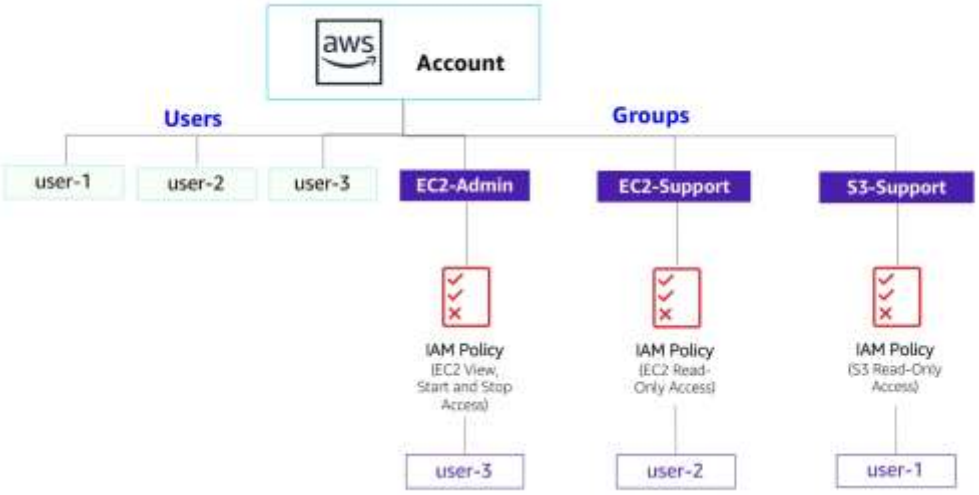
Experiment – 3(a)

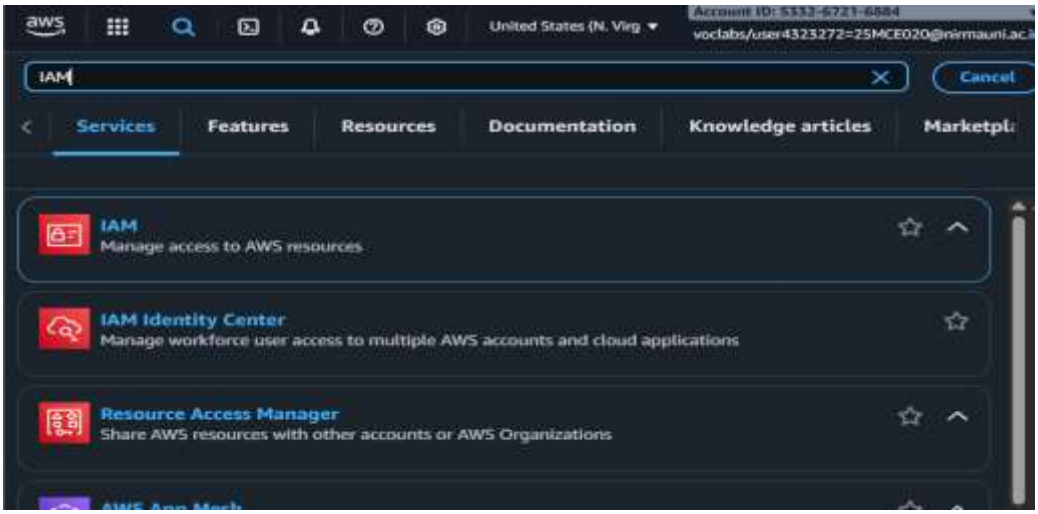
❖ **Aim:** Explore pre-created IAM Users and Groups, Inspecting IAM policies as applied to the pre-created groups, Following a real-world scenario, adding users to groups with specific capabilities enabled, Locating and using the IAM sign-in URL, and Experimenting with the effects of policies on service access.



Task 1: Explore the Users and Groups

In this task, you will explore the Users and Groups that have already been created for you in IAM.

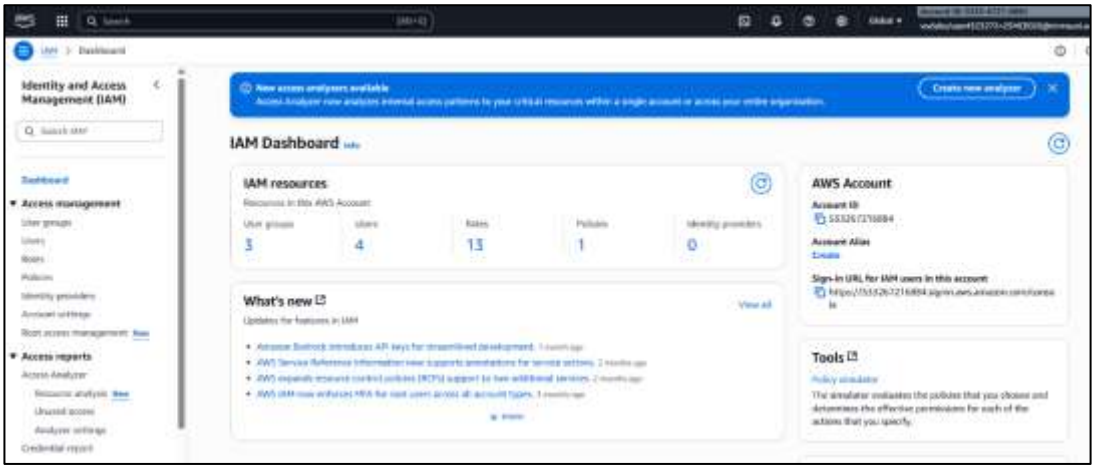
4. In the search box to the right of **Services**, search for and choose **IAM** to open the IAM console.



5. In the navigation pane on the left, choose **Users**.

The following IAM Users have been created for you:

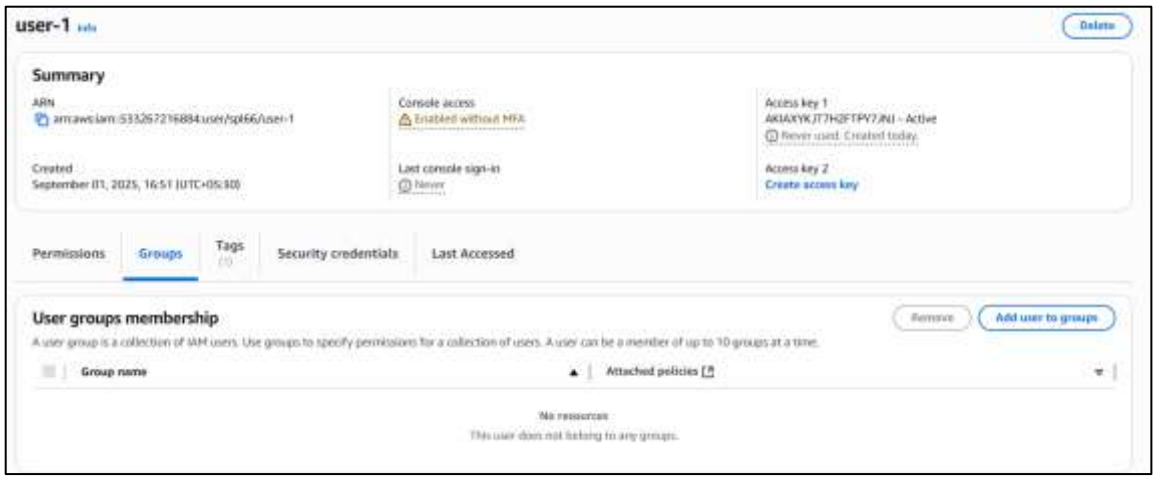
- user-1
- user-2
- user-3



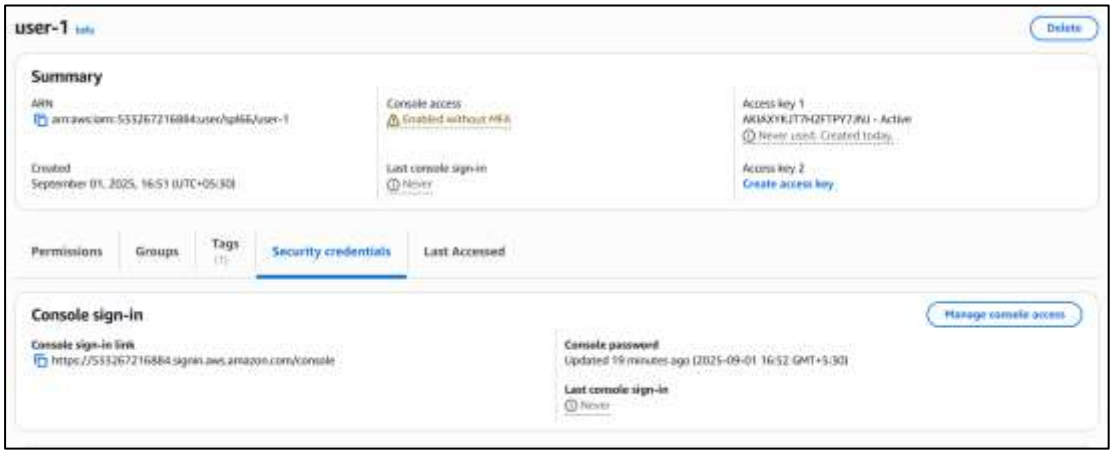
6. Choose the **user-1** link.
   This will bring to a summary page for user-1. The **Permissions** tab will be displayed.

7. Notice that user-1 does not have any permissions.

8. Choose the **Groups** tab.
   user-1 also is not a member of any groups.
9. Choose the **Security credentials** tab.



user-1 is assigned a **Console password**.

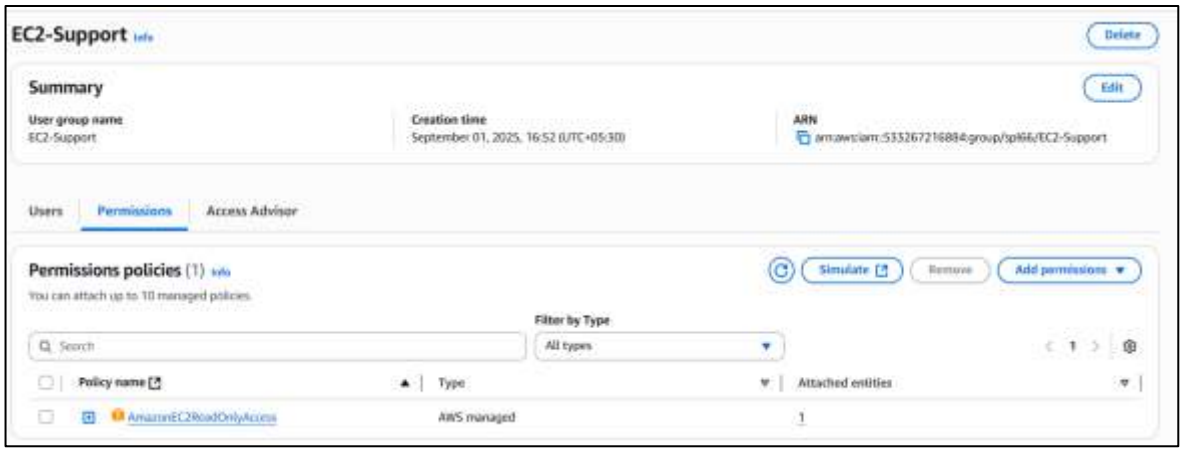


10. In the navigation pane on the left, choose **User groups**.
    The following groups have already been created for you:
    - EC2-Admin
    - EC2-Support
    - S3-Support

11. Choose the **EC2-Support** group link.
    This will bring you to the summary page for the **EC2-Support** group.

12. Choose the **Permissions** tab.

    This group has a Managed Policy associated with it, called **AmazonEC2ReadOnlyAccess**. Managed Policies are pre-built policies (built either by AWS or by your administrators) that can be attached to IAM Users and Groups. When the policy is updated, the changes to the policy are immediately apply against all Users and Groups that are attached to the policy.



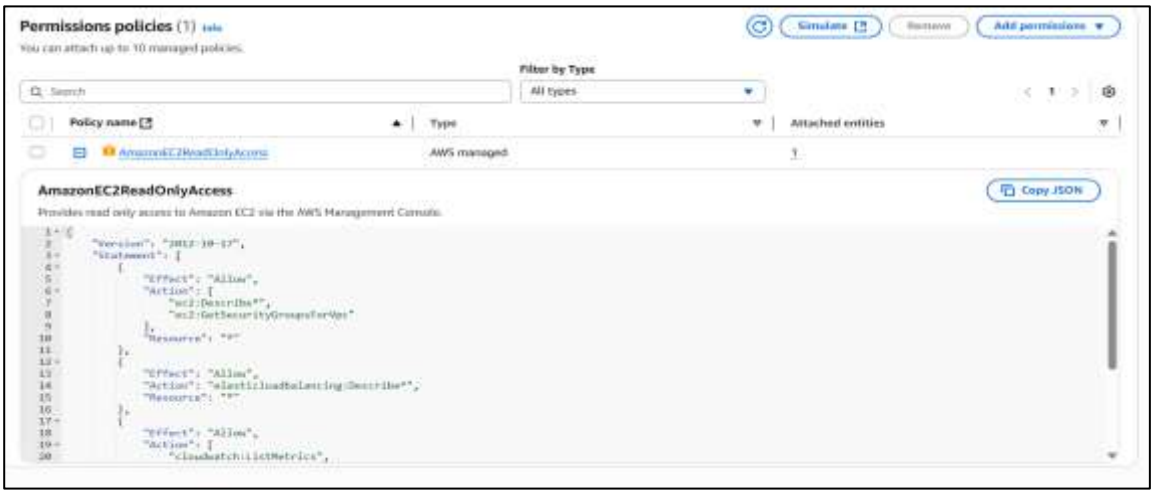
13. Choose the plus (+) icon next to the AmazonEC2ReadOnlyAccess policy to view the policy details.

    Note: A policy defines what actions are allowed or denied for specific AWS resources. This policy is granting permission to List and Describe information about EC2, Elastic Load Balancing, CloudWatch and Auto Scaling. This ability to view resources, but not modify them, is ideal for assigning to a Support role.

    The basic structure of the statements in an IAM Policy is:

    - **Effect** says whether to *Allow* or *Deny* the permissions.
    - **Action** specifies the API calls that can be made against an AWS Service (eg *cloudwatch:ListMetrics*).
    - **Resource** defines the scope of entities covered by the policy rule (eg a specific Amazon S3 bucket or Amazon EC2 instance, or * which means *any resource*).

14. Choose the minus icon (-) to hide the policy details.

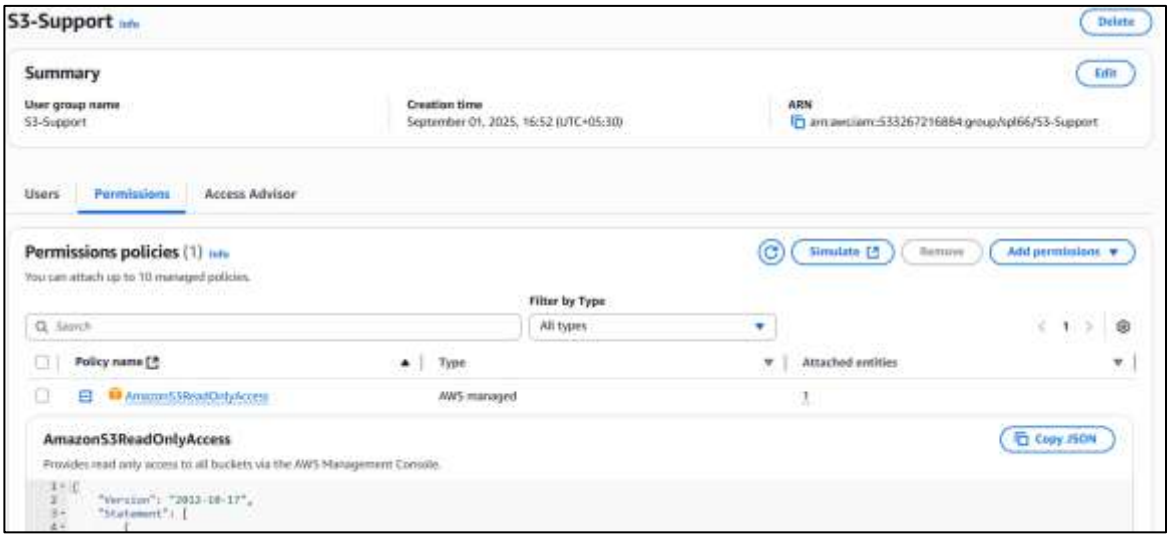


15. In the navigation pane on the left, choose **User groups**.

16. Choose the **S3-Support** group link and then choose the **Permissions** tab.
    The S3-Support group has the **AmazonS3ReadOnlyAccess** policy attached.

17. Choose the plus (+) icon to view the policy details.
    This policy grants permissions to Get and List resources in Amazon S3.

18. Choose the minus icon (-) to hide the policy details.

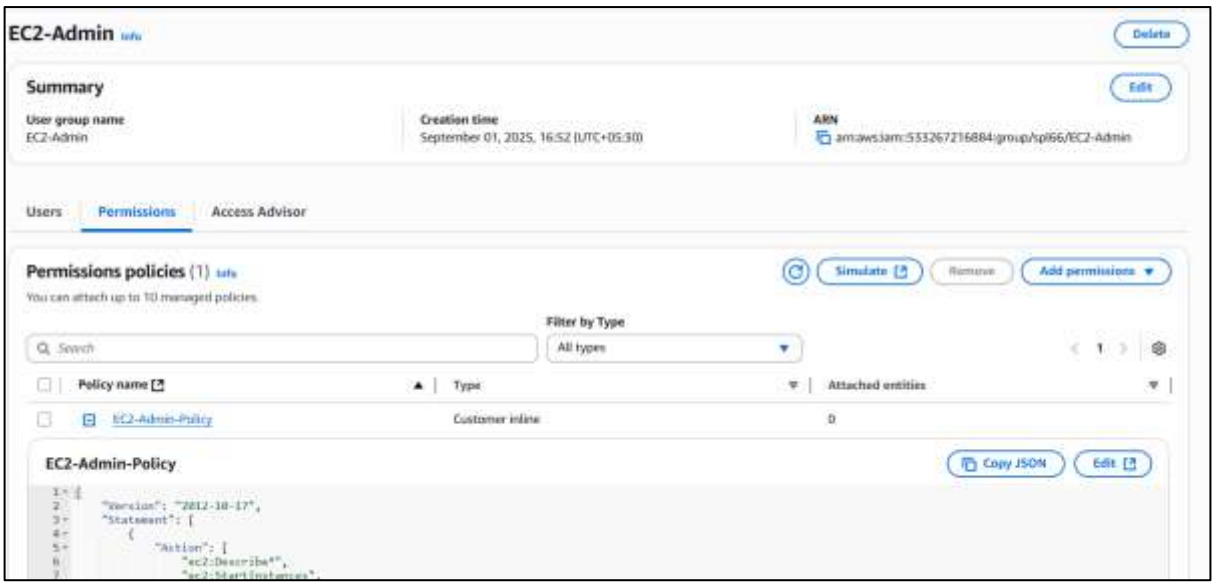


19. In the navigation pane on the left, choose **User groups**.

20. Choose the **EC2-Admin** group link and then choose the **Permissions** tab.
    This Group is slightly different from the other two. Instead of a *Managed Policy*, it has an **Inline Policy**, which is a policy assigned to just one User or Group.

21. Choose the plus (+) icon to view the policy details.

22. Choose the minus icon (-) to hide the policy details.



Task 2: Add Users to Groups

You have recently hired **user-1** into a role where they will provide support for Amazon S3. You will add them to the **S3-Support** group so that they inherit the necessary permissions via the attached *AmazonS3ReadOnlyAccess* policy.

You can ignore any "not authorized" errors that appear during this task. They are caused by your lab account having limited permissions and will not impact your ability to complete the lab.

Add user-1 to the S3-Support Group

23. In the left navigation pane, choose **User groups**.

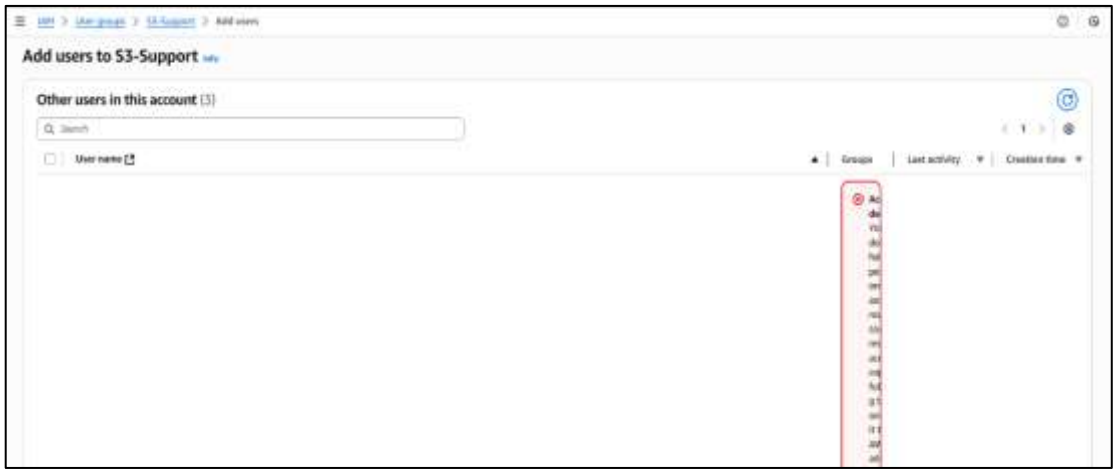
24. Choose the **S3-Support** group link.

25. Choose the **Users** tab.

26. In the **Users** tab, choose **Add users**.

27. In the **Add Users to S3-Support** window, configure the following:
    - Select **user-1**.
    - At the bottom of the screen, choose **Add users**.

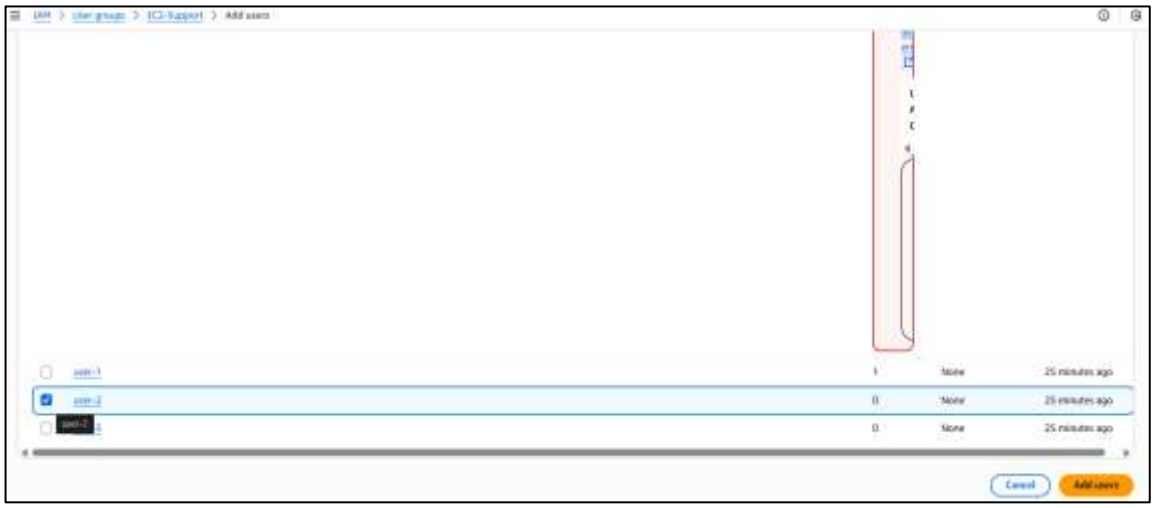
In the **Users** tab you will see that user-1 has been added to the group.



Add user-2 to the EC2-Support Group

You have hired **user-2** into a role where they will provide support for Amazon EC2.

28. Using similar steps to the ones above, add **user-2** to the **EC2-Support** group.
    user-2 should now be part of the **EC2-Support** group.

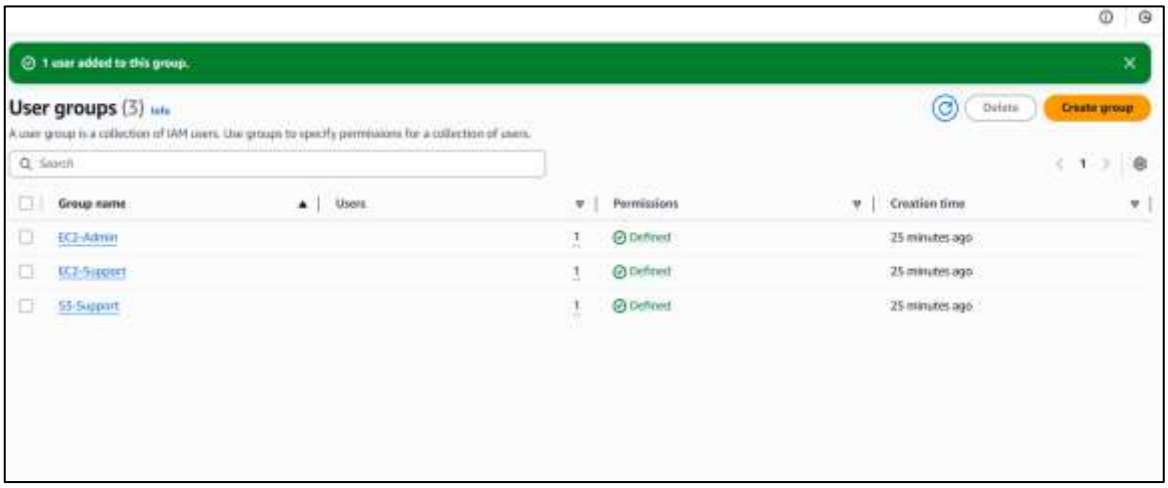


Add user-3 to the EC2-Admin Group

You have hired **user-3** as your Amazon EC2 administrator, who manage your EC2 instances.

29. Using similar steps to the ones above, add **user-3** to the **EC2-Admin** group.
    user-3 should now be part of the **EC2-Admin** group.

30. In the navigation pane on the left, choose **User groups**.
    Each Group should now have a **1** in the Users column, indicating the number of Users in each Group.
    If you do not have a **1** beside each group, revisit the above instructions above to ensure that each user is assigned to a User group, as shown in the table in the Business Scenario section.



Task 3: Sign-In and Test Users

In this task, you will test the permissions of each IAM User.

31. In the navigation pane on the left, choose **Dashboard**.
    A **Sign-in URL for IAM users in this account** link is displayed on the right. It will look similar to: *https://123456789012.signin.aws.amazon.com/console*
    This link can be used to sign-in to the AWS Account you are currently using.

32. Copy the **Sign-in URL for IAM users in this account** to a text editor.

33. Open a private (Incognito) window.
    Mozilla Firefox
    - Choose the menu bars at the top-right of the screen
    - Select **New private window**

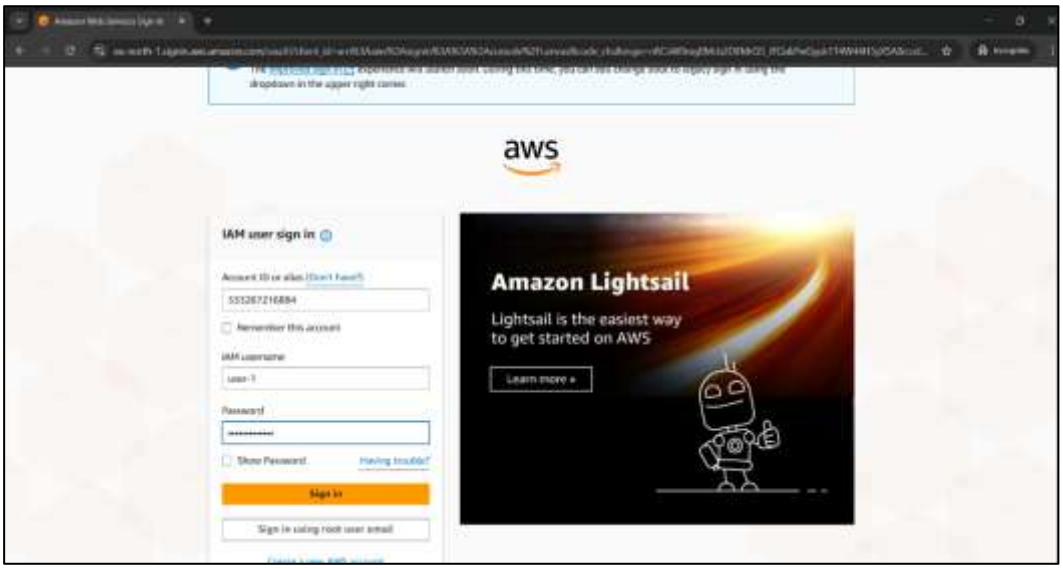
    Google Chrome
    - Choose the ellipsis at the top-right of the screen
    - Select **New Incognito Window**

    Microsoft Edge
    - Choose the ellipsis at the top-right of the screen
    - Choose **New InPrivate window**

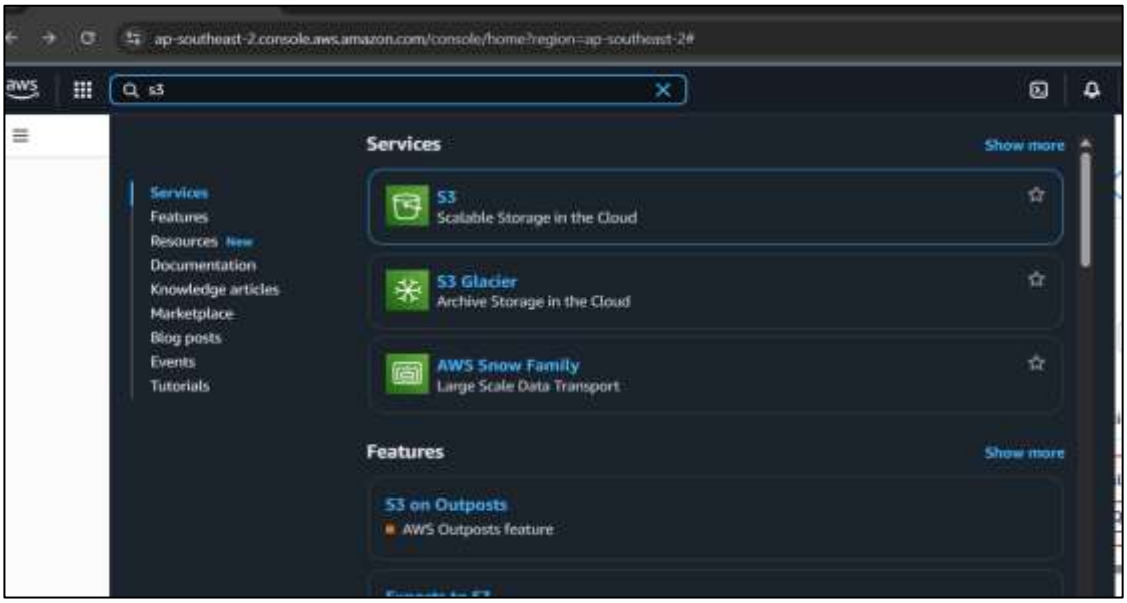
    Microsoft Internet Explorer
    - Choose the **Tools** menu option
    - Choose **InPrivate Browsing**

34. Paste the **IAM users sign-in** link into the address bar of your private browser session and press **Enter**.
    Next, you will sign-in as **user-1**, who has been hired as your Amazon S3 storage support staff.

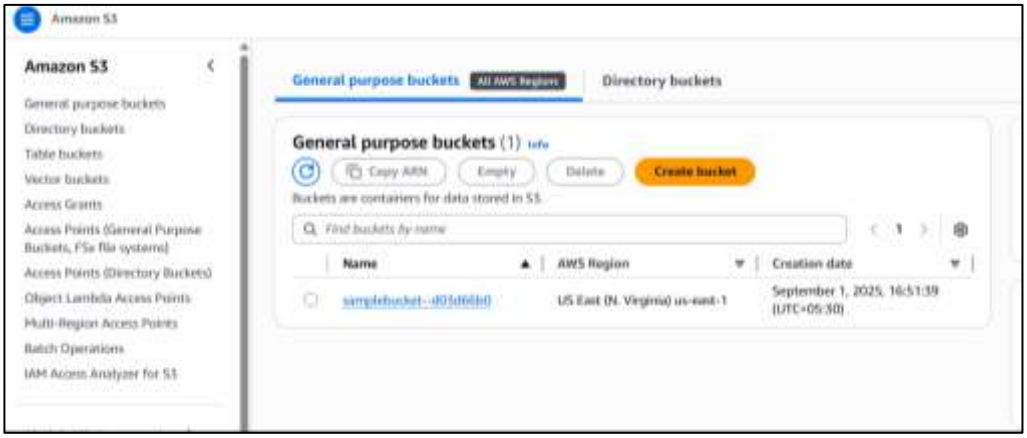
35. Sign-in with:
    - **IAM user name:** user-1
    - **Password:** Lab-Password1



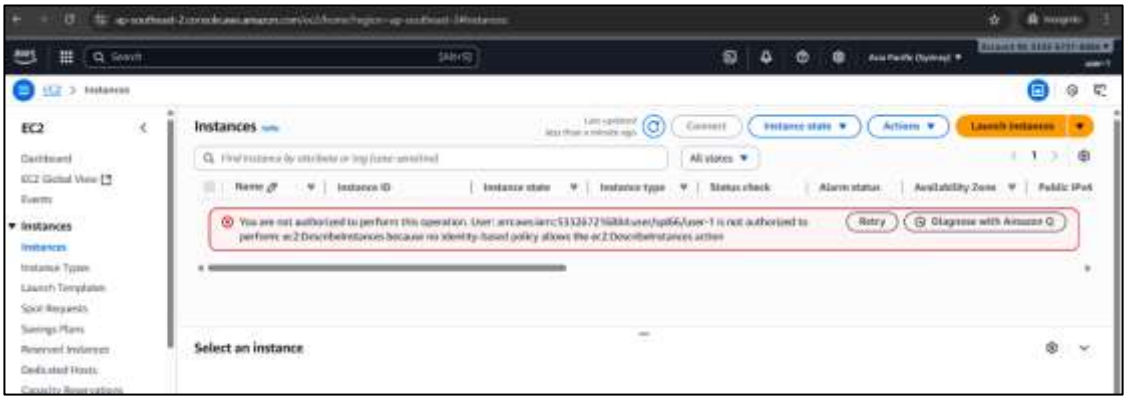
36. In the search box to the right of **Services**, search for and choose **S3** to open the S3 console.



37. Choose the name of the bucket that exists in the account and browse the contents.
    Since your user is part of the **S3-Support** Group in IAM, they have permission to view a list of Amazon S3 buckets and the contents.
    Note: The bucket does not contain any objects.
    Now, test whether they have access to Amazon EC2.

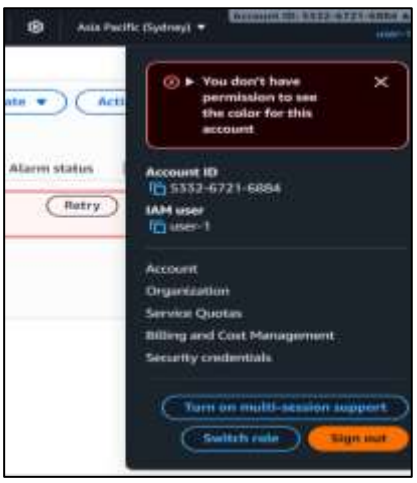


38. In the search box to the right of **Services**, search for and choose **EC2** to open the EC2 console.

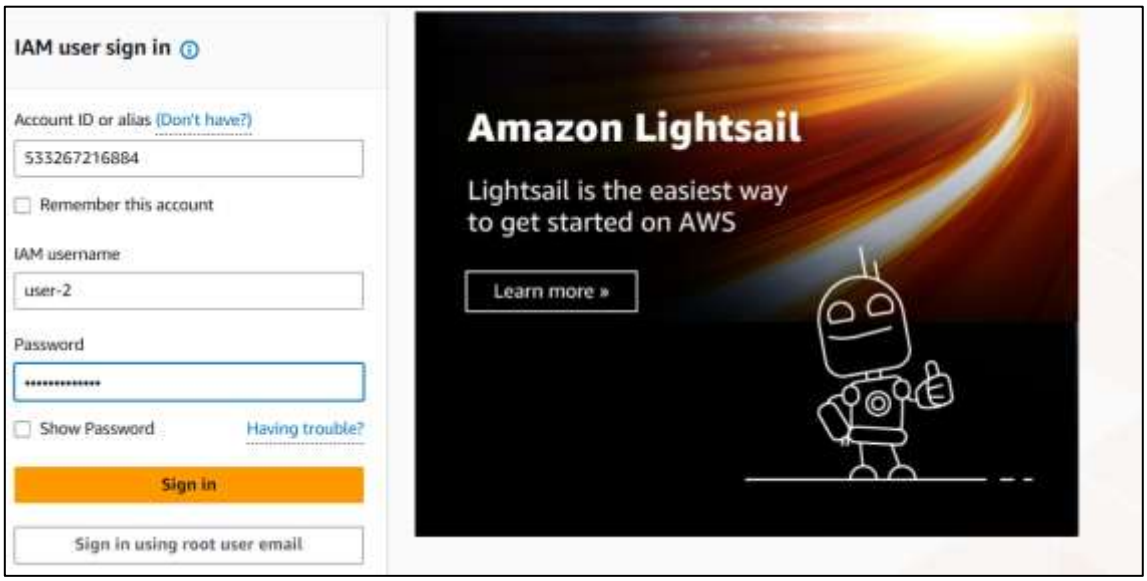


39. In the left navigation pane, choose **Instances**.
    You cannot see any instances. Instead, you see a message that states *You are not authorized to perform this operation*. This is because this user has not been granted any permissions to access Amazon EC2.
    You will now sign-in as **user-2**, who has been hired as your Amazon EC2 support person.

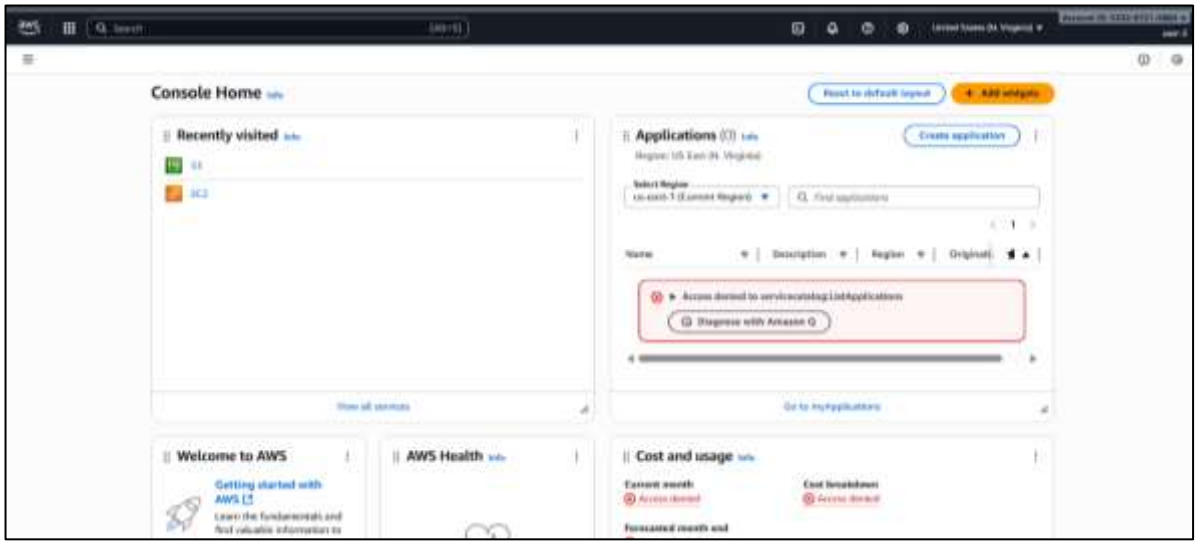
40. Sign user-1 out of the **AWS Management Console** by completing the following actions:
    - At the top of the screen, choose **user-1**
    - Choose **Sign Out.**



41. Paste the **IAM users sign-in** link into your private browser tab's address bar and press **Enter**.
42. Sign-in with:
    - **IAM user name:** user-2
    - **Password:** Lab-Password2



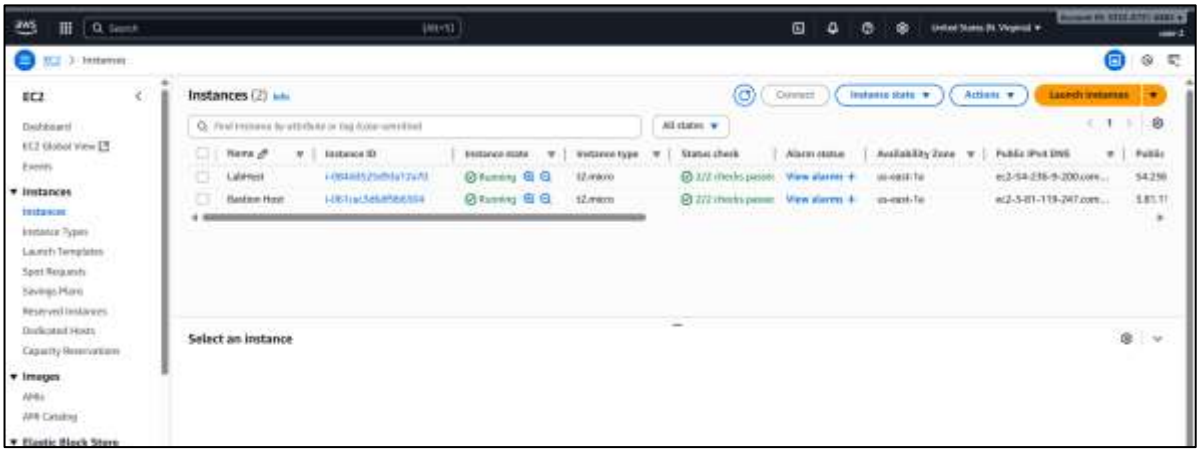
43. In the search box to the right of **Services**, search for and choose **EC2** to open the EC2 console.
44. In the navigation pane on the left, choose **Instances**.



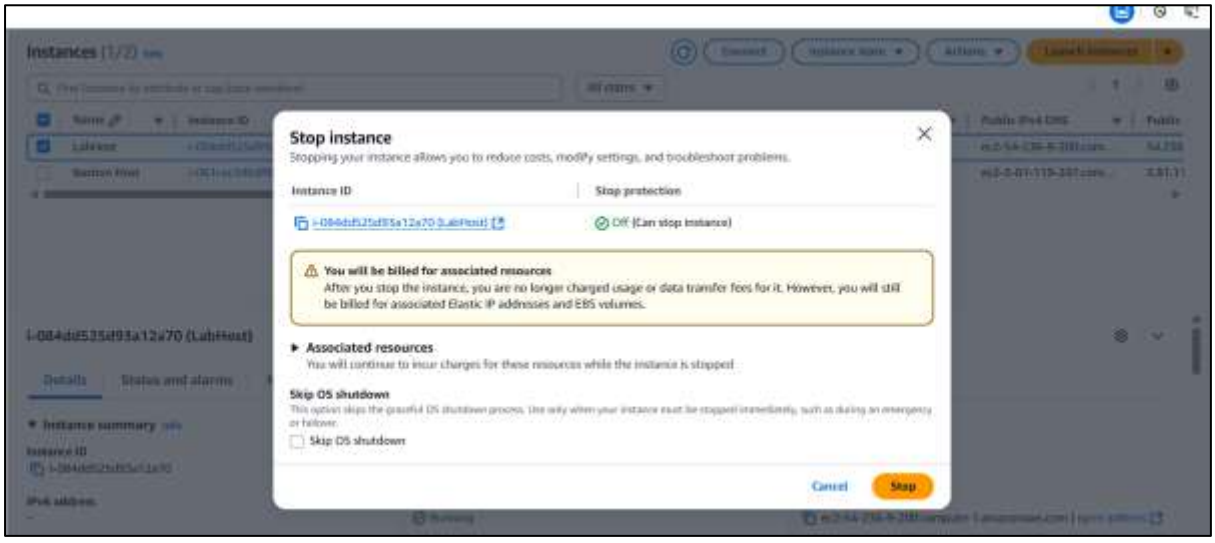
You are now able to see an Amazon EC2 instance because you have Read Only permissions. However, you will not be able to make any changes to Amazon EC2 resources.

If you cannot see an Amazon EC2 instance, then your Region may be incorrect. In the top-right of the screen, pull-down the Region menu and select the region that you noted at the start of the lab (for example, **N. Virginia**).

Select the instance named *LabHost*.

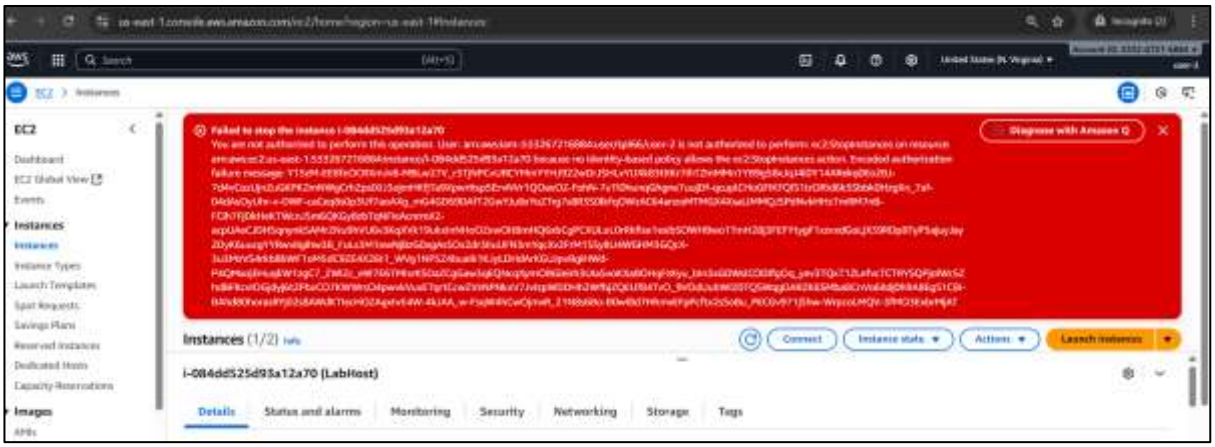


45. In the **Instance state** menu above, select **Stop instance**.



46. In the **Stop Instance** window, select **Stop**.

You will receive an error stating *You are not authorized to perform this operation*. This



demonstrates that the policy only allows you to view information, without making changes.

47. Choose the X to close the *Failed to stop the instance* message.

Next, check if user-2 can access Amazon S3.

48. In the search box to the right of **Services**, search for and choose **S3** to open the S3 console.

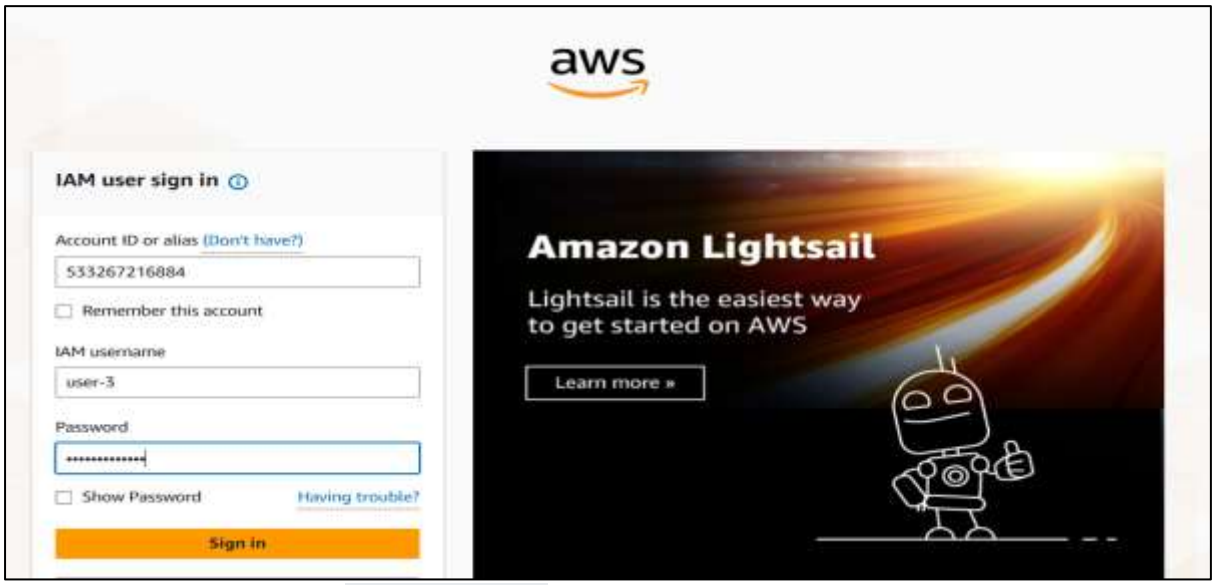
You will see the message **You don't have permissions to list buckets** because user-2 does not have permission to access Amazon S3.

You will now sign-in as **user-3**, who has been hired as your Amazon EC2 administrator.

Sign user-2 out of the **AWS Management Console** by completing the following actions:

- At the top of the screen, choose **user-2**
- Choose **Sign Out**

49. Paste the **IAM users sign-in** link into your private window and press **Enter**.
50. Paste the sign-in link into the address bar of your private web browser tab again. If it is not in your clipboard, retrieve it from the text editor where you stored it earlier.
51. Sign-in with:
    - **IAM user name:** user-3



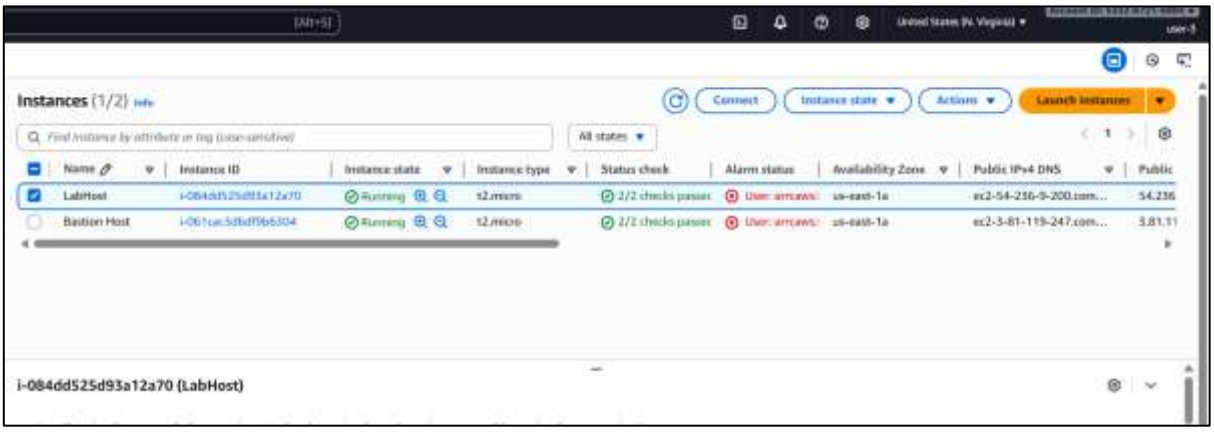
    - **Password:** Lab-Password3

52. In the search box to the right of **Services**, search for and choose **EC2** to open the EC2 console.
53. In the navigation pane on the left, choose **Instances**.

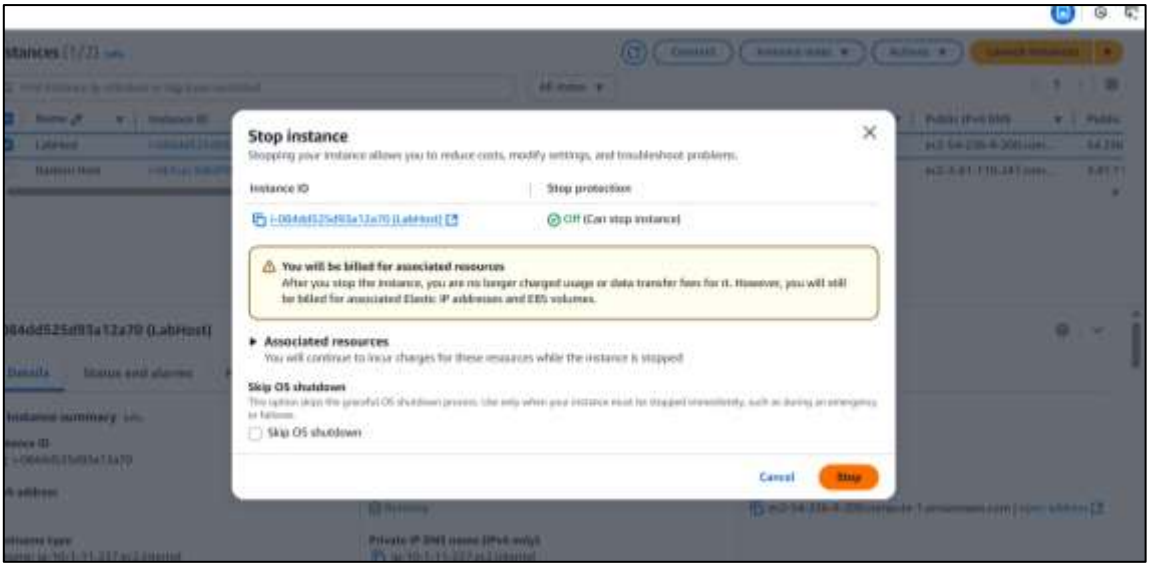
As an EC2 Administrator, you should now have permissions to Stop the Amazon EC2 instance.

Select the instance named *LabHost* .

If you cannot see an Amazon EC2 instance, then your Region may be incorrect. In the top-right of the screen, pull-down the Region menu and select the region that you noted at the start of the lab (for example, **N. Virginia**).

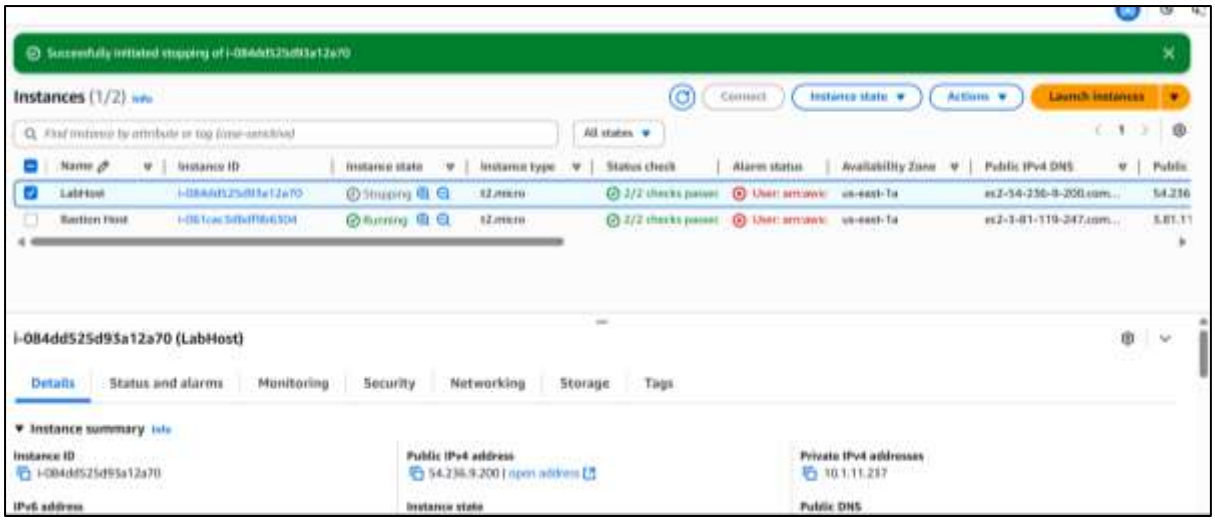


54. In the **Instance state** menu, choose **Stop instance**.



55. In the **Stop instance** window, choose **Stop**.

The instance will enter the *stopping* state and will shutdown.



❖ **Conclusion:**

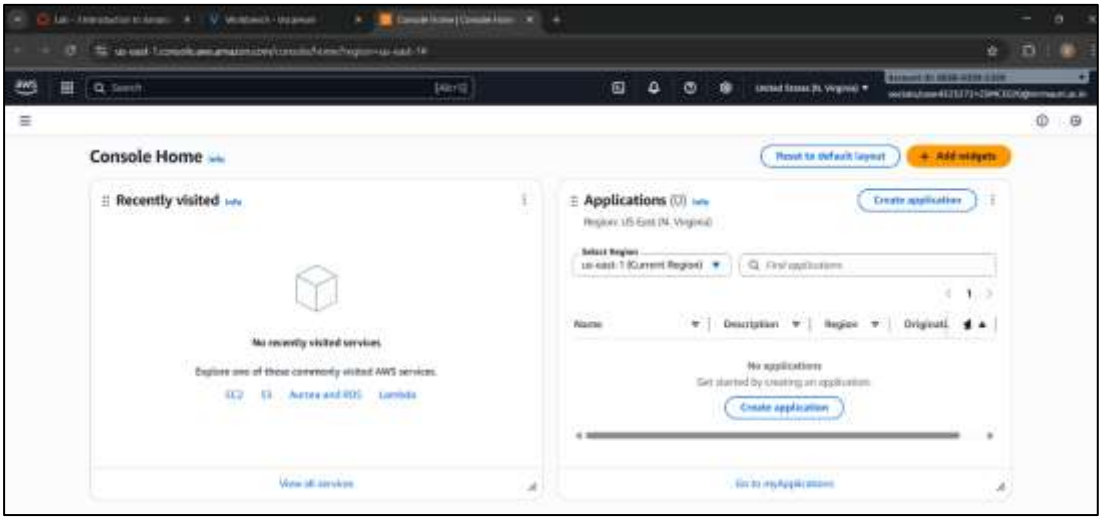
By applying IAM best practices in AWS, we ensured that every user only has the access they truly need. Creating separate groups like EC2-Admin, EC2-Support, and S3-Support allowed us to assign permissions in a clear and controlled way. This not only enforced the principle of least privilege but also reduced the risk of accidental or unauthorized actions. The final tests confirmed that the policies worked as intended—granting access where appropriate and blocking it where not—helping us maintain both security and efficiency in managing AWS resources.

Experiment – 3 (b)

- **Aim:** Learn the basic overview of launching, resizing, managing, and monitoring an Amazon EC2 instance. Launch a web server (using Putty) with termination protection enabled, Monitor Your EC2 instance, Modify the security group that your web server is using to allow HTTP access, Resize your Amazon EC2 instance to scale, Explore EC2 limits, Test termination protection, and Terminate your EC2 instance.
- **Accessing the AWS Management Console:**

1. At the top of these instructions, choose **Start Lab.**
    - The lab session starts.
    - A timer displays at the top of the page and shows the time remaining in the session.
    - Before you continue, wait until the circle icon to the right of the **AWS** link in the upper-left corner turns green.

2. To connect to the **AWS** Management Console, choose the **AWS** link in the upper-left corner.
    - A new browser tab opens and connects you to the console.

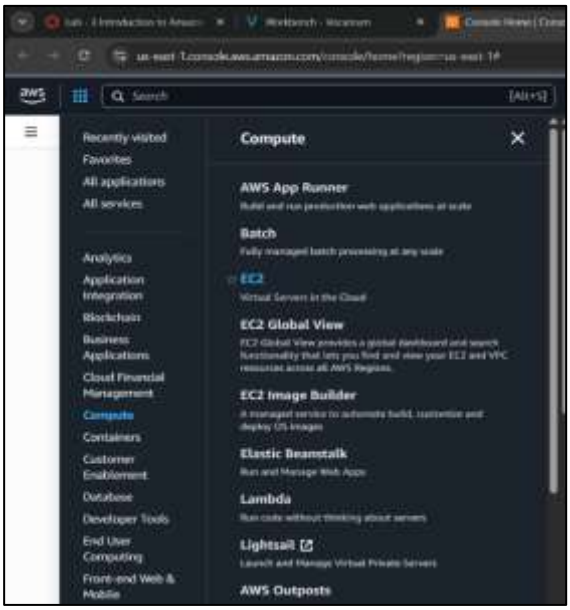


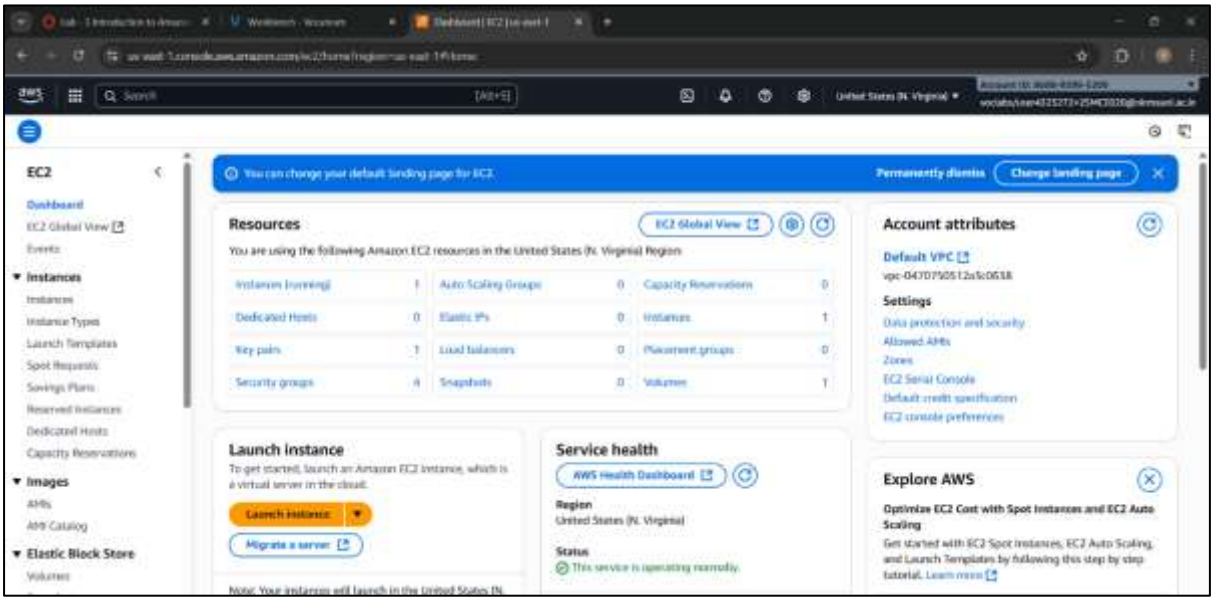
3. Arrange the AWS Management Console tab so that it displays along side these instructions. Ideally, you will be able to see both browser tabs at the same time, to make it easier to follow the lab steps.

Task 1: Launch Your Amazon EC2 Instance

In this task, you will launch an Amazon EC2 instance with *termination protection* and *stop protection*. Termination protection prevents you from accidentally terminating the EC2 instance and stop protection prevents you from accidentally stopping the EC2 instance. You will also specify a User Data script when you launch the instance that will deploy a simple web server.

4. In the AWS Management Console choose Services, choose Compute and then choose EC2.





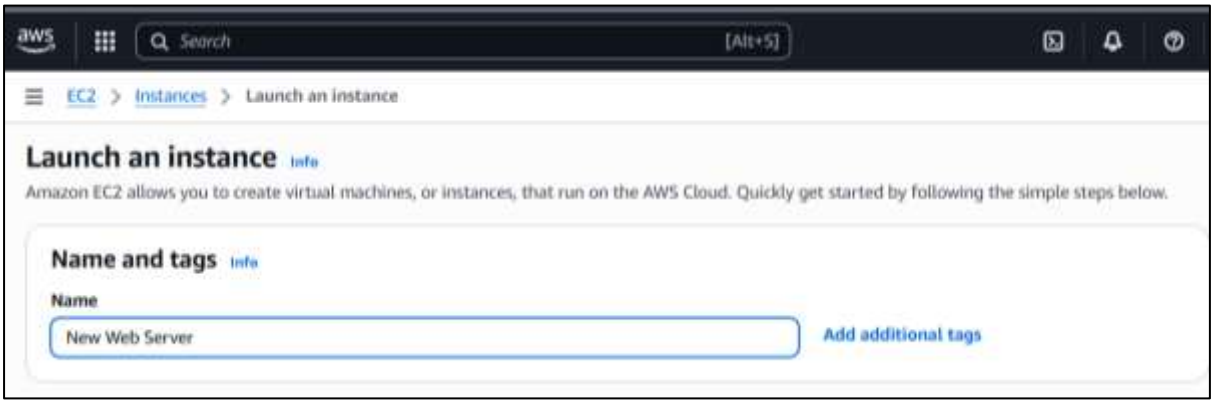
5. Choose the Launch instance menu and select **Launch instance.**

Step 1: Name and tags

6. Give the instance the name Web Server.

The Name you give this instance will be stored as a tag. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. This is useful when you have many resources of the same type — you can quickly identify a specific resource based on the tags you have assigned to it. Each tag consists of a Key and a Value, both of which you define. You can define multiple tags to associate with the instance if you want to.

In this case, the tag that will be created will consist of a *key* called Name with a *value* of Web Server.



Step 2: Application and OS Images (Amazon Machine Image)

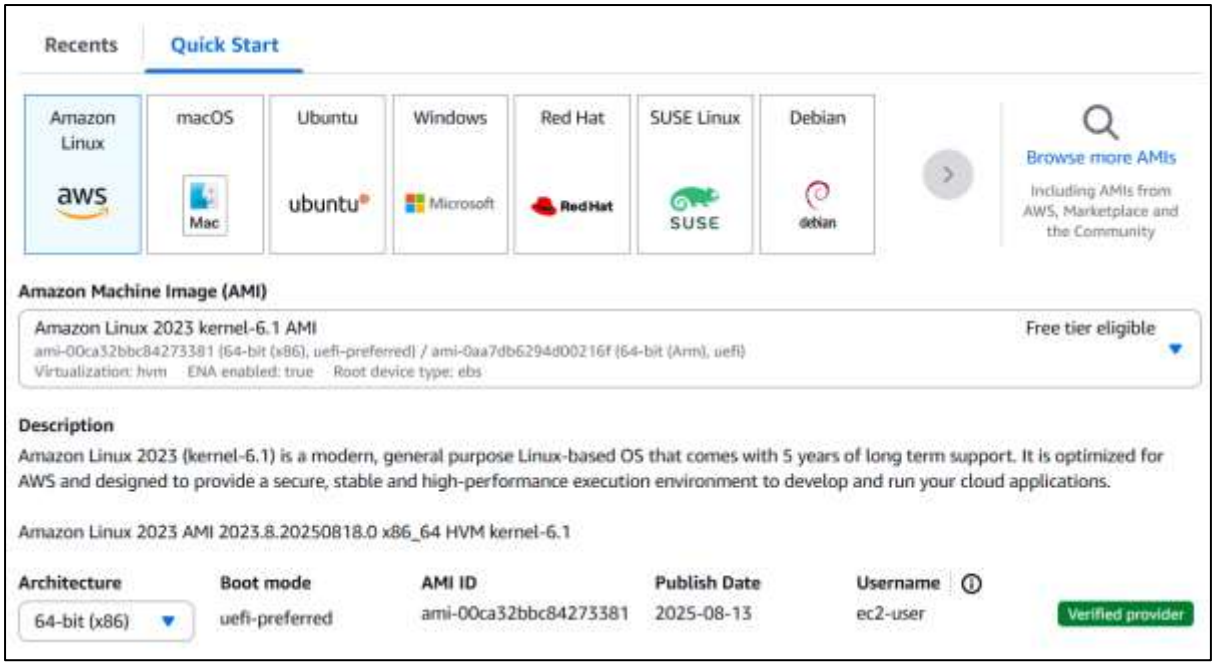
7. In the list of available *Quick Start* AMIs, keep the default Amazon Linux AMI selected.

8. Also keep the default Amazon Linux 2023 AMI selected.

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

- A template for the root volume for the instance (for example, an operating system or an application server with applications)
- Launch permissions that control which AWS accounts can use the AMI to launch instances
- A block device mapping that specifies the volumes to attach to the instance when it is launched

The Quick Start list contains the most commonly-used AMIs. You can also create your own AMI or select an AMI from the AWS Marketplace, an online store where you can sell or buy software that runs on AWS.



Step 3: Instance type

9. In the *Instance type* panel, keep the default t2.micro selected.

Amazon EC2 provides a wide selection of *instance types* optimized to fit different use cases. Instance types comprise varying combinations of CPU, memory, storage, and networking capacity and give you the flexibility to choose the appropriate mix of resources for your applications. Each instance type includes one or more *instance sizes*, allowing you to scale your resources to the requirements of your target workload.

The t2.micro instance type has 1 virtual CPU and 1 GiB of memory.

Step 4: Key pair (login)

10. For Key pair name - *required*, choose vockey.

Amazon EC2 uses public–key cryptography to encrypt and decrypt login information. To ensure you will be able to log in to the guest OS of the instance you create, you identify an existing key pair or create a new key pair when launching the instance. Amazon EC2 then installs the key on the guest OS when the instance is launched. That way, when you attempt to login to the instance and you provide the private key, you will be authorized to connect to the instance.

Step 5: Network settings

11. Next to Network settings, choose Edit.

12. For VPC, select Lab VPC.

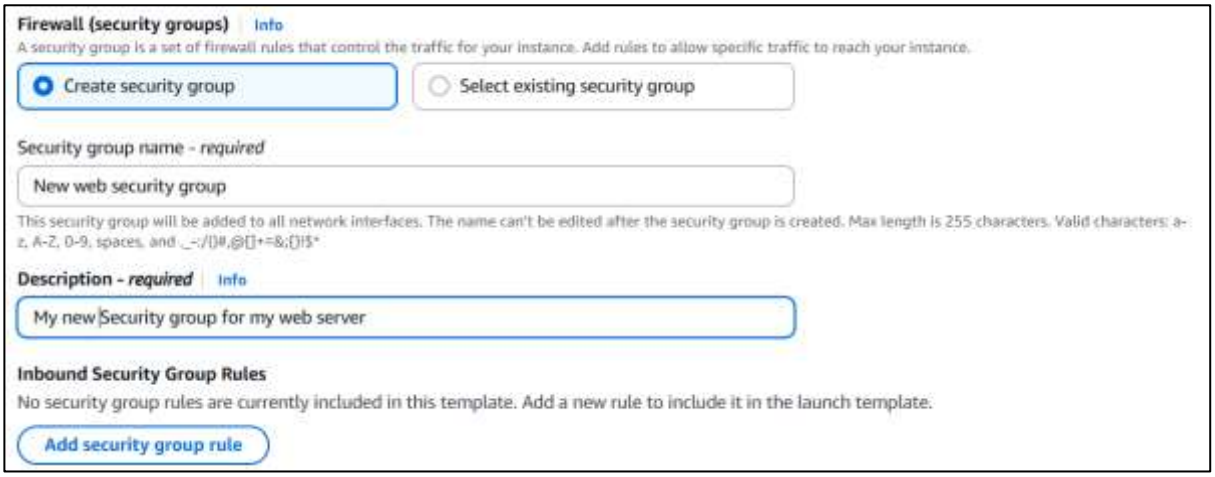
The Lab VPC was created using an AWS CloudFormation template during the setup process of your lab. This VPC includes two public subnets in two different Availability Zones.

Keep the default subnet **PublicSubnet1**. This is the subnet in which the instance will run. Notice also that by default, the instance will be assigned a public IP address.

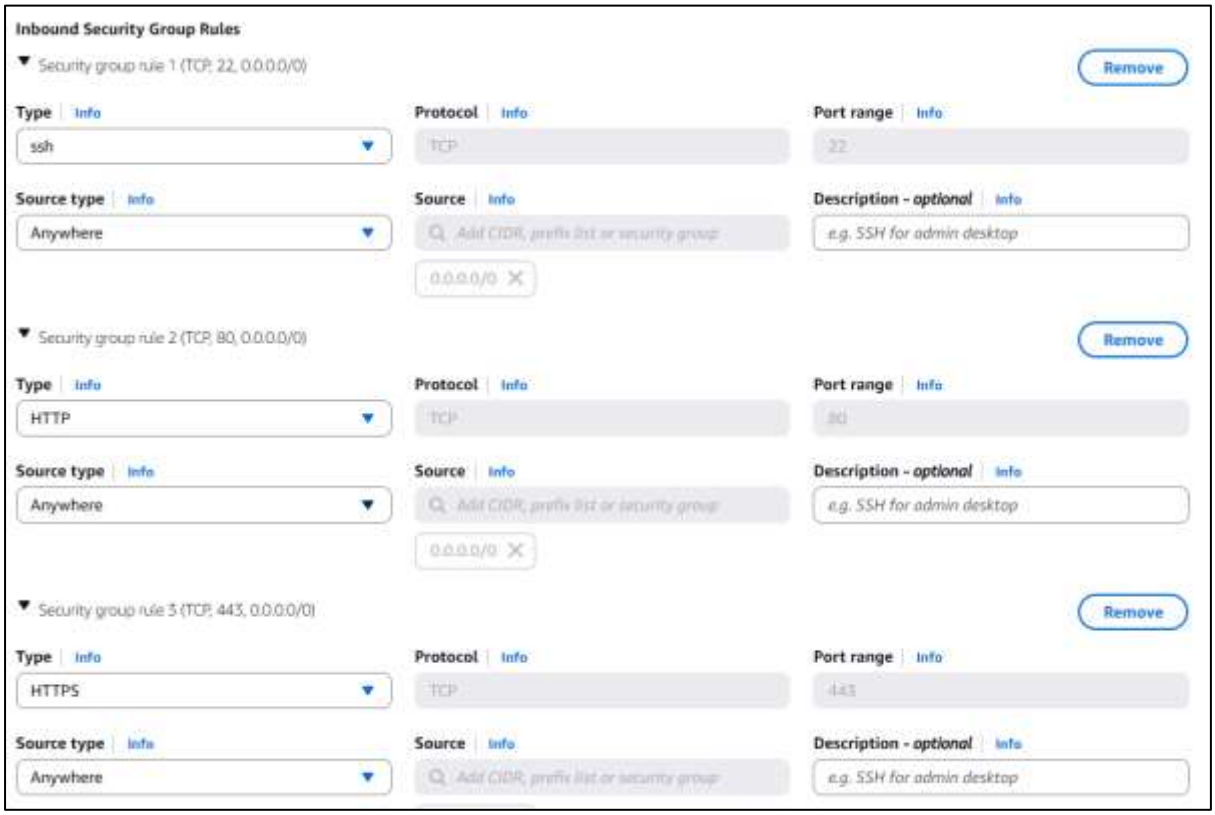
13. Under Firewall (security groups), choose Create security group and configure:

- Security group name: Web Server security group
- Description: Security group for my web server

A *security group* acts as a virtual firewall that controls the traffic for one or more instances. When you launch an instance, you associate one or more security groups with the instance. You add *rules* to each security group that allow traffic to or from its associated instances. You can modify the rules for a security group at any time; the new rules are automatically applied to all instances that are associated with the security group.



❖ Under Inbound security group rules, notice that one rule exists. Remove this rule.

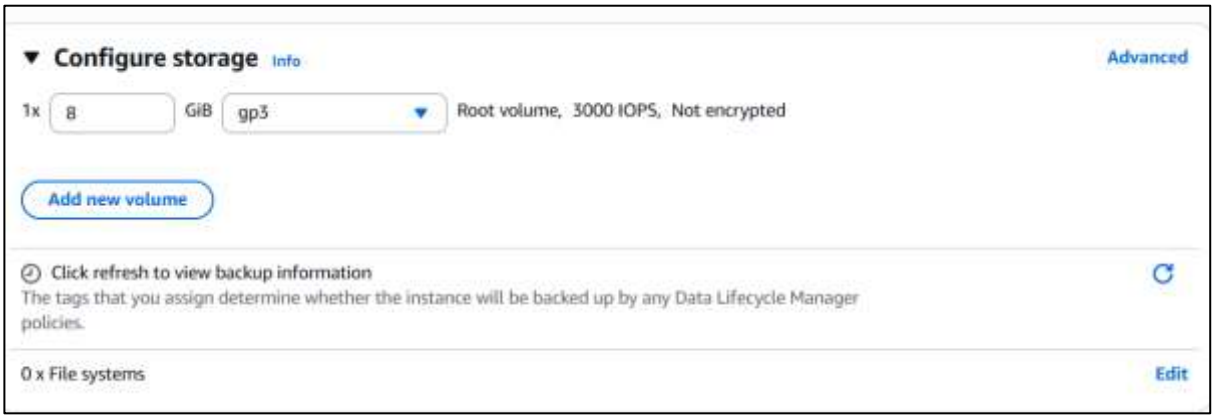


Step 6: Configure storage

14. In the *Configure storage* section, keep the default settings.

Amazon EC2 stores data on a network-attached virtual disk called *Elastic Block Store*.

You will launch the Amazon EC2 instance using a default 8 GiB disk volume. This will be your root volume (also known as a 'boot' volume).

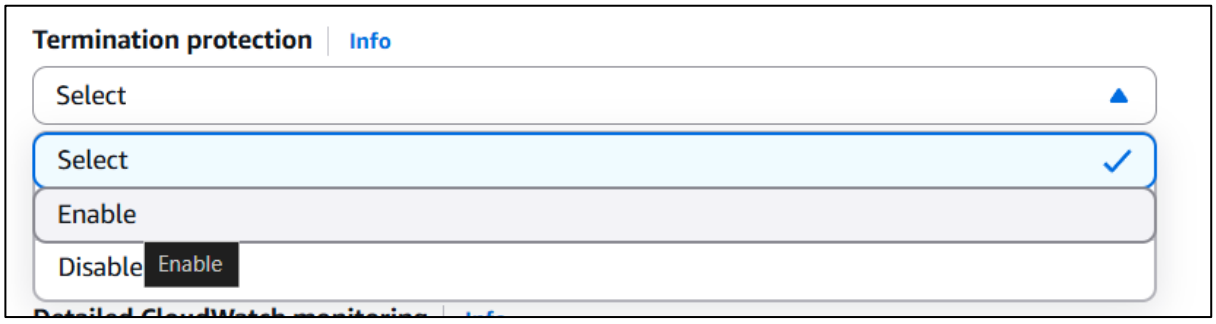


Step 7: Advanced details

15. Expand Advanced details.

16. For Termination protection, select Enable.

When an Amazon EC2 instance is no longer required, it can be *terminated*, which means that the instance is deleted and its resources are released. A terminated instance cannot be accessed again and the data that was on it cannot be recovered. If you want to prevent the instance from being accidentally terminated, you can enable *termination protection* for the instance, which prevents it from being terminated as long as this setting remains enabled.



17. Scroll to the bottom of the page and then copy and paste the code shown below into the **User data** box:
#!/bin/bash

dnf install -y httpd

systemctl enable httpd

systemctl start httpd

echo '<html><h1>Hello From Your Web Server!</h1></html>' > /var/www/html/index.html

When you launch an instance, you can pass *user data* to the instance that can be used to perform automated installation and configuration tasks after the instance starts.

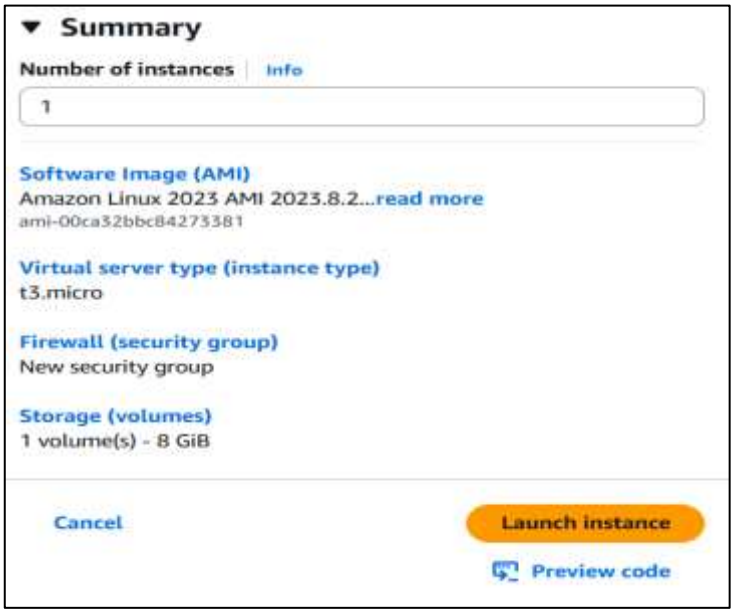
Your instance is running Amazon Linux 2023. The *shell script* you have specified will run as the *root* guest OS user when the instance starts. The script will:

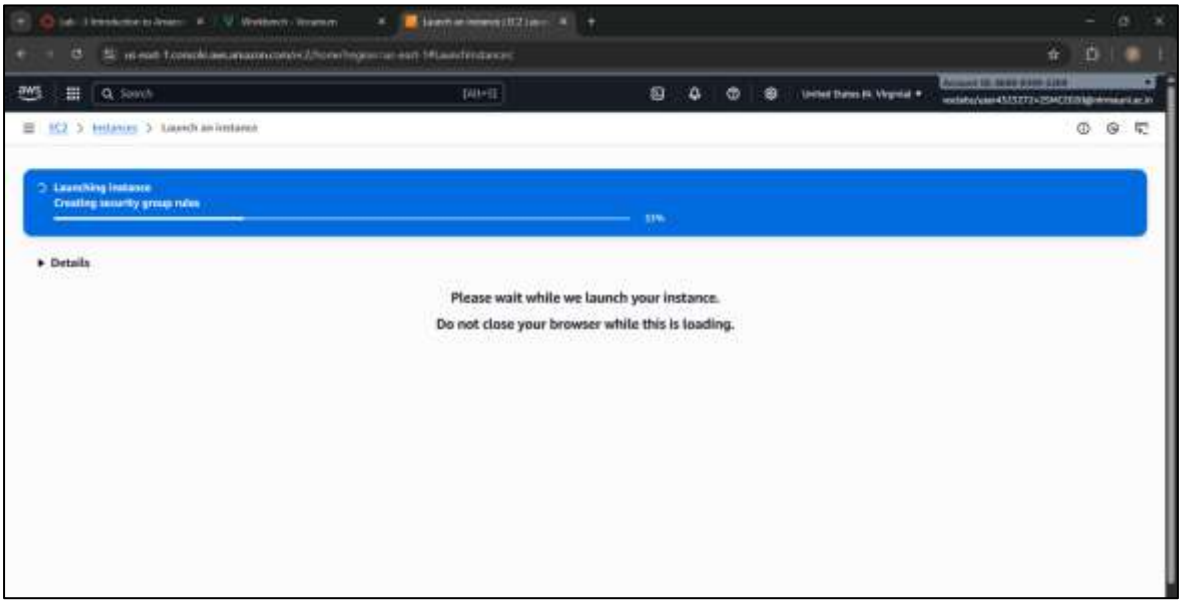
- Install an Apache web server (httpd)
- Configure the web server to automatically start on boot
- Run the Web server once it has finished installing
- Create a simple web page

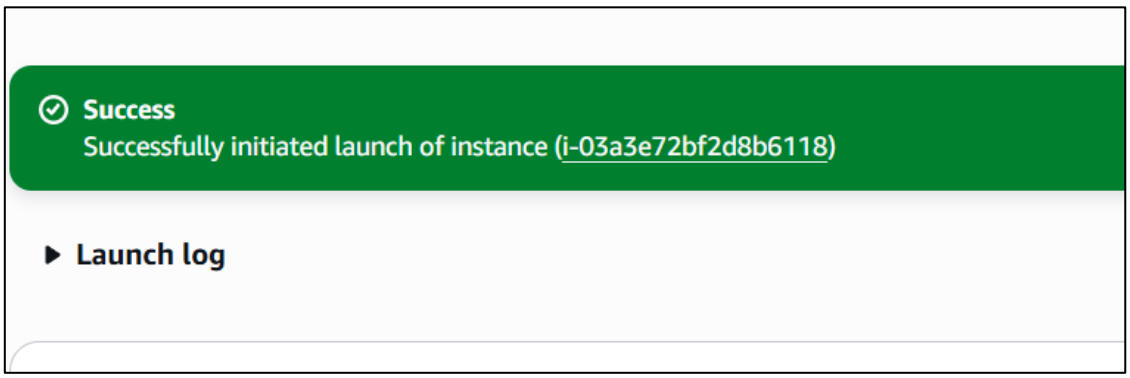
Step 8: Launch the instance

18. At the bottom of the Summary panel choose Launch instance

You will see a Success message.







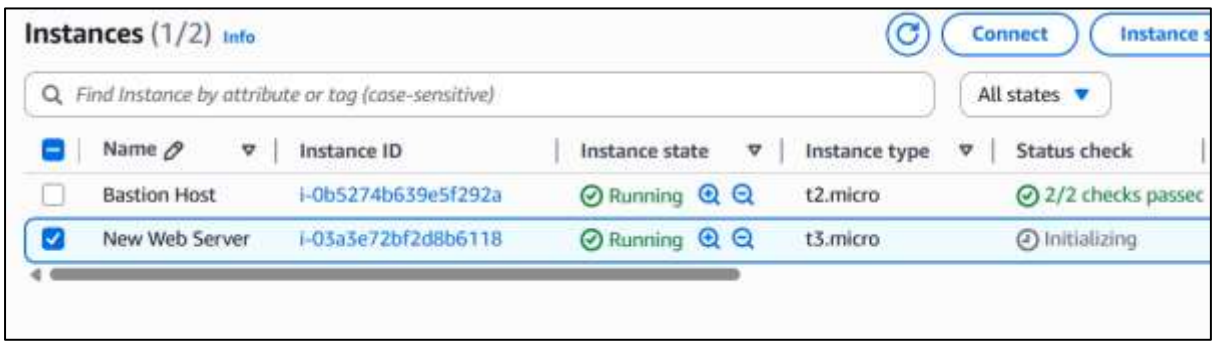
19. Choose View all instances

- In the Instances list, select Web Server.
- Review the information displayed in the Details tab. It includes information about the instance type, security settings and network settings.

The instance is assigned a *Public IPv4 DNS* that you can use to contact the instance from the Internet.

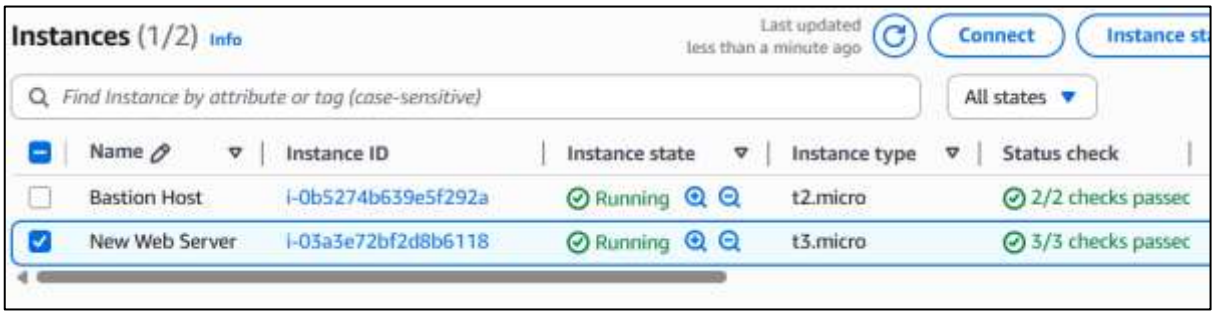
To view more information, drag the window divider upwards.

At first, the instance will appear in a *Pending* state, which means it is being launched. It will then change to *Initializing*, and finally to *Running*.



20. Wait for your instance to display the following:

- Instance State: *Running*
- Status Checks: *2/2 checks passed*



Task 2: Monitor Your Instance

Monitoring is an important part of maintaining the reliability, availability, and performance of your Amazon Elastic Compute Cloud (Amazon EC2) instances and your AWS solutions.

21. Choose the **Status checks** tab.

With instance status monitoring, you can quickly determine whether Amazon EC2 has detected any problems that might prevent your instances from running applications. Amazon EC2 performs automated checks on every running EC2 instance to identify hardware and software issues.

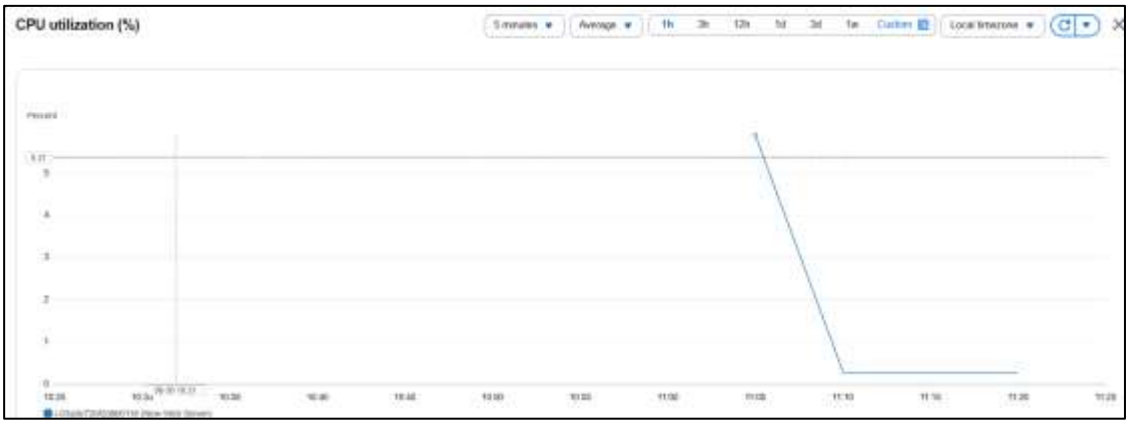
Notice that both the **System reachability** and **Instance reachability** checks have passed.

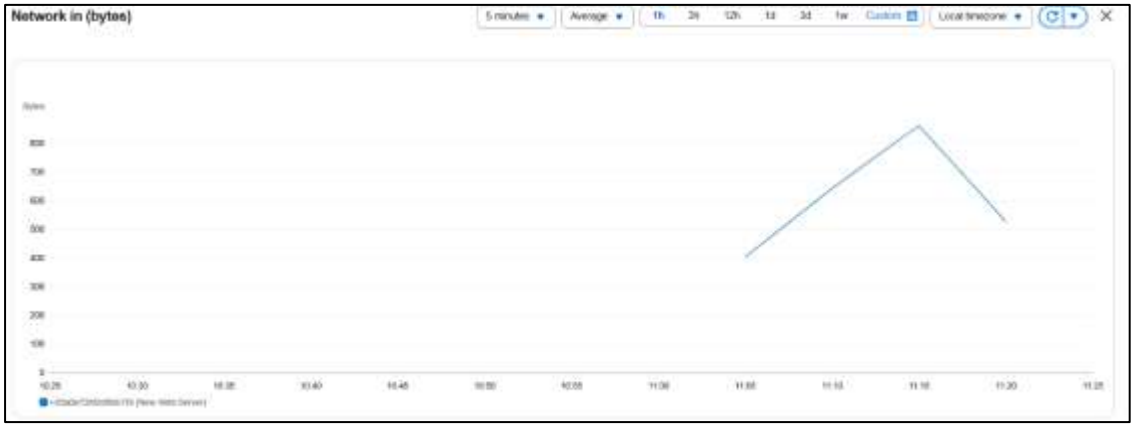
22. Choose the **Monitoring** tab.

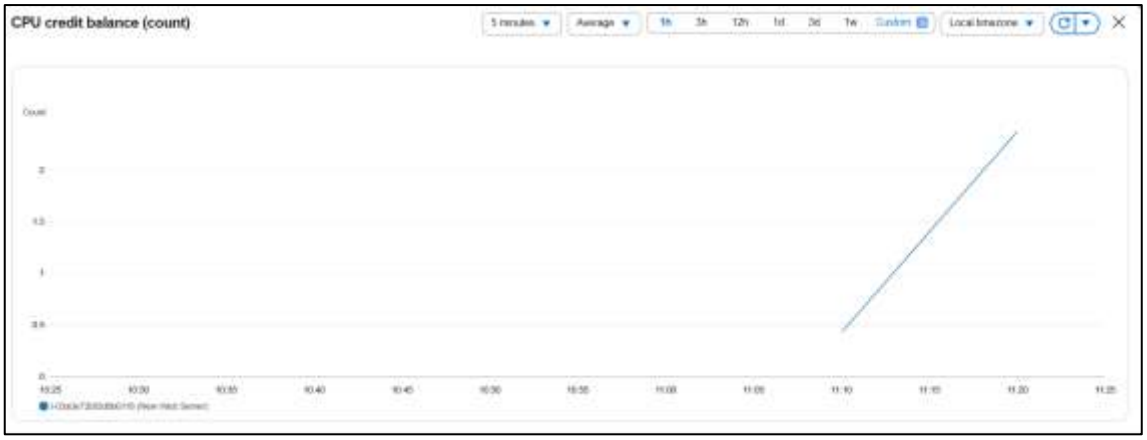
This tab displays Amazon CloudWatch metrics for your instance. Currently, there are not many metrics to display because the instance was recently launched.

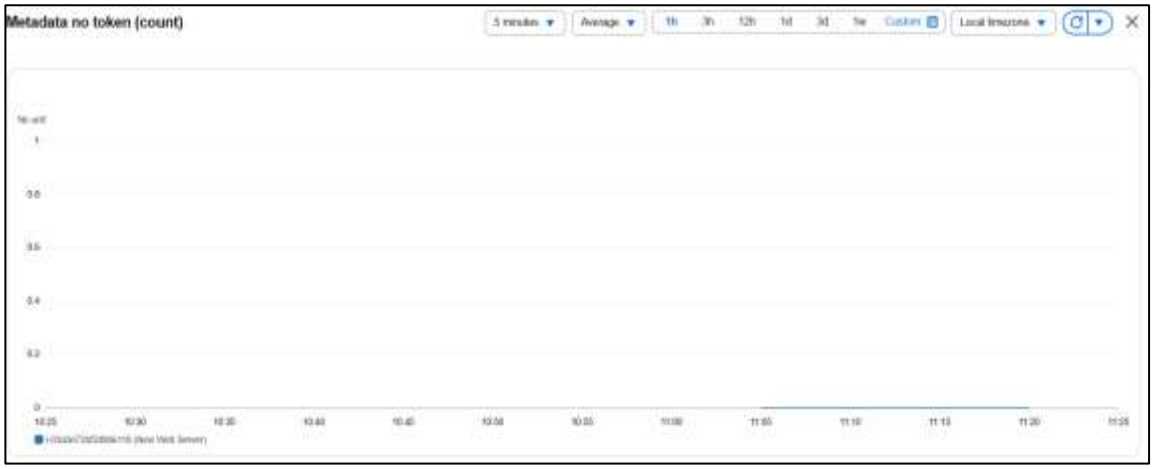
You can choose the three dots icon in any graph and select **Enlarge** to see an expanded view of the chosen metric.

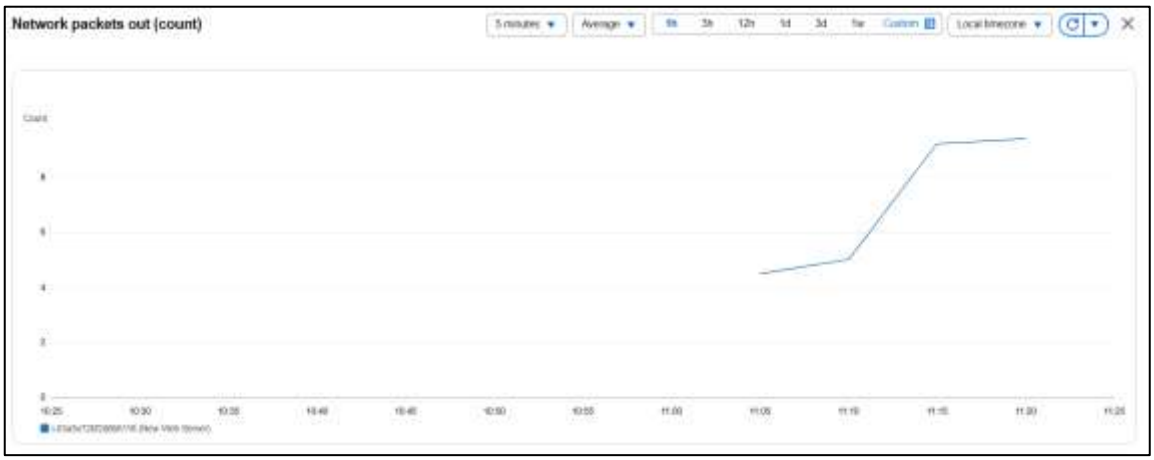
Amazon EC2 sends metrics to Amazon CloudWatch for your EC2 instances. Basic (five-minute) monitoring is enabled by default. You can also enable detailed (one-minute) monitoring.


CPU utilization (%)


Network in (bytes)

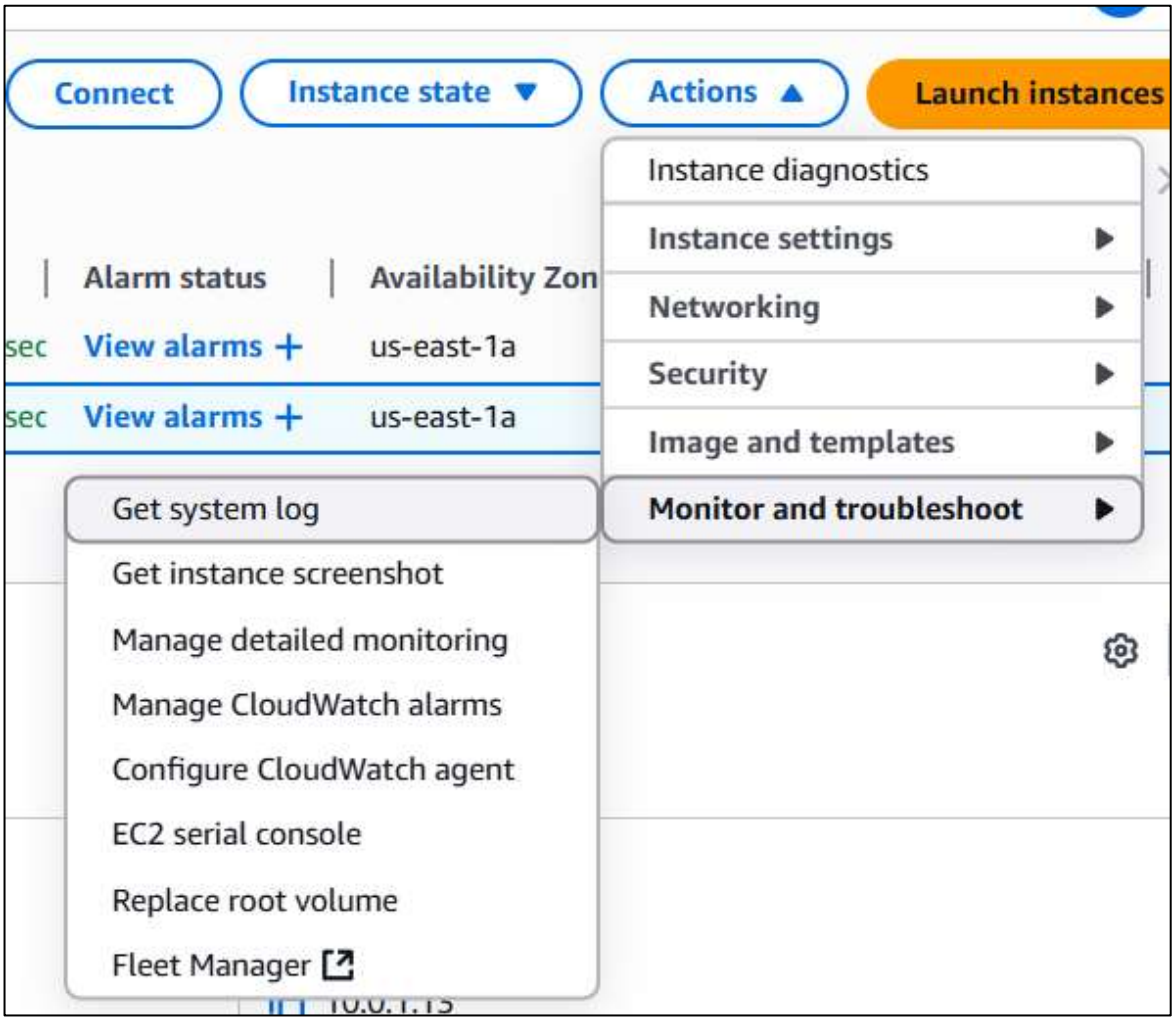

CPU credit balance (count)


Metadata no token (count)



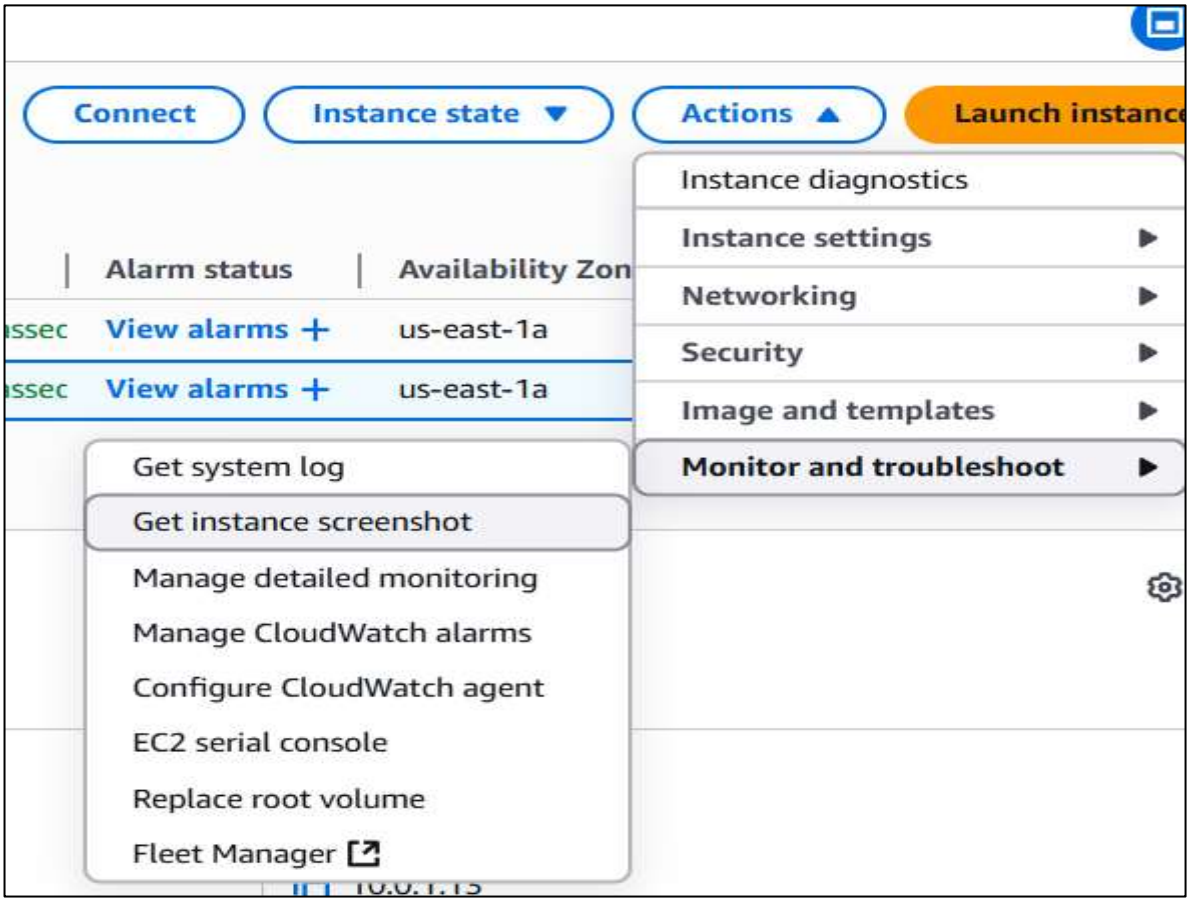
23. In the **Actions** menu towards the top of the console, select **Monitor and troubleshoot** **Get system log**.

The System Log displays the console output of the instance, which is a valuable tool for problem diagnosis. It is especially useful for troubleshooting kernel problems and service configuration issues that could cause an instance to terminate or become unreachable before its SSH daemon can be started. If you do not see a system log, wait a few minutes and then try again.



24. Scroll through the output and note that the HTTP package was installed from the **user data** that you added when you created the instance.

25. Choose Cancel.
26. Ensure Web Server is still selected. Then, in the Actions menu, select Monitor and troubleshoot Get instance screenshot.
27. Choose **Cancel**.



❖ Instance screenshot:

Task 3: Update Your Security Group and Access the Web Server
When you launched the EC2 instance, you provided a script that installed a web server and created a simple web page. In this task, you will access content from the web server.

28. Ensure Web Server is still selected. Choose the Details tab.

29. Copy the Public IPv4 address of your instance to your clipboard.

30. Open a new tab in your web browser, paste the IP address you just copied, then press Enter.

31. Keep the browser tab open, but return to the EC2 Console tab.

32. In the left navigation pane, choose Security Groups.

33. Select Web Server security group.

34. Choose the Inbound rules tab.
    The security group currently has no inbound rules.

35. Choose Edit inbound rules, select Add rule and then configure:
    - Type: *HTTP*
    - Source: *Anywhere-IPv4*
    - Choose Save rules

36. Return to the web server tab that you previously opened and refresh the page.

Task 4: Resize Your Instance: Instance Type and EBS Volume

As your needs change, you might find that your instance is over-utilized (too small) or under-utilized (too large). If so, you can change the *instance type*. For example, if a *t2.micro* instance is too small for its workload, you can change it to an *m5.medium* instance. Similarly, you can change the size of a disk.

❖ **Stop Your Instance**

Before you can resize an instance, you must *stop* it.

When you stop an instance, it is shut down. There is no runtime charge for a stopped EC2 instance, but the storage charge for attached Amazon EBS volumes remains.

37. On the **EC2 Management Console**, in the left navigation pane, choose **Instances** and then select the **Web Server** instance.

38. In the **Instance state** menu, select **Stop instance**.

39. Choose **Stop**

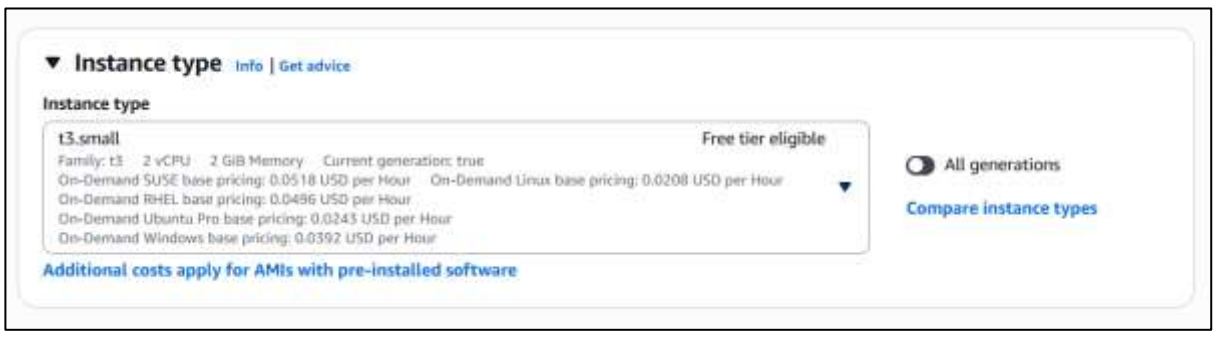
    Your instance will perform a normal shutdown and then will stop running.

40. Wait for the **Instance state** to display: *Stopped*.

❖ **Change The Instance Type and enable stop protection**

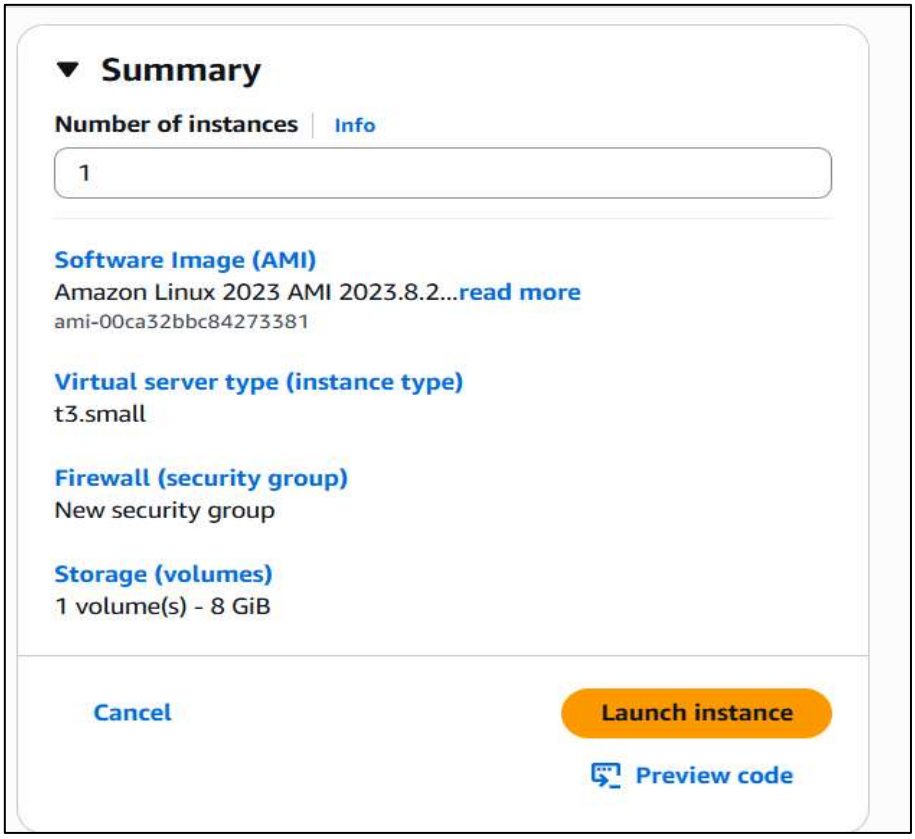
41. Select the Web Server instance, then in the **Actions** menu, select **Instance settings** **Change instance type**, then configure:
    - **Instance Type:** *t.small*
    - Choose **Apply**

    When the instance is started again it will run as a *t2.small*, which has twice as much memory as a *t2.micro* instance.



42. Select the Web Server instance, then in the **Actions** menu, select **Instance settings** **Change stop protection**. Select **Enable** and then **Save** the change.

When you stop an instance, the instance shuts down. When you later start the instance, it is typically migrated to a new underlying host computer and assigned a new *public* IPv4 address. An instance retains its assigned *private* IPv4 address. When you stop an instance, it is not deleted. Any EBS volumes and the data on those volumes are retained.



❖ **Resize the EBS Volume**

43. With the Web Server instance still selected, choose the **Storage** tab, select the name of the Volume ID, then select the checkbox next to the volume that displays.

44. In the **Actions** menu, select **Modify volume**.
    The disk volume currently has a size of 8 GiB. You will now increase the size of this disk.

45. Change the size to: 10 **NOTE**: You may be restricted from creating Amazon EBS volumes larger than 10 GB in this lab.

46. Choose **Modify**

47. Choose **Modify** again to confirm and increase the size of the volume.

Start the Resized Instance

You will now start the instance again, which will now have more memory and more disk space.

48. In left navigation pane, choose **Instances**.

49. Select the **Web Server** instance.

50. In the **Instance state** menu, select **Start instance**.

Conclusion:

From this experiment, we learned how to create and manage a virtual server (EC2 instance) on Amazon Web Services. We practiced launching the instance, attaching security groups, enabling termination protection, and adding a user data script to install a web server automatically. We also monitored the instance using AWS tools, resized its type and storage, and accessed the hosted web page through its public IP. This practical shows how EC2 provides flexibility, scalability, and reliability for running applications in the cloud. It also highlights how security groups and monitoring ensure safe and efficient operation of cloud resources.