**Task: Create Volume and Mount It on an EC2 Instance**

**Name:** Abhishek Hengade

**Batch:** B-56

This guide provides a complete walkthrough for provisioning and using a new Amazon Elastic Block Store (EBS) volume with an EC2 instance. EBS volumes act as durable, persistent block storage, similar to virtual hard drives, ideal for data that must outlive the instance itself.

**1. Understanding the Process**

The process involves three distinct stages:

1. **Creating the Volume:** Provisioning a new EBS volume in a specific Availability Zone.
2. **Attaching the Volume:** Connecting the new volume to a running EC2 instance in the same Availability Zone.
3. **Mounting the Volume:** From within the instance's operating system, formatting the volume with a file system and making it accessible through a directory.

**2. Step-by-Step Guide**

**Part 1: Create and Attach the EBS Volume**

**Step 1: Create the EBS Volume** Navigate to the **EC2 Dashboard** in the AWS Console. In the left menu, under **Elastic Block Store**, click **Volumes**, and then **Create volume**.

* **Volume type:** Choose a type, such as **General Purpose SSD (gp3)**.
* **Size:** Specify the size (e.g., 10 GiB).
* **Availability Zone:** This is the most critical setting. **You must select the same Availability Zone as the EC2 instance** you plan to attach it to.
* Click **Create volume**.

**Step 2: Attach the Volume to Your Instance** Once the volume's status is **"available,"** select it from the list. Click the **Actions** menu and choose **Attach volume**.

* In the dialog box, select your target **Instance** from the dropdown list.
* Note the **Device name** suggested by AWS (e.g., /dev/sdf).
* Click **Attach volume**. The volume's state will now change to **"in-use."**

**Part 2: Format and Mount the Volume (Inside the EC2 Instance)**

**Step 3: Connect to Your Instance and Identify the Volume** Connect to your Linux EC2 instance using SSH. Once connected, run the lsblk command to list the available block devices. You should see the new device you just attached (e.g., xvdf if you used /dev/sdf).

Bash

lsblk

The output will show the disk (xvdf) but it won't have a mount point yet.

**Step 4: Format the Volume** A new volume is raw and unformatted. You must create a file system on it. Use the mkfs command to format it with the ext4 file system.

Bash

sudo mkfs -t ext4 /dev/xvdf

*Replace /dev/xvdf with the device name shown in your lsblk output.*

**Step 5: Create a Mount Point and Mount the Volume** First, create a directory that will serve as the mount point for your volume. Then, use the mount command to attach the volume to that directory.

Bash

sudo mkdir /data

sudo mount /dev/xvdf /data

**Step 6: Verify the Mount** Run the df -h command to verify that the volume is successfully mounted. You should see a new entry for /data with the size you specified (e.g., 10G).

Bash

df -h

**Step 7: Make the Mount Persistent (Auto-mount on Reboot)** To ensure the volume is automatically mounted every time the instance reboots, you must add an entry to the /etc/fstab file. First, back up the file, then add the new entry.

Bash

# Backup the fstab file

sudo cp /etc/fstab /etc/fstab.orig

# Add the mount entry to the end of the file

echo '/dev/xvdf /data ext4 defaults,nofail 0 2' | sudo tee -a /etc/fstab

The nofail option prevents your instance from failing to boot if the EBS volume is ever detached or fails

