

Lab EBS

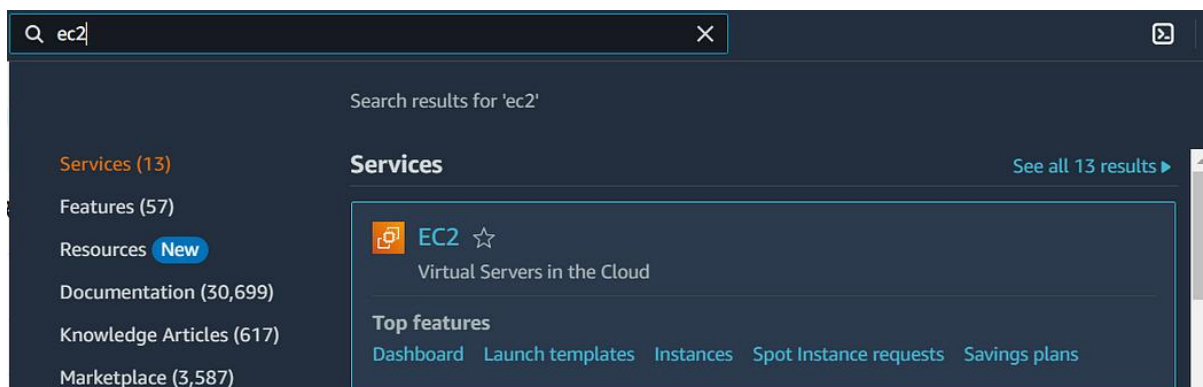
Creating and Attaching an EBS Volume

- Launch awslinux type resource
- Go to volume and check which availability zone volume has created
- EC2->Volume section

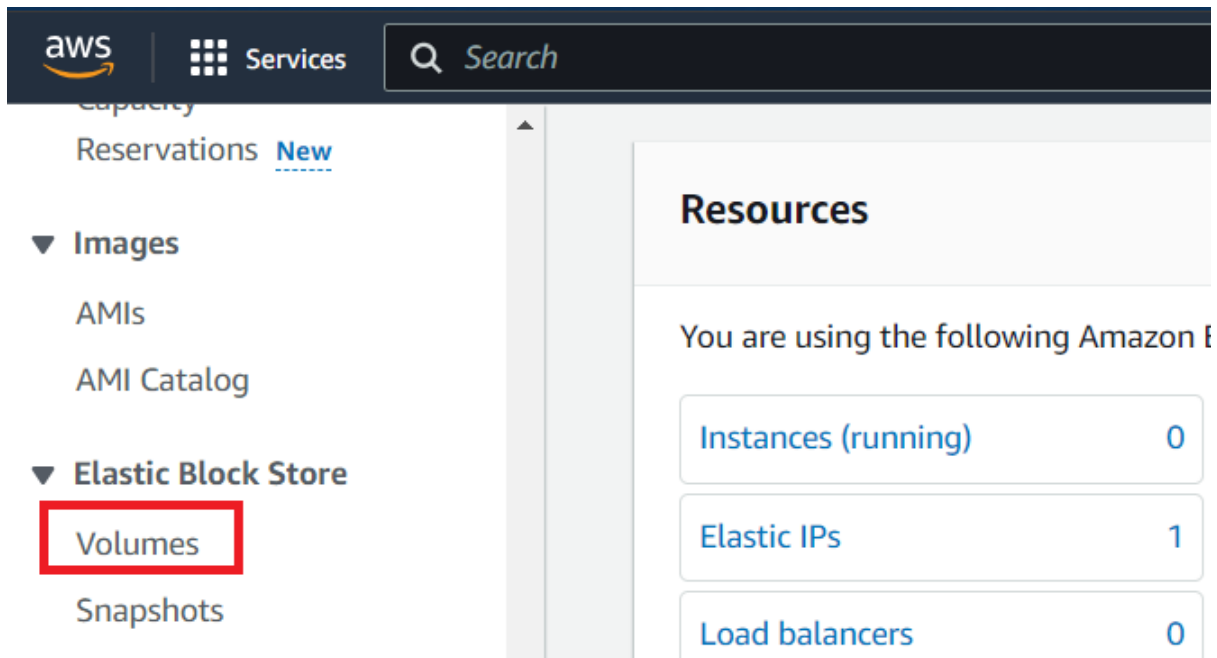
Launch Templates	vol-02b4f258939a5edfb	8 GiB	gp3	Okay
Spot Requests	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more	Volume state In-use	IOPS 3000	Throughput 125
Savings Plans	Fast snapshot restored No	Availability Zone ap-south-1b	Created Sun Jan 12 2025 01:50:26 GMT+0530 (India Standard Time)	Multi-Attach enabled No
Reserved Instances	Attached resources i-016737d019afec9c1 (awslinux): /dev/xvda (attached)	Outposts ARN -	Managed false	Operator -
Dedicated Hosts	Source Snapshot ID			
Capacity Reservations				
▼ Images				
AMIs				
AMI Catalog				
▼ Elastic Block Store				
Volumes				
Snapshots				

1. Create an EBS Volume:

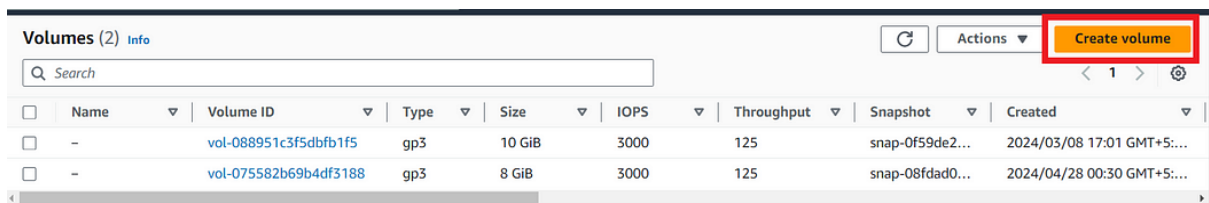
- Go to the AWS Management Console.
- Navigate to the EC2 Dashboard.



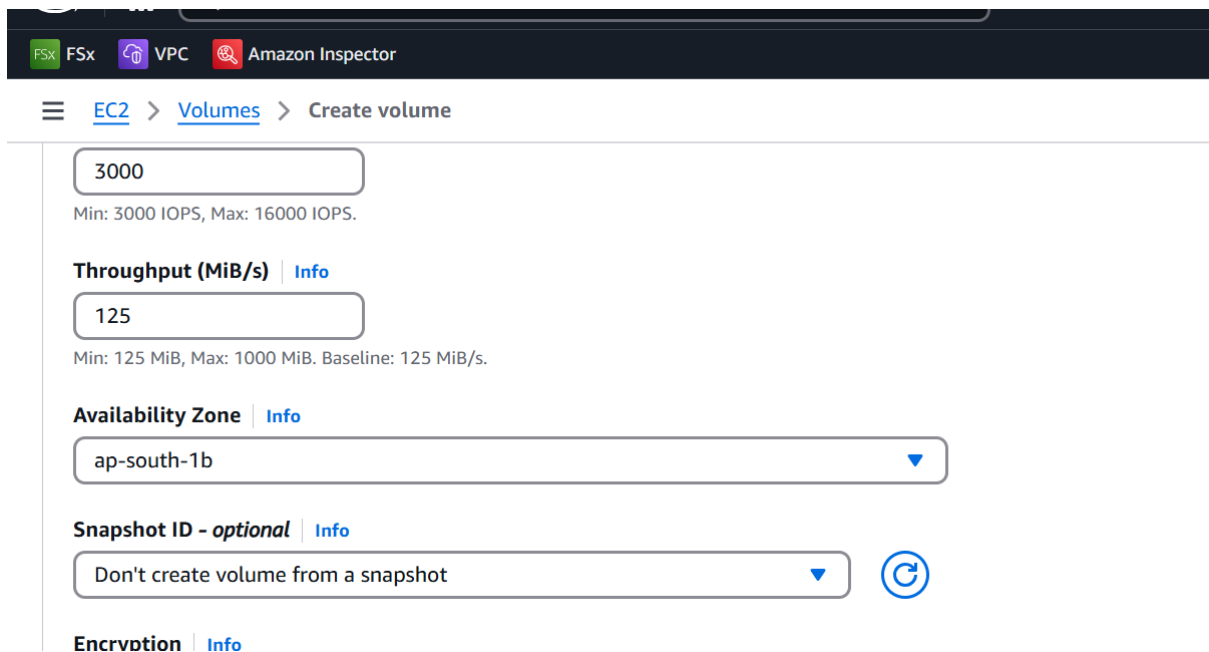
- Click on “Volumes” under “Elastic Block Store”.



- Click “Create Volume”.

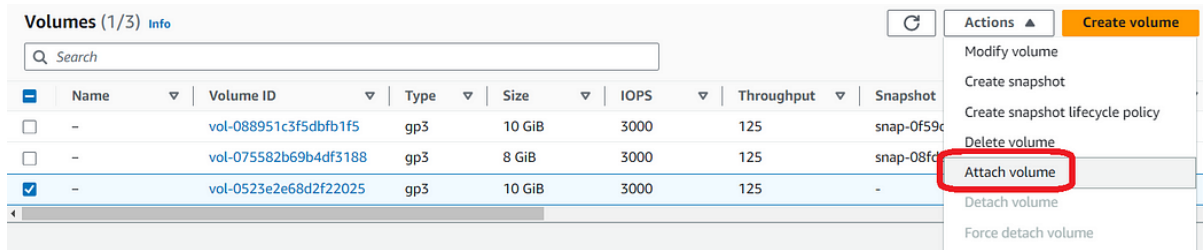


- Specify the volume type, size, and availability zone, then click “Create Volume”.



2. Attach the EBS Volume to an Instance:

- Select the newly created volume from the “Volumes” list.
- Click on “Actions” and select “Attach Volume”.



- Choose the instance to attach the volume to and specify the device name (e.g., `/dev/sdf`), then click “Attach Volume”.

vol-0243064ee96bf14b1

Availability Zone
ap-south-1b

Instance [Info](#)

i-016737d019afec9c1
(awslinux) (running)

Only instances in the same Availability Zone as the selected volume are displayed.

Device name [Info](#)

`/dev/sde`

Recommended device names for Linux: `/dev/xvda` for root volume. `/dev/sd[f-p]` for data volumes.

Newer Linux kernels may rename your devices to `/dev/xvdf` through `/dev/xvdp` internally, even when the device name entered here (and shown in the details) is `/dev/sdf` through `/dev/sdp`.

- Since the Volume is created in us-west-1a Availability zone, it will automatically detect all the instances in same availability zone. (EBS is AZ service).

3. Formatting and Using the EBS Volume

1. Connect to the EC2 Instance:
 - Use SSH to connect to your EC2 instance.
2. Format the EBS Volume:

```
[ec2-user@ip-172-31-2-152 ~]$ sudo fdisk -l
Disk /dev/xvda: 8 GiB, 8589934592 bytes, 16777216 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 174813BE-7DC8-40F8-A11F-AC3DAA194142
```

Device	Start	End	Sectors	Size	Type
/dev/xvda1	24576	16777182	16752607	8G	Linux filesystem
/dev/xvda127	22528	24575	2048	1M	BIOS boot
/dev/xvda128	2048	22527	20480	10M	EFI System

```
Partition table entries are not in disk order.
```

```
Disk /dev/xvdb: 10 GiB, 10737418240 bytes, 20971520 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
[ec2-user@ip-172-31-2-152 ~]$
```

```
sudo mkfs.ext4 /dev/xvdb
```

```
[ec2-user@ip-172-31-2-152 ~]$ sudo mkfs.ext4 /dev/xvdb
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 2621440 4k blocks and 655360 inodes
Filesystem UUID: 87499014-3421-440a-888a-7b38817af800
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

[ec2-user@ip-172-31-2-152 ~]$
```

3. Mount the Volume:

```
sudo mkdir /data
sudo mount /dev/xvdb /data
```

```
[ec2-user@ip-172-31-2-152 ~]$ sudo mkdir /data
[ec2-user@ip-172-31-2-152 ~]$ sudo mount /dev/xvdb /data
[ec2-user@ip-172-31-2-152 ~]$
[ec2-user@ip-172-31-2-152 ~]$ sudo df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M   0    4.0M   0% /dev
tmpfs           475M   0    475M   0% /dev/shm
tmpfs           190M  2.9M   188M   2% /run
/dev/xvda1      8.0G  1.6G   6.4G  20% /
tmpfs           475M   0    475M   0% /tmp
/dev/xvda128    10M   1.3M   8.7M  13% /boot/efi
tmpfs           95M    0     95M   0% /run/user/1000
/dev/xvdb       9.8G   24K   9.3G   1% /data
[ec2-user@ip-172-31-2-152 ~]$
```

4. Store Data:

- You can now store data in the /data directory.

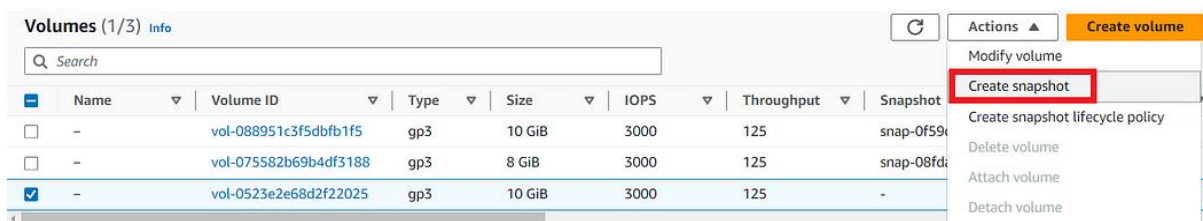
```
sudo cd /data
```

```
touch a
```

```
touch b
```

4. Create a Snapshot:

- Go to the “Volumes” section in the EC2 Dashboard.
- Select the volume, click “Actions”, and then “Create Snapshot”.
- Provide a description and click “Create Snapshot”.



Create snapshot [Info](#)

Create a point-in-time snapshot to back up the data on an Amazon EBS volume to Amazon S3.

Details

Volume ID

 [vol-0523e2e68d2f22025](#)

Description

Add a description for your snapshot

first snapshot

255 characters maximum.


Encryption [Info](#)

Not encrypted

Tags [Info](#)

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key

 Name



Value - optional

 First Snap



Remove

Add tag

5. Create a Volume from a Snapshot:

- Go to the “Snapshots” section.
- Select the snapshot, click “Actions”, and then “Create Volume”.
- Specify the new volume details, including the availability zone, and click “Create Volume”.

Snapshots (1/2) Info

Owned by me

Search

Refresh

Recycle Bin

Actions

Create snapshot

	Name	Snapshot ID	Volume size	Description	Storage tier	Size
<input type="checkbox"/>	-	snap-0abf3c9c6b7a5f406	10 GiB	first snapshot	Standard	14:5
<input checked="" type="checkbox"/>	-	snap-01e29103cb68d25f2	10 GiB	second snapshot	Standard	15:0

Create volume from snapshot

Create image from snapshot

Copy snapshot

Delete snapshot

Manage tags

IOPS [Info](#)

3000

Min: 3000 IOPS, Max: 16000 IOPS.

Throughput (MiB/s) [Info](#)

125

Min: 125 MiB, Max: 1000 MiB. Baseline: 125 MiB/s.

Availability Zone [Info](#)

ap-south-1c

Fast snapshot restore [Info](#)

Not enabled for selected snapshot

Encryption

Use Amazon EBS encryption as an encryption solution for your EBS resources associated with your EC2 instances.

☐ Encrypt this volume

- Choose the Availability zone where you want to create that volume.
- We have choose ap-south-c

6. Verification:

- After this Attach this volume to other instance in us-south-c

Attach volume [Info](#)

Attach a volume to an instance to use it as you would a regular physical hard disk drive.

Basic details

Volume ID

 vol-0beaa674e472e9ae6

Availability Zone

us-west-1b

Instance [Info](#)

i-0d5df07a3ed6ea019




Only instances in the same Availability Zone as the selected volume are displayed.

Device name [Info](#)

/dev/sdb

Recommended device names for Linux: /dev/xvda for root volume. /dev/sd[f-p] for data volumes.

 Newer Linux kernels may rename your devices to **/dev/xvdf** through **/dev/xvdp** internally, even when the device name entered here (and shown in the details) is **/dev/sdf** through **/dev/sdp**.

- Connect to that instance and mount the volume to /mydata folder.

```
[ec2-user@ip-172-31-24-67 ~]$  
[ec2-user@ip-172-31-24-67 ~]$ sudo mkdir /mydata  
[ec2-user@ip-172-31-24-67 ~]$ sudo mount /dev/xvdb /mydata  
[ec2-user@ip-172-31-24-67 ~]$  
[ec2-user@ip-172-31-24-67 ~]$ sudo df -h  
Filesystem      Size  Used Avail Use% Mounted on  
devtmpfs        4.0M   0  4.0M   0% /dev  
tmpfs           475M   0  475M   0% /dev/shm  
tmpfs           190M  2.9M  188M   2% /run  
/dev/xvda1      8.0G  1.6G  6.5G  20% /  
tmpfs           475M   0  475M   0% /tmp  
/dev/xvda128    10M  1.3M  8.7M  13% /boot/efi  
tmpfs           95M   0   95M   0% /run/user/1000  
/dev/xvdb       9.8G  40K  9.3G   1% /mydata  
[ec2-user@ip-172-31-24-67 ~]$
```

- And see whether the same data is present or not.
- Hence the verification is successful.