**SET UP AN AWS S3 CROSS-REGION REPLICATION**

* Cross-region replication is a bucket-level feature that enables automatic, asynchronous copying of objects across buckets in different AWS regions.
* S3 CRR can be configured from a single source S3 bucket to replicate objects into one or more destination buckets in another AWS region.
* S3 can replicate all or a subset of objects with specific key name prefixes.
* Object replicas in the destination bucket are exact replicas of the objects in the source bucket with the same key names and the same metadata.

**Requirements**

* **Create two buckets:** Create two buckets within AWS Management Console, where one bucket is a source bucket, and the other is a destination bucket. Source and destination buckets must be in different AWS regions.
* The source bucket owner must have permission to replicate objects in the destination bucket. Objects can be replicated from a source bucket to only one destination bucket.
* **Enable versioning:** Cross Region Replication can be implemented only when the versioning of both buckets is enabled.
* **Amazon S3 encrypts the data in transit across AWS regions using SSL:** It also provides security when data traverse across different regions.
* **Already uploaded objects will not be replicated:** If any kind of data already exists in the bucket, then that data will not be replicated when you perform the cross-region replication.
* If the source bucket owner also owns the object, the bucket owner has full permission to replicate the object. If not, the source bucket owner must have permission for the S3 actions s3: GetObjectVersion and s3:GetObjectVersionACL to read the object and object ACL

**Setting up an S3 Bucket with Cross-Region Replication:**

Step 1: Search and open S3 from the AWS management console.

Graphical user interface, application

Description automatically generated

Step 2: Click on the S3 console and it should take you to the S3 storage homepage.

Graphical user interface, text, application, website

Description automatically generated

Step 3: Create a bucket with your proffered name and chose a region of your choice. Here my bucket name of choice is [my-aws-test-bucket1](https://s3.console.aws.amazon.com/s3/buckets/my-aws-test-bucket1?region=us-east-1) with a us-east-1 region.

Graphical user interface, text, application

Description automatically generated

Step 4: Make sure that Bucket Versioning is enabled. If you want to add the tag for track storage cost, click on Add Tag add your preferred naming convention, and if you want to enable the encryption for a new object stored in the bucket click on enable. I left mine in the default setting.

Graphical user interface, text, application, Teams

Description automatically generated

Graphical user interface, application

Description automatically generated

Step 5: Repeat the same step for the second bucket, changing the name convention and add a different region.

Graphical user interface, application

Description automatically generated

Step 6: Now we have our two buckets. Next is to create an IAM role to replicate our data from [my-aws-test-bucket1](https://s3.console.aws.amazon.com/s3/buckets/my-aws-test-bucket1?region=us-east-1) to [my-aws-test-bucket2](https://s3.console.aws.amazon.com/s3/buckets/my-aws-test-bucket2?region=us-east-1). To create the role, navigate to the IAM tab

Graphical user interface, application

Description automatically generated

Step 7: On the IAM Dashboard, click on role and select Create role, select S3 from the drop-down menu, click on next, click on the Create policy tab, click on the JSON tab, and add your IAM policy in a JSON format. Please refer to my GitHub page listed below for my complete IAM policy. Click on next after your policy is added. Tags are optional and I won’t be adding any. Click on next and give your policy a naming convention. Then click on Create policy. The policy should give our S3 bucket the permission it needs to replicate our data.

<https://github.com/Jawonlaya/Personal-Portfolio/blob/6af6741fa3e6195fafc4fc726b485faa0411ace3/Cloud_Engineering/AWS/S3_Cross_Region_Replication/my-replication.json>

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

Graphical user interface, application

Description automatically generated

Step 8: Navigate back to our created S3 bucket. Click on the first bucket (my-aws-test-bucket1). Navigate to management and click on replication, then click on Create replication rule, give your create replication rule a naming convention, navigate to the destination tab and select the second bucket (my-aws-test-bucket2). Under the IAM role, attach the created S3 policy and click safe. You should see a successfully created Job ID

Graphical user interface, text

Description automatically generated

Graphical user interface, text, application, website

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

Step 9: Next is to test if our replication bucket is working. Navigate to the first S3 bucket and click on it. Click on the upload tab and upload any file of your choice. I will be using 2 image files for this project. You should see a successful Job ID. Now navigate to your second bucket and the same files should be replicated in your second S3 bucket.

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Conclusion: If you face any problem regarding any solution, please feel free to reach out to me and I will be glad to point you in the right direction. And please consider telling me what I could have done better in the write-up.

-Austine