



TrainerTests
.com

This lab demonstrates the steps from *Lab Guide – Create an EC2 Instance*.

My full AWS Architect Associate course can be found here:

<https://www.udemy.com/course/ultimateaws/?referralCode=7ED214B795C444141361>

Lab Guide: Creating an EC2 Instance in AWS

Objective

This lab will guide you through the step-by-step process of **creating an EC2 instance** in AWS. You will learn key concepts such as **Amazon Machine Images (AMI)**, **instance types**, **key pairs**, **security groups**, and **termination protection** while launching your own virtual server in AWS.

Prerequisites

- An **AWS account** with access to the **AWS Management Console**.
 - Basic knowledge of **EC2** and **networking concepts** is helpful but not required.
 - Familiarity with AWS **free tier** usage if applicable.
-

Step 1: Navigate to the EC2 Dashboard

1. **Log into the AWS Management Console**.
 2. **Select your AWS Region** in the top-right corner (e.g., **Ohio (us-east-2)**).
 3. **Search for "EC2"** in the AWS console's search bar.
 4. Click on **EC2** to enter the **EC2 Dashboard**.
-

Step 2: Launch an EC2 Instance

1. On the **EC2 Dashboard**, scroll down and click on **Launch Instance**.
 2. This process will create a new virtual server.
-

Step 3: Configure the Instance Settings

3.1 Naming Your Instance

- Enter a **name** for your instance, e.g., `RickCrisciWeb1`.

3.2 Choose an Amazon Machine Image (AMI)

- An **Amazon Machine Image (AMI)** is a pre-configured OS image.
- Select the **default Amazon Linux AMI**.

Key Concept:

- The AMI determines the **operating system** and **pre-installed software**.
 - AWS offers multiple AMI options, including **Windows, Ubuntu, Red Hat**, and more.
-

Step 4: Choose an Instance Type

- Choose **t2.micro** (which is eligible for AWS Free Tier).
 - **Instance Type Explanation:**
 - **t2.micro** has **1 vCPU** and **1 GB RAM**, sufficient for basic tasks.
 - AWS offers **many instance types** with varying CPU, memory, and network capabilities.
-

Step 5: Configure Key Pair for Secure Access

1. Click "**Create a new key pair**".
 2. Name it **RickCrisciWeb1** (or any name of your choice).
 3. **Download the key pair file** (`.pem`) and **store it securely**.
 4. **Key Pair Explanation:**
 - AWS uses **public-key cryptography** to secure SSH access.
 - The `.pem` file is required for **SSH authentication**.
-

Step 6: Configure Network Settings

1. Expand the **Network Settings** section.
 2. Click **Edit** to manually select a **specific subnet**.
 3. Choose a **public subnet** in **Availability Zone us-east-2A**.
 4. **VPC & Subnet Explanation:**
 - Every EC2 instance **must be inside a VPC**.
 - A **subnet** determines **which Availability Zone** the instance runs in.
 - A **public subnet** allows the instance to have a **public IP address**.
-

Step 7: Configure Security Group (Firewall Rules)

1. **Create a new security group:**
 - Name it **RickCrisciWebServers**.
 - Use the same name for the **description**.
2. **Add inbound rules:**
 - **Allow HTTP (port 80) from anywhere**.
 - **Allow HTTPS (port 443) from anywhere**.
 - **Allow SSH (port 22) from anywhere** (for management access).

Security Group Explanation:

- Security groups act as **firewalls**, controlling inbound/outbound traffic.
 - HTTP and HTTPS are needed for **web traffic**.
 - SSH allows **secure remote access** to the instance.
-

Step 8: Configure Storage (EBS Volume)

- Keep the default **GP2 General Purpose SSD**.
 - **Elastic Block Store (EBS) Explanation:**
 - **EBS is persistent block storage** for EC2.
 - The default **8GB SSD** is sufficient for most use cases.
-

Step 9: Enable Termination Protection

1. Expand **Advanced Details**.
2. Enable **Termination Protection**.

Termination Protection Explanation:

- Prevents accidental deletion of the instance.
 - Adds an extra step before **permanently terminating** the EC2 instance.
-

Step 10: Launch the Instance

1. Review your settings in the summary section.
 2. Click "**Launch Instance**".
 3. AWS will begin provisioning your virtual machine.
-

Step 11: Viewing and Managing Your Instance

1. Click on the **instance ID** to open its details.
2. Observe its **initial status as "Pending"**.
3. After a few moments, the status will change to **"Running"**.

Instance Details

- Check the **public and private IP addresses**.
 - Verify the **instance type, AMI, and VPC settings**.
-

Step 12: Monitor the Instance

1. Click the **Status Checks** tab.
2. Observe the **System Status Check** and **Instance Status Check**.
3. Wait until both checks pass.

Status Checks Explanation:

- **System Status Check** ensures AWS infrastructure is working correctly.
 - **Instance Status Check** verifies the OS is running properly.
-

Summary

In this lab, you successfully:

- **Created an EC2 instance** using the AWS console.
- **Configured an AMI, instance type, key pair, security group, and storage**.
- **Enabled termination protection** to prevent accidental deletion.
- **Verified the instance status** and monitoring details.

This EC2 instance can now be used for **web hosting, database management, or other applications**.

Next Steps

- **Connect to the EC2 instance via SSH** (if using Linux).
- **Deploy a web application** on the instance.
- **Explore additional configurations** like **Elastic Load Balancers (ELB)** and **Auto Scaling**.

For more details see my full AWS Architect Associate course:

<https://www.udemy.com/course/ultimateaws/?referralCode=7ED214B795C444141361>

