Experiment 10

Aim:

To create and work with a virtual machine on cloud (AWS).

Theory:

This simulation provides you with a basic overview of launching, resizing, managing, and monitoring an Amazon Elastic Compute Cloud (Amazon EC2) instance.

Amazon EC2 is a web service that provides resizable compute capacity in the cloud.

It's designed to make web-scale cloud computing intuitive and straight forward to use.

Amazon EC2 gives you fast access to new server instances, and you can quickly scale capacity both up and down as your computing requirements change.
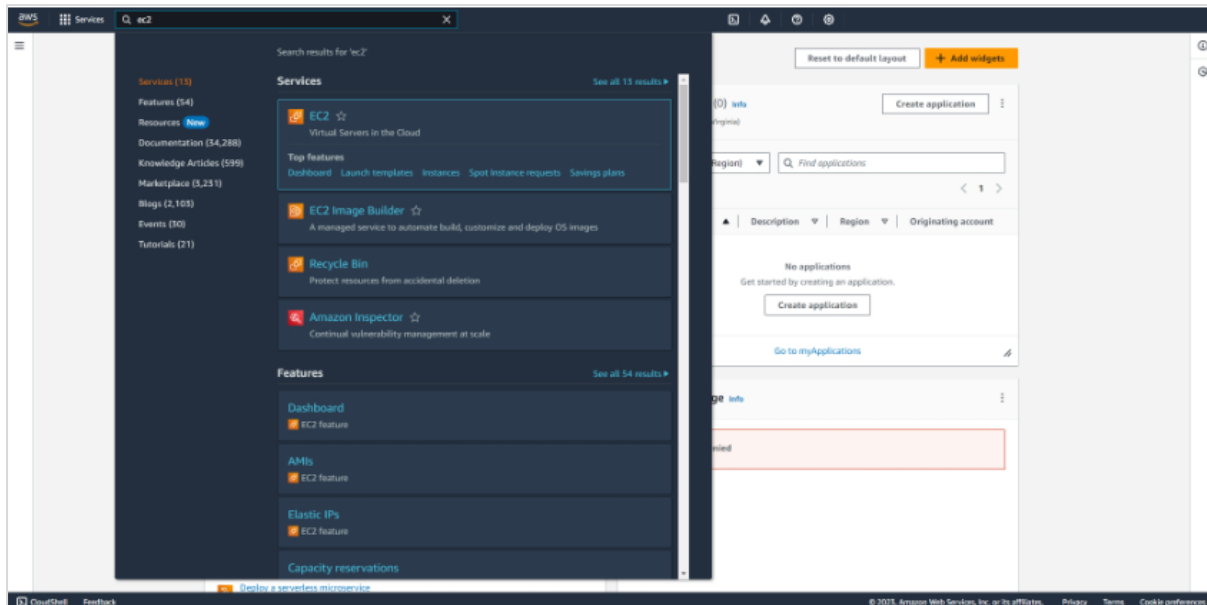
Task 1: Launching your EC2 instance

In this task, you launch an EC2 instance with termination protection. Termination protection prevents you from accidentally terminating an EC2 instance. You also deploy your instance with a user data script to deploy a simple web server.
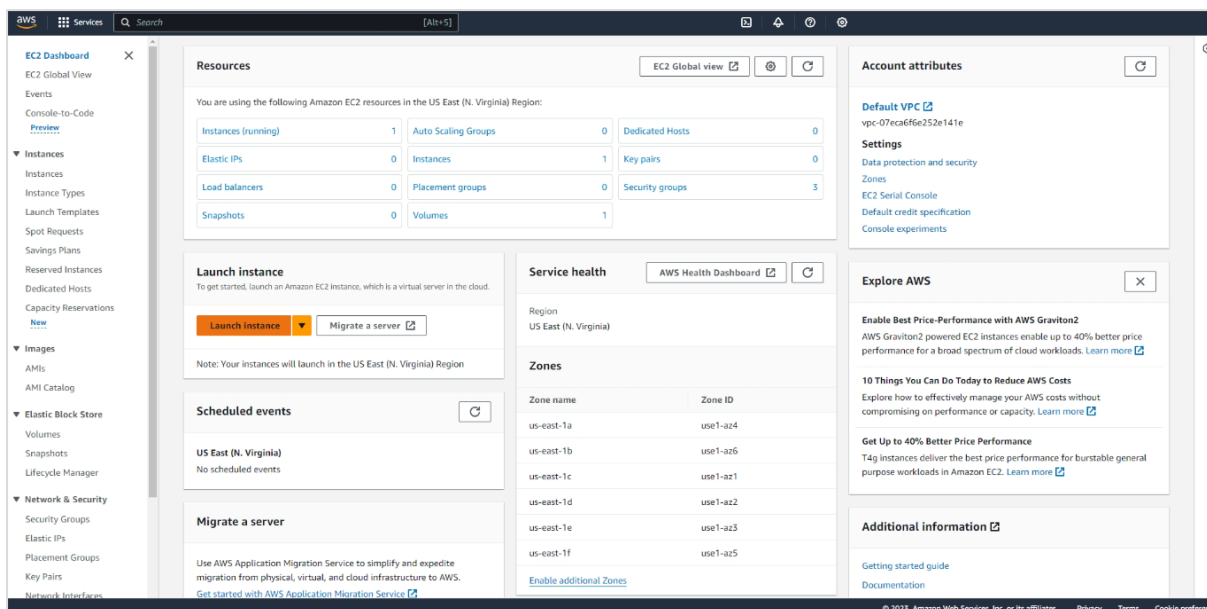
1. In the AWS Management Console in the **Search**, enter **EC2** and choose **Enter**.
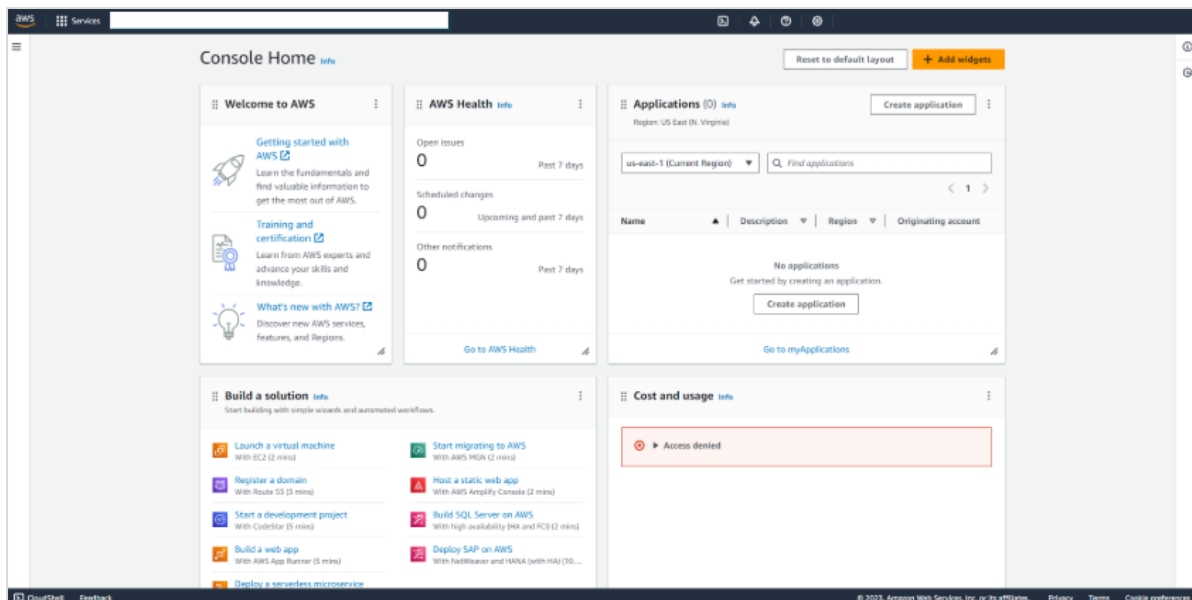
2. From the search results, choose **EC2**.

3. In the **Launch instance** section, choose **Launch instance**.

STEP 1: NAME YOUR EC2 INSTANCE

Using tags, you can categorize your AWS resources in different ways (for example, by purpose, owner, or environment). This categorization is useful when you have many resources of the same type. You can quickly identify a specific resource based on the tags that you have assigned to it. Each tag consists of a key and a value, both of which you define.

When you name your instance, AWS creates a key-value pair. The key for this pair is **Name**, and the value is the name that you enter for your EC2 instance.

4. In the **Name and tags** pane, in the **Name** text box, enter <u>Web-Server</u> then choose **Enter**.

5. Choose the **Add additional tags** link.

6. From the **Resource types** dropdown list, **Instances** is selected by default. Leave Instances selected and select **Volumes**.
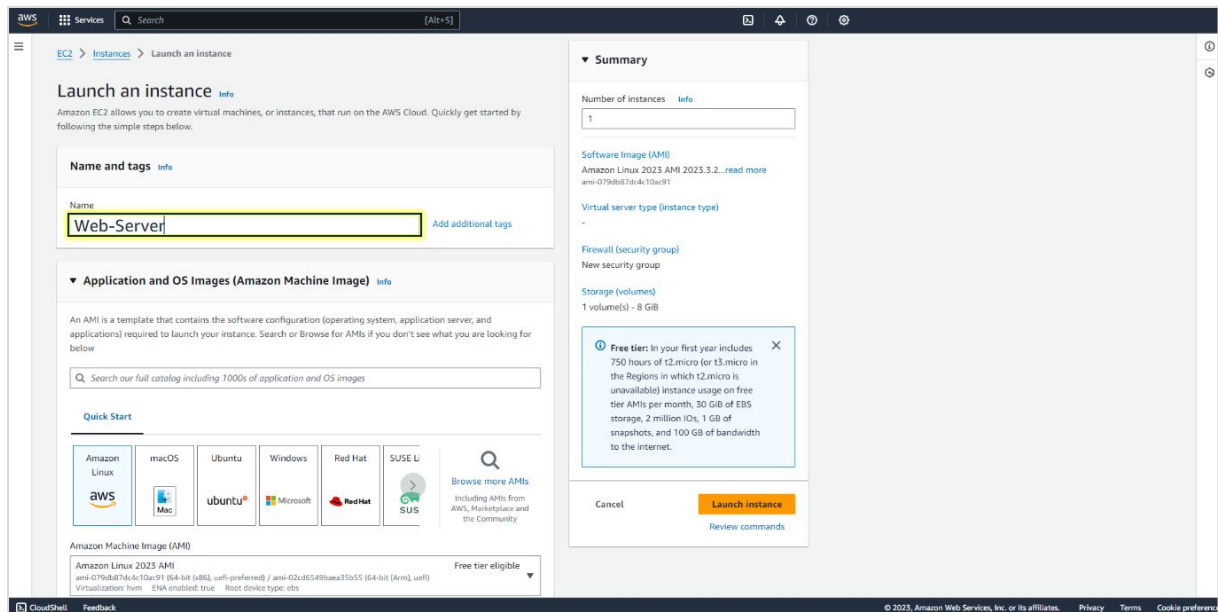

STEP 2: CHOOSE AN AMI

An Amazon Machine Image (AMI) provides the information required to launch an instance, which is a virtual server in the cloud. An AMI includes the following:

- Launch permissions that control which AWS accounts can use the AMI to launch instances

7. Locate the **Application and OS Images (Amazon Machine Image)** section. It's below the **Name and tags** section. In the search box, enter Windows Server 2019 Base and choose **Enter**.
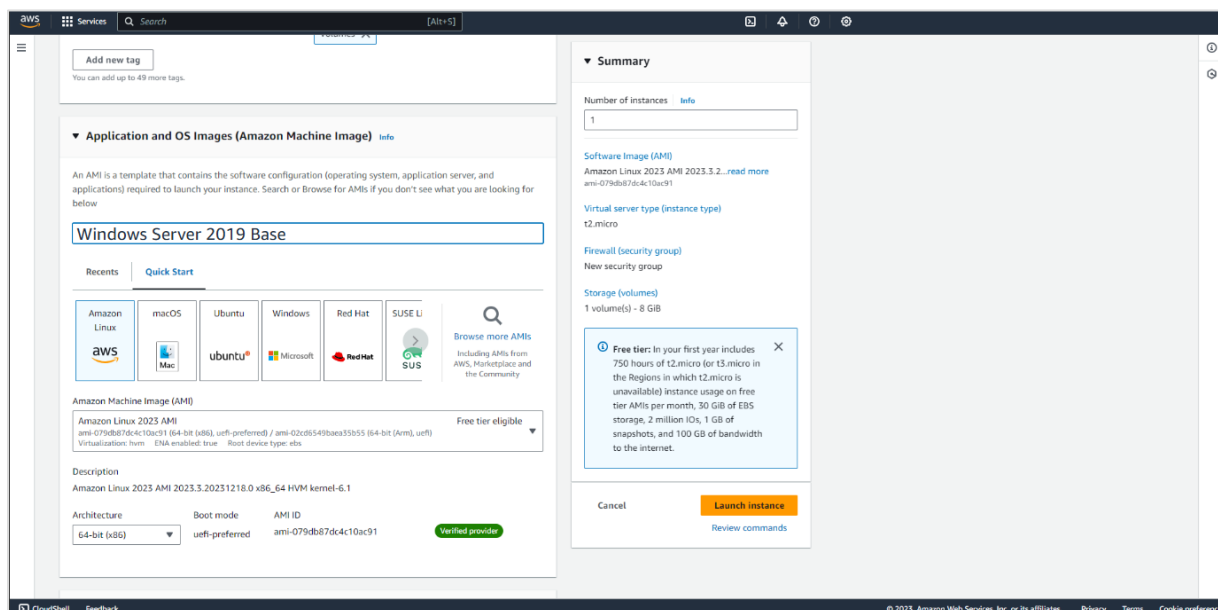
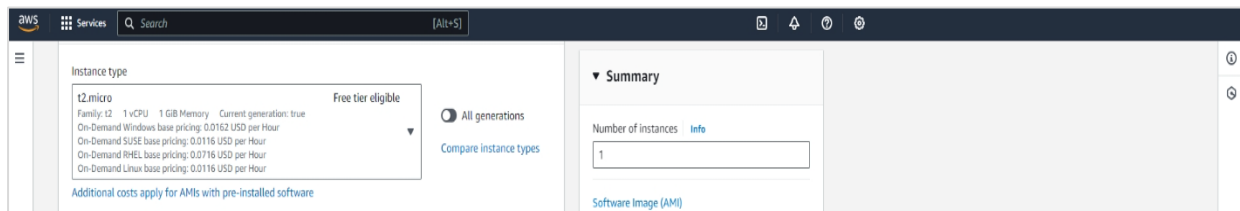7. Next to **Microsoft** Windows Server 2019 Base, choose **Select**.
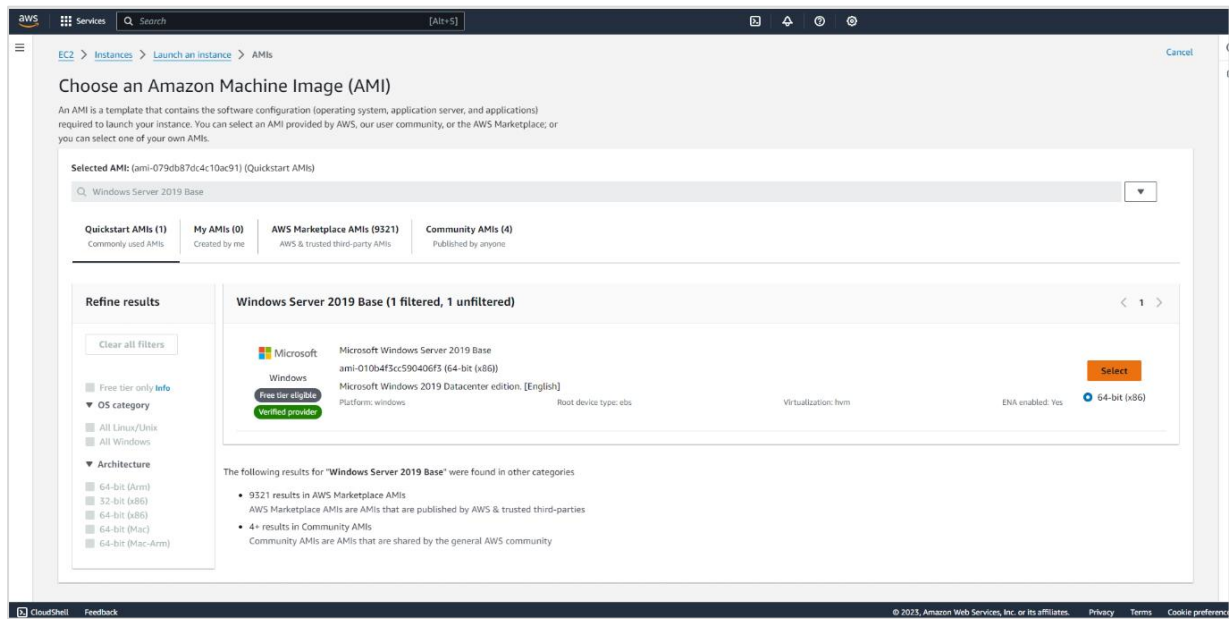
STEP 3: CHOOSE AN INSTANCE TYPE

In this step, you choose a **t2.micro** instance. This instance type has one virtual CPU and 1 GiB of memory.
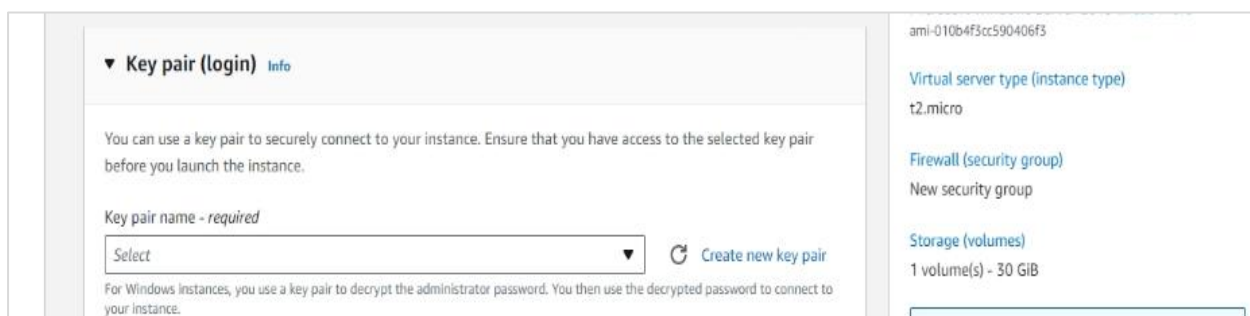
9. In the **Instance type** section, keep the default instance type, **t2.micro**.

STEP 4: CONFIGURE A KEY PAIR

In this simulation, you don't connect to your instance using an SSH key, so you don't need to configure a key pair.

10. In the **Key pair (login)** section, from the **Key pair name - required** dropdown list, choose **Proceed without a key pair (not recommended).**

STEP 5: CONFIGURE THE NETWORK SETTINGS

You use this pane to configure networking settings.

The virtual private cloud (VPC) indicates which VPC you want to launch the instance into. You can have multiple VPCs, including different ones for development, testing, and production.

> 11. In the **Network settings** section, choose **Edit**.
>
> 12. From the **VPC - required** dropdown list, choose **Lab VPC**.
>
> 13. For **Firewall** (**security groups**), choose **Select existing security group**.
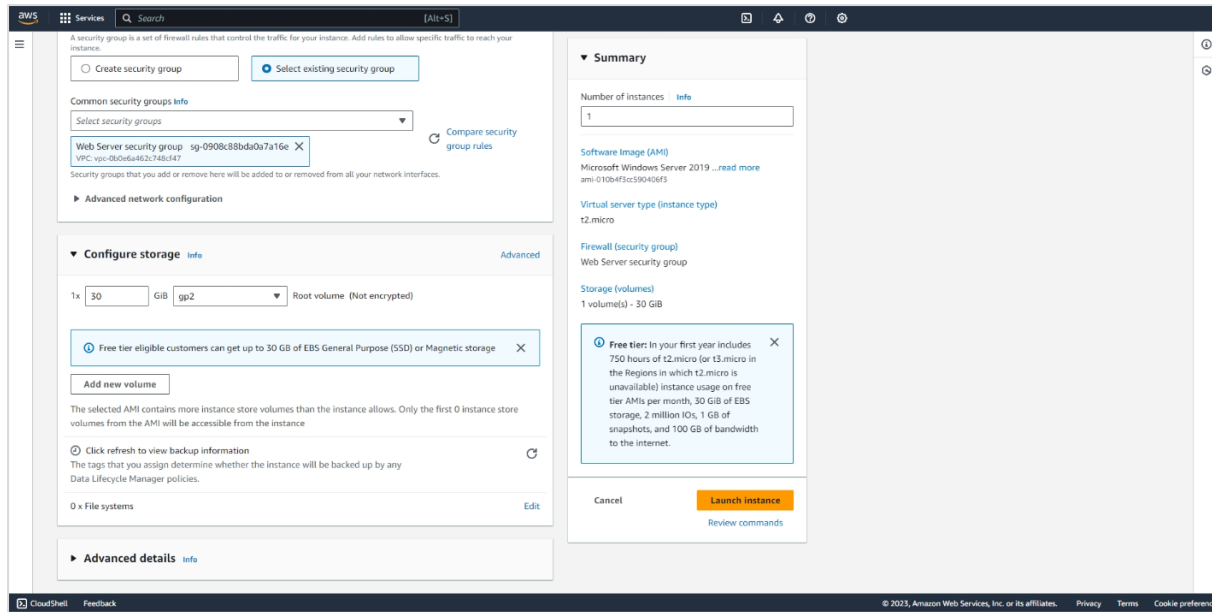>
> 14. From **Common security groups**, choose **Web Server security group**.
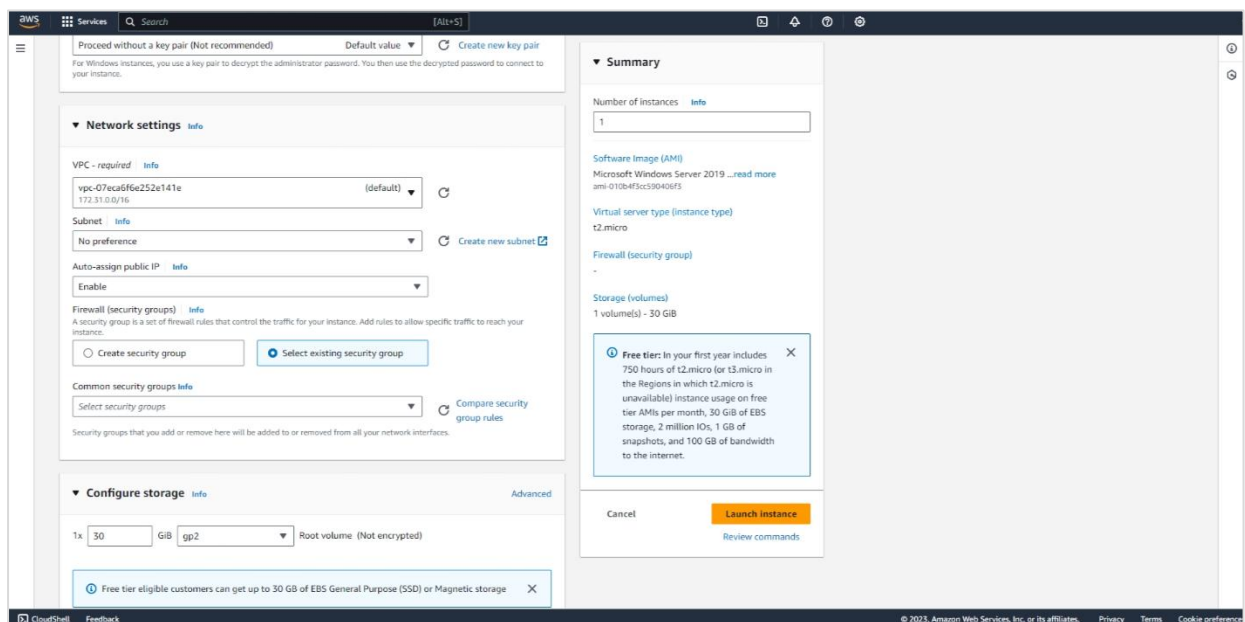
STEP 6: ADD STORAGE

Amazon EC2 stores data on a network-attached virtual disk called Amazon Elastic Block Store (Amazon EBS). You launch the EC2 instance using a default 30 GiB disk volume. This is your root volume (also known as a boot volume).
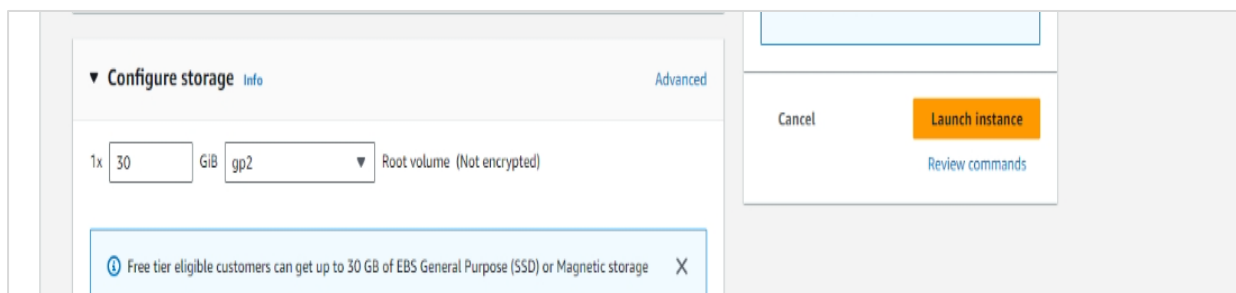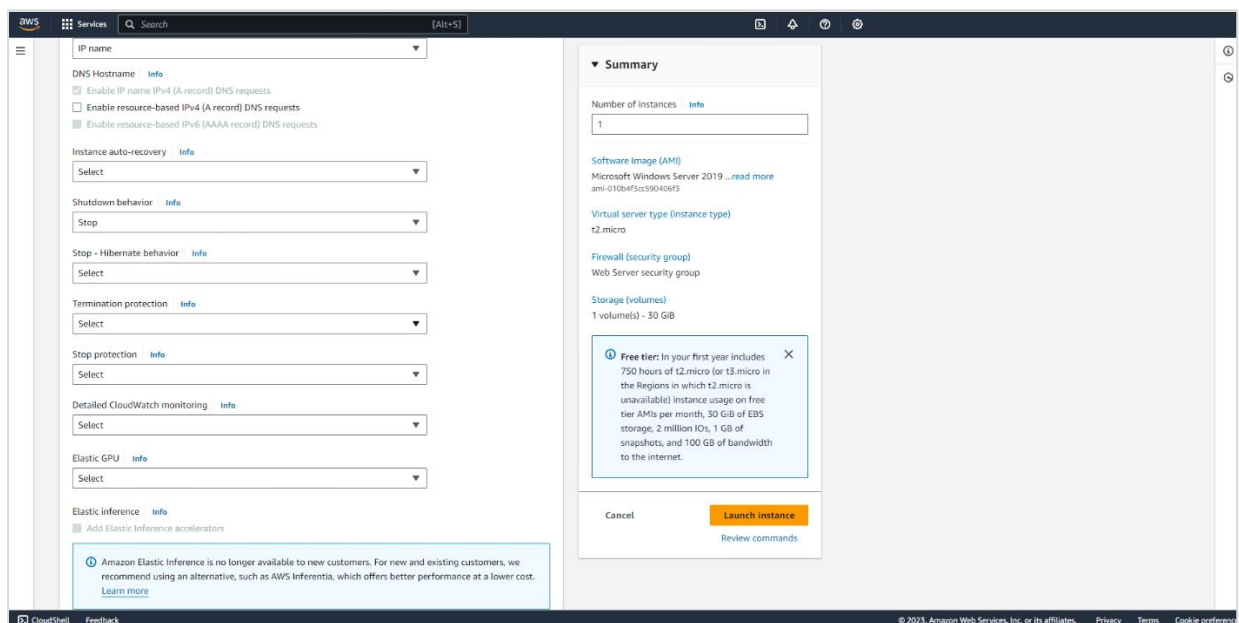
STEP 7: CONFIGURE ADVANCED DETAILS

15. Expand the **Advanced details** section.

16. For **IAM instance profile**, choose the role that begins with **LabStack** in the name.

17. From the **Termination protection** dropdown list, choose **Enable**.

18. Copy the following commands, and choose the User data text box. Then, choose Paste.

```
<powershell>

# Installing web server

Install-WindowsFeature -name Web-Server -IncludeManagementTools
```

```powershell
# Getting website code
wget    https://us-east-1-tcprod.s3.amazonaws.com/courses/CUR-TF-
100-

EDCOMP/v1.0.4.prod-ef70397c/01-Lab-ec2/scripts/code.zip -outfile

"C:\Users\Administrator\Downloads\code.zip"
# Unzipping website code
Add-Type -AssemblyName System.IO.Compression.FileSystem

function Unzip

{

    param([string]$zipfile, [string]$outpath)


[System.IO.Compression.ZipFile]::ExtractToDirectory($zipfile,
$outpath)

}

Unzip "C:\Users\Administrator\Downloads\code.zip" "C:\inetpub\"

# Setting Administrator password

$Secure_String_Pwd  =  ConvertTo-SecureString  "P@ssW0rD!"  -
AsPlainText -Force

$UserAccount = Get-LocalUser -Name "Administrator"

$UserAccount | Set-LocalUser -Password $Secure_String_Pwd

</powershell>
```
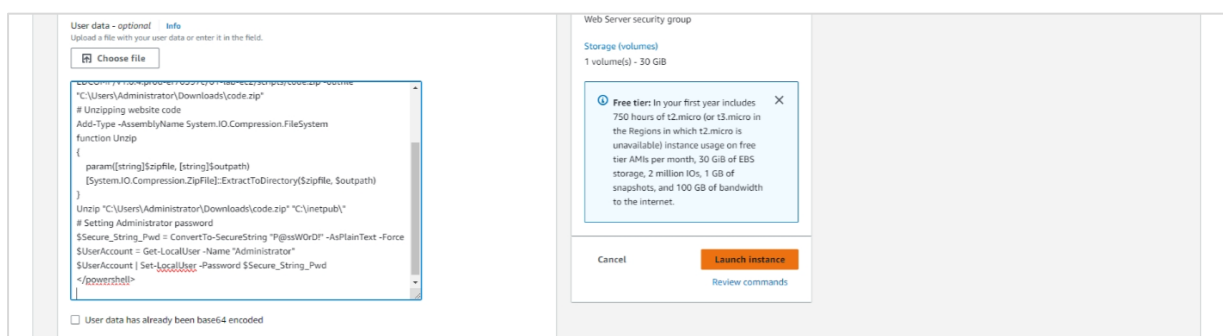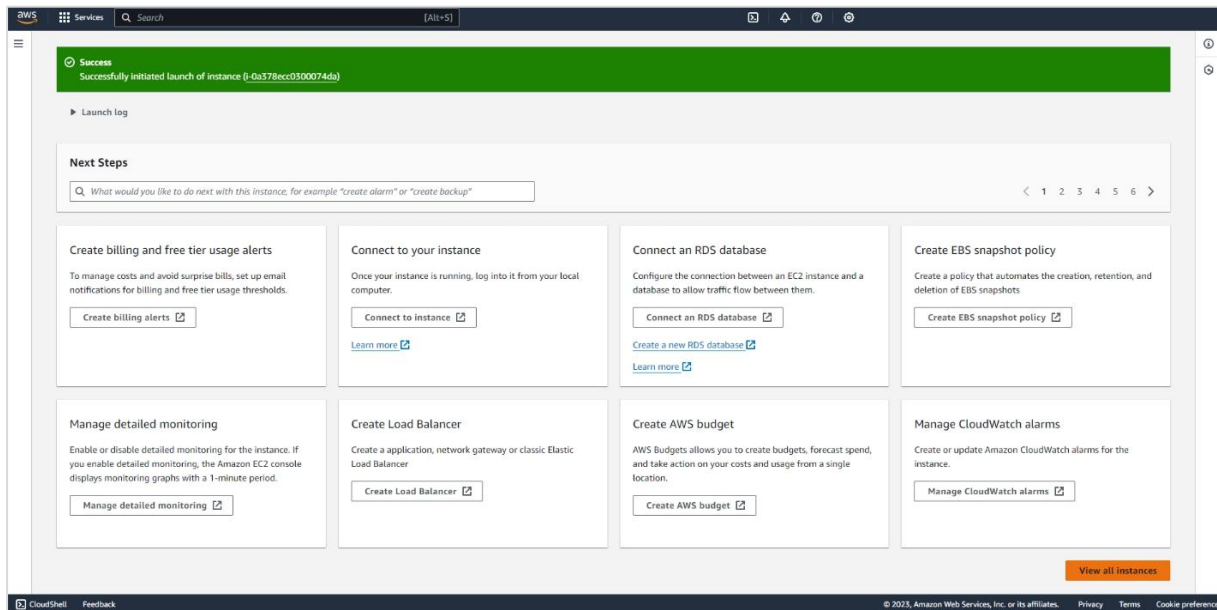

STEP 8: LAUNCH AN EC2 INSTANCE

Now that you configured your EC2 instance settings, it's time to launch your instance.

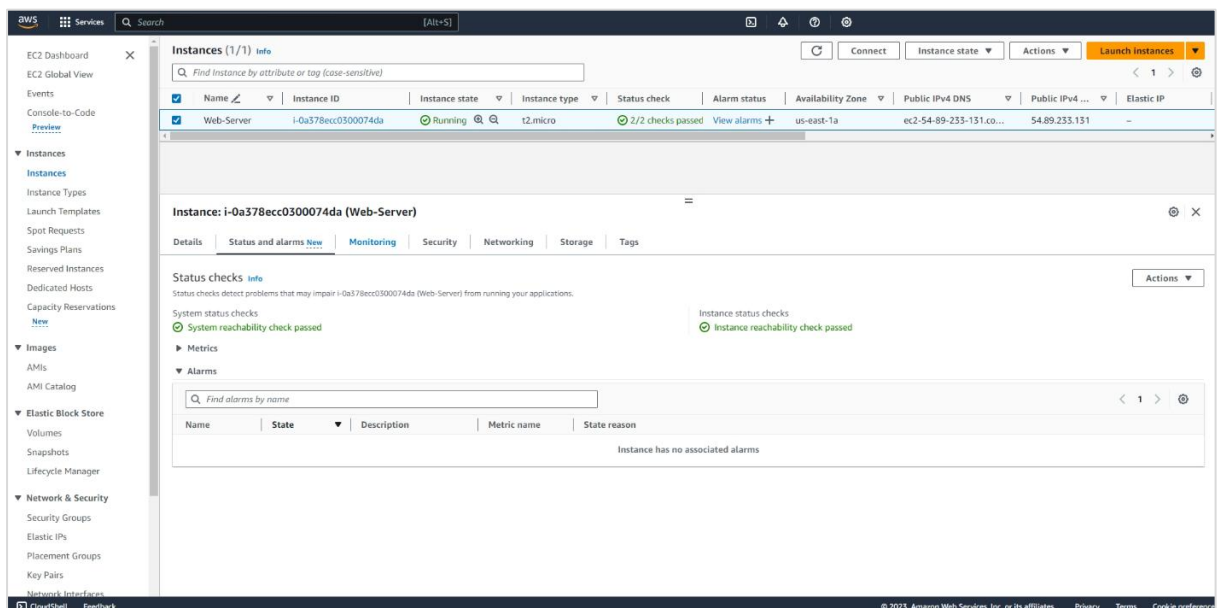19. In the **Summary** section, choose **Launch instance**.
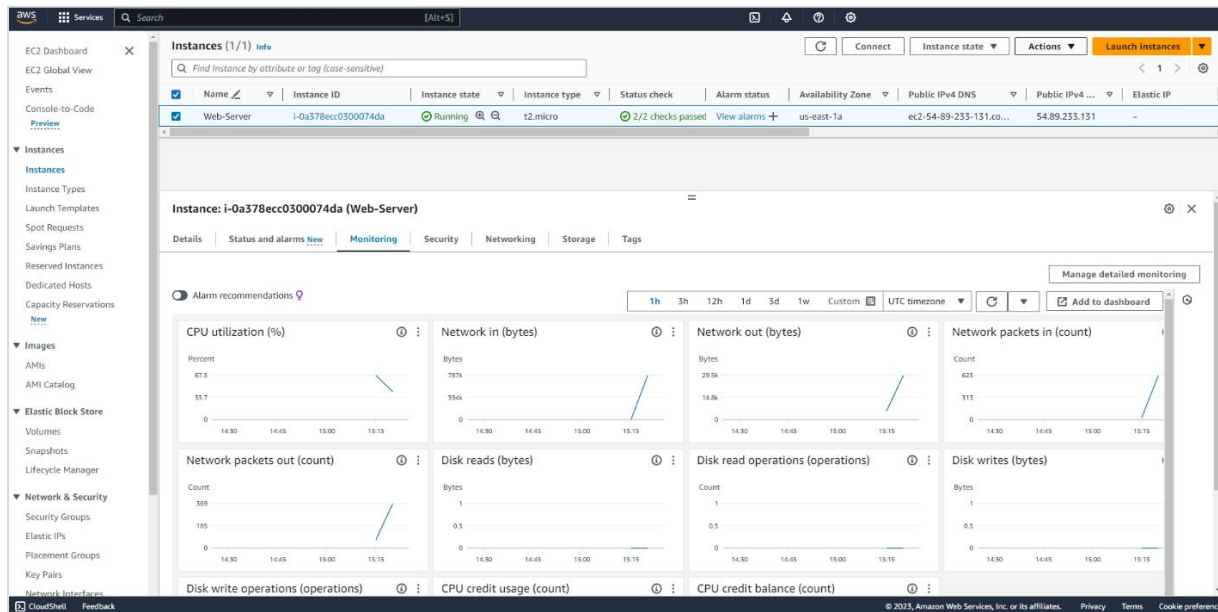
20. Choose **View all instances**.

Task 2: Monitor your instance

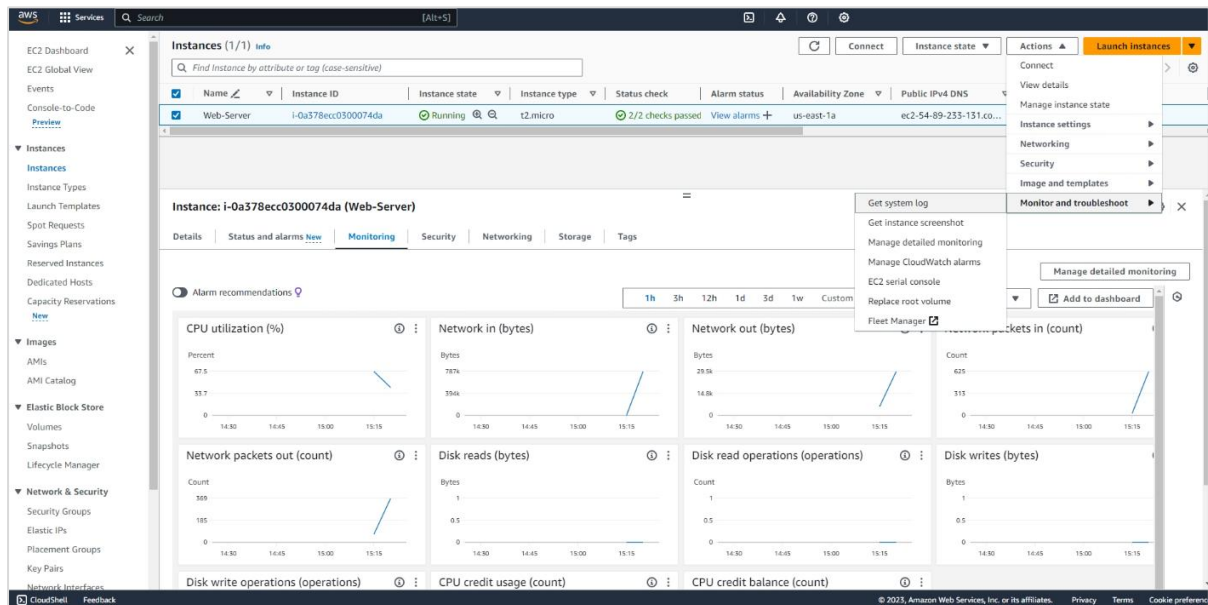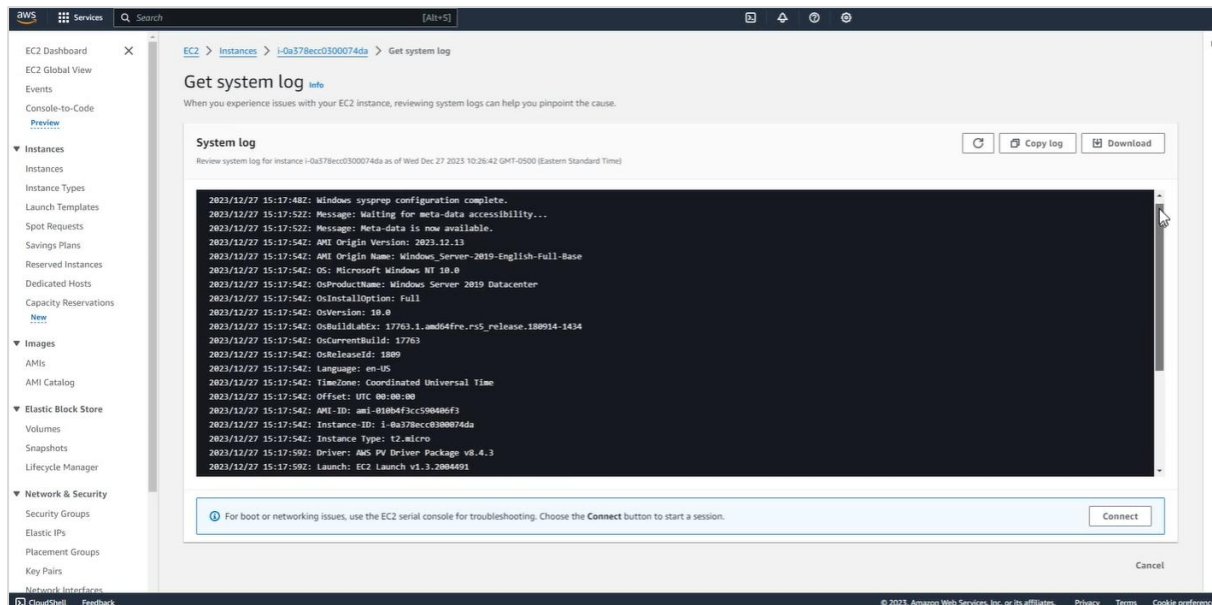24. Choose the **Status and alarms** tab. Review the information that's available to you.
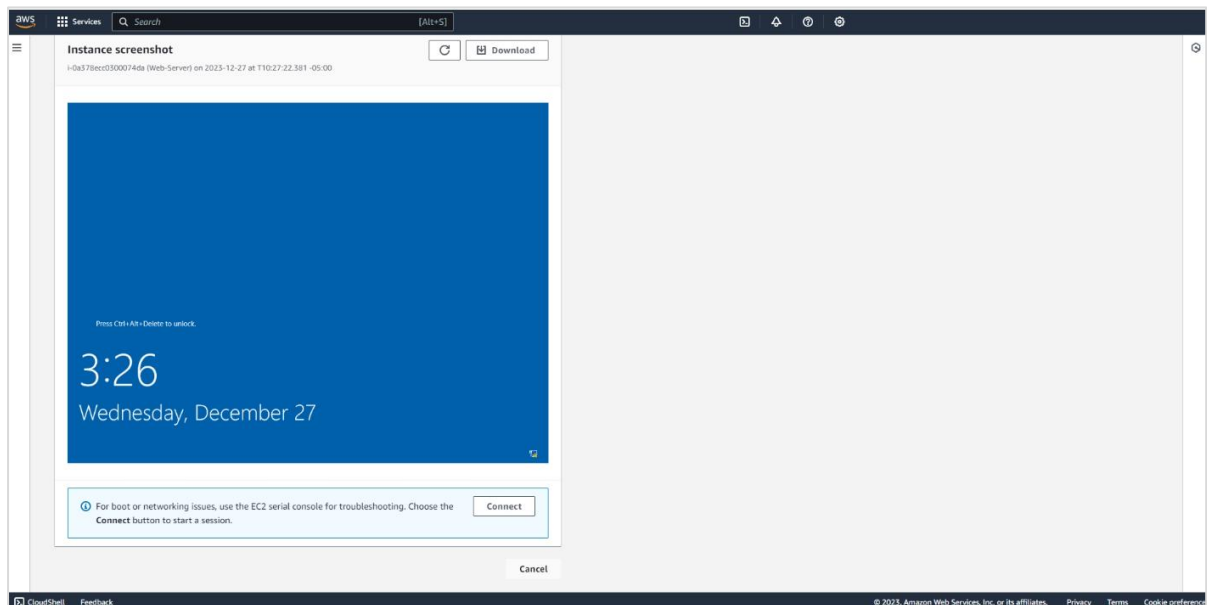
25. Choose the **Monitoring** tab.

26. At the top of the page, choose the **Actions** dropdown list. Choose **Monitor and troubleshoot** < **Get system log**.

27. In the **System** log, **review** the messages in the output.

28. To return to the Amazon EC2 dashboard, choose **Cancel**.

Conclusion- In conclusion, the objective of creating and working with a virtual machine on the AWS cloud platform has been successfully achieved. Through this process, the principles of cloud computing, virtualization, and on-demand resource provisioning were effectively demonstrated. The deployment of an AWS EC2 instance provided hands-on experience in configuring, managing, and utilizing scalable virtual environments.