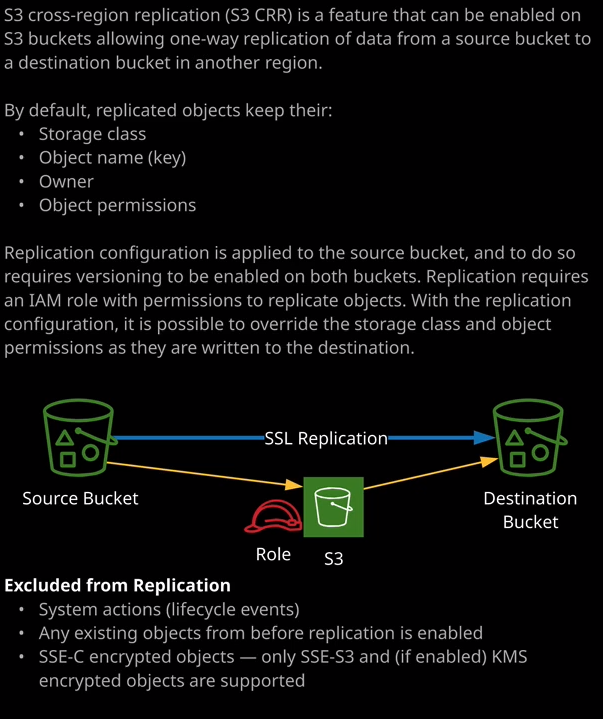
**S3 cross region replication also known as S3 CRR.**



Now, cross region replication is a feature that could be enabled on all S3 buckets, and **it allows for a one way replication of data from a source bucket to a destination bucket in a different region.** So that's the important thing to understand both for the exam and production usage. Both these buckets need to be in different regions. Now, I've already got a bucket created in US East 1, which is called ac-catpics1337. Now, to demonstrate this process, I'm going to create a new bucket. I'm going to create bucket. I'm going to call it catpics-backup1337 So, catpics-backup1337 and I'm not going to select the U.S. East 1 region because that's where my source bucket is located. For this case, I'm going to click on the dropdown. I'm going to select Asia Pacific Sydney, which is my home AWS region. So I'll select that and I'll accept the rest of the defaults. I'll click on "Next" and then go ahead and create the bucket. Now S3 cross region replication allows for the replication of objects and by default, these objects keep the same storage class, the same object to name, also known as the key, the same owner, and the same object permissions. Now cross region replication requires that versioning is enabled on both the source and the destination bucket. I've already got versioning enabled on the ac-catpics1337 bucket because that's something that I configured earlier in a previous lesson. So I'm going to go ahead and just confirm that. So I want to go inside this bucket go in to properties click on "Versioning" and just make sure that it's fully enabled and it is. I need to follow the same process on catpics-backup1337 which is in Sydney. So I'm going to click on "Properties." Go to versioning, enable versioning, and click "Save." Now at that point I've got versioning enabled on both buckets so I just need to configure the replications so the replication configuration between the source and the destination bucket. So I'm going to go ahead and do that. So I'll click on ac-catpics1337 which is the source bucket. I'll go to management, go to replication, and I'm going to go ahead and add a replication rule. **Now, as with most automated configurations inside S3, you're able to set a source either to be the entire bucket or filter it based on prefixes or tags.** Using prefixes or tags is beyond what you need to know for the associate level and it makes for a more complicated demo. So I'm going to leave it set to use the entire ac-catpics1337 bucket. I'll talk more about the replication criteria later in this lesson because there are some additional options that you can pick. But for now, I'm just going to leave it as default and set to replicate the entire bucket and then I'm going to click on "Next." **Now you're able to transition a storage class as part of this replication configuration.** If you recall in the previous lessons, I talked about how there were different storage classes available inside S3 and specifically I mentioned how you can use the one zone infrequent access to get significant cost reductions as long as you're willing to take the additional risk of data loss. Now, **one of the use cases for one zone IA is to use it for the destination configuration of cross region replication, because by definition you've already got the objects in the source bucket. You can replicate those to the destination bucket and override the storage class that's used. So because the destination bucket is the secondary store of data in certain situations, you're okay using one zone IA.** Now, in this particular case, I'm not going to select the option to change this storage class. That's where this is selected. You can pick the destination storage class. I'm not going to do that because I want to keep the demonstration nice and simple. You're also able to change the object ownership to the destination bucket owner. Now this is important because while I'm demonstrating this from two buckets in the same account, you are able to replicate between buckets in different AWS accounts, **so you can also change the ownership of the objects in the destination bucket to be the same as the destination bucket owner.** I don't need to do either of these things. I'm going to go ahead, select the destination bucket. The destination bucket is inside my account so I'm going to go ahead and pick the catpics-backup1337 bucket. So I'll select that click "Next." Now, the way that the permissions work as part of cross region replication is **you need to allocate a role and it's that role that will be used to take the object from the source bucket and to deposit them in the destination bucket. So you need to make sure you've got a role that has the required permissions.** Now you are able to create one that's part of doing this process, and that's what I'm going to do. I'm going to select create new role and I'm going to call the role replication role to keep it simple. So I've got the name. I've got it set to create a new role. I'm going to set the status to enabled, click on "Next" and then save the replication configuration. So that's this part set up. Now if you are following along this in your own environment, and you do get any sort of error about the replication rule of the replication configuration not being found. Then you can just go ahead and click on refresh a couple of times to remove that and if it still doesn't work, then just click on one of these other tabs and then click back to replication. It's sometimes just the user interface takes a couple of seconds to catch up to this replication rule, but at this point, I do have the rule in place, and it's working as intended. So let's have a look at how that looks. I'm going to click on the overview tab for this bucket and note how we have the Winky.jpg object inside this source bucket. So if you go to the S3 console and then I'll look in the backups bucket, notice how I don't have any objects in this bucket and it's important that you understand why this is. **A replication configuration is not retroactive. So when you enable a replication rule between a source and destination bucket, you're essentially replicating any objects added to that bucket from this point onward.** So if I go back to the source bucket, I'll click on "Upload," go to add files, and I pick a different object to upload maybe one of my other cats. So truffs.jpg. I select next, next, and then upload this object. This is a new object that's been added since we enabled the cross region replication and so once this is uploaded successfully, we can then go back to the S3 console and go to catpics-backup1337 after a couple of seconds we'll see this object replicated to the destination bucket. So that's critical to understand for the exam objects that are in a bucket before you add the replication configuration will not be replicated, so that's really important to understand. What's also important to understand is this **replication is only one way**. So if I click on uploads in the destination bucket and add another object, perhaps penny.jpg and I accept all of the defaults then while the object will be added to the destination bucket it if I go back to the S3 console, back to the source bucket, that one won't be replicated back. So the replication configuration is one way only from the source bucket to the destination bucket. Another important thing to understand about **replication is that system actions are not replicated. So if I've uploaded any objects, if I modify objects by uploading new objects, all of that is replicated from the source bucket to the destination bucket. But if I got system actions such as lifecycle events, so lifecycle rules, those are not replicated to the destination bucket**. So excluded from replication are system actions so lifecycle events. I mentioned also be that any existing objects are not replicated, so anything that existed in the bucket before their application configuration is enabled are not replicated. Now one last thing that I want you to be aware of and this is especially important for the exam is that **cross region replication does not support the replication of objects encrypted using SSE-C so that's the type of encryption where S3 handle the encryption and decryption, but it's done using customer managed keys that is not supported by cross region replication. By default, it does support objects that are encrypted using SSE-S3 so that's encryption where S3 is managing the encryption keys. Cross region replication by default does not support KMS encrypted objects, but you can enable it so that it does support them.** The way that you do that is going to management on the source bucket, go to replication, select the replication rule, and click on "Edit" and then enable this option so replicate objects encrypted with AWS KMS. Now the reason that you need to select this is that these objects will be reencrypted as part of that replication. So you need to specify to S3 cross region replication, the source key or keys that are used because these will be used to decrypt the object and then because KMS is a regional service and because keys are isolated in a particular region, you need to specify the KMS key that will be used on the destination side of this replication relationship. So any objects in the source bucket they'll be decrypted as part of this process and reencrypted before they're stored in the destination bucket and that's really all there is to it. You configure this and this is a relationship between the source and the destination bucket. It's not retroactive. The only things that are replicated are things that are added or updated since you enabled application configuration and this configuration is only one way but you can change the storage class as well as the permissions on any replicated objects.

Now for the exam, I want you to pay special attention to the why and the when you would use cross region replication. **So it might be that you're implanting a product with a global architecture, and you want to provide either resilience or performance replicas of any objects uploaded to S3. It might also be that you're purely using it for backup purposes, and that's okay. Any objects or any data that's uploaded to a source bucket could be replicated to a backup destination bucket and you can also change the storage class to be something less expensive.**