

Guided Lab: Creating an Amazon Machine Image (AMI) from an EBS-backed EC2 instance

Description

An Amazon Machine Image (AMI) is a blueprint for an EC2 instance, containing all the information needed to launch an instance with your chosen configuration. This includes the operating system, custom settings, and installed applications. You can create an AMI from an existing instance, which allows you to quickly launch multiple new instances that have the same setup.

In AWS, you get access to basic AMIs. These come with different operating systems and software packages you need to get started. You can launch an EC2 instance from a basic AMI and customize it by setting up the configurations you need or installing specific software. Then, you can create a new AMI from this modified version. This way, you save all your customizations, giving you a personalized AMI for your future use.

During the creation of an AMI, Amazon EC2 takes snapshots of the instance's root volume and any other Amazon Elastic Block Store (EBS) volumes that are attached to the instance. This process ensures that both the configuration of the instance and the state of all attached EBS volumes are replicated. Keep in mind that there are storage charges for the snapshots until you deregister the AMI and delete those snapshots.

In this lab, you will learn how to create an Amazon Machine Image (AMI) from an EBS-backed EC2 instance. These hands-on labs will help you understand how to incorporate AMIs into your projects without starting from scratch.

Prerequisites

Before starting this lab, it is expected that you have prior experience in creating EC2 instances and are well-versed with their basic components. In case you encounter any gaps in your knowledge, we recommend taking the following labs to gain sufficient knowledge:

- Creating an Amazon EC2 instance (Linux)
- Setting up a Web server on an EC2 instance Setting up a Web server on an EC2 instance

Objectives

In this lab, you will:

- Have a basic understanding of AMI
- Learn how to create an AMI
- Learn how to deregister an AMI
- Learn how to create an EC2 instance from a custom AMI

Subscribe to access AWS
PlayCloud Labs

Lab Steps

Creating the EC2 instances

Create an EC2 instance using the following configurations:

- **Name:** stallions
- **Instance type:** t2.nano
- **AMI:** Linux
- **Key pair:** Create a new key pair.
 - Key pair name: **td-stallions**
 - Key pair type: **RSA**
 - Private key file format: **.pem**

- **Security group name:** WebServerSG
- **Description:** enter '*Allows SSH and HTTP access*'.
- **Inbound Security Group Rules: (Add security group rule)**
 - Inbound rule 1:
 - **Type:** SSH
 - **Source Type:** My IP
 - Inbound rule 2
 - **Type:** HTTP
 - **Source Type:** Anywhere (0.0.0.0/0)

Review your instance configurations and click the **"Launch Instance"** button.

Setting up a Web server on an EC2 instance

In this section, we'll make changes to the EC2 instance by configuring a simple web server on it.

1. Open up a terminal, then run the command below to connect to your instance via SSH.

```
ssh -i /path/to/YOUR-KEY.pem ec2-user@YOUR-EC2-PUBLIC-IP
```

2. Once connected, run the command below:

```
# Step 1: Update the system
sudo yum update -y

# Step 2: Install Nginx
sudo yum install nginx -y

# Step 3: Start Nginx Service
sudo service nginx start

# Create a custom HTML page directly in the HTML
directory
echo '<h1>Welcome to my web page!</h1>' | sudo tee
/usr/share/nginx/html/mypage.html > /dev/null

# Add the configuration directly
echo 'server {
    listen 80 default_server;
    server_name _;
    root /usr/share/nginx/html;

    location / {
        index mypage.html;
    }
}' | sudo tee /etc/nginx/conf.d/server.conf > /dev/null

# Reload Nginx for the changes to take effect
sudo nginx -t && sudo service nginx reload

#Automatically start NGINX at boot time
sudo systemctl enable nginx
```

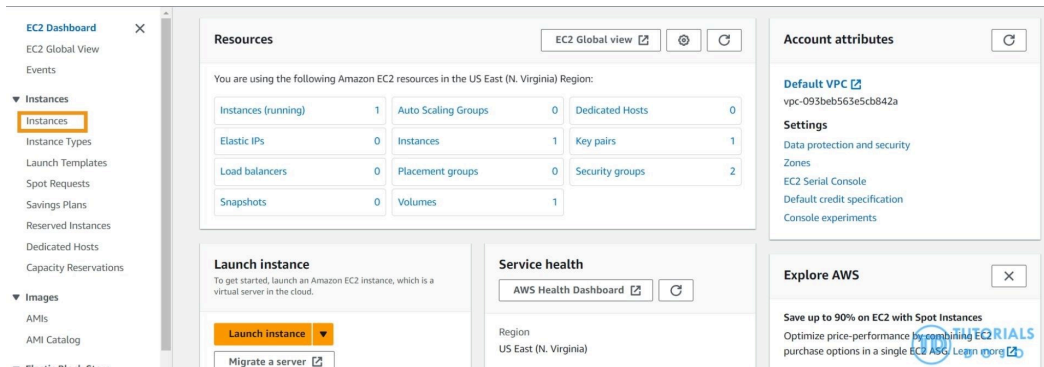
3. After restarting the nginx, check your web server.



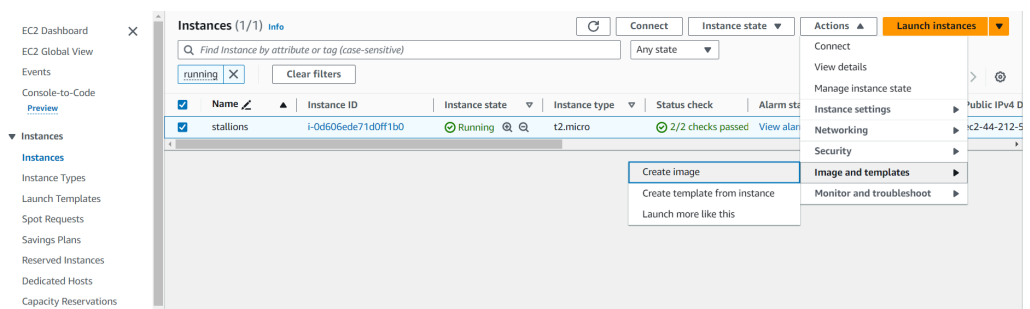
Creating an Amazon Machine Image (AMI) from an EBS-backed EC2 instance

In this section, we'll create an AMI from the EC2 instance, which will now include the configurations for the web server.

1. In the navigation pane, select the "Instances" tab.

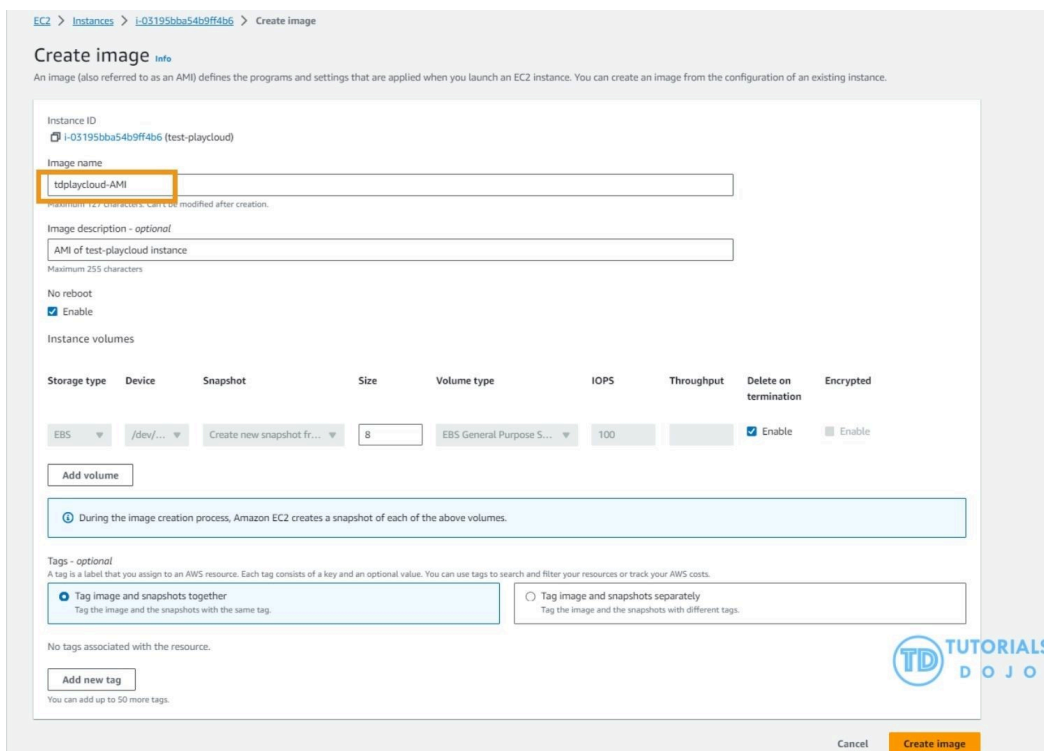


2. Choose the instance to create an AMI, click "Actions", and select "Create image" in the dropdown menu under "Image and templates".



3. Provide the following details on the "Create image" page:

- Image name: Enter 'tdplaycloud-AMI.'

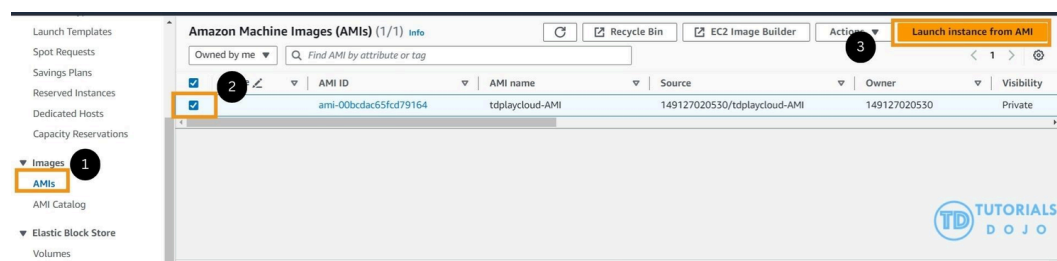


4. Enable the No Reboot option.
5. In the **"Instance volumes"** section, the default EBS volume setting indicates that an EBS volume will back your AMI.
6. Scroll down to the bottom of the page and click **"Create image."**
7. In the left-hand menu, click on **"AMIs."**
8. Select the AMI and verify if the status is pending under the **Details** tab.
9. The AMI creation process may take up to 5 minutes.
10. Confirm that the AMI has been successfully created by checking if the status is **available**.

Create an EC2 instance from the custom AMI

Now that we have our custom AMI, let's launch an instance from it to confirm that our web server configuration is preserved.

1. In the left-hand navigation panel, click on **"AMI"**, select the AMI that you want to use to create an EC2 instance and click on **"Launch instance from AMI"**.



2. In the **"Name and tags"** section, enter **"new-playcloudAMI-instance"** under Name.

EC2 > Instances > Launch an instance

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name

Add additional tags

3. In the “Instance type” section, ensure that the t2.micro instance type is selected.

▼ Instance type Info

Instance type

t2.micro

Family: t2 1 vCPU 1 GiB Memory Current generation: true

On-Demand Windows base pricing: 0.0162 USD per Hour

On-Demand SUSE base pricing: 0.0116 USD per Hour

On-Demand RHEL base pricing: 0.0716 USD per Hour

On-Demand Linux base pricing: 0.0116 USD per Hour

Free tier eligible

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

4. Select the existing Key Pair.

5. Review the summary section and click “Launch instance”.

EC2 > Instances > Launch an instance

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name

Add additional tags

▼ 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

AMI from catalog

Recents

My AMIs

Quick Start

Amazon Machine Image (AMI)

tdplaycloud-AMI

ami-00bcdac65fcd79164

Browse more AMIs

Including AMIs from

▼ Summary

Number of instances Info

Software Image (AMI)

AMI of test-playcloud instance...read more

ami-00bcdac65fcd79164

Virtual server type (instance type)

t2.micro

Firewall (security group)

launch-wizard-2

Storage (volumes)

1 volume(s) - 8 GiB

Cancel

Launch instance

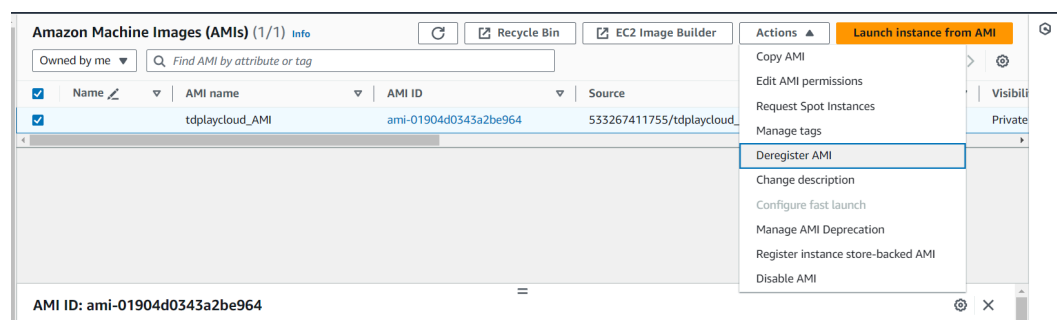
Review commands

7. A confirmation page will inform you that your instance is launching and the changes have been preserved.

Deregistering your AMI

To deregister an AMI in the Amazon EC2 console, follow these steps:

1. In the navigation pane, click on the AMIs option.
2. Select the AMI you want to deregister and make a note of its ID. This information may come in handy in the next step when you need to find and delete related snapshots.
3. Click on the Actions button and choose the Deregister AMI option.



4. When prompted to confirm that you want to deregister the AMI, click on Deregister AMI.

