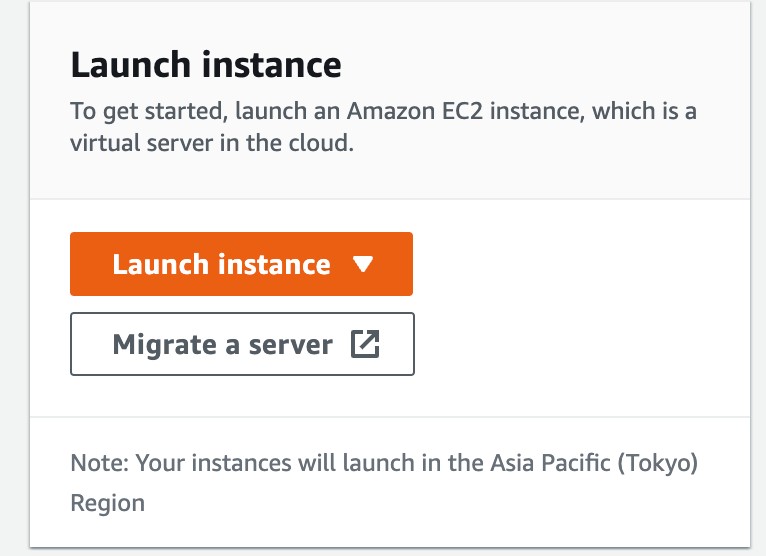
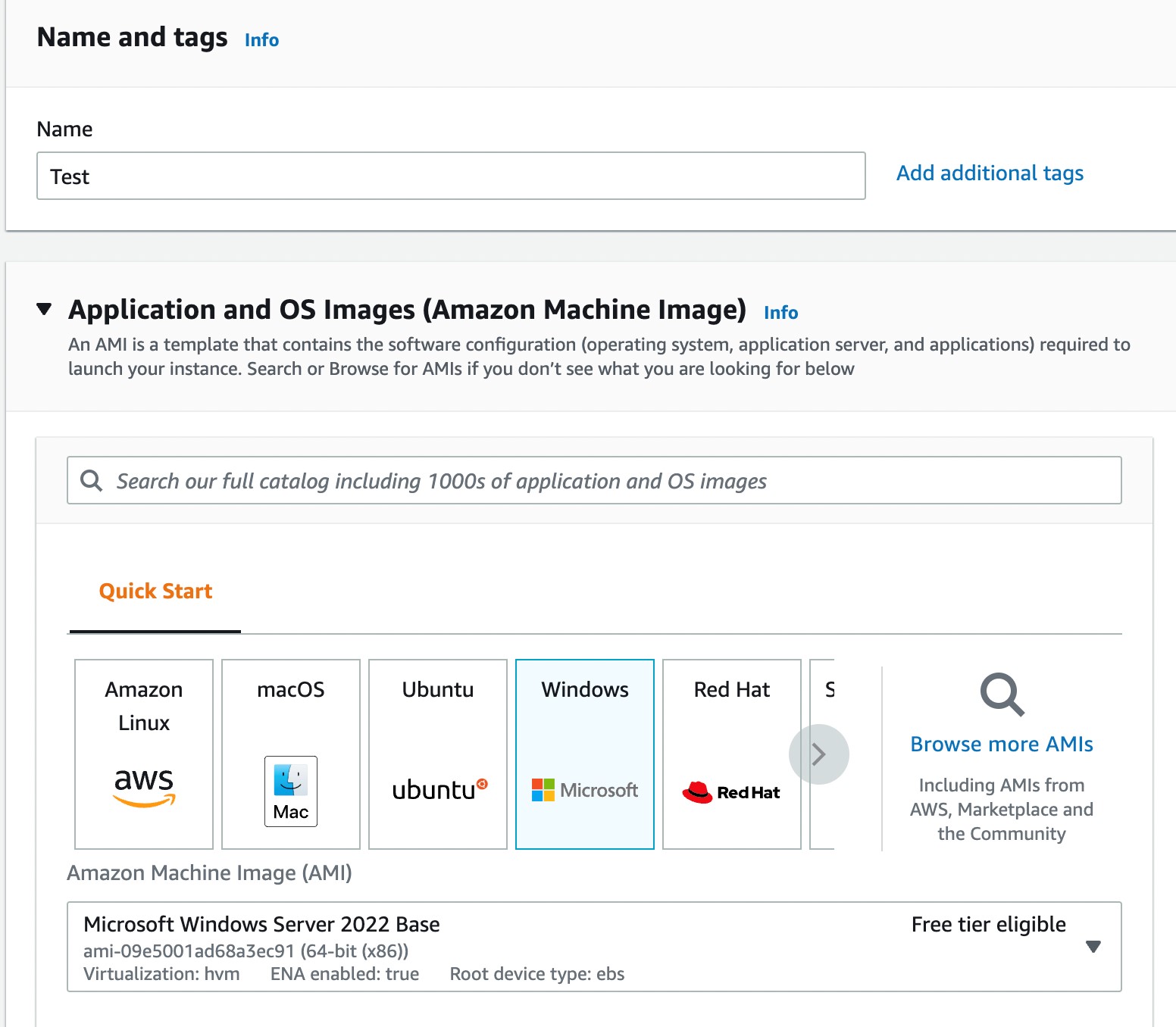
# **EXPERIMENT-1 : Create Windows and Linux EC2 instance and connect to it using RDP and putty respectively.**

## **WINDOWS**

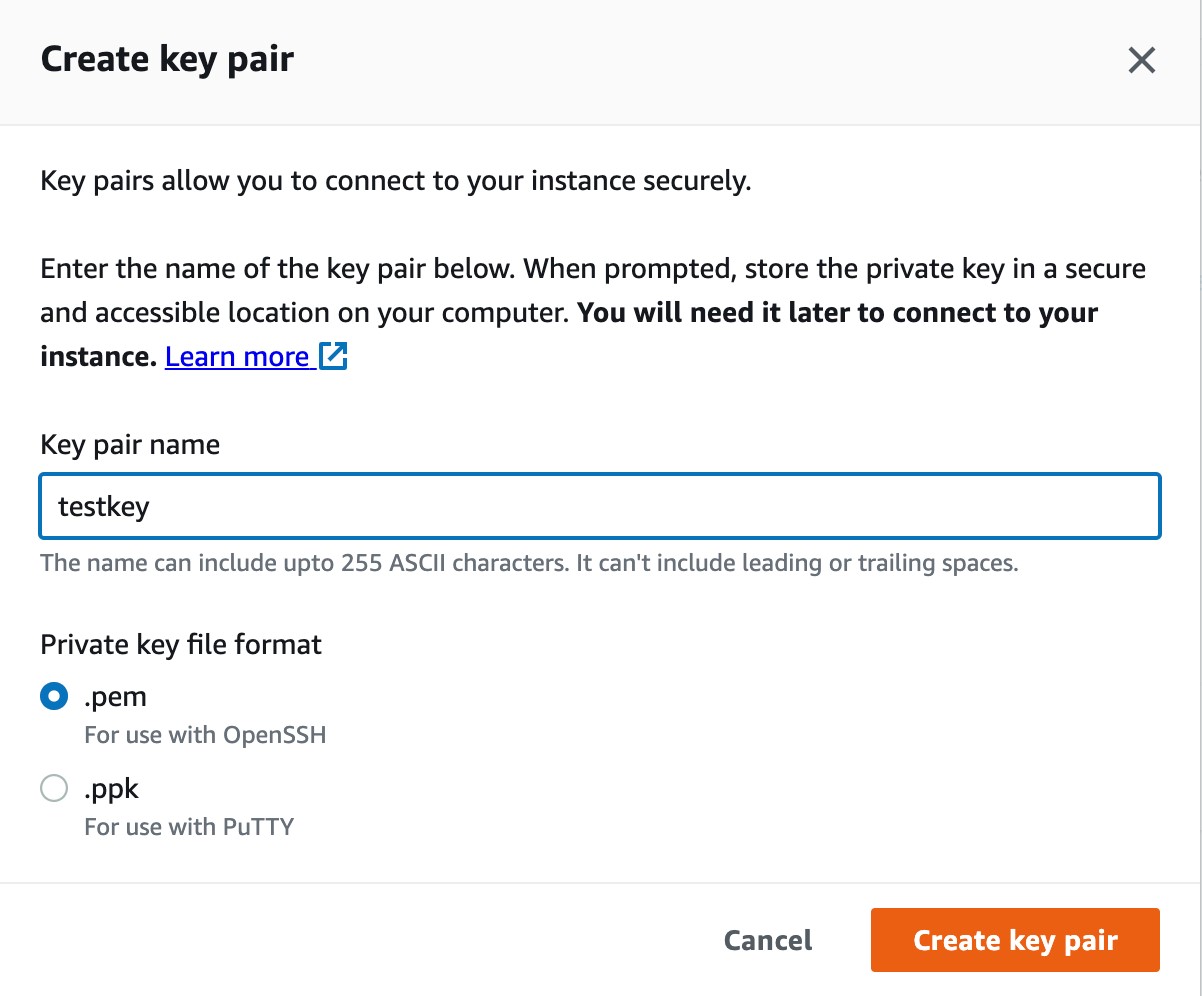
## **Step1**-Launching EC2 Instance

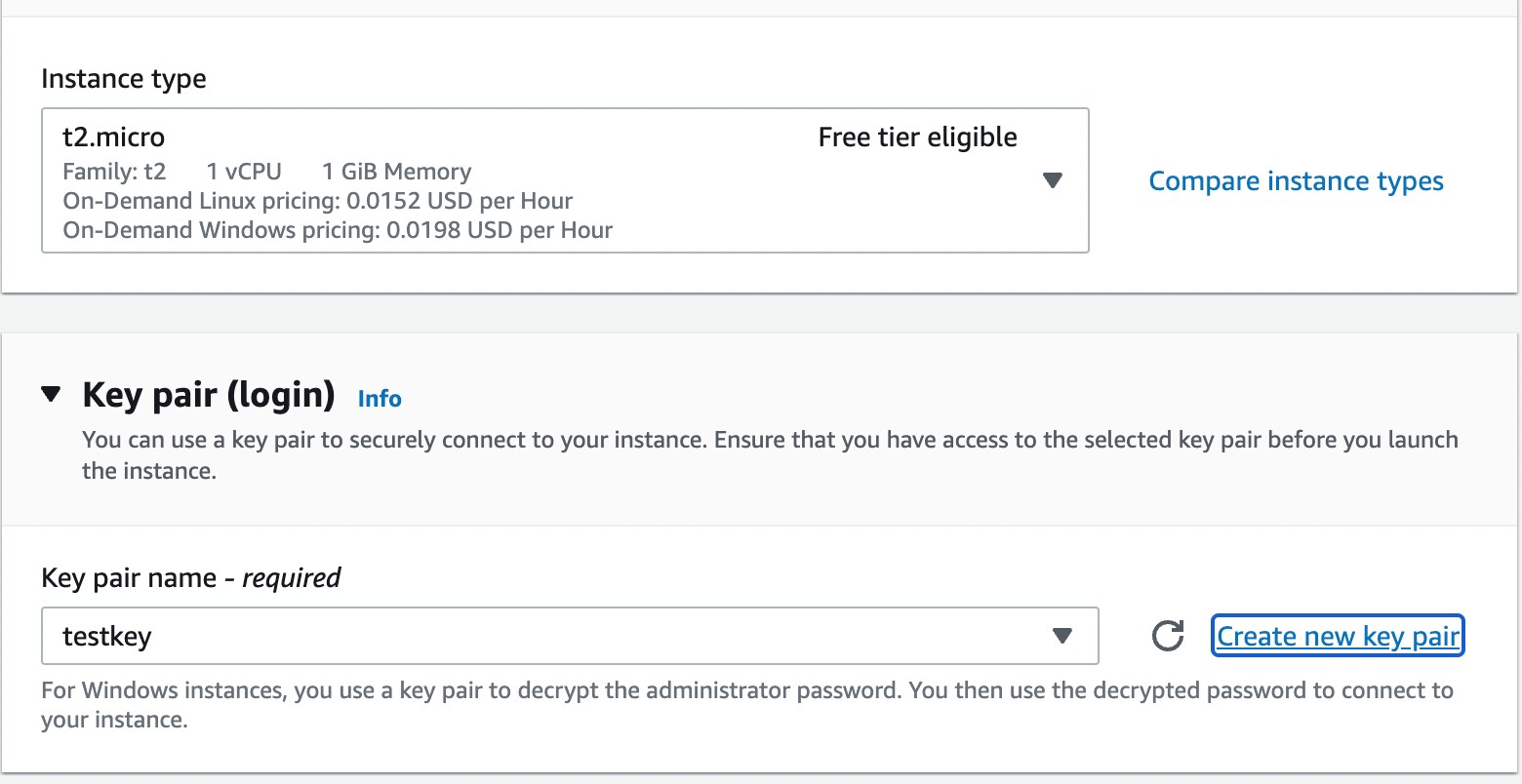


**Step-2:** Naming and choosing operating system for EC2 instance



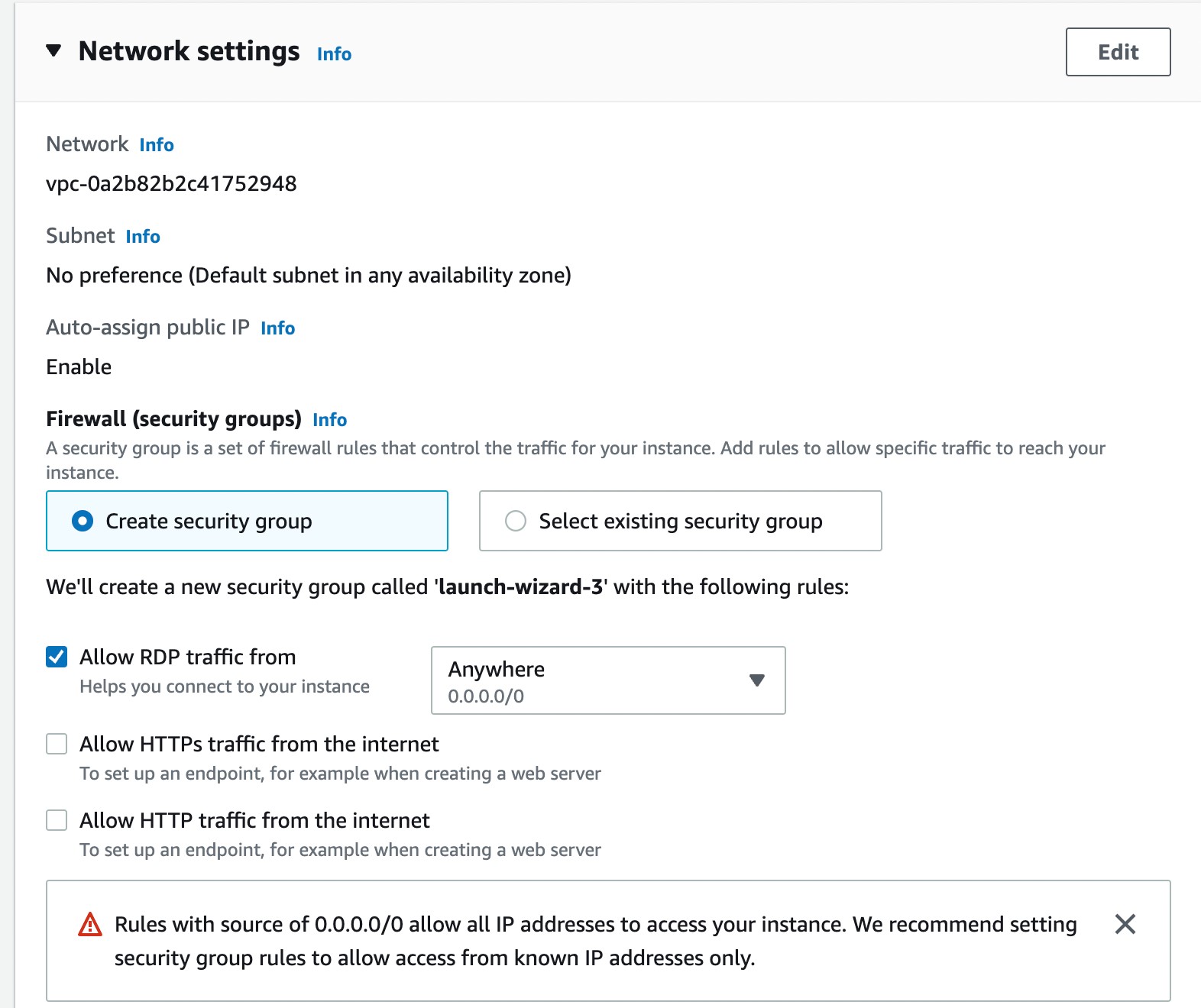
## **Step-3:** Creating key pair for instance



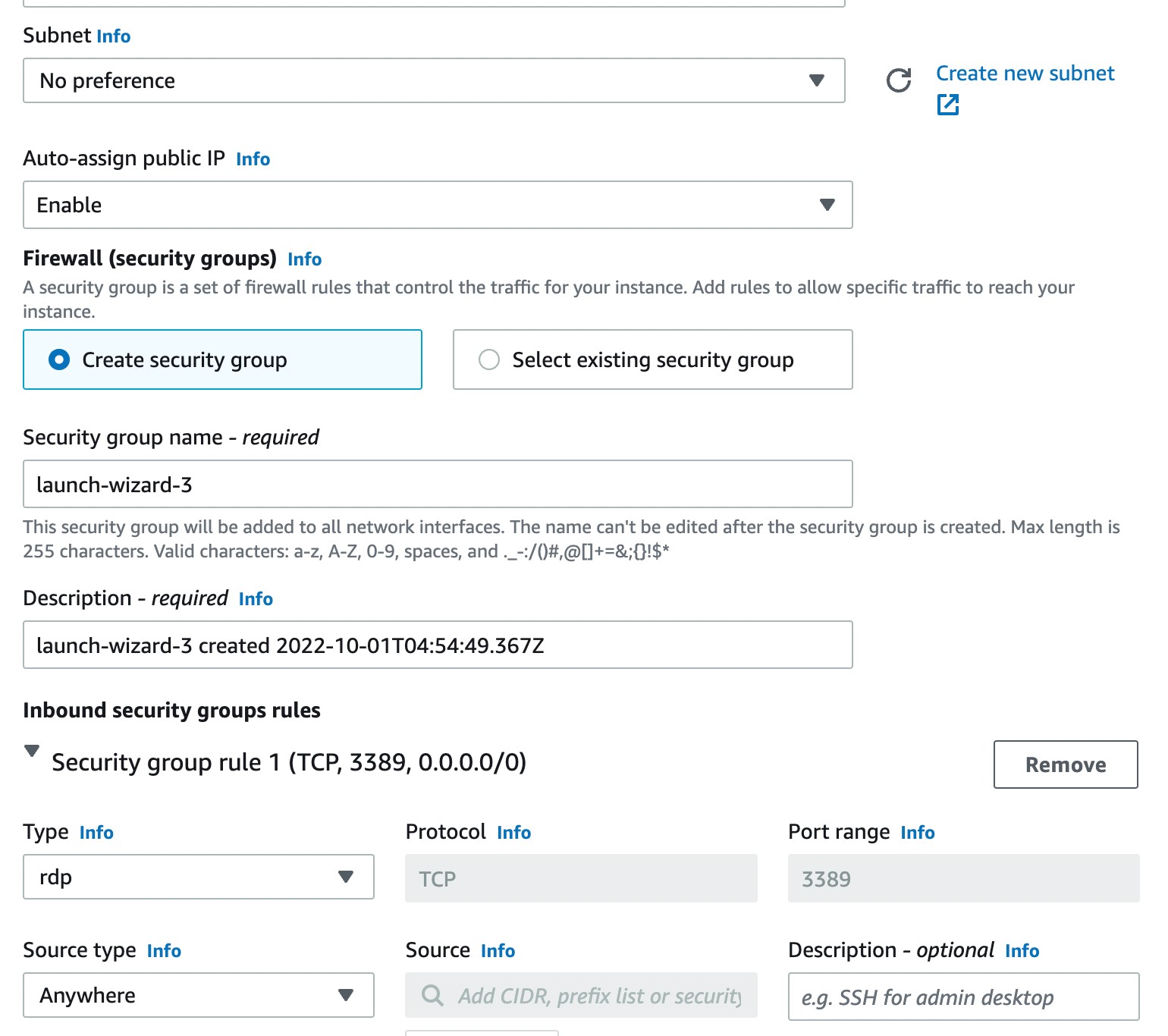


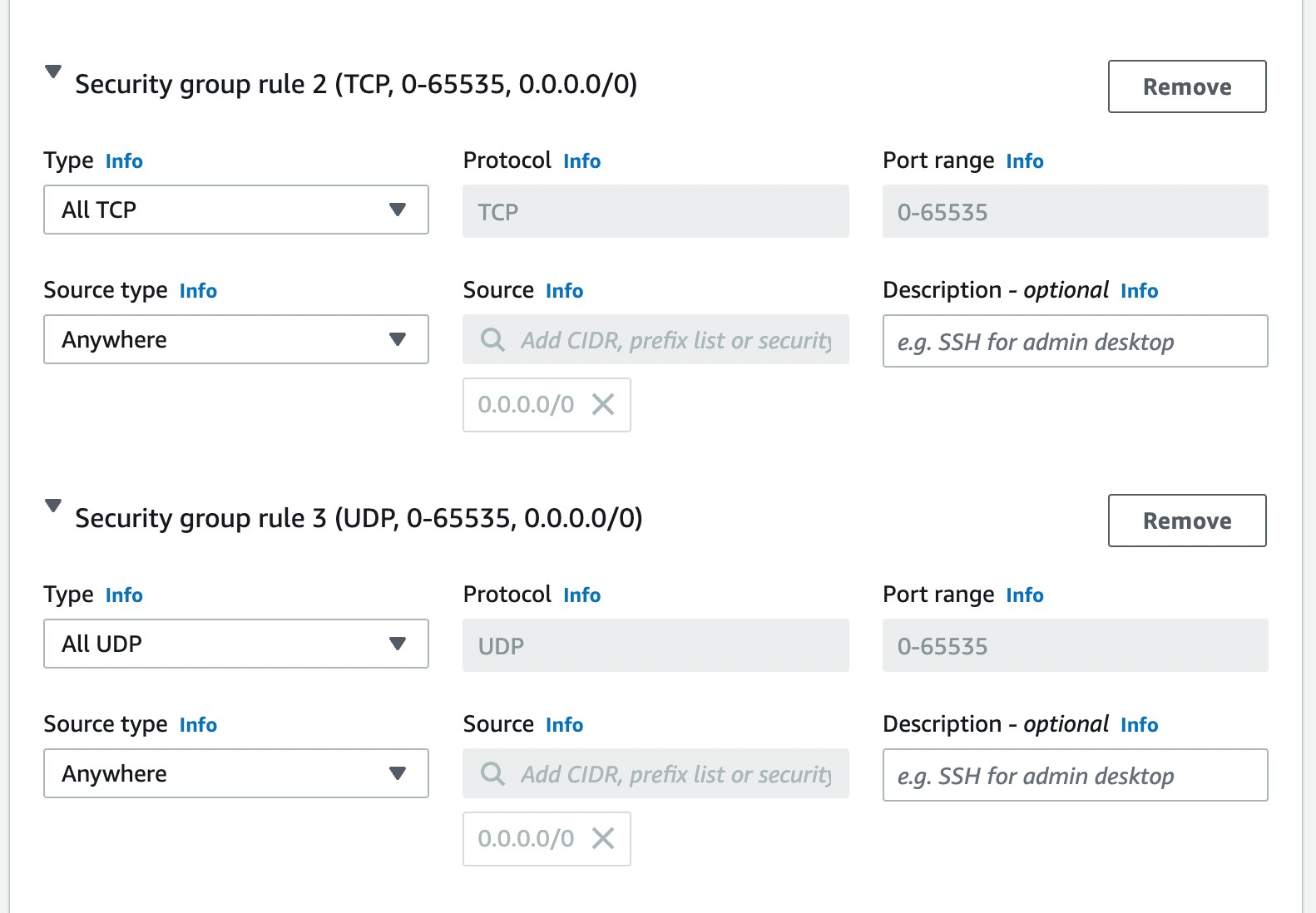
## **Step-4**:Creating new security groups of type All TCP and All UDP’

’



’

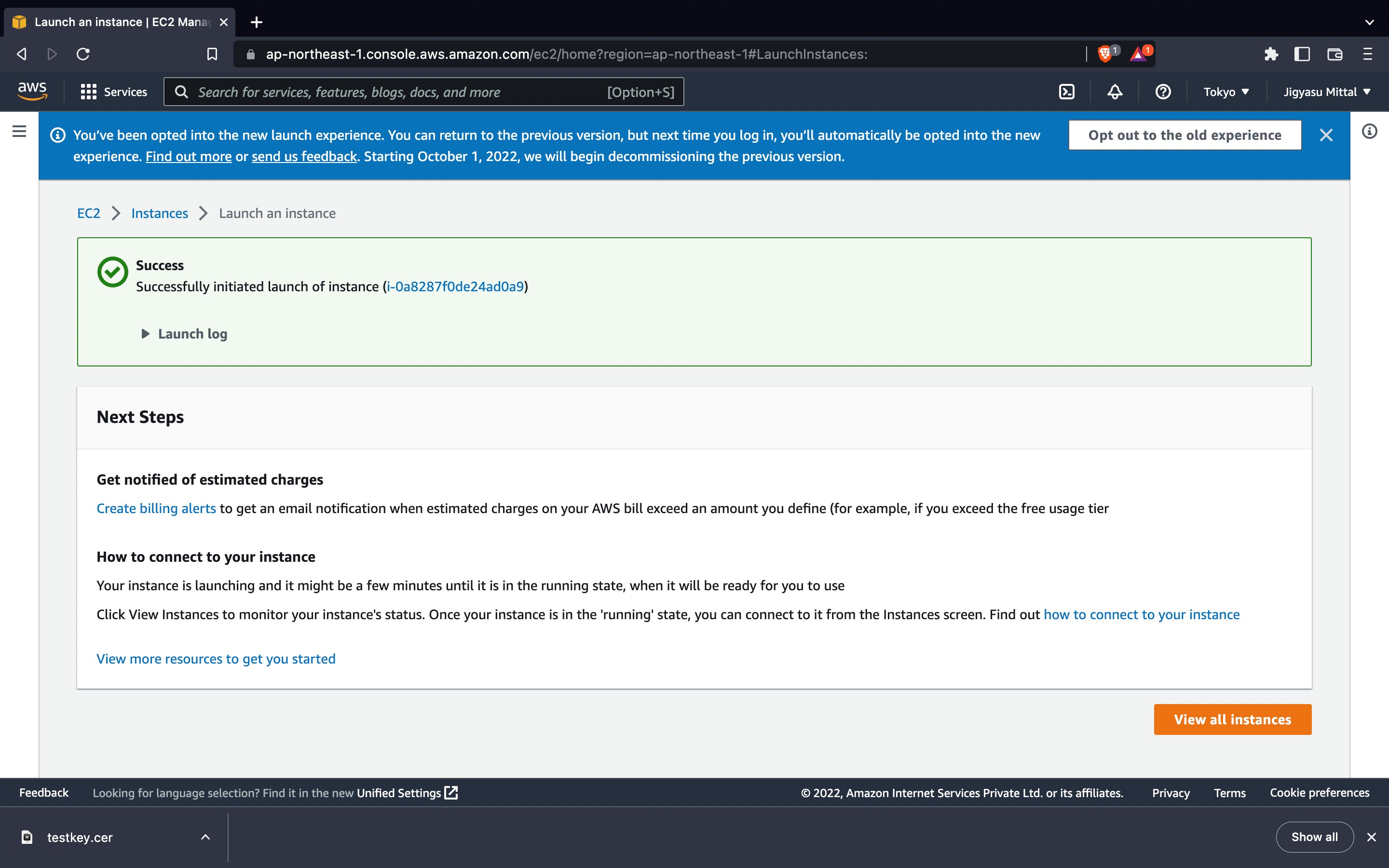




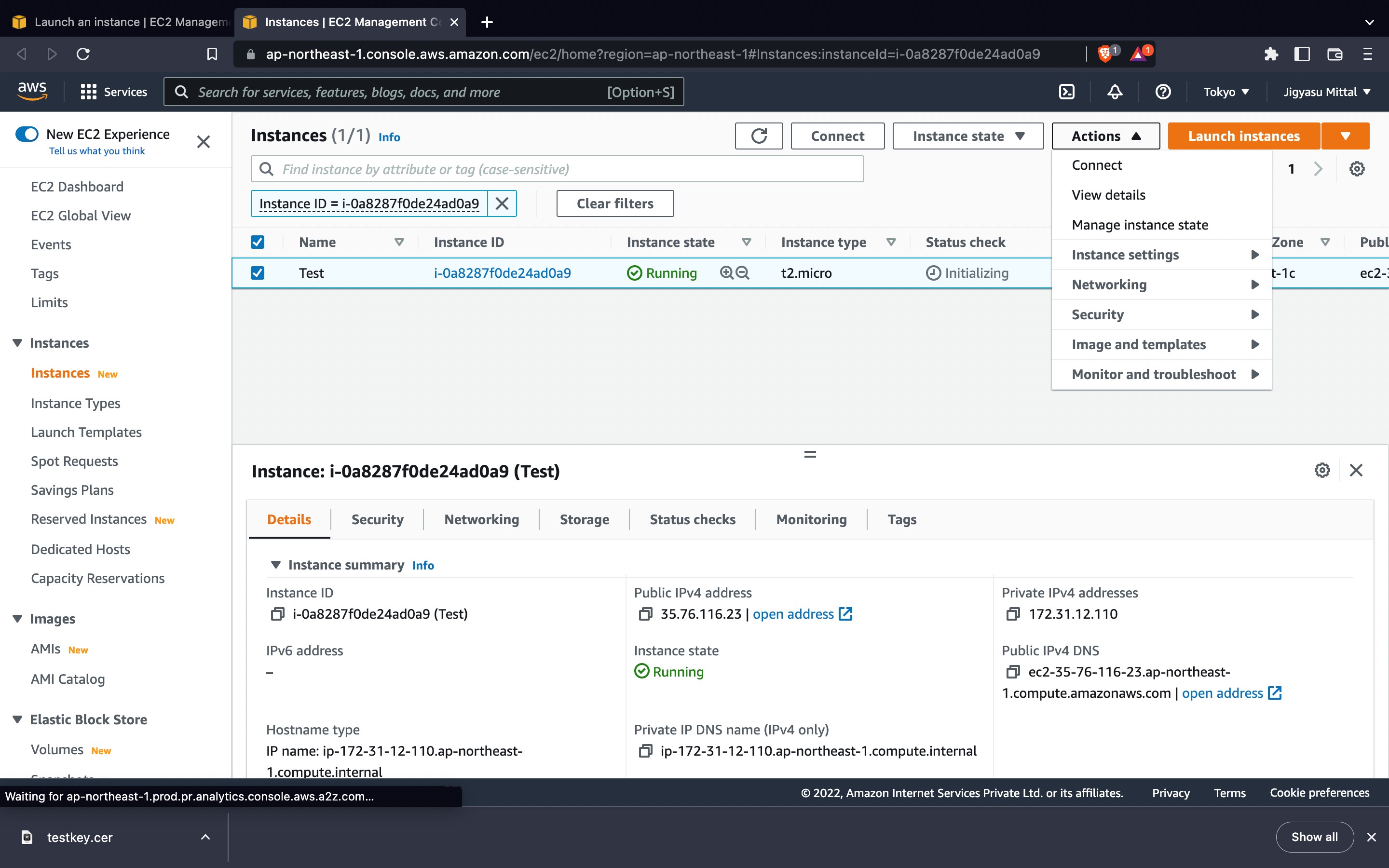
**Step-5:** Click on launch instance



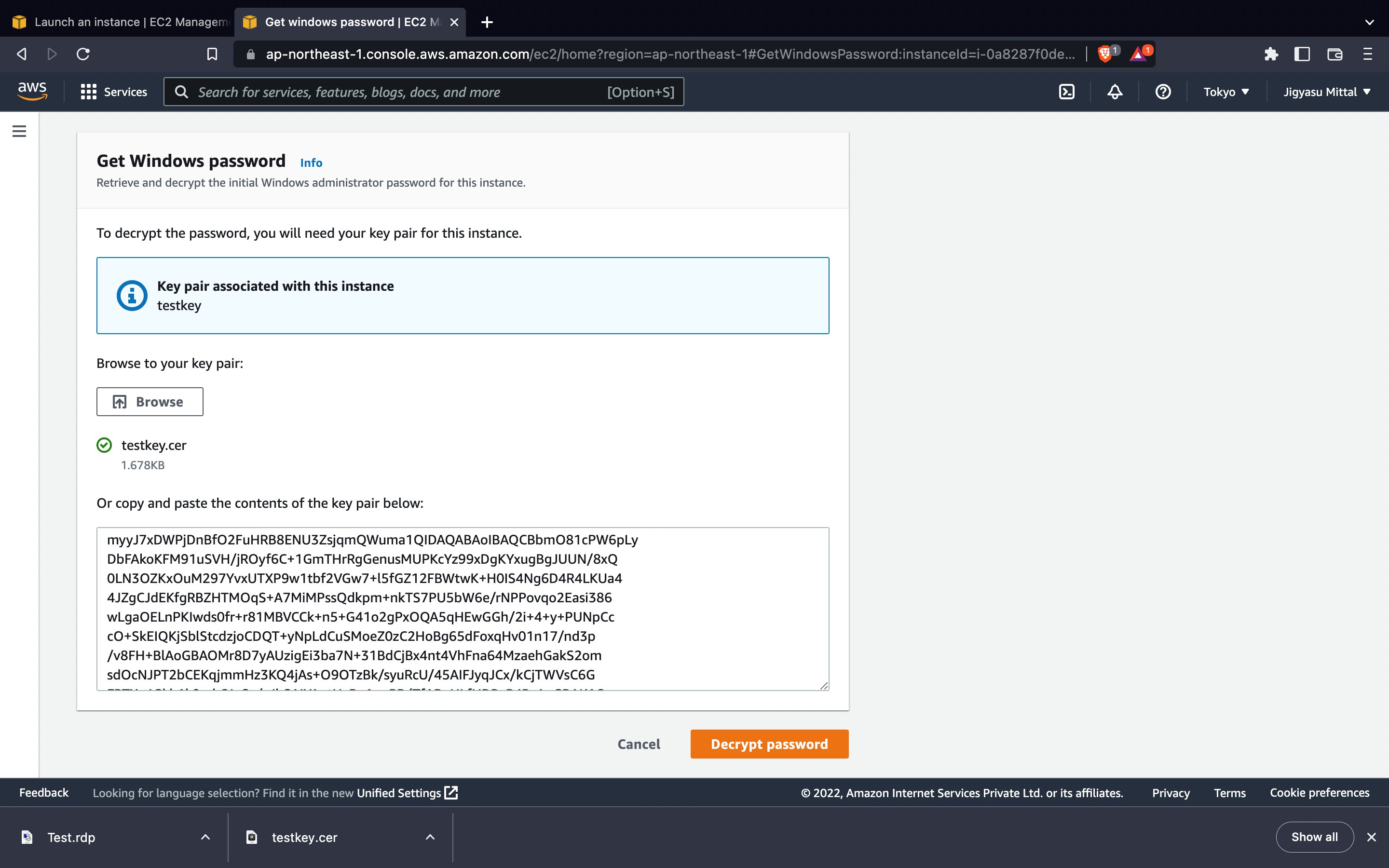
**Step-6:** We have successfully launched the instance.

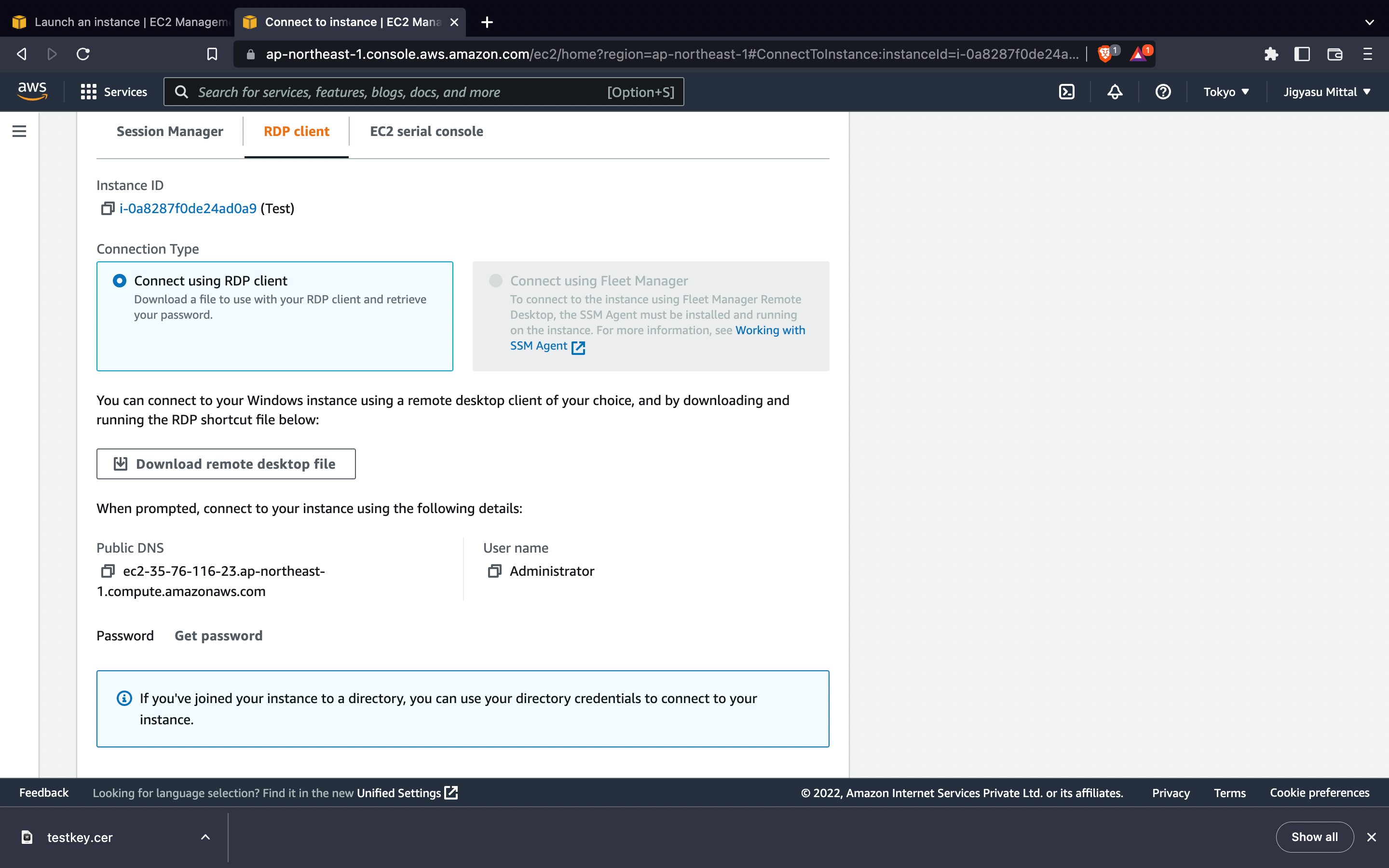


**Step-7:** Click on actions and choose ‘ Connect’

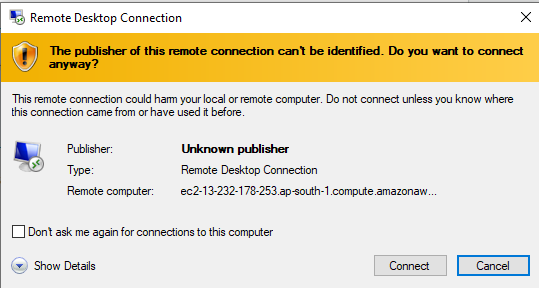


**Step-8:** Now get the password and decrypt that.

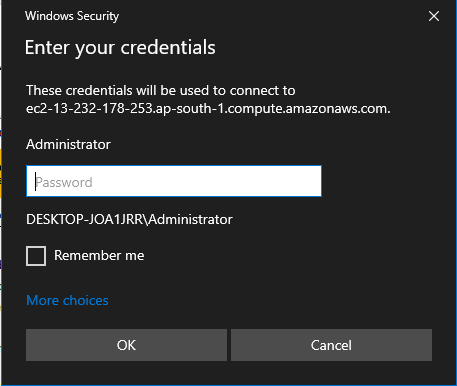




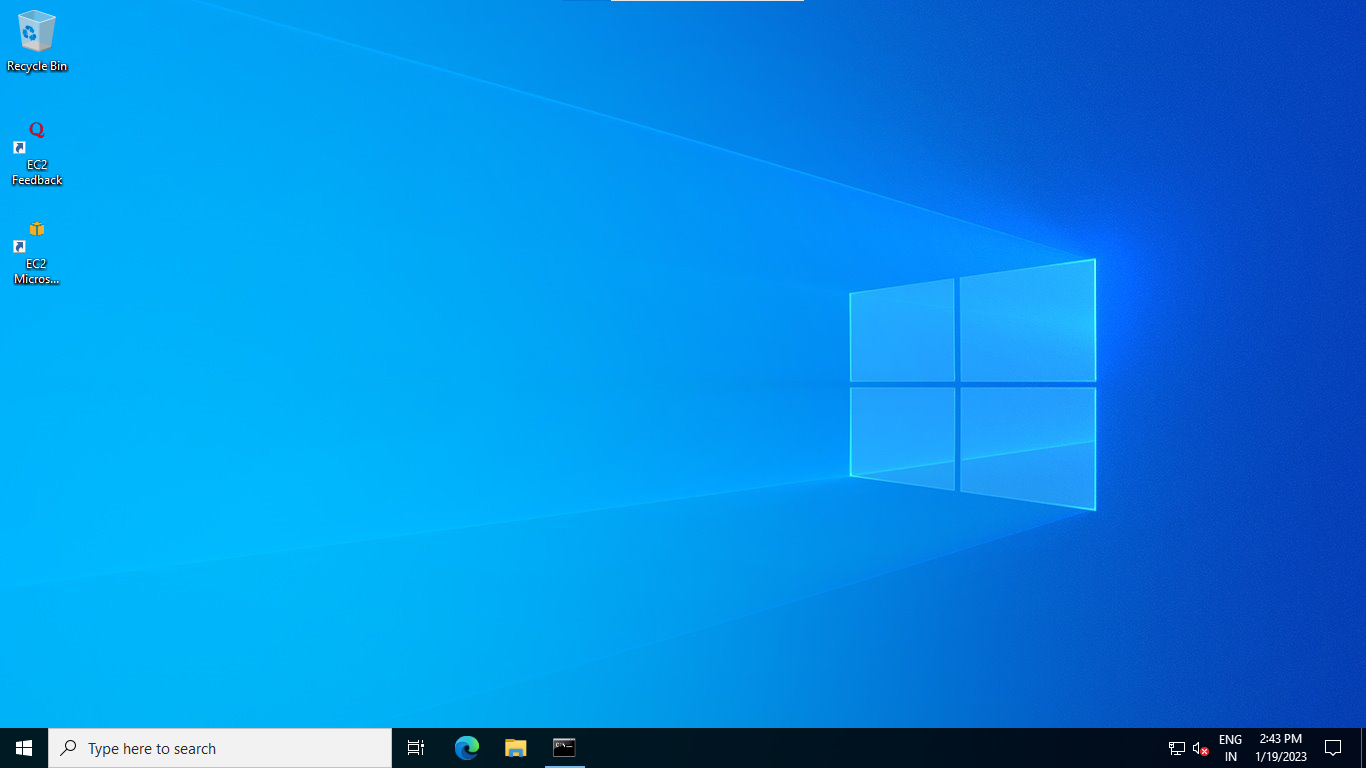
**Step-9:** Now open the remote desktop file and click on connect.



**Step-10:** Now fill the password that you decrypted using the pem file and click okay.



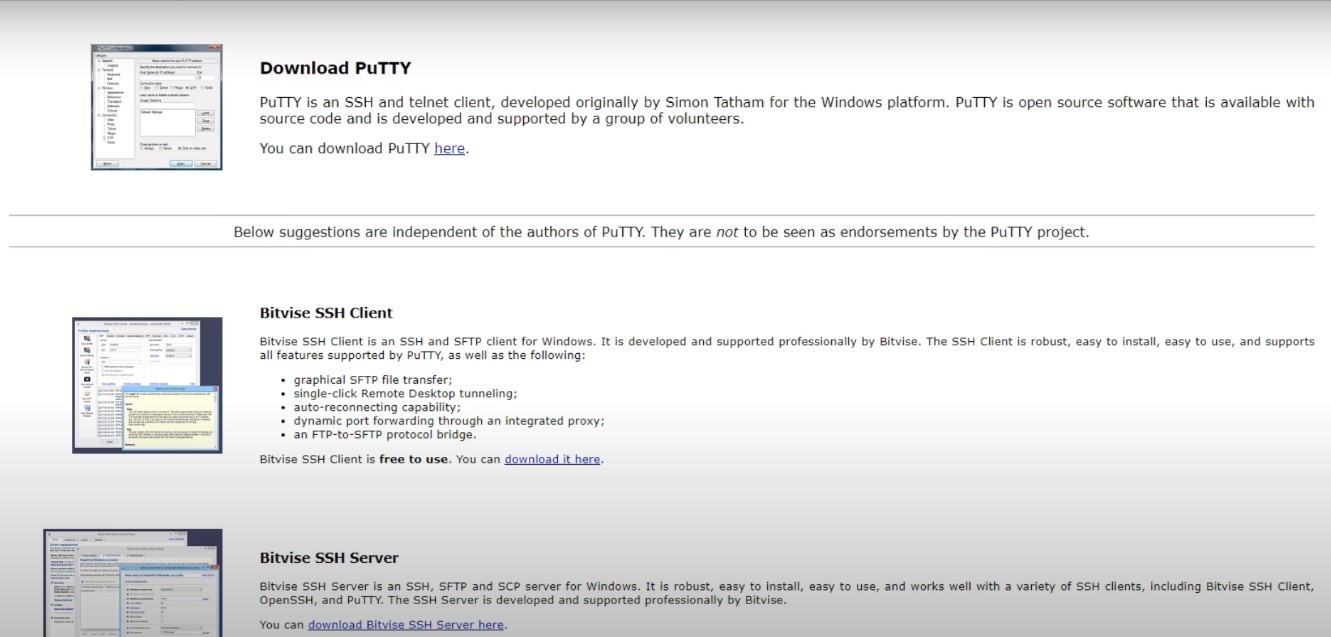
**Step-11:** We have successfully connected to EC2 instance.



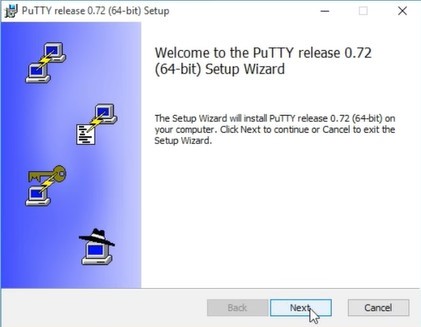
**LINUX**

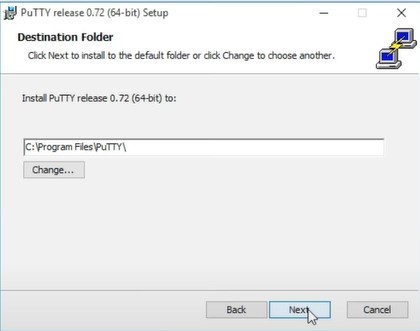
1. Connecting through putty

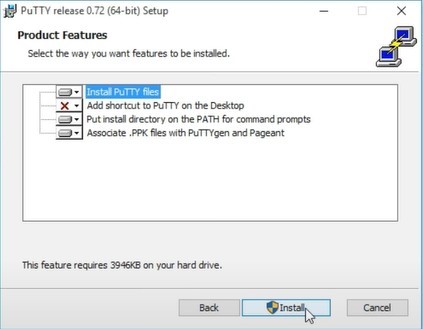
**Step-1:** First of all, we will download putty.

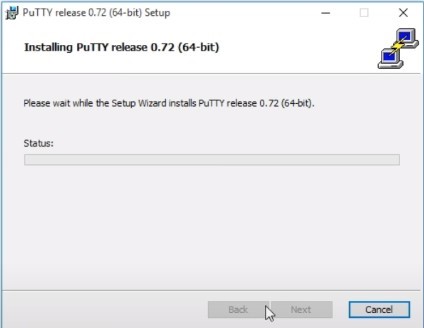


**Step-2**: Complete the installation process.



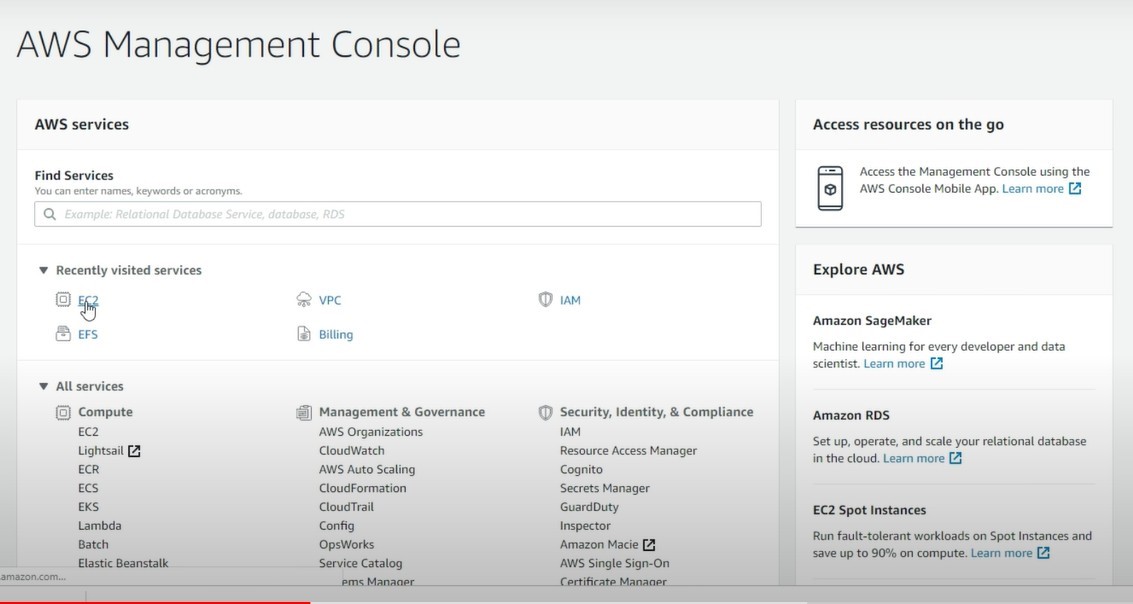








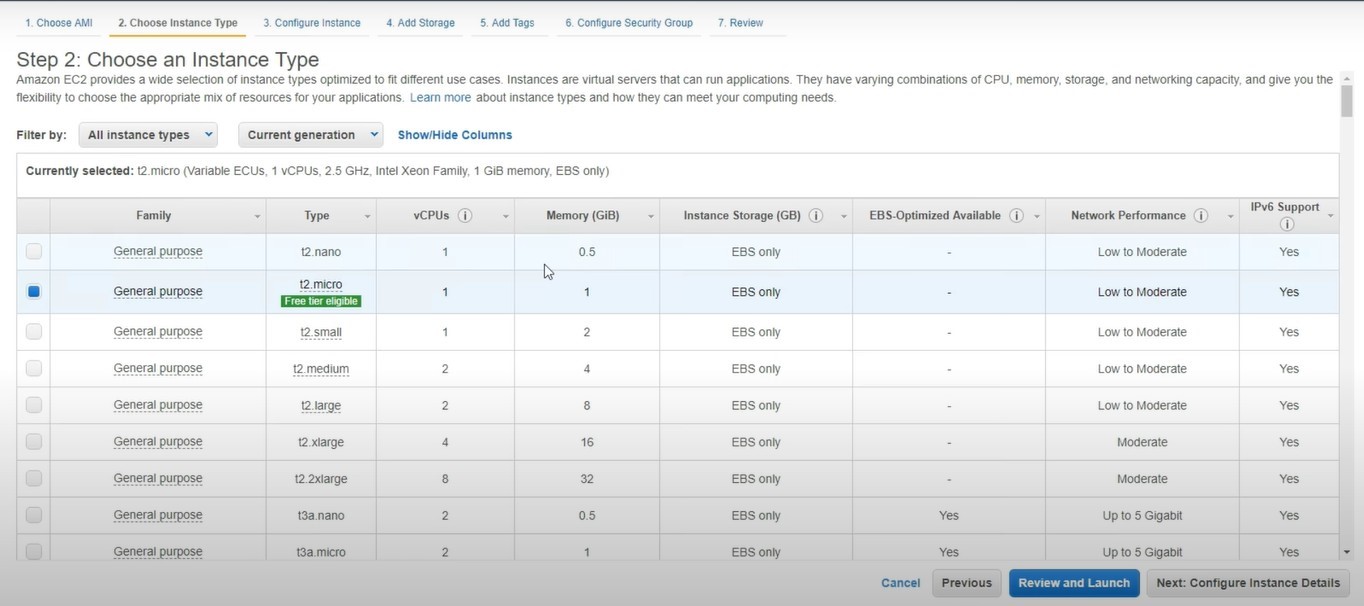
**Step-3**: Go to AWS console and go to EC2.



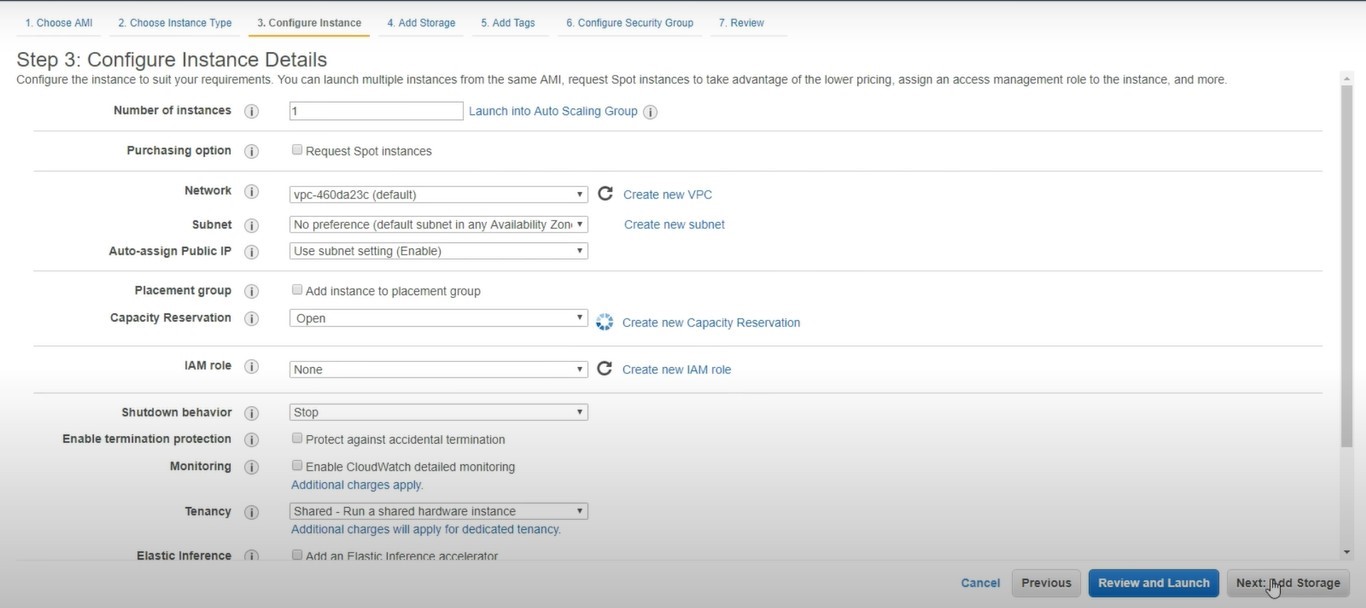
**Step-4:** Select the Amazon Linux 2 AMI



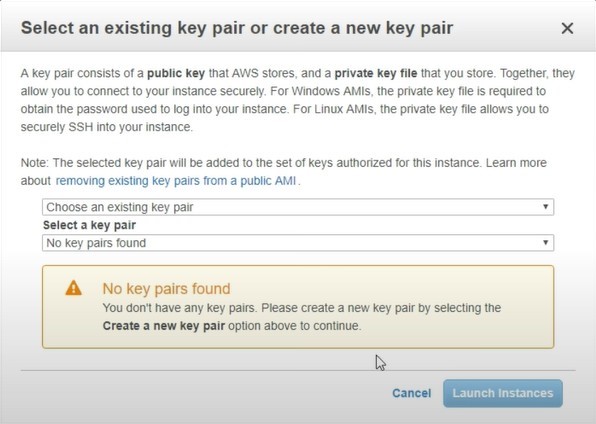
**Step-5:** Now choose the instance type.

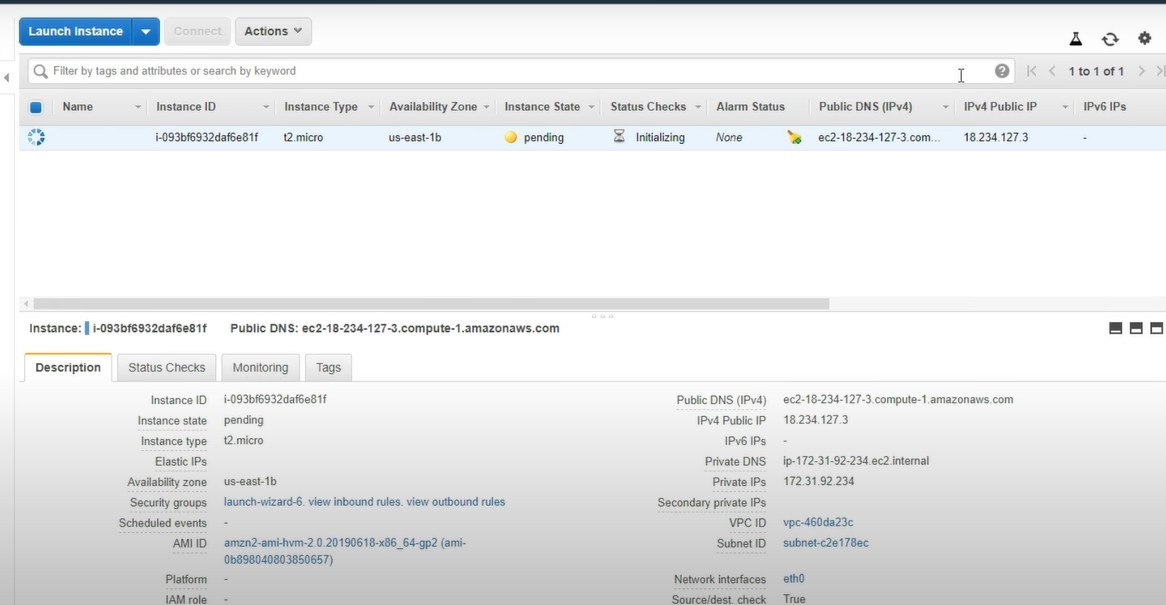


**Step-6:** Now configure the instance details.

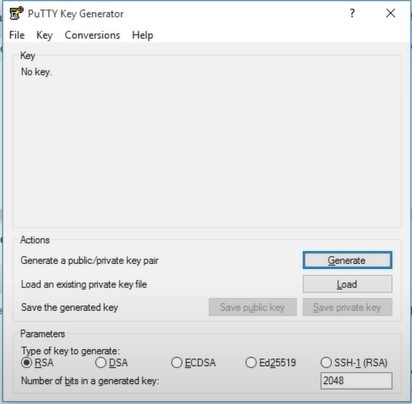


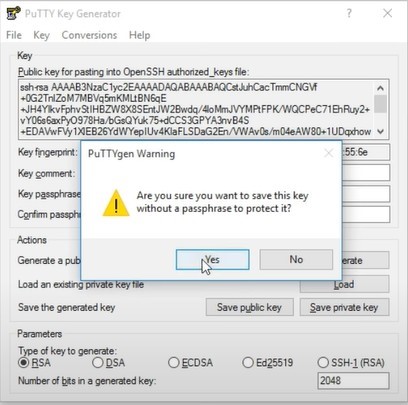
**Step-7:** Now select an existing key pair or create a new pair.



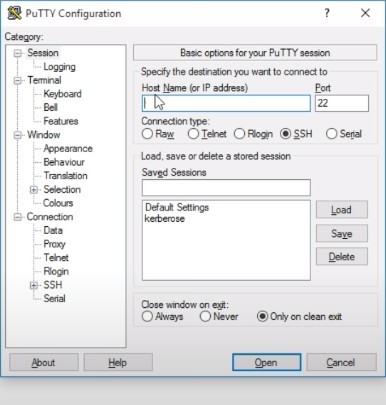


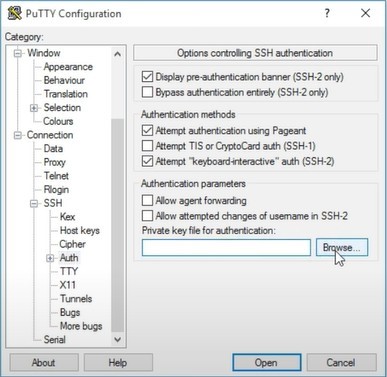
**Step-8:** Now convert the key from pme to ppk.





**Step-9:** Now fill in the putty configuration.



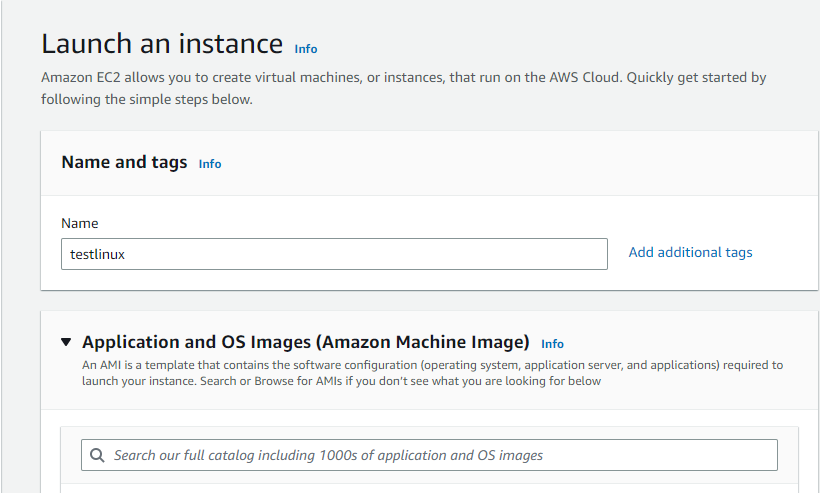


**Step-10:** Successfully launched

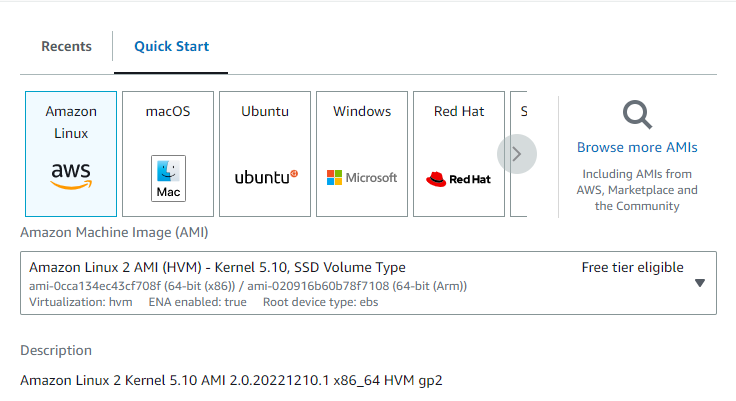


1. Connecting through RDP

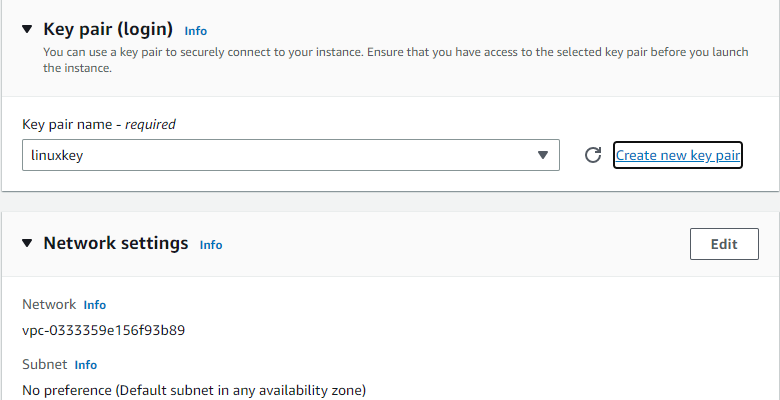
## **Step1**-Naming EC2 Instance.



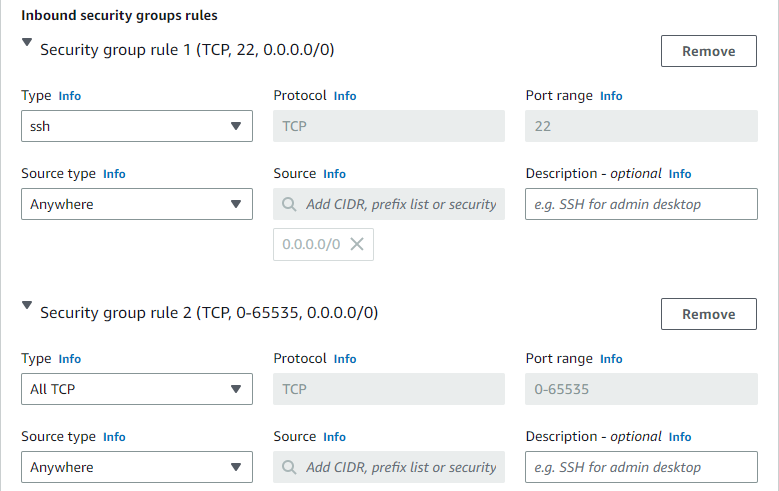
## **Step2**-Choosing the Amazon machine image as Linux.

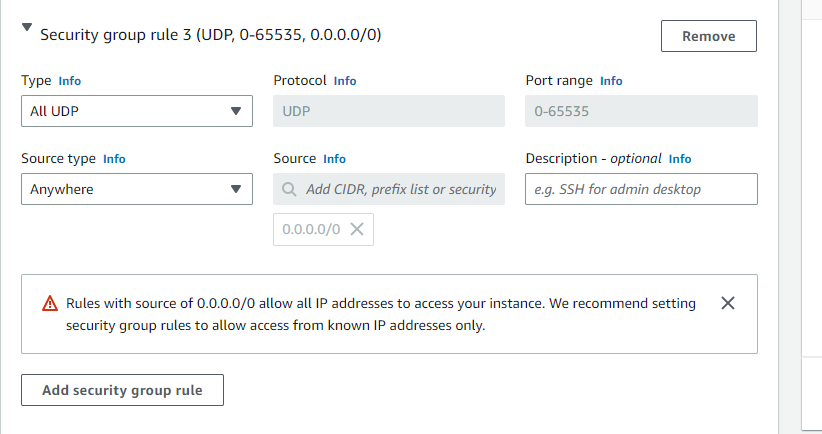


## **Step3**-Creating a new key pair.

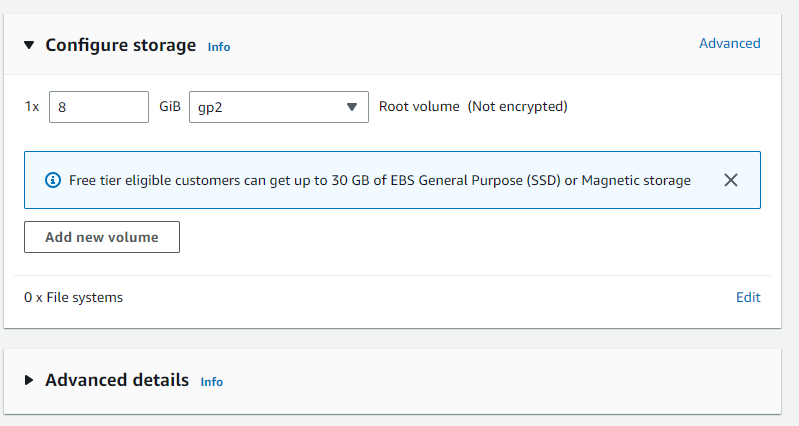


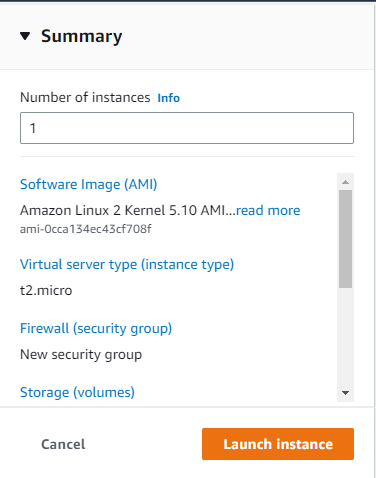
## **Step4**-Add security groups.



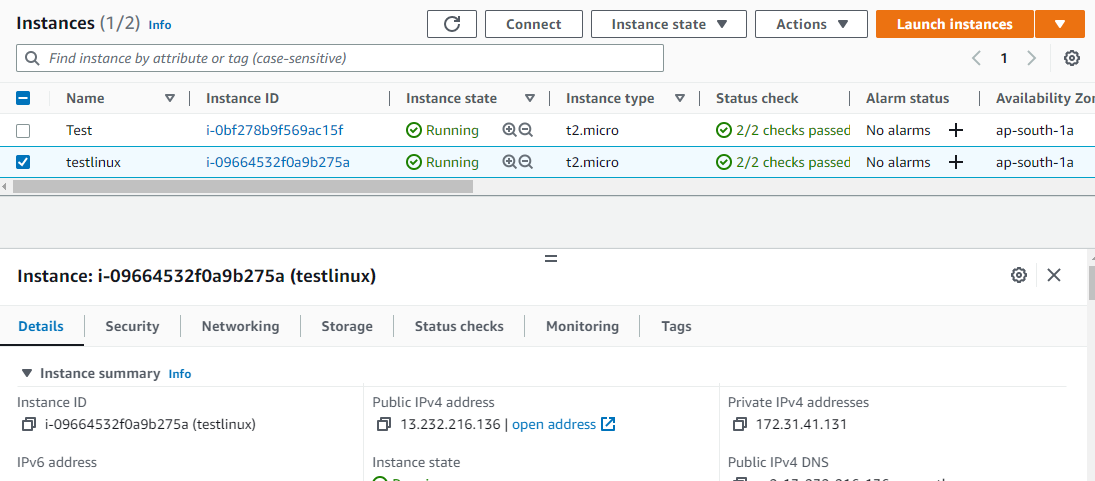


## **Step-5**-Configure the storage and click on launch instance.

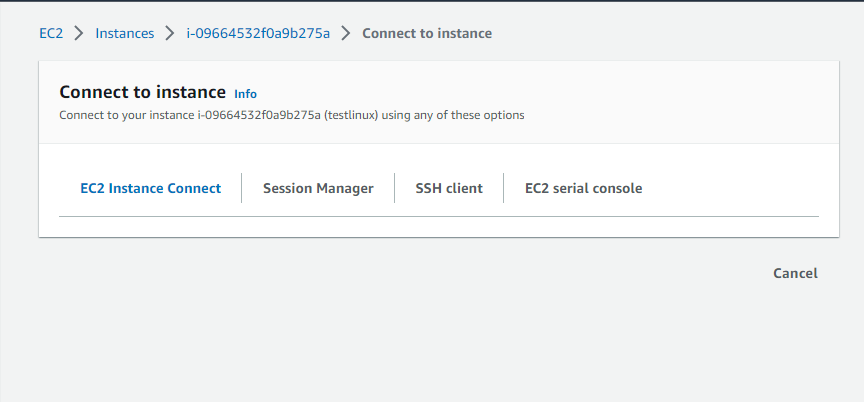




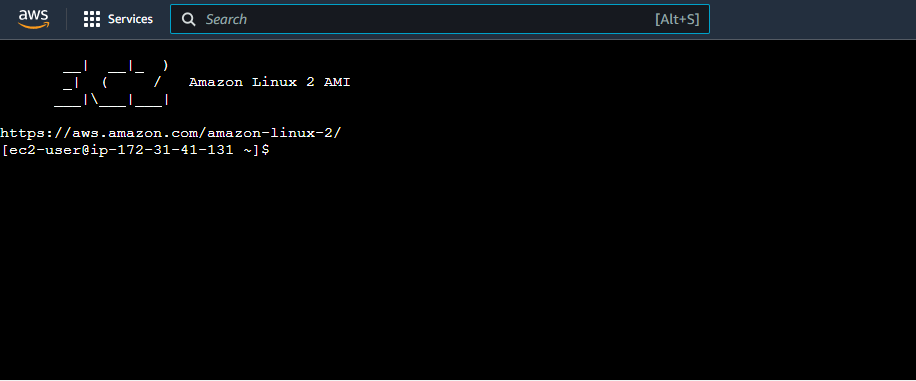
## **Step-6**- Now select the instance that you have created and connect.



## **Step-7**- Click on EC2 instance connect.



**Step-8**- We have successfully connected to the EC2 instance.



**QUESTIONS**

**Ques-1:** What are the various Compute services of AWS?

**Ans :** Amazon Web Services (AWS) offers a variety of compute services, including:

1. **Amazon Elastic Compute Cloud (EC2)** - allows users to launch and manage virtual machines in the cloud
2. **Amazon Elastic Container Service (ECS)** - enables users to deploy, run, and scale containerized applications
3. **AWS Lambda** - allows users to run code without provisioning or managing servers
4. **Amazon Lightsail** - provides a simplified way to launch and manage web applications and services
5. **AWS Elastic Beanstalk** - enables users to quickly deploy and run web applications and services
6. **AWS Batch** - allows users to run batch computing workloads on the AWS platform
7. **AWS App Runner** - enables users to build, test, and deploy containerized applications quickly
8. **AWS App Mesh** - allows users to monitor and control microservices running on Amazon ECS and Amazon EKS.

**Ques-2:** What are the different instance types and pricing plans in AWS EC2?

**Ans :** Amazon Elastic Compute Cloud (EC2) offers a variety of instance types optimized for different use cases. Some of the instance types available are:

* **General purpose:** These instances are suitable for a wide variety of workloads, including web servers, small and medium-sized databases, and development environments. Examples include the t2, m5, and m5a instances.
* **Compute optimized:** These instances are optimized for compute-intensive workloads, such as batch processing, distributed analytics, and high-performance computing. Examples include the c5 and c5n instances.
* **Memory optimized:** These instances are optimized for memory-intensive workloads, such as high-performance databases, in-memory caching, and real-time analytics. Examples include the r5, r5a, and x1e instances.
* **GPU instances:** These instances are optimized for graphics and parallel compute workloads, such as machine learning, data science, and gaming. Examples include the p3 and g4 instances.
* **ARM instances:** These instances are based on ARM architecture and optimized for scale-out workloads. Examples include m6g and c6g instances.

**Pricing** for EC2 instances is based on a pay-as-you-go model, with hourly or per-second charges. Pricing also varies based on the region and availability zone, as well as the specific instance type and any additional resources (such as storage or IP addresses) that you may need. The EC2 pricing page on the AWS website provides detailed pricing information and allows you to compare the costs of different instances and configurations.

**Ques-3:** What is AMI and its different possibilities in AWS?

**Ans :** An Amazon Machine Image (AMI) is a pre-configured virtual machine image, which is used to launch an instance in the Amazon Elastic Compute Cloud (EC2). It contains the necessary information to launch a fully-functional instance, including the operating system, application server, and application.

There are several different possibilities with AMIs in AWS, including:

1. **Creating your own custom AMI**: You can create your own AMI by launching an instance, configuring it as desired, and then creating an image of the instance.
2. **Using pre-built AMIs**: AWS provides a wide variety of pre-built AMIs for different operating systems, including Windows and Linux. These AMIs can be used to quickly launch an instance with a specific configuration.
3. **Shared AMIs**: You can also share your custom AMIs with other users, allowing them to launch instances using your image.
4. **Marketplace AMIs**: AWS Marketplace provides a wide variety of pre-built AMIs from independent software vendors (ISVs). These AMIs can be used to quickly launch instances with specific software configurations.
5. **Using 3rd Party AMIs**: You can also use AMIs from 3rd Party providers for different use cases like for Databases, Big Data, Analytics, Machine Learning, etc.

**Ques-4:** What is Region and Availability Zone in AWS?

**Ans :** In Amazon Web Services (AWS), a region is a geographical area that has multiple, isolated locations known as availability zones (AZs). Each region is completely independent and isolated from the others, and is designed to provide low latency and high throughput. Availability zones are physically separate locations within a region, each with redundant power, networking, and cooling. They are connected to each other via low-latency links. By using multiple availability zones, you can build highly available and fault-tolerant applications that can withstand the loss of an entire availability zone without interruption.