WHAT IS AWS EBS (ELASTIC BLOCK STORE)?

AWS Elastic Block Store (EBS) is Amazon's block-level storage solution used with the EC2 cloud service to store persistent data. This means that the data is kept on the AWS EBS servers even when the EC2 instances are shut down. EBS offers the same high availability and low-latency performance within the selected availability zone, allowing users to scale storage capacity at a low subscription-based pricing model. The data volumes can be dynamically attached, detached and scaled with any EC2 instance, just like a physical block storage drive. As a highly dependable cloud service, the EBS offering guarantees 99.999% availability.

What is an EBS volume?

An EBS volume is a virtual unit of logical storage provided by Amazon Web Services (AWS), that acts like an extremely reliable and high-performance hard drive. High-performance means that it can read and write very quickly. Like a physical hard drive, you can format it to handle different file systems, and plug and unplug it from an EC2 instance. An EBS volume is durable to reliably store your data with little risk of data loss. It is also persistent, meaning you can stop and restart it and the data remains. Unlike a physical hard drive, an EBS volume is scalable so that you can size it to fit your needs.

AWS EBS Volume Types

AWS Elastic Block Storage is different from the standard EC2 Instance Store, which merely provides temporary storage available on the physical EC2 host servers. The EC2 Instance Store is useful for temporary data content such as caches, buffers or files that are replicated across the hosted servers. For data that needs to be available persistently, regardless of the operating life of an EC2 instance,

the AWS EBS offers the following storage volume options:

- **General Purpose SSD (gp2):** An optimum balance between cost and performance for a variety of IT workloads. Use cases include virtual desktops, apps, dev and test environments, among others.
- **Provisioned IOPS SSD (io1):** The high-performance functionality serves particularly well for mission-critical IT workloads. Suitable use cases include large databases and business apps that require 16,000 IOPS or 250 MiB/s of throughput per volume.
- **Throughput Optimized HDD (st1):** A low cost alternative for large storage volume workloads with high performance throughput requirements. Examples include streaming workloads, big data applications, log processing and data warehousing.
- **Cold HDD (sc1):** An inexpensive alternative for use cases with a requirement to maintain minimal cost for large volume data storage. Examples include workloads that are accessed less frequently.

It's important to note that each storage option doesn't represent individual physical storage media, but a distributed system of storage options as per the categorized volume options. This AWS resource provides a detailed overview of the various EBS volume types.

AWS EBS includes powerful features that make it easier to store persistent data automatically and reliably while optimizing cost investments on the cloud storage. The prominent features include:

(This tutorial is part of our AWS Guide. Use the right-hand menu to navigate.)

Amazon EBS Snapshots

This feature allows point-in-time storage of data volumes incrementally, while only charging for the change in data volume. For instance, if 5GB of data was added to an existing 100GB of storage block with the snapshot, AWS will only charge for the additional 5GB of data. Snapshots can be expanded, replicated, moved, shared, copied, modified, managed and organized within and across AWS Availability Zones using the Amazon Data Lifecycle Manager and the Tag feature. All EBS Snapshots are stored in AWS S3 that guarantee up to 11x9's of durability. Snapshots are not stored as user accessible objects but accessed via the EBS API. The Snapshots are stored behind the Amazon Machine Images (AMI), providing all necessary information to recover data and launch EC2 instances in the cloud accordingly.

The Snapshot capability is key to business continuity plans for mission-critical apps and services. Users can define Recovery Time Objectives (RTO) and Recovery Point Objectives (RPO) and manage the snapshots to meet those objectives. In addition to the data backup and disaster recovery objectives, customers also use EBS Snapshots to capture production data for testing and development. EBS Snapshots and volumes also support encryption, allowing users to create custom CMK when needed from the AWS IAM management console.

Amazon EBS–Optimized Instances

The EBS Optimized Instances offer a burst of performance improvements for storage workloads that require short and intense periods of high device I/O operations. The throughput performance for EBS-optimized instances can vary between 4250 to 14,000 Mbps based on the instance type. For instance, the SSD GP2 volume option is designed to operate within 10 percent of its baseline and

burst performance, for 99 percent of the time that it's used as such. This capability allows low spec instances to replicate the high performance of larger instances for a limited period of the day. This feature allows users to right-size their instances while accommodating EBS demand spikes. As a result, the EBS volumes are optimized for a variety of storage use cases and the demand spikes do not impact end-user or customer experience. The EBS solutions are optimized by default or available on a low hourly pricing.

Details are available on Amazon EBS–Optimized Instances guide here.

Other notable features of the AWS EBS include:

- **Amazon EBS and NVMe:** AWS offers Non-Volatile Memory Express (NVMe) devices for its Nitro-based instance types designed for high performance computing. The Nitro hypervisor system reduces virtualization overhead and offers the same high performance as bare metal hardware. More details here.
- **Amazon EBS Volume Performance:** Since the AWS EBS is delivered as a raw storage solution, users need to configure and tune the systems to maximize performance on Linux instances. AWS offers a detailed guideline on producing optimal configurations for real-world workloads and benchmarking for a variety of I/O characteristics, EC2 instances and EBS volumes. More details here.
- **Amazon EBS Encryption:** the encryption solution supports all EBS volume types to secure data at rest, in transition, snapshots and instances. AWS promises minimal impact on the performance on the storage volume due to encryption. Users can manage control, policies and security of snapshots and volume data using the IAM and AWS Key Management Service (AWS KMS). More details here.
- **EBS CloudWatch Events:** Users can set policies and automated actions in response to EBS event notifications. These actions may include operational changes to the environment, activating AWS Lambda functions, batch jobs, adjusting EBS capacity and others. More details here.

The choice for AWS EBS volumes should be a part of an exhaustive cloud management strategy, along with the necessary resources on maximizing the value potential of the available AWS EBS tooling and features. The strategy should include an in-depth analysis of the storage volume I/O performance and data transmission throughput requirements, backup and disaster recovery objectives, as well as budgeting and financial planning for scalable workloads. With the vast selection of Amazon Elastic Block Store, businesses can then choose appropriate storage capabilities for a variety of use cases while maintaining an optimal balance between cost, performance and dependability of their persistent data storage portfolio.

Drawbacks of AWS EBS

While AWS EBS features are extremely useful for many applications, there are also drawbacks. Consider these while considering EBS for your needs.

1. EBS is not recommended for temporary storage

EBS is designed for persistence and durability, two features that are not needed for short-term use. It is expensive because you pay for the storage provisioned, not storage used. If you use a small fraction of an EBS for a few hours and then delete the data, you still have to pay for the entire

capacity. If you don't remember to manually delete the volume and snapshots, you will keep paying. A better alternative is EC2 instance storage.

2. Limitations in multi-instance access

You cannot share EBS volumes across multiple instances. Each EBS volume can attach to only one EC2 at a time. If you have multiple EC2 instances that need to use the same data, you have to manually attach and detach the EBS volume to each instance that needs the data. An alternative approach is to employ an Amazon Elastic File System (EFS) for shared data storage.

3. Limited durability compared to AWS S3 and AWS EFS

Both AWS S3 and AWS EFS have superior durability over EBS because they are replicated across many availability zones. An EBS functions in just one availability zone, so it is at greater risk if there is an outage. S3 has a distributed architecture, putting it at the top when it comes to durability. It is best used for large-scale and unstructured data. Amazon EFS serves multiple instances and is nearly as durable as S3. EBS is best studied for structured databases where low-latency and high performance are required.

4. Network latency issues

EBS communicates over the AWS network and is not directly attached to an instance, which makes for higher latency. When you need faster read/write speeds, EC2 instance store is the better choice.

5. Complexity in linkage and volume management

You cannot globally access or share EBS volumes, so you need to manually attach them to an instance. They support a specific level of IOPS, meaning you must reconfigure and reformat if you want to make a switch. Each attachment and detachment introduces the risk of data corruption. Lastly, scaling is a manual process. You will need to maintain proper documentation and tracking, which adds up to a complex management challenge.

6. Confusing naming conventions

Naming conventions are not automatic with AWS EBS, and the official AWS volume ID is not what your EC2 instance uses. The result can be disorganization and confusion. You will have to invest time and effort into managing all the naming confusion — resources that are better used elsewhere.

7. Scaling gets expensive

If you over-provision, you pay more. As you add snapshots, you also add a cost. If you use high-performance io2 volumes, you pay even more of a premium.