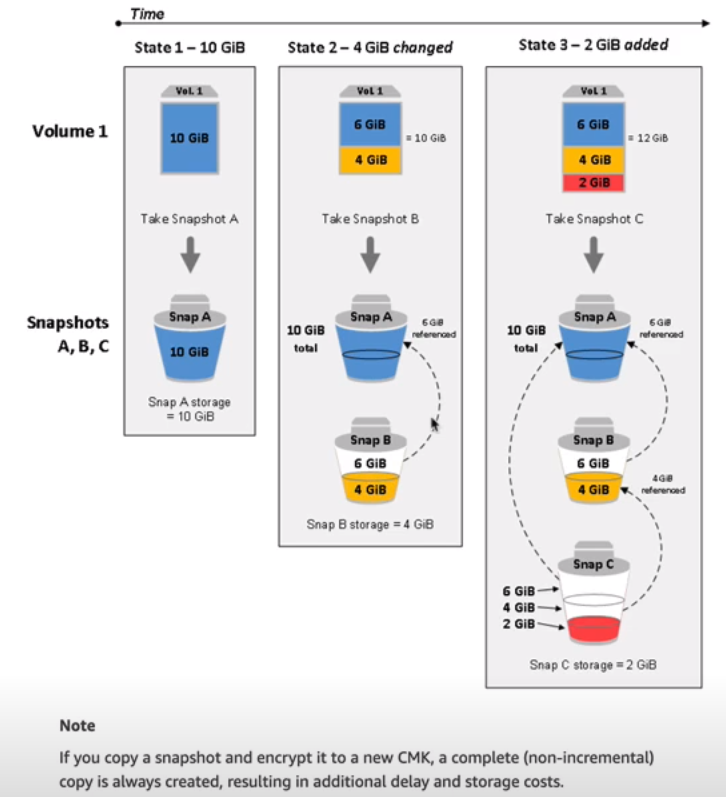


Okay. Welcome back in this lesson, I want to discuss EBS snapshots. **Snapshots are a point in time backup of an EBS volume. So when you first do a snapshot all of the data is copied from the EBS volume into S3 and it consumes the same amount of space as the volume and it takes time because it's copying all of that data from that volume. Now, if you do another snapshot after that, only the data which has changed since the last snapshot is stored and the same is true for all future snapshots of that volume.**



So architecturally, this is how it looks. This is taken from the AWS documentation. When we first take a snapshot, let's say it's a 10 GB volume then all of that 10 GB gets copied into the S3 bucket. If we take another snapshot when 4GB of that data's changed, then what happens is it refers to the first snapshot for any data that's not changed and just adds that additional 4GB of data. So every time you do a snapshot you're only storing the difference in data between the last snapshot and this one and you're **only charged for the data that's different between the last snapshot and this one. So snapshots offer a great way of implementing data backups of EBS volumes inside AWS.** Now performing a snapshot is actually really easy. You can take a volume and go to actions and then create snapshot.

**Now snapshots are known as crash consistent. What this means is that when you're taking the snapshot, the operating system and any applications that are inside it aren't aware that you're performing that snapshot and so the data that you're actually storing would be just like if you switched off a physical machine right now and the application is left with data that could potentially be in an inconsistent state.** Now to perform a truly consistent snapshot so you know that the data is in a good state to do that back up, you've got a number of choices.   
**The first option is that inside the guest operating system, you flush any in memory caches to disk so that just makes sure that any applications that are storing any data that hasn't been written to disk write that data to disk.   
The second option you've got is to shut down any instances that you're doing snapshots on** so I'll demonstrate this process. I'll right click, go to instance state, and then I'll stop the instance. Remember, this is stopping the virtual hardware of the instance. It's stopping the operating system, and in doing so, it **means the instance is shutting down and saving anything that's in memory to disk before it does so. Every application that's running in these instances will know that as the guest operating system shuts down, it should write out any data that it's got in in memory caching to disk** and so it will do that process. **Once that process is finished and the instance is in the stop state then we know that the data is consistent and we can do a snapshot. Now, it's always best practice whenever you're doing a snapshot of the root volume of an instance to make sure the instance is shut down before you do that.** So now that we're in a stop state I could go to EBS volumes, go to actions, create a snapshot, and then I'll want to give it a description so I'll just call it test snap for test snapshot. I can add key value tags if I wanted to. In this case, I won't. I'll just go ahead and click on Create Snapshot because this volume size is 8 GB. It means it's going to be copying 8 GB of data from EBS into S3. **Now because the data has been copied from the EBS source volume into S3, it's more resilient. Remember, the EBS volumes are stored in a particular availability zone in this case availability zone A. They do replicate between different storage systems in the availability zone, but they still depend on the reliability of availability zone A. If availability zone A in this example fails, then we won't be able to access the EBS source volume. Best case until the availability zone is restored. Worst case it could be a problem that's severe enough to actually lose the data that's stored on this volume. Now snapshots are taken from EBS and stored on S3.** We haven't covered S3 in detail yet, but **S3 replicates its data across multiple availability zones in that region. So straightaway, once we got the data out of EBS into S3, it can tolerate the failure of individual availability zones.**

Now, once the data is in S3 we can actually use the snapshot in some pretty creative ways. **The first thing we can do is to modify the permissions on the snapshot and share it with other AWS accounts. Now a snapshot has a number of different modes it can be in. By default it's private. You're able to add individual AWS accounts the permission to access this snapshot or alternatively, you can make it entirely public so that any AWS account can use it. You don't need to white list them. In addition to letting other accounts access a snapshot, you're able to use a snapshot that's the basis to create a new volume.** So if we look at our list of volumes, this is the original source volume that this snapshot was taken from and it's in availability zone 1b. What I'm able to do is right click on the snapshot, go to create volume, and create another volume with same data from the source volume but two store it in any availability zone I want **because the snapshot is stored in S3 you can use it as the basis to create another volume that's in another availability zone.** So this is creating the volume from the point of view of the snapshot but I'm also able to do the process by going to volume, creating a volume, specifying an availability zone, and then selecting the snapshot ID as basis. Now note that I'll be able to see all snapshots in this list that I've got access to and they're going to be lots of them. I can keep scrolling down almost indefinitely and there's going to be snapshots that are shared with my account, which contain blank data or specific operating systems or other type of test data. There's going to be hundreds, thousands, even hundreds of thousands of snapshots available in this list. It's far easier from an account perspective to go into snapshots, right click on it, and put create volume because that configures everything for you. So I could go ahead and create this volume in U.S. East 1a. What would happen at that point is a new volume would be created it we use the data in this snapshot to create the volume and we'd have a new identical volume in a different availability zone. **So it offers a great way of taking the contents of data that is scoped to a particular availability zone and creating it in another**.

**Now snapshots can also be copied between regions,** in the actions dropdown there is an item called copy and if you select that, you can copy the snapshot to a completely different region, maybe a European region. So EU Stockholm and once you copy that snapshot, you'll be able to create volumes in that region from that snapshot, just as you're able to create them in separate availability zones. So this might form part of a global disaster recovery plan where we're operating in a specific AWS region. Let's say, North Virginia, and we want to copy periodic snapshots of our key system components to an international region, in this case, EU Stockholm and by doing so, it means **you can spin up an entire backup infrastructure platform in a different region without much in the way of admin overhead.** It's a really powerful feature, and **snapshots are often used as part of a good solid DR plan**.

Now I won't be covering this in this lesson, and I'll be talking about this later in the course, but there is a product called **Lifecycle Manager or Data Lifecycle Manager**, **which allows you to automate the process of creating and deleting snapshots.** We'll be covering that in much more detail later on the course because it has started to feature on the exam but at this point in the course, just keep in mind that creating and deleting snapshots does not have to be a manual process. You are able to automate it using **Data Lifecycle Manager, also known as DLM.**

Now, ***in terms of billing for these snapshots, they are stored inside S3. So they do tend to be a lot cheaper than the equivalent EBS storage volume. You are only storing the data that is different between snapshots. So if I do 100 snapshots of this 8GB volume that I'm not going to be storing 100 times 8 GB If the data change on this volume is relatively low, and I'm only going to be storing the original 8 GB and then potentially tiny components inside each snapshot***. Now, if you've got any backup experience, you might hear the term incremental and immediately start getting nervous because **incremental relies on every backup in a chain working in orders to do a restore. With snapshots, if you do delete any of the incremental snapshots in the chain, it retains any of the data that's used in future versions of that snapshot. So you're never going to be in a position where you'll lose any data.** If I go to volumes, for example, and I select this volume that's in use. So this is the one that I've used previously I create another snapshot I'm just going to enter some random text for this one. I will create Snapshot because there's been very little data change this snapshot is going to be relatively quick to complete. There we go. It's completed already. I'll go back to this volume again and create another snapshot. Again random text create snapshot. So at this point I've got the three snapshots, the original one that I created at 8:12 a.m. The one following that at 8:27 and the one I'm just creating, which is 8:28 now with **traditional incremental backup systems, if I was to delete this snapshot, then it would cause this third one to be unusable because every single incremental backup relies on all of the previous ones to the last full successful backup but snapshots don't suffer from this limitation. If I was to delete this middle snapshot, so I'll do that. I'll go to delete, then snapshots is smart enough to retain any of the data that used for this third snapshot, so you will never lose data by pruning snapshots. Snapshots offer a really cost effective and efficient way to achieve backups in AWS**.

Now, if you recall at the very beginning of the course the fundamental section where I was talking about DR techniques and I talked about a **recovery time objective and a recovery point objective. For recovery point objective, I talked about how you could lower that value by performing more regular backups, essentially minimizing the potential for data loss. With snapshots because of the way that they work, you could, in theory, do a snapshot every single minute that the system is live and by doing that, because you're only storing the difference in data, it gives you a really great RPO without significant cost. So straightaway we can see how AWS is supporting your ability to have really cutting edge RTO and RPO values without incurring significant expense**. So that's a brilliant use case for snapshots. Just make sure that you're performing regular snapshots of your key business components, make sure that they're flushing their data to disk as often as possible, and you'll have really good recoverability in the event of a disaster and if you take those snapshots and copy them to another region, you'll also make sure that your system is capable of withstanding significant region wide disruption inside AWS. Now, that's everything I wanted to cover about snapshots. You don't really need to know anything else for the exam. Just be aware that snapshots are replicated across AZs in that region, and you can use snapshots either to create volumes in a different AZ or in a different region entirely and you can share these snapshots either publicly or with specific AWS accounts. I have not talked about encryption and how that affects snapshots yet. I've got a full lesson that's coming up in the EC2 advance topic, or are we talking about EBS volume encryption, which will also cover snapshots. Just be aware that when you do have an encrypted EBS volume that does impact the snapshots and I'll explain how in the EC2 advance topic. For now, that's everything I wanted to cover. So go ahead, mark this lesson as complete and when you ready, join me the next.