

FINAL REPORT

SUBJECT: CLOUD SECURITY AND PRIVACY

GROUP 4

MEMBER

BI12-201 Truong Gia Huy

BI12-328 Nguyen Le Nguyen

BI12-341 Nguyen Khang Ninh

BI12-216 Nguyen Ngoc Khoi

Contents

[**I- Introduction:** 3](#_Toc164507505)

[**II- Abstract:** 4](#_Toc164507506)

[**III- Amazon EC2:** 5](#_Toc164507507)

[**1- Definition:** 5](#_Toc164507508)

[**2- Key features:** 6](#_Toc164507509)

[**3- Basic Architecture:** 7](#_Toc164507510)

[**4- Roles:** 8](#_Toc164507511)

[**5- Provisioning Steps:** 9](#_Toc164507512)

[**6- Advantages and Disadvantages:** 10](#_Toc164507513)

[**a. Advantages** 10](#_Toc164507514)

[**b. Disadvantages:** 11](#_Toc164507515)

[**IV- Scenario:** 12](#_Toc164507516)

[**V- Demo:** 13](#_Toc164507517)

[**VI- Conclusion:** 20](#_Toc164507518)

[**VII- References:** 21](#_Toc164507519)

# **I- Introduction:**

Amazon EC2 (Elastic Compute Cloud) is a foundational service provided by Amazon Web Services (AWS), offering resizable compute capacity in the cloud. With EC2, users can rent virtual servers, known as instances, on which they can run their own applications. It provides a flexible, scalable, and reliable solution for hosting various applications, from simple websites to complex enterprise applications. In this article, we will explore the key features and benefits of Amazon EC2, including its role in providing a secure, cost-effective, and efficient cloud computing solution. Whether you are a startup, a small business, or a large enterprise, Amazon EC2 provides the tools and resources needed to run virtually any workload with ease and efficiency.

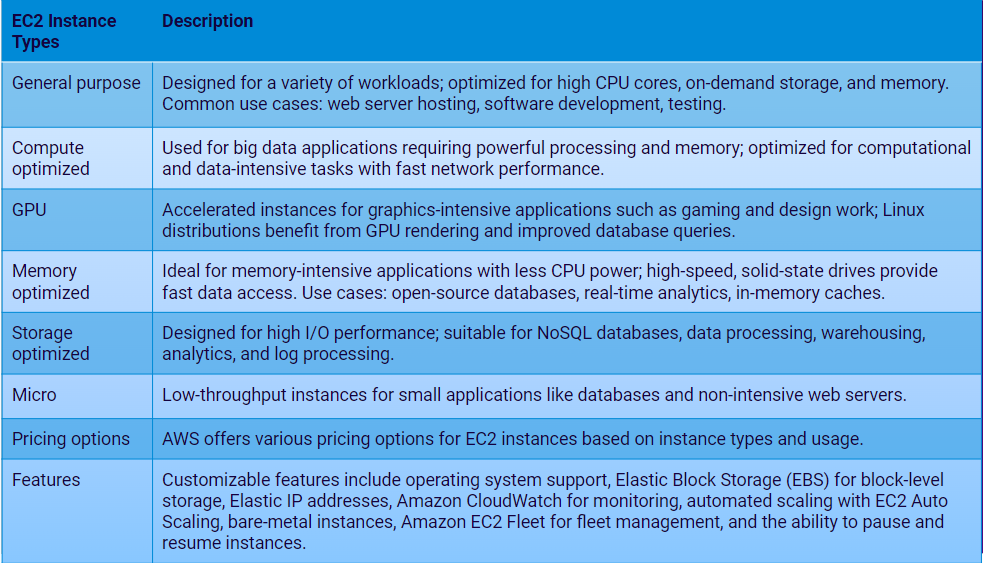
# **II- Abstract:**

Amazon Web Services (AWS) provides various services and Amazon Elastic Compute Cloud (Amazon EC2) is one of the services. Amazon EC2 may be used to create a virtual host server. Amazon EC2 provides a wide selection of instance AMIs (Amazon Machine Images) to choose from when creating a virtual server. In this project, we will discuss developing a simple web application on the Amazon EC2 instance.

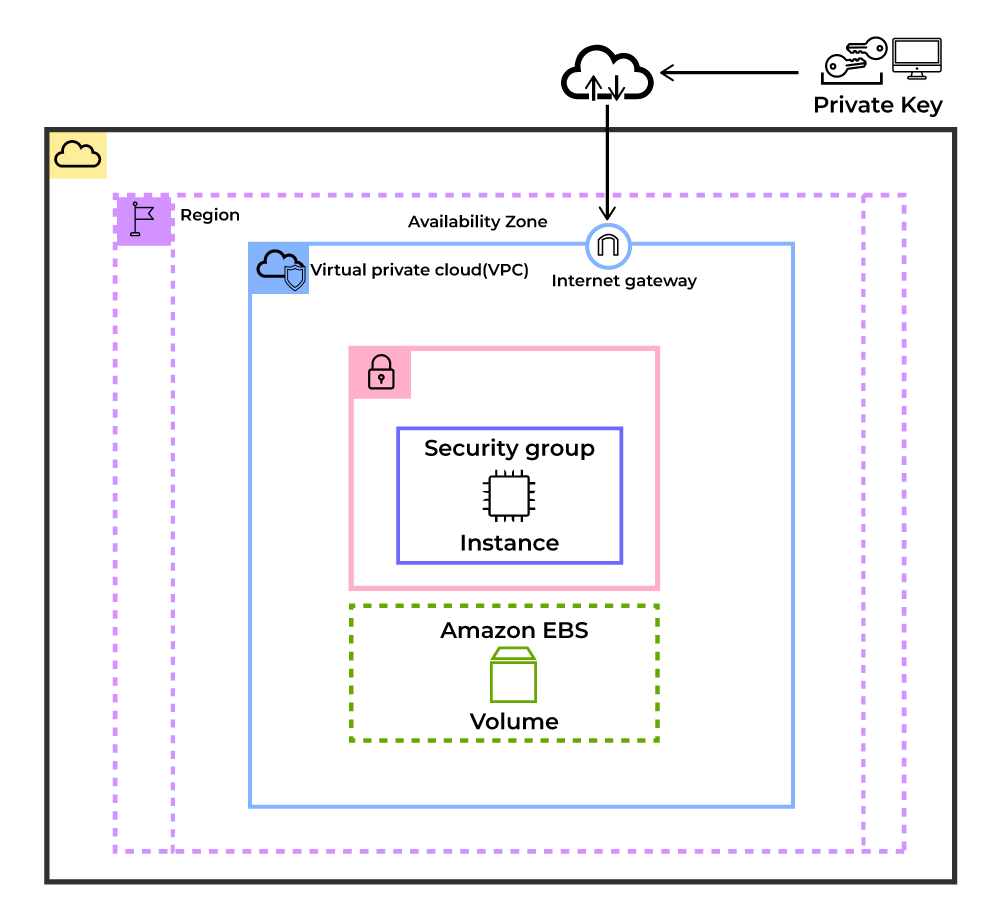
# **III- Amazon EC2:**

## **1- Definition:**

Amazon EC2, or Elastic Compute Cloud, is a key service in Amazon Web Services (AWS). It allows users to rent virtual servers, known as instances, for running their applications. EC2 offers resizable compute capacity, enabling users to scale resources according to demand, eliminating the need for physical hardware investment. Users have control over instance types, CPU, memory, storage, and networking configurations, making it suitable for various applications.



The table of the EC2 instance types

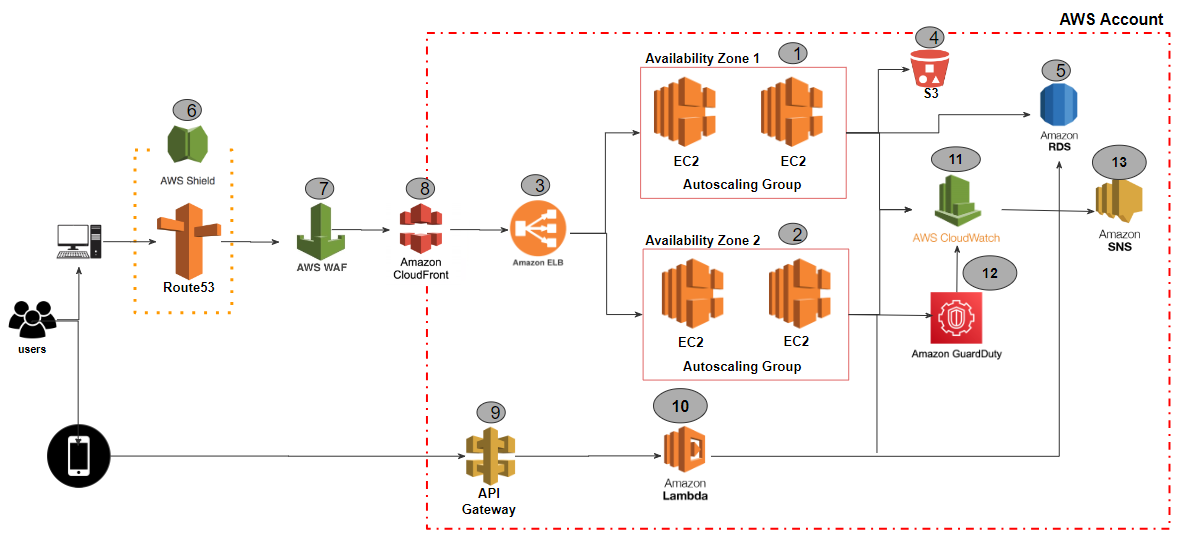


|  |
| --- |
| The EC2 instance deployed in VPC (Virtual Private Cloud) |

## **2- Key features:**

* Functionality: EC2 offers users a virtual computing platform with the ability to perform operations and launch additional instances. This enhances security for virtual devices. Users can customize their environment as needed throughout the virtual machine's lifespan. EC2 provides default AMI options supporting various operating systems and pre-configured resources, and users can also create custom AMIs. Storing user-defined AMIs allows for easy reference and avoids reconfiguration when creating new EC2 instances.
* Operating Systems: Amazon EC2 includes a wide range of operating systems to choose from while selecting your AMI. Not only are these selected options, but users are also even given the privilege to upload their own operating systems and opt for that while selecting AMI during launching an EC2 instance.
* Software: Amazon is single-handedly ruling the cloud computing market, because of the variety of options available on EC2 for its users. It allows its users to choose from various software present to run on their EC2 machines. This whole service is allocated to AWS Marketplace on the AWS platform. Numerous software like SAP, LAMP, Drupal, etc are available on AWS to use.
* Scalability and Reliability: EC2 offers the capability to adjust our computing capacity to match our requirements, whether that means increasing or decreasing it. This feature allows EC2 to handle dynamic situations effortlessly. Additionally, its reliability is greatly enhanced by the flexibility provided by volumes and snapshots.

## **3- Basic Architecture:**

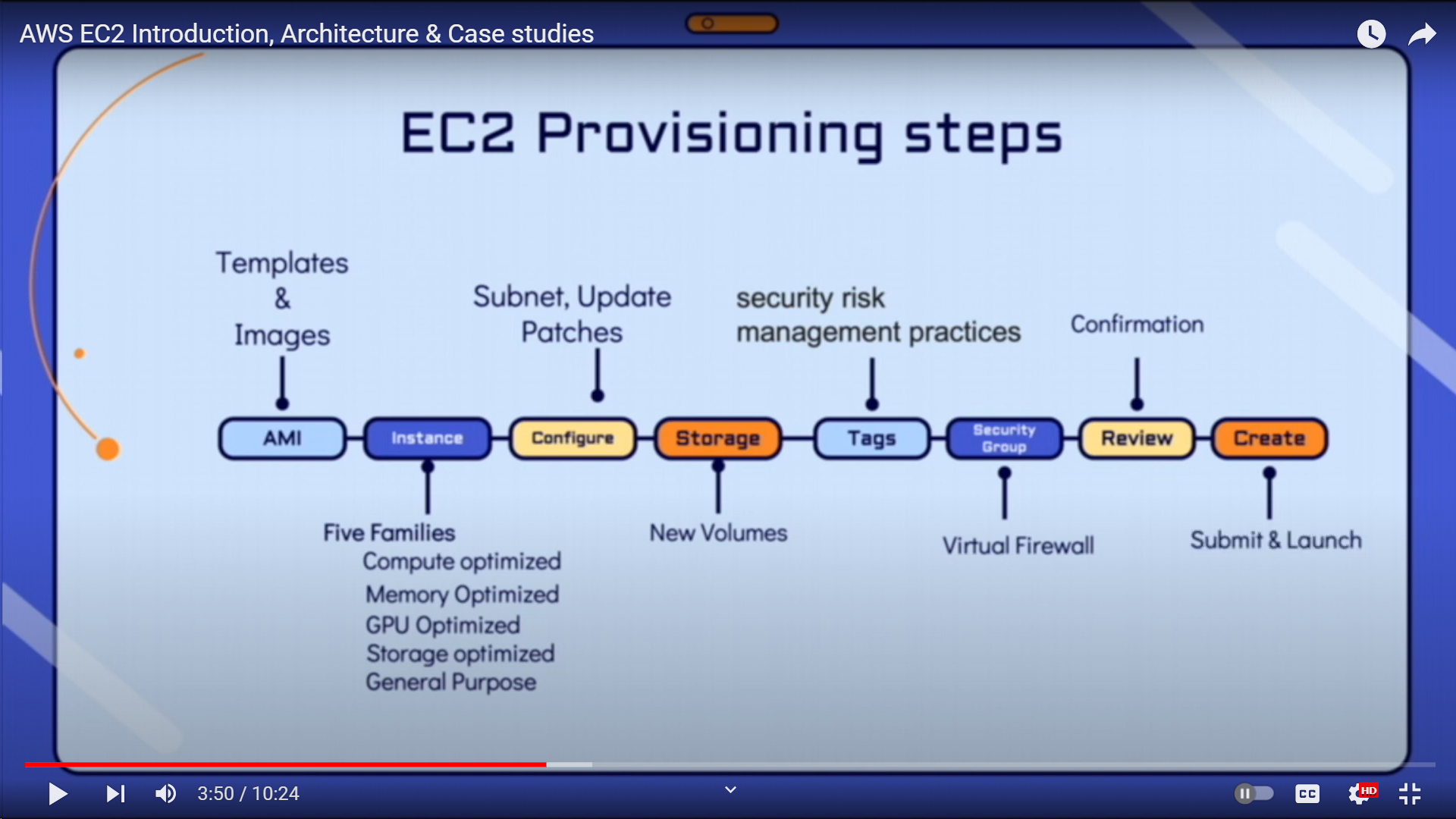


* DNS services with Amazon Route 53 – Provides DNS services to simplify domain management.
* Edge caching with Amazon CloudFront – Edge caches high-volume content to decrease the latency to customers.
* Edge security for Amazon CloudFront with AWS WAF – Filters malicious traffic, including cross site scripting (XSS) and SQL injection via customer-defined rules.
* Load balancing with Elastic Load Balancing (ELB) – Enables you to spread load across multiple Availability Zones and Amazon EC2 Auto Scaling groups for redundancy and decoupling of services.
* DDoS protection with AWS Shield – Safeguards your infrastructure against the most common network and transport layer DDoS attacks automatically.
* Firewalls with security groups – Moves security to the instance to provide a stateful, host-level firewall for both web and application servers.
* Caching with Amazon ElastiCache – Provides caching services with Redis or Memcached to remove load from the app and database, and lower latency for frequent requests.
* Managed database with Amazon Relational Database Service (Amazon RDS) – Creates a highly available, multi-AZ database architecture with six possible DB engines.
* Static storage and backups with Amazon Simple Storage Service (Amazon S3) – Enables simple HTTP-based object storage for backups and static assets like images and video.

## **4- Roles:**

* Backup and disaster recovery: Companies are leveraging EC2 as a medium for performing disaster recovery for both active and passive environments. The fact that the Amazon Elastic Compute Cloud can be turned up quickly in case of an emergency, means that businesses have access to a faster failover with minimal downtime for their applications.
* High performance computing: The need for HPC is exponentially on the rise, and EC2 provides specialized virtualized servers that provide both high performance networking and compute power. These can be used to perform a range of CPU intensive and number crunching tasks, including Big Data analytics and processing.
* Hosting environments: One of the foremost uses of EC2 is for hosting a variety of applications, software and websites on the cloud. Users are even hosting games on the cloud, turning the servers on and off when needed.
* Development and test environments: The scalable nature of EC2 means that organizations now have the ability to create and deploy large scale testing and development environments with unprecedented ease. The Amazon cloud does away with any heavy upfront investments for hardware, all the while providing as scalable solution
* Big Data Processing: EC2 is often used for big data processing and analytics workloads. Users can deploy clusters of EC2 instances to process large datasets using tools like Apache Hadoop, Apache Spark, and other data processing frameworks.

## **5- Provisioning Steps:**



* AMI (Amazon Machine Image): An Amazon Machine Image (AMI) is a template that contains the software configuration required to launch an EC2 instance. Choose an AMI that suits your needs, whether it's a standard, pre-configured, or a custom AMI. This image determines the initial software setup for your instance.
* Instance: Select an instance type based on your requirements. Each instance type offers a different combination of CPU, memory, storage, and networking capacity. Choose the instance type that best fits the needs of your application or workload.
* Storage: Configure the storage settings for your EC2 instance. By default, the instance comes with an Elastic Block Store (EBS) root volume. You can add additional volumes if needed. Select the volume type (SSD or HDD) and the size required for your instance.
* Tags: Add tags to your instance. Tags are metadata that can be used to categorize and manage AWS resources. Tags consist of a key-value pair and are useful for identifying, organizing, and tracking your instances. Proper tagging simplifies resource management.
* Security Group: Configure the security group for your instance. A security group acts as a virtual firewall, controlling inbound and outbound traffic. Define the rules that specify which traffic is allowed to reach the instance. Properly configuring security groups is crucial for the security of your instance.
* Review: Review all the configurations you have made for the instance. Ensure that everything is set up correctly before proceeding. This step allows you to double-check your settings and make any necessary changes before launching the instance.
* Create: Click on the "Launch" button to create and launch the instance. You will be prompted to select an existing key pair or create a new key pair. This key pair is necessary to securely connect to the instance using SSH (for Linux instances) or RDP (for Windows instances).

## **6- Advantages and Disadvantages:**

### **a. Advantages**

|  |  |
| --- | --- |
| **Reliability** | Amazon EC2 provides 99.9% availability in each Amazon EC2 region. The services are highly dependable, and instances can be easily and quickly replaced. |
| **Safety** | Amazon collaborates with Amazon VPC to provide secure networking and compute resources.  The compute instances are housed in a VPC (Virtual Private Cloud) and are assigned an IP address range.  => This function assists the user in determining which instances are exposed to the internet and which are kept private. |
| **Adaptability** | You can choose from a variety of instance types, software packages, instance storage, and operating systems on EC2. EC2 allows us to specify the memory, CPU, and boot partition size that is best for the operating system and application. |
| **Cost-cutting** | EC2 is inexpensive because it allows the user to choose plans based on their needs. This will allow the user to save money and make the best use of the available resources. EC2 reaps the benefits of Amazon’s scale by charging a very low fee in comparison to the services provided. |
| **Full-Service Computing Solution** | Amazon RDS, S3, Dynamo DB, and Amazon SQS are all compatible with EC2. This provides an all-in-one computing, processing, and storage solution. |
| **Elastic Web-Scale Computing** | Enterprises can quickly increase or decrease capacity. They can launch thousands of server instances at the same time. Furthermore, all server instances are managed by web service APIs, which can scale the servers up and down based on the needs. |

### **b. Disadvantages:**

* When you migrate to the cloud, Amazon Web Services may encounter some common cloud computing issues like downtime, limited control, and backup protection.
* AWS imposes resource limits by default, which vary by region. You can only launch a certain number of instances per area.
* Hardware-level changes occur in your application, which may result in poor performance and usage of your applications.
* Security Constraints: Because security is one of the most important features, AWS restricts some of its features that cannot be changed at all.

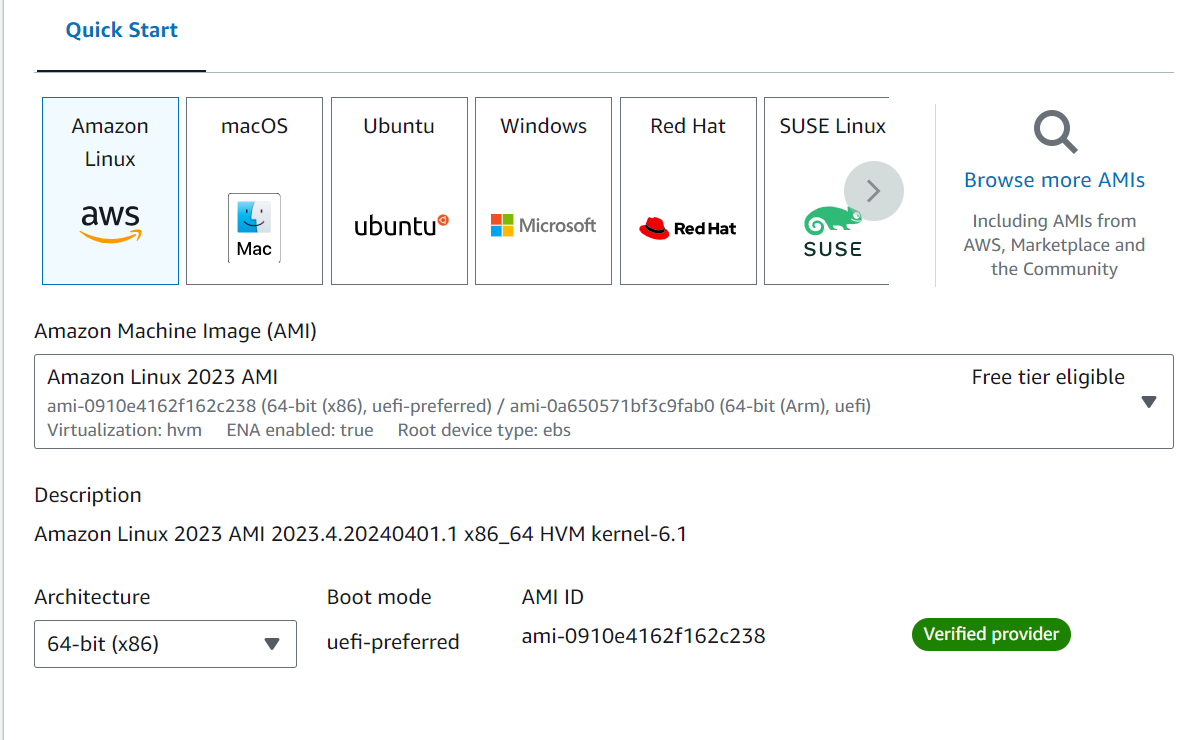
# **IV- Scenario:**

In this project, we will develop a simple application named “Hello World”. As it’s our first time using this Amazon EC2 instance, we will use a simple example to test if it works properly on Amazon EC2. This project is a starting point for us to learn how to use EC2 for hosting. We hope to get comfortable with EC2's features and be ready for more complicated projects later on.

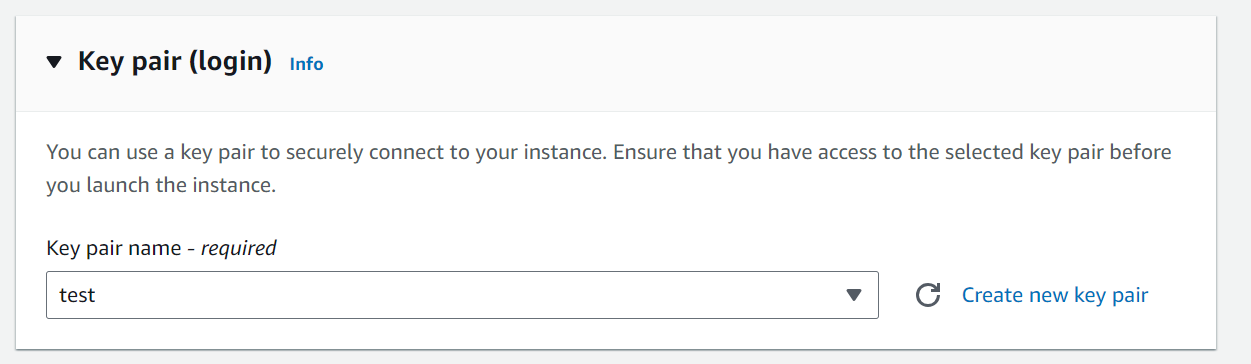
# **V- Demo:**

First, we select EC2 from the Services Menu. Then we select Launch Instance and Create an EC2 Instance.

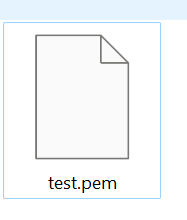
For the appropriate operating system, for this task I will use Amazon Linux.



Then, we’ll create a Key Pair.

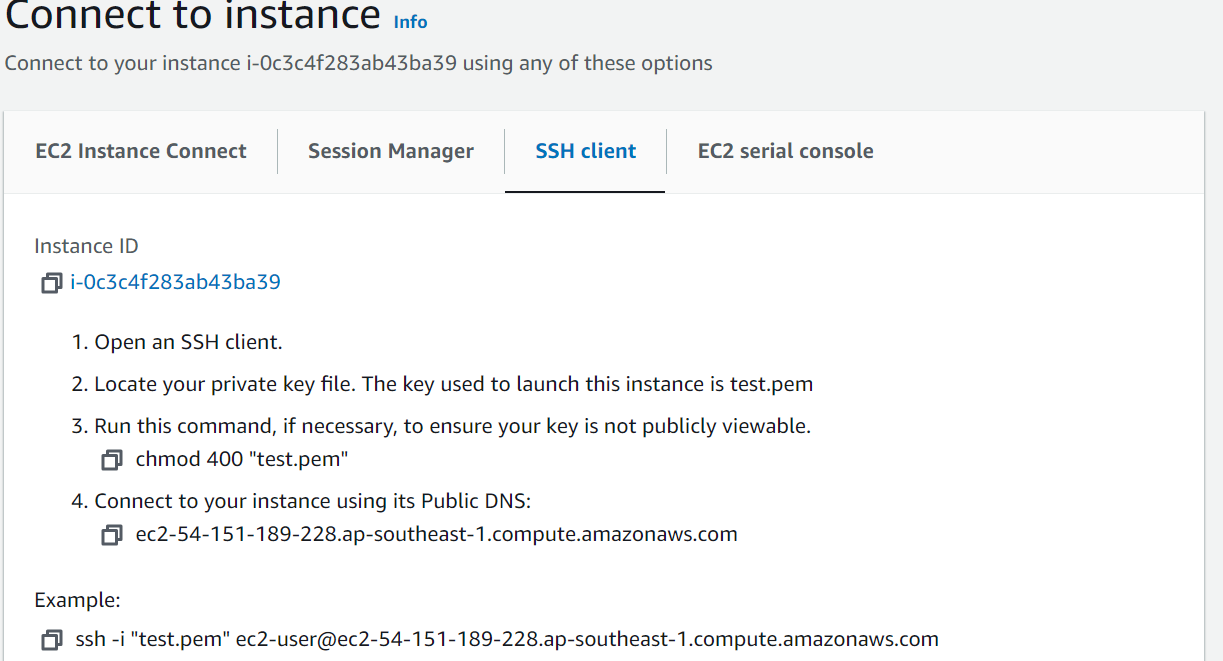


This pem/ppk file will be the specific key whose presence will enable you to login from anywhere



We have successfully created & launched the instance.

After initializing the EC2 Instance, we will have to connect it to the VM.



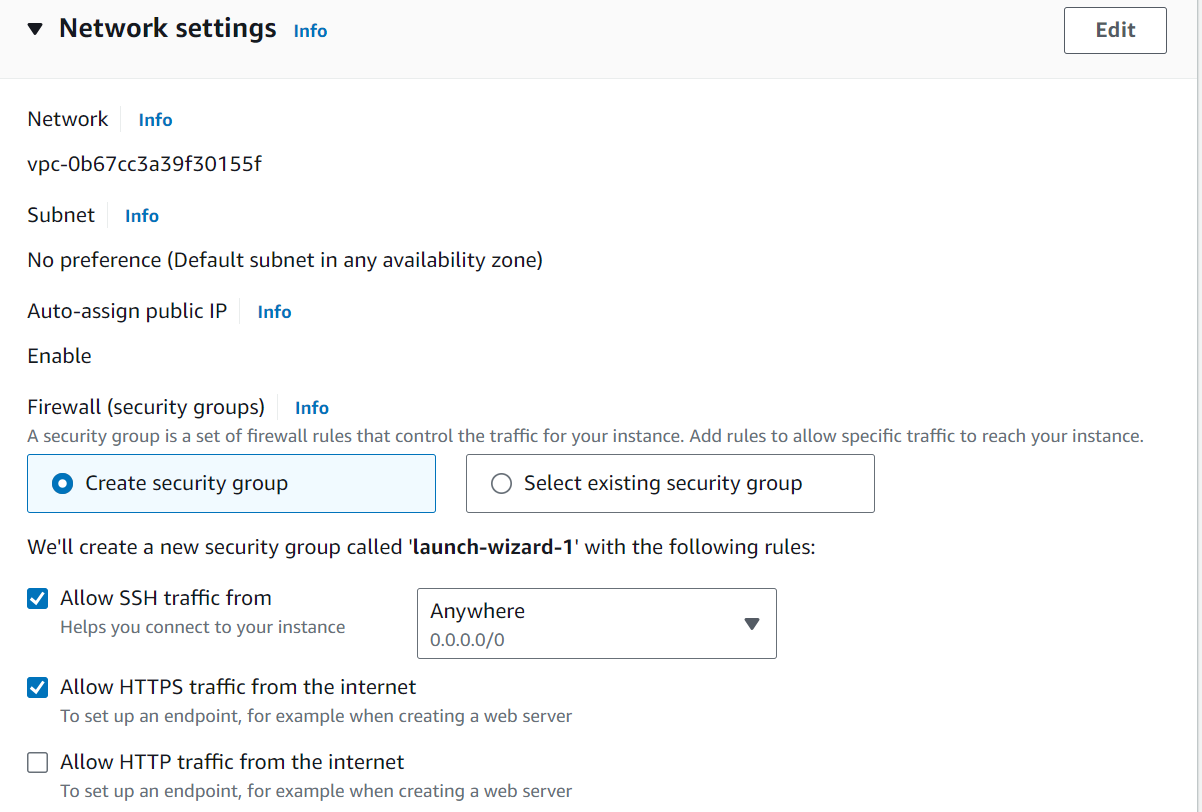
For this we have three methods:

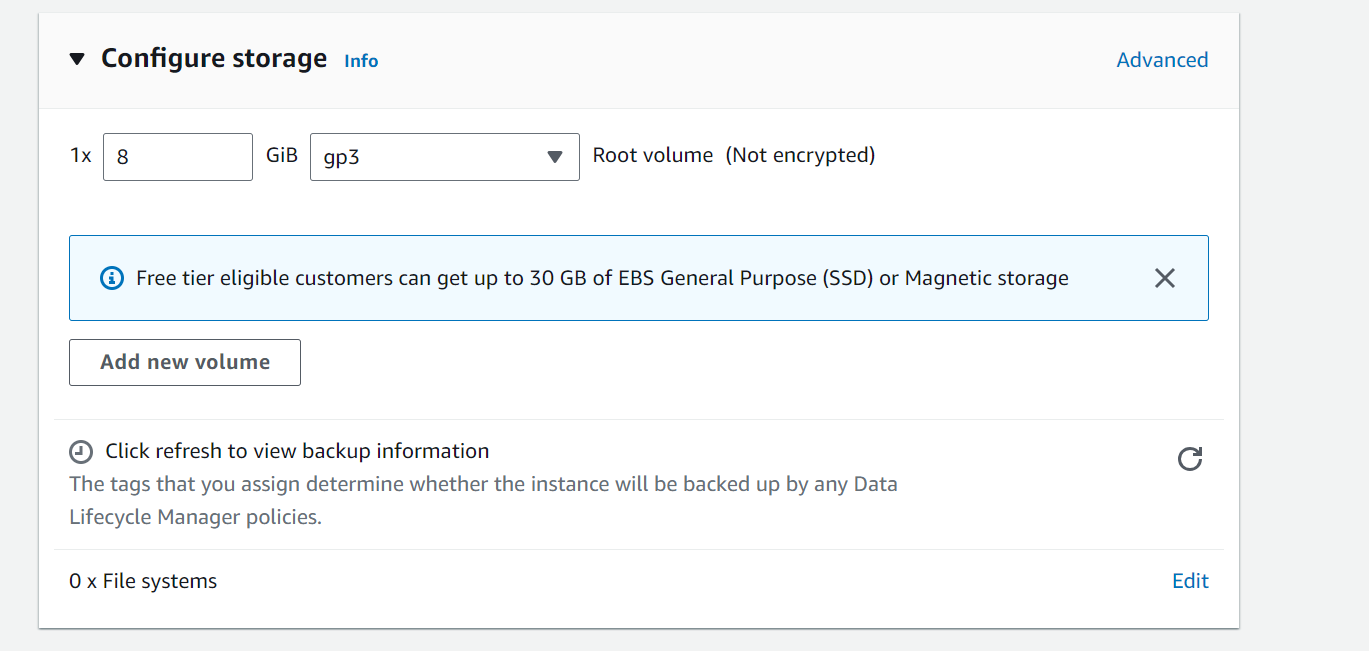
1. We will use the PuTTY to convert .pem file(The key which we downloaded while creating the instance) to .ppk and launch it on our local System/machine

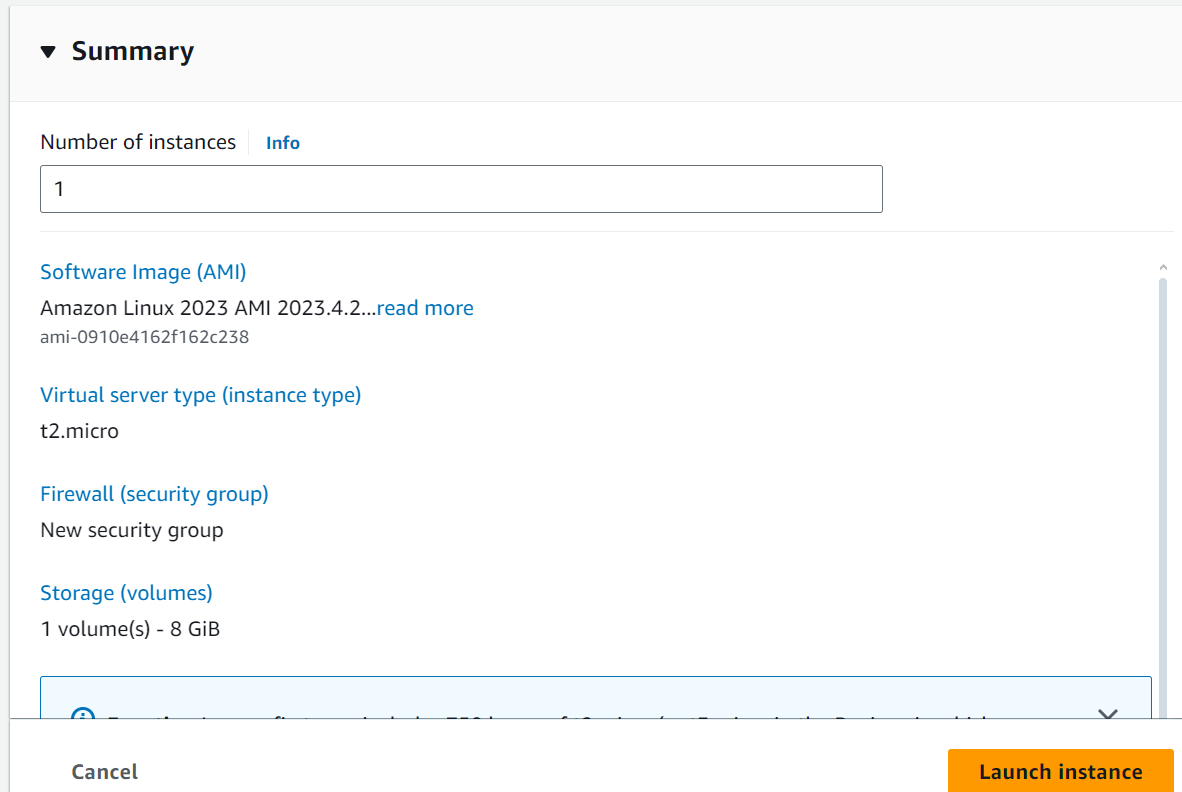
2. If we have downloaded the .ppk file then we can directly initialize the VM on our local Machine/System.

3. We will directly connect to the VM through the AWS platform

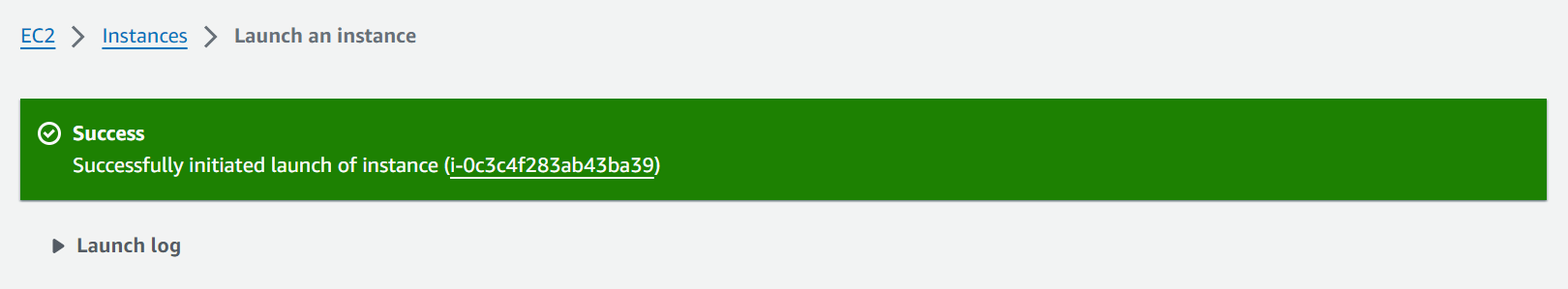
I have used the 3rd method.

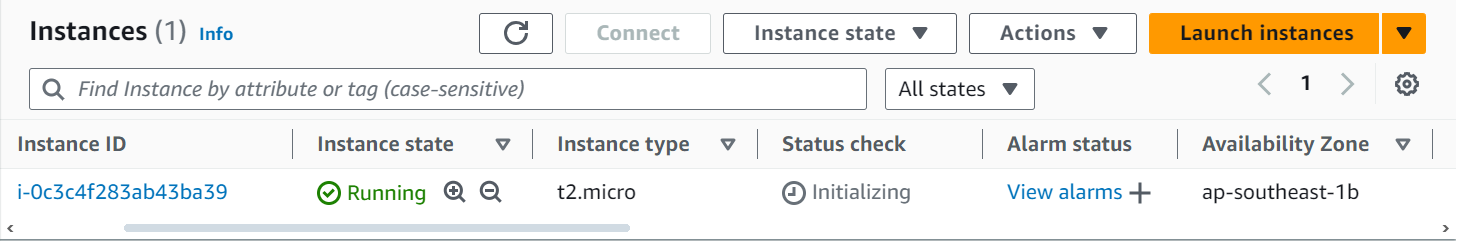


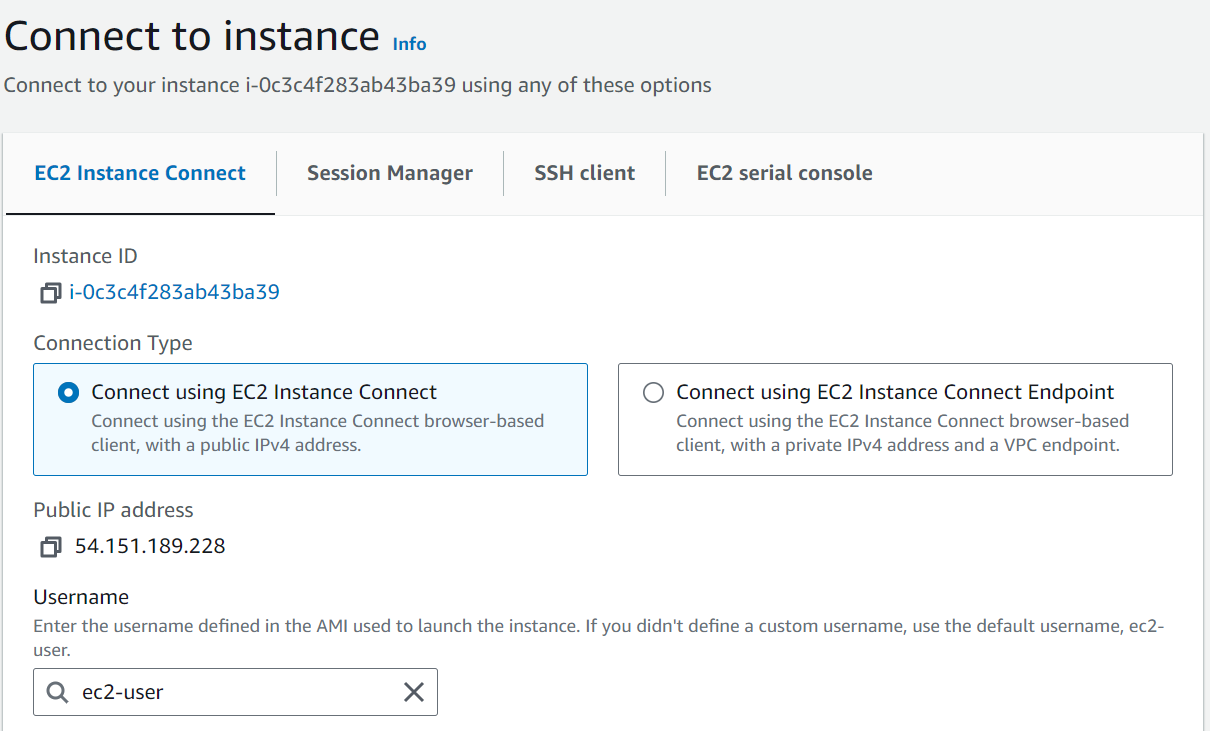




Finally, we’ll launch the instance:







Here is the terminal of Linux after we connect successfully.



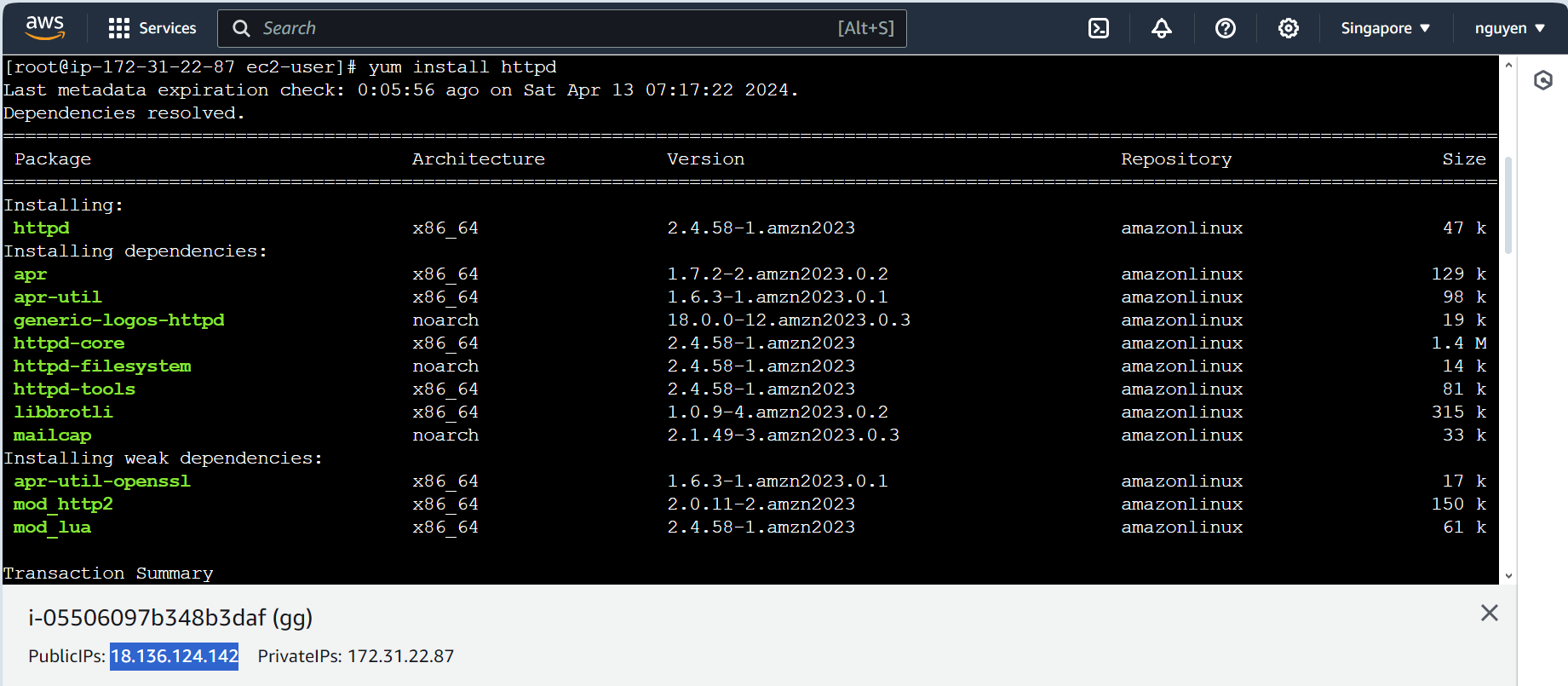
After connecting with the Instance, we will run the following commands on the console:

1. sudo su -: switch to the root user and start a new shell session with root privileges, inheriting the root user's environment.

2. yum update -y: updates all installed packages on the system to their latest available versions without asking for confirmation.

3. yum install -y httpd: installs the Apache HTTP Server (httpd) without asking for confirmation.

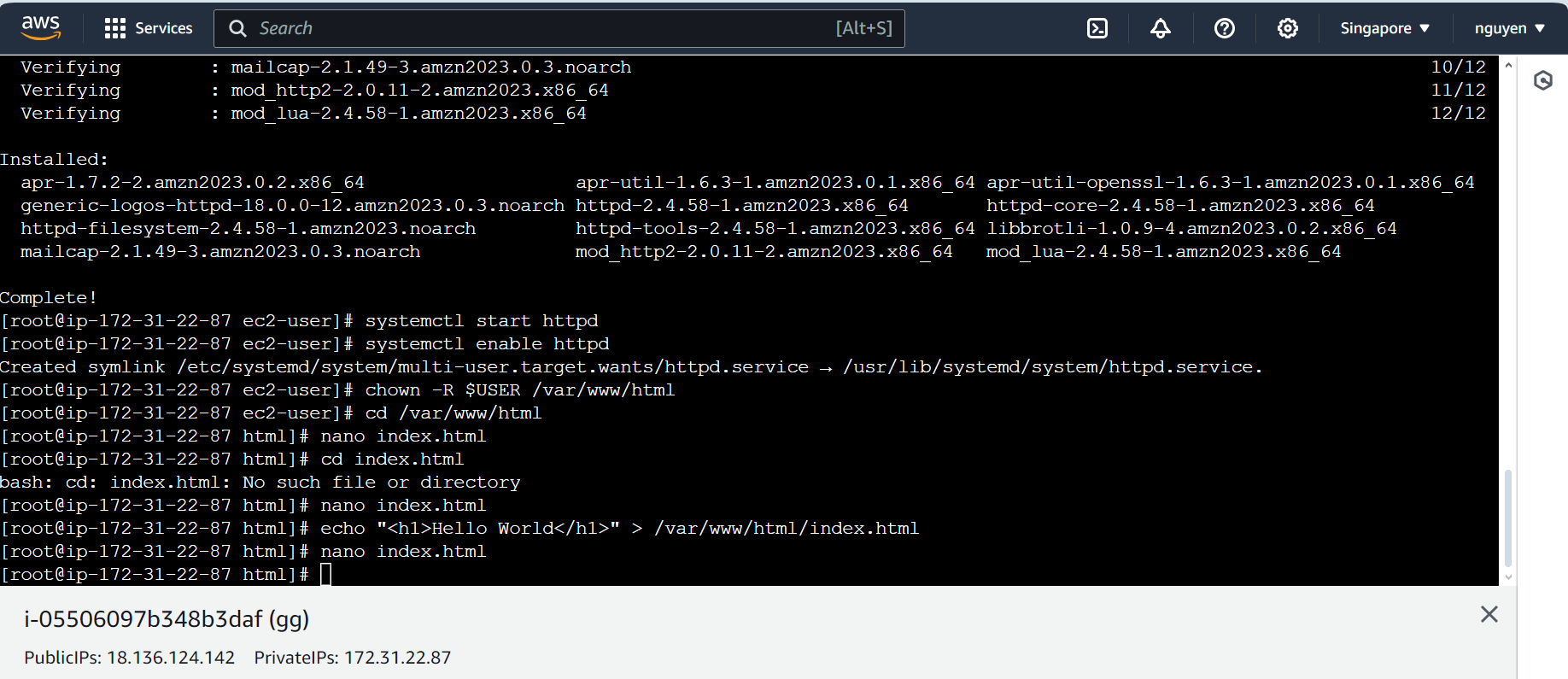
4. systemctl status httpd: check if the httpd has any problems after the installation.



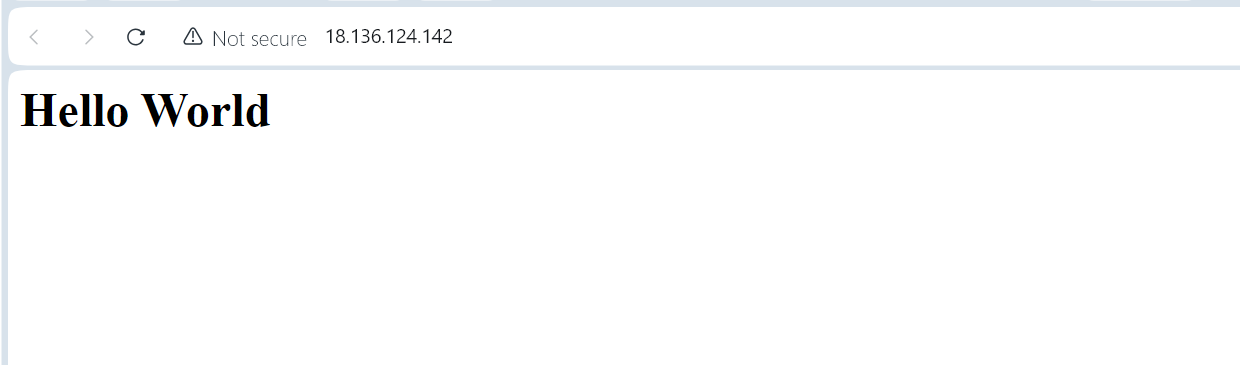
**Edit the Inbound Rules:**

Check the status of httpd and then enable & start httpd using the following commands

* systemctl status httpd: check if the httpd is working or not.
* systemctl enable httpd: used to configure the Apache HTTP Server service to start automatically at system boot.
* systemctl start httpd: is a command used to start the Apache HTTP Server service. After running this command, the server will begin accepting and processing HTTP requests.



Now open the public ipv4 address allocated to the EC2 instance we created in the new tab. We will be able to see the “Hello World” web application:



# **VI- Conclusion:**

In this project, we have learned about how to develop the web app by using the Amazon EC2 application. Our target for this project is displaying a simple web application to test for our future projects. Amazon EC2 is a really suitable environment for the developers who want to test their project. Through this project, we gained hands-on experience deploying a web application on EC2, setting the stage for more complex deployments. This project serves as a valuable starting point for exploring the capabilities and advantages of using Amazon EC2 for hosting web applications.

# **VII- References:**

<https://www.geeksforgeeks.org/what-is-elastic-compute-cloud-ec2/>

<https://www.awsforbusiness.com/amazon-ec2-use-cases/#google_vignette>

<https://www.tutorialspoint.com/amazon_web_services/amazon_web_services_basic_architecture.htm>

# 