

++++++  
 Amazon EBS Volume  
 ++++++

Elastic Block Store volume is a block level storage device that can be associated with an EC2 instance

EBS Volumes can be used as both primary storage and secondary storage

The primary EBS volume acts as root volume and should be created and attached to the instance at the time of instance launch.

Storage can be increased in the future if needed. This primary EBS volume cannot be detached from the instance.

The secondary volume can be attached, detached and modified at any time (Additional Volume)

An instance can have one primary EBS volume and multiple secondary volumes

One Primary EBS Volume can be only associated with one instance and we can attach a single secondary EBS Volume to multiple Instances, provided they are in same Availability Zone.

There are 5 types of EBS volumes:

+++++

General Purpose SSD (gp2) -- Provides balance of both price and performance and is generally chosen by default.

Provisioned IOPS SSD (io1) -- Most expensive of the volume types with highest performance and well-suited for tasks with heavy workloads.

Throughput Optimized HDD (st1) -- A low-cost volume that focuses on optimizing throughput and is generally used for large sequential workloads dealing with big data warehouses. These volumes cannot be used as root volumes for EC2 instances.

Cold HDD (sc1) -- least expensive of the volume types and specifically designed for workloads which are accessed less frequently. These volumes also cannot be used as root volumes for EC2 instances.

Magnetic (Standard) -- Previous generation magnetic volumes which cannot be used as root volumes for EC2 instances

#### Resizing EBS Volumes

-----

In this digital world of ever enlarging data it's not enough to build solutions with hard-coded amount of storage that cannot scale.

If the volume associated with the EC2 instance fills up, we have to increase the size of the volume.

Advantages of using EBS Volumes  
 High availability and flexibility

Data can be kept persistently on a file system even after shut downing the instance

Enables snapshots, which capture the data stored at a point in time and can be restored at any time.

The snapshots enables us to create a volume and attach it to another instance if needed.

Can be resized at any time as and when required

Comes equipped with encryption (and encryption-at-rest).

EBS Volumes can be attached, detached and associated with other instances at any point in time (exception the primary volume)

+++++

**\*\*Today's Lab Task\*\***

+++++

- 1) Create an EC2 Linux instance with Amazon Linux 2 AMI (it will have EBS root volume with 8 GB)
- 2) Create Additional EBS volume with 10 GB
- 3) Attach EBS Additional volume of 10 GB to EC2 for storing data
- 4) Connect to EC2 instance and verify volumes (EBS both volumes shud display)
- 5) Create a directory and mount EBS Additional volume to created directory
- 6) Store the data in EBS Additional volume mounted directory (create few files)
- 7) Detach EBS Additional volume from EC2 and stop/terminate EC2 instance
- 8) Create new EC2 instance and attach previously created EBS Additional volume
- 9) Connect to new EC2 instance and verify volumes (EBS volume shud display)
- 10) Create a directory and mount EBS Additional volume to created directory
- 11) Verify data which stored previously in EBS available or not (it should present)
- 12) Stop Instances and delete volumes

+++++

List the attached EBS volumes: â€œ

\$ lsblk

Check if the volume has any data using the â€œsudo file -s /dev/xvdfâ€œ command.

\$ sudo file -s /dev/xvdf

Format partition using the command â€œsudo mke2fs -j /dev/xvdfâ€œ.

\$ sudo mke2fs -j /dev/xvdf

Mounting this disk partition as â€œnewdiskâ€œ

a. Switch to root by using

\$ sudo su

b. create directory with a name newdisk

\$ sudo mkdir newdisk

c. Mount EBS volume to newdisk using the command

\$ sudo mount /dev/xvdf newdisk

d. Check mounts using df -h

\$ df -h

Navigate to newdisk directory (it is available under root (/))

\$ cd newdisk

Creat new file in newdisk directory and store the data

\$ touch f1.txt

-----  
1) Detach Volume from EC2 instance and Stop EC2 instance

2) Create New EC2 instance

3) Attach EBS Additional Volume

4) Connect to newly created EC2 instance

5) Check volumes available in EC2 using (lsblk command)

6) Create directory

\$ mkdir newdrive

6) Mount EBS volume to directory

```
$ sudo su
$ sudo mount /dev/xvdf newdrive
```

5) Verify data available in EBS volume came to our directory or not

```
$ ls -l newdrive
```

```
+++++
```

=> Root volumes contains operating system

=> We will use EBS Additional volumes to store our application & application configuration files

```
+++++
```