Introduction to Amazon EC2

Overview

This lab provides you with a basic overview of launching, resizing, managing, and monitoring an Amazon EC2 instance.

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers.
Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon's proven computing environment. Amazon EC2 reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as your computing requirements change.

Amazon EC2 changes the economics of computing by allowing you to pay only for capacity that you actually use. Amazon EC2 provides developers the tools to build failure resilient applications and isolate themselves from common failure scenarios.

Topics covered

By the end of this lab, you will be able to:

- Launch a web server 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
- Terminate your EC2 instance

Start Lab

Task 1: Launch Your Amazon EC2 Instance

In this task, you will launch an Amazon EC2 instance with *termination protection*. Termination protection prevents you from accidentally terminating an EC2 instance. You will deploy your instance with a User Data script that will allow you to deploy a simple web server.

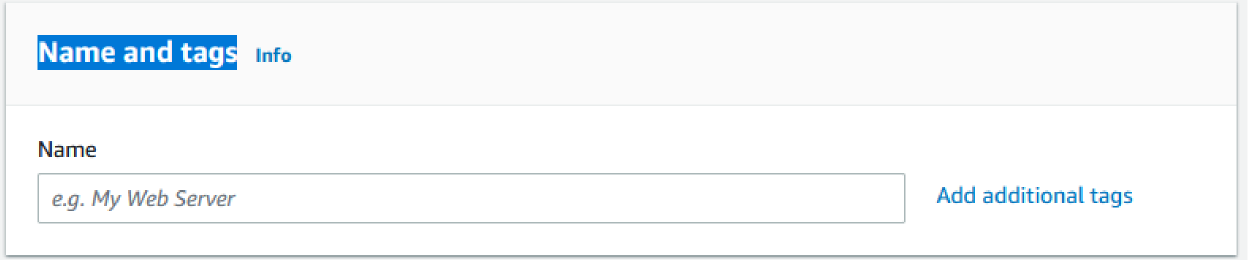
1. In the **AWS Management Console** on the Services menu, click **EC2**.
2. At the top right of the screen, if you see **New EC2 Experience** toggle to use the new UI. It is enabled by default.
3. Click Launch instance > **Launch instance**.

Provide the follow information under

Launch an instance

Step 1: Name and tags

- Provide a tag value for your instance example → **Linux-server**



Name and tags Info

Name

e.g. My Web Server

Add additional tags

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.

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

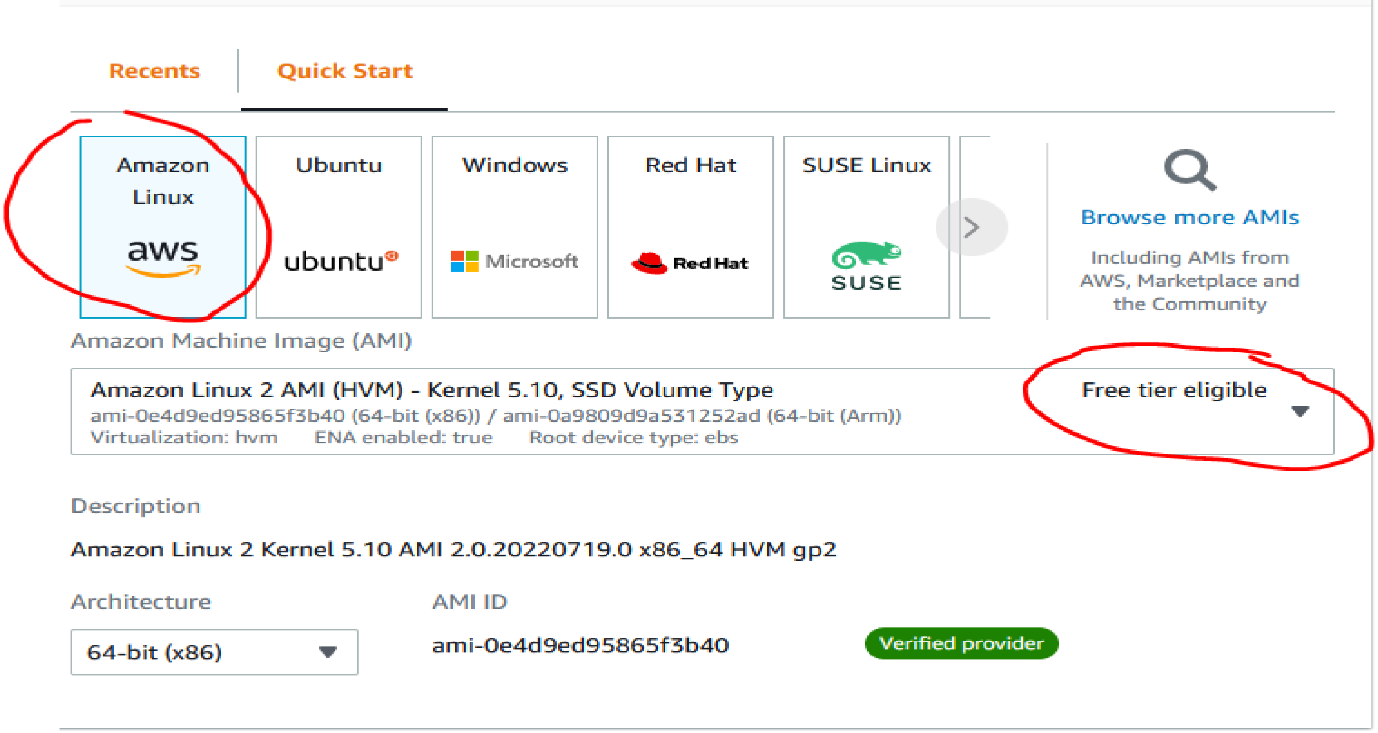
An **Amazon Machine Image (AMI)** provides the information required to launch an instance, which is a virtual server in the cloud.

An AMI includes:

1. A template for the root volume for the instance (for example, an operating system or an application server with applications)
2. Launch permissions that control which AWS accounts can use the AMI to launch instances
3. 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.

- Click **Quick Start**
- Select **Amazon Linux AWS**
- Under **Amazon Machine Image (AMI**) select **Amazon Linux 2 AMI** (HVM) (make sure it says **Free tier eligible**)



Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type
ami-0e4d9ed95865f3b40 (64-bit (x86)) / ami-0a9809d9a531252ad (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description

Amazon Linux 2 Kernel 5.10 AMI 2.0.20220719.0 x86_64 HVM gp2

Architecture: 64-bit (x86)

AMI ID: ami-0e4d9ed95865f3b40

Verified provider

Step 3: Instance Type

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.

- Click on the drop down and select **t2.micro**.
  - A **t2.micro** instance type has 1 virtual CPUs and 1 GiB of memory.

Step 4: Key pair (login)

- Select an existing key pair or create a new key pair.
  - Please refer to previous lab and use the Keypair that was created.

Do not create multiple keypair per region.

Amazon EC2 uses public–key cryptography to encrypt and decrypt login information. To log in to your instance, you must create a key pair, specify the name of the key pair when you launch the instance, and provide the private key when you connect to the instance.

- Click the **key pair name** drop-down and select an existing keypair

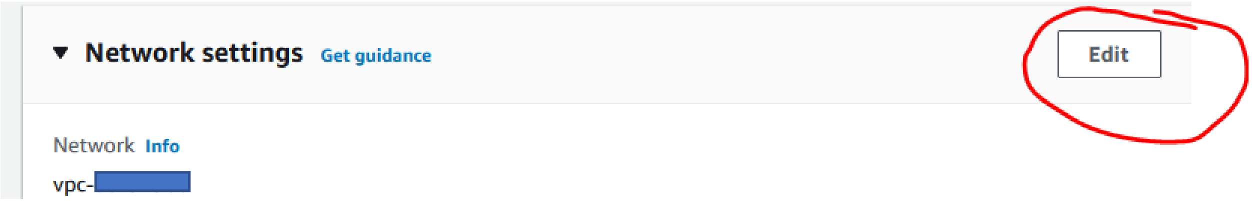
If do not already have a Keypair – Click on Create new key pair to create a new keypair.

Step 5: Network settings

This includes networking settings.

The **Network** indicates which Virtual Private Cloud (VPC) you wish to launch the instance into. You can have multiple networks, such as different ones for development, testing and production.

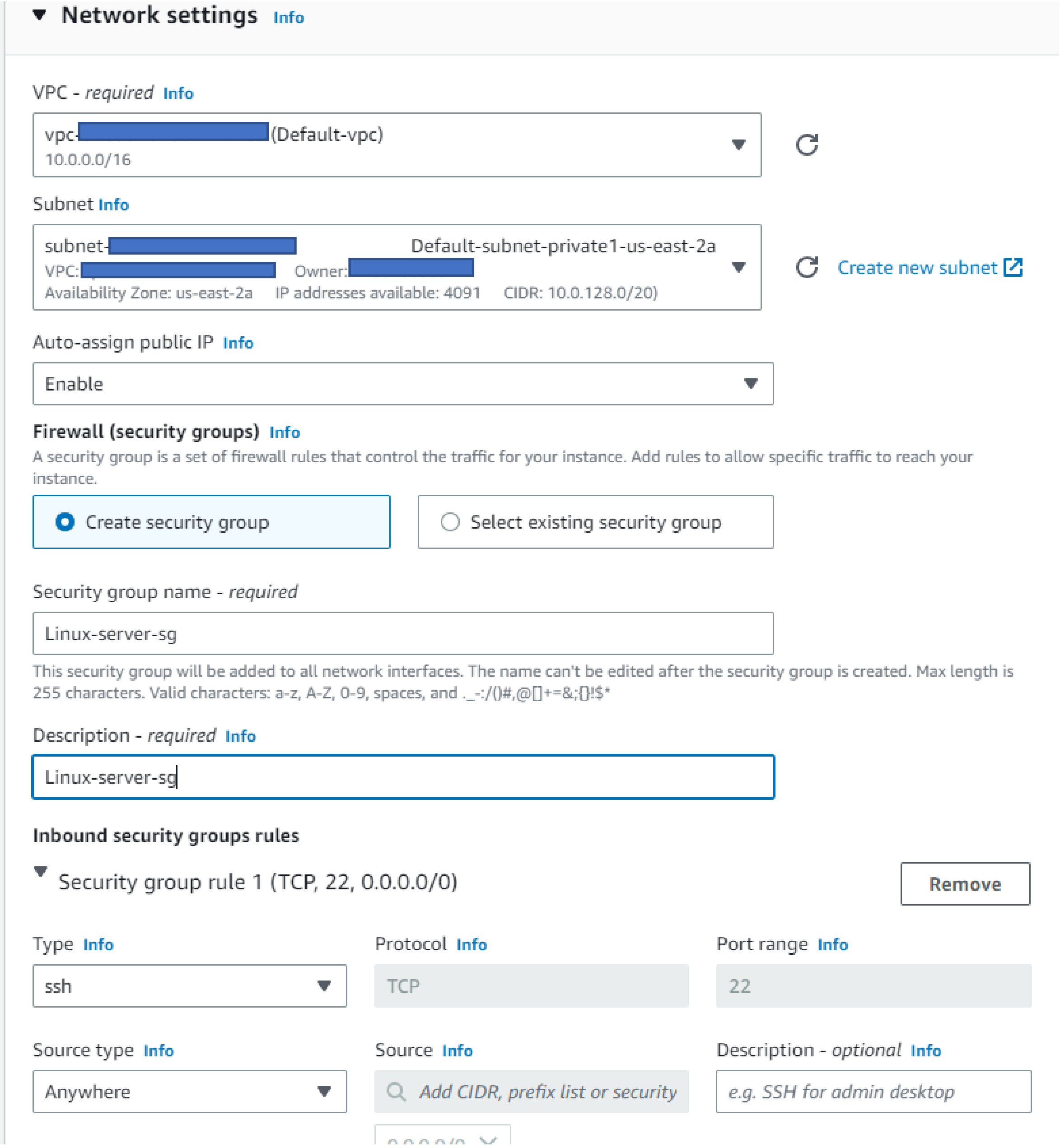
- To the right of **Network settings**, click on **edit**



Network Info

vpc-[redacted]

- Under **VPC**, select **Default VPC**.
- Under **Subnet**, Leave as is
- Under **Auto-assign public IP**, select Enable



▼ Network settings Info

VPC - *required* Info

vpc-[redacted](Default-vpc)
10.0.0.0/16

Subnet Info

subnet-[redacted] Default-subnet-private1-us-east-2a
VPC:[redacted] Owner:[redacted]
Availability Zone: us-east-2a IP addresses available: 4091 CIDR: 10.0.128.0/20)

Create new subnet

Auto-assign public IP Info

Enable

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

● Create security group ○ Select existing security group

Security group name - *required*

Linux-server-sg

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=&;{}!$*

Description - *required* Info

Linux-server-sg

Inbound security groups rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)

Remove

Type Info	Protocol Info	Port range Info
ssh	TCP	22

Source type Info	Source Info	Description - *optional* Info
Anywhere	*Add CIDR, prefix list or security* 0.0.0.0/0	*e.g. SSH for admin desktop*

Firewall (security Groups)

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.

- Select **Create Security Group**
- Security group name
- Description

Step 6: Configure storage.

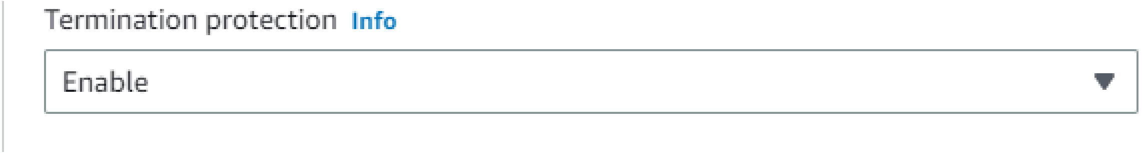
Amazon EC2 stores data on a network-attached virtual disk called *Elastic Block Store*.

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).

- Leave this section as default

Step 7: Advance details

- Scroll down, then expand **Advanced Details**.
- Under T**ermination protection,** select **Enable**



Termination protection Info

Enable

When an Amazon EC2 instance is no longer required, it can be *terminated*, which means that the instance is stopped and its resources are released. A terminated instance cannot be started again. 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.

At the bottom of the page in the **User data** section, copy and paste the attached user data content.

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

Your instance is running Amazon Linux, so you will provide a *shell script* that will run when the instance starts.

- Click on the link below, copy the data and paste into the **User data** field:

User data Script

The script will:

  - Install an Apache web server (httpd)
  - Configure the web server to automatically start on boot
  - Activate the Web server
  - Create a simple web page
-
- Click **Launch Instances**

Your instance will now be launched.

- Click **View Instances**

The instance will appear in a *pending* state, which means it is being launched. It will then change to *running*, which indicates that the instance has started booting. There will be a short time before you can access the instance.

The instance receives a *public DNS name* that you can use to contact the instance from the Internet.

Select Your **Web Server** and the **Details** tab displays detailed information about your instance.

To view more information in the Details tab, drag the window divider upwards.

Review the information displayed in the **Details** tab. It includes information about the instance type, security settings and network settings.

- Wait for your instance to display the following:
    1. **Instance State:** running
    2. **Status Checks:** 2/2 checks passed

Congratulations! You have successfully launched your first Amazon EC2 instance.

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.

- Click 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.

- Click the **Monitoring** tab.

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

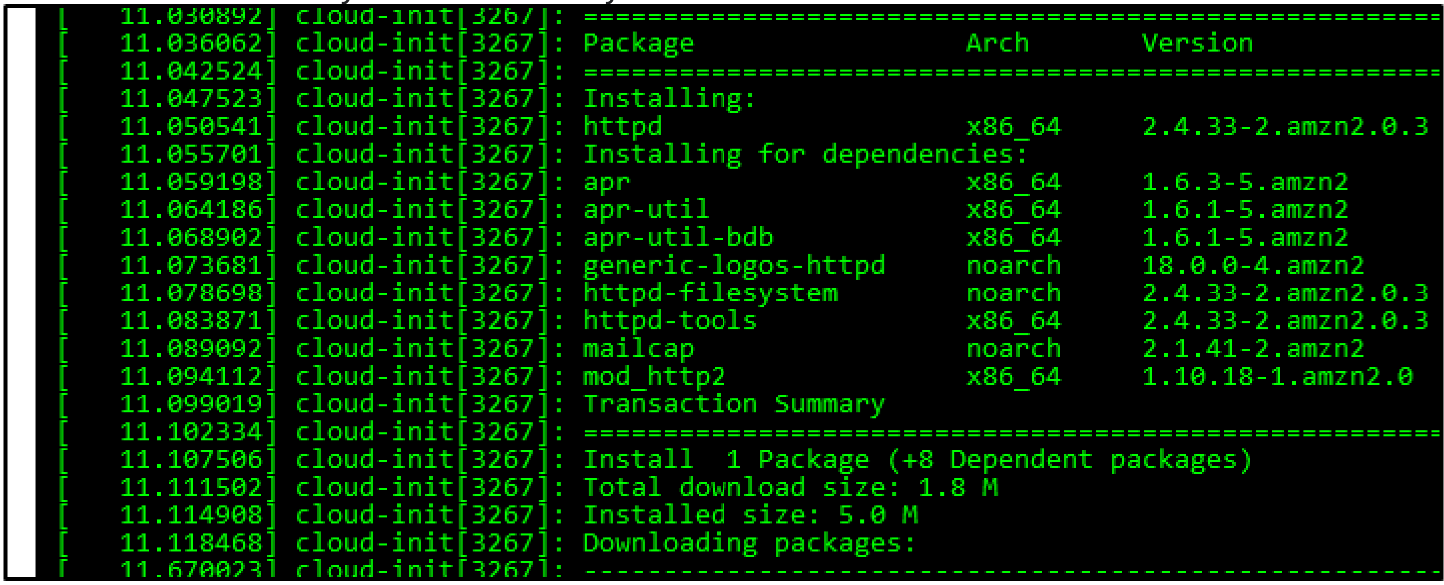
You can click on a graph to see an expanded view.

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

- In the **Actions** menu, select **Monitor and troubleshooting 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 several minutes and then try again.*

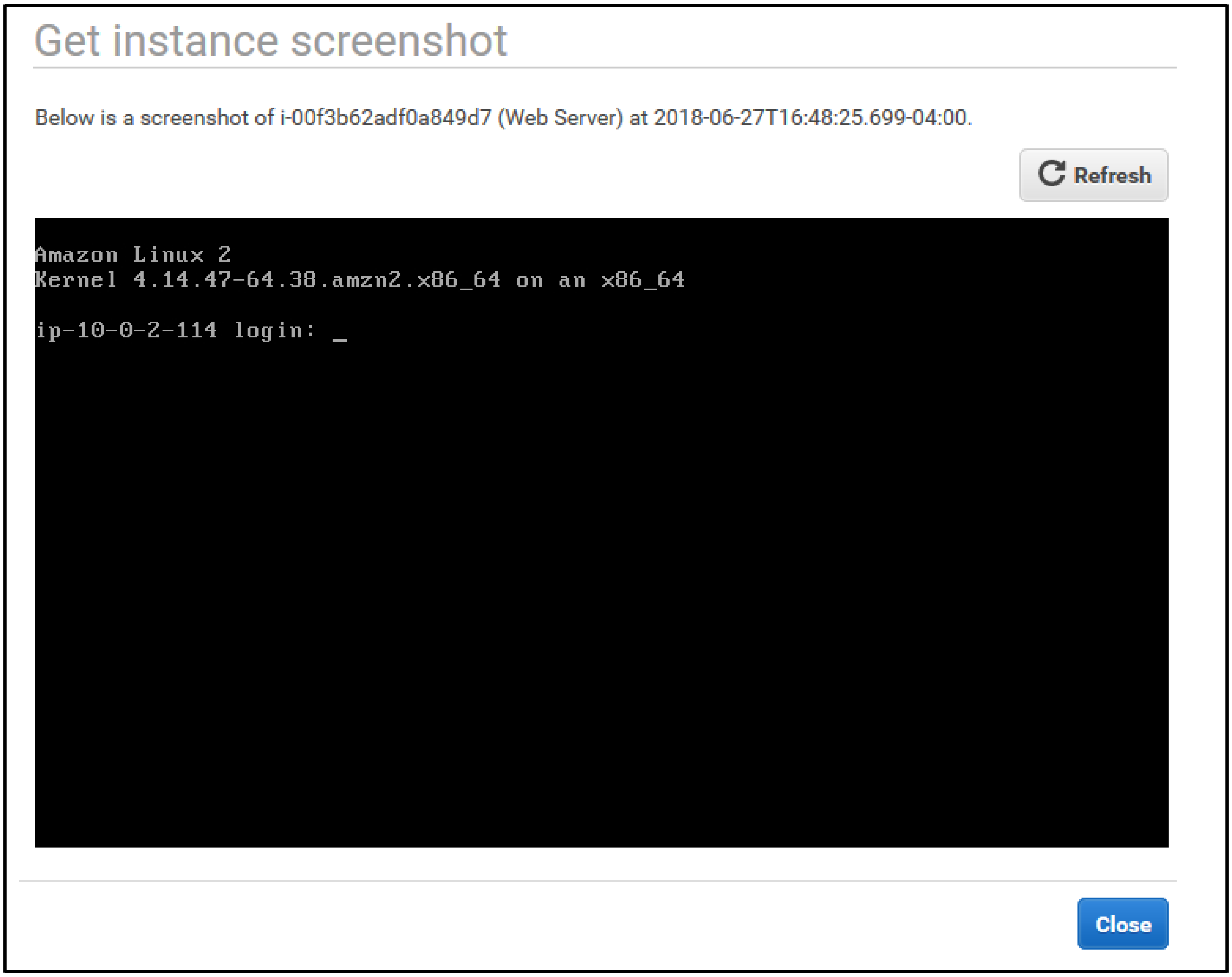
- Scroll through the output and note that the HTTP package was installed from the **user data** that you added when you created the instance.



```
[    11.030892] cloud-init[3267]: ====================================================
[    11.036062] cloud-init[3267]: Package                  Arch       Version
[    11.042524] cloud-init[3267]: ====================================================
[    11.047523] cloud-init[3267]: Installing:
[    11.050541] cloud-init[3267]: httpd                    x86_64     2.4.33-2.amzn2.0.3
[    11.055701] cloud-init[3267]: Installing for dependencies:
[    11.059198] cloud-init[3267]: apr                      x86_64     1.6.3-5.amzn2
[    11.064186] cloud-init[3267]: apr-util                 x86_64     1.6.1-5.amzn2
[    11.068902] cloud-init[3267]: apr-util-bdb             x86_64     1.6.1-5.amzn2
[    11.073681] cloud-init[3267]: generic-logos-httpd      noarch     18.0.0-4.amzn2
[    11.078698] cloud-init[3267]: httpd-filesystem         noarch     2.4.33-2.amzn2.0.3
[    11.083871] cloud-init[3267]: httpd-tools              x86_64     2.4.33-2.amzn2.0.3
[    11.089092] cloud-init[3267]: mailcap                  noarch     2.1.41-2.amzn2
[    11.094112] cloud-init[3267]: mod_http2                x86_64     1.10.18-1.amzn2.0
[    11.099019] cloud-init[3267]: Transaction Summary
[    11.102334] cloud-init[3267]: ====================================================
[    11.107506] cloud-init[3267]: Install  1 Package (+8 Dependent packages)
[    11.111502] cloud-init[3267]: Total download size: 1.8 M
[    11.114908] cloud-init[3267]: Installed size: 5.0 M
[    11.118468] cloud-init[3267]: Downloading packages:
[    11.670023] cloud-init[3267]: ----------------------------------------------------
```

- Click **Cancel**
- In the **Actions** menu, select **Monitor and troubleshoot Get instance screenshot**.

This shows you what your Amazon EC2 instance console would look like if a screen were attached to it.



Get instance screenshot

Below is a screenshot of i-00f3b62adf0a849d7 (Web Server) at 2018-06-27T16:48:25.699-04:00.

```
Amazon Linux 2
Kernel 4.14.47-64.38.amzn2.x86_64 on an x86_64

ip-10-0-2-114 login: _
```

If you are unable to reach your instance via SSH or RDP, you can capture a screenshot of your instance and view it as an image. This provides visibility as to the status of the instance, and allows for quicker troubleshooting.

- Click **Cancel**

Congratulations! You have explored several ways to monitor your instance.

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.

- Click the **Details** tab.
- Copy the **Public IPv4 address** of your instance to your clipboard.
- Open a new tab in your web browser, paste the IP address you just copied, then press **Enter**.

Question: Are you able to access your web server? Why not?

You are **not** currently able to access your web server because the *security group* is not permitting inbound traffic on port 80, which is used for HTTP web requests. This is a demonstration of using a security group as a firewall to restrict the network traffic that is allowed in and out of an instance.

To correct this, you will now update the security group to permit web traffic on port 80.

- Keep the browser tab open, but return to the **EC2 Management Console** tab.

- In the left navigation pane, click **Instances**.

- Select your instance.

- Click on **Security** Tab.

- Click on your security group.

- Click **Edit inbound rules**.

- Click **Add rule** then configure:

- **Type:** *HTTP*
- **Source:** *Anywhere*
- Click **Save rules**

The new Inbound HTTP rule will create an entry for both IPV4 IP address (0.0.0.0/0) as well as IPV6 IP address (::/0).

Note: using "Anywhere", or more specifically, using 0.0.0.0/0 or ::/0 is not a recommended best practice for production workloads.

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

You should see the message *Hello From Your Web Server!* Or similar

Congratulations! You have successfully modified your security group to permit HTTP traffic into your Amazon EC2 Instance.

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 *t3.small* 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 charge for a stopped EC2 instance, but the storage charge for attached Amazon EBS volumes remains.

- On the **EC2 Management Console**, in the left navigation pane, click **Instances**.

Web Server should already be selected.

- Click **Instance state** > **Stop instance**.

- Click **Stop**

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

- Wait for the **Instance State** to display: stopped

Change The Instance Type

- Select your **Web Server**

- In the **Actions** menu, select **Instance settings Change instance type**, then configure:

- **Instance type:** *t3.small*
- Click **Apply**

When the instance is started again it will be a *t3.small*, which has twice as much memory as a *t3.micro* instance.

Resize the EBS Volume

- In the left navigation menu, click **Volumes**.

- 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.

- Change the size to: 10

- Click **Modify**

- Click **Yes** to confirm and increase the size of the volume.

- Click **Close**

Start the Resized Instance

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

- In left navigation pane, click **Instances**.

- Click on **Instance state** > **Start instance**.

Note: An EBS volume being modified goes through a sequence of states: Modifying, Optimizing, and finally Complete.

Congratulations! You have successfully resized your Amazon EC2 Instance. In this task you changed your instance type from *t3.micro* to a *t3.small*. You also modified your root disk volume from 8 GiB to 10 GiB.

Task 5: Explore EC2 Limits

Amazon EC2 provides different resources that you can use. These resources include images, instances, volumes, and snapshots. When you create an AWS account, there are default limits on these resources on a per-region basis.

- In the left navigation pane, click **Limits**.

- Click the **All limits** and select **Running instances**.

Notice that there is a limit for the instance type based on the number of vCPUs required. For example, **For running On-Demand All Standard**, you have a limit of 1280 vCPUs. A t3.micro instance requires 2 vCPUs. Therefore, in this region, you could launch 640 t3.micro instances in this region.

You can request an increase for many of these limits.

Task 6: Test Termination Protection

You can delete your instance when you no longer need it. This is referred to as *terminating* your instance. You cannot connect to or restart an instance after it has been terminated.

In this task, you will learn how to use *termination protection*.

- In left navigation pane, click **Instances**.

- In the Instance state menu, select **Terminate instance**.

- In the **Terminate instance** dialogue box, click Terminate

Note that there is a message that says: *Failed to terminate instances i-0aws5436tfr32.*

This is a safeguard to prevent the accidental termination of an instance. If you really want to terminate the instance, you will need to disable the termination protection.

- Close the displayed error message.

- In the Actions menu, select **Instance settings Change termination protection**.

- Deselect **Enable**

- Click Save

You can now terminate the instance.

- In the Instance state menu, select **Terminate instance**.

- In the **Terminate instance** dialogue box, click Terminate

Congratulations! You have successfully tested termination protection and terminated your instance.