Cross Region Replication

* Cross Region Replication is a feature that replicates the data from one bucket to another bucket which could be in a different region.
* It provides asynchronous copying of objects across buckets. Suppose X is a source bucket and Y is a destination bucket. If X wants to copy its objects to Y bucket, then the objects are not copied immediately.

**Some points to be remembered for Cross Region Replication**

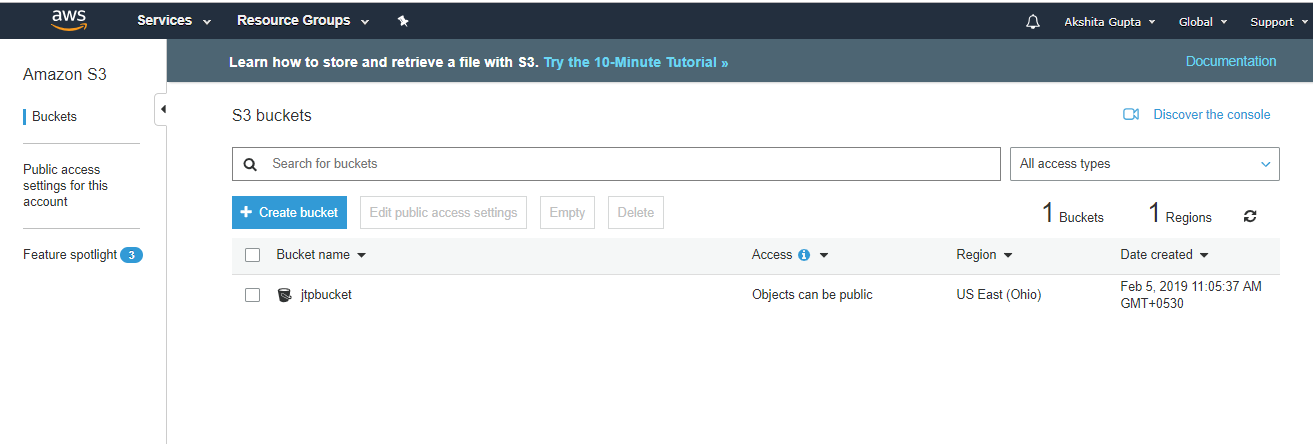
* **Create two buckets:** Create two buckets within AWS Management Console, where one bucket is a source bucket, and other is a destination bucket.
* **Enable versioning:** Cross Region Replication can be implemented only when the versioning of both the buckets is enabled.
* **Amazon S3 encrypts the data in transit across AWS regions using SSL:** It also provides security when data traverse across the 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.

**Use cases of Cross Region Replication**

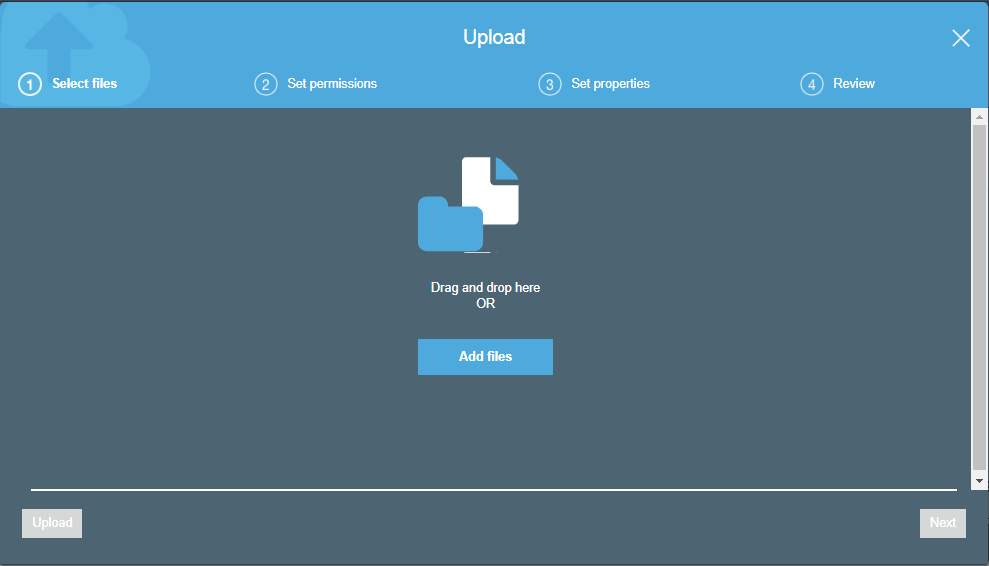
* **Compliance Requirements**  
  By default, Amazon S3 stores the data across different geographical regions or availability zone to have the availability of data. Sometimes there could be compliance requirements that you want to store the data in some specific region. Cross Region Replication allows you to replicate the data at some specific region to satisfy the requirements.
* **Minimize Latency**  
  Suppose your customers are in two geographical regions. To minimize latency, you need to maintain the copies of data in AWS region that are geographically closer to your users.
* **Maintain object copies under different ownership:** Regardless of who owns the source bucket, you can tell to Amazon S3 to change the ownership to AWS account user that owns the destination bucket. This is referred to as an owner override option.

**Let's understand the concept of Cross Region Replication through an example.**

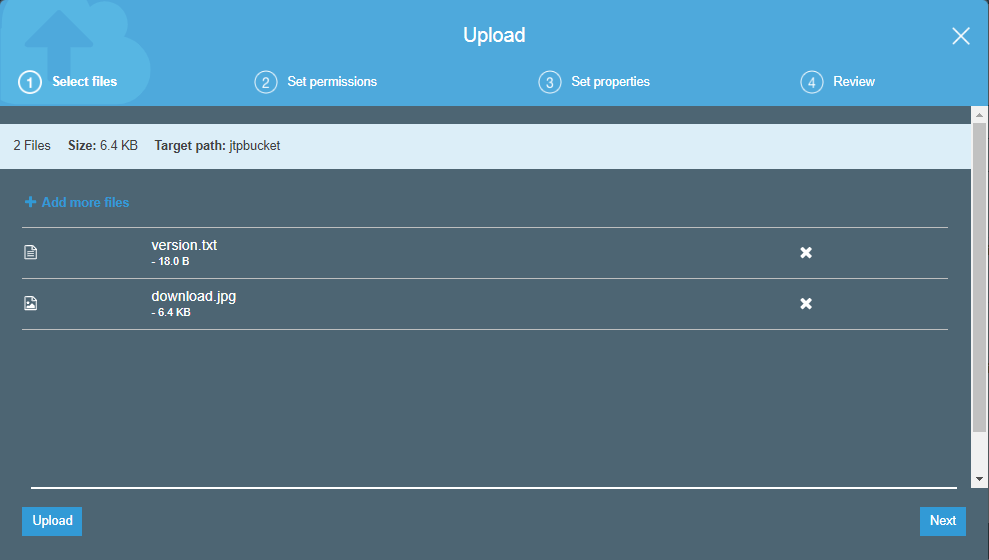
* Sign in to the AWS Management Console.
* Now, we upload the files in a **jtpbucket**. The jtpbucket is an s3 bucket created by us.



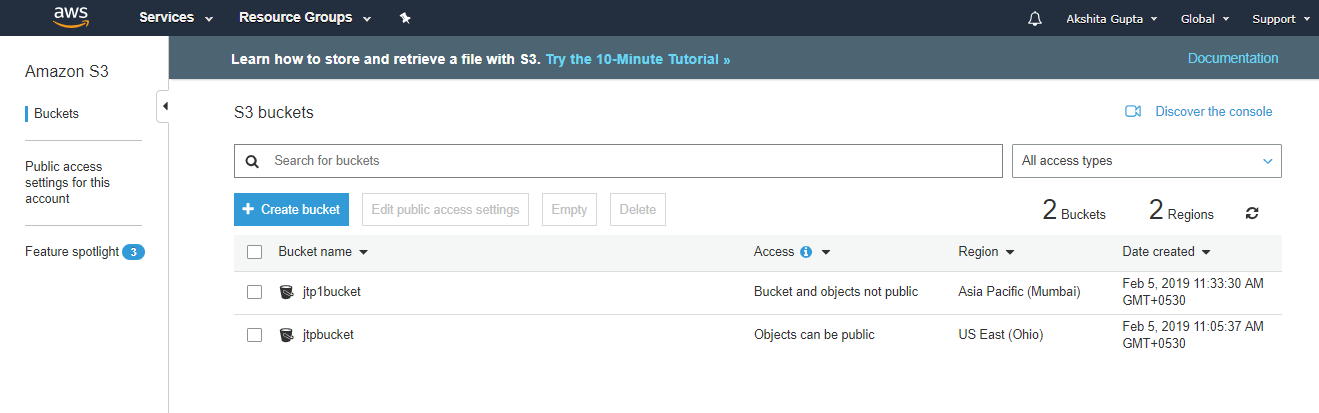
* Add the files in a bucket.



* Now, we add two files in a bucket, i.e., version.txt and download.jpg.

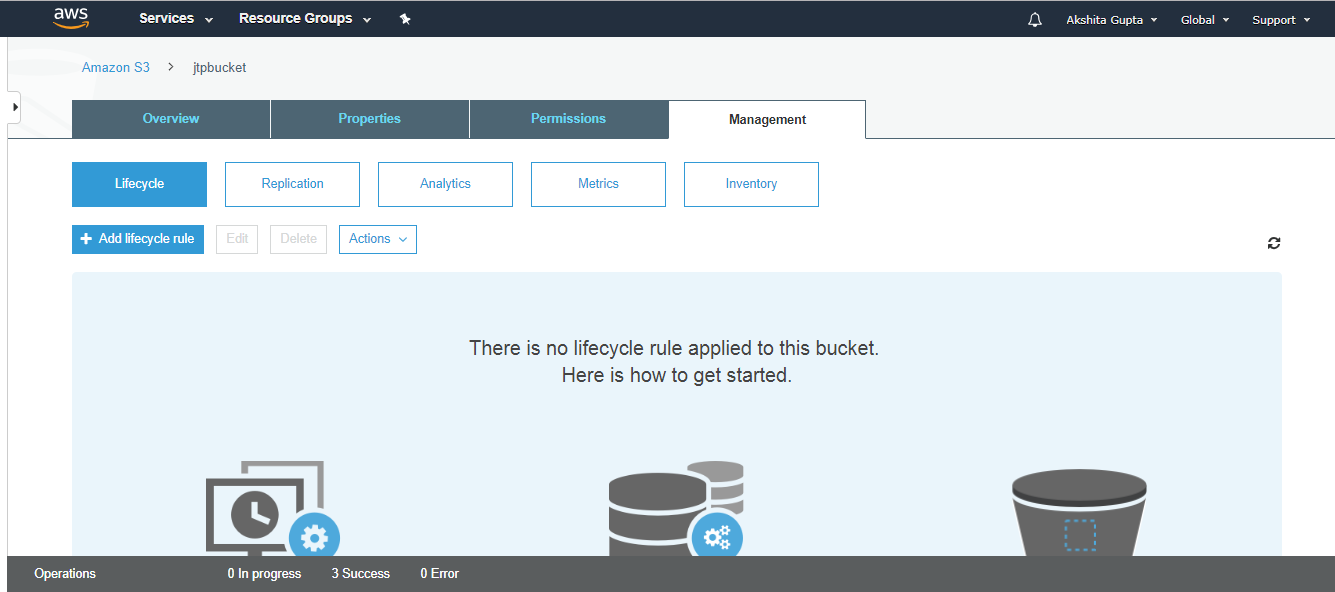


* Now, we create a new bucket whose name is **jtp1bucket** with a different region.

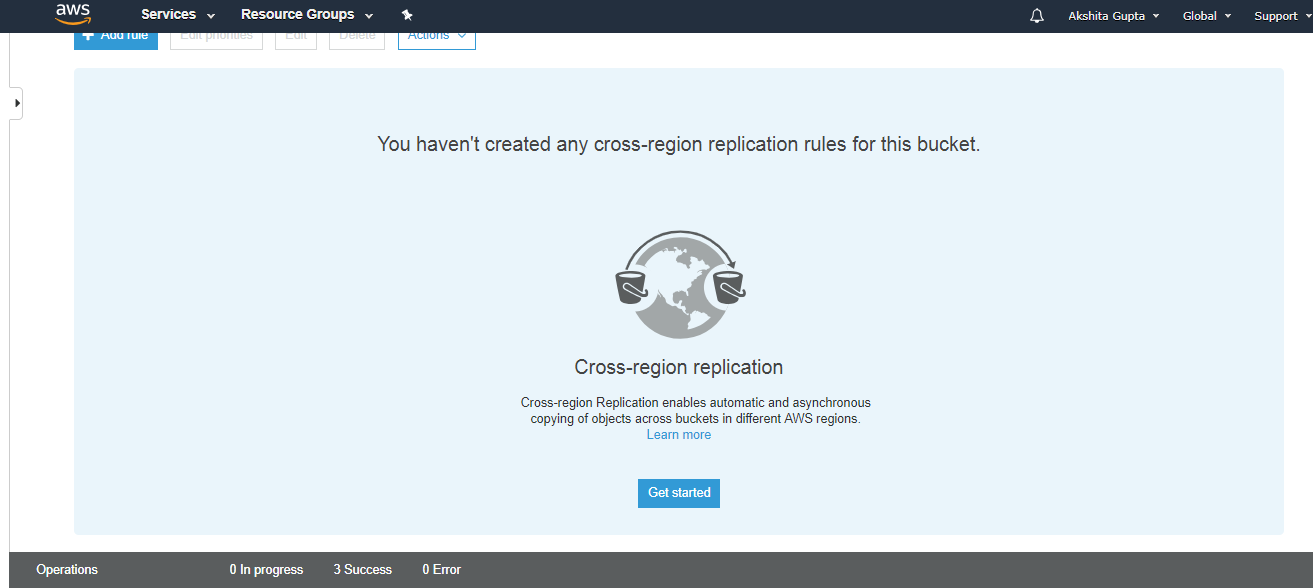


Now, we have two buckets, i.e., jtpbucket and jtp1bucket in s3.

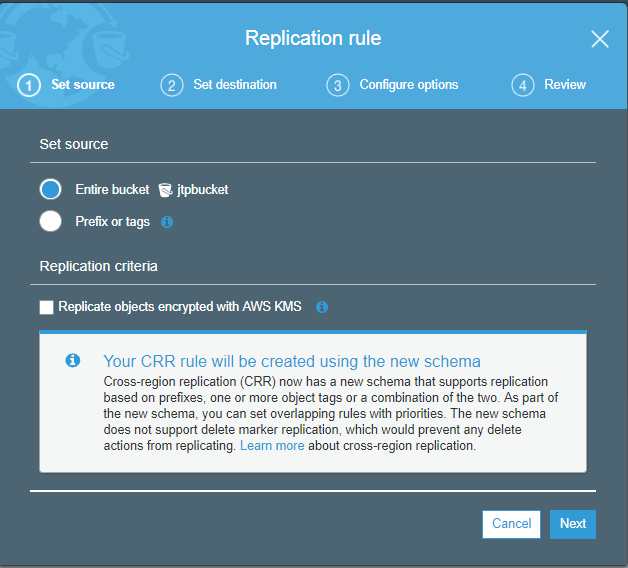
* Click on the **jtpbucket** and then move to the Management of the **jtpbucket**.



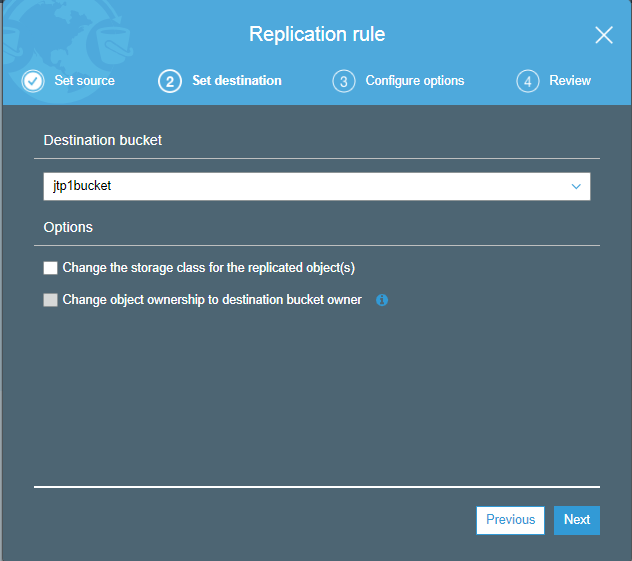
* Click on the **Replication**. On clicking, the screen appears as shown below:



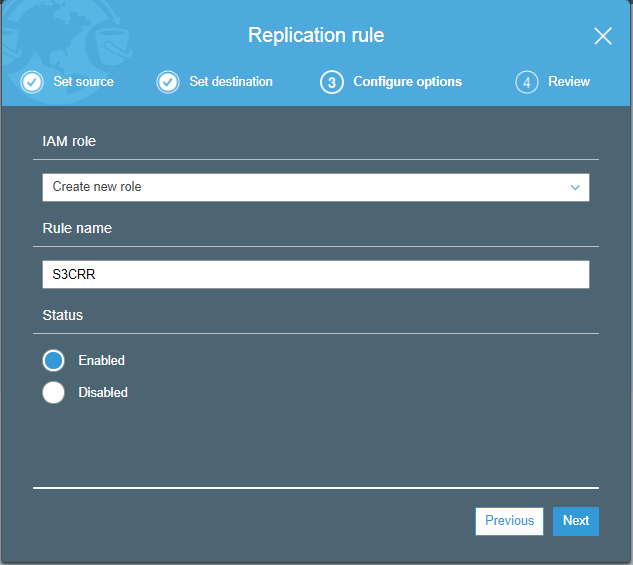
* Click on the **Get started** button.
* Enable the versioning of both the buckets.
* You can either replicate the entire bucket or tags to the destination bucket. Suppose you want to replicate the entire bucket and then click on the Next.



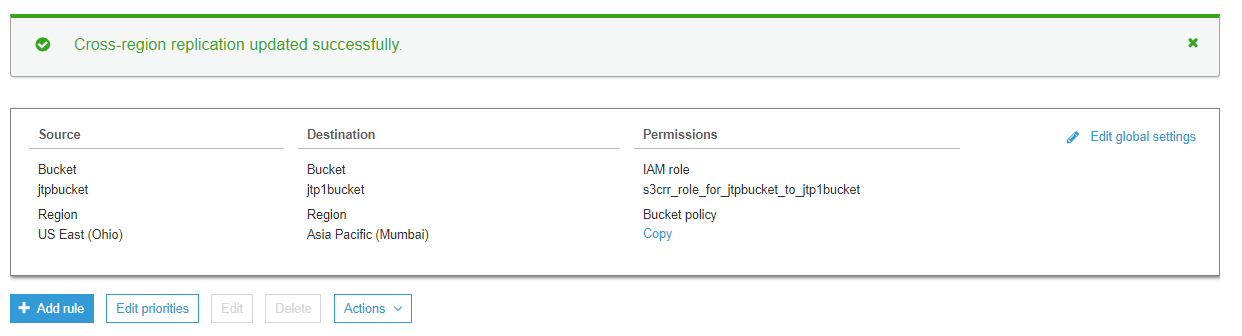
* Enter your destination bucket, i.e., jtp1bucket.



* Create a new IAM role, and the role name is S3CRR and then click on the Next.

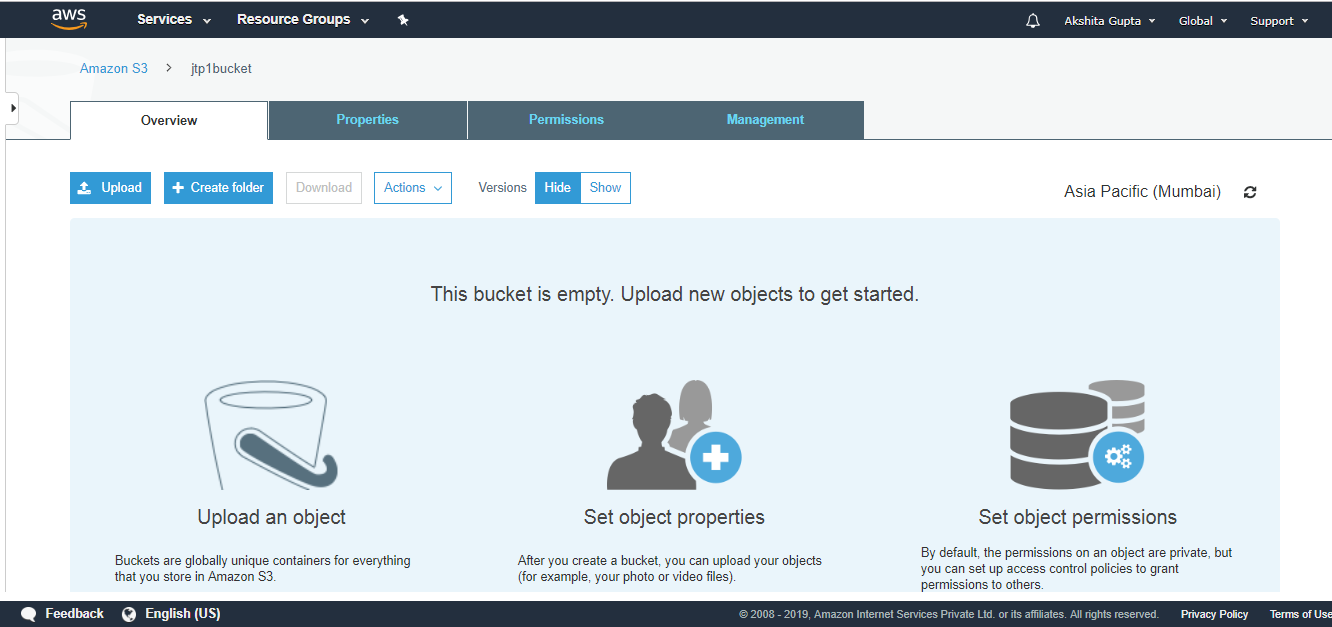


* After saving the settings, the screen appears as shown below:



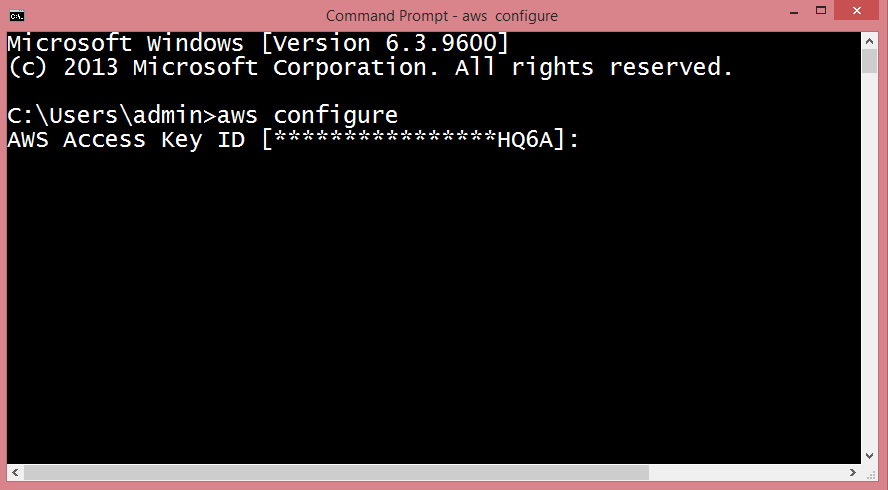
The above screen shows that the Cross region replication has been updated successfully. We can also see the source bucket and destination with their associated permissions.

* Now, we will see whether the files have been replicated from jtpbucket to the jtp1bucket. Click on the **jtp1bucket**.

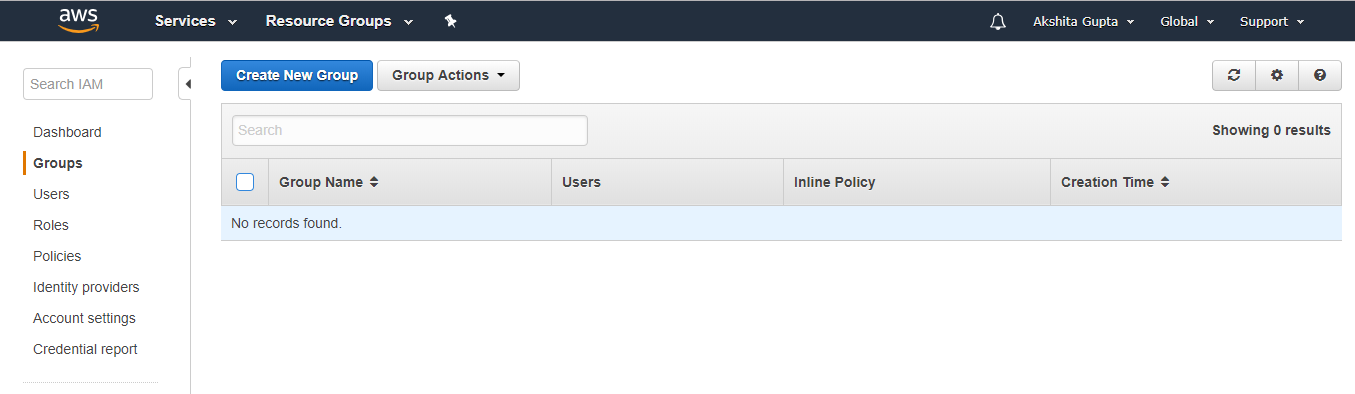


The above screen shows that the bucket is empty. Therefore, we can say that the objects do not replicate from one bucket to another bucket automatically, we can replicate only by using AWS CLI (Command Line Interface). To use the AWS CLI, you need to install the CLI tool.

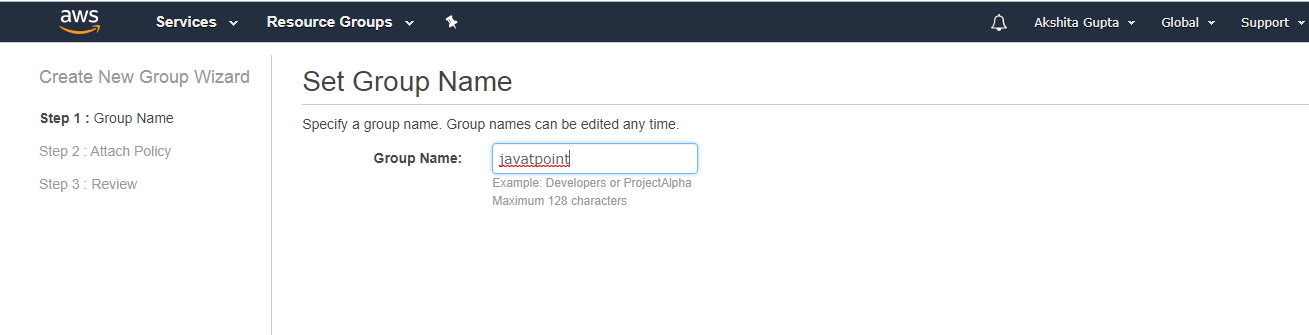
* After installation, open the cmd and type **aws configure**.



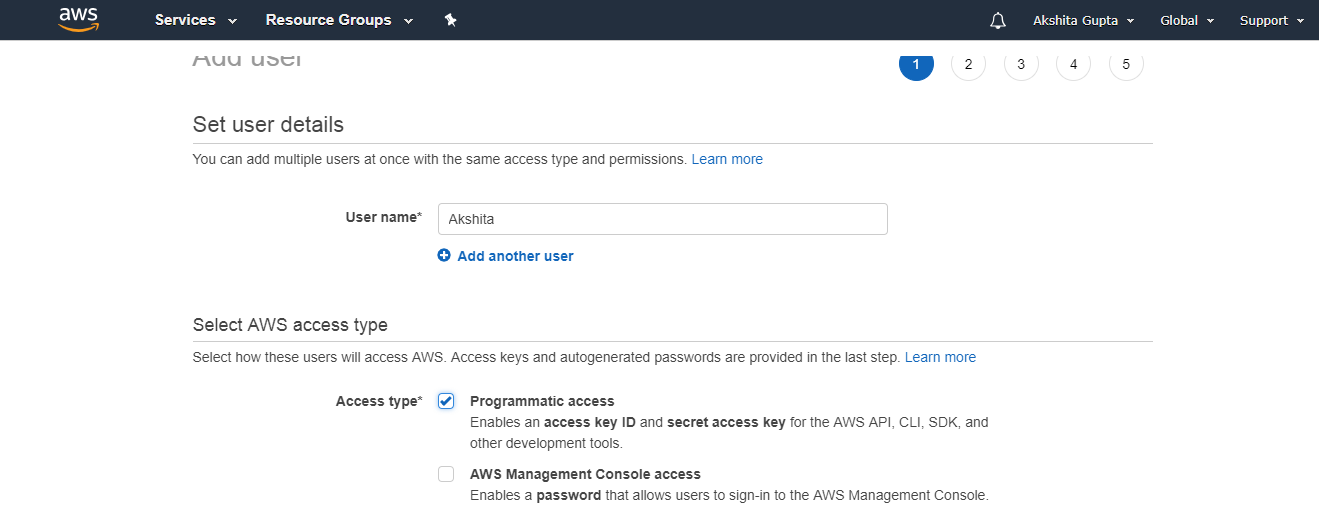
* Now, we need to generate the Access Key ID which is a user name and secret access key which is a password. To achieve this, we first need to create an IAM Group.



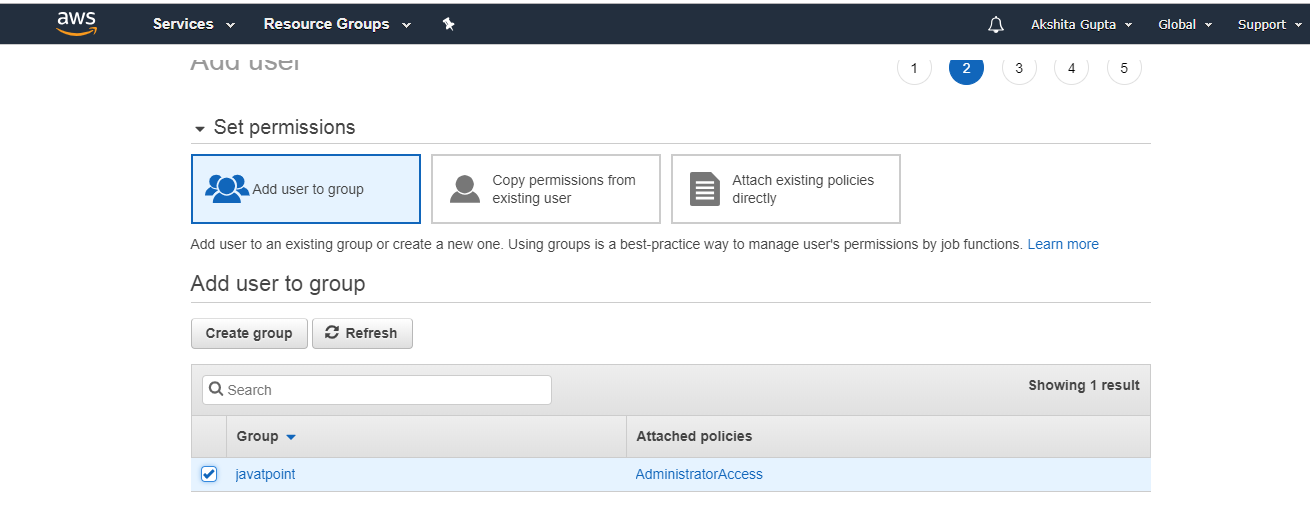
* Set the Group Name, i.e., javatpoint.



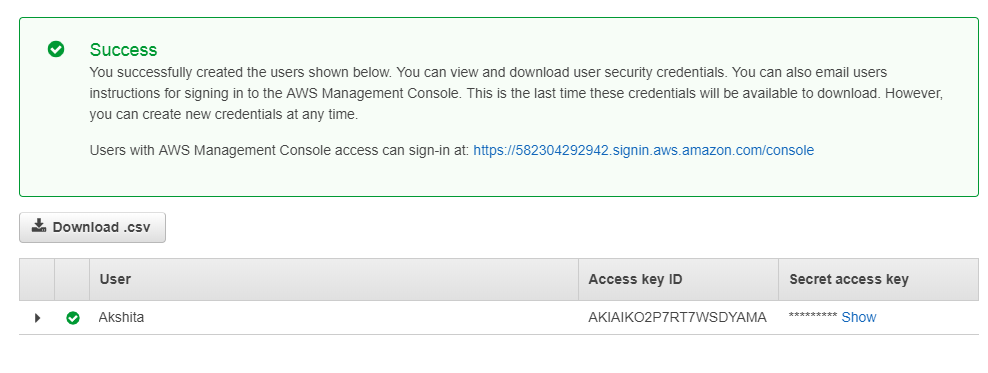
* Check the **AdministratorAccess** policy to access the AWS console through AWS CLI.
* Now, create an IAM User.
* Add the user name with programmatic access.



* Add the user to a group, i.e., javatpoint.

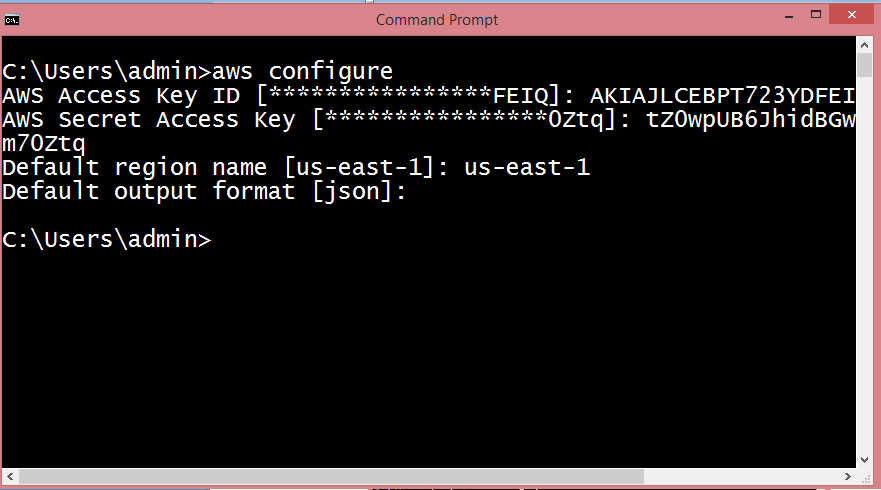


* Finally, the user is created.

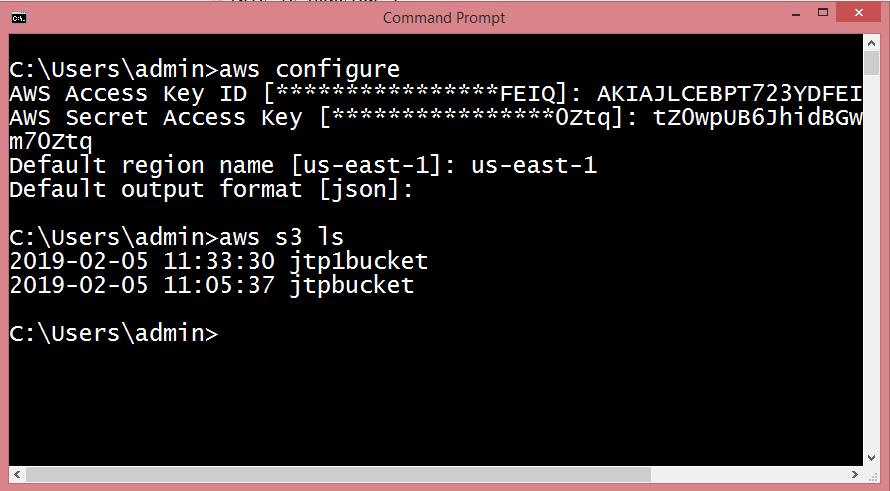


From the above screen, we observe that access key and scret access key have been generated.

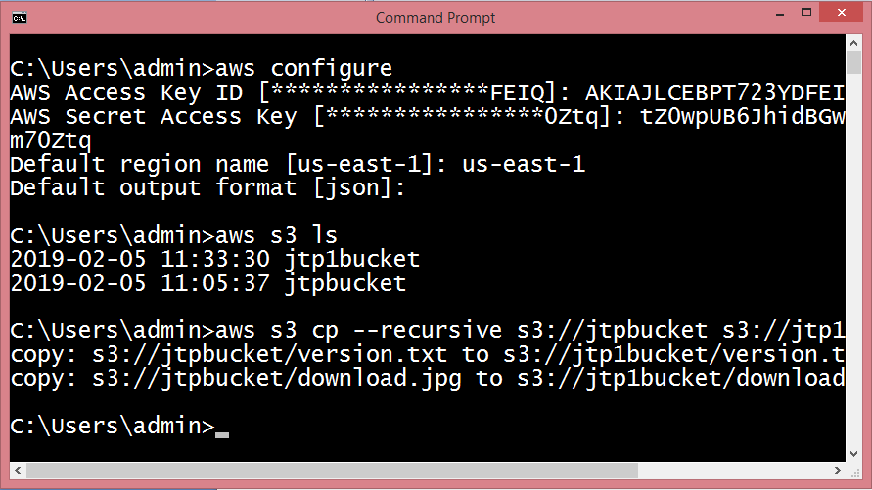
* Copy the access key and secret access key to the cmd.



* To view the S3 buckets, run the command **aws s3 ls**.

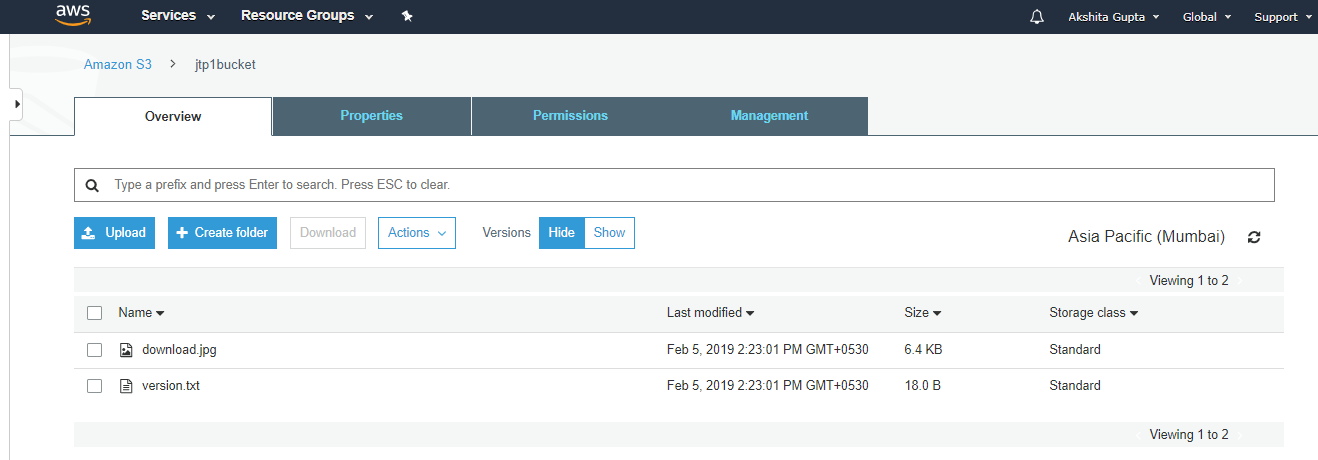


* To copy the objects of **jtpbucket to jtp1bucket, run the command aws s3 cp?recursive s3://jtpbucket s3://jtp1bucket**.



The above screen shows that the objects of **jtpbucket** have been copied to the **jtp1bucket**.

* Click on the **"jtp1bucket"**.



From the above screen, we observed that all the files in the original bucket have been replicated to another bucket, i.e., jtp1bucket.

Note: If any further changes made in the original bucket will always be copied to its replicated bucket.

Important points to be remembered:

* Versioning must be enabled on both the source and destination buckets.
* The regions of both the buckets must be unique.
* All the files in an original bucket are not replicated automatically, and they can be replicated through AWS CLI. All the subsequent files are replicated automatically.
* Files in a file cannot be replicated to multiple buckets.
* Delete markers are not replicated.
* Delete versions or Delete markers are not replicated.