



## AWS EC2 instance store vs. EBS

Tags: ACC

As students come to grips with AWS services a common question is whether 'Instance Store' storage still has a place in modern AWS solutions. This is a reasonable question given the complication of working with ephemeral storage.

### Instance Stores

Let's start with what Instance storage is. This is the disk that is physically attached to virtualization host. This is the closest (lowest latency) storage available to your instance (*other than RAM*). By comparison EBS storage is storage on a remote network connected SAN or NAS and may be competing for I/O with thousands of other instances.

For 90% of use cases the difference in latency will be irrelevant, and certainly not worth the complication introduced by the ephemeral nature of the Instance Store storage. The performance difference in latency, especially with SSD Instance Stores, can have a profound impact on compute intensive workloads where that CPU may be constantly waiting for data to be read from disk.

From time to time the ephemeral nature of Instance Stores is cited as a security advantage. While may be true and of some value, [IMO] this is a false sense of security and should not be relied on if you are serious about your data security. The wiping is done by a 'process' on the host. If the Host fails suddenly those disks will not be wiped until the host is repaired, or the disks are moved to another host, or destroyed. Until that happens the data will be intact on the disks. Therefore the security wiping is not 'guaranteed' and the data on those disks cannot be assumed to be any more secure than any other storage on that site.

### Ephemeral storage vs. EBS

Discussion about ephemeral storage vs EBS is often phrased as two extremes, with Instance Stores being at risk of disappearing at any moment vs. EBS which is rock solid and will never fail. While they are at either ends of the continuum, neither are extremes, they are in some cases closer than often considered.

It is true that Instance Stores will be wiped if the instance is stopped. However we have been happily using a technology with similar characteristics since the beginning of electronic computing. RAM is ephemeral storage and we use it without thinking about it. We know it has this limitation and we compensate by having processes that store important information on non-ephemeral media (tape, disk, EEPROM, etc) so that we can recall it when needed. Viewing Instance Stores in this way allows you to think of it in terms of a balance between the performance advantage and the support solutions needed to allow you to take advantage of it. Just as we do with RAM.

At the other end is the idea that EBS is so reliable that you don't need to worry about disk failure. EBS volumes are not indestructible. The presented volumes have a failure rate of ~0.2%. This is about 20 times better than a standard disk. However if your app is super critical and you cannot risk an impact, this effectively puts you in the same situation as with Instance stores. You will need to engineer a fault tolerant solution. The AWS design pattern is to

always **'Design for Failure'** i.e. assume it will fail and design to pick up the load elsewhere. For a high proportion of systems EBS is an adequate (and cost effective) solution. However if your application is very sensitive the solution to make EBS fault tolerant will be similar to making an Instance Store fault tolerant.

## Treatments

Depending on the nature of the data and use case there are a range of options to be considered.

- A RAID mirror can be a good method to manage a disk level failure, as it provides time to manage the outage to move to a new instance. However it alone is not a full solution as you will continue to run an a defective instance until it is fixed, and it does not protect against other non-disk type failures.
- If the requirement is only for fast Read, the data does not change frequently, and RTO / RPO are not onerous, Use S3 to host the master copy and basic snapshots or Lambda to pick up and write the incremental changes back to S3.
- If the data is changing, or the RPO & RTO are short, you may want to consider file level or block level replication to a standby Instance with Instance Store volumes. AWS do not have a services for this at present, but it can be done with 3rd party products.
- If the data is changing, or the RPO is short but RTO is not, you may want to consider file level or block level replication to an instance with EBS volume as a form of up-to-date / real-time snapshot to use to build a new instance. AWS do not have a services for this at present, but it can be done with 3rd party products.
- Possibly the design is no longer optimal and you should consider a High Performance Compute (HPC) solution which has additional safeguards and recovery methods built in.
- I am sure you can think of more when you start to think outside the box.

## Conclusion

Instance stores still have value especially when it comes to massive IOPS at low latency.

Instance Stores may be ephemeral, but EBS storage is not 100% reliable either. Decide the level of acceptable Risk and **'Design for Failure'** accordingly, regardless of the technology.

Until you get to the rarified atmosphere of high performance compute, EBS storage provides plenty of grunt and a whole bunch of flexibility to meet most of your EC2 needs.

---

If you need help, please email [Pluralsight Support \(opens email form\)](#) for 24/7 assistance.