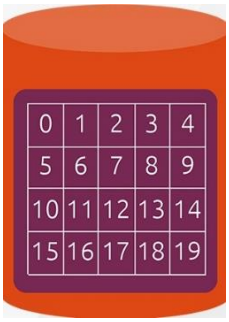
AWS EBS

Elastic Block Storage

❖ EC2 BLOCK STORAGE:

- Block storage chops data into blocks.
- Block storage, sometimes referred to as **Block-Level Storage**, is a technology that is used to store data files on **Storage Area Networks (SANs)** or cloud-based storage environments.
- Block storage breaks up data into blocks and then stores those blocks as separate pieces, each with a unique identifier.
- When a user or application requests data from a block storage system, the underlying storage system reassembles the data blocks and presents the data to the user or application.



➢ EC2 INSTANCE ROOT DEVICE VOLUME TYPES:

- When you launch an instance, the root device volume contains the image used to boot the instance.
- All AMIs are categorized as either backed by backed by instance store or Amazon EBS.

INSTANCE STORE-BACKED AMI: The root device for an instance launched from the AMI is an instance store volume created from a template stored in Amazon S3.

EBS-BACKED AMI: The root device for an instance launched from the AMI is an Amazon Elastic Block Store (EBS) volume created from an Amazon EBS snapshot.

➢ EC2 INSTANCE STORE:

- Instance store volumes provide temporary block-storage.
- These are virtual devices whose underlying hardware is physically attached to the host computer that is running the instance.
- These volumes are considered **Ephemeral data**, meaning the data on the volumes only exists for the duration of the life of the instance.
- data in the instance store is lost under any of the following circumstances:
    - The underlying disk drive fails
    - The instance stops
    - The instance hibernates
    - The instance terminates

ADVANTAGES:

- Very fast I/O (2-100X of EBS)
- No Extra cost. It is included in the cost of EC2 instance.
- Ideal for storing temporary information – cache, scratch files etc.

DISADVANTAGES:

- Slow boot up (up to 5 minutes)
- Ephemeral storage (data is lost when hardware fails or instance is terminated)
- Cannot take a snapshot or restore from snapshot
- Fixed size based on instance type
- You cannot detach and attach it to another EC2 instance.

➢ ELASTIC BLOCK STORE (EBS):

- EBS provides persistent block storage volumes for EC2 instances in the AWS cloud.
- Each EBS volume is automatically replicated within its Availability Zone.
- EBS volumes are **highly available** and **reliable storage volumes** that can be attached to any running or stopped instance that is in the same AZ.
- An EBS volumes can be attached to only one instance at a time, but multiple volumes can be attached to a single instance.

EBS PERFORMANCE:

- EBS volumes measure input/output operations in **IOPS**.
- AWS measures **IOPS** in 256kb chunks.
- For Example, a 512kb operation would count as **2 IOPS**.

➢ EBS VOLUME TYPES:

- EBS provides the following volume types, which differ in performance characteristics and price:
  - Solid State Drive
  - Hard Disk drive
  - Previous Generation

SOLID STATE DRIVE (SSD):

- SSD is the best option of IOPS-intensive use cases such as tractional workloads, databases & boot volumes.

GENERAL PURPOSE SSD (gp2):

- Provides a balance of price and performance. We recommend these volumes for most workloads.
- Volume size of 1Gib to 16Tib.

PROVISIONED IOPS SSD (io1):

- Provides high performance for mission-critical, low-latency, or high-throughput workloads.
- Volume size of 4Gib to 16Tib.

HARD DISK DRIVE (HDD):

- HDD is the best option for throughput-intensive use cases like storage, MapReduce and log processing.

THROUGHPUT OPTIMIZED HDD (st1):

- A low-cost HDD designed for frequently accessed, throughput-intensive workloads.
- The volume size of 500Gib to 16Tib.

COLD HDD (sc1):

- Lowest cost HDD volumes designed for less frequently accessed workloads.
- The volume size of 500Gib to 16Tib.

PREVIOUS GENERATION:

MAGNETIC (STANDARD):

- Magnetic volumes are backed by hard disk drives and can be used for workloads with smaller datasets where data is accessed infrequently or when performance consistency isn't of primary importance.
- Low storage cost.
- Volume size of Min 1Gib Max 1Tib.

➢ EBS SNAPSHOT:

- A Snapshot is a point-in-time backups of EBS Volumes that are stored in S3.
- Snapshot is a good solution for a disaster-recovery of your EBS volumes.
- Snapshots are **incremental** in nature. Only changed blocks are saved, thus reducing costs.

WHY USE EBS SNAPSHOTS:

- Backup data on EBS Volumes
- Meet Recovery Point Objectives (RPO)
- Copy Volumes within or across Availability Zones
- Copy Volumes to another region for Disaster Recovery
- Create Amazon Machine Images (AMI's)

➢ EBS ENCRYPTION:

- EBS uses **KMS Customer Master Keys (CMK's)** to generate data (Encryption) keys to encrypt and decrypt data on EBS volumes.
- Data is encrypted on the host of the EC2 instance. This means data in-transit to an encrypted EBS volume is also encrypted.
- Turning on Encryption automatically encrypts:

DATA AT REST:

- Data volumes and boot volumes
- Snapshots

DATA IN TRANSIT:

- Between EC2 instances and EBS volumes
- Between EBS volumes and EBS snapshots

UNENCRYPTED SNAPSHOTS:

- Snapshot can be shared with all AWS community by modifying permissions to Public.
- Snapshots can also be shared with select AWS accounts (Permission needs to be private).

ENCRYPTED SNAPSHOTS:

- Can't be shared as public snapshots. Can only be shared with select accounts
- The receiving accounts must be given permissions on the CMK used to encrypt the shared snapshots.
- Encrypted snapshot that was encrypted by the default CMK cannot be shared

➢ DATA LIFECYCLE MANAGER (DLM):

- DLM is a total solution for creating, deleting, and retaining EBS volume snapshots on a custom schedule.
- You can configure snapshot lifecycle policies to carry the required EBS snapshot tasks.
- A lifecycle policy applies to any of the tags specified.
- DLM will apply AWS tags on snapshots creation for easier management.
- Can also automate EBS snapshots with CloudWatch events, but that is for individual EBS volumes.
  - Define policies for regular backup schedules.
  - Retain backups for compliance/audit purpose.
  - Control snapshot costs by automatically deleting old backups.
  - Identify volumes to backup using tags.
  - Use IAM to control DLM policy access.