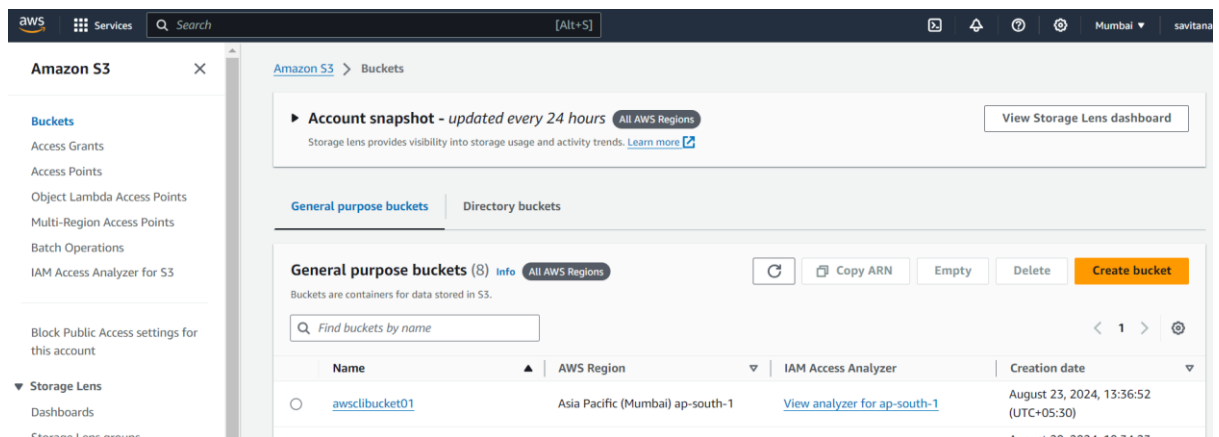


Task 1 - Copy a object from one bucket to another bucket using cli

a. Create one bucket

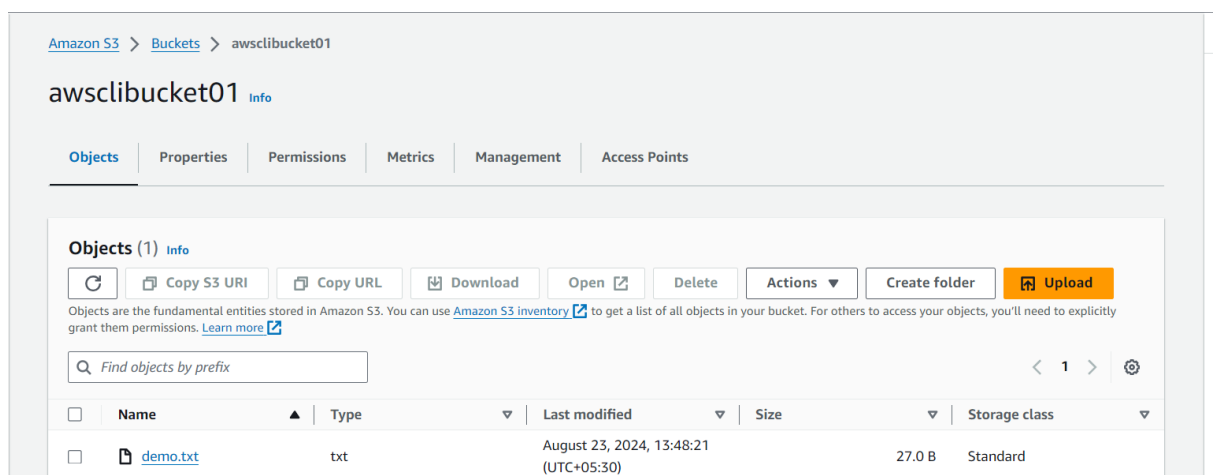
```
C:\Users\HP>aws s3 mb s3://awsclibucket01
make_bucket: awsclibucket01
```



b. Upload object into the bucket

```
C:\Users\HP>aws s3 cp desktop/aws/demo.txt s3://awsclibucket01
upload: desktop\aws\demo.txt to s3://awsclibucket01/demo.txt

C:\Users\HP>
```



c. Create one more destination bucket to copy object from existing bucket

```
C:\Users\HP>aws s3 mb s3://destinationclibucket01
make_bucket: destinationclibucket01

C:\Users\HP>
```

Amazon S3 > Buckets

Account snapshot - updated every 24 hours All AWS Regions
Storage lens provides visibility into storage usage and activity trends. [Learn more](#)

General purpose buckets | Directory buckets

General purpose buckets (8) Info All AWS Regions Refresh Copy ARN Empty

Buckets are containers for data stored in S3.

Find buckets by name

	Name	AWS Region	IAM Access Analyzer
<input type="radio"/>	awslclibucket01	Asia Pacific (Mumbai) ap-south-1	View analyzer for ap-south-1
<input type="radio"/>	crs3destinationbucket	Asia Pacific (Tokyo) ap-northeast-1	View analyzer for ap-northeast-1
<input type="radio"/>	destinationclibucket01	Asia Pacific (Mumbai) ap-south-1	View analyzer for ap-south-1

d. Now copy object from awslclibucket01 to destinationclibucket01

```
C:\Users\HP>aws s3 sync s3://awslclibucket01 s3://destinationclibucket01
copy: s3://awslclibucket01/demo.txt to s3://destinationclibucket01/demo.txt

C:\Users\HP>
```

Amazon S3 > Buckets > destinationclibucket01

destinationclibucket01 Info

Objects | Properties | Permissions | Metrics | Management | Access Points

Objects (1) Info Refresh Copy S3 URI Copy URL Download Open Delete Actions Create folder Upload

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Find objects by prefix

	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	demo.txt	txt	August 23, 2024, 13:57:19 (UTC+05:30)	27.0 B	Standard

Task2 : Implement s3 life cycle policy using CLI

- We are trying to apply life cycle policy to existing bucket (awsclibucket01)
- We have create one json file and editing as per requirement

```
[root@ip-172-31-11-241 ~]# cat > lifecycle.json
{
  "Rules": [
    {
      "ID": "MoveToIA",
      "Filter": {
        "Prefix": ""
      },
      "Status": "Enabled",
      "Transitions": [
        {
          "Days": 30,
          "StorageClass": "STANDARD_IA"
        }
      ],
      "Expiration": {
        "Days": 365
      },
      "NoncurrentVersionTransitions": [
        {
          "NoncurrentDays": 30,
          "StorageClass": "GLACIER"
        }
      ],
      "NoncurrentVersionExpiration": {
        "NoncurrentDays": 365
      },
      "AbortIncompleteMultipartUpload": {
        "DaysAfterInitiation": 7
      }
    }
  ]
}
[root@ip-172-31-11-241 ~]# aws s3api put-bucket-lifecycle-configuration --bucket
awsclibucket01 --lifecycle-configuration file://lifecycle.json
[root@ip-172-31-11-241 ~]#
```

- Lifepolicy have been applied

Amazon S3 > Buckets > awsclibucket01 > Lifecycle configuration

Lifecycle configuration [Info](#)

To manage your objects so that they are stored cost effectively throughout their lifecycle, configure their lifecycle. A lifecycle configuration is a set of rules that define actions that Amazon S3 applies to a group of objects. Lifecycle rules run once per day.

Lifecycle rules (1) [Refresh](#) [View details](#) [Edit](#) [Delete](#) [Actions](#) [Create lifecycle rule](#)

Use lifecycle rules to define actions you want Amazon S3 to take during an object's lifetime such as transitioning objects to another storage class, archiving them, or deleting them after a specified period of time. [Learn more](#)

< 1 > [Settings](#)

	Lifecycle rul...	Status	Scope	Current versi...	Noncurrent v...	Expired obje...	Incomplete ...
<input type="radio"/>	MoveToIA	Enabled	Entire bucket	Transition to Standar	Transition to Glacier f	-	Permanently delete

Task3- Impletement cross region replication using CLI

- Enabled versioning of awsclibucket01

```
C:\Users\HP>aws s3api put-bucket-versioning --bucket awsclibucket01 --versioning-configuration Status=Enabled
C:\Users\HP>
```

Amazon S3 Buckets > awsclibucket01

awsclibucket01 [Info](#)

Objects **Properties** Permissions Metrics Management Access Points

Bucket overview

AWS Region Asia Pacific (Mumbai) ap-south-1	Amazon Resource Name (ARN) arn:aws:s3:::awsclibucket01	Creation date August 23, 2024, 13:36:52 (UTC+05:30)
--	---	--

Bucket Versioning [Edit](#)

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning
Enabled

Multi-factor authentication (MFA) delete
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the [MFA delete console](#).

b. create new bucket in another region and Enabled versioning of crrtargetbucket

```
C:\Users\HP>aws s3api create-bucket --bucket crrtargetbucket --region us-east-1
{
  "Location": "/crrtargetbucket"
}

C:\Users\HP>aws s3api put-bucket-versioning --bucket crrtargetbucket --versioning-configuration Status=Enabled
C:\Users\HP>
```

Amazon S3 Buckets > crrtargetbucket

crrtargetbucket [Info](#)

Objects **Properties** Permissions Metrics Management Access Points

Bucket overview

AWS Region US East (N. Virginia) us-east-1	Amazon Resource Name (ARN) arn:aws:s3:::crrtargetbucket	Creation date August 23, 2024, 14:47:14 (UTC+05:30)
---	--	--

Bucket Versioning [Edit](#)

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning
Enabled

c. now applying replication rule to awsclibucket(sourcebucket)

```
[root@ip-172-31-11-241 ~]# cat > replication.json
{
  "Role": "arn:aws:iam::664418982701:role/CRRclirole",
  "Rules": [
    {
      "Status": "Enabled",
      "Priority": 1,
      "DeleteMarkerReplication": { "Status": "Disabled" },
      "Filter": { "Prefix": "" },
      "Destination": {
        "Bucket": "arn:aws:s3:::crrtargetbucket"
      }
    }
  ]
}
```

```
[root@ip-172-31-11-241 ~]# aws s3api put-bucket-replication --bucket awsclibucket01 --replication-configuration file:///replication.json
[root@ip-172-31-11-241 ~]#
```

[root@ip-172-31-11-241 ~]# aws s3api put-bucket-replication --bucket awsclibucket01 --replication-configuration file:///replication.json

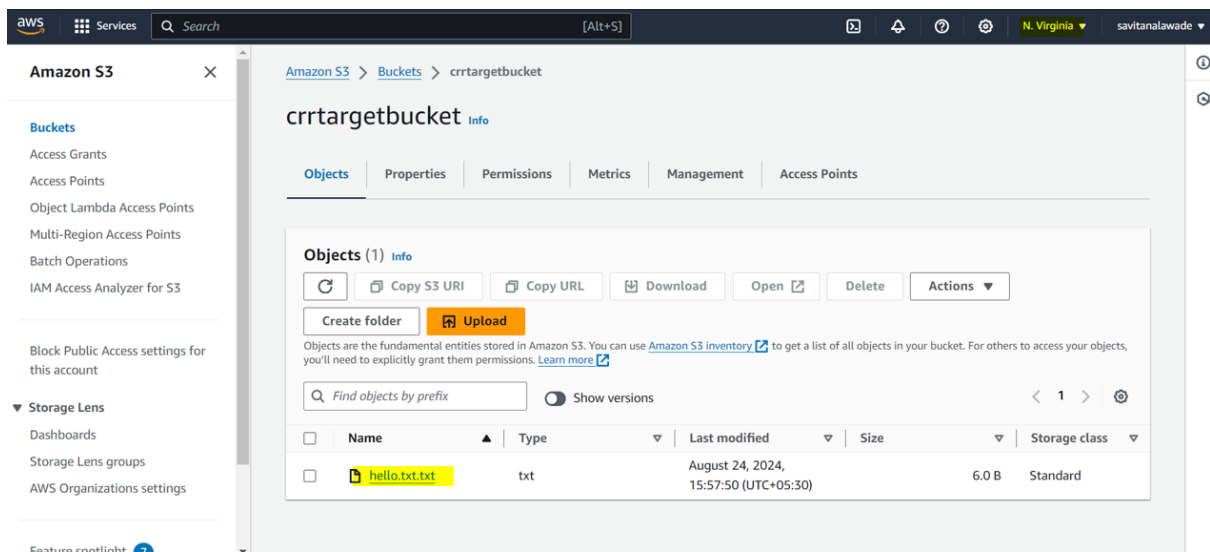
The screenshot shows the Amazon S3 console interface for the bucket 'awsclibucket01'. On the left, there is a navigation menu with options like Buckets, Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Storage Lens, and AWS Organizations settings. The main content area displays the 'Replication rules' section. It shows a table with one replication rule:

Replication rule name	Status	Destination bucket	Destination Region	Priority	Scope	Storage class	Replica owner	Replication Time Control
Y2ESOGExZD ETyJ4M00NT Y1LTjM2YTN TQyNTE3MjVl ODQ5	Enabled	s3://crrtargetbucket	US East (N. Virginia) us-east-1	1	Entire bucket	Same as source	Same as source	Disabled

- d. After applying CRR now I have uploaded one file to source bucket it should visible in destination as well.

```
C:\Users\HP>aws s3 cp desktop/aws/hello.txt.txt s3://awsclibucket01
upload: desktop\aws\hello.txt.txt to s3://awsclibucket01/hello.txt.txt
C:\Users\HP>
```

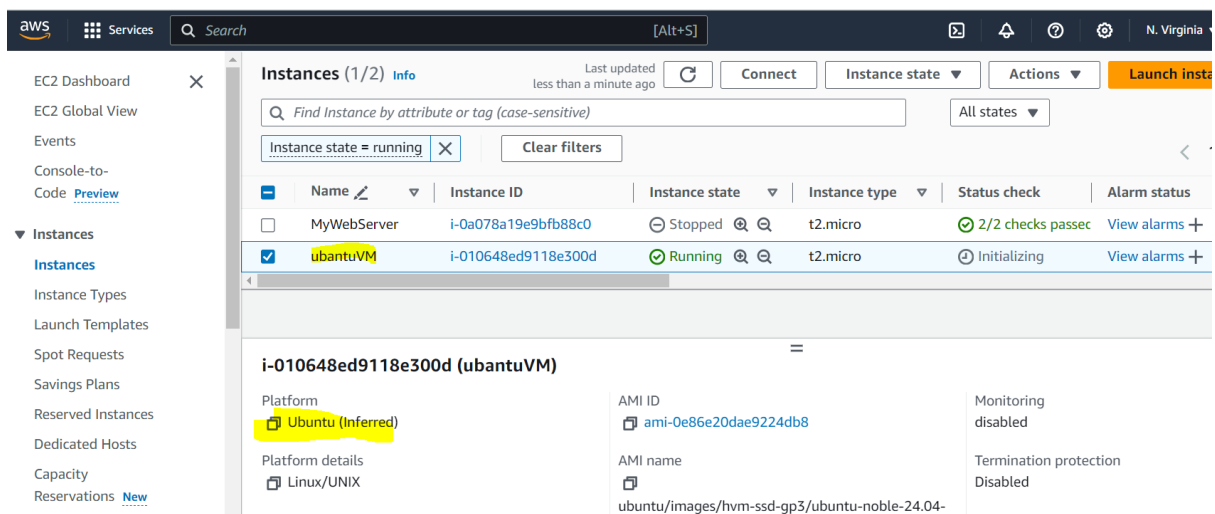
e. We can see object is there is destination bucket.



Created one instance using ubuntu OS

24th August 2024

1. created one instance with ubuntu OS



2.Connected to putty as a ubuntu user

ubuntu@ip-172-31-92-47: ~

```
Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-92-47:~$
```

3.AWS CLI is successfully installed

root@ip-172-31-92-47: ~

```
inflating: aws/dist/docutils/writers/s5_html/themes/small-black/pretty.css
inflating: aws/dist/docutils/writers/s5_html/themes/small-white/pretty.css
inflating: aws/dist/docutils/writers/s5_html/themes/small-white/framing.css
inflating: aws/dist/docutils/writers/latex2e/default.tex
inflating: aws/dist/docutils/writers/latex2e/docutils.sty
inflating: aws/dist/docutils/writers/latex2e/titlepage.tex
inflating: aws/dist/docutils/writers/latex2e/titlingpage.tex
inflating: aws/dist/docutils/writers/latex2e/xelatex.tex
inflating: aws/dist/docutils/writers/html5_polyglot/responsive.css
inflating: aws/dist/docutils/writers/html5_polyglot/minimal.css
inflating: aws/dist/docutils/writers/html5_polyglot/plain.css
inflating: aws/dist/docutils/writers/html5_polyglot/tuftig.css
inflating: aws/dist/docutils/writers/html5_polyglot/template.txt
inflating: aws/dist/docutils/writers/html5_polyglot/math.css
inflating: aws/dist/docutils/writers/odf_odt/styles.odt
inflating: aws/dist/docutils/writers/pep_html/template.txt
inflating: aws/dist/docutils/writers/pep_html/pep.css
inflating: aws/dist/docutils/writers/html4css1/template.txt
inflating: aws/dist/docutils/writers/html4css1/html4css1.css
oot@ip-172-31-92-47:~# ./aws/install
ou can now run: /usr/local/bin/aws --version
oot@ip-172-31-92-47:~# aws --version
aws-cli/2.17.37 Python/3.11.9 Linux/6.8.0-1012-aws exe/x86_64.ubuntu.24
oot@ip-172-31-92-47:~#
```

4.Attached IAM role to instance

```

root@ip-172-31-92-47:~# aws configure list
      Name                               Value                               Type    Location
      ----                               -
profile                               <not set>                          None     None
access_key                            *****ESYZ                        iam-role
secret_key                            *****fBVl                        iam-role
region                                us-east-1                          imds
root@ip-172-31-92-47:~#

```

5.Created user and group

```

root@ip-172-31-46-93:~# aws iam create-user --user ubuntu-user001
{
  "User": {
    "Path": "/",
    "UserName": "ubuntu-user001",
    "UserId": "AIDAZVMTVDMW55IQBYKYR",
    "Arn": "arn:aws:iam::664418982701:user/ubuntu-user001",
    "CreateDate": "2024-08-24T13:42:14+00:00"
  }
}
root@ip-172-31-46-93:~# aws iam create-group --group ubuntugrp01
{
  "Group": {
    "Path": "/",
    "GroupName": "ubuntugrp01",
    "GroupId": "AGPAZVMTVDMW6MGUSNW5O",
    "Arn": "arn:aws:iam::664418982701:group/ubuntugrp01",
    "CreateDate": "2024-08-24T13:42:40+00:00"
  }
}
root@ip-172-31-46-93:~#

```

6.Created AccessKey & secretaccesskey

```

root@ip-172-31-46-93:~# aws iam create-access-key --user-name ubuntu-user001
{
  "AccessKey": {
    "UserName": "ubuntu-user001",
    "AccessKeyId": "AKIAZVMTVDMWQU7BUEPH",
    "Status": "Active",
    "SecretAccessKey": "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",
    "CreateDate": "2024-08-24T13:47:34+00:00"
  }
}
root@ip-172-31-46-93:~#

```

7.Created instance using aws cli


```

root@ip-172-31-46-93:~# aws ec2 run-instances --image-id ami-0522ab6e1ddcc7055 -
-count 1 --instance-type t2.micro --key-name AWSdevops --security-group-ids sg-0
700c20b660b917ac
{
  "Groups": [],
  "Instances": [
    {
      "AmiLaunchIndex": 0,
      "ImageId": "ami-0522ab6e1ddcc7055",
      "InstanceId": "i-0d6b9ae0cc40a0836",
      "InstanceType": "t2.micro",
      "KeyName": "AWSdevops",
      "LaunchTime": "2024-08-24T13:36:22+00:00",
      "Monitoring": {
        "State": "disabled"
      },
      "Placement": {
        "AvailabilityZone": "ap-south-1a",
        "GroupName": "",
        "Tenancy": "default"
      },
      "PrivateDnsName": "ip-172-31-43-40.ap-south-1.compute.internal",
      "PrivateIpAddress": "172.31.43.40",
      "ProductCodes": [],
      "PublicDnsName": "",

```

Output

Instances (2) Info									
<div> <div>Find Instance by attribute or tag (case-sensitive)</div> <div>All states</div> </div> <div> <div>Instance state = running</div> <div>Clear filters</div> </div> <div> <div>Last updated less than a minute ago</div> <div>Connect</div> <div>Instance state</div> <div>Actions</div> <div>Launch instances</div> </div>									
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP	
<input type="checkbox"/>		i-0d6b9ae0cc40a0836	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1a	ec2-13-2	
<input type="checkbox"/>	ubuntuVM	i-001aab9bb62a43def	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1a	ec2-3-11	

