Volume Types

**General Purpose SSD**[**¶**](https://notes.arkalim.org/notes/aws%20solutions%20architect%20associate/elastic%20block%20storage%20(ebs)/#general-purpose-ssd)

* Good for system boot volumes, virtual desktops
* Storage: 1 GB - 16 TB
* **gp3**
  + **3,000 lOPS baseline** (max 16,000 - independent of size)
  + 125 MiB/s throughput (max 1000MiB/s - independent of size)
* **gp2**
  + **Burst IOPS up to 3,000**
  + **\3 IOPS per GB**
  + **Max IOPS: 16,000** (at 5,334 GB)

**Provisioned IOPS SSD\*\***[**¶**](https://notes.arkalim.org/notes/aws%20solutions%20architect%20associate/elastic%20block%20storage%20(ebs)/#provisioned-iops-ssd)

* Optimized for **Transaction-intensive Applications** with high frequency of **small & random IO operations**. They are sensitive to increased I/O latency.
* Maintain high IOPS while keeping I/O latency down by maintaining a **low queue length** and a high number of IOPS available to the volume.
* **Supports EBS Multi-attach** (not supported by other types)
* **io1** or **io2**
  + Storage: **4 GB - 16 TB**
  + Max IOPS: **64,000 for Nitro EC2 instances** & **32,000 for non-Nitro**
  + **50 lOPS per GB** (64,000 IOPS at 1,280 GB)
  + io2 have more durability and more IOPS per GB (at the same price as io1)
* **io2 Block Express**
  + Storage: 4 GiB - **64 TB**
  + Sub-millisecond latency
  + Max IOPS: 256,000
  + 1000 lOPS per GB

**Hard Disk Drives (HDD)**[**¶**](https://notes.arkalim.org/notes/aws%20solutions%20architect%20associate/elastic%20block%20storage%20(ebs)/#hard-disk-drives-hdd)

* Optimized for **Throughput-intensive Applications** that require **large & sequential IO operations** and are less sensitive to increased I/O latency (big data, data warehousing, log processing)
* Maintain high throughput to HDD-backed volumes by maintaining a **high queue length** when performing large, sequential I/O
* **Cannot be used as boot volume** for an EC2 instance
* Storage: 125 MB - 16 TB
* **Throughput Optimized HDD (st1)**
  + Optimized for large sequential reads and writes (Big Data, Data Warehouses, Log Processing)
  + **Max throughput: 500 MB/s**
  + **Max IOPS: 500**
* **Cold HDD (sc1)**
  + For infrequently accessed data
  + Cheapest
  + **Max throughput: 250 MB/s**
  + **Max IOPS: 250**

Encryption[¶](https://notes.arkalim.org/notes/aws%20solutions%20architect%20associate/elastic%20block%20storage%20(ebs)/#encryption)

* Optional
* For Encrypted EBS volumes
  + Data at rest is encrypted
  + **Data in-flight between the instance and the volume is encrypted**
  + All snapshots are encrypted
  + All volumes created from the snapshot are encrypted
* Encrypt an un-encrypted EBS volume
  + Create an EBS snapshot of the volume
  + Copy the EBS snapshot and encrypt the new copy
  + Create a new EBS volume from the encrypted snapshot (the volume will be automatically encrypted)

Snapshots[¶](https://notes.arkalim.org/notes/aws%20solutions%20architect%20associate/elastic%20block%20storage%20(ebs)/#snapshots)

* **Data Lifecycle Manager (DLM)** can be used to automate the creation, retention, and deletion of snapshots of EBS volumes
* Snapshots are incremental
* Only the most recent snapshot is required to restore the volume

RAID[¶](https://notes.arkalim.org/notes/aws%20solutions%20architect%20associate/elastic%20block%20storage%20(ebs)/#raid)

* **RAID 0**
  + Improve performance of a storage volume by distributing reads & writes in a stripe across attached volumes
  + If you add a storage volume, you get the straight addition of throughput and IOPS
  + For high performance applications
* **RAID 1**
  + Improve data availability by mirroring data in multiple volumes
  + For critical applications

**EFS Storage tiers**

* **Standard** - for frequently accessed files
* **Infrequent access (EFS-IA)**- cost to retrieve files, lower price to store

**Security**

* EFS Security Groups to control network traffic
* POSIX Permissions to control access from hosts by IAM User or Group

**AMI -**When the new AMI is copied from region A into region B, it automatically creates a snapshot in region B because AMIs are based on the underlying snapshots.

**Instance Metadata**

* Url to fetch metadata about the instance (http://169.254.169.254/latest/meta-data)
* This URL is internal to AWS and can only be hit from the instance

**Sticky Sessions(Sessions Affinity)**

* Requests coming from a client is always redirected to the same instance based on a cookie. After the cookie expires, the requests coming from the same user might be redirected to another instance.
* **Only supported by CLB & ALB**
* Used to ensure the user doesn’t lose his session data, like login or cart info, while navigating between web pages.
* Stickiness may cause load imbalance
* Cookies could be
  + Application-based (TTL defined by the application)
  + Load Balancer generated (TTL defined by the load balancer)

**Auto Scaling group:** Even if an ASG is deployed across 3 AZs, minimum number of instances to remain highly available is still 2. If you have an ASG with running instances and you delete the ASG, the instances will be terminated and the ASG will be deleted.

**Workflows**

**Step Functions**

* Used to build serverless workflows to orchestrate Lambda functions
* Represent flow as a JSON state machine
* Maximum workflow execution time: 1 year
* Features: sequence, parallel, conditions, timeouts, error handling, etc.

**Simple Workflow Service (SWF)**

* Outdated service (step functions are preferred instead)
* Code runs on EC2 (not serverless)
* **Ensures that a task is never duplicated** (could replace standard SQS queues)
* 1 year max runtime
* **Built-in human intervention step**
* Step Functions are recommended to be used for new applications, except:
  + If you need external signals to intervene in the processes
  + If you need child processes that return values to parent processes

**S3:** S3 delivers strong read-after-write consistency (if an object is overwritten and immediately read, S3 always returns the latest version of the object). S3 is strongly consistent for all GET, PUT and LIST operations.

**S3 Static website**

* <bucket-name>.s3-website-<region>.amazonaws.com
* <bucket-name>.s3-website.<region>.amazonaws.com

**Pre-Signed URL**

* Pre-signed URLs for S3 have temporary access token as query string parameters which allow anyone with the URL to temporarily access the resource before the URL expires (default 1h)
* Pre-signed URLs inherit the permission of the user who generated it
* Uses:
  + Allow only logged-in users to download a premium video
  + Allow users to upload files to a precise location in the bucke

**Aurora**

* **Writer Endpoint** (Cluster Endpoint)
  + Always points to the master (can be used for read/write)
  + Each Aurora DB cluster has one cluster endpoint
* **Reader Endpoint**
  + Provides load-balancing for read replicas only (used to read only)
  + If the cluster has no read replica, it points to master (can be used to read/write)
  + Each Aurora DB cluster has one reader endpoint
* **Custom Endpoint:**
  + Used to point to a subset of replicas
  + Provides load-balanced based on criteria other than the read-only or read-write capability of the DB instances like instance class (ex, direct internal users to low-capacity instances and direct production traffic to high-capacity instances)

**High Availability and Read scaling**

* **Writer Endpoint** (Cluster Endpoint)
  + Always points to the master (can be used for read/write)
  + Each Aurora DB cluster has one cluster endpoint
* **Reader Endpoint**
  + Provides load-balancing for read replicas only (used to read only)
  + If the cluster has no read replica, it points to master (can be used to read/write)
  + Each Aurora DB cluster has one reader endpoint
* **Custom Endpoint**:
  + Used to point to a subset of replicas
  + Provides load-balanced based on criteria other than the read-only or read-write capability of the DB instances like instance class (ex, direct internal users to low-capacity instances and direct production traffic to high-capacity instances)

**FSx**

* Allows us to launch **3rd party high-performance file systems** on AWS
* Useful when we don’t want to use an AWS managed file system like S3
* **Can be accessed from your on-premise infrastructure**

**FSx for windows**

* Shared File System for Windows (like EFS but for Windows)
* Supports **SMB** protocol, Windows **NTFS**, Microsoft **Active Directory** integration, ACLs, user quotas
* Built on SSD, scale up to 10s of GB/s, millions of IOPS, 100s PB of data
* Supports Multi-AZ (high availability)
* Data is backed-up daily to S3
* **Does not integrate with S3** (cannot store cold data)

**Redshift**

* AWS managed **data warehouse** (10x better performance than other data warehouses)
* Based on **PostgreSQL**
* Used for **Online Analytical Processing (OLAP)** and high performance querying
* **Columnar storage** of data with **massively parallel query execution** in **SQL**
* Faster querying than [Athena](https://notes.arkalim.org/notes/aws%20solutions%20architect%20associate/Athena) due to indexes
* Can be used for both long complex queries as well as short fast queries
* Need to provision instances as a part of the Redshift cluster (pay for the instances provisioned)
* Auto-healing feature

Loading data in Redshift

1. **S3**
   * Use **COPY command** to load data from an S3 bucket into Redshift
   * **Without Enhanced VPC Routing**
     + data goes through the public internet
   * **Enhanced VPC Routing**
     + data goes through the VPC without traversing the public internet
2. **Kinesis Data Firehose**
   * **Sends data to S3** and issues a **COPY** command to load it into Redshift
3. **EC2 Instance**
   * Using **JDBC driver**
   * Used when an application needs to write data to Redshift
   * Optimal to write data in batches

**Neptune**

* AWS managed **graph database**
* Used for **high relationship data** (eg. social networking)
* Highly available across **3 AZ** with up to **15 read replicas**
* **Point-in-time recovery** due to continuous backup to S3
* Support for KMS encryption at rest + HTTPS for in-flight encryption