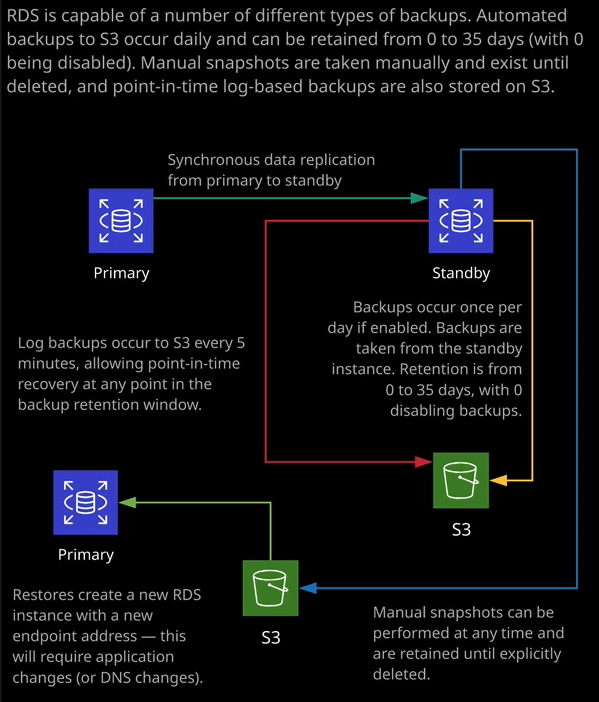
Welcome back and in this lesson, I want to talk about how RDS handles backups and restores. **RDS is a really capable product that has a full set of backup and restoration capabilities built in and because RDS is a database as a service product, one of the key benefits of the product is that it removes a lot of the manual overhead or manual labor that you need to do as an administrator. So rather than having to install a database product on an EC2 instance, manage storage, manage configuration, manage back up and restorations, resilience, read scaling, and all of the other features that a database administrator requires. With RDS, it's all handled for you by the product.** So in this lesson, I want to focus specifically on backup and restores. Now I'm following on immediately from the last lesson where I created the RDS lesson database instance.



**Now RDS has got a number of different ways that it handles backups. The first one of these is what's called a data base snapshot and this at its most simple is a manual snapshot that you perform, so to do it, you select the database instance, you can go to actions and you can take a snapshot** and I'm going to call the snapshot rdslesson-wordpress that'll just tell me that I've performed the WordPress installation. I'm going to go ahead and take snapshot. Now that's going to actually perform a snapshot based backup of this RDS instance. It will actually take the storage or freeze it momentarily, and it'll take a storage level snapshot of the database instance. **Now this will be available until I delete it. It's a manual snapshot. It has no automatic retention.** **Essentially, this snapshot will exist even after the database is deleted, and that's a really critical point to understand for the exam. If you need a backup of an RDS instance, that will remain in existence until you delete it then it needs to be a snapshot. Now you can perform these manually through the console, but you can also use the AWS command line tools, the APIs, or even a Lambda function to run these automatically but they are manual from the perspective that they do need to be initiated by something. So if you need long term backups, if you need backups that exist past the lifetime of the RDS instance you need to use snapshots**.

Now, RDS is also capable of doing **backups automatically**, so let's have a look at that. I'll go to databases and I'll go inside this database instance and then just click on "Maintenance and Backups." So, at the moment, backups are automatically configured. If I go to modify, let's have a look at this database instance and just look at how that looks inside the configuration and we'll scroll down and look for the backup section and note here we've got a backup retention period of seven days. Now backups that are performed automatically as part of this automatic backup process. They have a retention period. Now that retention period can be set from anywhere from zero days, which means that this automatic process is disabled all the way up to 35 days. Now it's important you understand this limit for the exam. 35 days is the maximum retention period for automatic backups, so keep this in mind and backups occur during the backup window that you specify. So it's once per day starting at this time, and you give the backup window duration and essentially, the backup happens once per day during this window. Now, just like with snapshots, you can use these backups to restore a new database, and I'll show you how that works later in this lesson. From a storage perspective, the first backup is a full backup of the data used by the RDS instance. So it's not the data that's allocated. It's not the entire space. So if I've allocated 100 gb for this RDS instance, but only use, say, 10 or 20 GB. The first backup would be 10 or 20 gb. It's the data consumed. Backups from that point onwards are incremental, so only the data that's changed and that's why it's important to understand that because RDS provides backup storage equal to the allocated storage, you often don't have to pay for your backups inside RDS because the first backup uses the data consumed, not the data allocated. **Backups after that are incremental, so only storing data change and so if you've got a large allocation, say one terabyte, but you're only using a small amount of data then generally, all of your backups can actually be stored within that free allocation, so that's important to understand for the exam**.

Now, remember, **these backups have a retention window, and so if you delete the RDS instance, these backups won't exist from a long-term perspective after you delete the RDS instance. So it's especially important that for long term backups you use snapshots. These automatic backups are not designed to be long term. They're designed to be backups with the retention window up to 35 days that are automatically managed by the system.** So keep that in mind; it is really important.

Now, in addition to this snapshot, so we can see now on my screen, I've got one manual snapshot and one automated snapshot. We also have the ability to **restore from a point in time**. So in addition to performing these automatic backups, **RDS is also doing backups of transactional information that's occurring constantly and also backing that up to the same location. What that allows is to perform a point in time recovery of the database so I can click on "Databases." I could go to actions and restore to a point in time and these log backups are occurring every five minutes and so generally I can click on "Custom" and select a backup restoration time of my choosing and automatically restore the database to the state that it was in at that point in time.** So that's a really powerful feature. So you've got all of these different mechanisms. You've got snapshots, automatic backups, and then the point in time log backups and they all occur at the same time as part of the ability to restore an RDS instance. Now, what I want to talk about next is how that restore actually happens. So I'm going to go to my snapshots because I took a snapshot when I'd already installed WordPress and that's what I want to use to do a restore to make another RDS instance. Now you might have picked up that I said that very specifically. The reason for that is if I select any RDS back up and go to actions and I restore the backup. So let's say that I'm restoring the snapshot. What it is actually going to do is to create a brand new database instance. It does not restore over the existing database. So in this case, I'm going to have to change this to db.t3.medium. Somebody changed the db instance class that's used. I'm going to set it not to be a multi AZ deployment. I'll be needing to specify a different database identifier. So I'll call this RDS lesson restore. I'll have to set all of the networking and security options, so I'll put it in VPC One. I'll use the same subnet group. I'll set it to not be publicly accessible. Generally I have no preference on the availability zone but I know that this first database is deployed in 1a. So let's say that I'm simulating a failure. I'm going to deploy this in 1b. I'll leave everything else the same. The same port, the same parameter and options group. I'll scroll down and I'll go and restore the database instance, and it's going to take that snapshot and create a **brand new database instance, and it will have its own cname.** So this is important. **It will not be accessible using the same DNS name as the original**. The **restore** I that started off moments ago was from a snapshot but if I select this primary active database instance and go to actions and restore to point in time, I'm still prompted to provide a new database identifier. **So RDS is not capable of restoring over the top of an existing instance and why that matters is that no matter what method you're doing for the restore, you will get a brand new cname for the database and so you will need to perform some level of application reconfiguration.**

Remember, with this WordPress instance, if I edit the configuration file again, so sudo nano WordPress or wp-config.php. If I scroll down, as well as the database name, database user, database password I needed to specify the database host and this is specific to that initial database that I created. Anything that I restore will have a different cname, a different database host and so there will always be an element of application reconfiguration whenever you do a database restore.

**Now, one other benefit of using manual snapshots is that these snapshots can be copied over between different AWS regions.** So if I select this manual snapshot that I did which is this RDS lesson with WordPress, I can go to actions and then copy snapshot and I can copy this snapshot either in the same region, but I could also specify a different region. So once I've copied that snapshot to that different region. I **could then spin up in RDS instance in that new region, so it offers potentially a way to have some multi region fail over for your RDS instances.** So if you have an automated process that takes a database snapshot and copies that snapshot to a different AWS region, then you've got the ability always to spin up a brand new RDS instance in a different region as part of a disaster recovery plan. Now, **if you're copying snapshots between regions and that snapshot is of an encrypted database instance that's natural itself will also be encrypted. Now because KMS is a regional service, if you're copying snapshots between regions, then you'll need to specify a different key.** So let's say that I was copying this to a different AWS region I'd need to specify a different master key to use as part of that copy. So that is important to keep in mind. Snapshots that have taken of encrypted data bases will themselves be encrypted. You need to pay a little bit more care because you need to make sure that you've got a key in that new region that you can use to re-encrypt that snapshot. It's handled as part of the process, so you don't have to explicitly re-encrypt the snapshot. But you do need to be aware that if you are using encrypted snapshots, there are extra elements at play. You need to make sure that KMS has got appropriate master keys in all of the regions that you're copying these snapshots to, and you don't need to limit it to one. You could copy a single snapshot to multiple regions if you had really extensive resiliency and DR requirements. So keep that in mind for the exam is potentially a little bit beyond what you will need to be aware of but if you do need to deploy RDS in production, knowing that you've got this multi region snapshot capability does offer a number of advantages. Okay, so at this point, what's actually happened is the RDS lesson restore database instance that's finished creating from that snapshot and the first thing that occurs is once the provisioning of that database instance finishes it itself is going to perform a snapshot back up. So if you go to snapshots, we'll be out to see that we've got this original RDS lesson hyphen with WordPress snapshots. So that's one that I created and we've also got two additional ones. We've got the RDS lesson original automatic backup, which is this one on the bottom and we've also got this new snapshot that was created from this new database so you can see the database instance or cluster here. This refers to the new database instance, the restore database, and so that has now its own snapshot. So that's part of the standard process of provisioning an RDS instance, whether it's a brand new one or a restoration from backup, it always configures its own snapshot when it's first created. Now I mentioned earlier that it will have its own cname so if I go to database instances and then RDS lesson restore. It will have its own interface endpoint DNS name. So if I copy that into my clipboard, move back over to the terminal and replace the existing one that I've got. So I'm going to go to the end of this line. I'm going to delete the DNS host name that exists for our original database instance and I'm going to paste in the restored database. So I'll save that file, move back to my console, go back to my cat pictures blog and refresh that is going to fail and that's because I wanted to illustrate another issue with restoring RDS instances. So if I go back to this newly restored RDS instance, look, it's not using the same security group. So just because you do a restore of a snapshot, it doesn't mean it keeps all of the configuration. You need to pay special attention to that. Now, to fix that, I need to modify this database instance, the newly restored database instance, so I'm going to scroll down and in networking and security, I'm going to look for security group and I'm going to change it to the RDS security group. Remember, this is the same security group that I used to create the initial database instance. So I'm going to change it to that and click on "Continue," I'll be informed that any **modifications can either be done during the next scheduled maintenance window or run immediately. Now, normally, if you're doing this in production, you'll want to wait for the maintenance window because changes can cause outages.** For this demonstration, though, I'm going to apply immediately and modify the database instance. Now these changes usually occur relatively quickly, especially changing the security group. We can see straightaway it's already done. So if I go back to the tab that has my cat pictures blog and just do a refresh, I should see relatively quickly once this has changed that it now opens the cat pictures blog. So this is running off our restored database. So keep that in mind for the exam. The important thing to understand is that **RDS supports snapshot based backups. It supports automatic backups ranging from 1 to 35 retention days, with zero being disabled. It supports point in time recovery, which is done using many backups that occur every five minutes. So that's transaction information that's stored and that allows you to restore a backup to a specific point in time but no matter what the method that you do, the DNS name of the new database is going to be different than the existing one and so you will need to modify application connection information. And in the case of** WordPress, that's done by changing this configuration file. And that's pretty much everything that I wanted to cover in this lesson. I just wanted to give you an overview of the backup and restore capabilities of RDS. Now, normally, if you are managing your own database instance, managing backups and restorations is normally a really intensive part of that process. So this is another reason to utilize RDS. There aren't many reasons why you'd want to use your own self managed database server and especially managing backups and recoveries can be extremely admin intensive. So it's another great reason to use RDS and I want you to be fully aware of the different options for the exam.