AWS 1

CCNP Lab 6

Jason Liu

CCNP – Mr. Mason & Mr. Hansen

Period 6, 7, 8

*Lab 6: AWS Introduction*

Amazon Web Service (AWS) is a “Cloud” platform. The Cloud provides on-demand IT resources via the Internet, with the goal of providing easy and scalable access to those computing services. AWS provides many Amazon proprietary services through their AWS cloud, including computing and storage software. Specifically, they run Extensible Markup Language (XML) and JavaScript Object Notation (JSON) for their Application Programming Interfaces (API). The purpose of the Cloud is to lessen the need for businesses and organizations to require expensive hardware and maintenance of machinery in their own facilities.

The purpose of this write-up is slightly different than the previous ones. We have been introduced to the AWS Academy Cloud Foundations course. The course is intended to teach students about the basic cloud computing concepts, of which, we experienced the first 3 labs: AWS Identity and Access Management (IAM), VPS and Web Server establishment via AWS, and Amazon Elastic Compute Cloud (EC2) introduction.

Senior administrator and teacher Jeffrey Mason believes that being able to explain a concept demonstrates having digested the information well enough. Thus, this lab will comprise of our analysis and individual interpretation to explain the content of the listed labs above in a user-friendly manner that appeals to inexperienced audiences.

Lab 1: AWS IAM

**Objective**

Lab one presents an introduction to the Amazon service that manages users, user permissions, and security credentials within the cloud. It is called AWS Identity and Access Management (IAM). Security is important as Amazon is not responsible for internal customer decisions within the Cloud and their responsibility to implement their own security measures. In this lab, you will enter a pre-configured scenario to explore users, user permissions, IAM policies, adding users to groups, and assigning permissions via an Admin role. The lab will also integrate an outside perspective using an IAM sign-in link to view the effects of user permissions via a standard internet web browser.

**Key Terms**

**Identity and Access Management (IAM):** AWS service that specifies user permissions, IAM policies, and who has access and under what conditions.

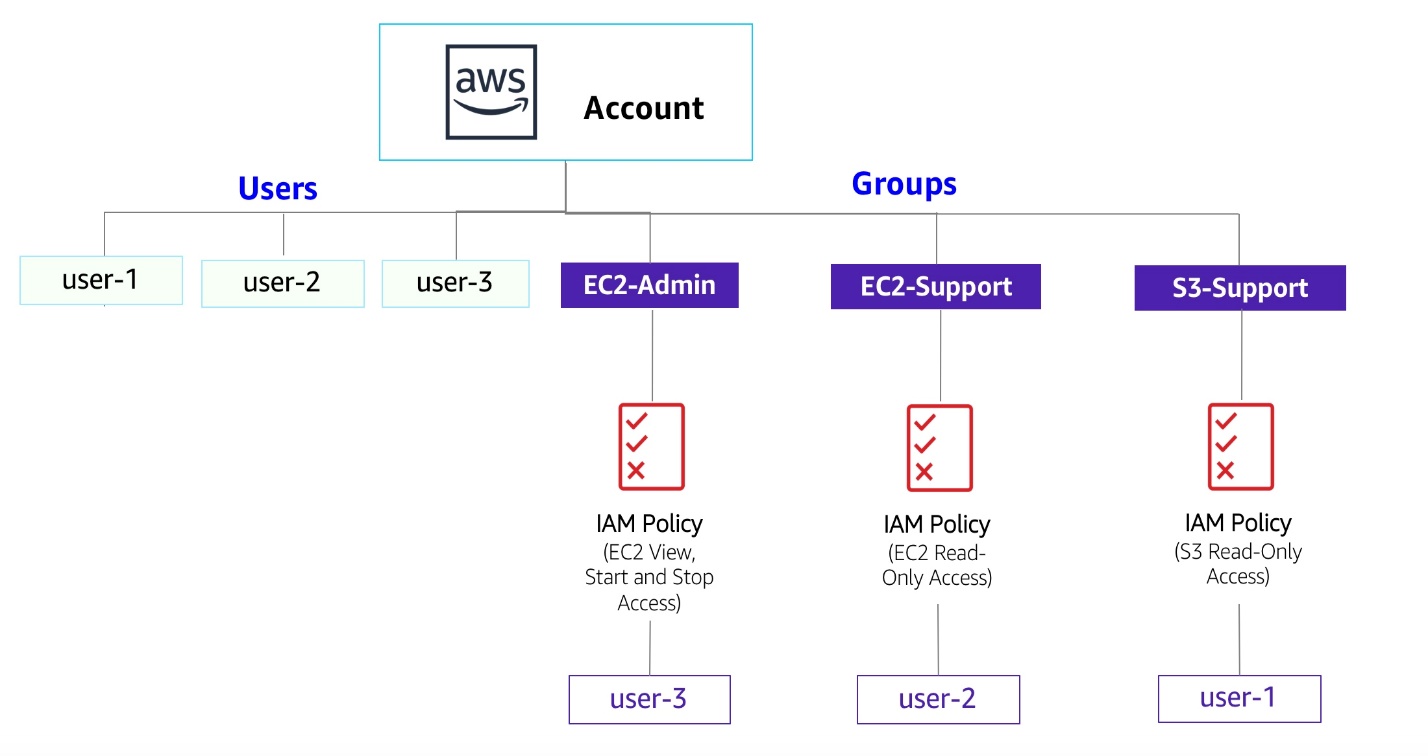
**Users:** You can create Users, a profile per user, and assign them individual security credentials (access keys, passwords, and multi-factor authentication devices). You can also manage which operations they are allowed to perform.

**Roles:** Rather than be uniquely associated to one user, a Role is an identity that contains a set of permissions policies, in which any number of users can assume said Role.

**Instance:** A virtual server in the AWS cloud. Configure and managed by a computing service, such as EC2.

**Elastic Compute Cloud (EC2):** AWS service that runs virtual computers for the user’s computer applications.

**Simple Storage Service (S3):** AWS service that provides data storage in the cloud. It has a scalable storage infrastructure.

**Topological Overview**

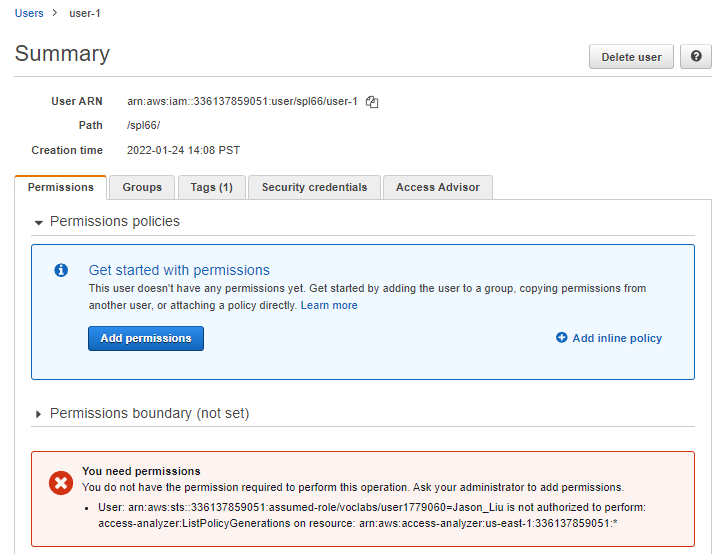
**General Procedure**

1. View and explore users and groups in IAM.
2. View and understand their policies and permissions.
3. Assign different users to different groups.
4. Sign-in and test user permissions via an internet browser.
5. Assign user permissions via the Admin role.

**Accessing Lab Version of AWS Management Console**

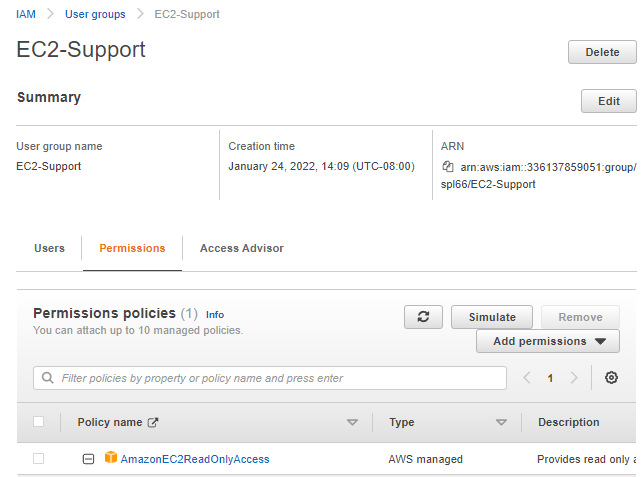
As lab contents are AWS course propriety, you must access their interactive lab via the AWS Academy Cloud Foundations course, Module 4: Lab 1 – Introduction to AWS IAM.

**Task 1: View and explore users and groups in IAM**

1. Presented in the AWS Management Console, select **IAM** from the Services Menu.
2. In the navigation panel on the left, choose Users.
3. Three users have already been configured.
4. Select any user.
   1. See permissions tab. Notice the user does not have any permissions.
   2. See groups tab. Notice the user is not a member of any groups.
   3. See security credentials tab. Notice the user is assigned a “console password.”

User-1 Summary Page, Permissions tab.

1. In the navigation panel on the left, choose User Groups.
2. Three groups have already been configured.
3. Select **EC2-Support** group.
   1. See permissions tab. Notice the group has a managed policy associated with it, called **AmazonEC2ReadOnlyAccess**.
   2. Press the (+) icon to expand policy details.

EC2-Support User Group Summary and Policy Details

**Policies:** A policy defines what actions are allowed or denied for specific AWS resources.

1. In the navigation panel on the left choose User Groups. Repeat step 7 to explore the other 2 groups’ permissions. The basic structure of the statements in an IAM Policy is:
   1. **Effect** says whether to Allow or Deny the permissions.
   2. **Action** specifies the API calls that can be made against an AWS Service.
   3. **Resource** defines the scope of entities covered by the policy rule, such as an Amazon S3 bucket, an Amazon EC2 instance, or \* which means any resource.

**Managed Policies:** Pre-built policies (by AWS or administrators) that can be attached to IAM Users and Groups. Changing the policy immediately applies its effects to all Users and Groups attached

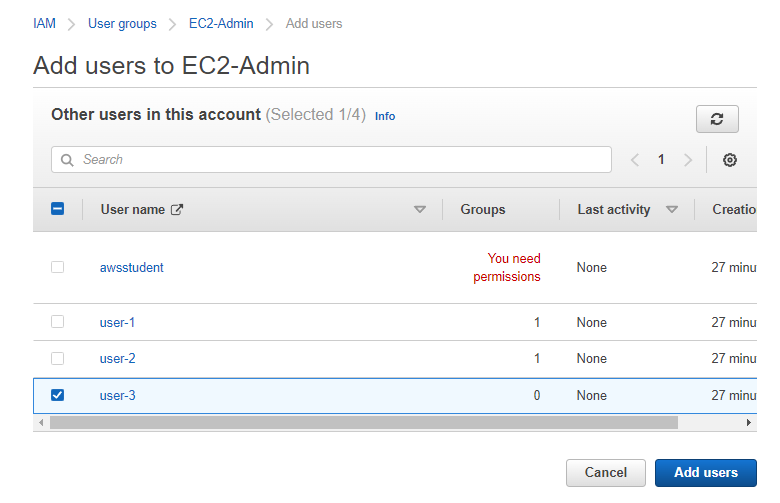
**Inline Policy:** A policy assigned to just one User or Group. Often used to apply permissions for one-off situations.

**Task 2: Add Users to Groups**

You will now attempt to organize users into the following manner:

|  |  |  |
| --- | --- | --- |
| User | Group | Permissions |
| user-1 | S3-Support | Read-Only access to Amazon S3 |
| user-2 | EC2-Support | Read-Only access to Amazon EC2 |
| user-3 | EC2-  Admin | View, Start and Stop Amazon EC2 instances |

1. We will now add users to the groups. In the left navigation panel, choose User Groups.
   1. Go to **S3-Support** group.
   2. Select the Users tab.
   3. Select Add users.
      1. Select the box next to **user-1**.
   4. Finish by selecting Add users at the bottom of the screen.
2. Repeat step 9 for **EC2-Support** group, adding **user-2** instead.
3. Repeat again for **EC2-Admin** group, adding **user-3** instead.

Each Group should now have a 1 in the Users column for the number of Users in each Group.

Adding user-3 to EC2-Admin group.

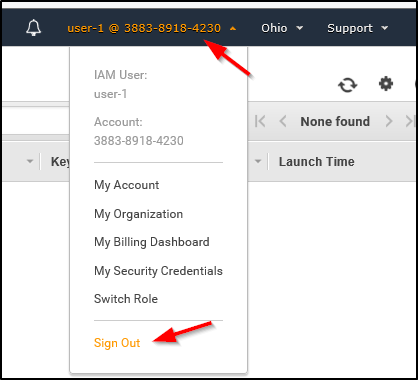
**Task 3: Test User Permissions**

You will now sign into each IAM user to test permissions.

1. In the navigation panel on the left, choose Dashboard.
   1. An IAM users sign-in link is displayed on the right. It will look similar to: *https://123456789012.signin.aws.amazon.com/console*
   2. Copy this link.
   3. Past this link into a standard incognito internet browser (e.g Mozilla Firefox, Google Chrome, etc.)
2. You will now enter user-1.
   1. Sign-in with:
      1. IAM username: *user-1*
      2. Password: *Lab-Password1*
   2. In the Services menu, choose S3.
   3. Enter the S3 bucket and browse its 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 that the bucket does not contain any objects.

* 1. In the Services menu, choose EC2.
  2. In the left navigation panel, choose Instances.

You will *NOT* be able to see any instances. You are not authorized to perform this operation because this user has not been granted any permissions to access Amazon EC2.

1. Sign out of user-1.

Signing out of user-1.

1. You will now enter user-2.
   1. Copy the link into your browser again.
   2. Sign-in with:
      1. IAM username: *user-2*
      2. Password: *Lab-Password2*
   3. In the Services menu, choose EC2.
   4. In the navigation panel on the left, choose Instances.
   5. Select the instance named *LabHost*.
   6. In the Instance State menu above, select Stop Instance.
   7. Stop the Instance.

You are now able to see an Amazon EC2 instance because you have Read Only permissions. However, you will receive an error stating *You are not authorized to perform this operation* because you cannot make changes.

1. You will now enter user-3.
   1. Copy the link into your browser again.
   2. Sign-in with:
      1. IAM username: *user-3*
      2. Password: *Lab-Password3*
   3. In the Services menu, choose EC2.
   4. In the navigation panel on the left, choose Instances.
   5. Select the instance named *LabHost*.
   6. In the Instance State menu above, select Stop Instance.
   7. Stop the Instance.

User-3 (Admin) has the permissions to manage instances. The instance will enter the *stopping* state and will shut down.

**Conclusion**

AWS IAM is a comfortable introduction to the AWS interface, understanding users, groups, policies, and permissions. We also learned how to add users to groups that have managed policies, as well as figuring out how to test those permissions. Exploring pre-created IAM users and groups gave us a fundamental experience in navigating the AWS interface.

Lab 2: AWS VPC

**Objective**

Lab two presents the Virtual Private Cloud (VPC), a service that provides a logistical platform in which the user can control all aspects of the virtual networking environment, including connectivity, security, subnets, and a range of resources. In this lab, you will create a VPC, create pre-defined subnets, establish security groups, and customize an EC2 instance to run a web server to launch via the VPC.

**Key Terms**

**Amazon Virtual Private Cloud (Amazon VPC):** A cloud that enables you to launch Amazon Web Service resources into a virtual network that you defined. It resembles traditional networking topologies. It is scalable and can span multiple Availability Zones.

**Availability Zone (AZ):** A discrete data centers with redundant power, networking, and connectivity in an AWS Region.

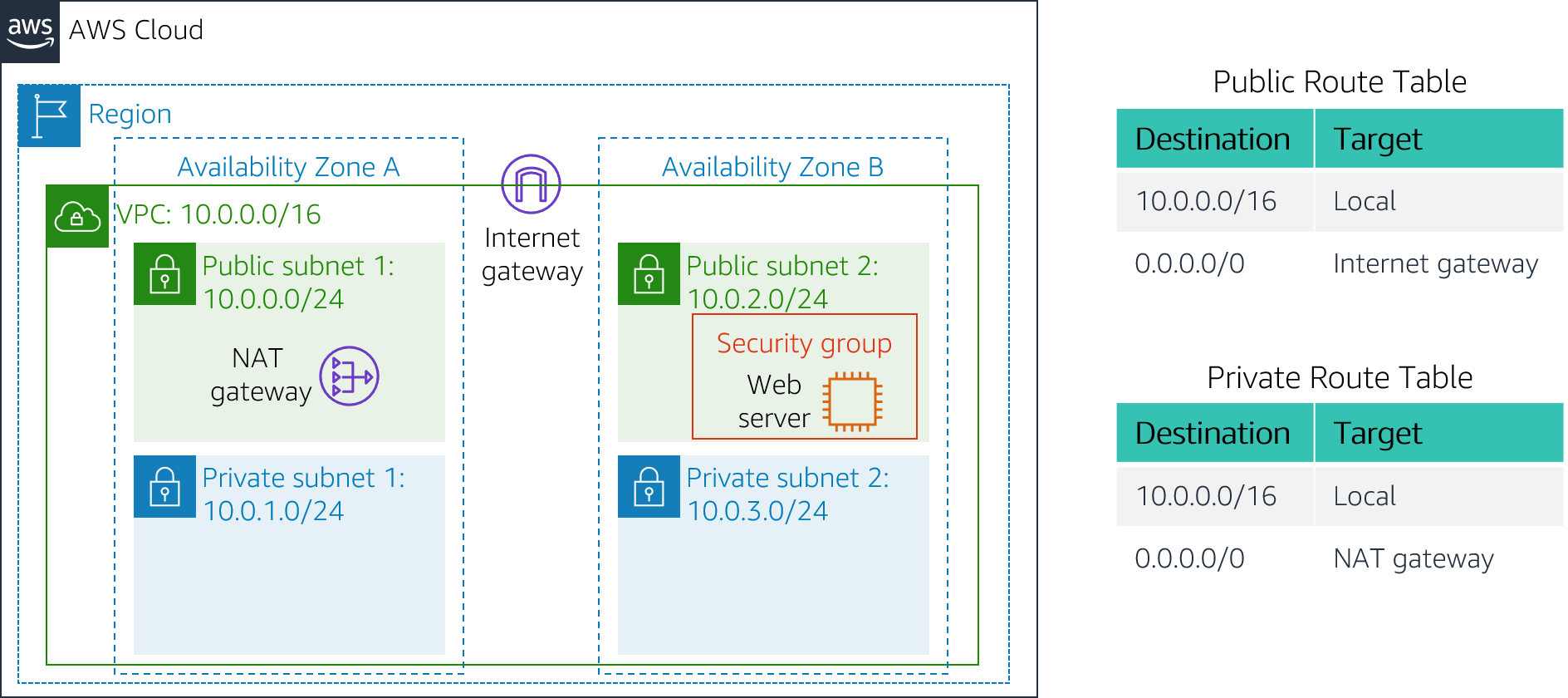
**Internet gateway (IGW):** A VPC component that allows communication between instances in your VPC and the Internet.

**Public Subnet:** A subnet IP group that has traffic routed to an Internet Gateway.

**Private Subnet:** A subnet IP group that does not route to the Internet Gateway.

**Instance:** A virtual server in the AWS cloud. Configure and managed by a computing service, such as EC2.

**Elastic Compute Cloud (EC2):** AWS service that runs virtual computers for the user’s computer applications.

**Topological Overview**

**General Procedure**

1. Use the VPC wizard to create an Internet Gateway and two subnets for one Availability Zone.
2. Create two additional subnets in a second Availability Zone.
3. Create a VPC Security Group.
4. Launch a Web Server Instance in EC2 into the new VPC.

**Accessing Lab Version of AWS Management Console**

As lab contents are AWS course propriety, you must access their interactive lab via the AWS Academy Cloud Foundations course, Module 5: Lab 2 – Build your VPC and Launch a Web Server.

**Task 1: Create a VPC**

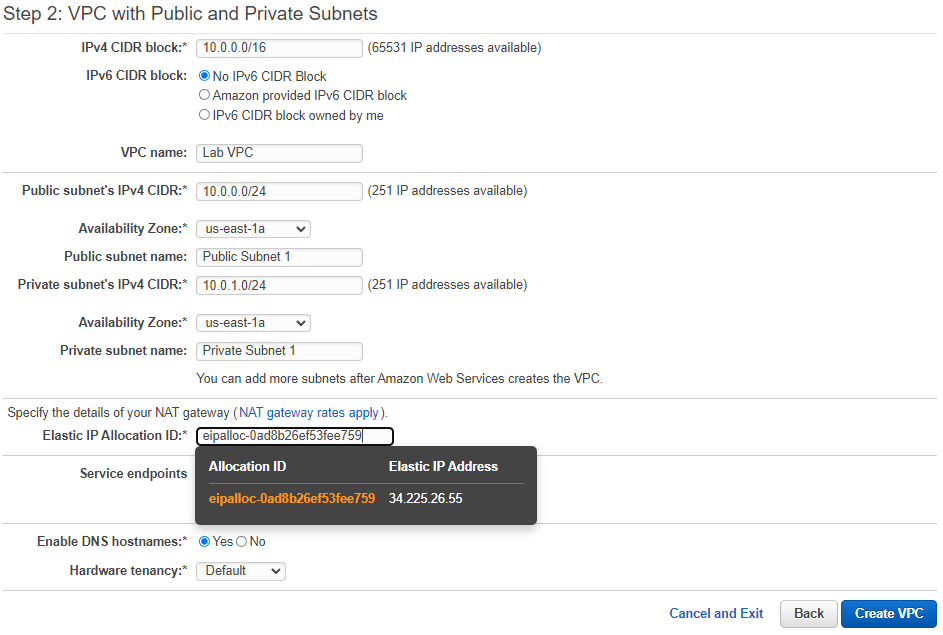
1. Presented in the AWS Management Console, select **VPC** from the Services Menu.
2. Choose Launch VPC Wizard.
3. In the navigation panel on the left, choose VPC with Public and Private Subnets.
4. Choose Select then configure the following:
   1. VPC name: *Lab VPC*
   2. Availability Zone: Select the *first* Availability Zone
   3. Public subnet name: *Public Subnet 1*
   4. Availability Zone: Select the *first* Availability Zone (the same as used above)
   5. Private subnet name: *Private Subnet 1*
   6. Elastic IP Allocation ID: Select the displayed IP address in the dropdown menu.
5. Finish configuration.

The Public Subnet has a CIDR of 10.0.0.0/24, which means that it contains all IP addresses starting with 10.0.0.x.

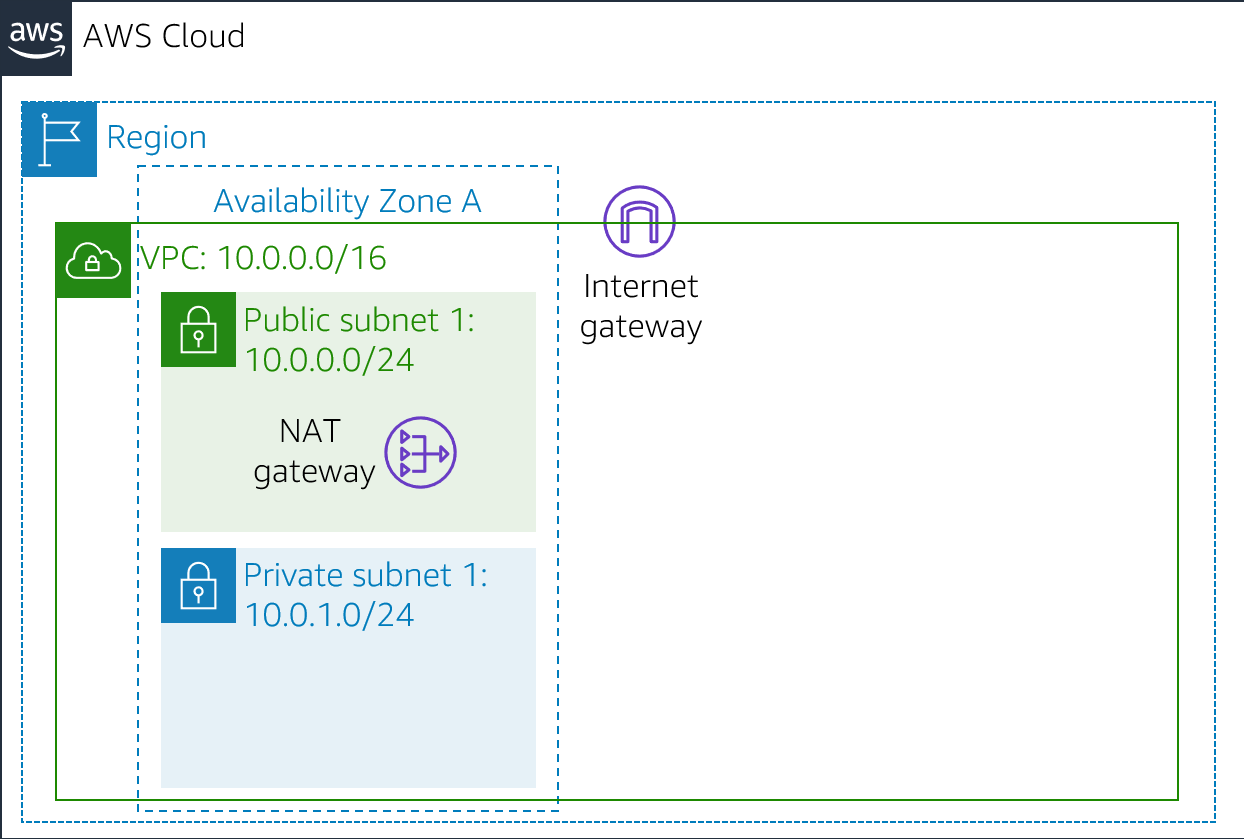
The Private Subnet has a CIDR of 10.0.1.0/24, which means that it contains all IP addresses starting with 10.0.1.x.

**Classless Inter-Domain Routing (CIDR):** The CIDR block determines the maximum number of IP addresses that can be created and used in the VPC.

The wizard will also create a **NAT Gateway**: A gateway used to provide internet connectivity to EC2 instances in the private subnets.



Finished Configuration before launch.

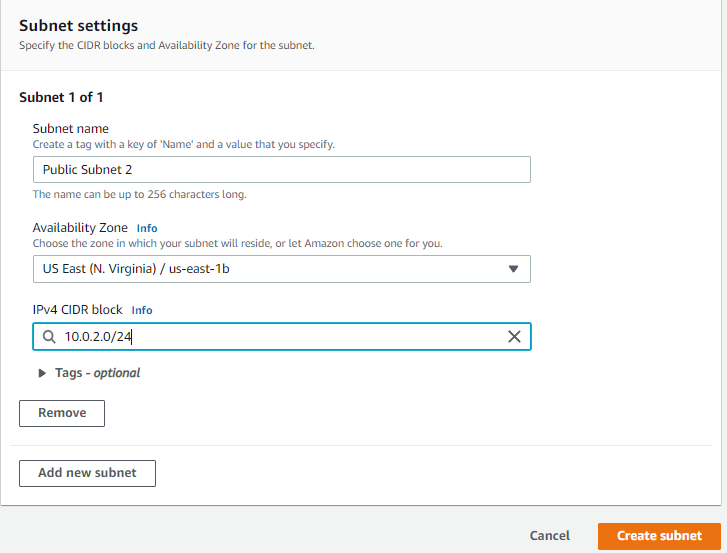


Current Topology Progress: 2 subnets and NAT gateway established.

**Task 2: Create 2 Additional Subnets**

You will create two additional subnets for a second Availability Zone.

1. In the navigation panel on the left, choose Subnets.
2. You will create a second Public Subnet. Choose Create Subnet and configure:
   1. VPC ID: *Lab VPC*
   2. Subnet name: *Public Subnet 2*
   3. Availability Zone: Select the *second* Availability Zone
   4. IPv4 CIDR block: 10.0.2.0/24

This subnet will have all IP addresses starting with 10.0.2.x.

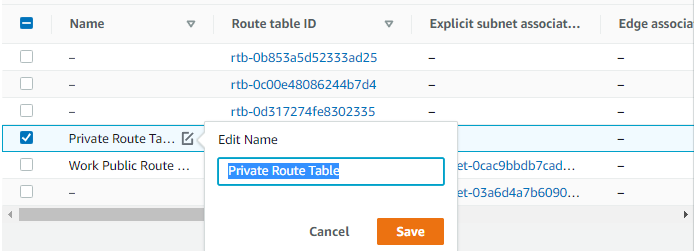
Subnet settings for the second Public Subnet.

1. You will create a second Private Subnet. Choose Create Subnet and configure:
   1. VPC ID: *Lab VPC*
   2. Subnet name: *Private Subnet 2*
   3. Availability Zone: Select the *second* Availability Zone
   4. CIDR block: 10.0.3.0/24

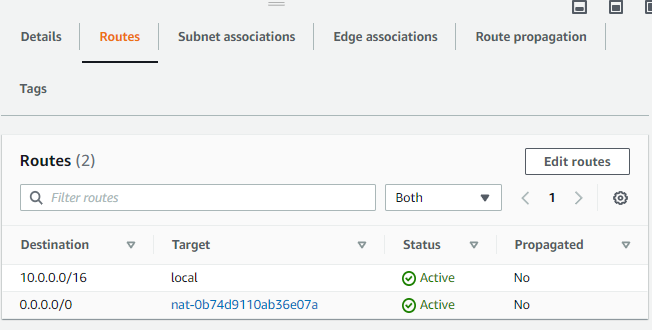
This subnet will have all IP addresses starting with 10.0.3.x.

You will now route the Private Subnets to the NAT Gateway. This way they can connect to the Internet while still keeping the resources private. Routing is done via a routing table.

**Route Table:** An internal table that contains a rules, called routes, that are used to determine and control where traffic is directed in a network. Each subnet will have a routing table.

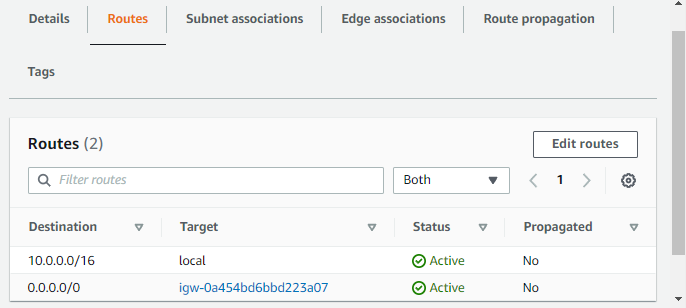
1. In the navigation panel on the left, choose Route Tables.
   1. Select the checkbox next to the route table with *Main = Yes* and *VPC = Lab VPC*.
   2. Name the route table by pressing the edit name icon upon hovering the hyphen and name it: *Private Route Table* and save.

Naming the main Route Table of the Lab VPC: *Private Route Table*

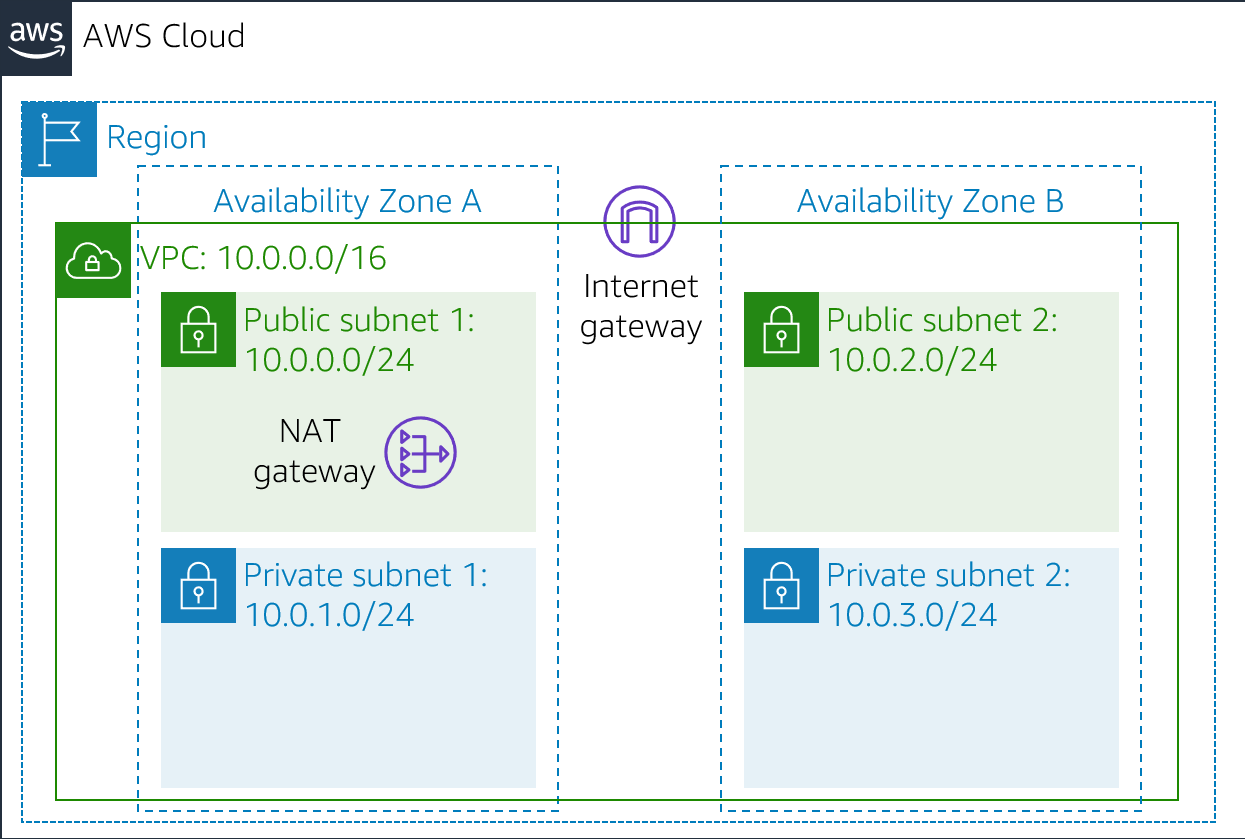
If you select the Routes tab in the lower panel, notice that the Destination 0.0.0.0/0 is set to Target nat-xxxxxxxx. This means that traffic destined for the internet (0.0.0.0/0) will be sent to the NAT Gateway. The NAT Gateway will then forward the traffic to the internet.

Routes tab. Note Destination address and NAT Target. This will forward to the NAT gateway.

1. You will now associate this route table to the Private Subnets. In the lower panel, select the Subnet Associations tab.
   1. Choose Edit Subnet Associations.
   2. Select the checkbox next to both Private Subnet 1 and Private Subnet 2.
   3. Save the associations.
2. You will now configure the Route Table for Public Subnets.
   1. Select the checkbox next to the route table with ***Main = No*** and ***VPC = Lab VPC*.**
   2. **Name the route table: *Public Route Table* and save.**

If you select the Routes tab in the lower panel, notice that the Destination 0.0.0.0/0 is set to Target igw-xxxxxxxx, which is the Internet Gateway. This means that internet-bound traffic will be sent straight to the internet via the Internet Gateway.

Routes tab. Note Destination address and IGW Target. This will forward to the Internet gateway.

1. You will now associate this route table to the Public Subnets. In the lower panel, select the Subnet Associations tab.
   1. Choose Edit Subnet Associations.
   2. Select the checkbox next to both Public Subnet 1 and Public Subnet 2.
   3. Save the associations.

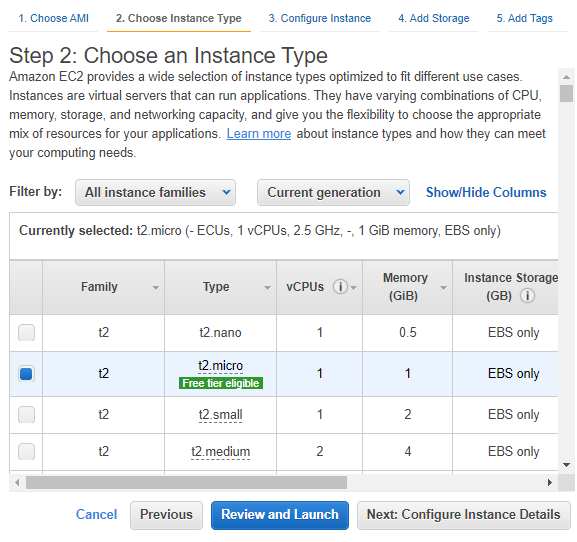
Current Topology Progress: 2 Availability Zones Created.

**Task 3: Create a Security Group**

You will now create a VPC security group, which acts as a virtual firewall. You can apply several security groups per launched instance.

1. In the navigation panel on the left, choose Security Groups.
2. Select “Create Security Group” and then configure:
   1. Security group name: *Web Security Group*
   2. Description: *Enable HTTP access*
   3. VPC: *Lab VPC*
3. In the Inbound Rules panel, select Add Rule and configure:
   1. Type: *HTTP*
   2. Source: *Anywhere-IPv4*
   3. Description: *Permit web requests*
4. Finalize and create security group by selecting “Create Security Group.”

**Task 4: Launch a Web Server Instance**

You will now create and launch an Amazon EC2 instance into the VPC. It will act as a web server.

1. Back in the Services menu, choose EC2.
2. You will select an Amazon Machine Image (AMI) for your instance. Select Launch Instance.

**Amazon Machine Image (AMI):** The information required to launch an instance. You *must* specify the AMI and its details to launch the instance.

**Instance Type:** Defines the hardware resources assigned to the instance.

1. At the top, choose “Select” in the first row of Amazon Linux 2.
2. Select *t2.micro*. This is the instance type. (See figure to the right).

AMI configuration: Instance Type

1. You will now configure the instance to launch in a Public Subnet of the second Availability Zone. Select “Next: Configure Instance Details” and configure:
   1. Network: *Lab VPC*
   2. Subnet: *Public Subnet 2*
   3. Auto-assign Public IP: Enable
2. The following script will be run when the instance launches for the first time. It will load and configure a PHP web application.
   1. At the bottom of the page, expand the Advanced Details tab.
   2. Copy and paste this code into the User data box.

#!/bin/bash

# Install Apache Web Server and PHP

yum install -y httpd mysql php

# Download Lab files

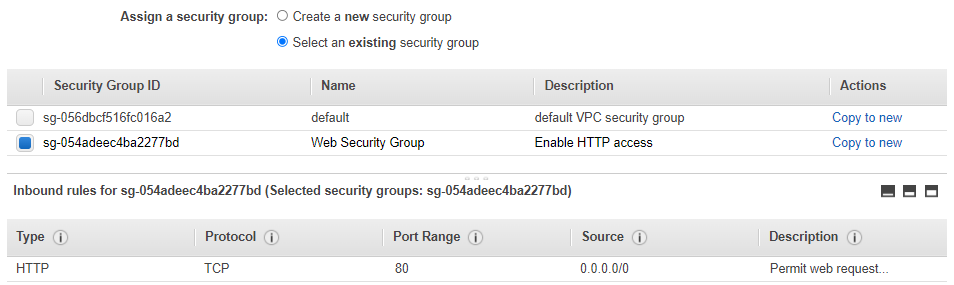
wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-100-ACCLFO-2/2-lab2-vpc/s3/lab-app.zip

unzip lab-app.zip -d /var/www/html/

# Turn on web server

chkconfig httpd on

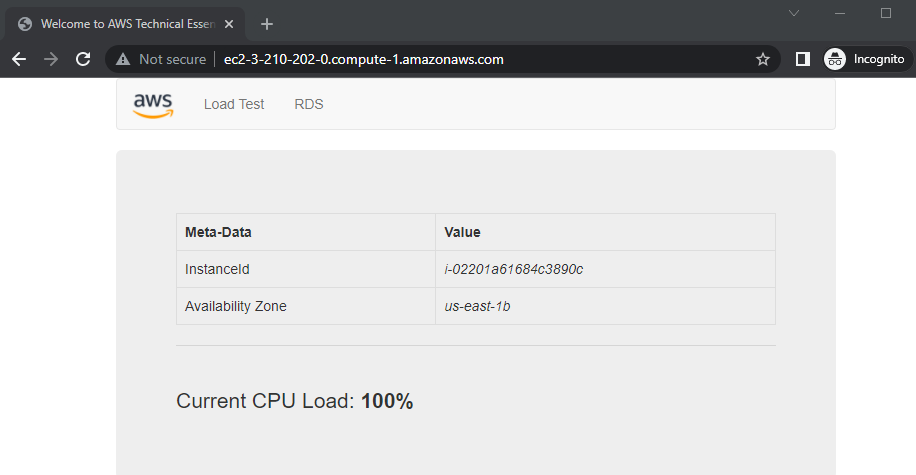
service httpd start

1. Proceed through instance configuration.
   1. Choose “Next: Add Storage.” Default settings will be used.
   2. Choose “Next: Add Tags.” Tags identify resources. Choose “Add Tag” and configure:
      1. Key: *Name*
      2. Value: *Web Server 1*
   3. Choose “Next: Configure Security Group.” Configure the instance to use the Web Security Group that you created earlier.
      1. Select “Select an existing security group.”
      2. Select “Web Security Group.” This is the Security Group you created. It permits HTTP access.

Selecting an existing security group to configure onto the instance.

1. Choose “Review and Launch.” Continue through the clarification messages.
   1. Review and select Launch.
   2. In the Select keypair dialog, select the acknowledgment checkbox.
2. See statuses by viewing the isntances.
   1. Wait until Web Server 1 shows 2/2 checks passed in the “Status Checks” column. You may have to refresh every so often.
3. You will now connect to the web server running on the EC2 instance.
   1. Select Web Server 1.
   2. At the bottom of the page, copy the **Public DNS (IPv4)** value show in the Description tab.
   3. Paste the **Public DNS** value into a standard incognito internet browser.

You will see a web page displaying the AWS instance and its metadata.



Web page of the instance’s metadata. Copied from the public DNS address.

**Conclusion**

The AWS VPC is an important fundamental service for networking, with a new approach relative to the traditional hardware methods. In this lab we learned about Availability Zones and subnets, while ensuring security through the configuration of using the NAT gateway, Security Groups, and private subnets. There was also an early introduction to the EC2 instance configuration, going through and understanding the process of AMI configurations.

Lab 3: AWS EC2

**Objective**

Lab three is an introduction to Amazon Elastic Compute Cloud (EC2) , the web service that provides resizable compute capacity in the cloud. In other words, the user has control of all computing resources and parameters for their server instances. This lab will go in further depth to configure and manage those instances within the EC2 environment and navigate the interface step-by-step.

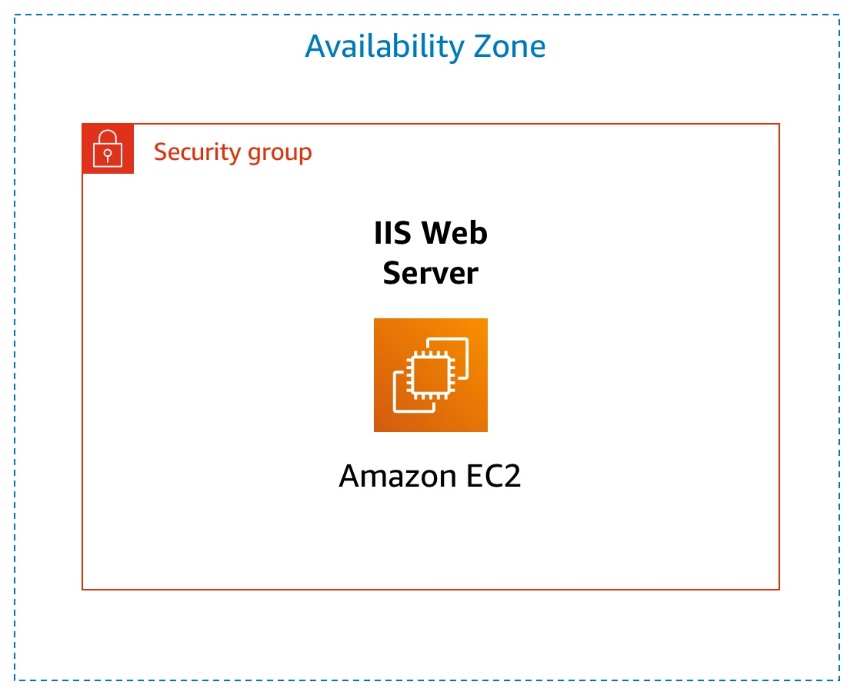
**Key Terms**

**Elastic Compute Cloud (EC2):** AWS service that runs virtual computers for the user’s computer applications.

**Instance:** A virtual server in the AWS cloud. Configure and managed by a computing service, such as EC2.

**Amazon Machine Image (AMI):** The details which provides the information required to launch an instance. You *must* specify the AMI and its details to launch the instance.

**Instance Type:** Defines the hardware resources assigned to the instance. Preset instance types vary memory, storage, and CPU.

**Topological Overview**

**General Procedure**

1. Configure and launch an Instance.
2. Monitor the Instance.
3. Update security group and access the web server.
4. Resize Instance parameters.
5. Explore EC2 Limits.
6. Test Termination Protection.

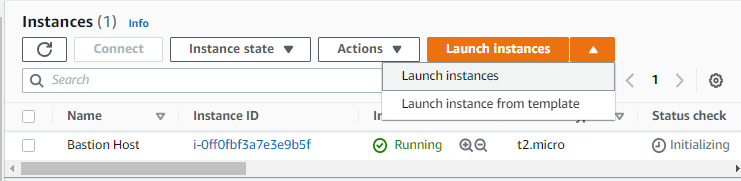
**Accessing Lab Version of AWS Management Console**

As lab contents are AWS course propriety, you must access their interactive lab via the AWS Academy Cloud Foundations course, Module 6: Lab 3 – Introduction to Amazon EC2.

**Task 1: Launch an Amazon EC2 Instance**

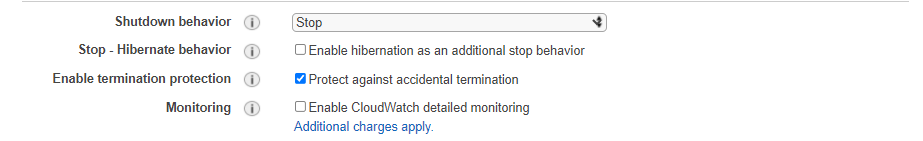
You will launch an Amazon EC2 instance with termination protection. Also, you will deploy your instance with a User Data script that will allow you to create a simple web server.

**Termination protection:** Prevents accidental termination of an EC2 instance.

1. On the Services menu, choose EC2.
2. Choose Launch Instance, and then Launch Instances.

Launch Instance. .Process

An AMI contains a template for the root volume. The root volume is used to boot the instance. AMI also contains permissions for the accounts.

1. At the top, choose “Select” in the first row of Amazon Linux 2.
   1. Select t2.micro. It has 1 virtual CPU and 1 GiB of memory.
2. Choose “Next: Configure Instance Details.”
3. You will now configure the instance details.
   1. For Network, select *Lab VPC*. The Lab VPC was created in the lab generation. It has 2 public subnets in two different availability zones.
4. For “Enable termination protection”, select “Protect against accidental termination.”

Enable Termination Protection.

1. Expanding the Advanced Detail tab, paste the following into the User Data field.
   1. Amazon Linux runs on shell script.

#!/bin/bash

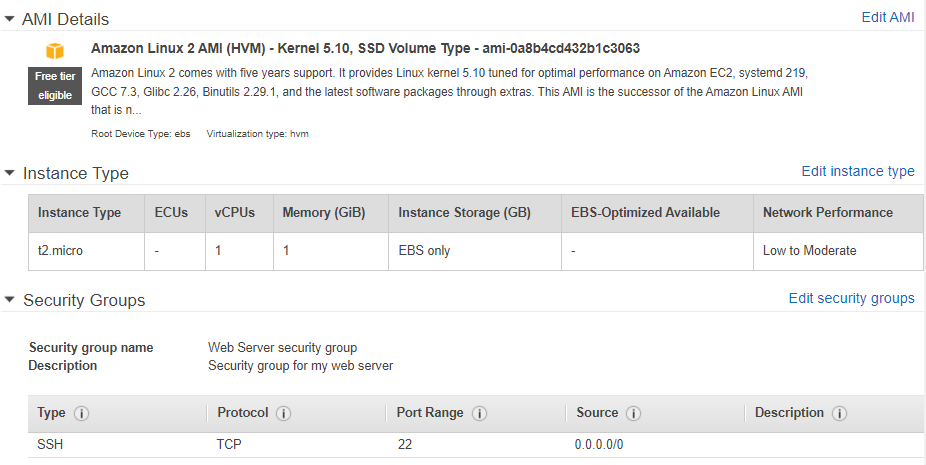
yum -y install httpd

systemctl enable httpd

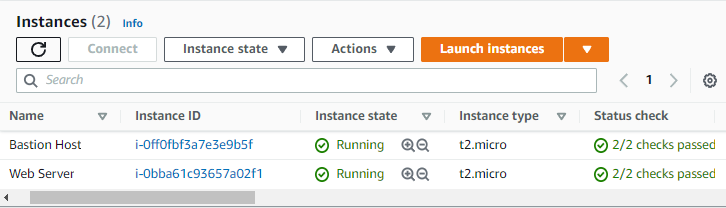
systemctl start httpd

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

* 1. The script will Install an Apache web server, start on boot, activate it, and create a web page for it.

1. Proceed through instance configuration.
   1. Choose “Next: Add Storage.” EC2 stores data via a virtual disk called Elastic Book Store. Continue.
   2. Choose “Next: Add Tags.” Name the tag by choosing “Add Tag” and configuring:
      1. Key: *Name*
      2. Value: *Web Server*
   3. Choose “Next: Configure Security Group.” As stated in the previous lab, a security group acts as the virtual firewall. Configure:
      1. Security group name: *Web Server security group*
      2. Description: *Security group for my web server*
   4. Choose “Review and Launch.” Proceed without a key pair. Launch instances.

Final Review before Launching Instance.

1. Finalize.
   1. Wait for your instance to display:
      1. Instance State: *running*
      2. Status Checks: *2/2 checks passed*
   2. Description tab displays detailed data of the running instance.
   3. Details tab displays configuration settings.

Web Server Instance is properly running and has passed the status checks.

You have now configured an instance with termination protection.

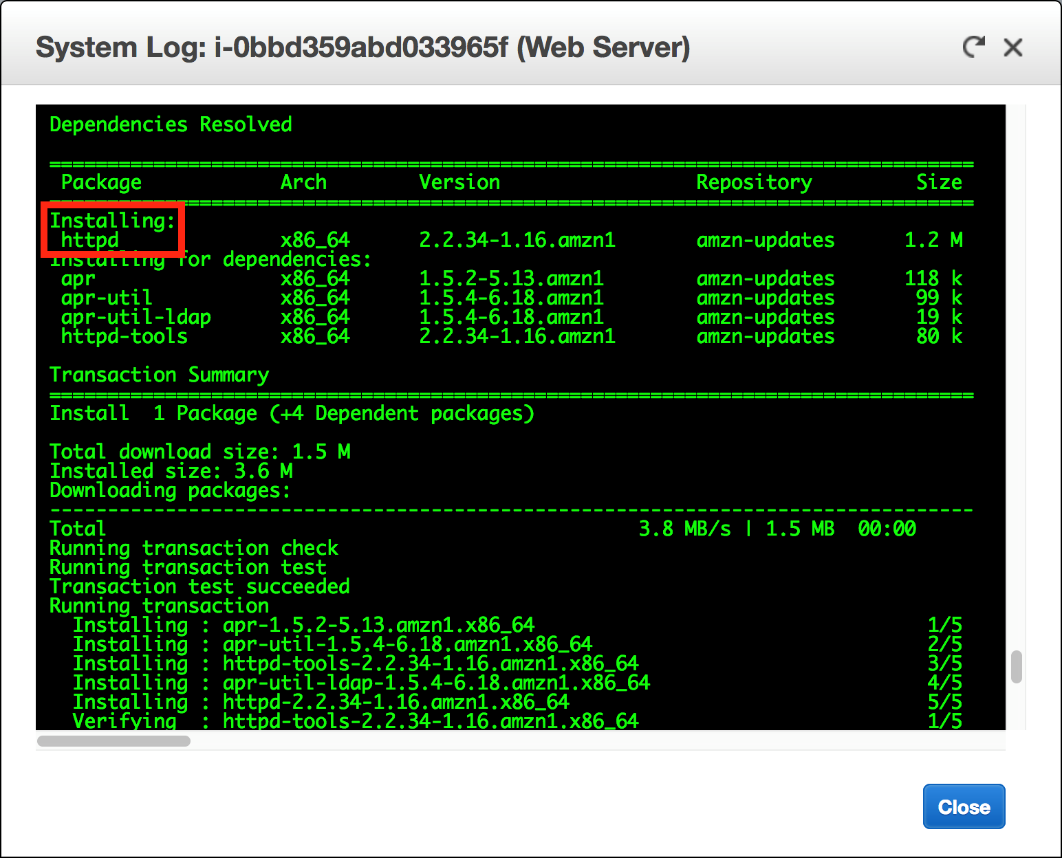
**Task 2: Monitor the Instance**

You will now learn how to monitor the instance, as it is important in maintaining its reliability, availability, and performance.

1. View tabs.
   1. Choose the Status Checks tab. This will give quick status tests for certain hardware and software issues. This includes System reachability and Instance reachability.
   2. Choose the Monitoring tab. This tab shows Amazon CloudWatch metrics, including a graph in the expanded view. EC2 sends metrics to Amazon CloudWatch.

**Amazon CloudWatch:** A monitoring and management services that provides data for Amazon applications and infrastructure resources, including AWS.

1. In the Actions menu, select “Monitor and troubleshoot.” Then, get system log.
   1. The System Log displays the console output of the instance.



System Log Output. Note the HTTP package from the User Data script during creation.

1. In the Actions menu, select “Monitor and troubleshoot” again. Then, get instance screenshot.
   1. This shows the Amazon EC2 instance console in physical presentation.



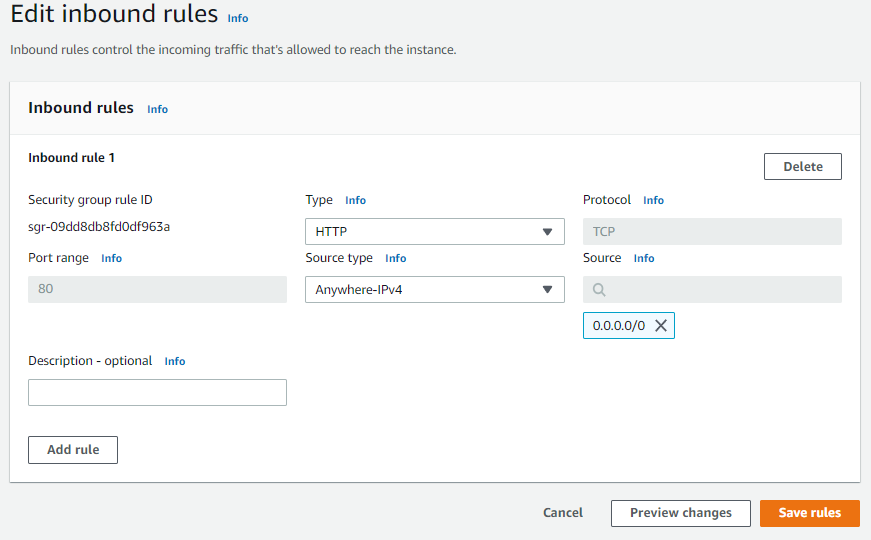
Instance Screenshot.

**Task 3: Update Security Group and Web Server**

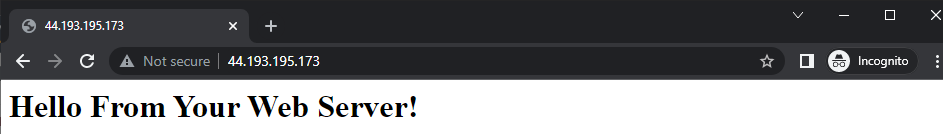
You will now access contents from the web server.

1. You will first try to access the web server. Access the Details Tab by selecting the instance.
   1. Copy the IPv4 Public IP of your instance to your clipboard.
   2. Open a standard internet web browser and paste the IP address.

You should not be able to access the Web Server. The Security Group does not permit HTTP, also known as port 80. This is proof of the firewall function.

1. Return to EC2 Management Console tab. In the navigation panel on the left, choose Security Groups and select the Web Server security group.
   1. Choose Inbound rules tab. Choose Edit inbound rules and configure:
      1. Type: HTTP
      2. Source: Anywhere-IPv4
   2. Save rules. You just edited the security group.

Editing inbound rules for instance.

1. Refresh the web page. You should see the message *Hello From Your Web Server!*

**Task 4: Resize your Instance**

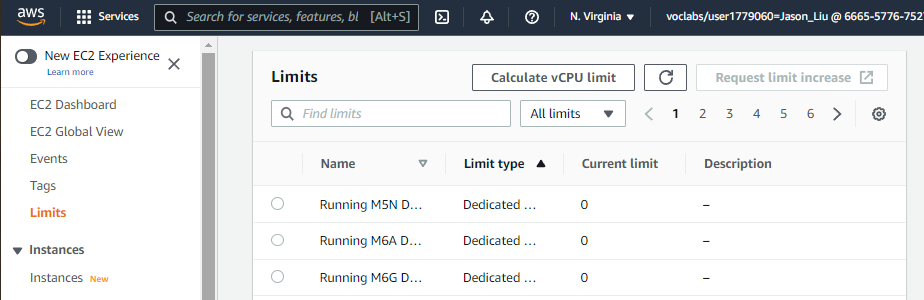
You will now change the instance type.

1. To resize an instance, you must stop it first.
   1. In EC2, on the navigation panel on the left, choose Instances. “Web Server should already be selected.
   2. Under the Instance State menu, select Stop Instance, and then select Stop.
   3. Wait for Instance State to display: “Stopped.”
2. In the Actions menu, select “Instance Settings > Change Instance Type,” and configure:
   1. Instance Type: *t2.small*
   2. Apply. This has twice as much memory as a *t2.micro* instance.
3. You will now resize the EBS Volume.
   1. In the navigation panel on the left, choose Volumes.
   2. In the Action menu, select Modify Volume.
   3. You will increase the size of the disk. It is currently 8 GiB. Change the size to 10 GiB.
   4. Finalize by selecting Modify and confirm.
4. You will now start the instance again. In the navigation panel on the left, choose Instances.
   1. Under the Instance State menu, select Start Instance, and then select Start.

You have successfully started the instance with more memory and disk space.

**Task 5: Explore EC2 Limits**

EC2 provides a range of resources, such as instances, volumes, images, and snapshots as explored in the previous tasks. There are limits of these resources per region.

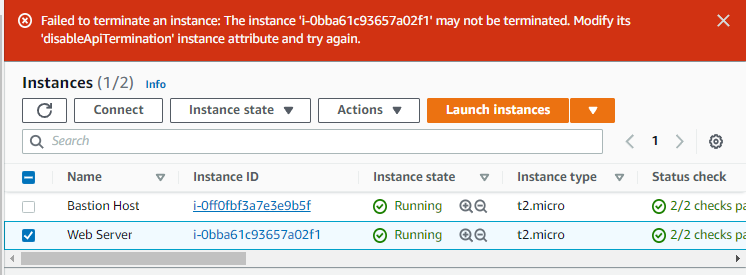
1. In the navigation panel on the left, choose Limits.
   1. From the drop-down list, select Running Instances.You can now see the limits of your resources.

Limits menu on EC2 Management Console

**Task 6: Test Termination Protection**

Recall that you have configured termination protection on the instance. You will now try to delete your instance.

**Terminated Instance:**A deleted instance. You can delete when you no longer need it. You cannot restart or interact with the instance after its termination.

1. In the navigation panel on the left, choose Instances.
2. Under the Instance State menu, select “Terminate Instance.”
   1. Choose Terminate.

You will face the error message: *Failed to terminate the instance i-1234567xxx. The instance 'i-1234567xxx' may not be terminated. Modify its 'disableApiTermination' instance attribute and try again.*

Termination Protection prevented the termination of the instance. You will have to disable the Termination Protection.

1. In the Actions menu, select “Instance settings > Change termination protection.”
   1. Uncheck the box next to Enable.
   2. Save.

You can now terminate the instance.

1. Under the Instance state menu, select Terminate instance.
   1. Choose Terminate.

You have successfully terminated the instance.

**Conclusion**

The AWS EC2 is a resourceful and essential computing tool for utilizing the potential of AWS. In this lab, we have successfully understood and configured each component of the instance configuration, as well as the ability to edit these parameters at a later time. Also, we have learned how to monitor the status and conditions of the instance. This is helpful in troubleshooting and being informed of the current condition of the system. We also explored one of the security measures: Termination Protection. As a component of the security group, termination protection is significant in preventing accidental data loss. In conclusion, we explored and interacted with intrinsic parts of the EC2 computing.