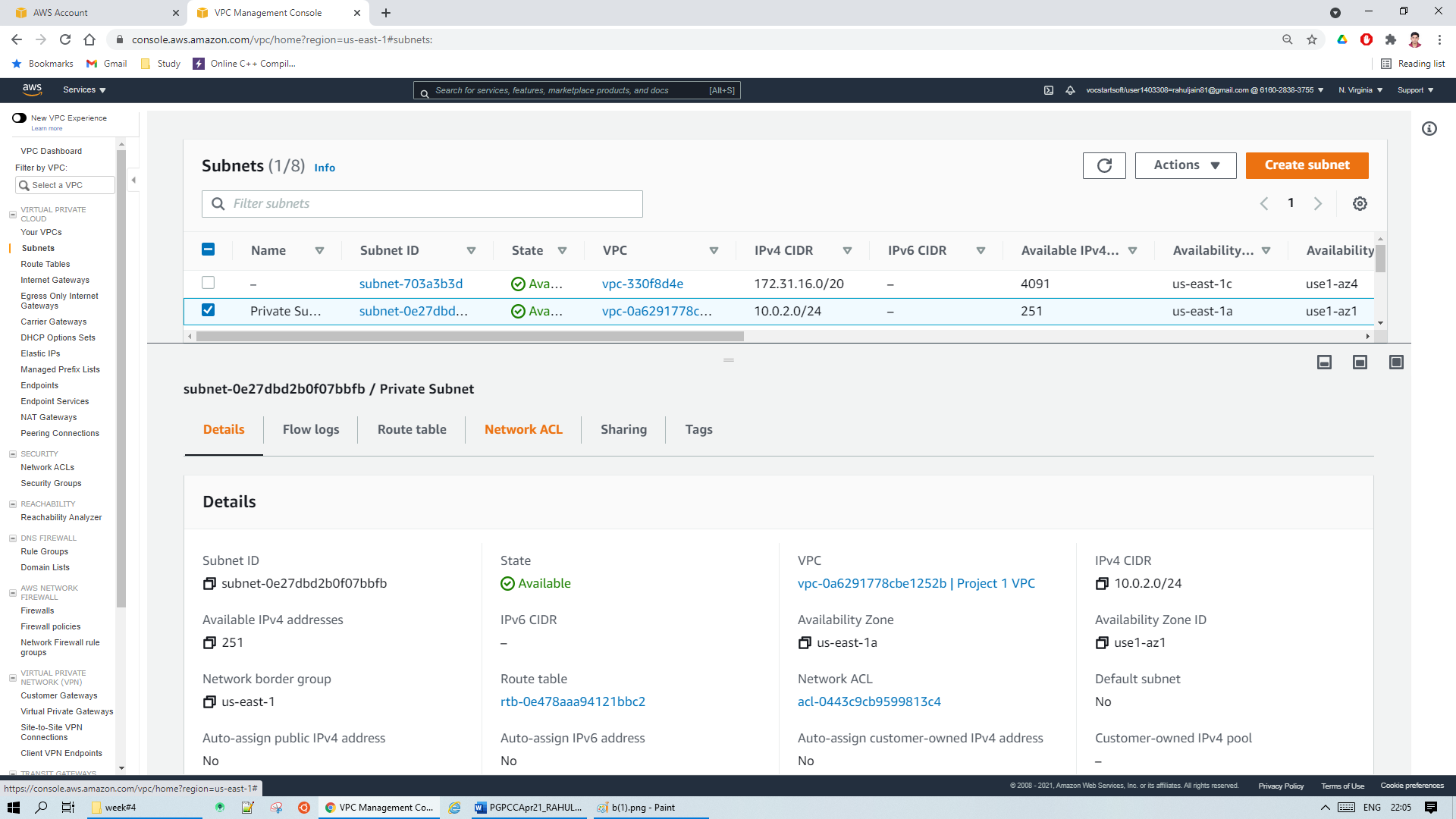
|  |  |  |
| --- | --- | --- |
| Step number | b |  |
| Step name | Creation of public subnet |  |
| Instructions | 1) Navigate to VPC->Subnets  2) Click on "Create Subnet"  3) Enter the following fields  Name tag : Public Subnet  VPC : Select the Project 1 VPC  IPv4 CIDR block : 10.0.1.0/24  The other 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 |

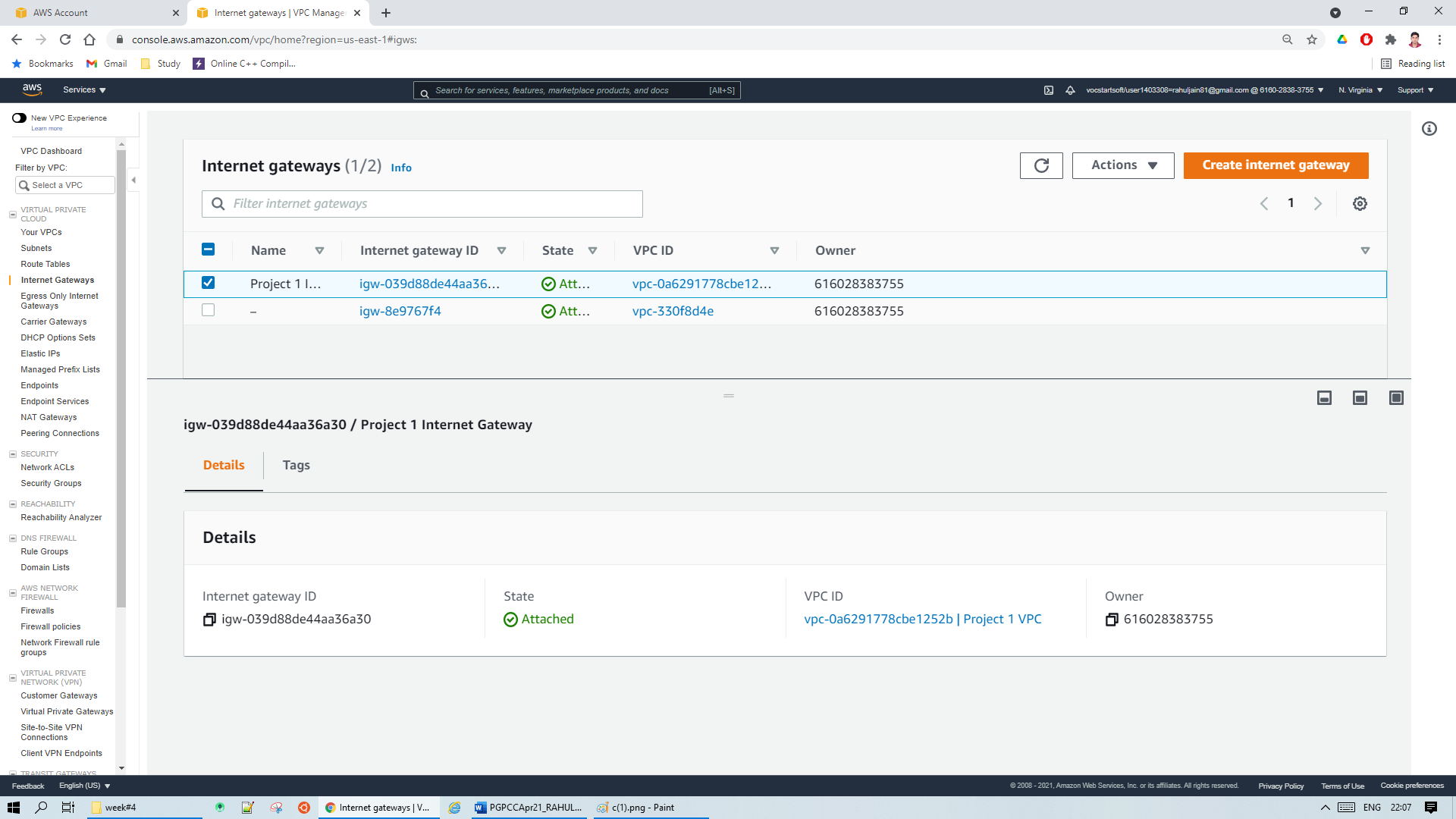
**<Insert Screenshot c(1) here>**

****

**Step 2 : Internet Gateway and VPC**

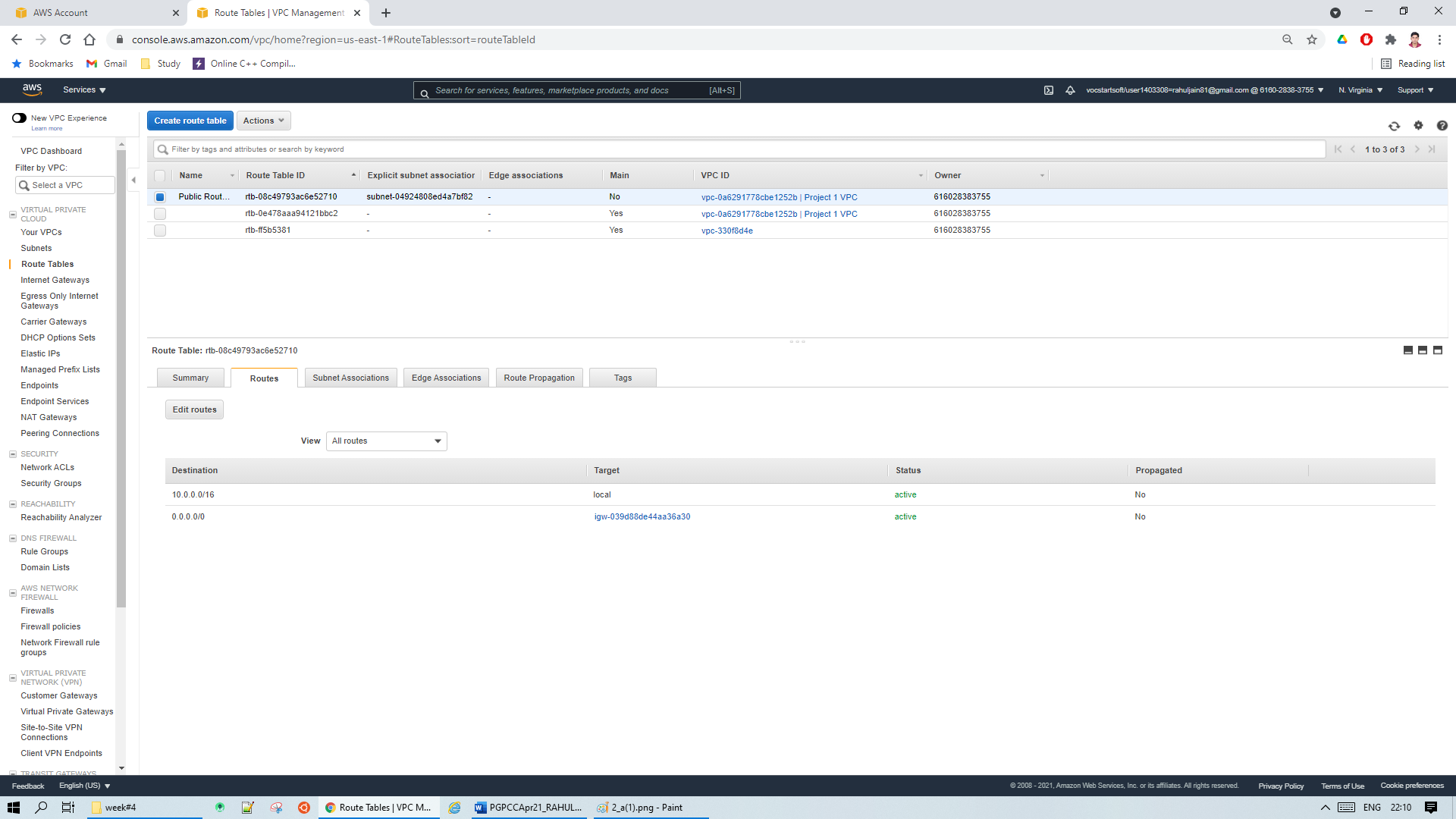
|  |  |  |
| --- | --- | --- |
| Step number | a |  |
| Step name | Creation and Configuration of Internet Gateway |  |
| Instructions | 1) Navigate to VPCs->Internet Gateway  2) Click on "Create Internet Gateway"  3) Enter the name tag "Project 1 Internet Gateway" and click on "Create Internet Gateway"  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 >**

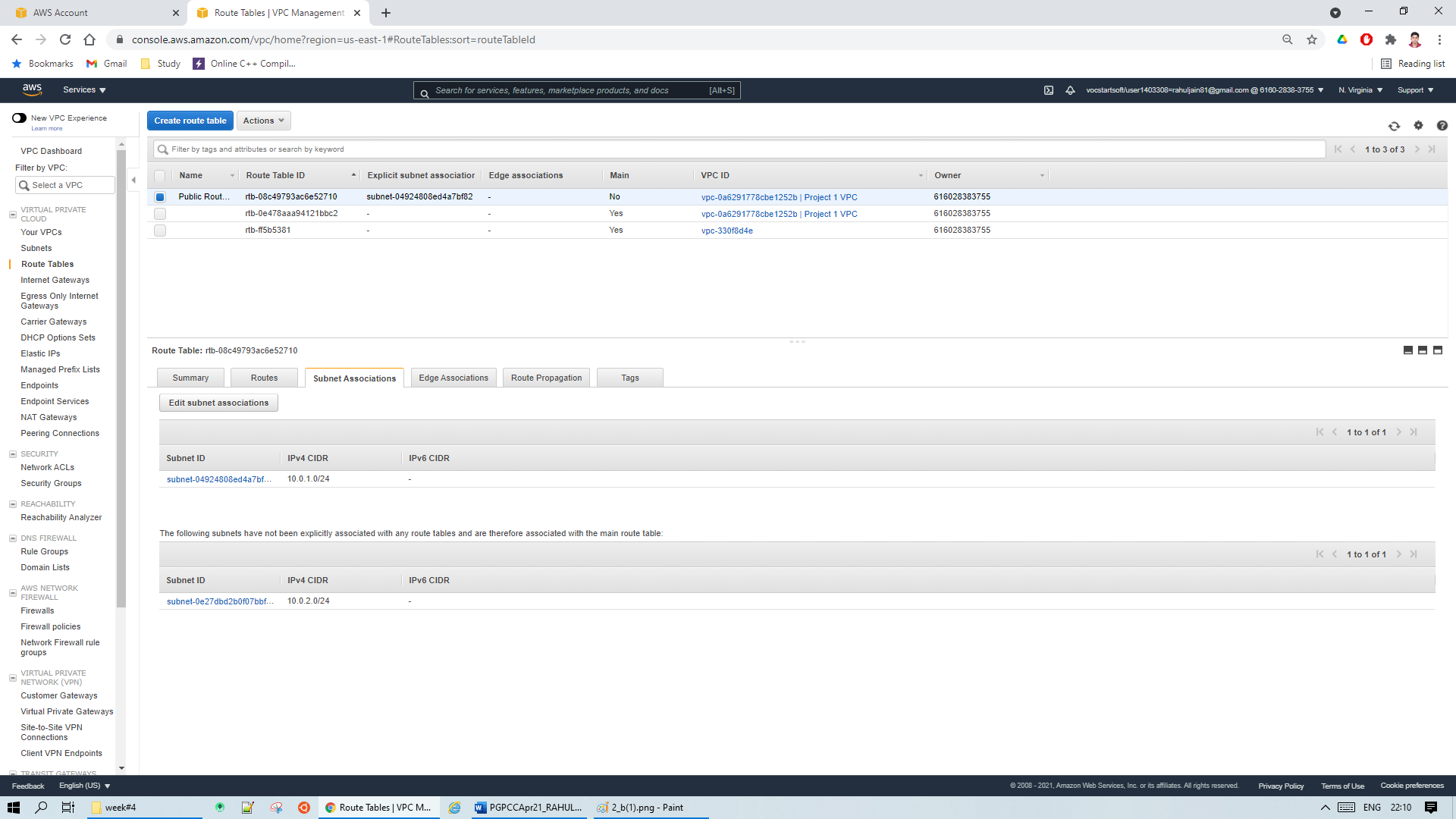
****

|  |  |  |
| --- | --- | --- |
| 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>**

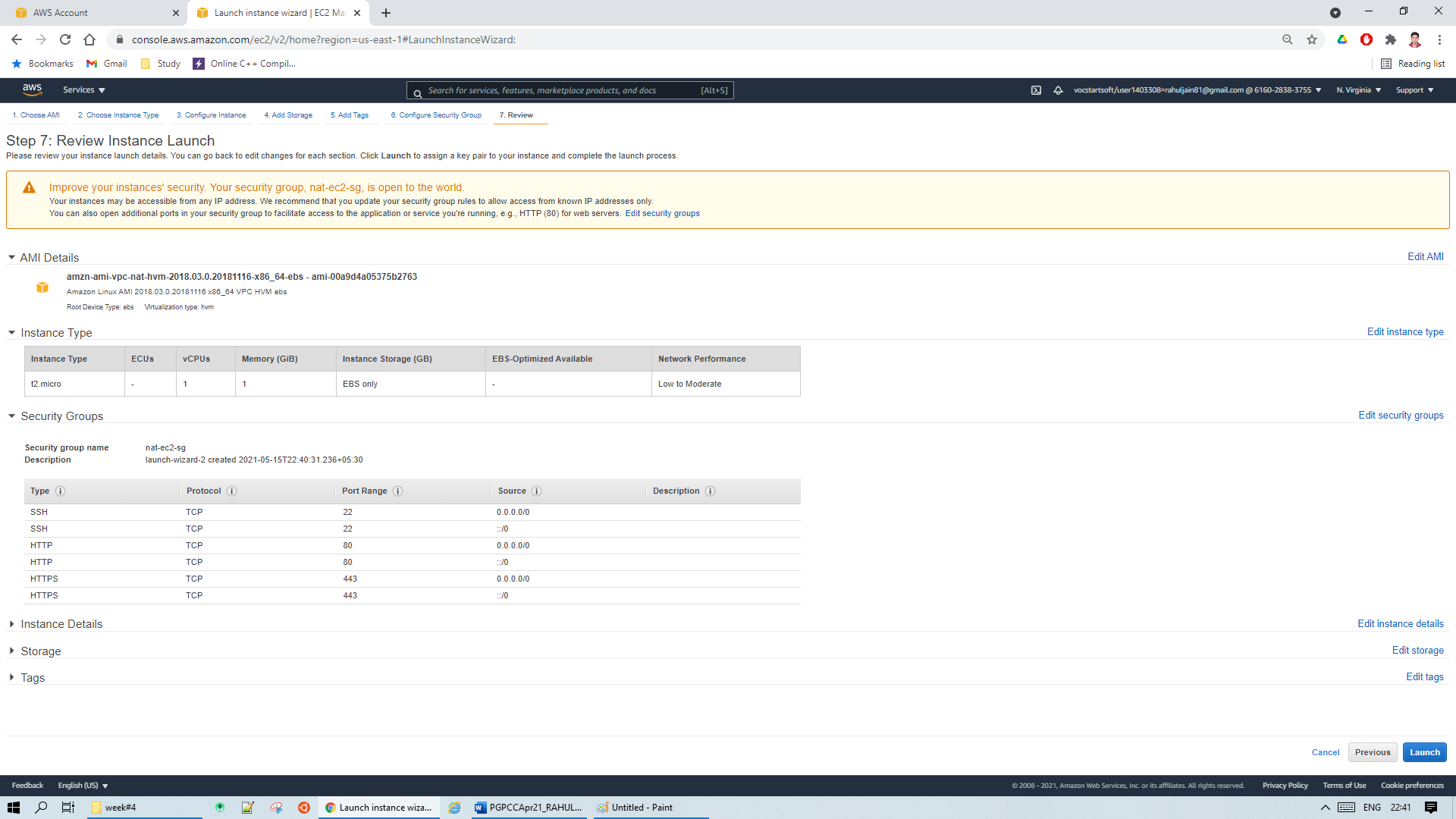
****

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

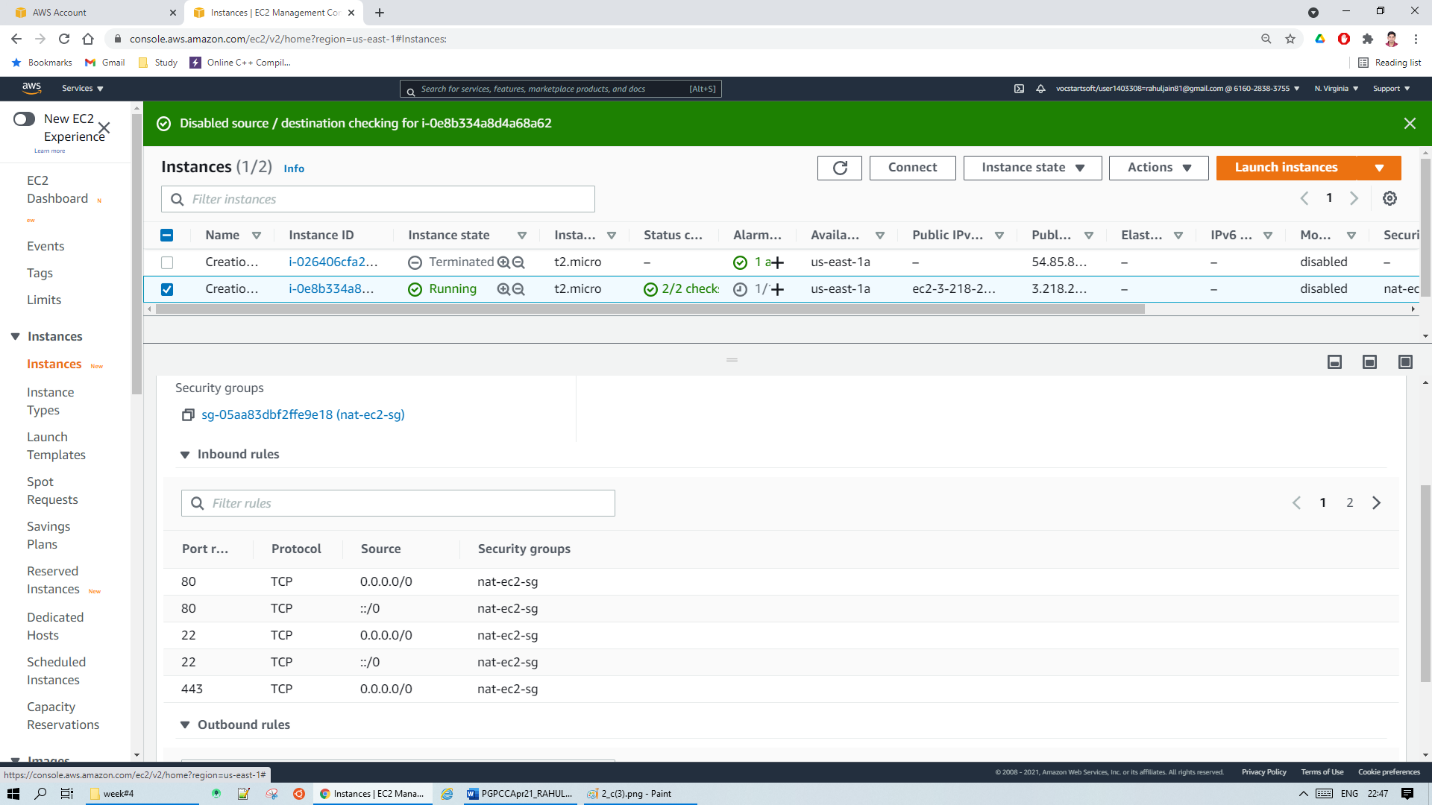
****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step number | c |  |  |  |
| Step name | Creation of NAT instance |  |  |  |
| 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 Community AMI for NAT instance  - Select the instance type t2.micro  - Select Network as "Project 1 VPC" and subnet as "Public Subnet"  - For the security group, open the ports 22,80 and 443 for source set to "Anywhere"  4) Launch the instance  5) Once the instance is launched, select the instance and click on Actions->Networking -> Change Source/Dest. Check and disable Source/Destination Check |  |  |  |
| Expected screenshots | 1. Configure Instance Details 2. Security Group Rules 3. Instance after creation |  |  |  |
|  |  |  |  |  |

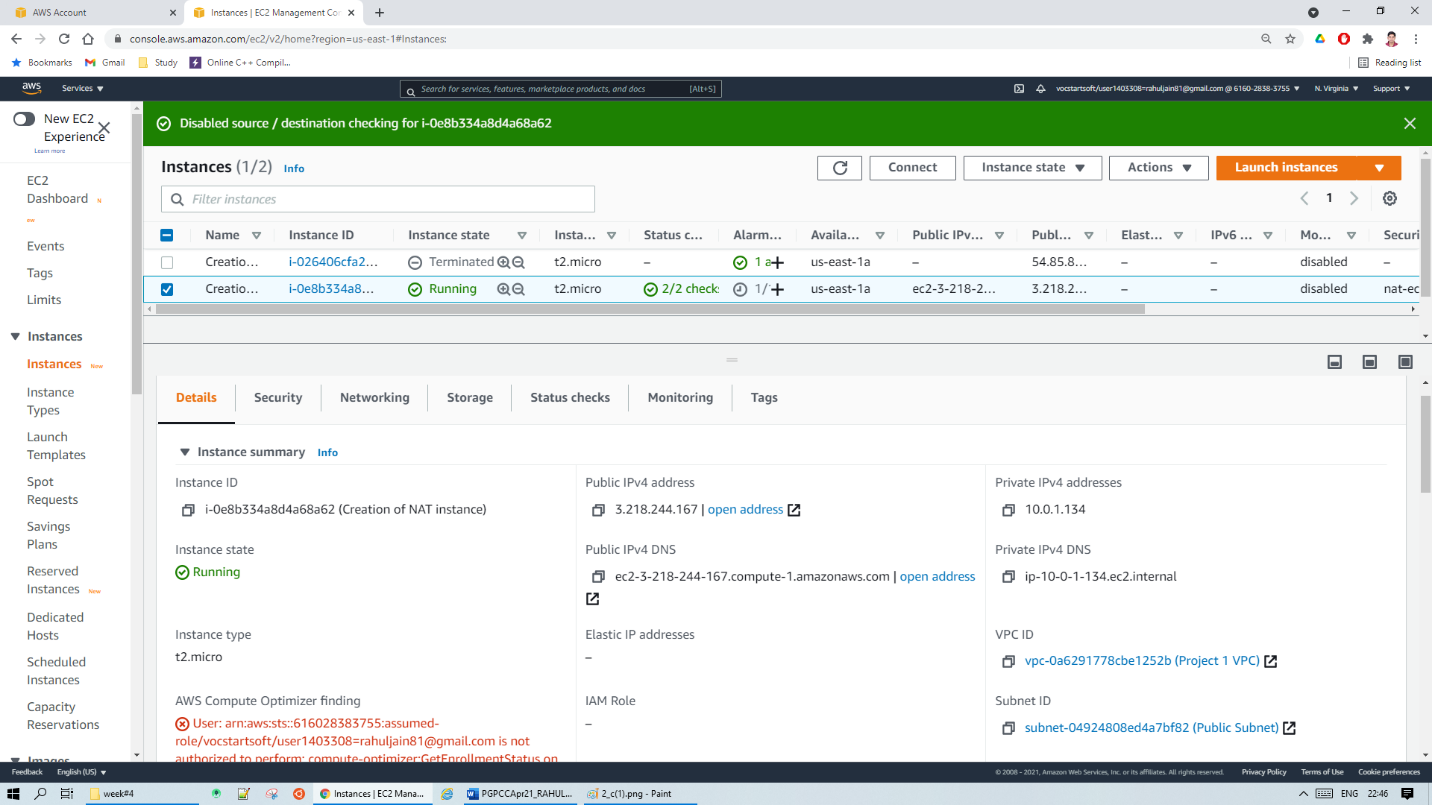
**<Insert Screenshot c(1) here>**

****

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

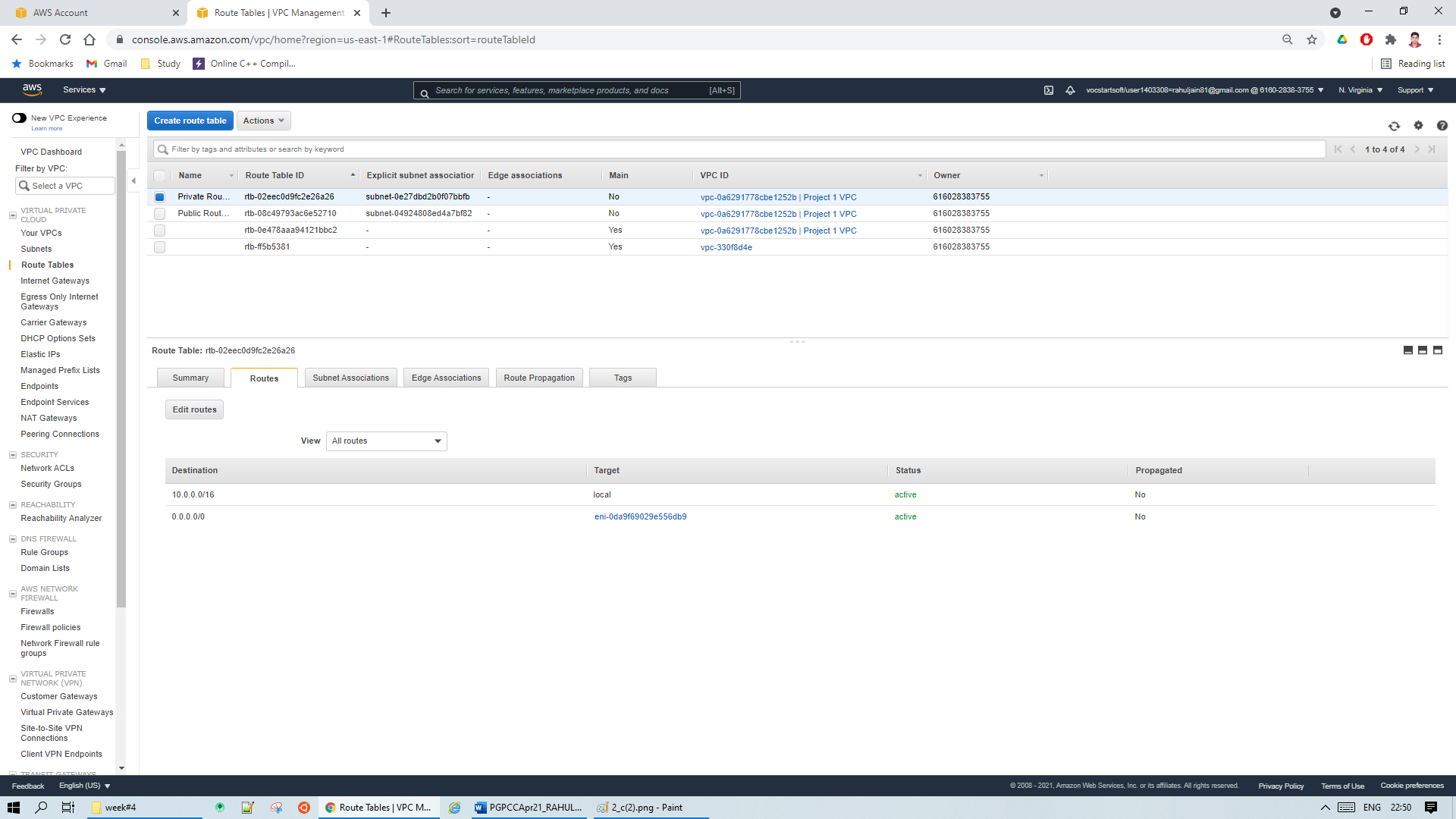
****

**<Insert Screenshot c(3) here>**

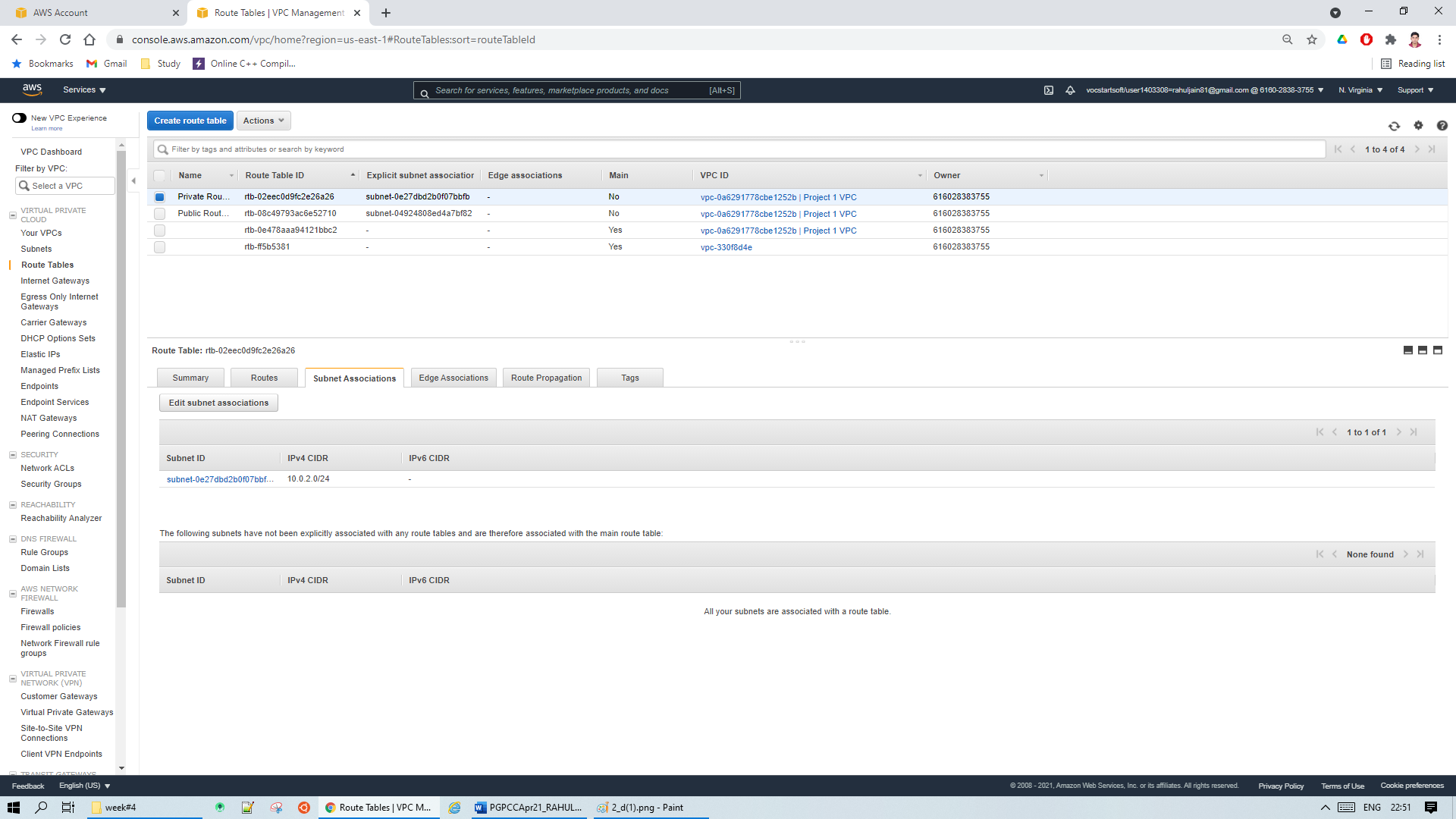
****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 Instances and select the NAT instance 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 >**

****

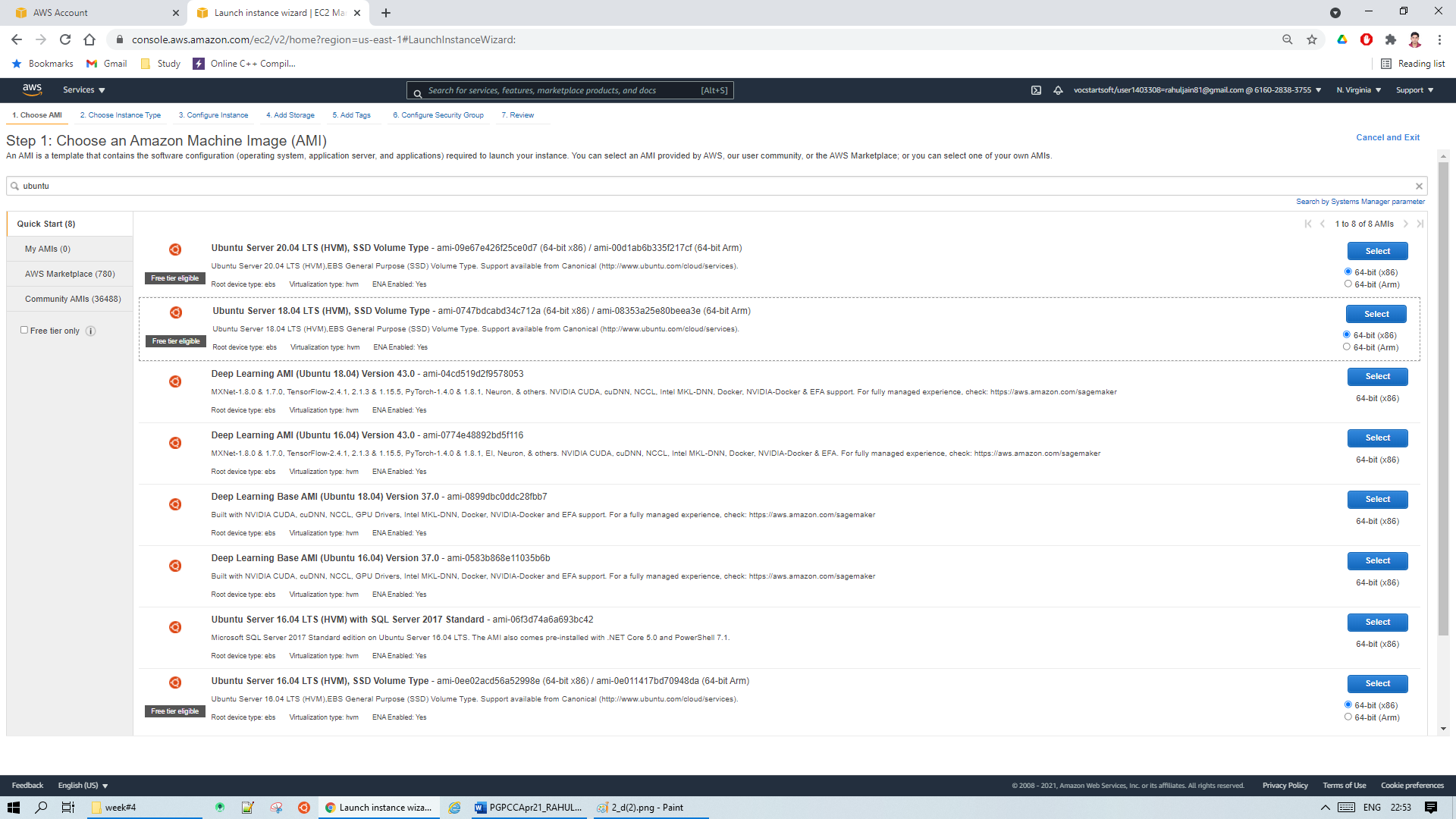
**<Insert Screenshot for d(2) here>**

****

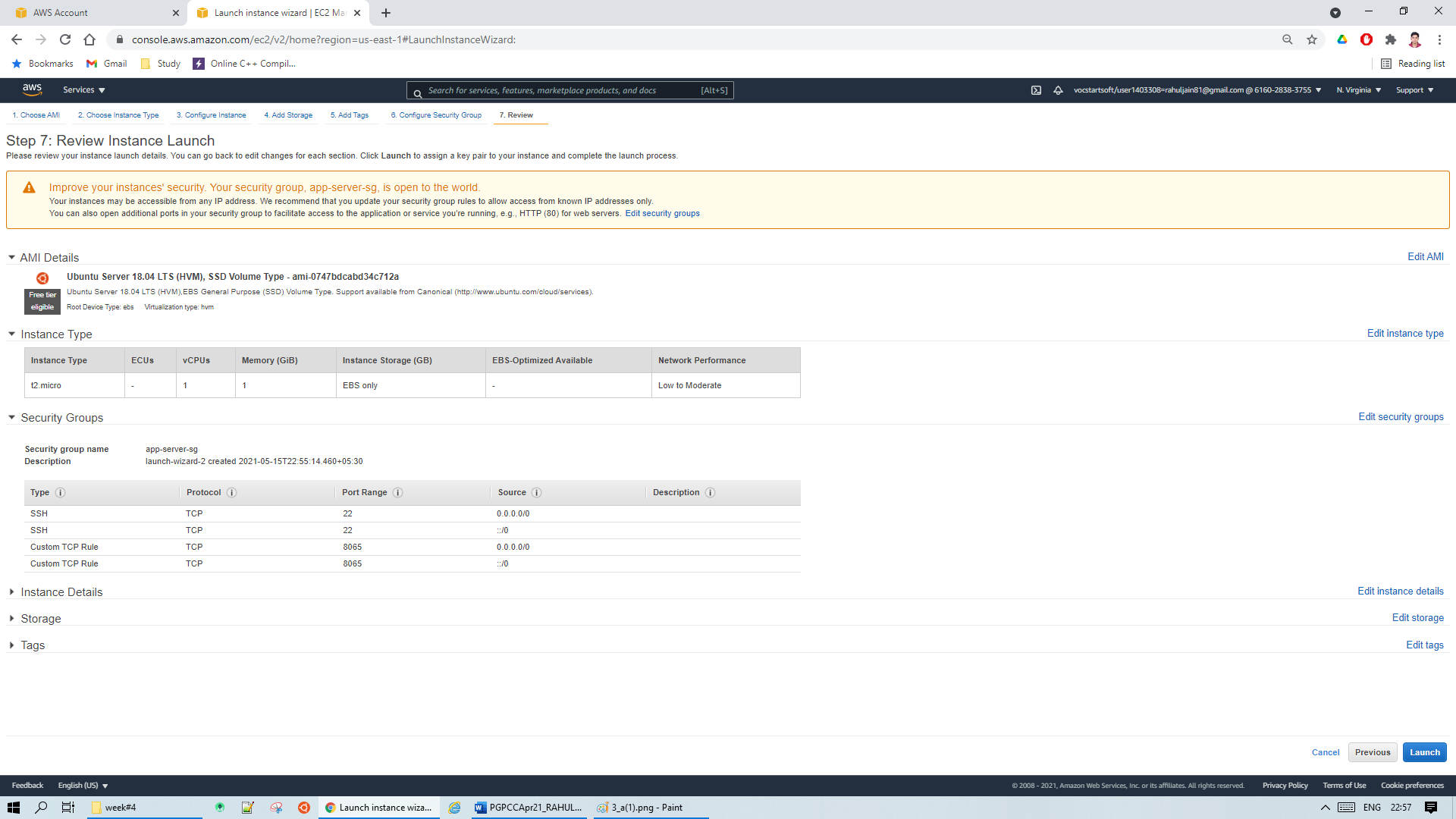
**Step 3 : Creation of database and application servers**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 Ubuntu 18.04 LTS  - Select the instance type t2.micro  - Select Network as "Project 1 VPC" and subnet as "Public Subnet"  - For the security group, open the ports 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 |  |  |  |

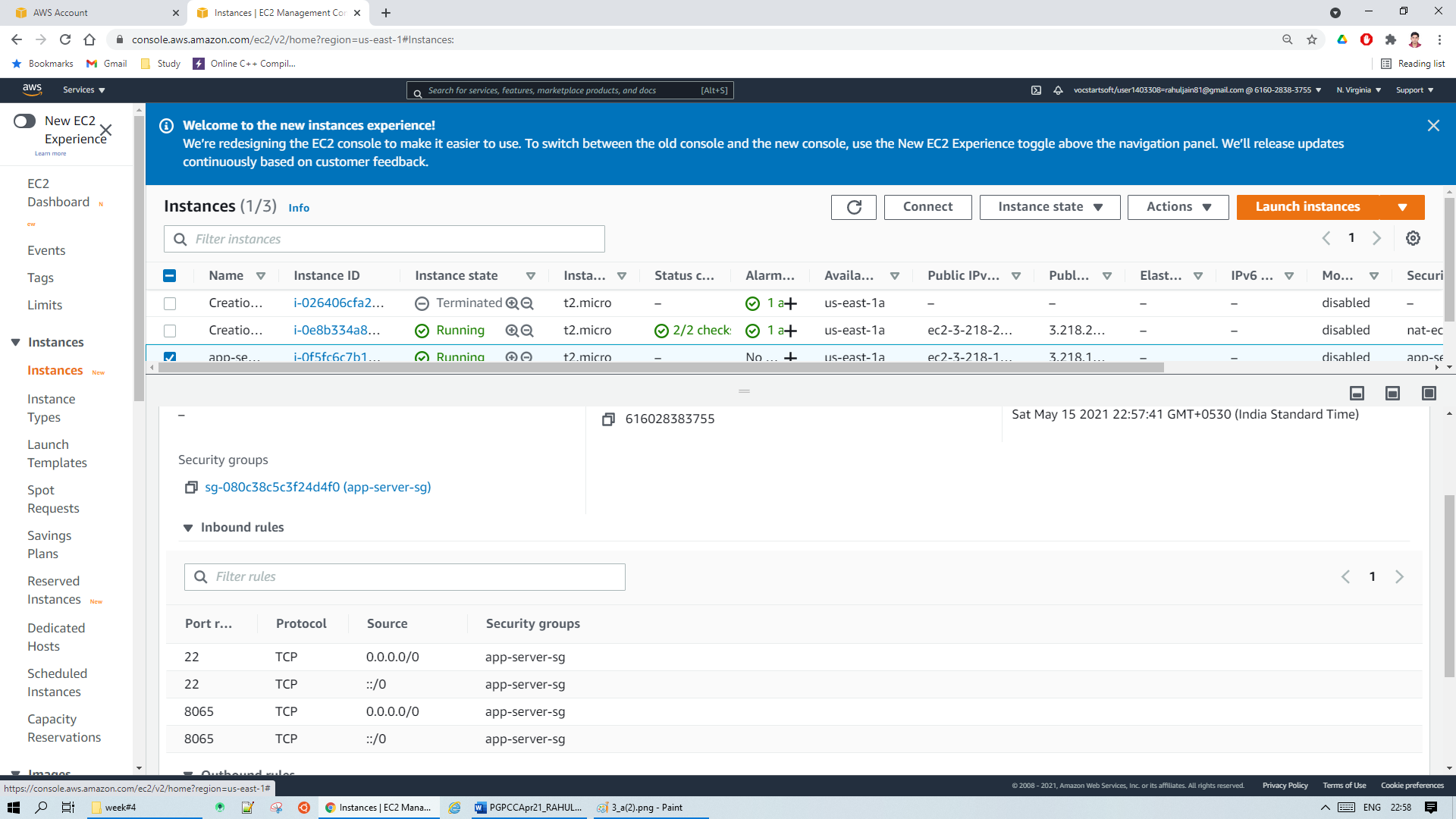
**<Insert screenshot a(1) here>**

****

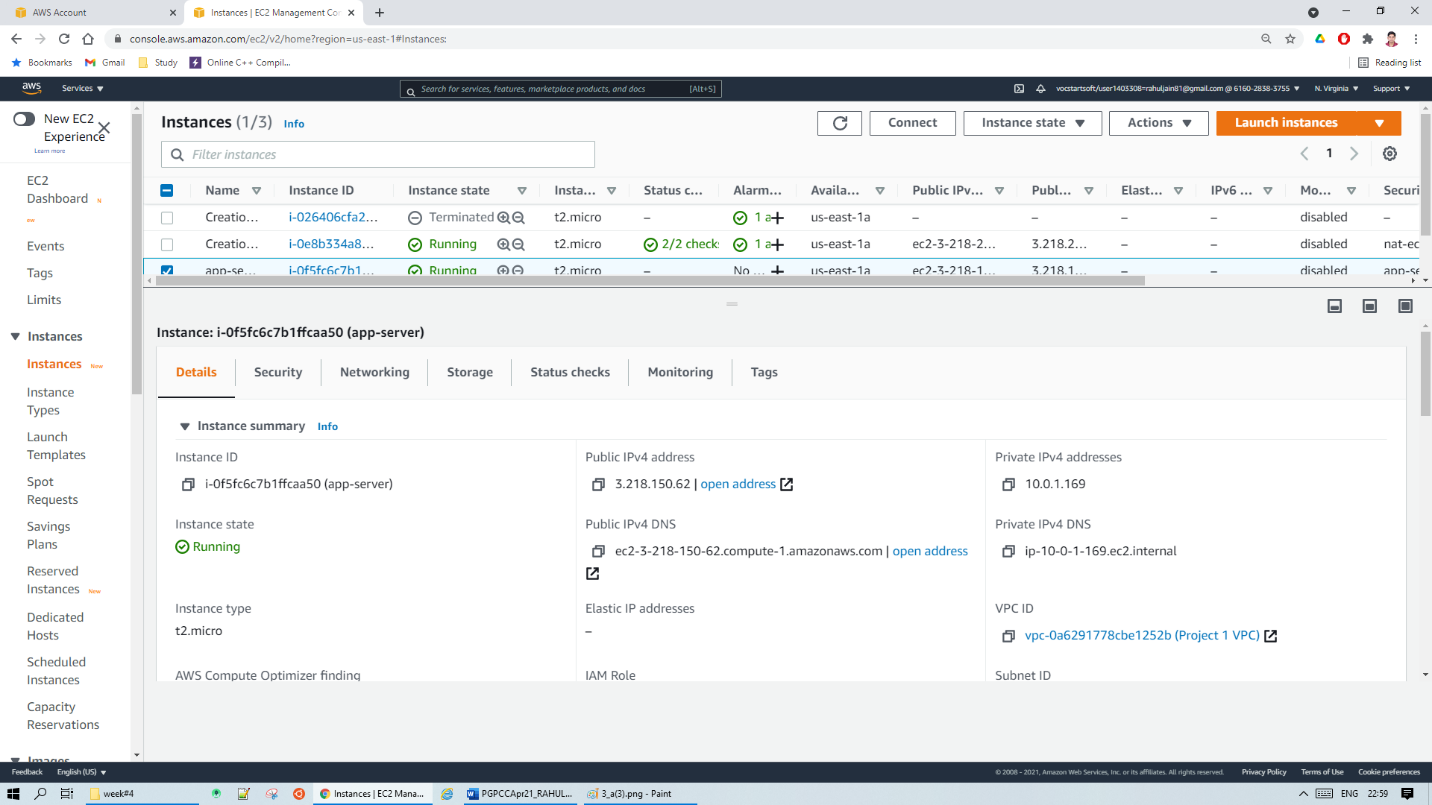
**<Insert screenshot a(2) here>**

****

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

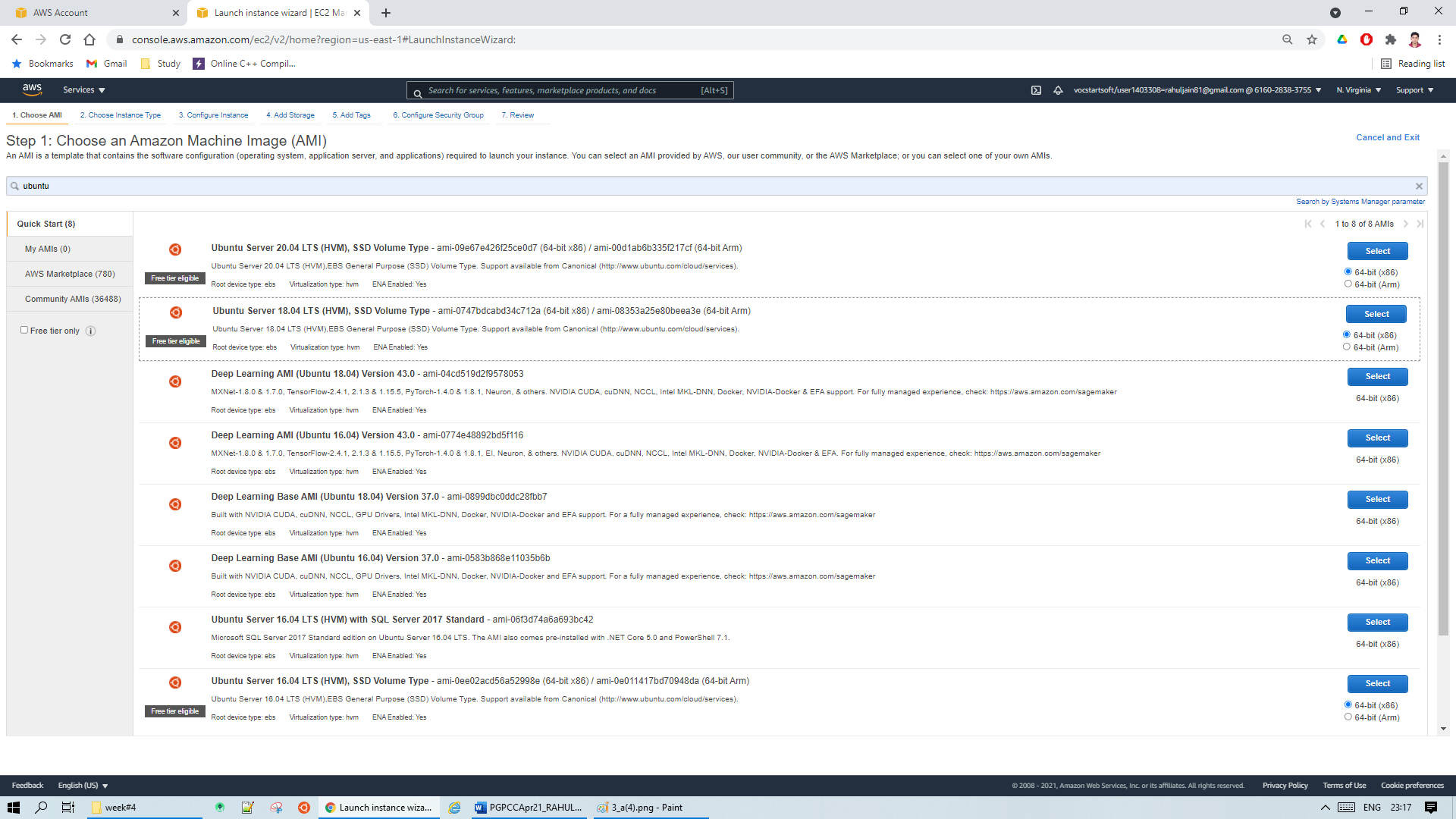
****

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

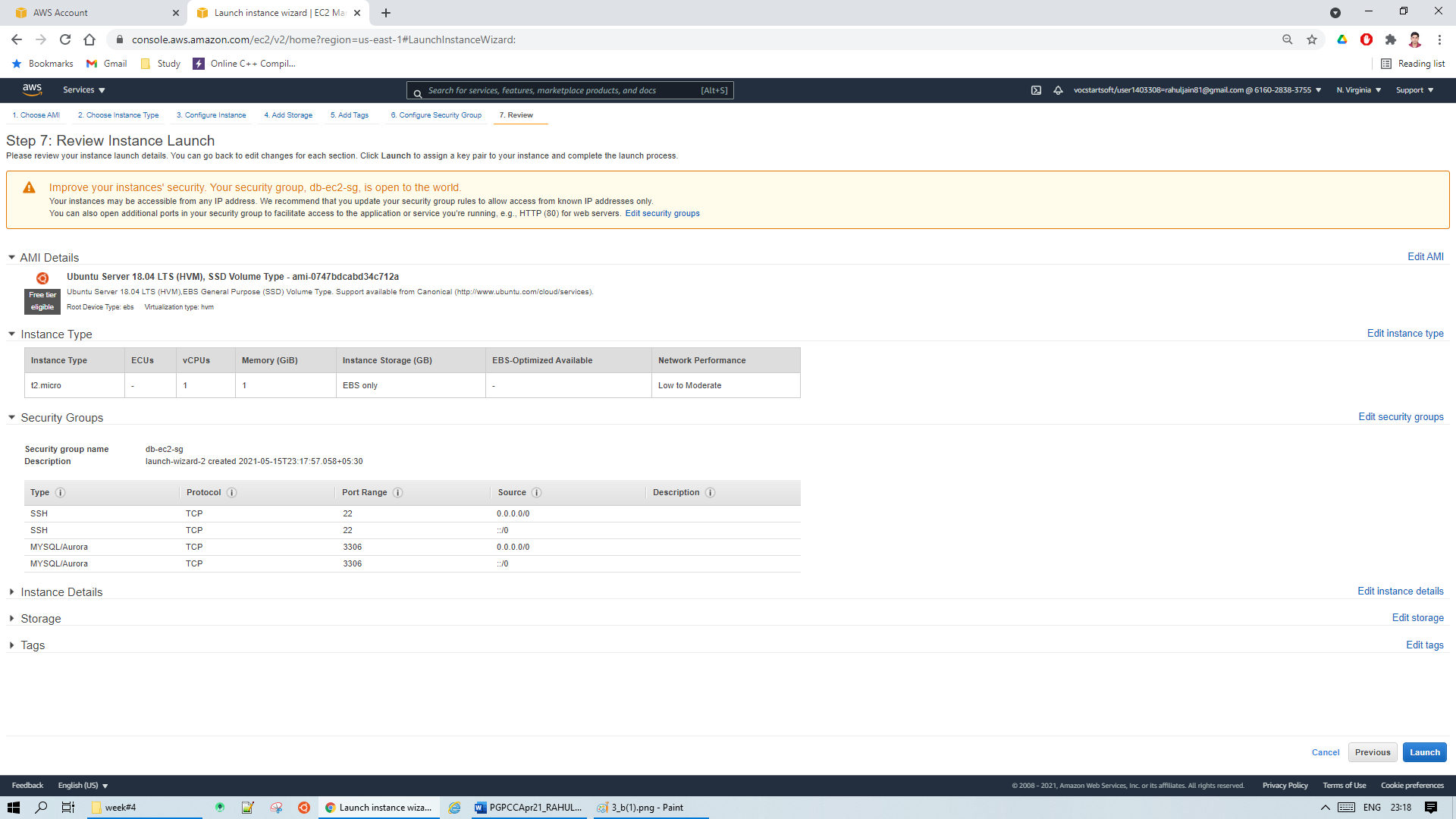
****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 Ubuntu 18.04 LTS  - Select the instance type t2.micro  - Select Network as "Project 1 VPC" and subnet as "Private Subnet"  - For the security group, open the ports 22 and 3306 for source set to "Anywhere"  4) Launch the instanch 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 |  |  |  |

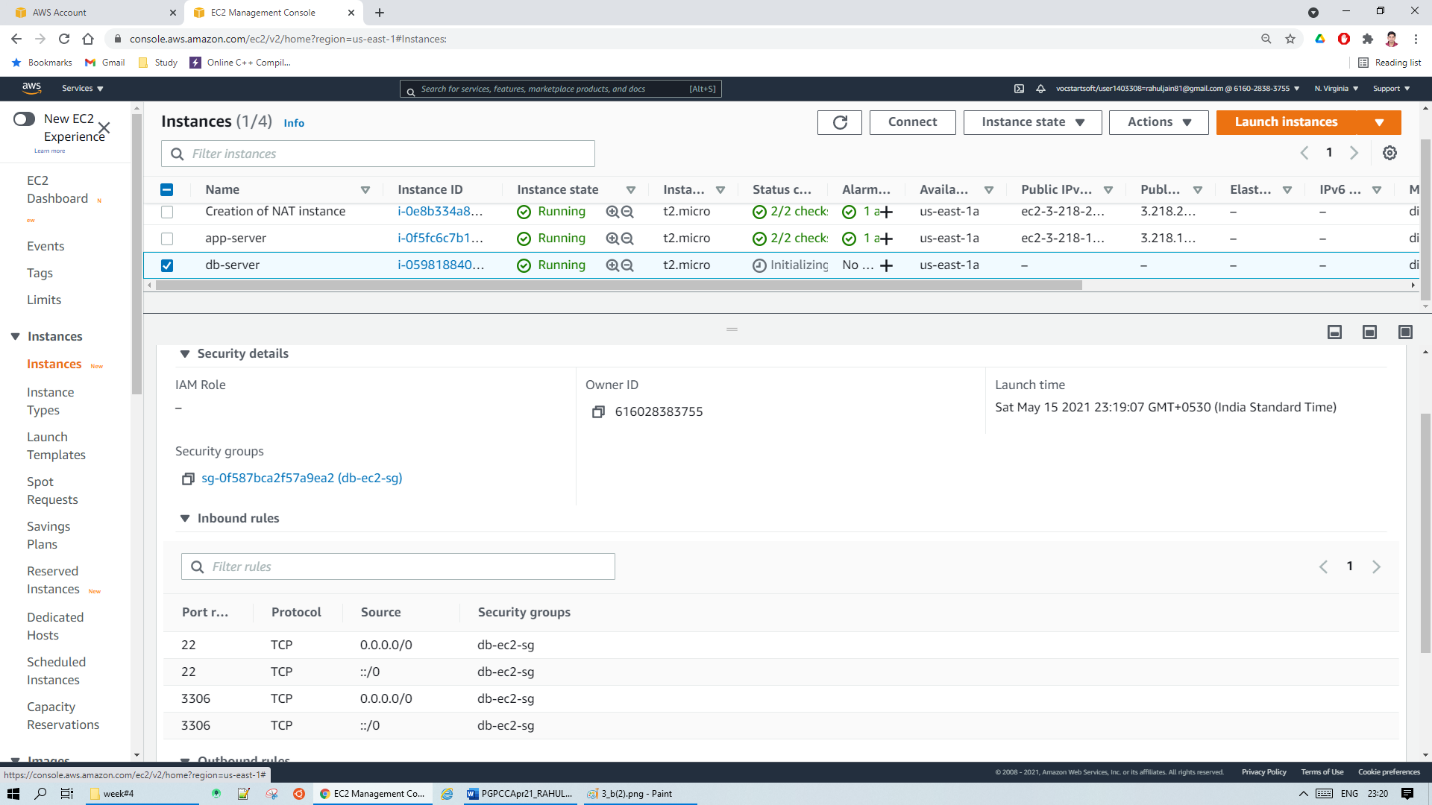
**<Insert screenshot b(1) here>**

****

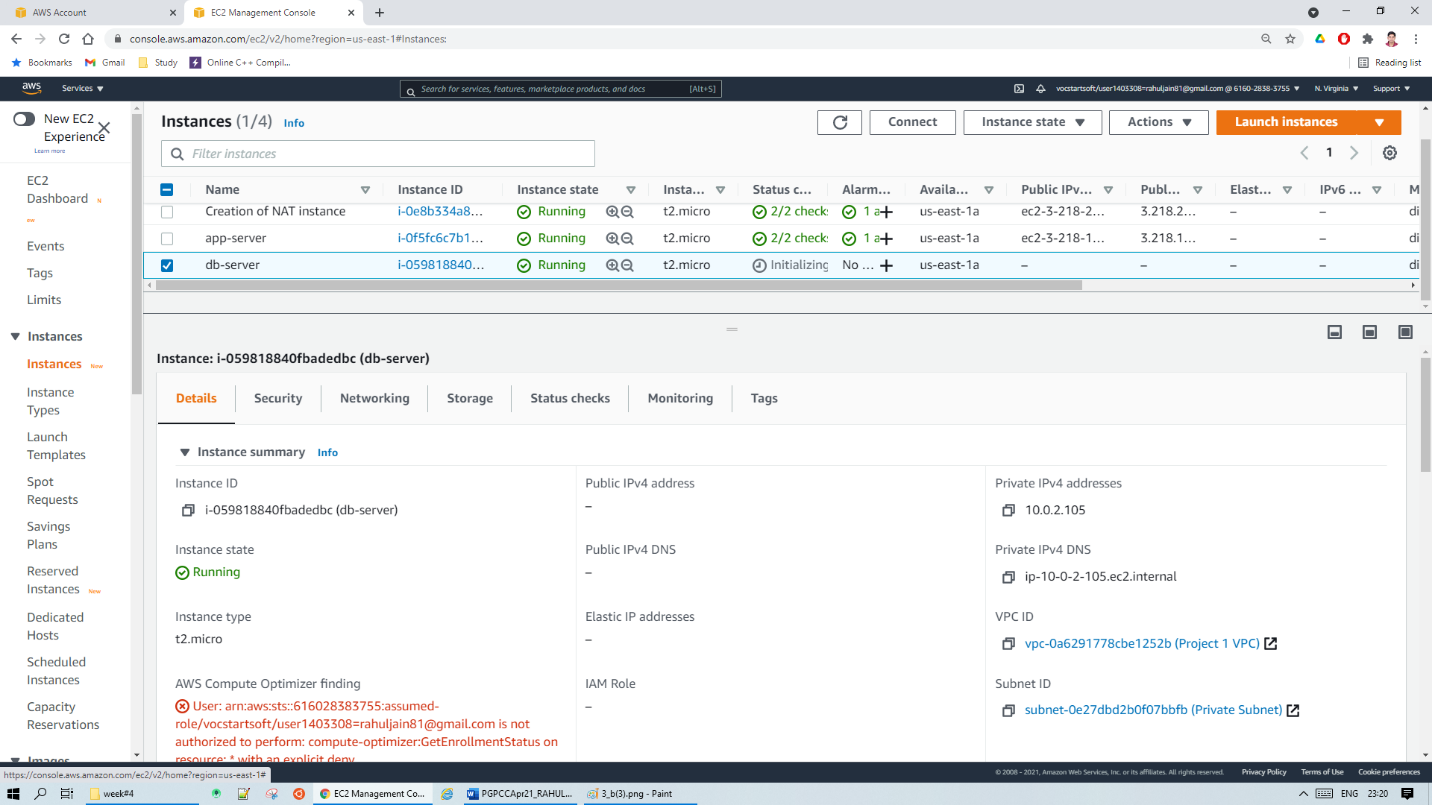
**<Insert screenshot b(2) here>**

****

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

****

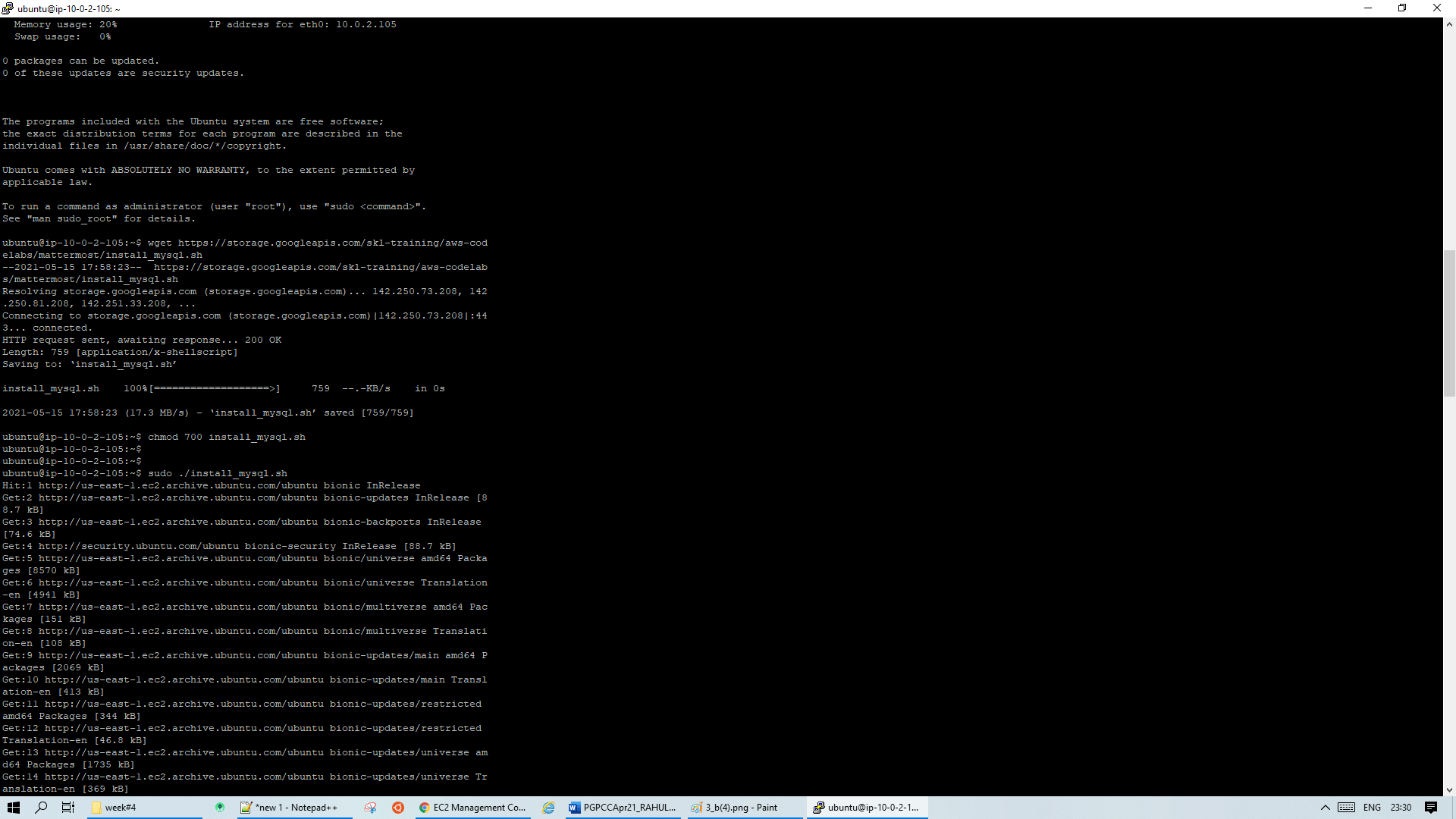
**<Insert screenshot b(4) here>**



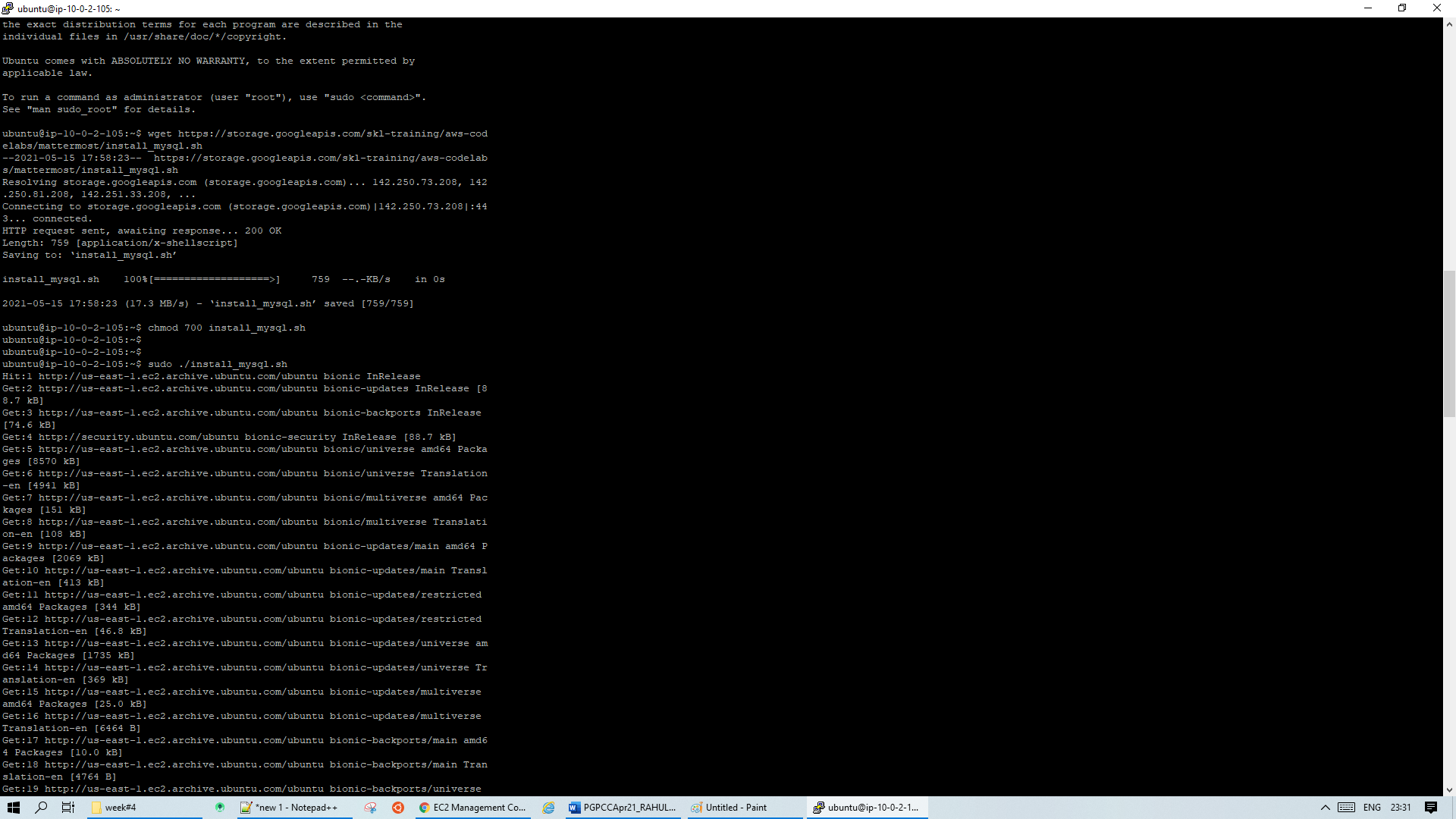
**Step 4: Application and Database Installation and Testing**

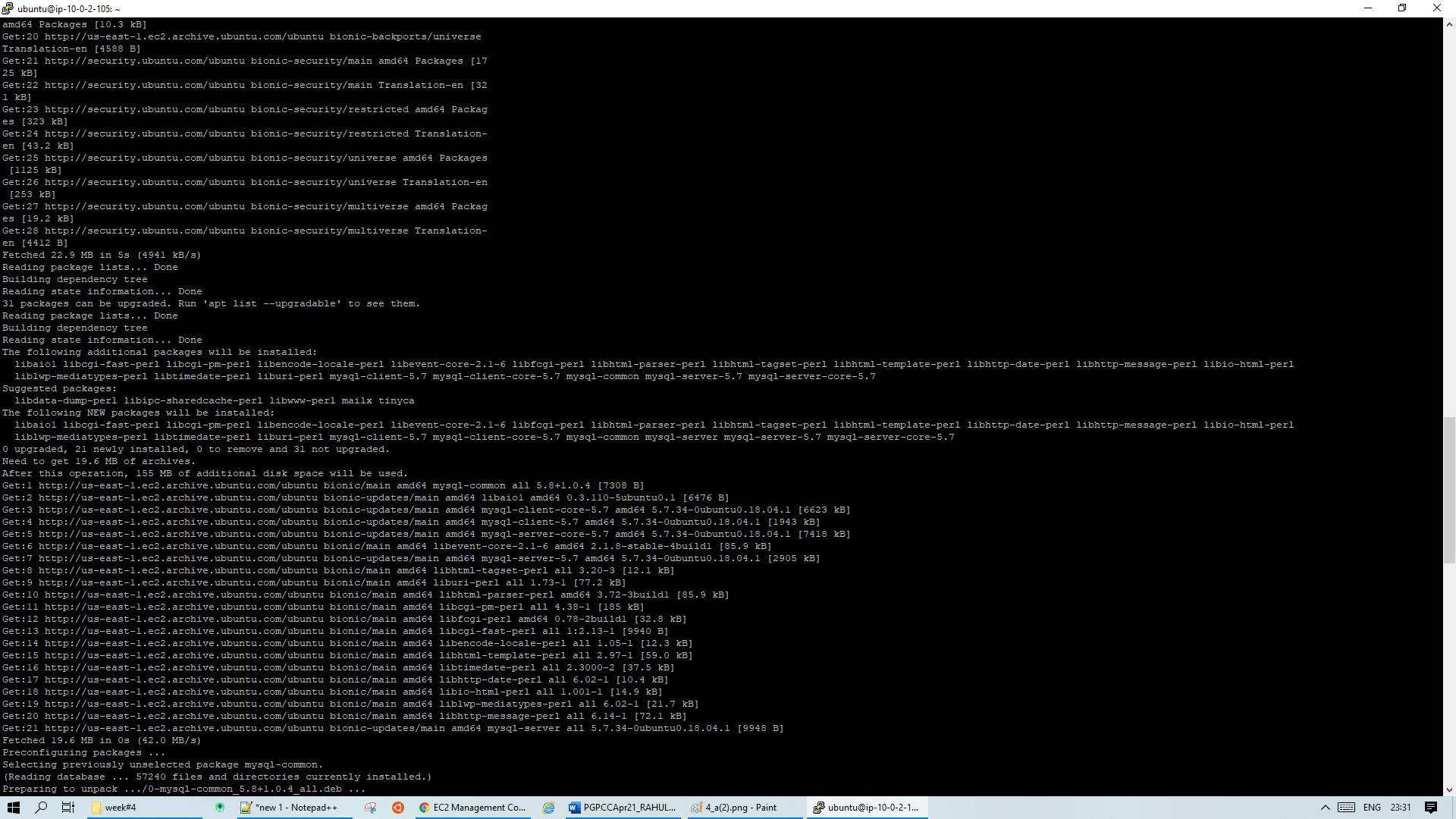
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step number | a |  |  |  |
| Step name | Installation and configuration of MySQL |  |  |  |
| Instructions | 1) Copy the database pem file into the application server using the below command  *scp -i <application server pem file> <database server pem file > ubuntu@<application server public IP>:/home/ubuntu*  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> ubuntu@<private IP of database server>*  4) Enter the following commands to install and configure MySQL on the database server *wget https://storage.googleapis.com/skl-training/aws-codelabs/mattermost/install\_mysql.sh*  *chmod 700 install\_mysql.sh*  *sudo ./install\_mysql.sh*  5) Type *exit* to exit the database server and go back to the application server | | | |
| Expected screenshots | 1. Downloading of the provided script 2. Executing the script |  |  |  |

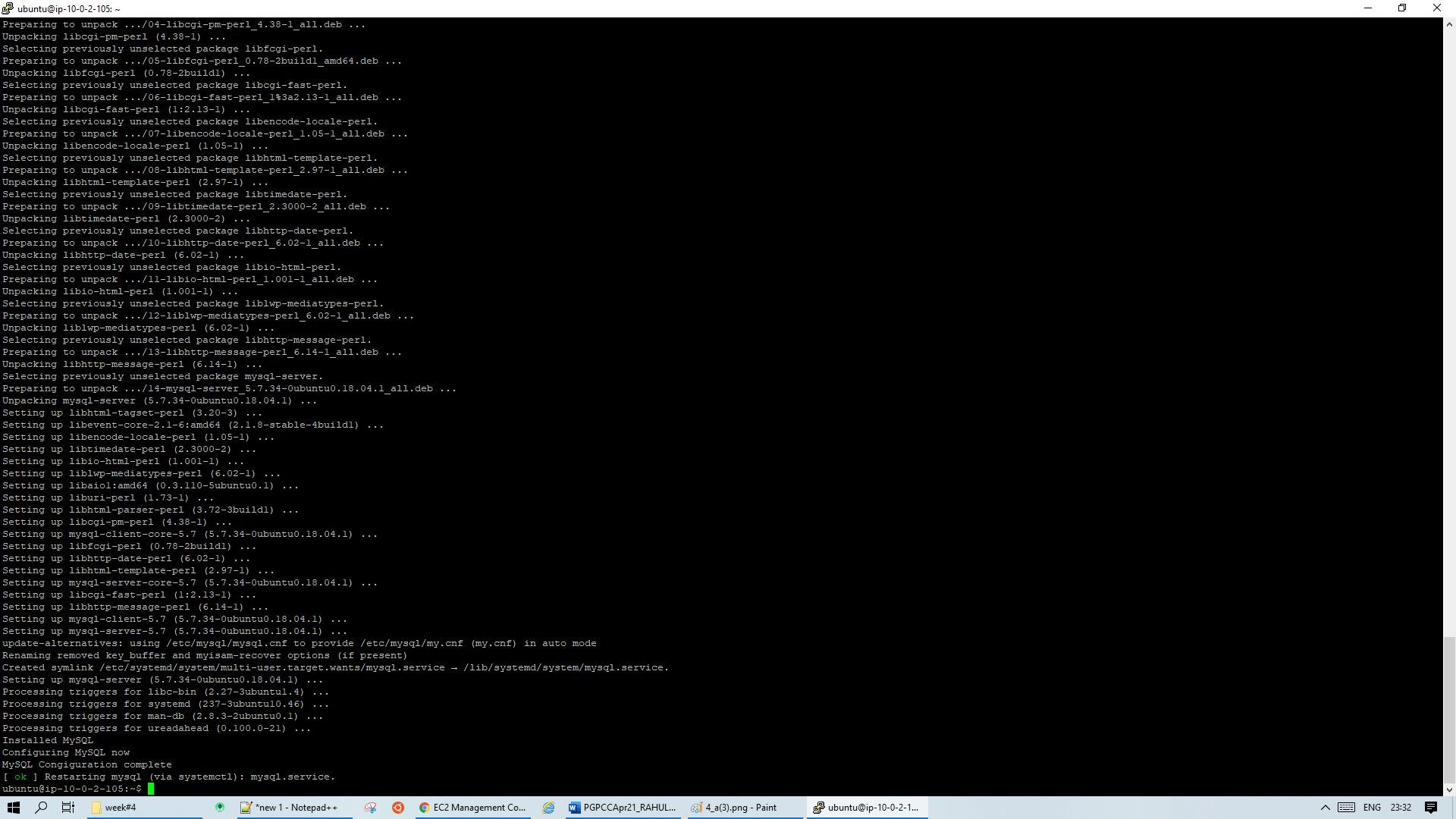
**<Insert screenshot a(1) here>**

****

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

****

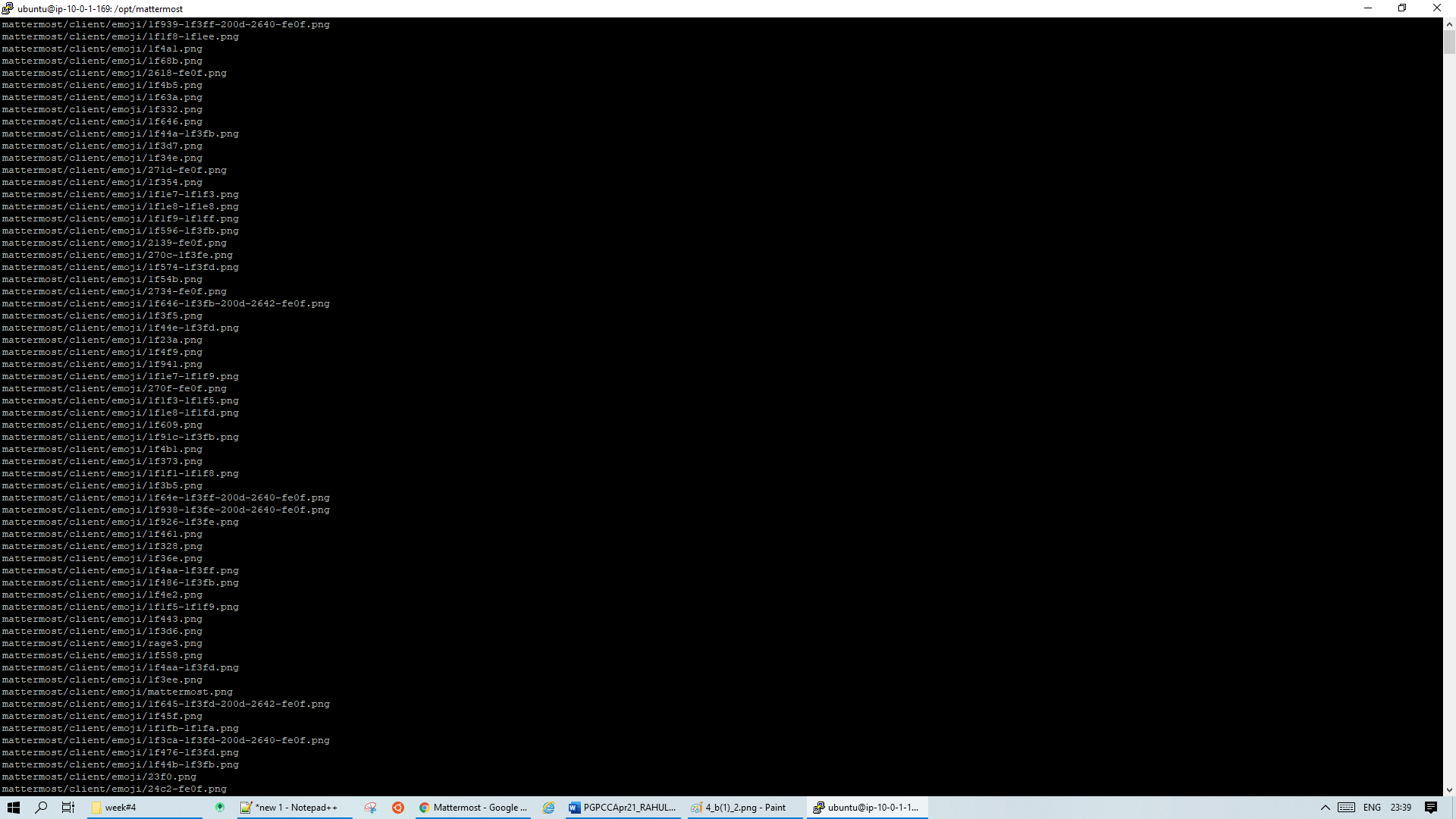


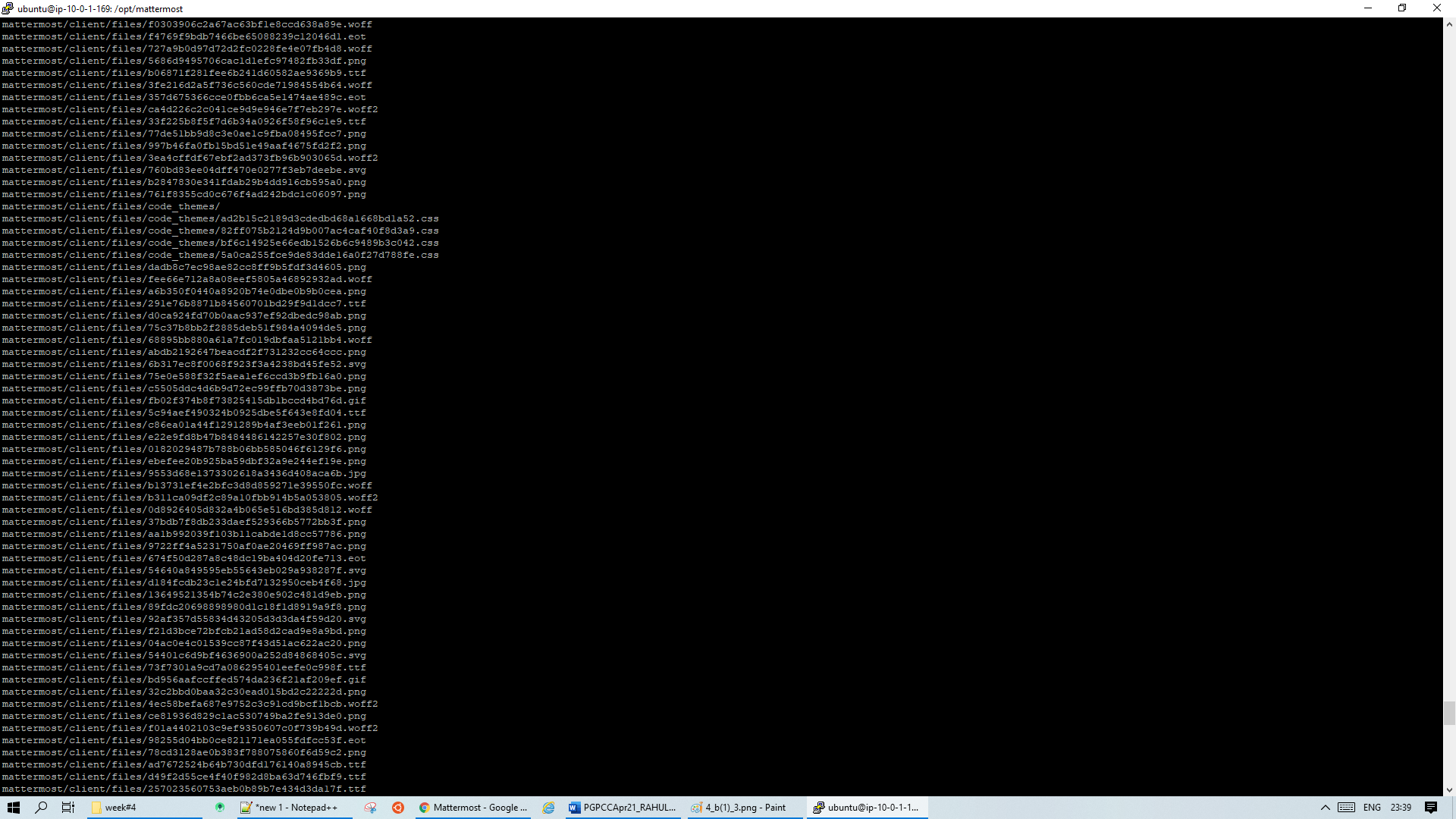


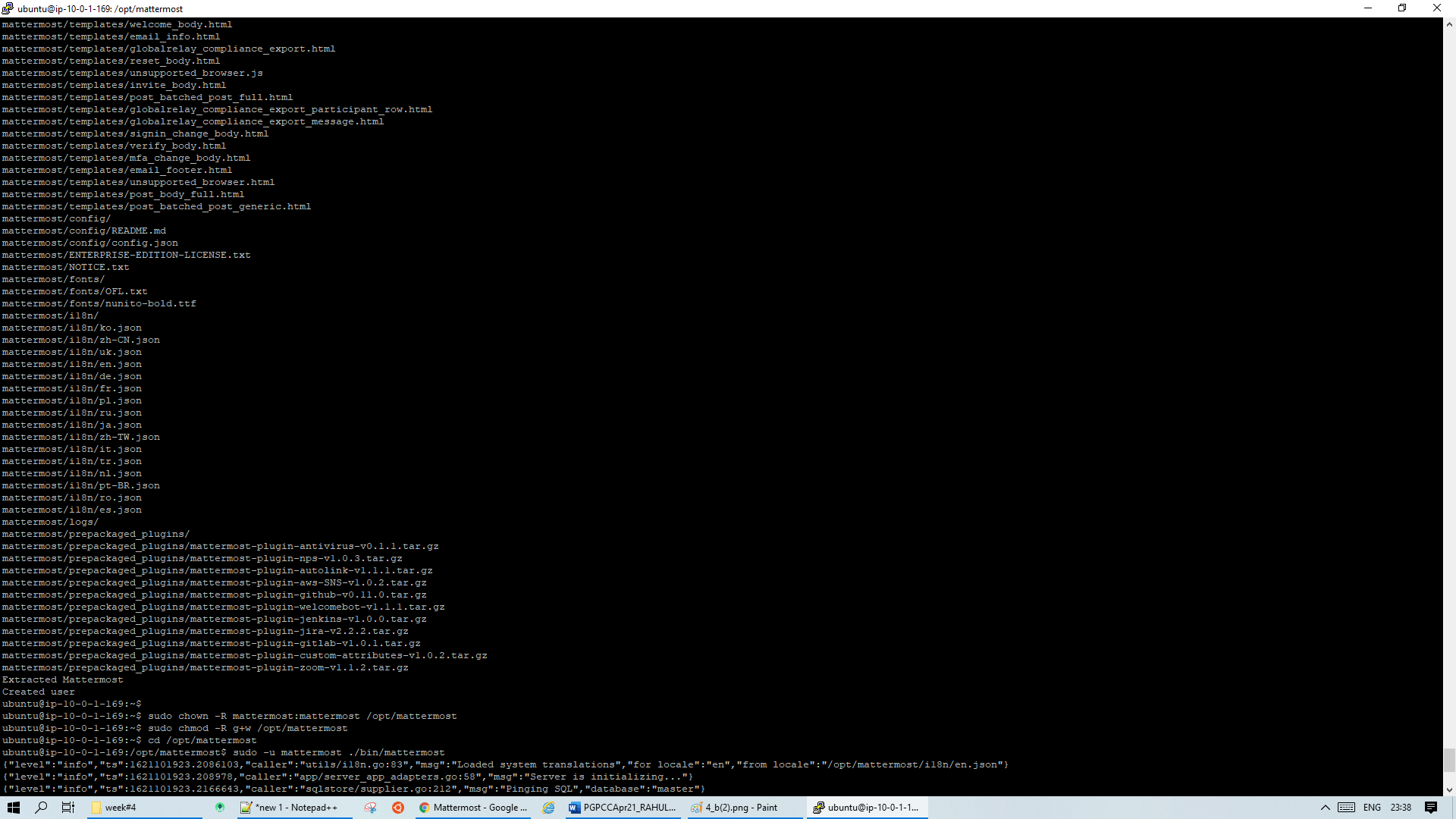
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step number | b |  |  |  |
| Step name | Installation and configuration of Mattermost |  |  |  |
| Instructions | 1) Enter the following commands after logging into the application server via SSH to install and configure Mattermost  *wget* [*https://storage.googleapis.com/skl-training/aws-codelabs/mattermost/mattermost\_install.sh*](https://storage.googleapis.com/skl-training/aws-codelabs/mattermost/mattermost_install.sh)  *chmod 700 mattermost\_install.sh*  *sudo ./mattermost\_install.sh <private IP of MySQL server>*  Example : sudo ./mattermost\_install.sh 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  <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>**

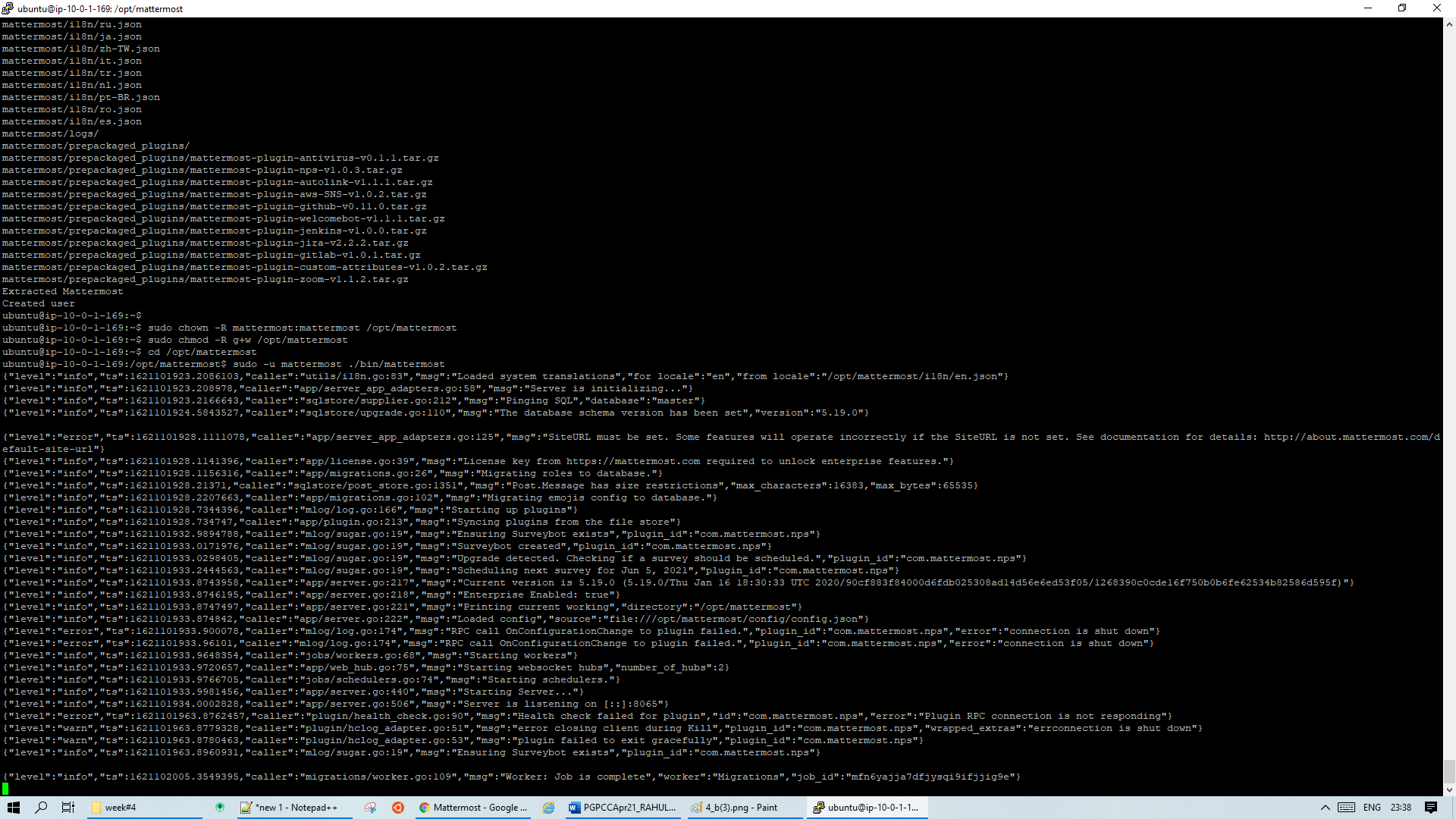
**Execution of script from start was not captured as too many logs were coming.**

****

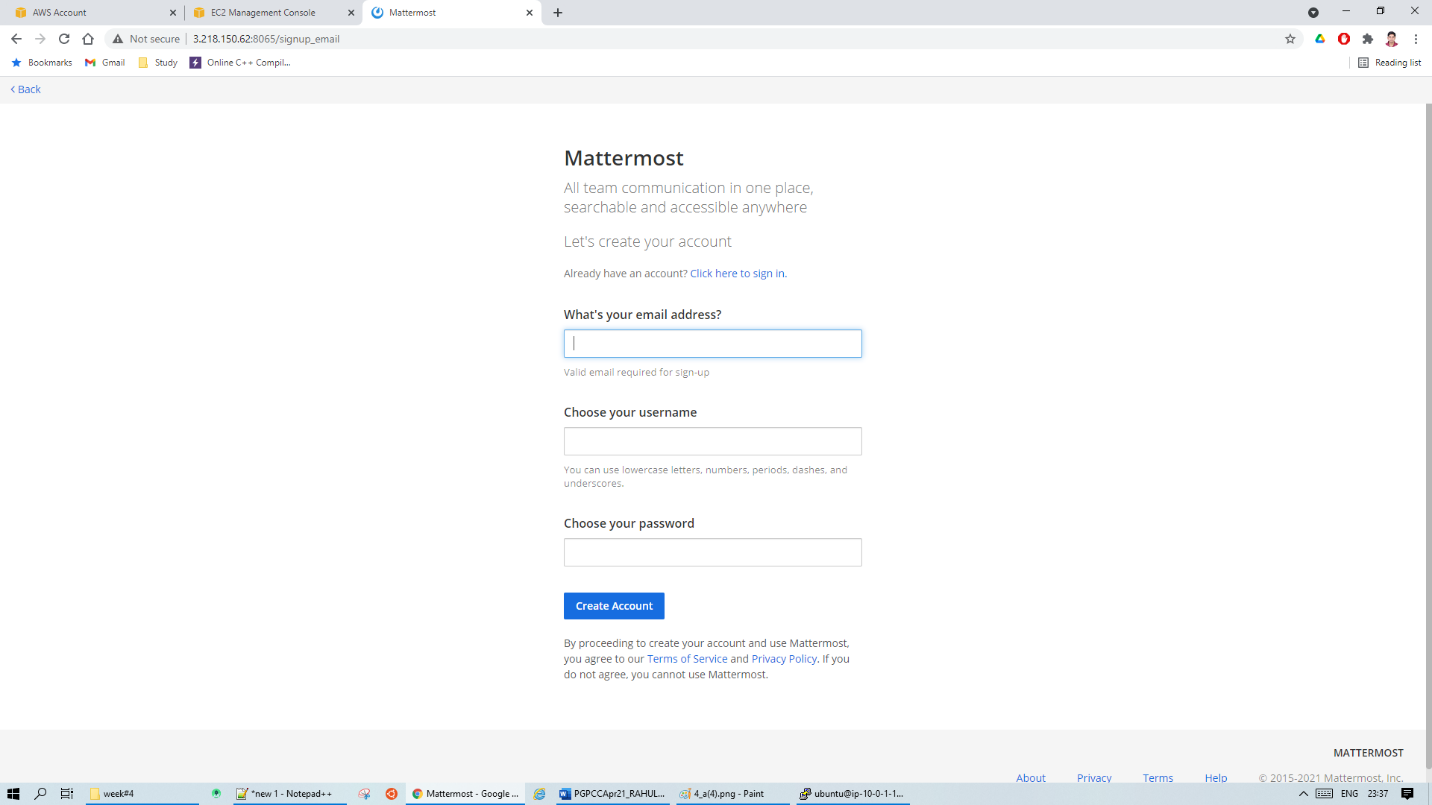
****

****

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

****

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

****

**Step 5: Answer the following questions**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Answer the following questions** | | | | **Marks** |
| Q1 | What is the default setting for DNS hostnames when a new VPC is created? | | | 1 |
|  | a) Enabled |  |  |  |
|  | b) Disabled |  |  |  |
|  | c) Can be set during VPC creation |  |  |  |
|  | d) Depends on the region used |  |  |  |
|  | Enter your answer here | b) Disabled |  |  |
|  |  |  |  |  |
| Q2 | What is the term used for the application server when we use it to log into the database server | | | 1 |
|  | a) Bastion Host |  |  |  |
|  | b) NAT Gateway |  |  |  |
|  | c) Tunnel Interface |  |  |  |
|  | d) SSH Gateway |  |  |  |
|  | Enter your answer here | a) Bastion Host |  |  |
|  |  |  |  |  |
| Q3 | The database server security group in this exercise has to keep port 3306 open. Which protocol uses this port to communicate? | | | 1 |
|  | a) HTTPS |  |  |  |
|  | b) RDP |  |  |  |
|  | c) TCP |  |  |  |
|  | d) SCP |  |  |  |
|  | Enter your answer here | c) TCP |  |  |
|  |  |  |  |  |
| Q4 | Which port is being used by Mattermost to communicate with the client application | | | 1 |
|  | a) 8080 |  |  |  |
|  | b) 80 |  |  |  |
|  | c) 443 |  |  |  |
|  | d) 8065 |  |  |  |
|  | Enter your answer here | d) 8065 |  |  |
|  |  |  |  |  |
| Q5 | Which of the following is a reason why we cannot set the CIDR block for the public subnet to 10.0.2.0/16, assuming the values for the other CIDR blocks are the same as mentioned in the instructions? | | | 1 |
|  | a) CIDR block overlaps with existing block |  |  |  |
|  | b) CIDR block is not a valid CIDR |  |  |  |
|  | c) CIDR block does not fall within the VPC |  |  |  |
|  | d) There is no reason, this is a perfectly valid CIDR |  |  |  |
|  | Enter your answer here | a) CIDR block overlaps with existing block |  |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q6 | In this exercise, you have been asked to create 3 EC2 instances - the application server, the database server and the NAT instance. Each of these instances have their own security groups with a set of ports to be kept open. One of those ports is entirely unnecessary for the given architecture to function. Which of the ports given in the option below is it? | | | 1 |
|  | a) Port 22 on the NAT instances |  |  |  |
|  | b) Port 3306 on the database server |  |  |  |
|  | c) Port 443 on the NAT instance |  |  |  |
|  | d) Port 22 on the application server |  |  |  |
|  | Enter your answer here | a) Port 22 on the NAT instances |  |  |
|  |  |  |  |  |
| Q7 | Describe the steps you would take to increase security of the servers you have deployed so that they are not reachable from external sources | | | 4 |
|  |  | | |  |
|  | * Use security groups. * Augment Security Groups with NACLs. * Implement Firewalls * Internet gateways as the single point of access between your VPC network’s resources and the public internet. * Service auditing is a way of knowing what services are running on a given system, which ports they are using for communication, and what protocols are accepted. This information can help you configure which services should be publicly accessible, firewall settings, and monitoring and alerting. * Disable Directory Indexes. * Using a VPN is, effectively, a way to map out a private network that only your servers can see. * Establishing a certificate authority (CA) and managing certificates for your servers allows each entity within your infrastructure to validate the other members’ identities and encrypt their traffic. * Isolating execution environments refers to any method in which individual components are run within their own dedicated space. | | |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Q8 | Describe the steps required to deploy the given application in an autoscaling environment | | | 5 |
|  |  |  |  |  |
|  | step#1  Create 1 EC2 instance using the 7 step workflow (use t2.micro instance type only)  a) Use the usual Amazon Linux AMI in AZ1  step#2  Create a LB with TG  a) Associate the EC2 instance to it  b) Ensure all AZ are selected when creating the TG  c) Once the instance turns "healthy" hit the LB to ensure the page is being served  step#3  Create a autoscaling group  a) Use the bootstrap script for the autoscale launch config, tag = "autoscale"  b) Min instance = 1, max = 2  c) Setup the cloud watch alarm (add 1 instance if cpu>80, remove when <30)  step#4  Manually terminate the manuualy created EC2 instance, wait for 2-3 mins and auto scaling kicks into picture and creates a new EC2 instance. | | |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  | **Max marks** | **15** |

|  |  |
| --- | --- |
| **Grades distribution** |  |
| MCQs | 6 (1 mark each) |
| Subjective questions | 9 marks (4+5) |
| Implementation screenshots | 10 marks (0.5 marks each) |
| Total | 25 marks |