SVKM'S NMIM'S Nilkamal School of Mathematics, Applied Statistics & Analytics
Master of Science (Data Science)

Practical-1 Infrastructure as a service using AWS.

Writeup:-

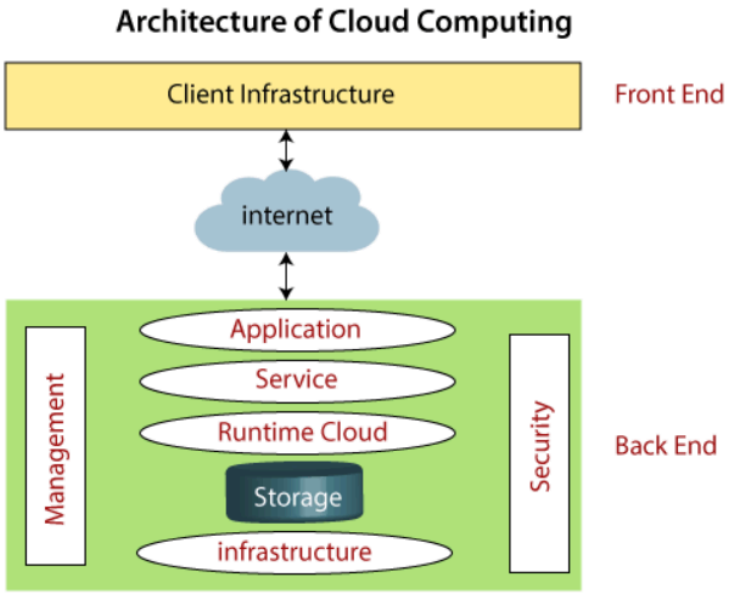
1. **Cloud Computing architecture**

Cloud computing technology is used by both small and large organizations to store the information in the cloud and access it from anywhere at any time using the internet connection.

Cloud computing architecture is a combination of service-oriented architecture and event-driven architecture.

Cloud computing architecture is divided into the following two parts -

1. **Front End**
2. **Back End**



Front End

The front end is used by the client. It contains client-side interfaces and applications that are required to access the cloud computing platforms. The front end includes

web servers (including Chrome, Firefox, internet explorer, etc.), thin & fat clients, tablets, and mobile devices.

Back End

The back end is used by the service provider. It manages all the resources that are required to provide cloud computing services. It includes a huge amount of data storage, security mechanism, virtual machines, deploying models, servers, traffic control mechanisms, etc

Components of Cloud Computing Architecture

There are the following components of cloud computing architecture -

1. Client Infrastructure

Client Infrastructure is a Front end component. It provides GUI (Graphical User Interface) to interact with the cloud.

2. Application

The application may be any software or platform that a client wants to access.

3. Service

A Cloud Services manages that which type of service you access according to the client's requirement.

Cloud computing offers the following three type of services:

i. Software as a Service (SaaS) – It is also known as **cloud application services**. Mostly, SaaS applications run directly through the web browser means we do not require to download and install these applications. Some important example of SaaS is given below –

Example: Google Apps, Salesforce Dropbox, Slack, Hubspot, Cisco WebEx.

ii. Platform as a Service (PaaS) – It is also known as **cloud platform services**. It is quite similar to SaaS, but the difference is that PaaS provides a platform for software creation, but using SaaS, we can access software over the internet without the need of any platform.

Example: Windows Azure, Force.com, Magento Commerce Cloud, OpenShift.

iii. Infrastructure as a Service (IaaS) – It is also known as **cloud infrastructure services**. It is responsible for managing applications data, middleware, and runtime environments.

Example: Amazon Web Services (AWS) EC2, Google Compute Engine (GCE), Cisco Metapod.

4. Runtime Cloud

Runtime Cloud provides the **execution and runtime environment** to the virtual machines.

5. Storage

Storage is one of the most important components of cloud computing. It provides a huge amount of storage capacity in the cloud to store and manage data.

6. Infrastructure

It provides services on the **host level**, **application level**, and **network level**. Cloud infrastructure includes hardware and software components such as servers, storage, network devices, virtualization software, and other storage resources that are needed to support the cloud computing model.

7. Management

Management is used to manage components such as application, service, runtime cloud, storage, infrastructure, and other security issues in the backend and establish coordination between them.

8. Security

Security is an in-built back end component of cloud computing. It implements a security mechanism in the back end.

9. Internet

The Internet is medium through which front end and back end can interact and communicate with each other.

2. IAAS

Infrastructure-as-a-Service (IaaS) is a form of cloud computing service that offers **compute**, **storage** and **networking resources** on-demand, usually on a pay-as-you-go basis. Businesses can purchase resources on-demand and as-needed instead of having to buy the hardware outright.
While IaaS gives you virtualized resources such as servers, disks, networks, and IP addresses, you are still responsible for administering the operating system, data, applications, middleware and runtimes. A dashboard or an API gives you complete control over the entire infrastructure.
IaaS gives you the flexibility to purchase only the computing you need and scales them up or down as needed. If you are looking to migrate an application as-is from an on-premises data center to the cloud, choose the IaaS model. You will be able to proceed with the migration with minimum changes.
Because of its speed of deployment, IaaS is a quick and flexible way to build up and take down development and testing environments.
Examples of IaaS include Rackspace, Amazon Web Services (AWS) Elastic Compute Cloud (EC2), Microsoft Azure, Google Compute Engine (GCE) and Joyent.

Advantage:-

IaaS is advantageous when scalability and quick provisioning are key. Cloud Service Providers can provide a variety of hardware configurations with pre-configured operating systems such as Linux or Windows. You can also use rapid provisioning patterns with Infrastructure as Code (IaC) to create packages of IT resources that can be bundled and deployed into ready-made environments.

Disadvantage:-

Since the infrastructure is under the control of the service provider, outages in the service provider controlled infrastructure can affect the customer infrastructure.Troubleshooting is more difficult because IaaS customers do not have complete visibility to the cloud service provider infrastructure.If peak usage is high, monthly costs may be much higher than expected.

3. AWS

AWS (Amazon Web Services) is a comprehensive, evolving cloud computing platform provided by Amazon. It includes a mixture of infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS) and packaged software-as-a-service (SaaS) offerings. AWS offers tools such as compute power, database storage and content delivery services.
AWS offers many different tools and products for enterprises and software developers in 245 countries and territories. Government agencies, education institutions, nonprofits and private organizations use AWS services.
Amazon web service is an online platform that provides scalable and cost-effective cloud computing solutions.

4. AWS services

In the rapid revolution of Cloud Computing, AWS facilitates with wide variety of services respect to the fields and needs. The following are the top AWS services that are in wide usage:

Amazon EC2(Elastic Compute Cloud) : It provides the Scalable computing power via cloud allowing the users to run applications and manage the workloads over their remotely.
Amazon S3 (Simple Storage Service): It offers scalable object Storage as a Service with high durability for storing and retrieving any amount of data.
AWS Lambda: It is a service in Serverless Architecture with Function as a Service facilitating serverless computing i.e., running the code on response to the events, the background environment management of servers is handled by aws automatically. It helps the developers to completely focus on the logic of code build.
Amazon RDS (Relational Database Service): This is an aws service that simplifies the management of database providing high available relational databases in the cloud.
Amazon VPC (Virtual Private Cloud): It enables the users to create isolated networks with option of public and private expose within the AWS cloud, providing safe and adaptable configurations of their resources.

5. EC2

Amazon Elastic Compute Cloud (Amazon EC2) provides on-demand, scalable computing capacity in the Amazon Web Services (AWS) Cloud. Using Amazon EC2 reduces hardware costs so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage. You can add capacity (scale up) to handle compute-heavy tasks, such as monthly or yearly processes, or spikes in website traffic. When usage decreases, you can reduce capacity (scale down) again.

Features of Amazon EC2

- **Instances**

  Virtual servers.

- **Amazon Machine Images (AMIs)**

  Preconfigured templates for your instances that package the components you need for your server (including the operating system and additional software).

- **Instance types**

  Various configurations of CPU, memory, storage, networking capacity, and graphics hardware for your instances.

- **Key pairs**

  Secure login information for your instances. AWS stores the public key and you store the private key in a secure place.

- **Instance store volumes**

  Storage volumes for temporary data that is deleted when you stop, hibernate, or terminate your instance.

- **Amazon EBS volumes**

  Persistent storage volumes for your data using Amazon Elastic Block Store (Amazon EBS).

- **Regions, Availability Zones, Local Zones, AWS Outposts, and Wavelength Zones**

  Multiple physical locations for your resources, such as instances and Amazon EBS volumes.

- **Security groups**

A virtual firewall that allows you to specify the protocols, ports, and source IP ranges that can reach your instances, and the destination IP ranges to which your instances can connect.

- **Elastic IP addresses**

  Static IPv4 addresses for dynamic cloud computing.

- **Tags**

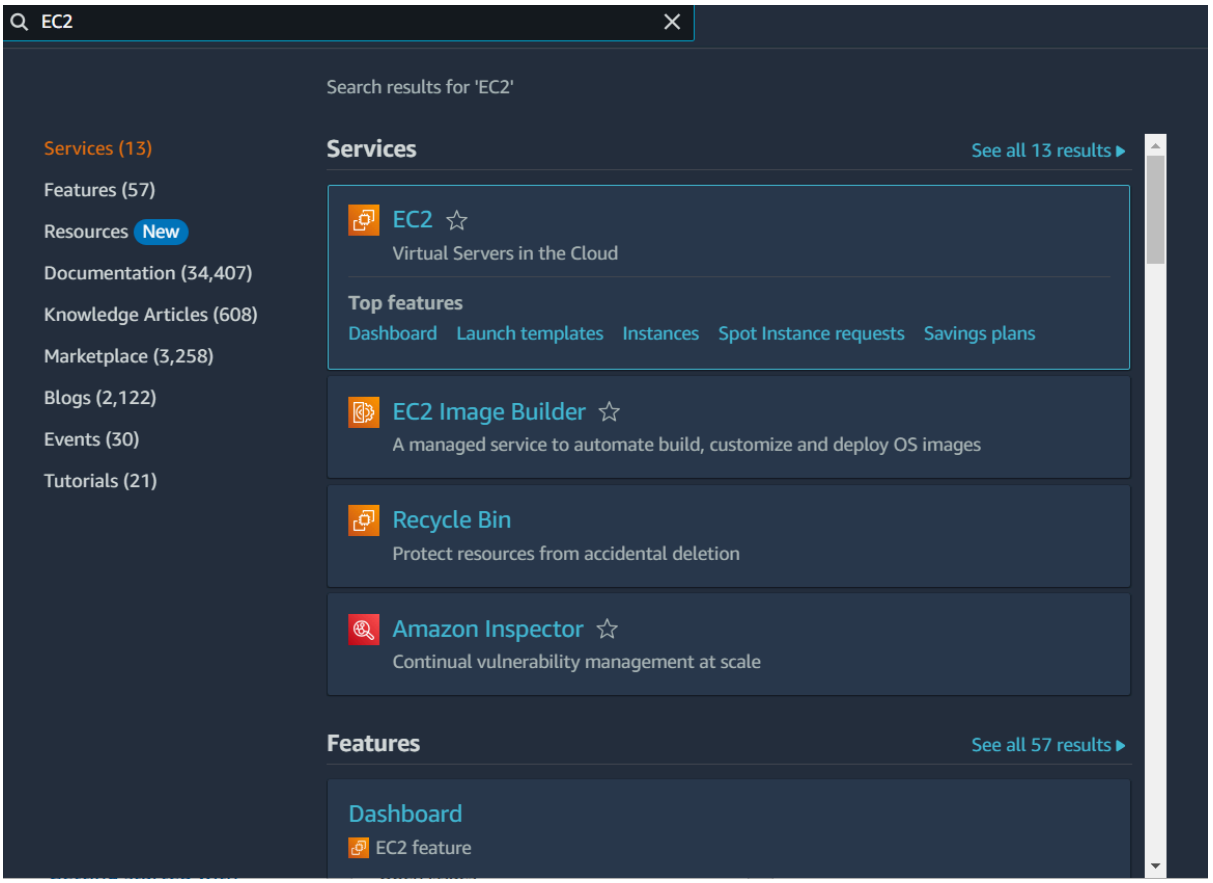
  Metadata that you can create and assign to your Amazon EC2 resources.

- **Virtual private clouds (VPCs)**

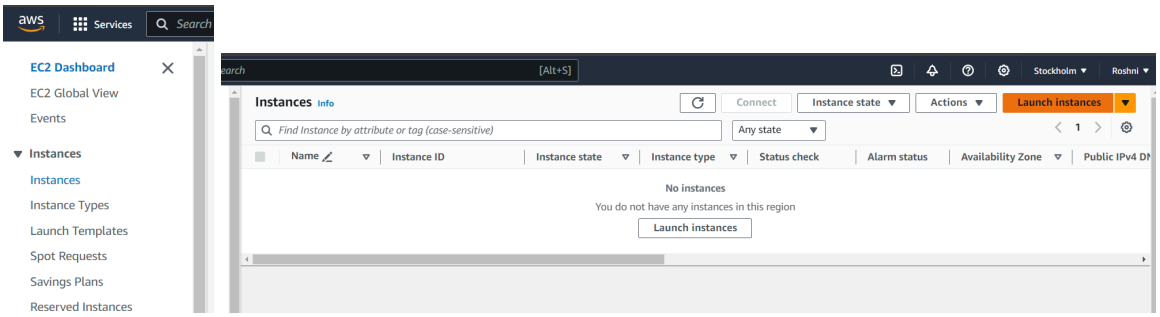
  Virtual networks you can create that are logically isolated from the rest of the AWS Cloud. You can optionally connect these virtual networks to your own network.

Implement the windows machine using AWS ec2.

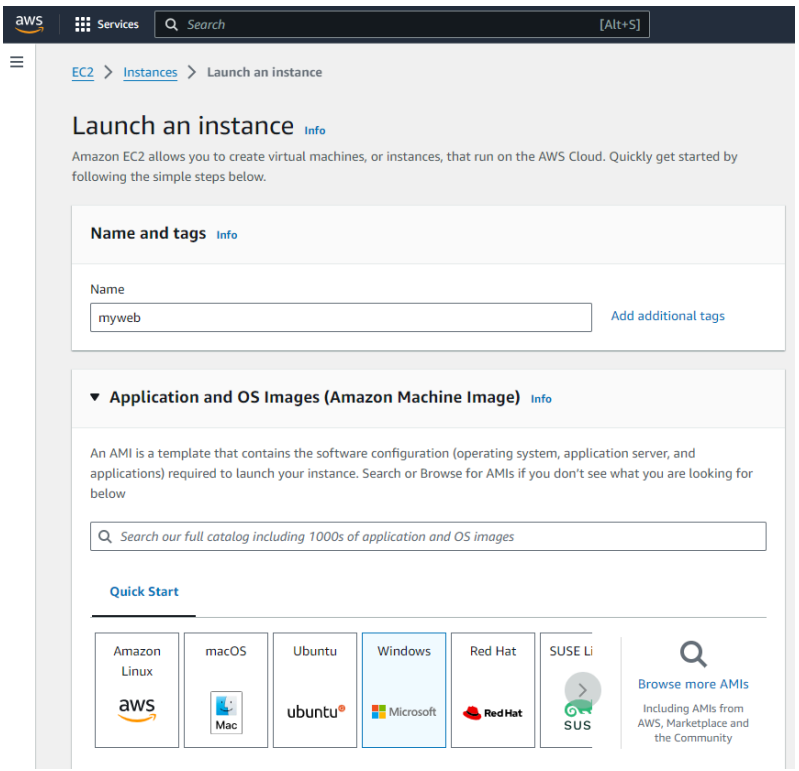
Step 1:-Login to https://aws.amazon.com/ and select EC2



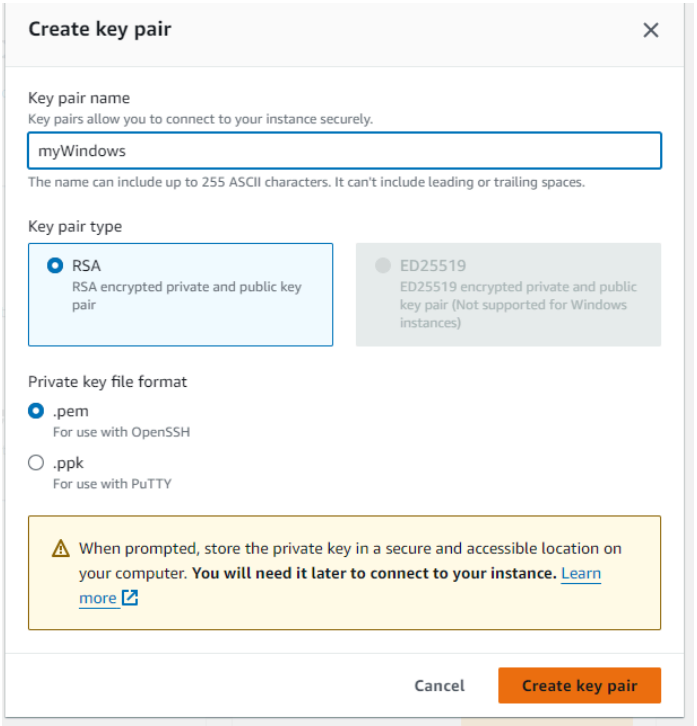
Step 2:- Select Instances from the left corner of the dashboard under EC2 and click on Launch instance



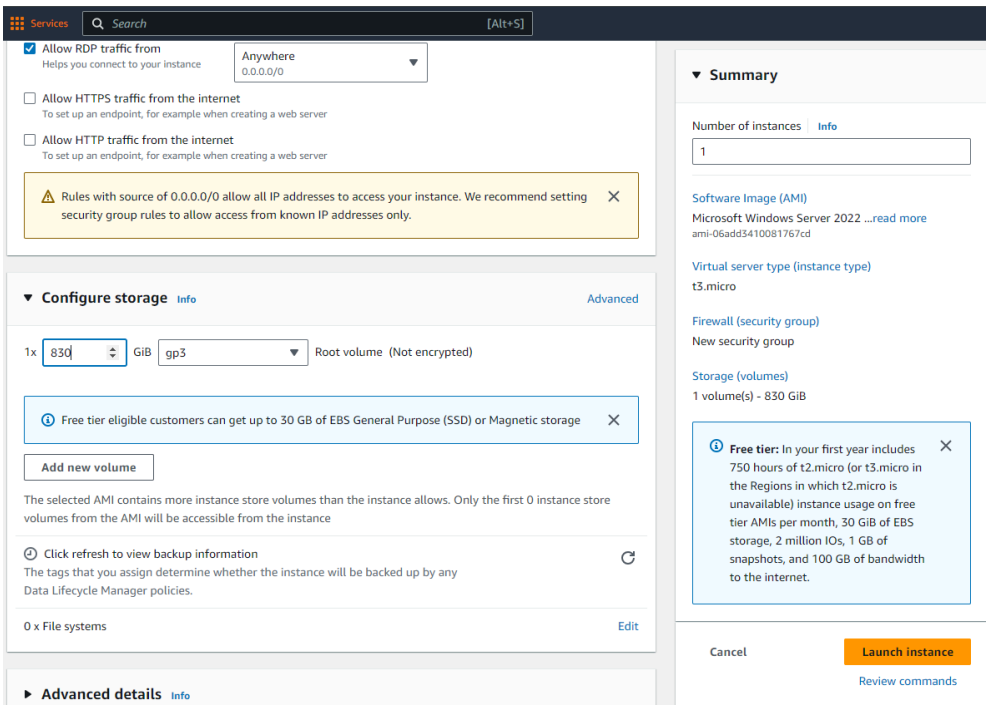
Step 3:- Provide the name of the instance and select Windows under Application and OS



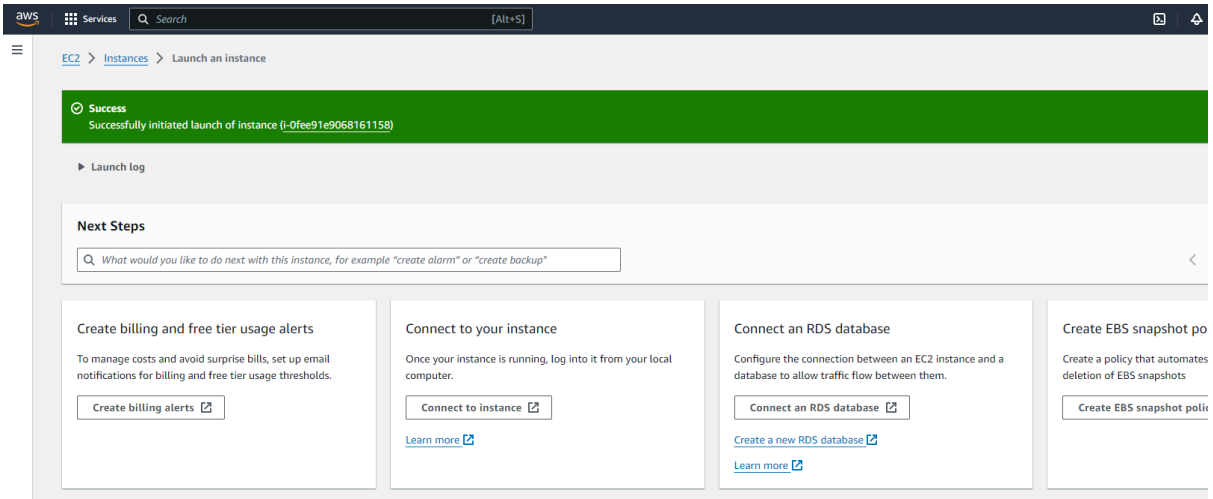
Step 4:- For key pair -> Create a new key pair ->Provide any name to the key pair name -> select type as RSA ->Private key file format as (.pem) and then click on create key pair.



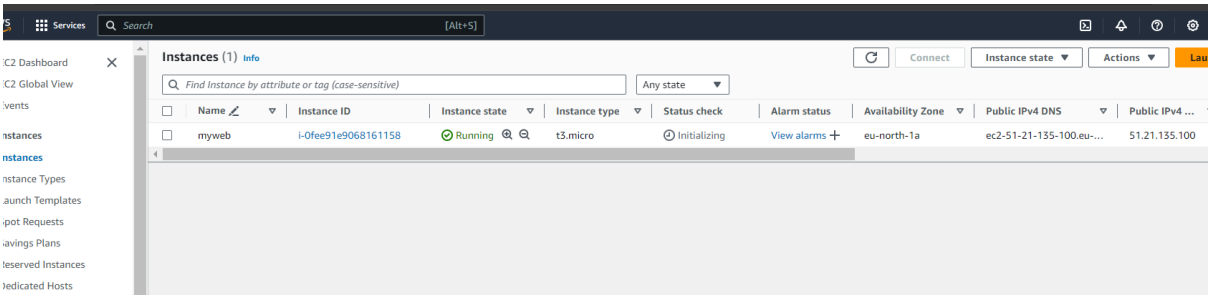
Step 5:-Click on Launch Instance



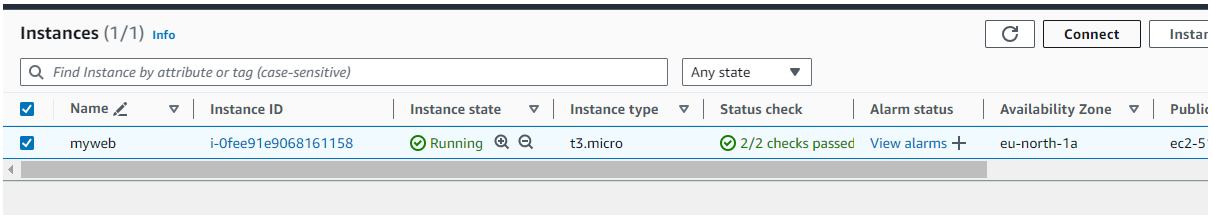
Step 6:- Your Instance will be launched successfully.



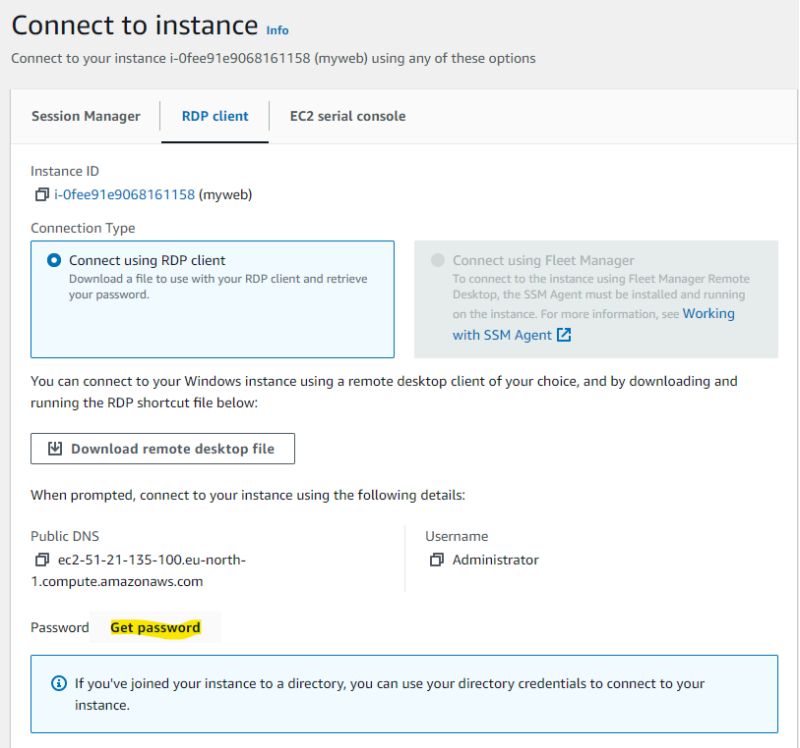
Step 7:- Go to Instances -> Refresh the page to see your created Instance



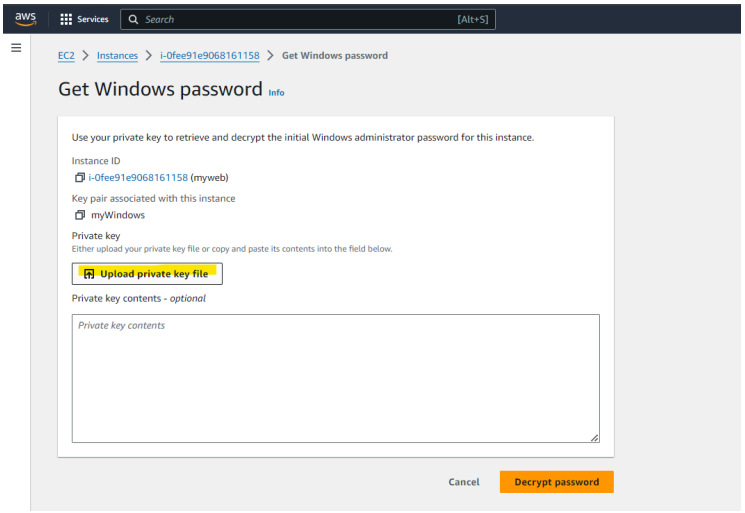
Step 8:-Click on your created Instance and than select Connect option at the top.

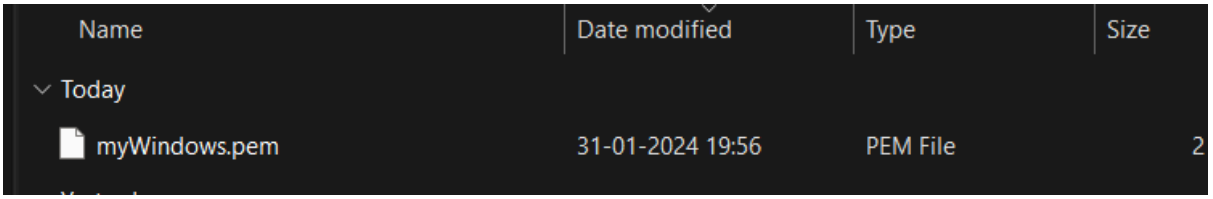


Step 9:- Select RDP Client and click on Get password

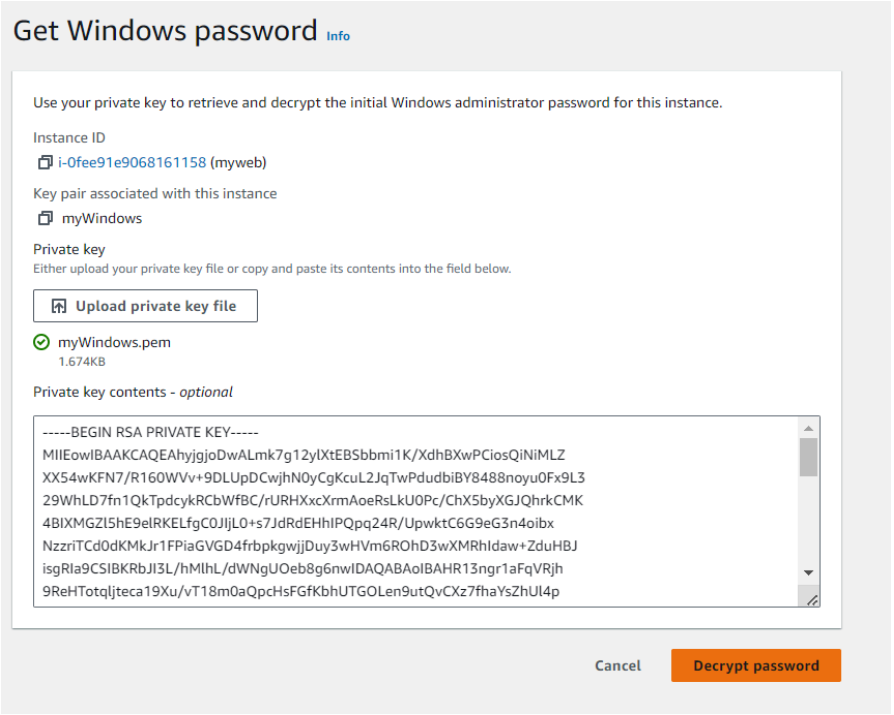


Step 10:-Click on Upload private key file (the key pair(.pem) file which got downloaded while creating the key pair for your Instance.)

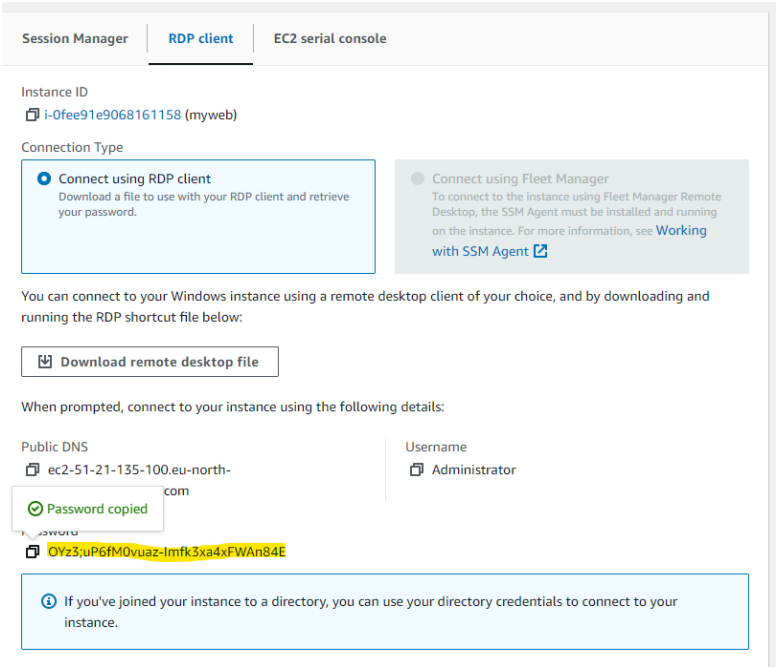




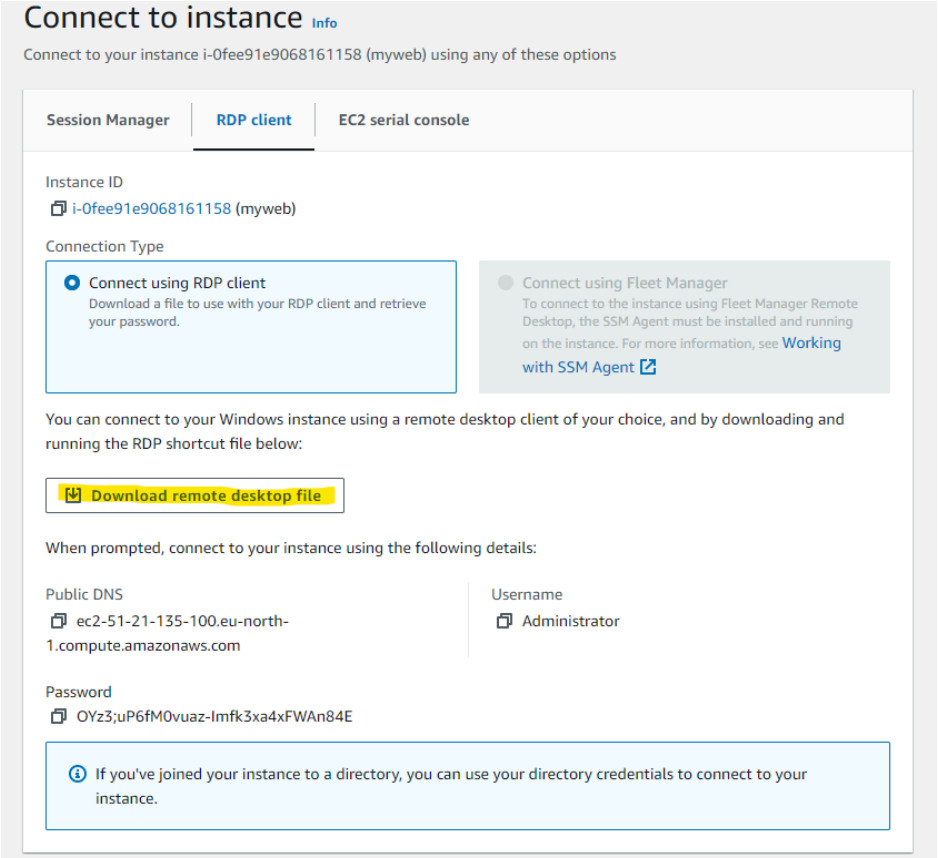
Step 11:- After uploading private key file click on Decrypt password



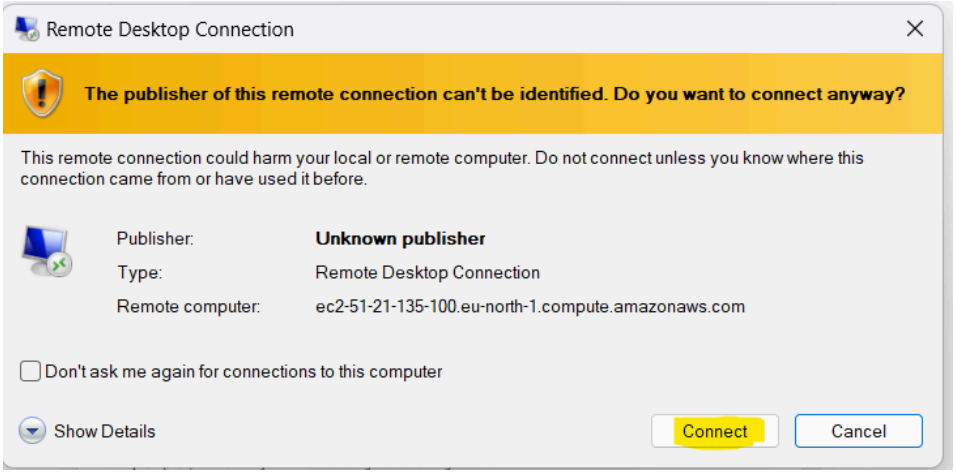
Step 12:- Save the password



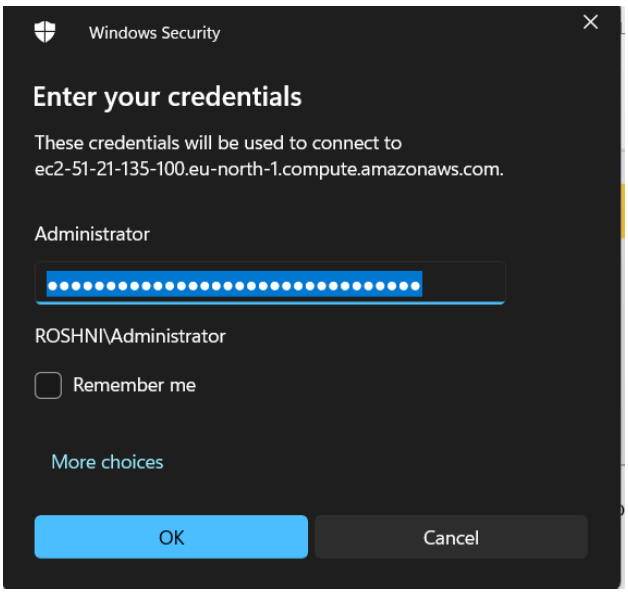
Step 13:- Click on Download remote desktop file



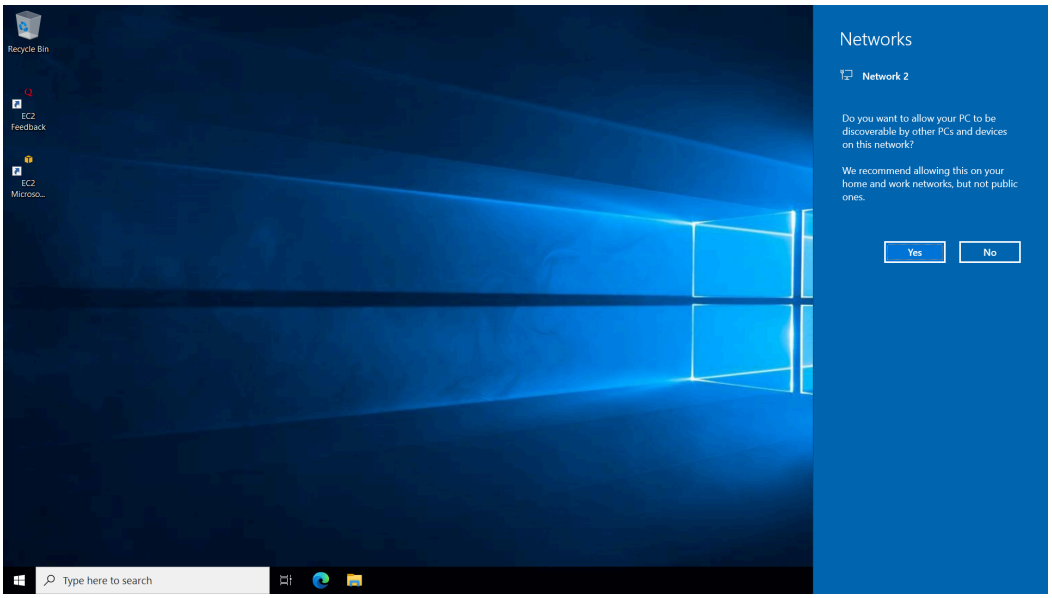
Step 14:- Open the remote desktop file and click on connect.



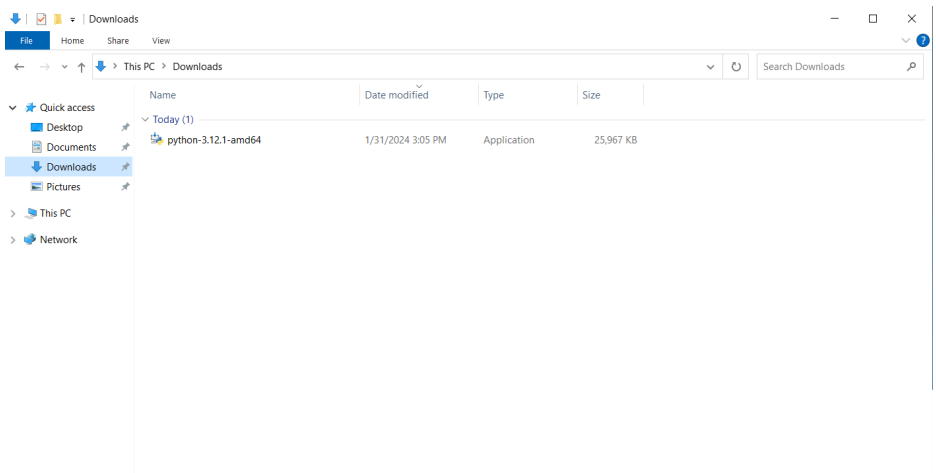
Step 15:- Provide the password which you had saved after decrypting password and click ok



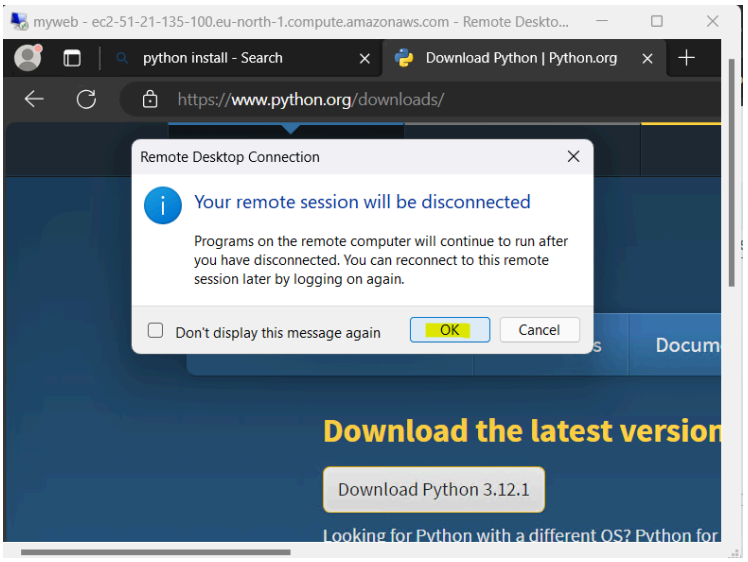
Step 16:- Your Windows Desktop Instance will get launched.

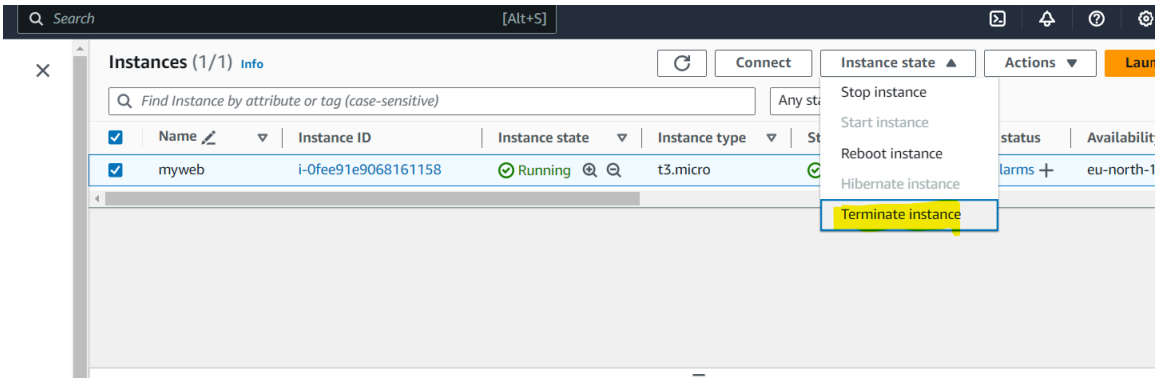


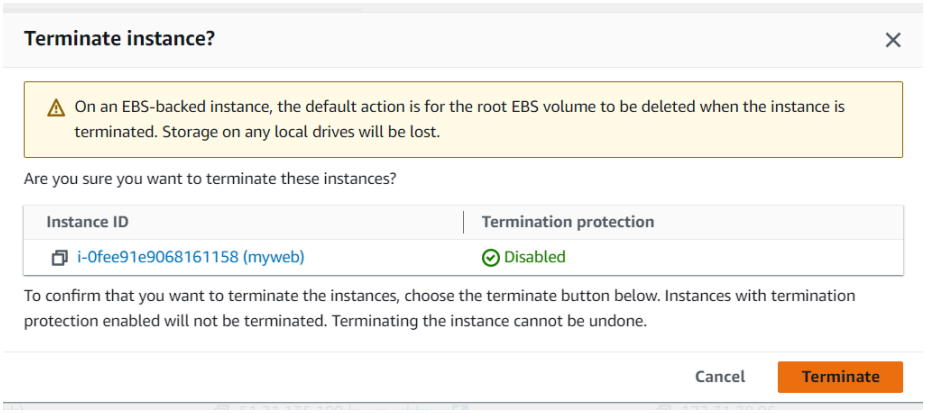
Step 17:- You and install any software and explore your Windows instance.

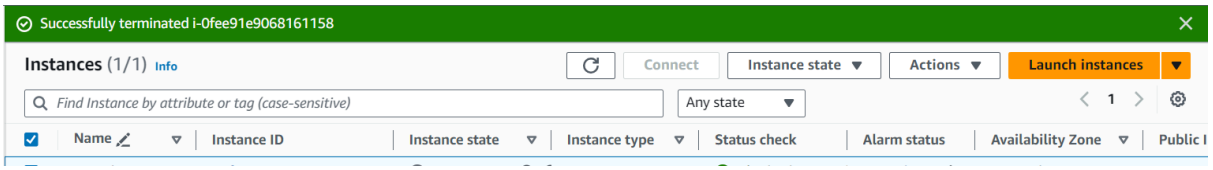


Step 18: Close your RDP and Go to Instance and terminate the Instance



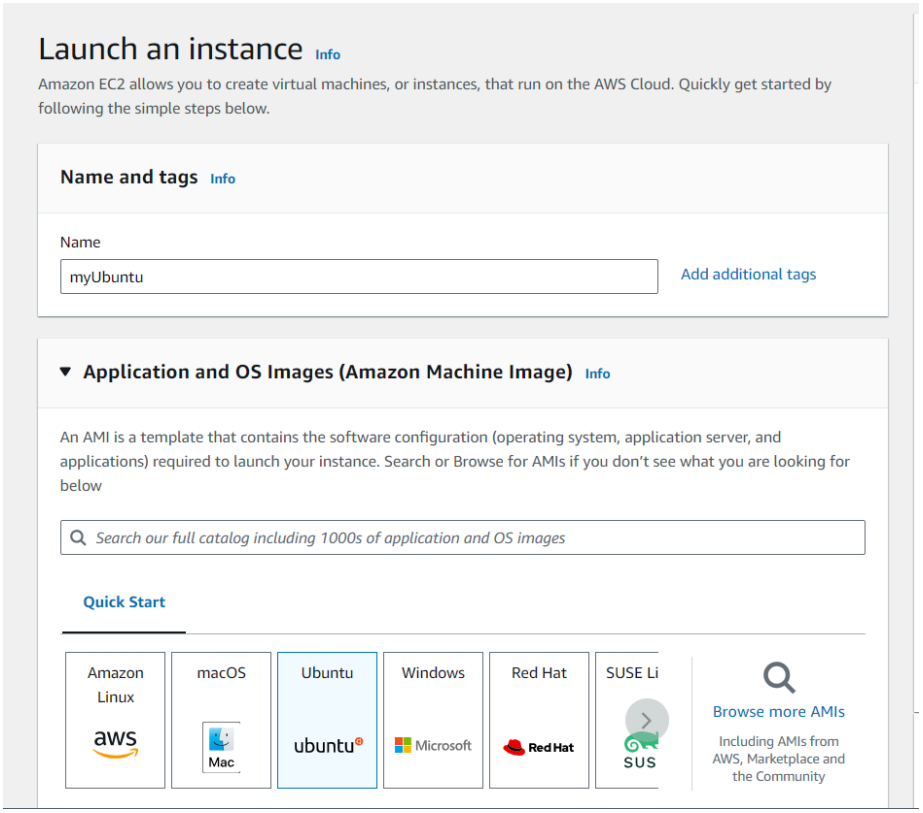



Terminate instance?
On an EBS-backed instance, the default action is for the root EBS volume to be deleted when the instance is terminated. Storage on any local drives will be lost.
Are you sure you want to terminate these instances?
Instance ID
Termination protection
i-0fee91e9068161158 (myweb)
Disabled
To confirm that you want to terminate the instances, choose the terminate button below. Instances with termination protection enabled will not be terminated. Terminating the instance cannot be undone.
Cancel
Terminate

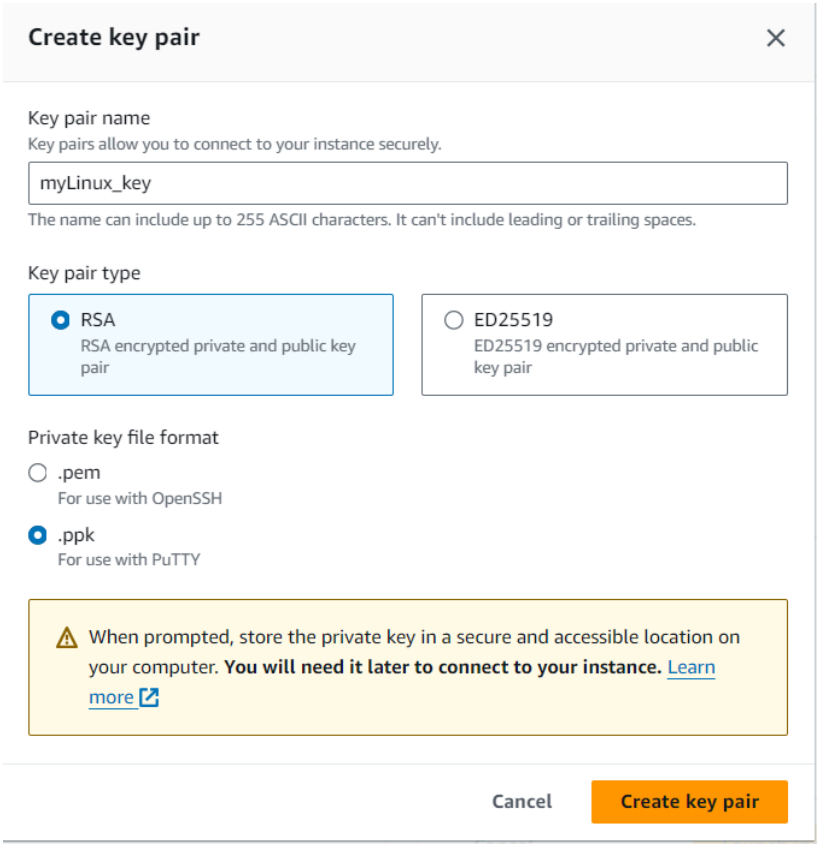

Successfully terminated i-0fee91e9068161158
Instances (1/1) Info
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
Any state
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone

Implement Ubuntu machine using AWS ec2 and execute the Linux commands.

Step 1:-Launch a new Instance for Linux and select Application and OS Provide name and then click on Ubuntu

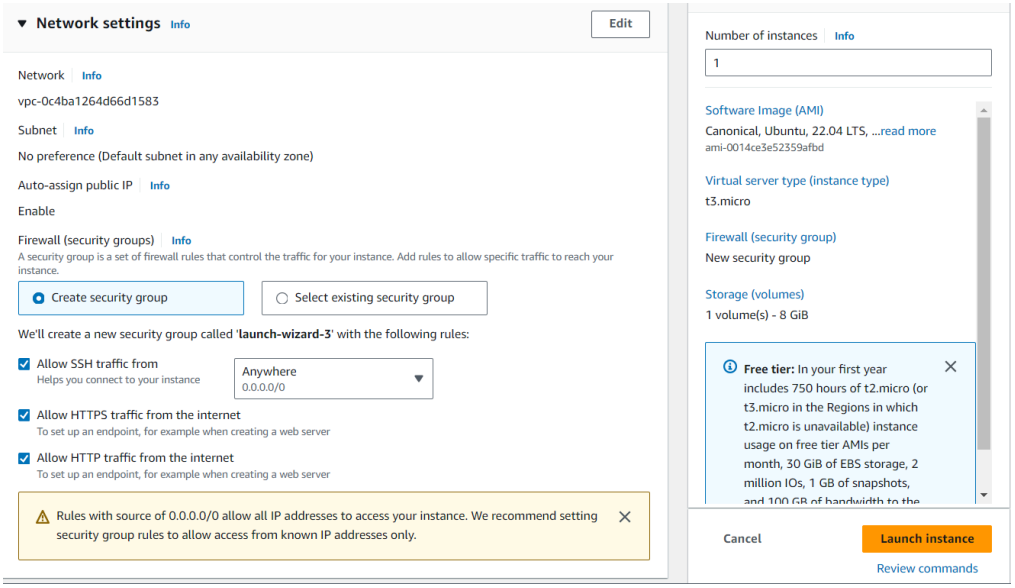


Step 2:-Create a new key pair -> Select the file format as (.ppk)



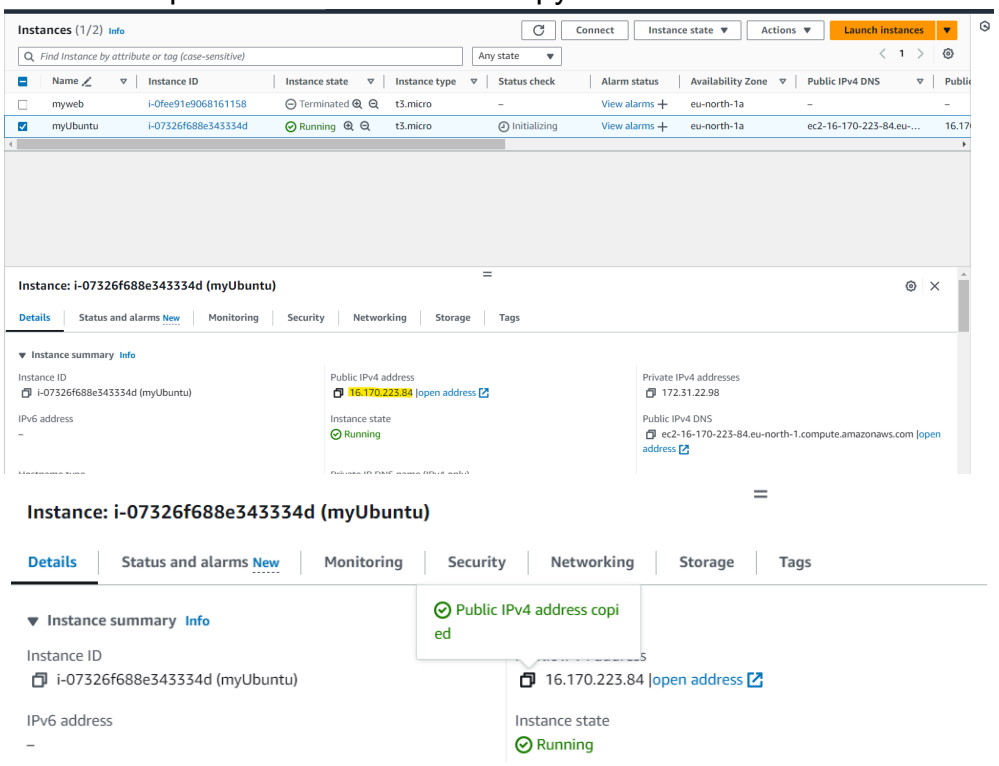
Step 3:-Download the PuTTY file from Google and select the `putty.exe` (the SSH and Telnet client itself) according to your system bits.

Step 4:- Allow all the (Network Settings) traffic under the Linux instance and Launch it

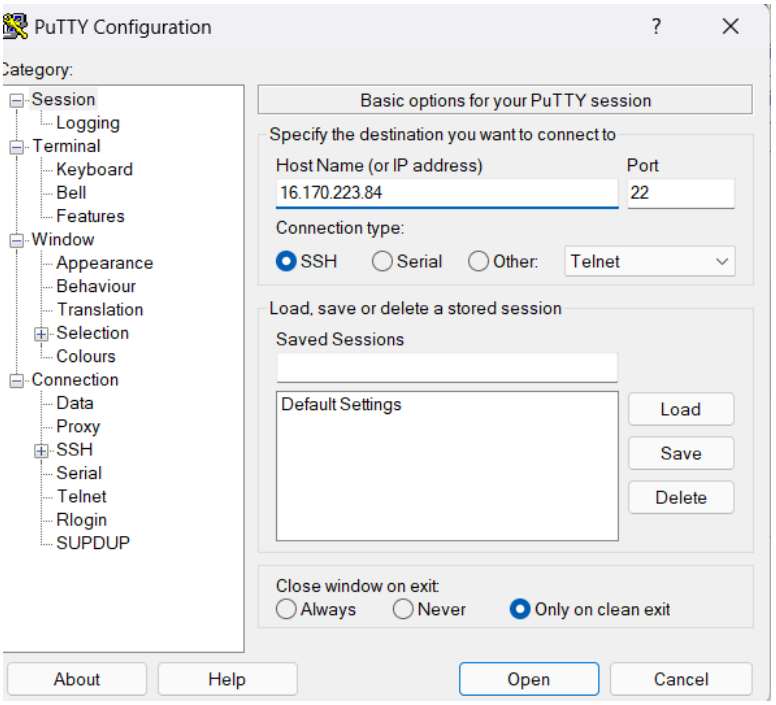


Your Instance is launched successfully.

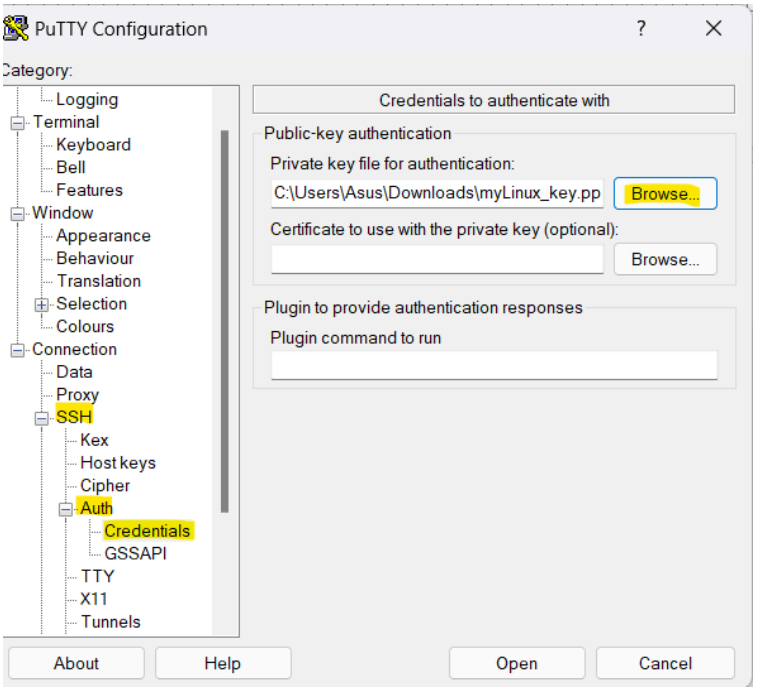
Step 5:- Select the particular Instance and copy the Public IPV4 address



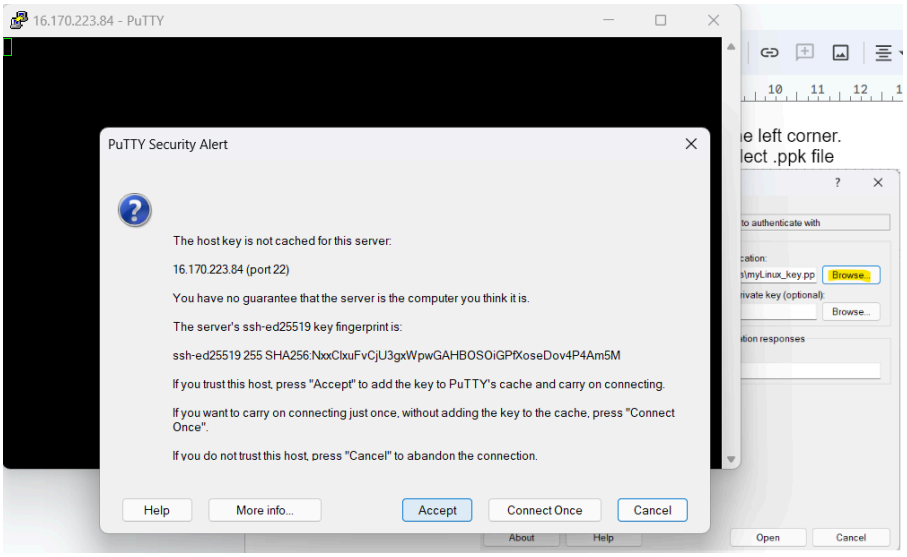
Step 6:- Open the PuTTY application and paste the IP address.



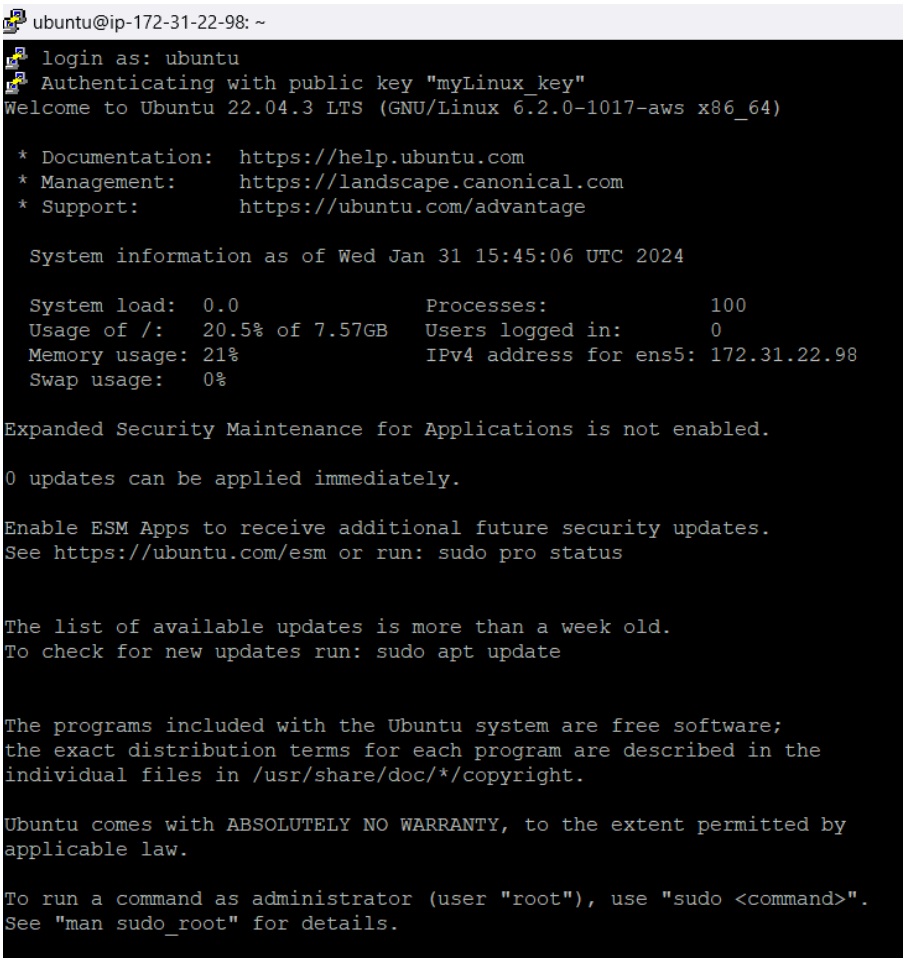
Step 7:- Under the PuTTY Category Section at the left corner.
Go to SSH ->Auth->Credentials->Browse and select .ppk file



Step 8:- Click on Open and Accept.



Step 9:- Provide ‘ubuntu’ name while login to PuTTy



Step 10:-Run the following commands.

1. Disk information in human readable format

```
ubuntu@ip-172-31-22-98:~$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
/dev/root         7941576 1654424   6270768  21% /
tmpfs              473556       0    473556   0% /dev/shm
tmpfs              189424     836    188588   1% /run
tmpfs                5120       0      5120   0% /run/lock
/dev/nvme0n1p15    106858    6186    100673   6% /boot/efi
tmpfs               94708       4     94704   1% /run/user/1000
ubuntu@ip-172-31-22-98:~$ df -H
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       8.2G  1.7G  6.5G  21% /
tmpfs           485M     0  485M   0% /dev/shm
tmpfs           194M  857k  194M   1% /run
tmpfs           5.3M     0  5.3M   0% /run/lock
/dev/nvme0n1p15 110M  6.4M  104M   6% /boot/efi
tmpfs            97M  4.1k   97M   1% /run/user/1000
```

2. Create a folder with your name

```
ubuntu@ip-172-31-22-98:~$ mkdir roshni
ubuntu@ip-172-31-22-98:~$ ls
roshni
```

3. Create a file with your cityname and add your address in it

```
ubuntu@ip-172-31-22-98:~$ nano mumbai.txt
```

ubuntu@ip-172-31-22-98: ~

```
  GNU nano 6.2
Kohinoor Cit, Opp.Holy Cross School, Kurla West, Mumbai-400070.
```

4. Display the created file

```
ubuntu@ip-172-31-22-98:~$ cat mumbai.txt
Kohinoor Cit, Opp.Holy Cross School, Kurla West, Mumbai-400070.
ubuntu@ip-172-31-22-98:~$
```

5. Copy the contents of the created file in other file and print it

```
ubuntu@ip-172-31-22-98:~$ cat mumbaicity.txt
Kohinoor Cit, Opp.Holy Cross School, Kurla West, Mumbai-400070.
```

6. Install firefox/python 3

```
ubuntu@ip-172-31-22-98:~$ python3
Python 3.10.12 (main, Nov 20 2023, 15:14:05) [GCC 11.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hii \n My Name is Roshni")
Hii
 My Name is Roshni
```

Step 11:-Terminate the Instance