

EC2 Instance Storage

EBS Volume

Elastic block store volume is a network drive that you can attach to your EC2 instance while they run. It is a volume which means you have to provision the capacity first, define how many GB of EBS storage you want in advance and IO/S.

It is storage for your EC2 instances. It is like SSD/hard drive for your virtual machine, they allow storing data and let them persist just like a normal SSD/hard drive. Except they are connected via network rather than physically.

EBS volume can be mounted to one instance at a time! **And they are bound to a specific availability zone.** Meaning you can not attach a EBS in us-east-1a to an EC2 in us-east-1b! In order to move the volume across a availability zone you have to snapshot it to do so.

They can be detached quickly compared to physical drive that you have to plug it out physically, and attach to another EC2 instance.

You will be billed for the provisioned capacity.

Delete on termination attribute

By default, the root EBS volume for EC2 is deleted on termination, this option is checked by default

By default, any other attached EBS volume are not deleted on termination, this option is not checked by default

You can choose to preserve root EBS volume by unchecking the box to save some data if you so choose.

EBS snapshots

Is a backup of your EBS volume at any point of time. You don't need to detach it to do snapshot but is recommended.

Then you can copy snapshot across AZ or Region.

You can also recreate a EBS volume from a snapshot which contains all of the data from the snapshot. This is how you would be copying data across region/AZ.

EBS snapshot archive

You can move a snapshot to an "archive tier" that is 75% cheaper, but the reason why it is cheaper because it will take 24 to 72 hours to restore the archive. Cheap storage but expensive retrieval.

EBS recycle bin

You can set up rules that to retain deleted snapshots up from 1 day to 1 year for you to recover in case of accidental deletion.

Fast snapshot restore

Force full initialization of snapshot to have no latency on the first use. This is recommended if your snapshot is large, however, this causes lots of money.

EBS volume type

When you pick EBS you get to choose the actual hardware underneath that's backing up your storage. Currently there are six types:

1. gp2/gp3 SSD: General purpose SSD volume. Balance price and performance
2. io1/io2 SSD: Highest-performance SSD for mission-critical low-latency or high-throughput workloads. Use this if your EC2 contains some very important task that needs the IO speed
3. std1 HDD: Low cost hard-disk drive that is designed for frequent access, throughput-intensive work
4. sc1 HDD: Lowest cost hard-disk drive for less frequent accessed workload

Only gp2/gp3 and io1/io2 can be used as root volumes for booting

Throughput = How fast can your storage read/write data. This is measured in MB/s. How much data can you transfer either read or data from/to the disk can you do in a second.

IOPS = Input/output operations per second. Tells you the amount of read/write operation that your storage can do in a second. If this number is small then under heavy use your drive will not be able to keep up with the request and might drop some. If say someone is reading a picture from the drive at the same time another person is storing things into the disk.

General purpose SSD (gp2/gp3)

It is cost effective storage with low latency.

1 Gib - 16 TiB capacity

gp3 is the newer volume type, the old one is gp2. It allows you to increase IOPS to 16,000 and throughput up to 1000 MiB independently.

gp2 is the older type. The size of the volume and IOPS are linked, max at 16,000. With 3 IOPS per GB.

Provisioned IOPS SSD (io1/io2)

This is for critical business application that require HIGHEST IO performances, or those that need more than 16,000 IOPS.

This is good for database workloads that is sensitive to the speed of the input/output writing speeds.

4 GiB - 16 TiB capacity

The IOPS for io1/io2 are capped at 32,000 for normal EC2 instances. But for nitro EC2 instances they are boosted to 64,000 IOPS.

Now-days there is no reason to pick io1 since io2 offers more durability and gives more IOPS per GiB that you allocate at the same price.

io2 Block Express: Gives you even more IOPS up to 256,000 maximum! Sub-millisecond latency.

IOPS EBS support multi-attach!

Hard disk drives

Cannot be used as boot volume

125 GiB - 16 TiB

Throughput optimized are st1, and have max IOPS of 500. Good for big data, data warehouses, log processing, dumping for large amount of data.

For data that are infrequently accessed good to use cold hard-drive disk sc1. Max IOPS is 250, gives you the lowest cost.

EBS Multi-attach

This feature is only available to the io1/io2 family. It allows you to attach the same EBS volume to multiple EC2 instances that is in the same availability zone. The EBS volume of course needs to also be in the same availability duh.

Recommended for higher application availability in clustered Linux application, or if your application need to do concurrent write.

You are limited to 16 EC2 instances at a time. No more!

EBS encryption

If your EBS is encrypted then you get the following:

- Data is encrypted inside the volume
- Data in transit to and from an instance is encrypted
- All the snapshots are encrypted
- All volumes created from snapshots are also encrypted as well

The encryption leverages keys from KMS services, and has minimum impact on performance.

Copying an unencrypted snapshot allows encryption! **Create an EBS snapshot of the unencrypted volume -> encrypt the EBS snapshot using the Copy snapshot function -> Create new EBS volume from the snapshot which is now encrypted**

EFS

Elastic file system. It is a managed Network file system (NOT NAS, network attached file storage like EBS!!!) that can be mounted on many EC2.

EFS works with EC2 instances in different availability zones.

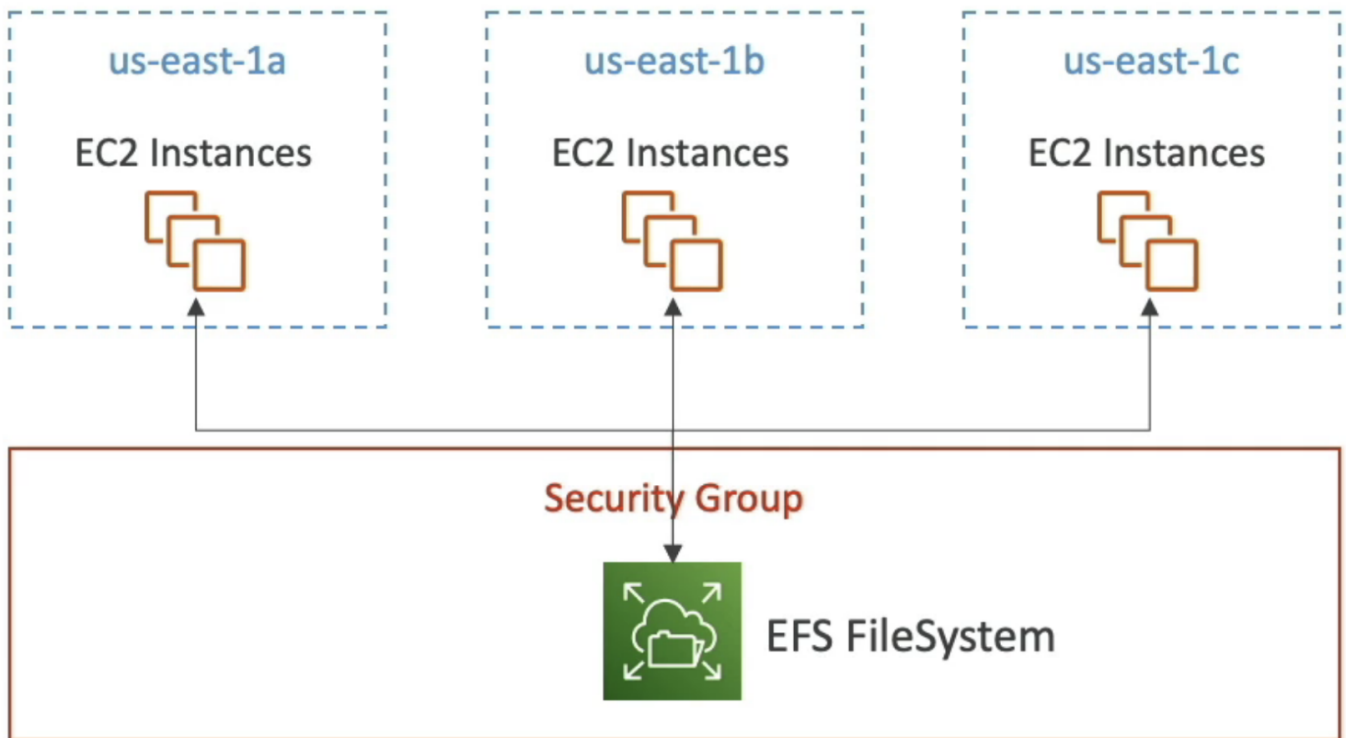
It is highly available, scalable, but very expensive compared to EBS. It has a pay per use model (how many GiB you read/write you will be paying it in \$/GiB you used).

Recommended for content management, web serving, data sharing between EC2 instances, and Wordpress?

A security group is used to control access to the EFS. AND EFS is only compatible with Linux based AMI no Windows machine can access the EFS. You can enable encryption, and is a standard file system in Linux.

When you create EFS you will have to attach security groups in order to allow EC2 instances with the corresponding security group to attach to it. Remember Security group you can define actual rules of traffic or reference another security group, if the EC2 instance is in that group then it is allowed access!

EFS scales automatically, pay-per-use, don't need to provision capacity.



EFS configurations

You get 1000s of NFS clients each with 10 GB+ throughput, can grow to petabyte automatically.

Have performance mode that you can set:

- General purpose: latency-sensitive use cases for CMS (wordpress) or a web server
- Max I/O: Higher latency, throughput, highly parallel. This is good for big data and media processing

Have throughput mode that you can set:

- Bursting: 1 TB = 50MiB/s + burst of up to 100MiB/s. **Scale with the file system size.**
- Provisioned: Set your throughput regardless of storage size size, like 1 GiB/s for 1 TB storage
- Elastic: Automatically scale throughput up or down based on your workloads, good for unpredictable workloads

Have storage tiers:

- Standard: For frequently accessed files
- Infrequent access (EFS-IA): Cost money in order to retrieve files, lower price to store, you have to enable it using a lifecycle policy

You can set up lifecycle policy for files i.e. put the file into EFS-IA after N days.

Availability and durability:

- Standard: EFS can be set multi-az, great for production
- One zone: One AZ, good for development, backup enabled by default, compatible with IA (EFS One zone-IA) big discounts!

EBS vs EFS

- EBS can only attach to one instance at a time. The only exception is the io1/io2 volume type of EBS which can be attached by multiple EC2 using multi-attach, but is still limited to 16 EC2 instances
- EBS are locked to a specific availability zone. Need snapshot for replication across AZ
- Root EBS gets terminated if your EC2 gets terminated
- EFS allows you to be attached to 100s of EC2 instances even across AZ!
- Good for sharing website files, but is only for Linux EC2 instances
- Higher cost > EBS of course, it is more advance and is multi-az. But you can use EFS-IA for cost saving for those infrequently accessed files

EC2 instance store

There is another type of block storage type for an EC2 instance and that is called the EC2 instance store. EBS are network attached device meaning that the speed is limited by the networking speed, although it is still good. An alternative is the EC2 instance store which are physical hardware disk that's connected on the server.

EC2 instance store have better I/O performance compared to EBS instance.

But the caviar is that EC2 instance store is an ephemeral storage, meaning that the data on disk are lost once you stop/terminate the EC2 instance.

EC2 instance store is recommended for buffer / cache /scratch data / temporary content. It is not for LONG-TERM STORAGE!

Backup and replication are YOUR responsibility if you choose EC2 instance store as storage.

AMI

Stands for Amazon machine image. AMI are like a template of a computer's root drive, they contain the OS and the software all pre-packaged into an image.

When you launch an EC2 you will be launching from an AMI, which will just extract OS and the software that are pre-packaged from the AMI to initialize the root volume.

You can use a public AMI which AWS provides, or you can make your own AMI and maintain them yourself, or from AWS marketplace AMI which someone else created the AMI and is up for sale for you to use. **If you do build your own AMI it will be for a specific region, but you can copy**

it across regions.

Custom AMI creation process

To make your own AMI here is the process:

1. Start an EC2 instance and then customize it, i.e. download the specific software that you want your AMI to contain
2. Right click the EC2 instance after you have done customizing it, and under Image and templates -> Create image
3. To launch an EC2 with your own AMI you just have to select it from My AMI tab

Again AMI have all the pre-packaged software installed by default, you can go on your way without installing them yourself.

Revision #7

Created 2023-02-10 18:52:03 UTC by Tamarine

Updated 2023-07-27 01:55:35 UTC by Tamarine