**Job-Ready Project:**

**Managing EBS Volumes**

***with***

**LVM on AWS EC2**

In this project, we will demonstrate how to configure Amazon Elastic Block Store (EBS) volumes on an EC2 instance, manage them using Logical Volume Management (LVM), and perform file system operations. The project covers the full lifecycle from volume creation to logical volume extension and file system management.

**Project Overview**

1. Create and **Attach** EBS Volumes to an EC2 instance.

2. Initialize and Configure LVM, including creating physical volumes, volume groups, and logical volumes.

3. **Mount** and **Format** the logical volume to make it **usable**.

4. Perform Operations on the mounted volume.

5. Extend the Logical Volume and resize the file system.

**Steps:**

**Step 1: Launch an EC2 Instance**

1. Go to the AWS Management Console.

2. Launch a new EC2 instance:

- AMI: Amazon Linux 2/Ubuntu 20.04 LTS

- Instance Type: t2.micro (or as required)

- Security Group: Ensure **SSH** access is allowed.

3. SSH into your EC2 instance:

ssh -i your-key.pem ec2-user@ec2-instance-ip

**Step 2: Create and Attach EBS Volumes**

1. In the AWS console, **create** three new EBS volumes:

- Volume 1: 10 GB (attached as `/dev/**xvdf**`)

- Volume 2: 15 GB (attached as `/dev/**xvdg**`)

- Volume 3: **20** GB (attached as `/dev/**xvdh**`)

2. Attach these volumes to the EC2 instance.

3. Verify the volumes are attached:

**lsblk**

**Step 3: Initialize LVM**

1. Switch to root:

**sudo su**

2. Initialize the attached EBS volumes as physical volumes for LVM:

**pvcreate /dev/xvdf /dev/xvdg /dev/xvdh**

3. Verify the physical volumes:

**pvs**

4. Create a volume group named `azizul\_vg`:

**vgcreate** azizul\_v**g** /dev/xvd**f** /dev/xvdg

5. Check the volume group:

**vgs**

6. Create a logical volume named `azizul\_lv` with 10 GB size:

**lvcreate** -L **10**G -n azizul\_lv azizul\_vg

7. Verify logical volume:

**lvdisplay**

**Step 4: Format and Mount the Logical Volume**

1. Create a **directory** to mount the logical volume:

**mkdir /mnt/**azizul\_lv\_mount

2. **Format** the logical volume with the **ext4** filesystem:

**mkfs.ext4** /dev/azizul\_vg/azizul\_lv

3. **Mount** the logical volume:

**mount** /dev/azizul\_vg/azizul\_lv /**mnt**/azizul\_lv\_mount

4. Verify that the volume is mounted:

**df -h**

**lsblk**

**Step 5: Create and Manipulate Files on the Volume**

1. Navigate to the mounted directory and **create** **files**:

cd /mnt/azizul\_lv\_mount

mkdir **devops**

cd devops

vim **hello.txt** # Add "This is devops" in the file

2. View the file contents:

cat hello.txt

**Step 6: Unmount and Remount the Volume**

1. Unmount the logical volume:

**umount** /mnt/azizul\_lv\_mount

2. **Remount** the volume to confirm the file persists:

**mount** /dev/azizul\_vg/azizul\_lv /mnt/azizul\_lv\_mount

cat /mnt/azizul\_lv\_mount/devops/**hello.txt**

**Step 7: Mount the Remaining Volume**

1. Create a directory for the third volume `/dev/xvd**h**`:

**mkdir** /mnt/azizul\_**disk**\_mount

2. Format the volume and mount it:

mkfs.ext4 /dev/xvd**h**

**mount** /dev/xvdh /mnt/azizul\_**disk**\_mount

3. Verify the new volume is mounted:

**df -h**

**Step 8: Extend the Logical Volume**

1. Extend the logical volume `azizul\_lv` by an additional **5** GB:

**lvextend** -L +5G /dev/azizul\_vg/azizul\_lv

2. Resize the filesystem to use the added space:

**resize2fs** /dev/azizul\_vg/azizul\_lv

3. Verify the extended size:

**df -h**

**Step 9: Final Checks**

1. List block devices and logical volumes to ensure everything is in order:

**lsblk**

**lvs**

2. Confirm the disk space after the extension:

**df -h**

**Project Wrap-up**

In this project, we successfully managed EBS volumes on an EC2 instance using LVM. We created physical volumes, volume groups, and logical volumes, **formatted** and **mounted** them, and extended the logical volume to meet **storage** **needs**. We also demonstrated basic file operations and volume management tasks, making the disk space **usable** efficiently.