Simulation: Getting Started with Amazon EC2

Simulation overview and objectives

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

Amazon EC2 is a web service that provides resizable compute capacity in the cloud. It's designed to make web-scale cloud computing intuitive and straight forward to use. Amazon EC2 gives you fast access to new server instances, and you can quickly scale capacity both up and down as your computing requirements change.

This simulation is limited in its ability to accessed by a screen reader. If you are using a screen reader, please use the Simulation Instructions located in the player window to understand how to perform the actions.

OBJECTIVES

After completing this simulation, you will know how to do the following:

- Launch an EC2 instance with termination protection turned on.
- Monitor your EC2 instance.
- Modify the security group that your web server is using to allow HTTP access.
- Connect to your EC2 instance using AWS Systems Manager Fleet Manager.
- Manage the state of an EC2 instance.
- Change your EC2 instance type.
- Test termination protection.
- Explore Amazon EC2 limits.

DURATION

This simulation requires approximately **60 minutes** to complete. You will have a total time of 180 minutes to complete this simulation.

Task 1: Launching your EC2 instance

In this task, you launch an EC2 instance with termination protection. Termination protection prevents you from accidentally terminating an EC2 instance. You also deploy your instance with a user data script to deploy a simple web server.

1. In the AWS Management Console in the **Search**, enter **EC2** and choose **Enter**.

2. From the search results, choose **EC2**.

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

STEP 1: NAME YOUR EC2 INSTANCE

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

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

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

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

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

STEP 2: CHOOSE AN AMI

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

- Launch permissions that control which AWS accounts can use the AMI to launch instances
- A template for the root volume for the instance (for example, a cleanly installed operating system or one preconfigured)
- A block device mapping that specifies the volumes to attach to the instance when it's 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.

7. Locate the **Application and OS Images (Amazon Machine Image)** section. It's below the **Name and tags** section. In the search box, enter Windows Server 2019 Base and choose Enter.

8. Next to **Microsoft Windows Server 2019 Base**, choose **Select**.

STEP 3: CHOOSE AN INSTANCE TYPE

Amazon EC2 provides a wide selection of instance types that are 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 so that you can scale your resources to the requirements of your target workload.

In this step, you choose a **t2.micro** instance. This instance type has one virtual CPU and 1 GiB of memory.

9. In the **Instance type** section, keep the default instance type, **t2.micro**.

Note: When creating your own instance type, always check which instance type is the right one for your purpose.

STEP 4: CONFIGURE A KEY PAIR

Amazon EC2 uses public key cryptography to encrypt and decrypt login information. To log in to this instance, 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.

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

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

STEP 5: CONFIGURE THE NETWORK SETTINGS

You use this pane to configure networking settings.

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

11. In the **Network settings** section, choose **Edit**.

12. From the **VPC - *required*** dropdown list, choose **Lab VPC**.

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

13. For **Firewall (security groups),** choose **Select existing security group**.

14. From **Common security groups**, choose **Web Server security group**.

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.

STEP 6: ADD STORAGE

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

STEP 7: CONFIGURE ADVANCED DETAILS

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

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

When you no longer require an EC2 instance, you can terminate it, which means that the instance stops, and Amazon EC2 releases the instance's resources. You cannot restart a terminated instance. If you want to prevent your users from accidentally terminating the instance, you can turn on (enable) termination protection for the instance, which prevents users from terminating instances.

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

When you launch an instance in Amazon EC2, you have the option of passing user data to the instance. These commands can be used to perform common automated configuration tasks and even run scripts after the instance starts.

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

```
<powershell>
# Installing web server
Install-WindowsFeature -name Web-Server -IncludeManagementTools
# Getting website code
wget https://us-east-1-tcprod.s3.amazonaws.com/courses/CUR-TF-100-
EDCOMP/v1.0.4.prod-ef70397c/01-Lab-ec2/scripts/code.zip -outfile
"C:\Users\Administrator\Downloads\code.zip"
# Unzipping website code
Add-Type -AssemblyName System.IO.Compression.FileSystem
function Unzip
{
   param([string]$zipfile, [string]$outpath)
   [System.IO.Compression.ZipFile]::ExtractToDirectory($zipfile, $outpath)
}
Unzip "C:\Users\Administrator\Downloads\code.zip" "C:\inetpub\"
# Setting Administrator password
$Secure_String_Pwd = ConvertTo-SecureString "P@ssW0rD!" -AsPlainText -Force
$UserAccount = Get-LocalUser -Name "Administrator"
$UserAccount | Set-LocalUser -Password $Secure_String_Pwd
```

```
</powershell>
```

The script does the following:

- Installs a Microsoft Internet Information Services (IIS) web server
- Creates a simple web site
- Sets the password for the Administrator user

STEP 8: LAUNCH AN EC2 INSTANCE

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

19. In the **Summary** section, choose **Launch instance**.

A message indicates that you have successfully initiated the launch of your instance.

20. Choose **View all instances**.

The instance appears in a **Pending** state, which means that it's being launched. It then changes to **Running**, which indicates that the instance has started booting. There will be a short time before you can access the instance. For this simulation, the waiting period has been condensed.

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

21. Next to your **Web-Server**, select the ☐ check box. This will show the **Details** tab. Review the **Details** tab which displays information about your instance.

22. Choose the **Security** tab and review the information that's available to you.

23. Choose the **Networking** tab and review the information that's available to you. Next, choose **Continue**.

Your instance should 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 EC2 instances and your AWS solutions.

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

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.

25. Choose the **Monitoring** tab.

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

You can choose a graph to see an expanded view.

Amazon EC2 sends metrics to Amazon CloudWatch for your EC2 instances. Basic (5 minute) monitoring is turned on by default and is free. You can turn on detailed (1 minute) monitoring. With detailed monitoring, you are charged per metric that you send to CloudWatch.

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

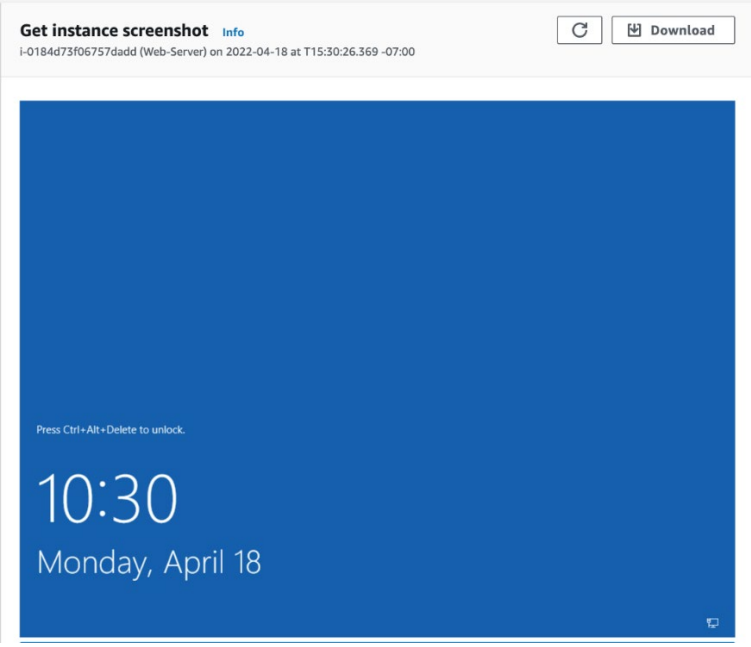
The system log displays the console output of the instance, which is a valuable tool for problem diagnosis. It's especially useful for troubleshooting service configuration issues that could cause an instance to terminate or become unreachable. If you don't see a system log, wait a few minutes and then try again.

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

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

29. With your **Web-Server** selected, choose the **Actions** dropdown list, and choose **Monitor and troubleshoot** **Get instance screenshot**.

This option shows you what your EC2 instance console would look like if a screen were attached to it. Because this is a Windows instance, the screenshot shows a locked log-in screen.



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

30. At the bottom of the page, choose **Cancel**.

Task 3: Updating your security group and accessing 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 access content from the web server.

You can't currently access your web server because the security group is not permitting inbound traffic on port 80, which is used for HTTP web requests. This next step is a demonstration of how to use a security group as a firewall to restrict the network traffic that's allowed in and out of an instance.

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

31. In the left navigation pane, choose **Security Groups**.

32. Next to **Web Server security group**, select the check box.

33. Choose the **Inbound rules** tab.

The security group currently has no rules.

34. Choose **Edit inbound rules**, and then choose **Add rule**, and configure the following options:

    - **Type:** Choose **HTTP**.
    - **Source:** Choose **Anywhere-IPv4**.

Note: Notice the *"Rules with source of 0.0.0.0/0 allow all IP addresses to access your inbound port 80. We recommend setting security group rules to allow access from known IP addresses only."* While this is true and common best practice, this simulation allows access from any IP address anywhere to simplify both the security group configuration and testing of the website running on your EC2 instance.

In this simulation, you can only add a new ingress rule. You cannot change a rule after it's created. Double check the configuration before choosing **Save rules**.

35. Choose **Save rules**.

In a live environment, you would be able to copy the public IPv4 address and paste it into a browser to ensure that the SG and user data script deployed.

Task 4: Connecting to your instance using AWS Systems Manager Fleet Manager

With the Fleet Manager capability of AWS Systems Manager, you can remotely manage and configure your managed nodes. A managed node is any machine configured for Systems Manager.

When you started this simulation, your AWS user was automatically given permissions to use Systems Manager. In addition, the AWS Identity and Access Management (IAM) policy that you selected when configuring your EC2 instance turned on Systems Manager for your Web-Server instance.

One convenient feature of Fleet Manager is the ability to connect to your EC2 instance using a browser. In this task, you connect to your Windows desktop using Fleet Manager.

36. Search for **Systems Manager** and choose **Enter**.

37. Choose **Systems Manager.**

38. In the left navigation pane, choose **Fleet Manager.**

39. Under **Managed nodes**, select ☑ your **Web-Server** EC2 instance.

40. From the Node actions dropdown list, choose **Connect**, then **Connect with Remote Desktop.**

41. Enter the **Username:** Administrator

42. Enter the **Password:** P@ssW0rD!

43. Choose **Connect**.

After several seconds, the panel displays the Windows desktop. You can navigate this desktop just like you would on a local computer. As you learned earlier, with Amazon EC2, you can quickly access compute resources. Instead of buying physical hardware and configuring an operating system, all you have to do is launch an EC2 instance, and all of that work is done for you automatically in minutes.

44. To disconnect from your **Web-Server** instance, choose **Action** and then choose **End session**.

45. In the pop-up window, choose **End session** again.

Task 5: Resizing your instance

As your needs change, you might find that your instance is overutilized (too small) or underutilized (too large). If so, you can change the instance type.

STOP YOUR INSTANCE

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

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

46. In the AWS Management Console, search for **EC2** and choose **Enter**. Then, choose **EC2**.

47. On the **EC2 Management Console**, in the left navigation pane, choose **Instances**.

48. Select the check box next to your **Web-Server** instance. At the top of the page, choose the **Instance state** dropdown list, and choose **Stop instance**.

49. In the **Stop instance?** pop-up window, choose **Stop**.

    Your instance performs a normal shutdown and then stops running.

50. Wait for the **Instance state** to display **Stopped**.

CHANGE THE INSTANCE TYPE

51. Select the check box next to your **Web-Server**. From the **Actions** dropdown list, select **Instance settings** **Change instance type**, and then configure the following option:

- **Instance type:** Select **t2.nano**.

52. Choose **Apply**.

Note: You are restricted from using other instance types in this simulation.

START THE RESIZED INSTANCE

When the instance is started again, it is a t2.nano instance. You now start the instance again, which has less memory but more disk space.

53. Next to your **Web-Server**, select the ☐ check box.

54. From the **Instance state** ⌄ dropdown list, choose **Start instance**.

After the instance is restarted, the **Instance state** displays **Running**. Choose **Continue**.

Task 6: Testing termination protection

You can delete your instance when you no longer need it. This is referred to as terminating your instance. You can't connect to or restart an instance after it's terminated.

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

55. Select the check box next to your **Web-Server** instance. From the **Instance state** ⌄ dropdown list, choose **Terminate instance**.

    Notice that Termination protection is enabled. This is a safeguard to prevent the accidental termination of an instance.

56. Choose **Terminate** to see what will happen if you try to terminate the instance.

    If you really want to terminate the instance, you need to turn off termination protection.

57. From the **Actions** dropdown list, choose **Instance settings**, and then choose **Change termination protection**.

58. The check box for ☐ **Enable** will be selected. Clear the checkbox to disable.

59. Choose **Save**.

60. Now, try to terminate the instance again. From the **Instance state** ⌄ dropdown list, choose **Terminate instance**.

61. The instance state will now successfully be terminated. Choose **Terminate**.

Summary

In this simulation, you created an EC2 instance and learned to manage instance properties such as the instance type. You modified security group settings to make the website reachable, and you learned how to use termination protection to prevent instance deletion. You learned how to stop, start, and terminate an EC2 instance. Finally, you learned how to find the EC2 limits for your AWS account. Great job!

For more information about AWS Training and Certification, see *https://aws.amazon.com/training/*.

Your feedback is welcome and appreciated.
If you would like to share any feedback, suggestions, or corrections, please provide the details in our *AWS Training and Certification Contact Form*.