

Guided Lab: How to launch an Amazon EC2 Linux instance

Description

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that allows you to easily create and manage virtual servers in the cloud. With Amazon EC2, you can set up and configure your own operating system and applications as per your requirements.

An Amazon EC2 instance is a virtual server that can be launched on AWS Cloud. When you launch an instance, it is secured with a key pair, which is used to prove your identity, and a security group that works as a virtual firewall to control incoming and outgoing traffic. When connecting to your instance, you must provide the private key of the key pair that you specified while launching the instance.

In this lab, you will be using Amazon EC2 to launch a virtual server with a Linux operating system. This hands-on experience with cloud computing will help you understand how to use Amazon EC2 as a start for your own projects.

Objectives

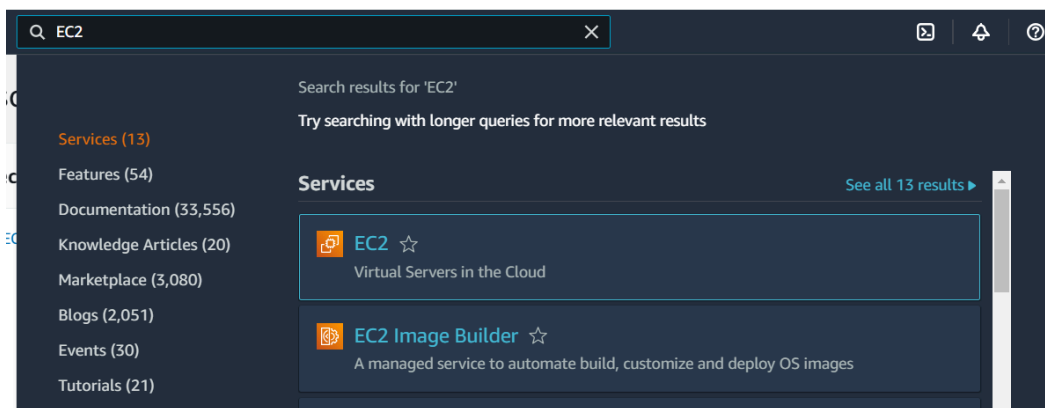
In this lab, you will learn how to:

- Create an EC2 instance (t2.micro)
- Configure a security group for SSH access
- Connect to the instance through SSH
- Learn about the Stop, Reboot, and Terminate operations

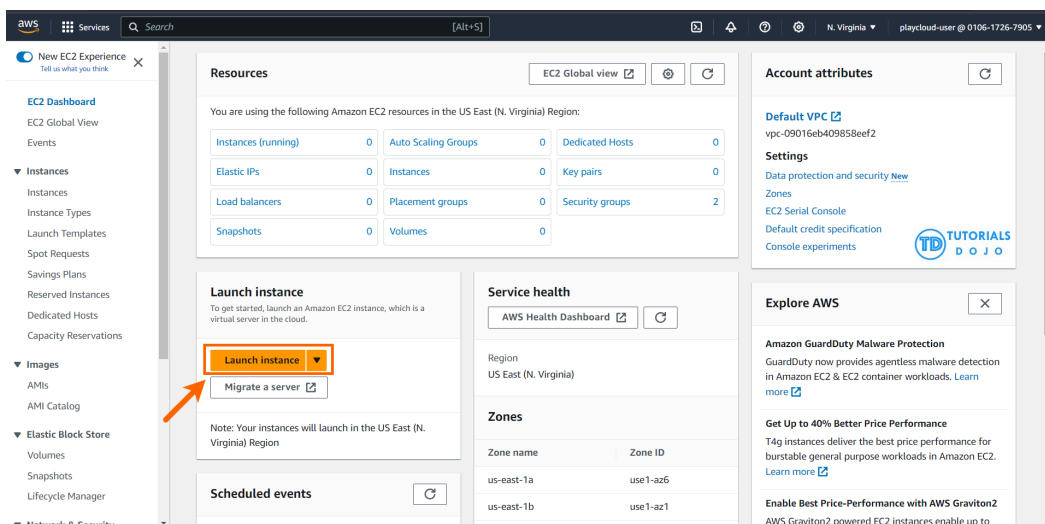
Lab Steps

Creating an Amazon EC2 instance (Linux)

1. Navigate to the search bar, type "EC2", and click to open the EC2 Dashboard.



2. Click on the 'Launch Instance' button.



3. In the "Name and tags" section, you can add a name and create tags as key/value pairs. It's recommended to tag AWS resources in production environments to stay organized, but it's not mandatory. You can skip this section if you don't want to create any tags for this lab.

4. You will need to select an Amazon Machine Image (AMI), which is basically a template of an Operating System platform that you can use as a foundation to create your instance.

For this lab, choose Ubuntu.

▼ Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu®

Windows

Microsoft

Red Hat

Red Hat

SUSE Li

SUSE

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type
ami-0fc5d935ebf8bc3bc (64-bit (x86)) / ami-016485166ec7fa705 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2023-09-19

Architecture

AMI ID

Verified provider

64-bit (x86)

ami-0fc5d935ebf8bc3bc

TD TUTORIALS DOJO

5. For the EC2 instance type, choose t2.micro.

6. In the Key pair section, you can create a new key pair by clicking on the "Create new key pair" button. Once you do this, enter "MyKeyPair" as the name of the key pair, keep the default values for Key pair type and Private key file format, and then click the "Create key pair" button. This will initiate the download of the key pair as a file named "MyKeyPair.pem" on your local system. This file contains a private key which you can use to connect to the EC2 instance via SSH.


▼ Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

MyKeyPair

Create key pair

 TUTORIALS
D O J O

×

Key pair name

Key pairs allow you to connect to your instance securely.

MyKeyPair

The name can include upto 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type


☒ RSA
RSA encrypted private and public key pair

☐ ED25519
ED25519 encrypted private and public key pair

Private key file format

☒ .pem
For use with OpenSSH

☐ .ppk
For use with PuTTY

⚠ When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** [Learn more](#) 

Cancel

Create key pair

7. In the Network Settings section, ensure that Allow SSH traffic from the checkbox is checked and Anywhere is selected under Security groups (Firewall).

▼ Network settings

Info

Edit

Network

Info

vpc-09016eb409858eef2

Subnet

Info

No preference (Default subnet in any availability zone)

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

We'll create a new security group called 'launch-wizard-1' with the following rules:

☒ Allow SSH traffic from

Helps you connect to your instance

Anywhere
0.0.0.0/0

☐ Allow HTTPS traffic from the internet

To set up an endpoint, for example when creating a web server

☐ Allow HTTP traffic from the internet

To set up an endpoint, for example when creating a web server

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

×

AWS Warning: The default configuration for the security group that is about to be created will allow SSH access from any source IP address (0.0.0.0/0). This warning is to remind you that production environments should have more restrictive security controls. However, for the purposes of this lab, this configuration is acceptable.

8. In the Configure storage section, ensure the default values of 8 GiB and gp2 Root volume are selected.

▼ Configure storage

Info

Advanced

1x

8

GiB

gp2

Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

×

Add new volume

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

0 x File systems

Edit

9. Expand the section by clicking on Advanced Details, and take a moment to review the available configurations.

▼ Advanced details [Info](#)

Purchasing option [Info](#)

☐ Request Spot Instances

Domain join directory [Info](#)

Select



[Create new directory](#)

IAM instance profile [Info](#)

Select



[Create new IAM profile](#)

Hostname type [Info](#)

IP name

DNS Hostname [Info](#)

☒ Enable IP name IPv4 (A record) DNS requests

☒ Enable resource-based IPv4 (A record) DNS requests

☐ Enable resource-based IPv6 (AAAA record) DNS requests

Instance auto-recovery [Info](#)

Select



10. Before clicking on the 'Launch instance' button to create your instance, make sure to review all of your settings.

▼ Summary



Number of instances [Info](#)

1

Software Image (AMI)

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image
build on 2023-09-19
ami-0fc5d935ebf8bc3bc

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB



Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

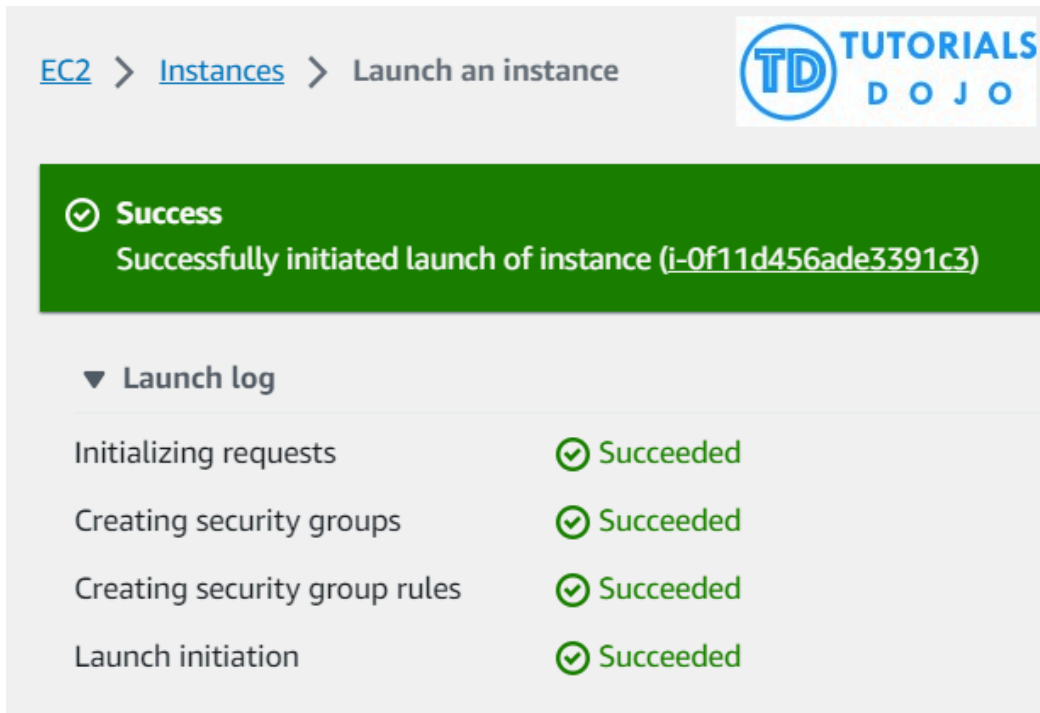


Cancel

Launch instance

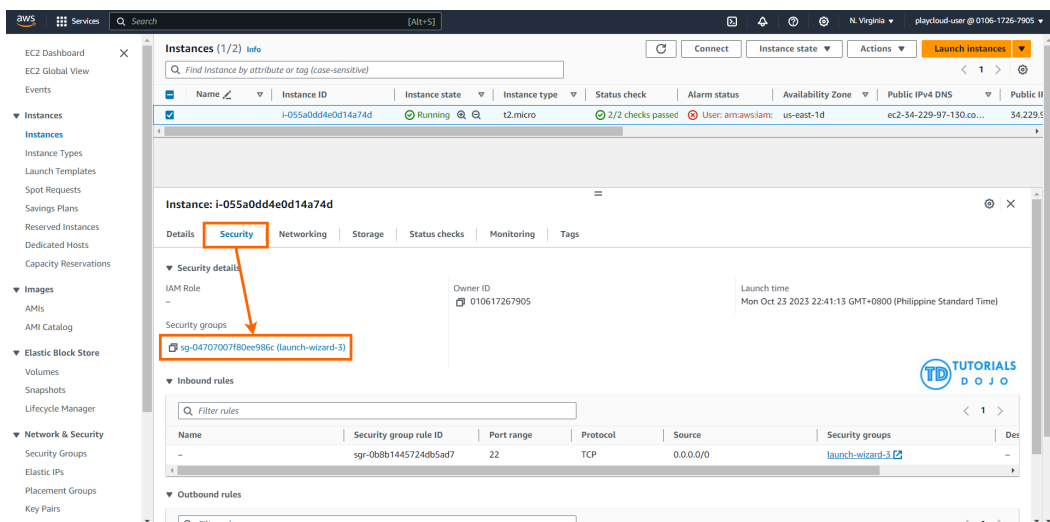
[Review commands](#)

11. After clicking on the 'Launch instance' button, a confirmation page will appear to let you know that the process has started.



Configuring a security group for SSH access

1. Go to EC2 Dashboard and click the "Instances (running)" under Resources.
2. Select the instance you want to set up Security groups for by clicking the checkbox.
3. Navigate to the Security tab. Then, click on the security group ID, which typically begins with "sg-".



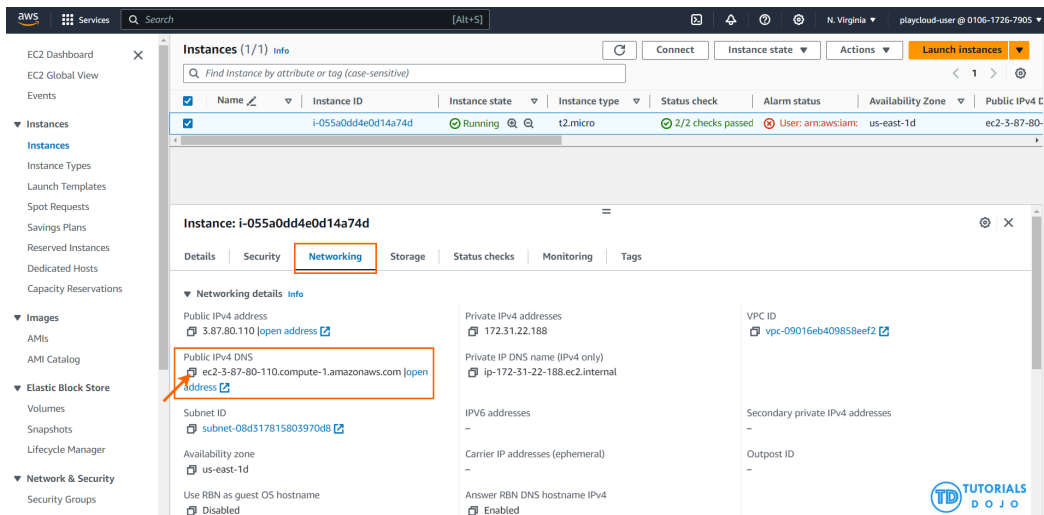
4. To connect to your Linux instance using SSH from your IP address, you can add rules to a security group.

The screenshot displays the AWS Management Console interface for a security group named 'sg-04707007f80ee986c - launch-wizard-3'. The 'Inbound rules' tab is active, showing a table with one rule: SSH access from 0.0.0.0/0 on port 22. The 'Edit inbound rules' button is highlighted with a red box. Below the table, the 'Add rule' dialog is open, showing a dropdown menu for 'Source' with 'My IP' selected. The dialog also shows the 'Protocol' as TCP and 'Port range' as 22. The 'Save rules' button is visible at the bottom right of the dialog.

To enhance the security of your instance, it is important to only authorize a specific IP address or range of addresses when setting up a rule to access it. Using 0.0.0.0/0 will allow all IPv4 addresses to access your instance via SSH. Similarly, using ::/0 will enable all IPv6 addresses to access your instance. To avoid these two options and provide a more secure solution, it is recommended to specify a particular IP address or range of addresses.

Connecting to the instance through SSH

1. After launching an instance, it may take a few minutes for it to be ready for connection.
2. Find the public DNS name or IP address of your instance to connect to it.



3. Ensure that an SSH client is installed on your local computer by typing 'ssh' in the command line. If the command is not recognized, install an SSH client.

4. To connect to your instance using SSH, open a terminal and use the ssh command. Specify the path and file name of the private key (.pem), the username for your instance, and the public DNS name or IPv6 address for your instance.

```
ssh -i "path/to/your/key.pem" ubuntu@your-instance-public-dns
```

```
C:\Users\nesto>ssh -i Downloads/MyKeyPair.pem ubuntu@ec2-3-87-80-110.compute-1.amazonaws.com
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-1012-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue Oct 24 13:38:03 UTC 2023

System load:  0.0          Processes:    98
Usage of /:   20.8% of 7.57GB   Users logged in: 0
Memory usage: 20%          IPv4 address for eth0: 172.31.22.188
Swap usage:  0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Tue Oct 24 13:34:09 2023 from 120.29.111.193
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-22-188:~$ |
```

PEM (Privacy Enhanced Mail) and PPK (PuTTY Private Key) are both formats for storing private keys, which are used in public key cryptography. Here's a comparison of the two:

PEM:

- It is a base64 container format for encoding keys and certificates.
- It is kind of the de facto standard for Linux, Mac, and Windows PowerShell users.
- The .pem file is what you download from AWS when you create your key pair. This is a one-time download, and you cannot download it again.
- To use a PEM file with SSH, you can use the `-i` option followed by the path to your PEM file. For example: `ssh -i mykey.pem myusername@mydomain.example` .

PPK:

- PPK is a format used by PuTTY, a Windows SSH client.
- It does not support the .pem format. Hence, you have to convert it to .ppk format using PuTTYgen.
- To use a PPK file with PuTTY, you need to load the PPK file in PuTTYgen and then save it as a private key. You can then use this private key to log into your server.

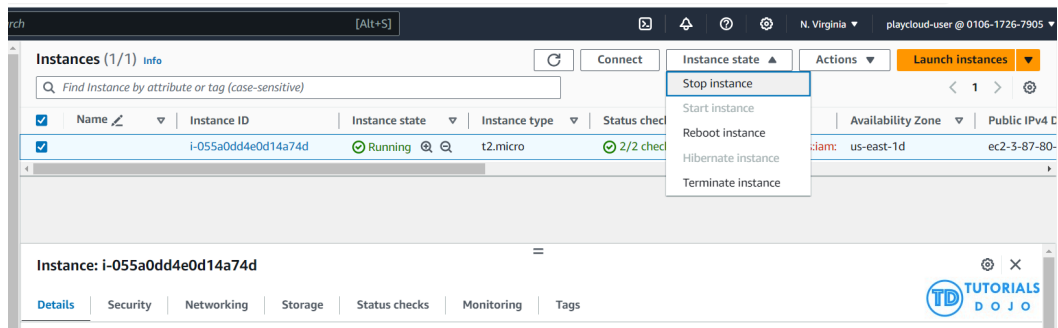
It's crucial to keep private keys secure and never share them with unauthorized individuals as they are essential for secure communication over networks.

Stop, Reboot, and Terminate operations

Stopping an EC2 instance

Steps:

- Navigate to the EC2 dashboard.
- Select the instance you want to stop.
- Click the "Instance state" dropdown menu.
- Click "Stop" from the dropdown menu.



Effects:

When you stop an instance, the following is *lost*:

- Data stored on the RAM.
- Data stored on the instance store volumes.
- The public IPv4 address that Amazon EC2 automatically assigned to the instance upon launch or start. To retain a public IPv4 address that never changes, you can associate an Elastic IP address with your instance.

When you stop an instance, the following *persists*:

- Any attached Amazon EBS volumes.
- Data is stored on the attached Amazon EBS volumes.
- Private IPv4 addresses.
- IPv6 addresses.
- Elastic IP addresses associated with the instance. Note that when the instance is stopped, you are charged for the associated Elastic IP addresses.

Rebooting an EC2 instance

Steps:

- Navigate to the EC2 dashboard.
- Select the instance you want to reboot.
- Click the "Instance state" dropdown menu.
- Click "Reboot" from the dropdown menu.

Effects:

An instance reboot is equivalent to an operating system reboot. In most cases, it takes only a few minutes to reboot your instance.

When you reboot an instance, it keeps the following:

- Public DNS name (IPv4)
- Private IPv4 address
- Public IPv4 address
- IPv6 address (if applicable)
- Any data on its instance store volumes

Rebooting an instance doesn't start a new instance billing period (with a minimum one-minute charge), unlike stopping and starting your instance.

Terminating an EC2 instance:

Steps:

- Navigate to the EC2 dashboard.
- Select the instance you want to terminate.
- Click the "Instance state" dropdown menu.
- Click "Terminate" from the dropdown menu.

Effects:

- The instance will be shut down, and the virtual machine that was provisioned for you will be permanently taken away, and you will no longer be charged for instance usage.
- Any data that was stored locally on the instance will be lost.
- Any attached EBS volumes will be detached and deleted unless they are set to persist after termination.

