**AWS – Elastic Block Store (Volumes)**

EBS stands for Elastic Block Store, and it is a block-level storage service provided by Amazon Web Services (AWS). EBS allows you to create and attach persistent block storage volumes to your EC2 instances, enabling you to store data and use it independently from the lifecycle of the instances.

* 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.
* Additional volumes can be attached/detached [mounted and un-mounted to a path].

**There are 5 types of EBS volumes**

1. **General Purpose SSD (gp2)** -- Provides balance of both price and performance and is generally chosen by default.
2. **Provisioned IOPS SSD (io1)** -- Most expensive of the volume types with highest performance and well-suited for tasks with heavy workloads.
3. **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.
4. **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.
5. **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 is 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 side 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).

**EBS Volumes 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 should 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 should 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:**

$ lsblk

**Check if the volume has any data using the sudo file -s /dev/xvdf command.**

$ sudo file -s /dev/xvdf

**Format partition using the command sudo mke2fs -j /dev/xvdf .**

$ sudo mke2fs -j /dev/xvdf

**Create directory with a name newdisk**

$ sudo mkdir newdisk

**Mount EBS volume to newdisk using the command**

$ sudo mount /dev/xvdf newdisk

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

$ cd newdisk

**Create new file in newdisk directory and store the data**

$ touch f1.txt

**To Un-mount Disk with Instance**

$ sudo umount /dev/xvdf

1) Detach Volume from EC2 instance and Stop EC2 instance

2) Create New EC2 instance (make sure they are in same AZe)

3) Attach EBS Additional Volume.

4) Connect to newly created EC2 instance.

5) Check volumes available in EC2 using (lslbk 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.

**EBS Snapshots**

EBS snapshots in Amazon Web Services (AWS) are point-in-time copies of EBS volumes. They capture the entire state of an EBS volume, including all the data stored on it, and store it in Amazon S3 (Simple Storage Service).

* Snapshots are used to create Volume backups.
* Snapshots are regional specific (volumes are zone specific).
* Snapshot can't be attached with EC2 instance (volume can be attached with EC2).
* From Volume we can create a snapshot and from snapshot we can create volume

Snapshot --------------> volume ---------------> snapshot

**Snapshot Lab Task**

1) Create EC2 instance in 1a availability zone (It will have default root volume in 1a zone)

2) Connect to EC2 instance and create few files in Ec2 instance (files will store in root volume)

3) Create Snapshot from the root volume of our EC2 instance (volume is in 1a zone)

4) From Created Snapshot create EBS volume in 1b zone

5) Create EC2 instance in 1b availability zone (it will create default root volume in 1b zone)

6) Attach the volume (created using snapshot) to EC2 instance which we have created in 1b AZ

7) Mount the volume to a directory and see the data.