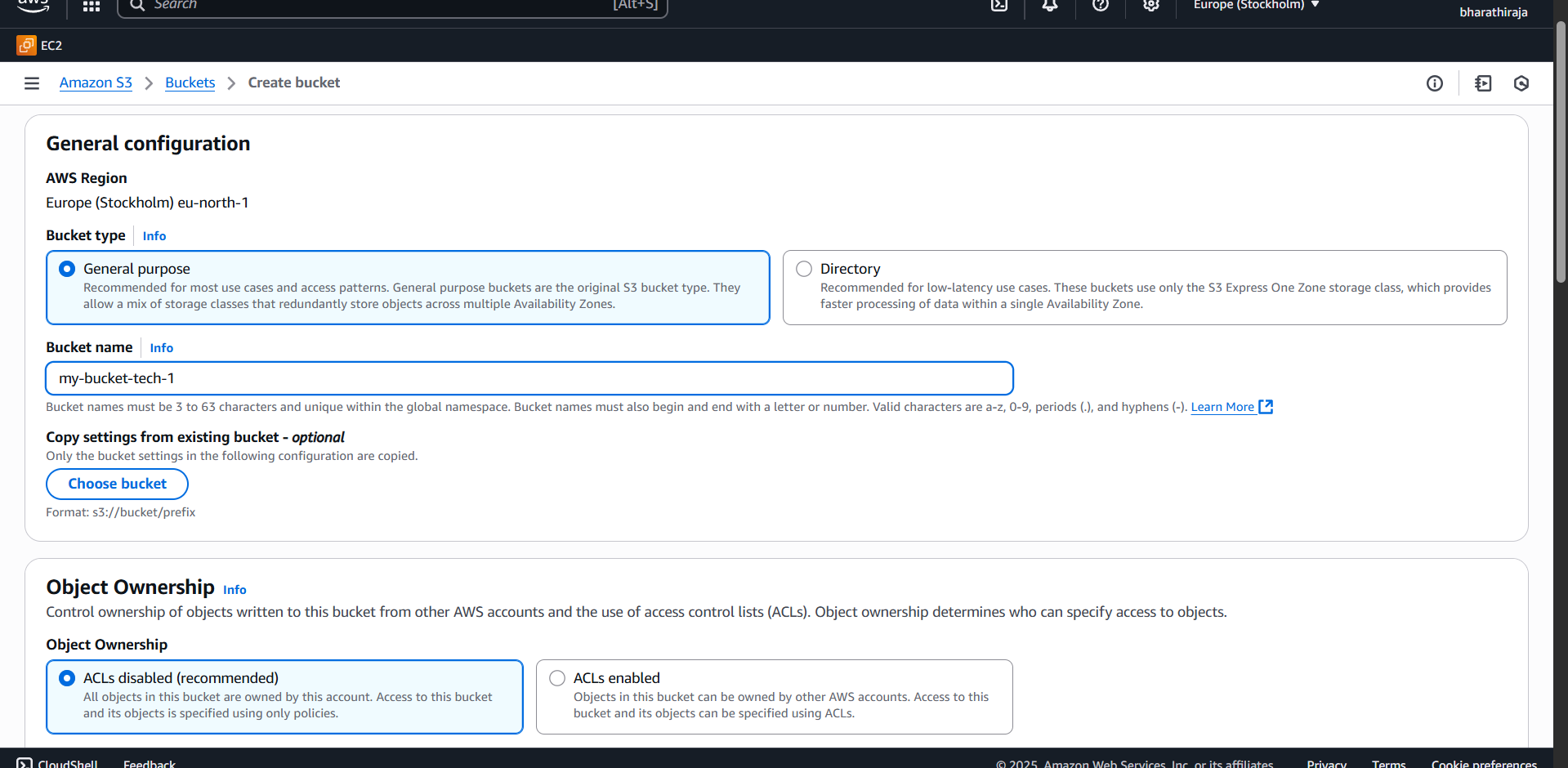
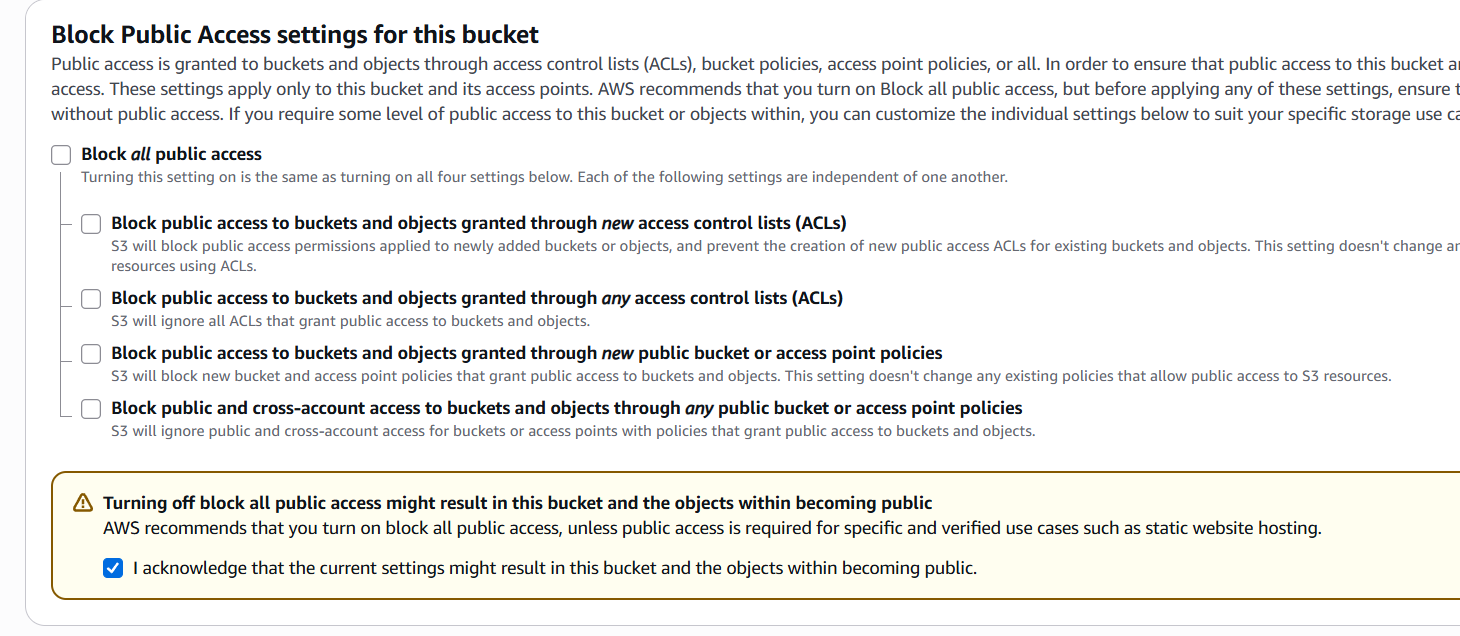
1. Create an S3 bucket and upload some objects to S3.

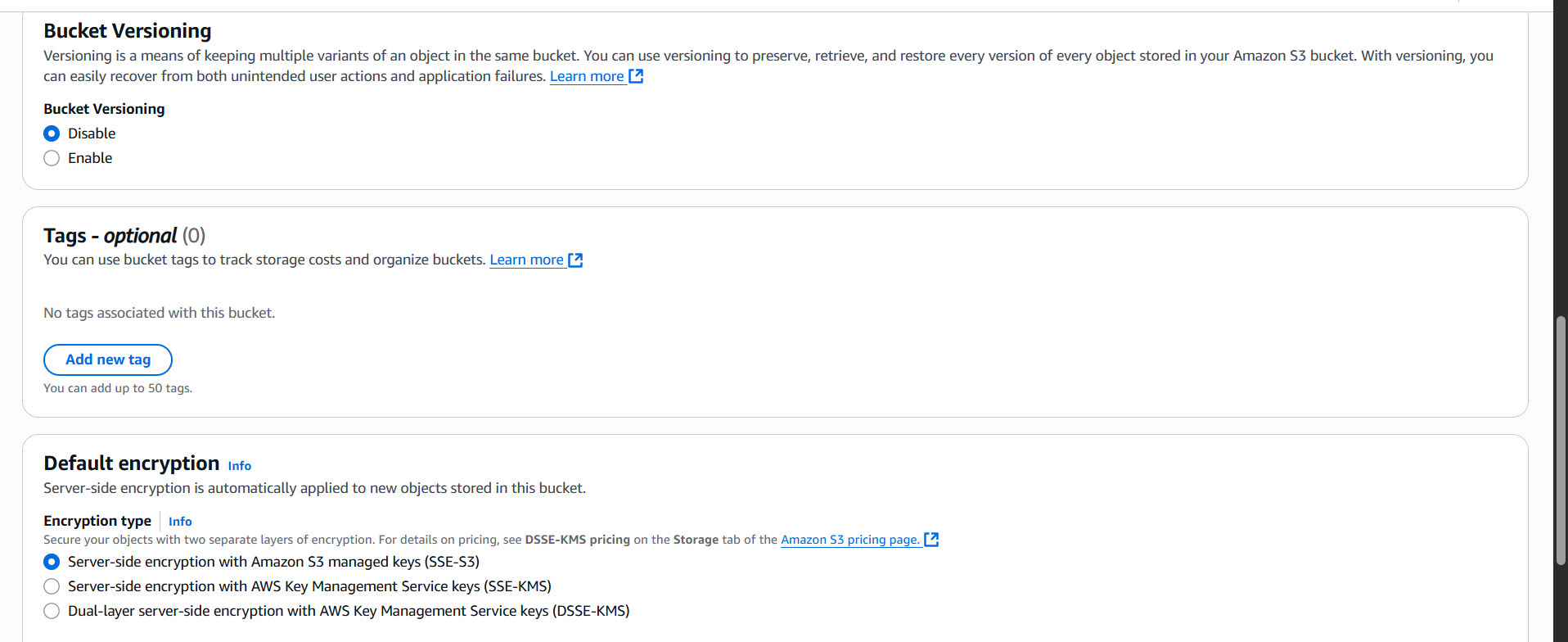
Go to the AWS service and open S3 bucket part, give click on create bucket



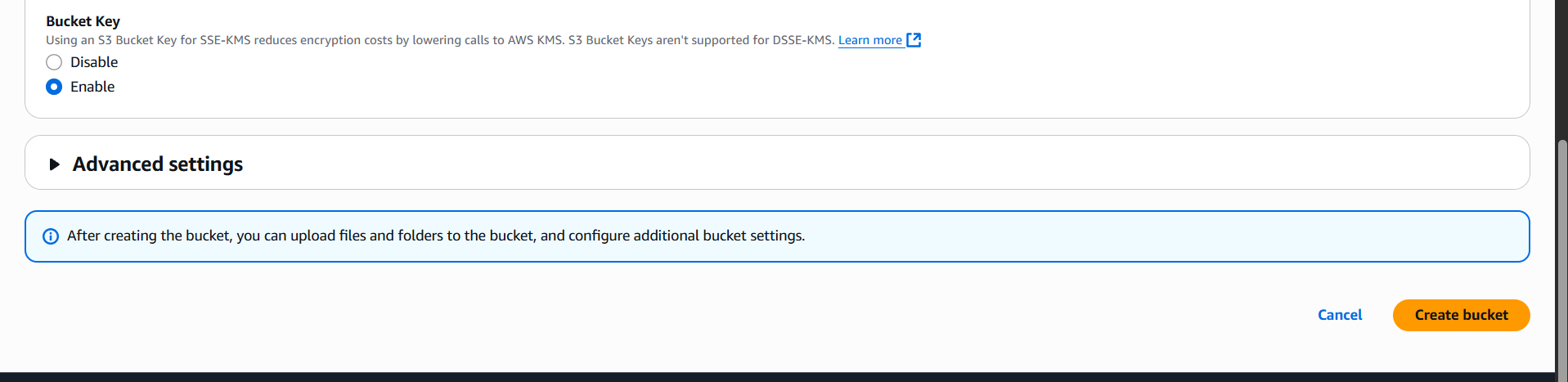


Give the bucket version also, disable, don’t change anything without knowing

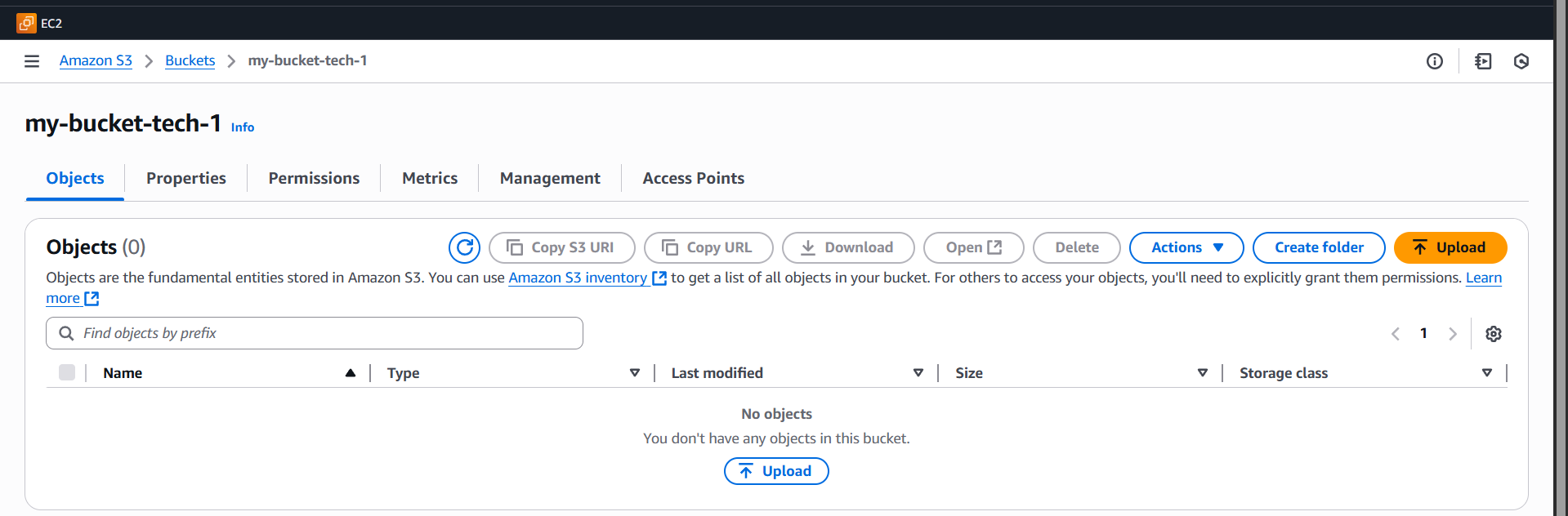
Server-side encryption with Amazon S3 managed keys (SSE-S3) => select this option



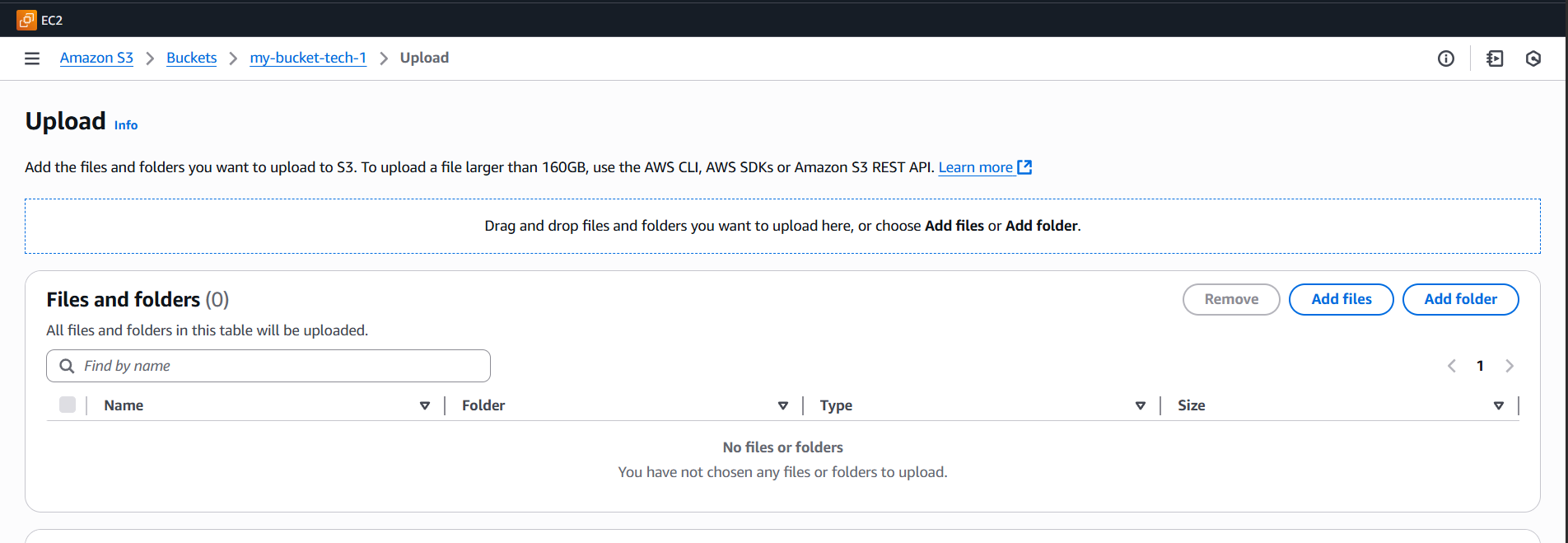
Bucket key should enable and click on create



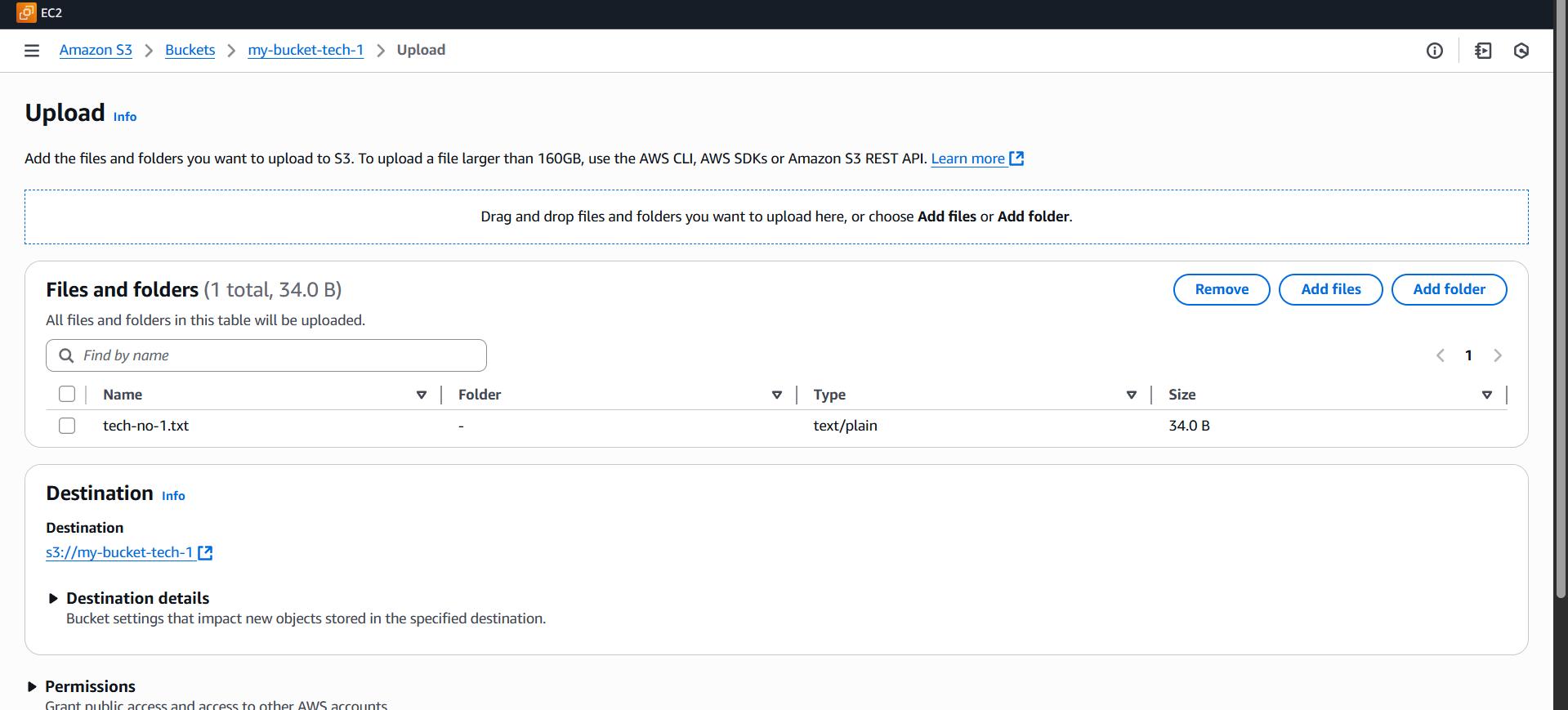
See successfully created



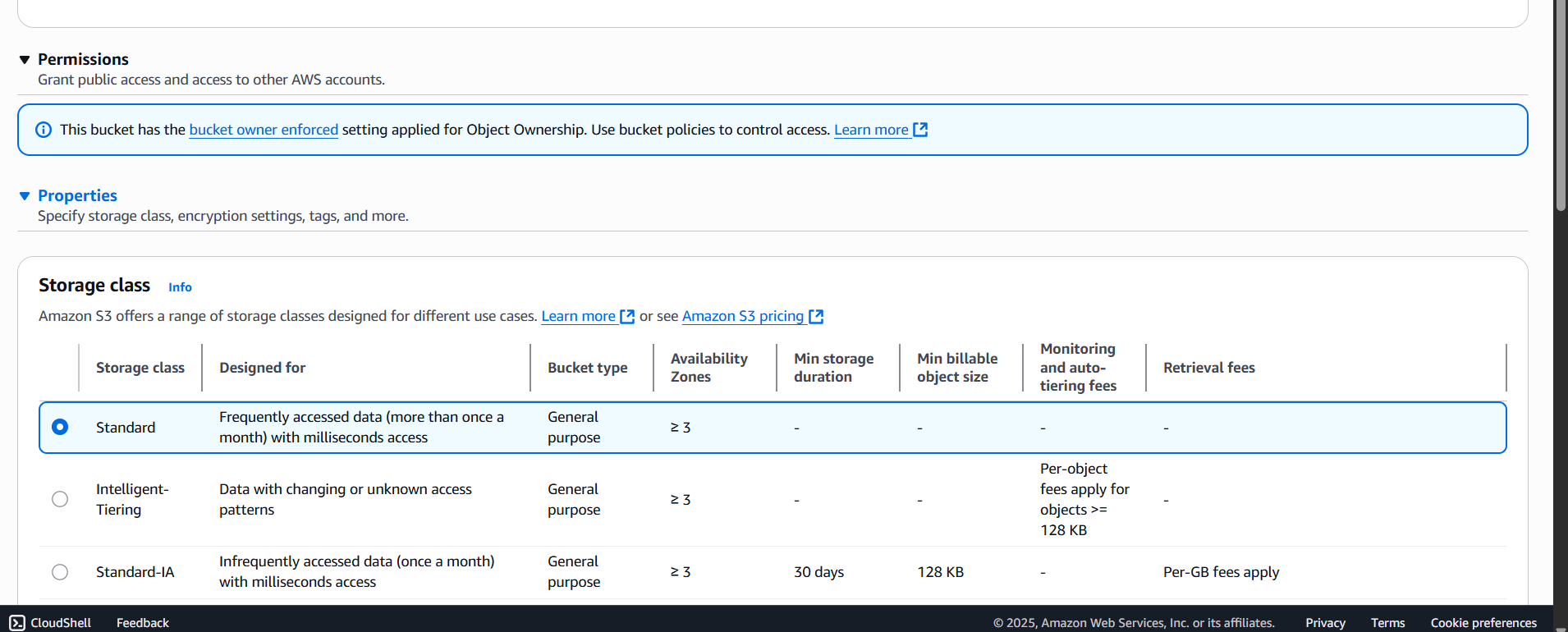
Click the upload option and click on add file



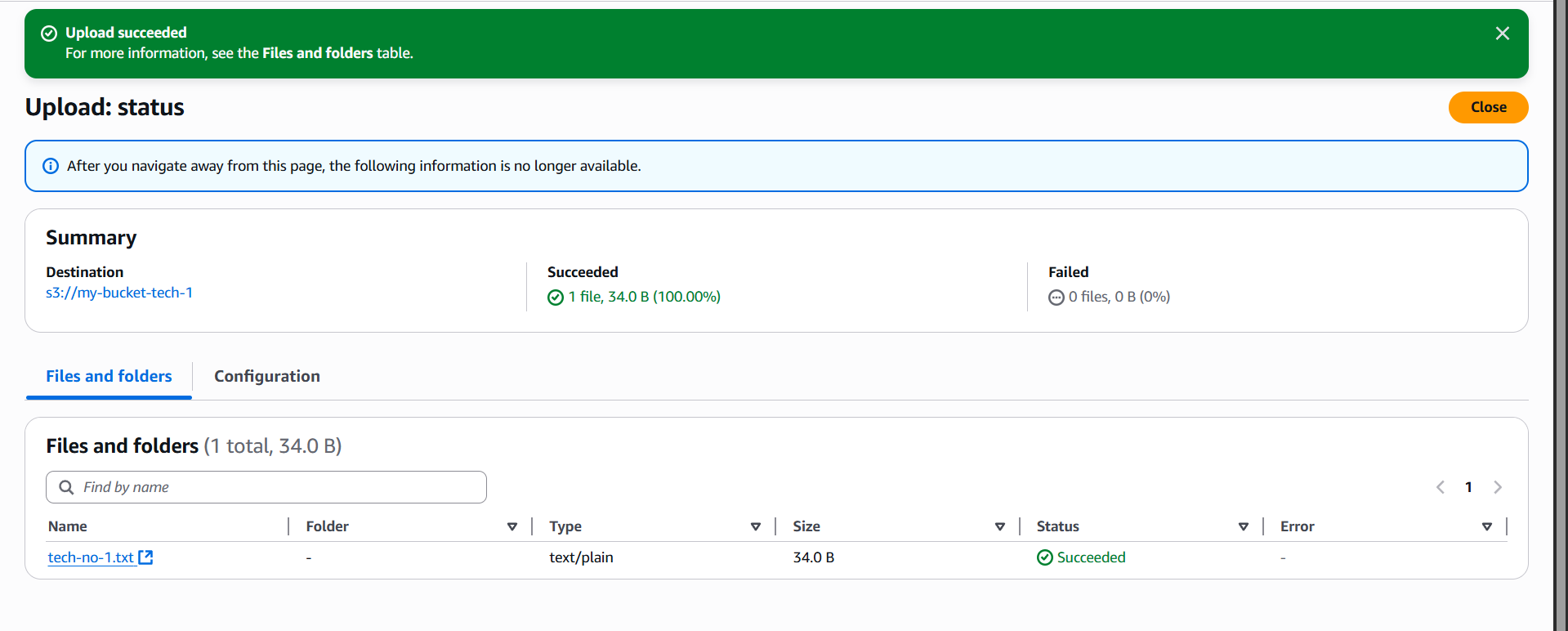
See I had one txt file, name it is rech-no-1



See, there is lot of permission



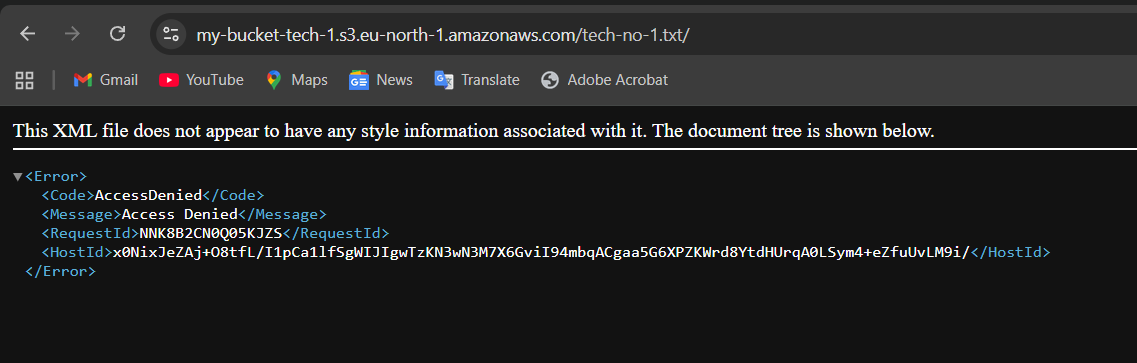
Uploaded



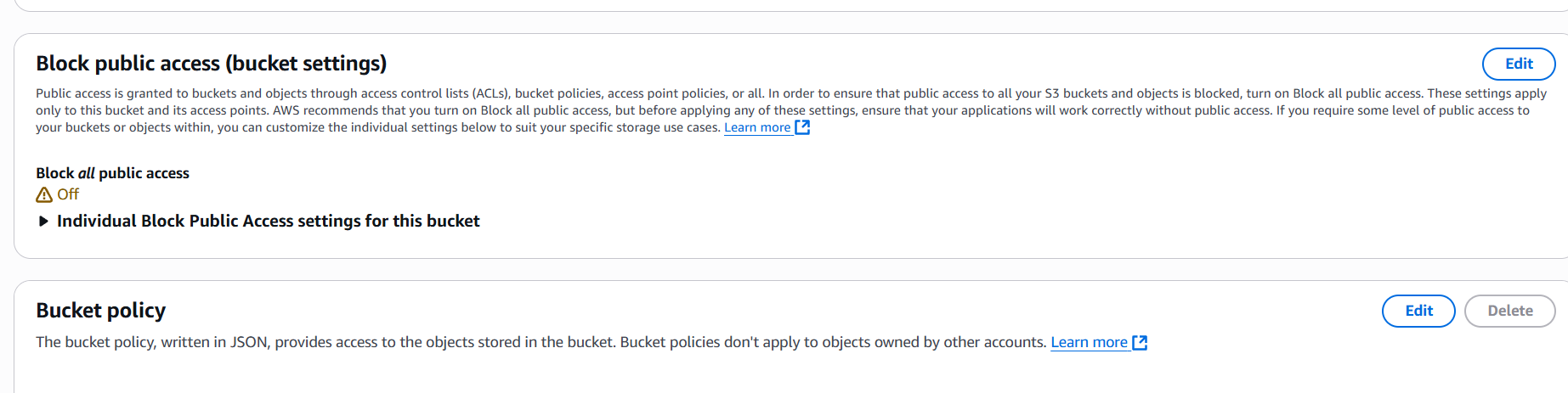
There is a file name, click on that file it will show like that   
In the right corner, there is a copy and paste it in the browser



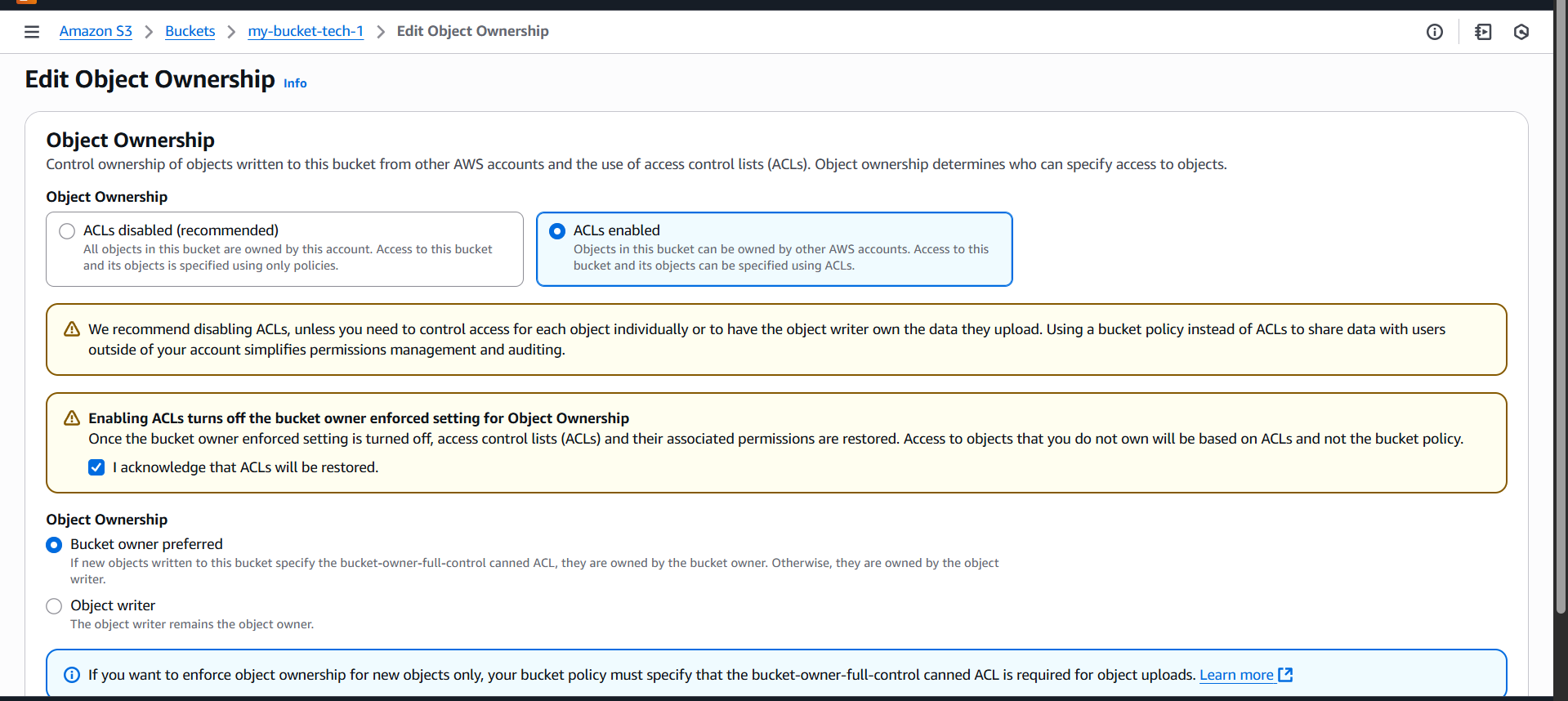
https://my-bucket-tech-1.s3.eu-north-1.amazonaws.com/tech-no-1.txt/  
Access is denied because we should permit to read and write

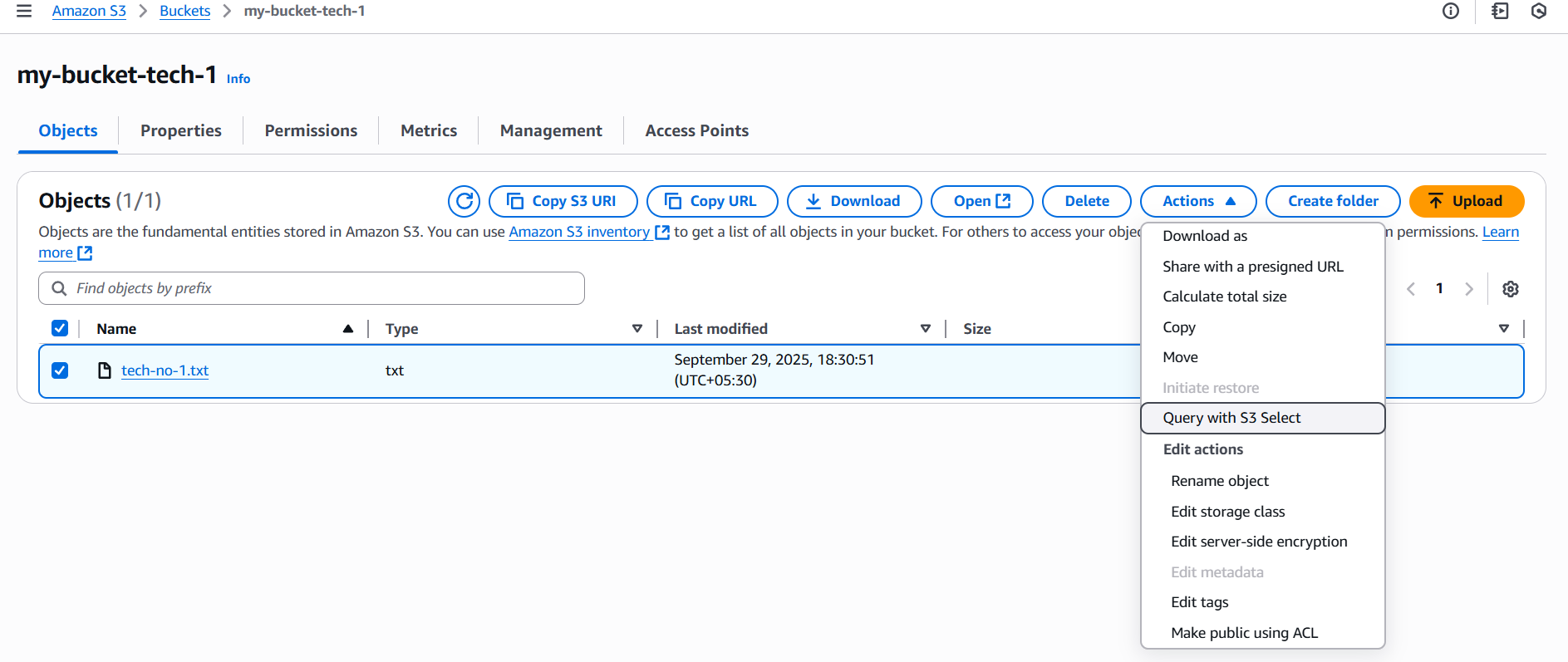


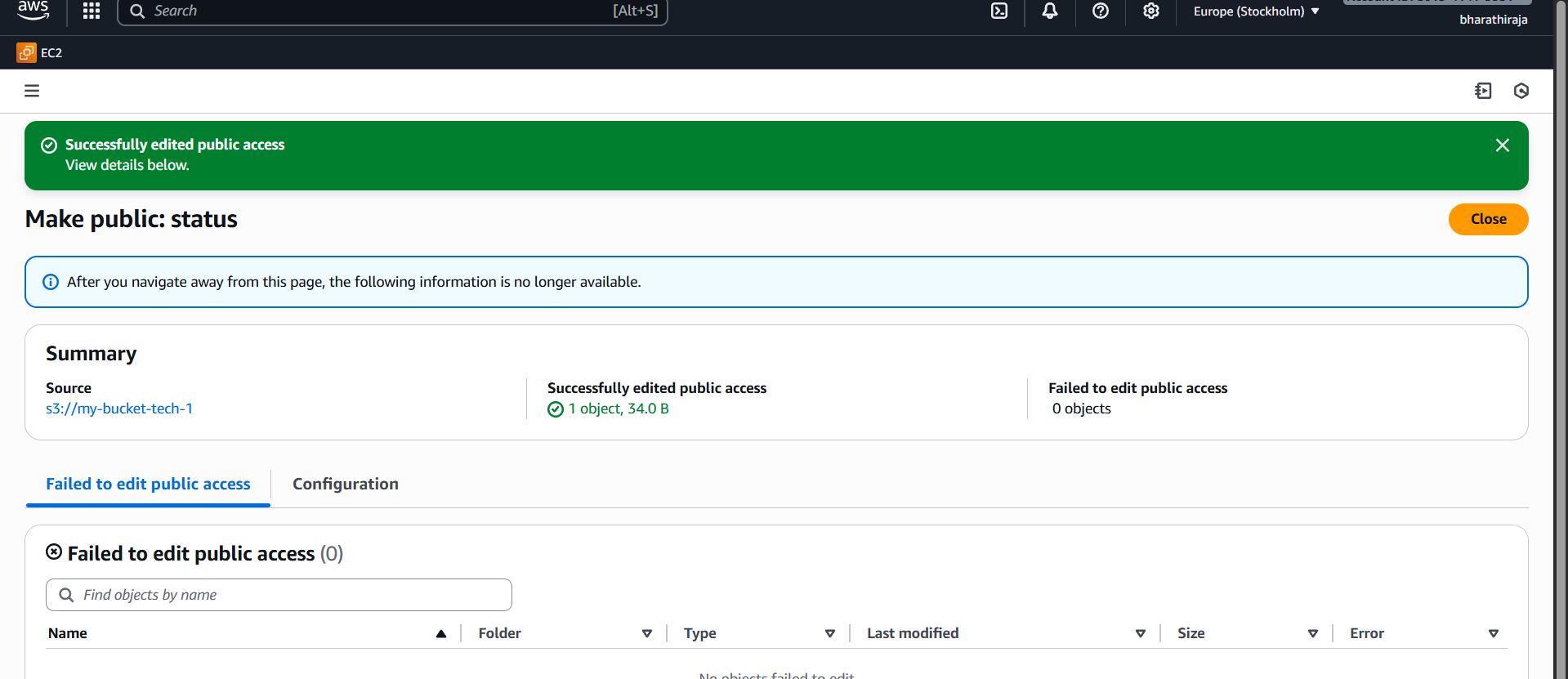
See there is blocking our access, so we have to allow our access



This is a way to give permission

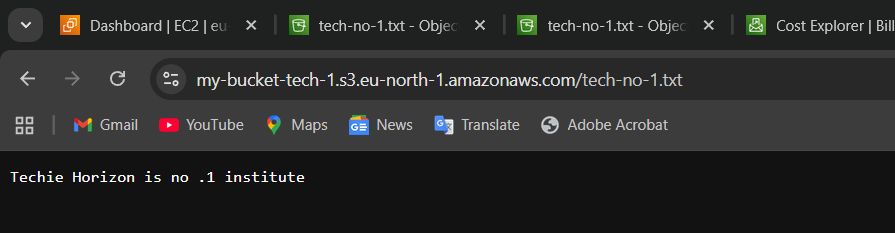


Save the changes go to next option this => make public using ACL  


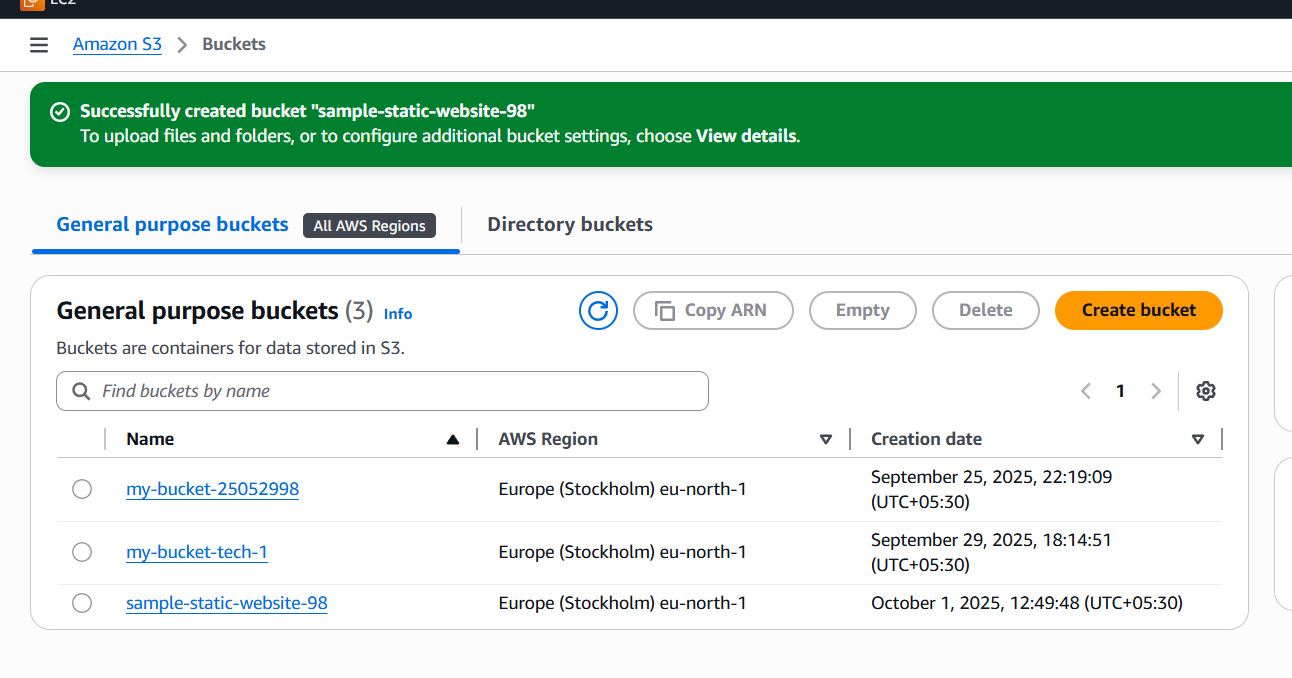


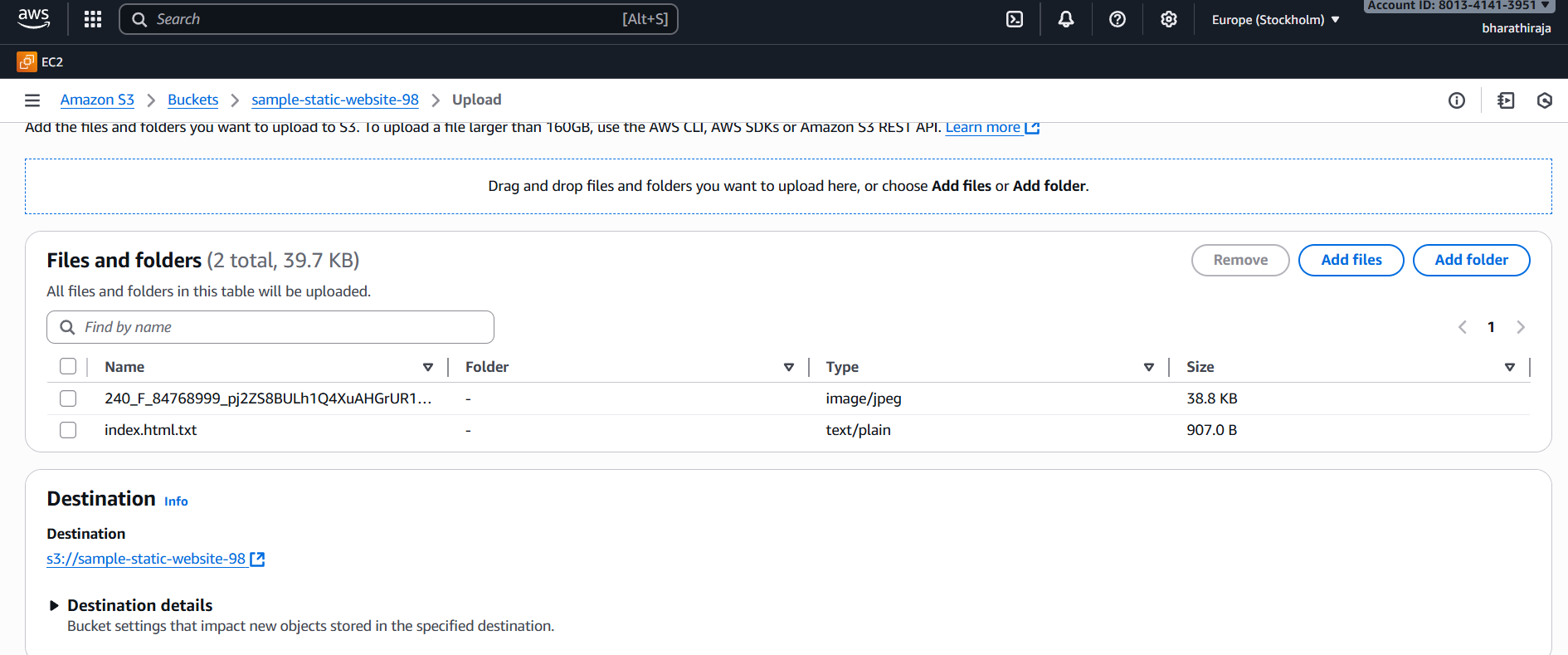
Give the acknowledgment and disable public access

After we see the result

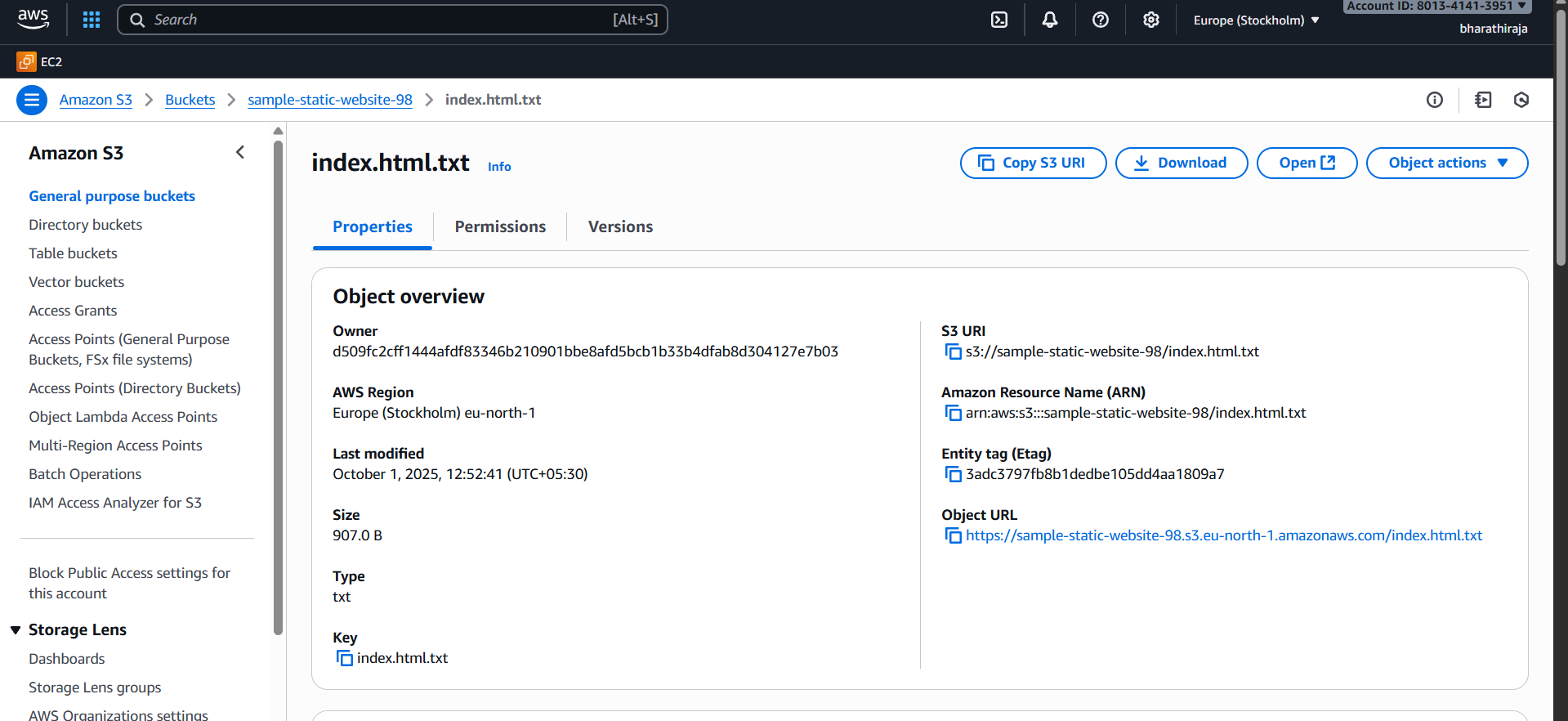


1. Deploy a static website in the S3 bucket.

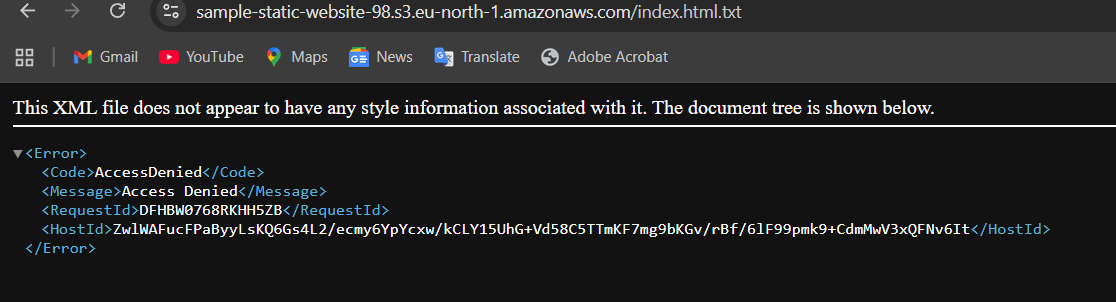
First, create a bucket sample static website -98

Then, view the upload file and add  


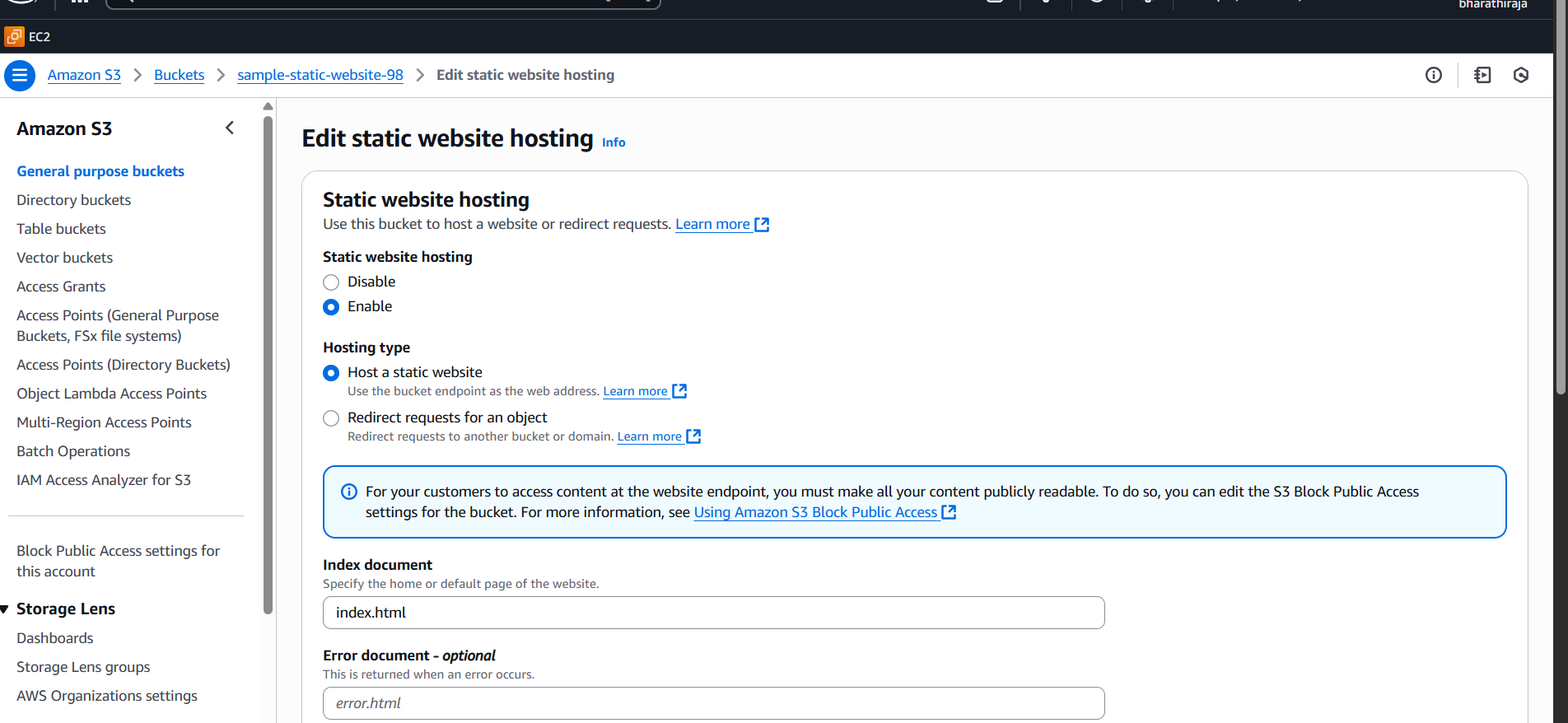
Uploaded file there id URL MEAN UNIFORM REGISTER LINK



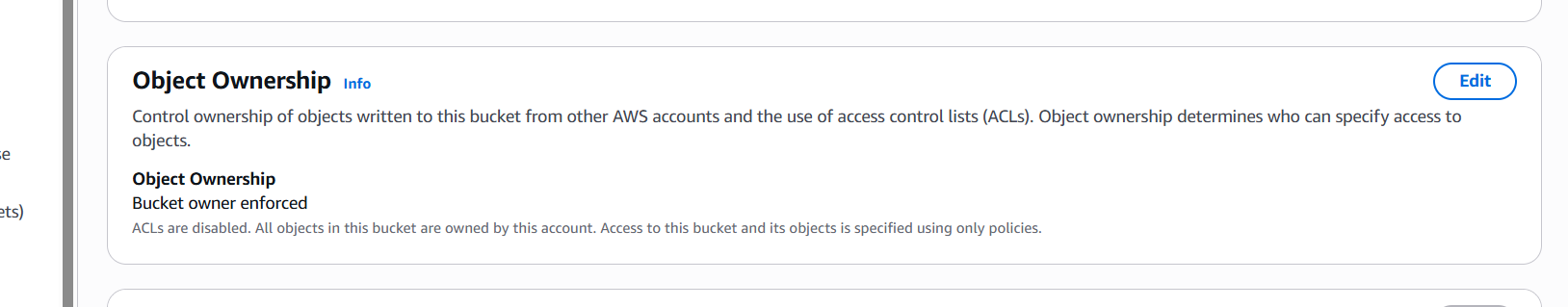
Copy the URL paste it on the website

  
It shows the access is denied because we should give permission

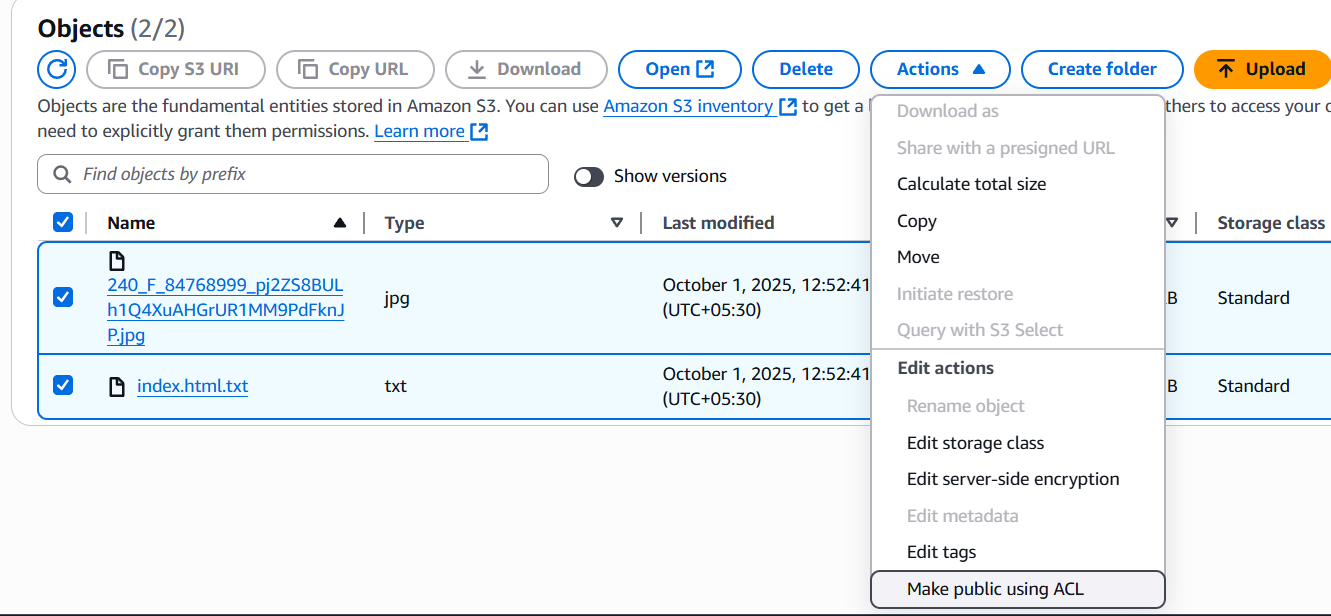
These kinds of permission we should give or else we can't access our host application



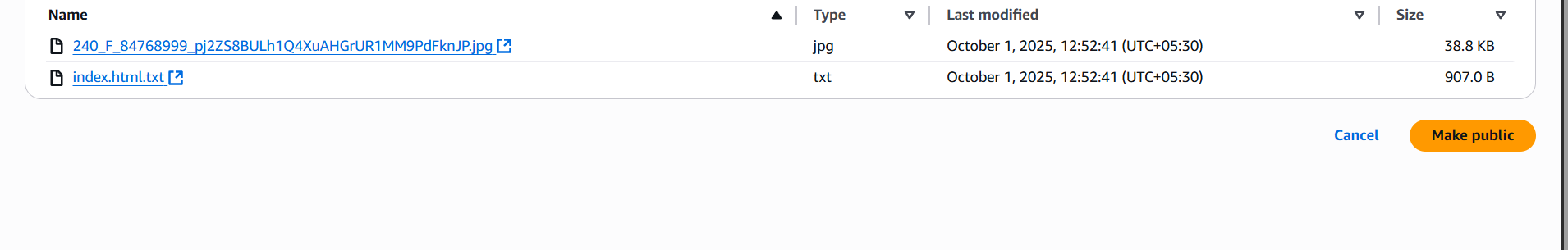
Ownership permission also we should also enable that access



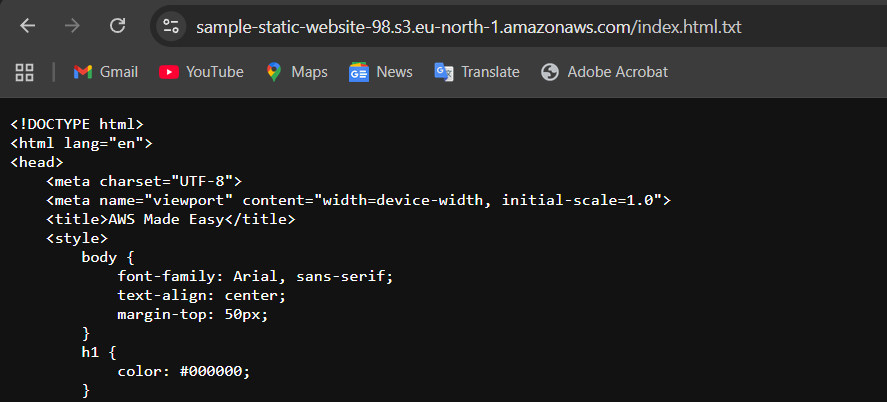
Make public using ACL



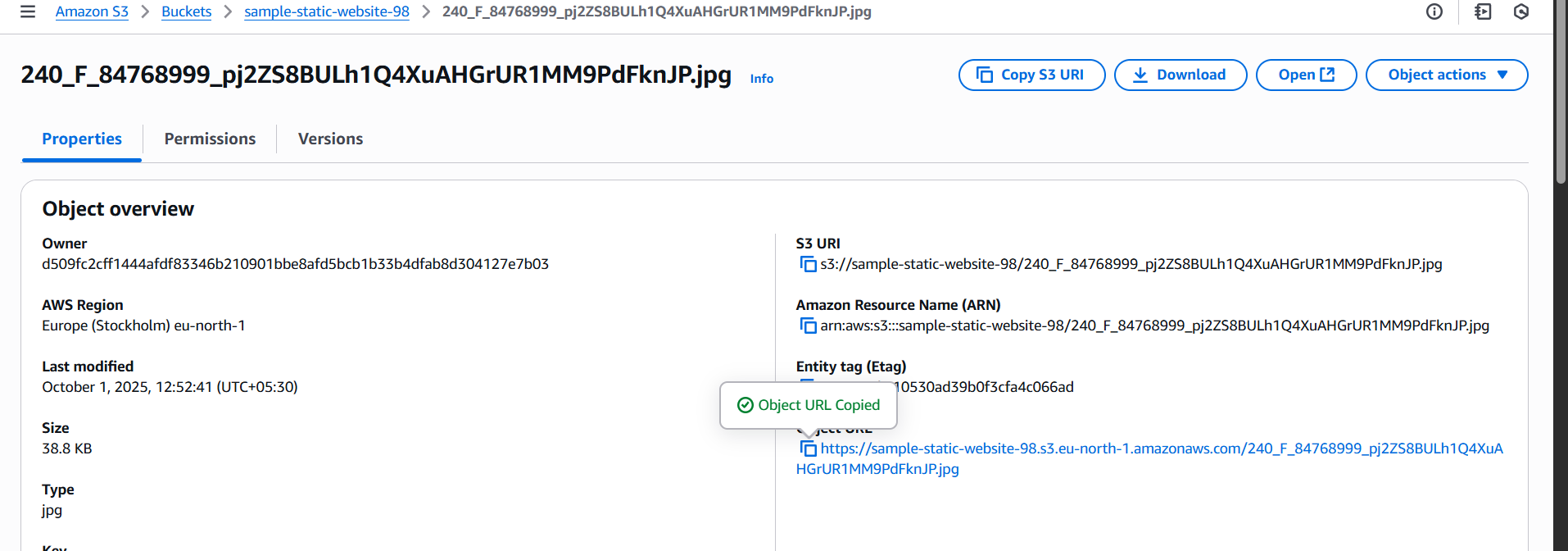
Click on the Make public



Copy the link and paste it on the website; it will show whatever changes we make in the file



See the URL, copy and paste on the website

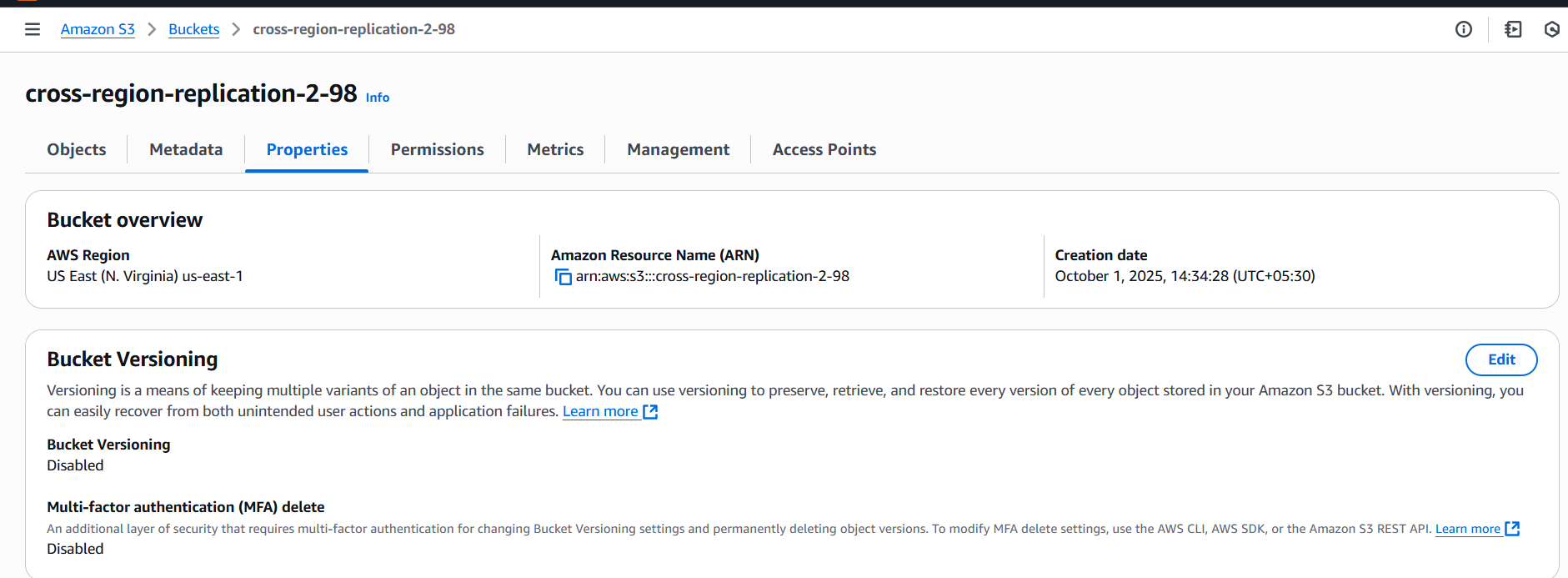


1. Enable cross-region replication on S3 buckets.

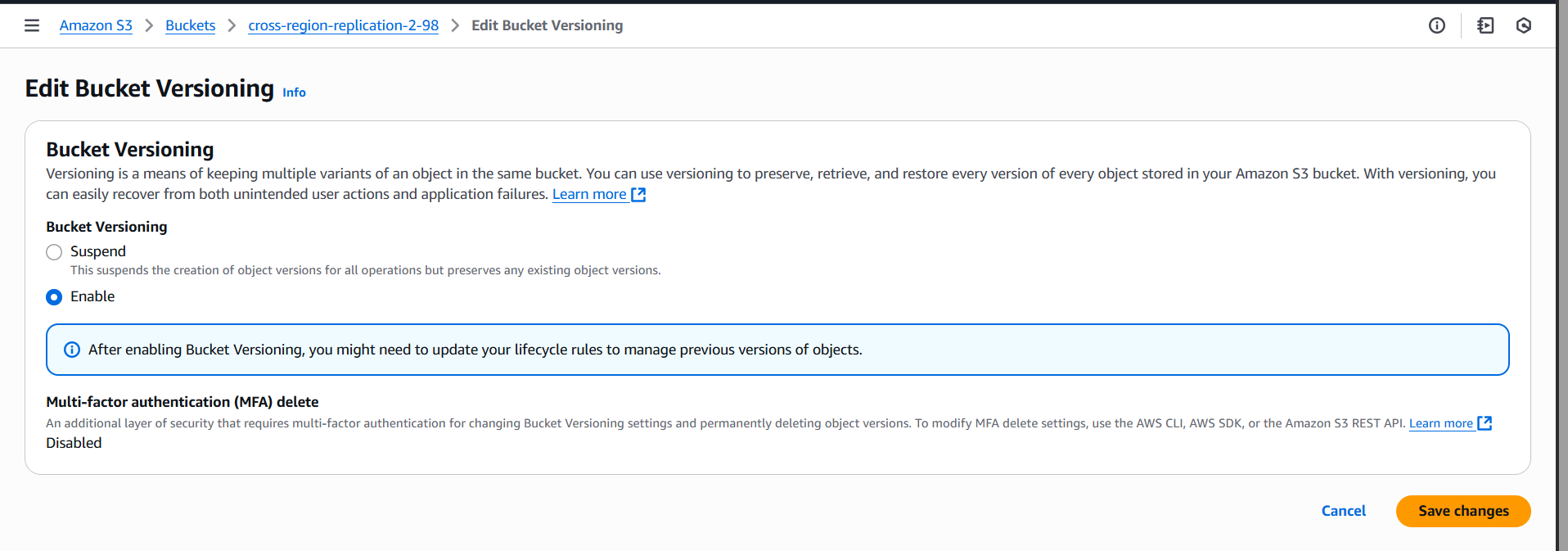
I created two buckets in different regions



We have to edit the bucket versioning default it is disable but we should enable the option in both buckets

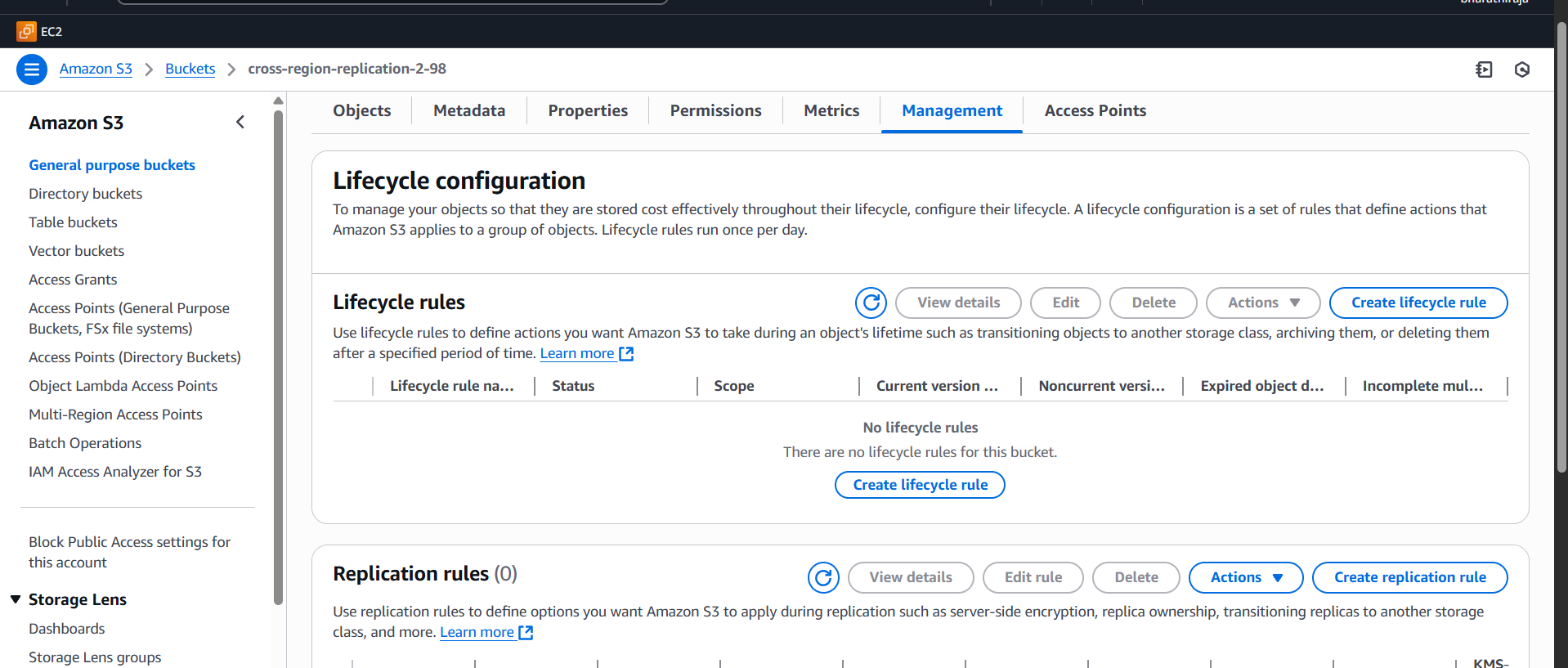


See like that enable and save change in both buckets

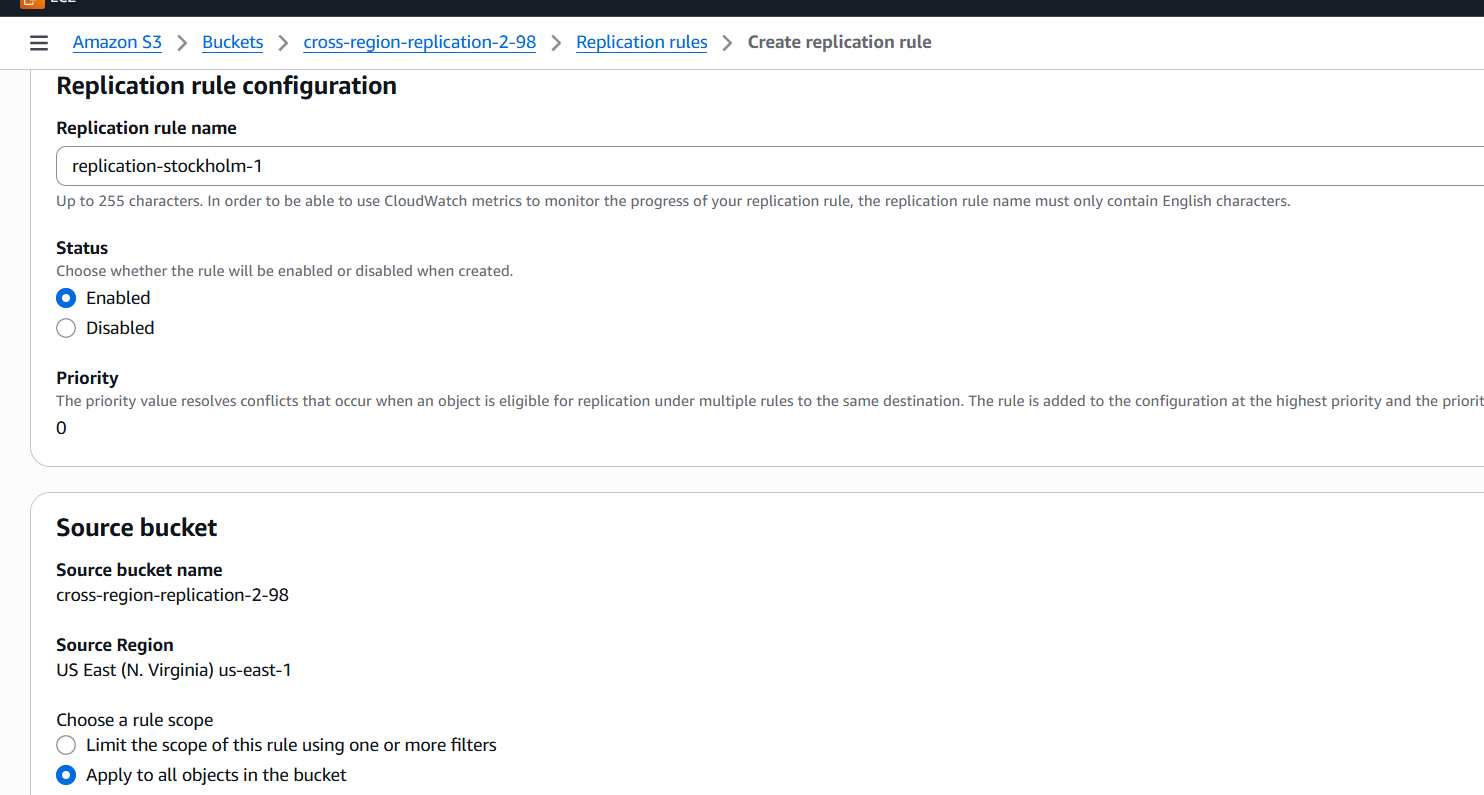


Open the bucket-2, go to management there is replicate option we have to give some details

**Create replication rule**



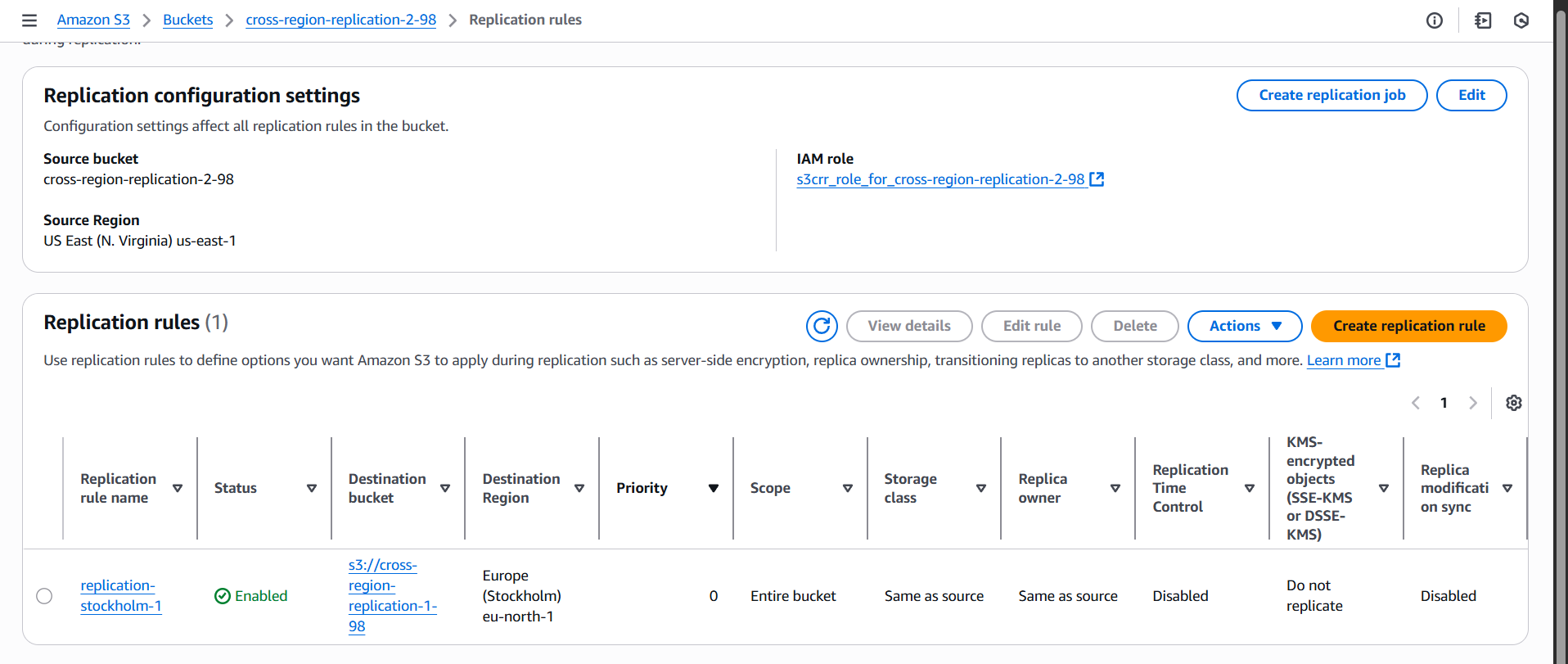
Name and apply to all objects in the bucket

choose a bucket in this account   
IAM ROLE create iam role



That much details is enough, don’t give extra naything without knowing



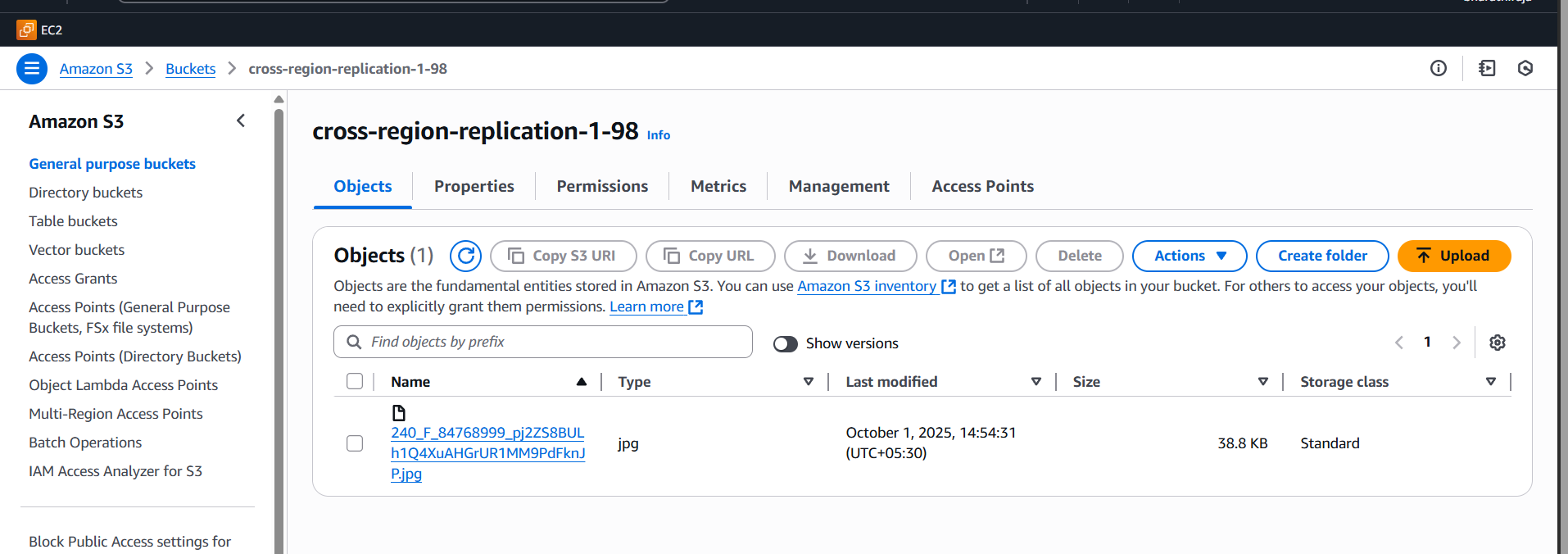


Replicate rule is opened

Upload one file 2 bucket it will replicate to bucket 1

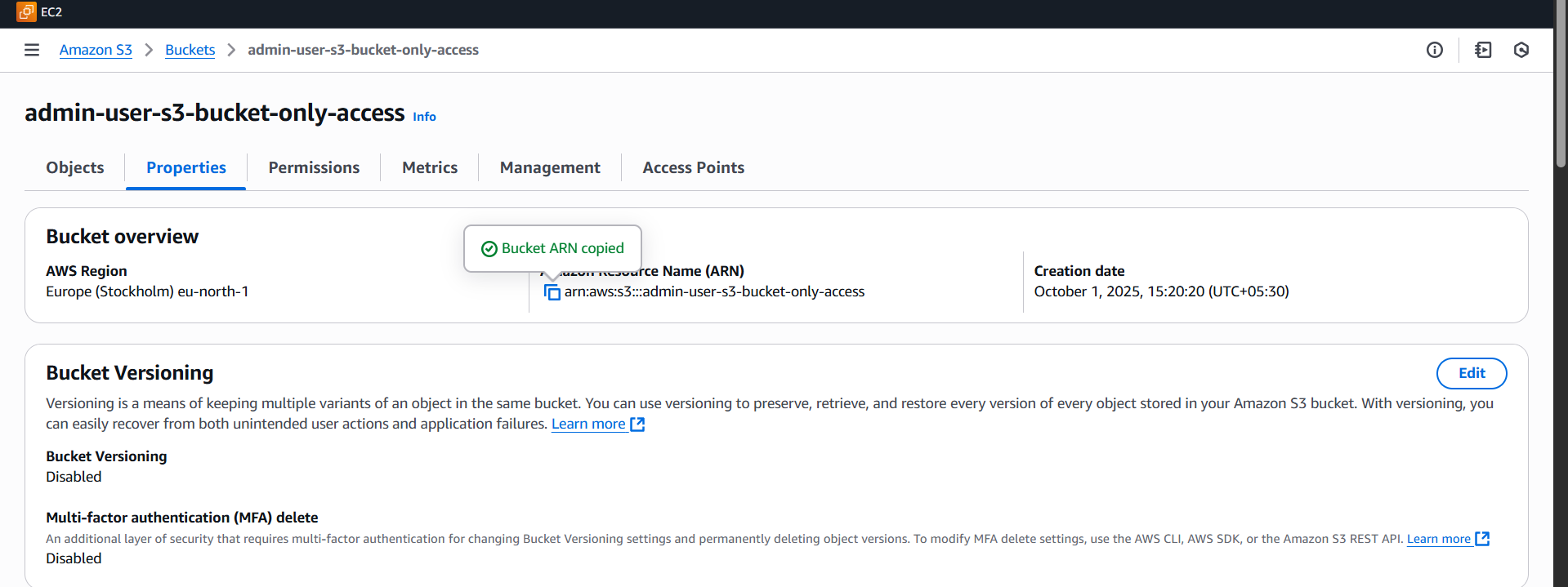


Here you can see which file upload from bucket-2 replicates bucket 1

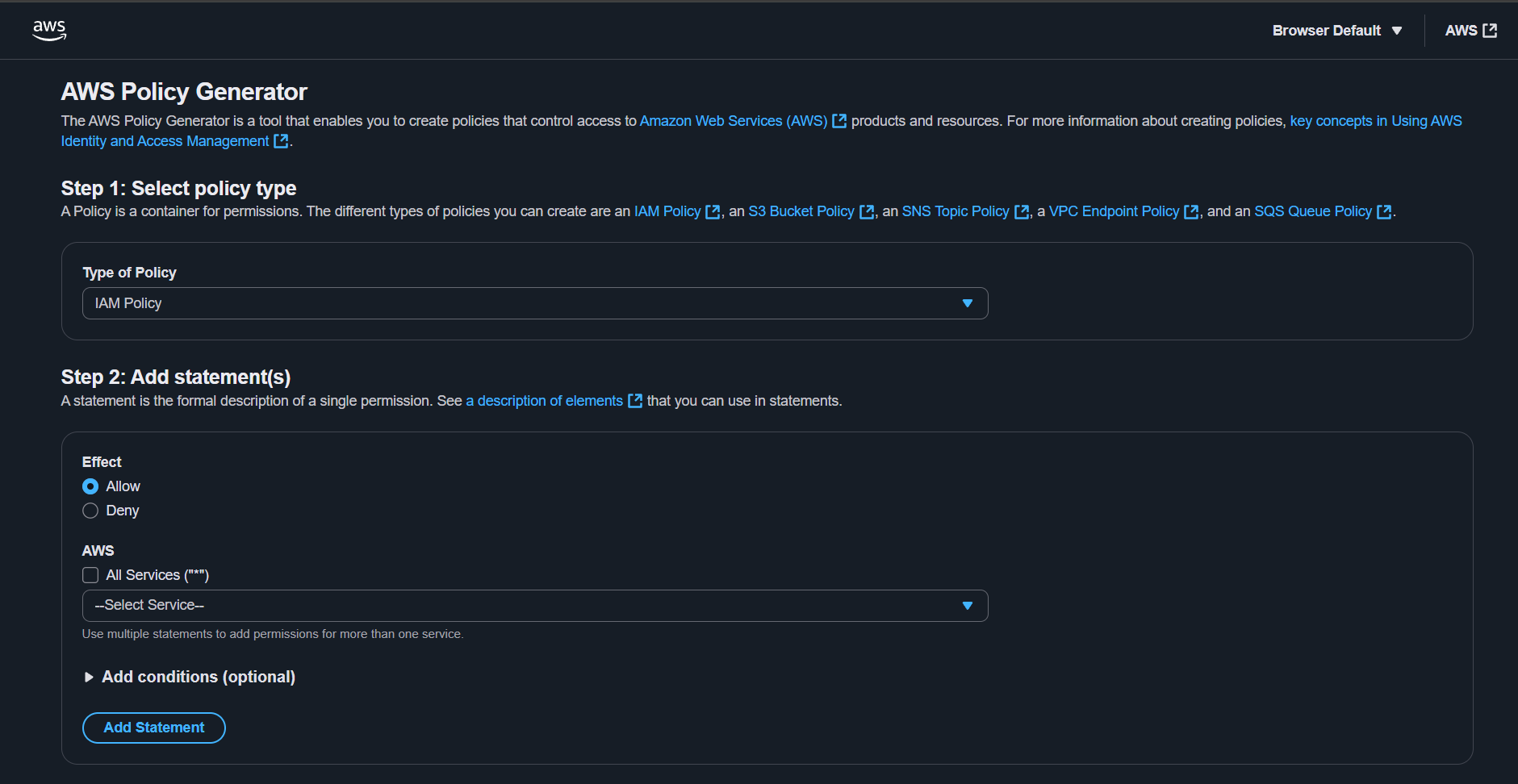


1. Configure a bucket policy so only the Admin user can see the objects of the S3 bucket.

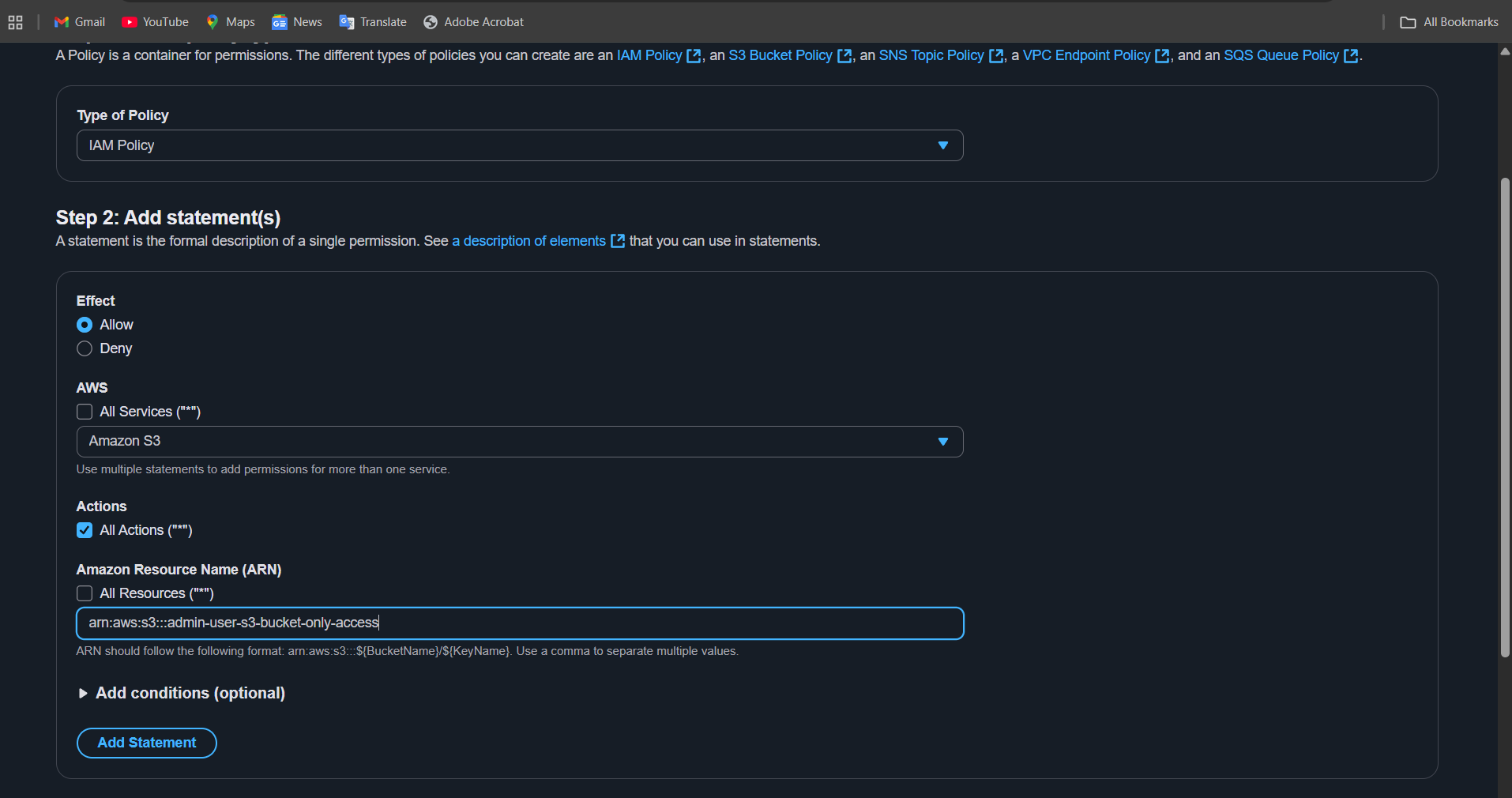
Create a bucker => go to the properties copy the ARN id



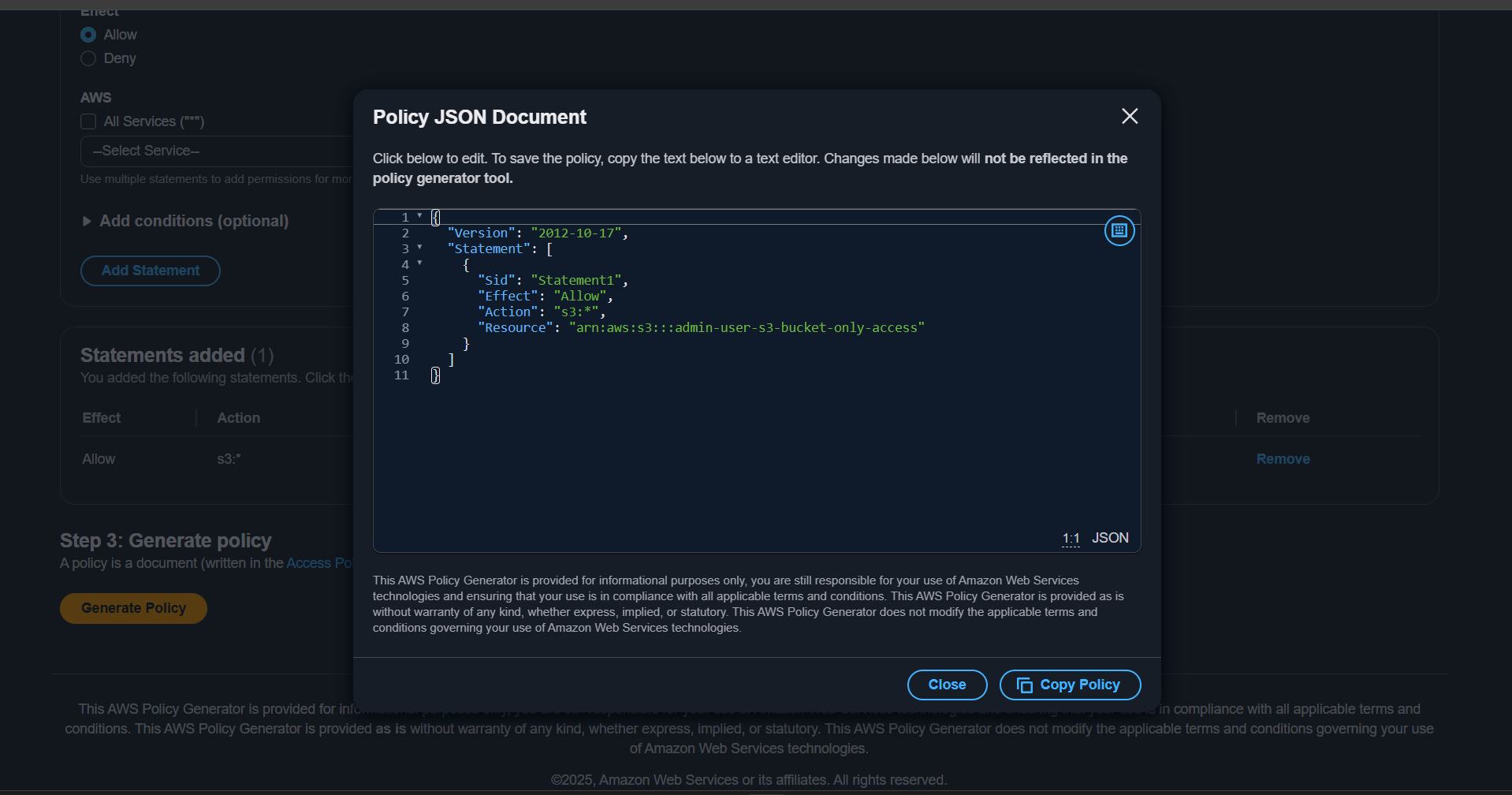
Go to the browser search AWS policy generator



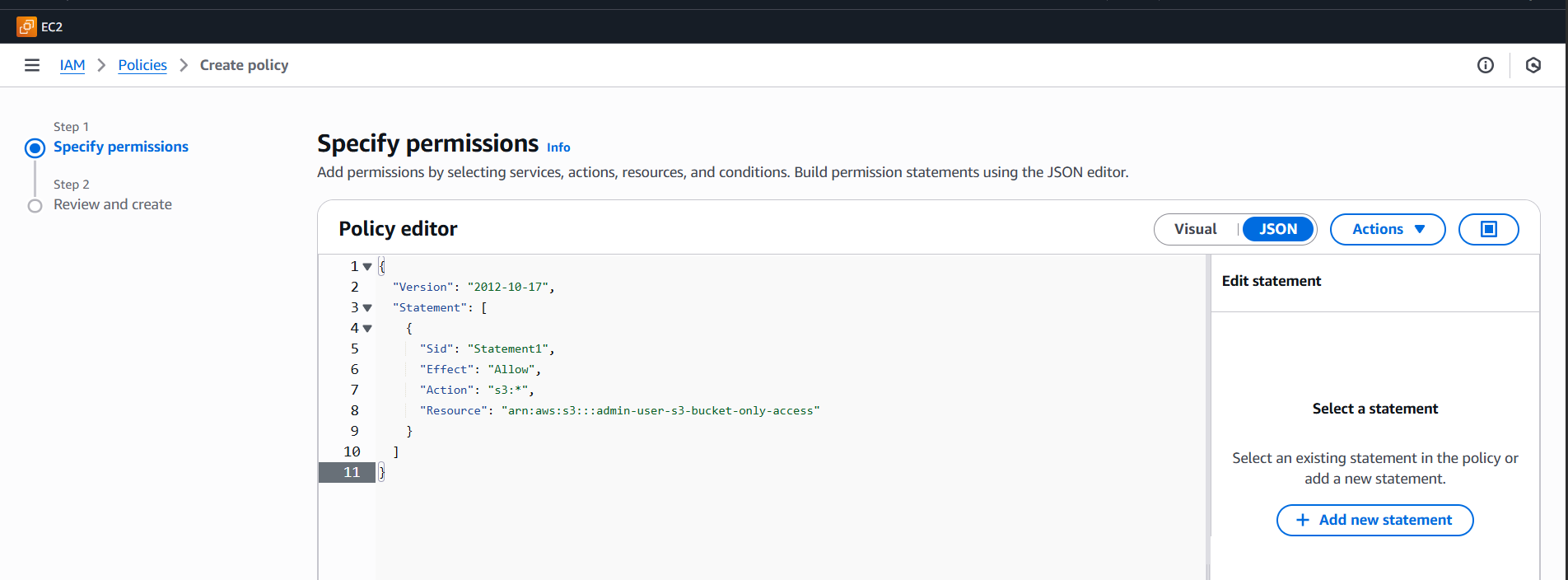
Give these details and paste the ARN id



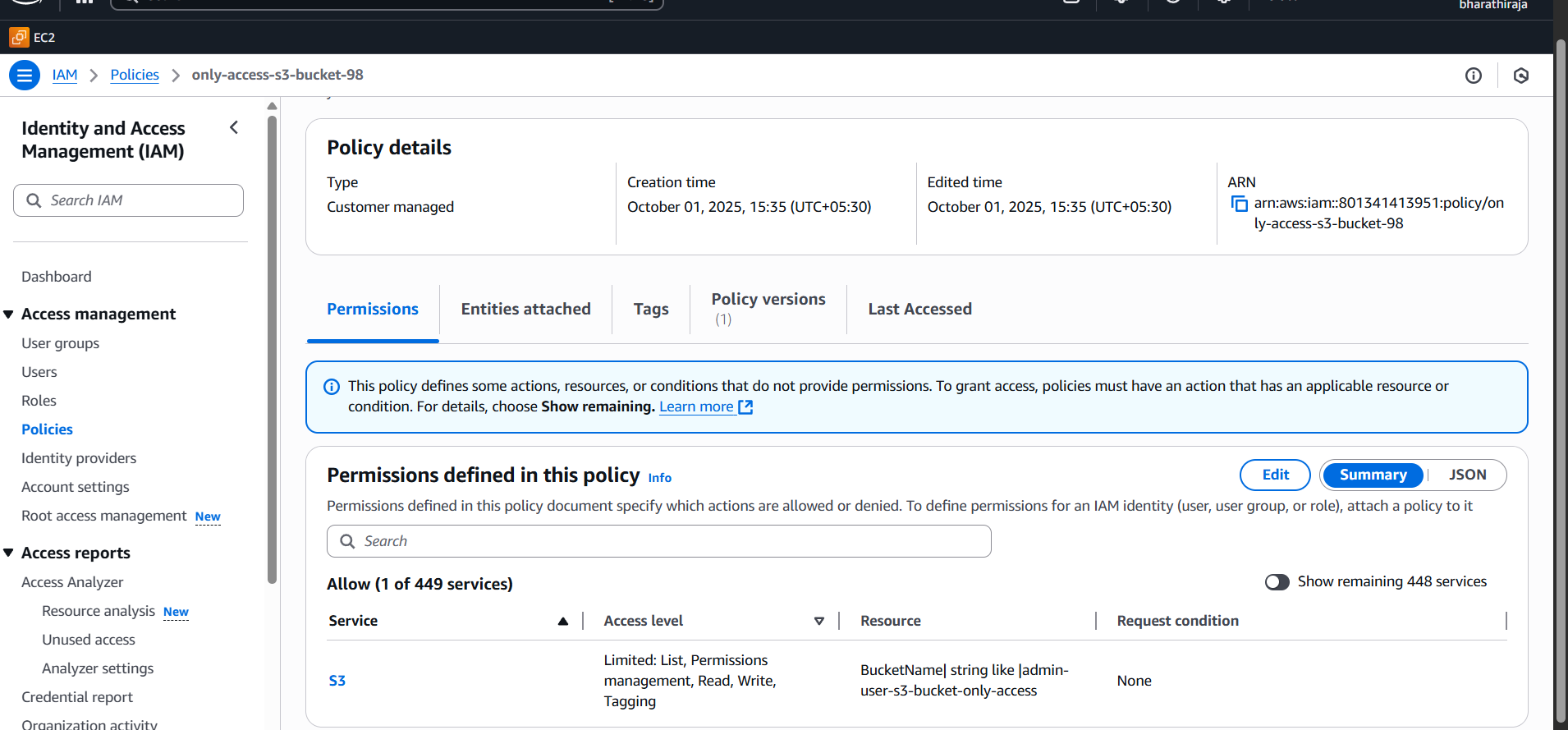
Give an add statement policy will generate



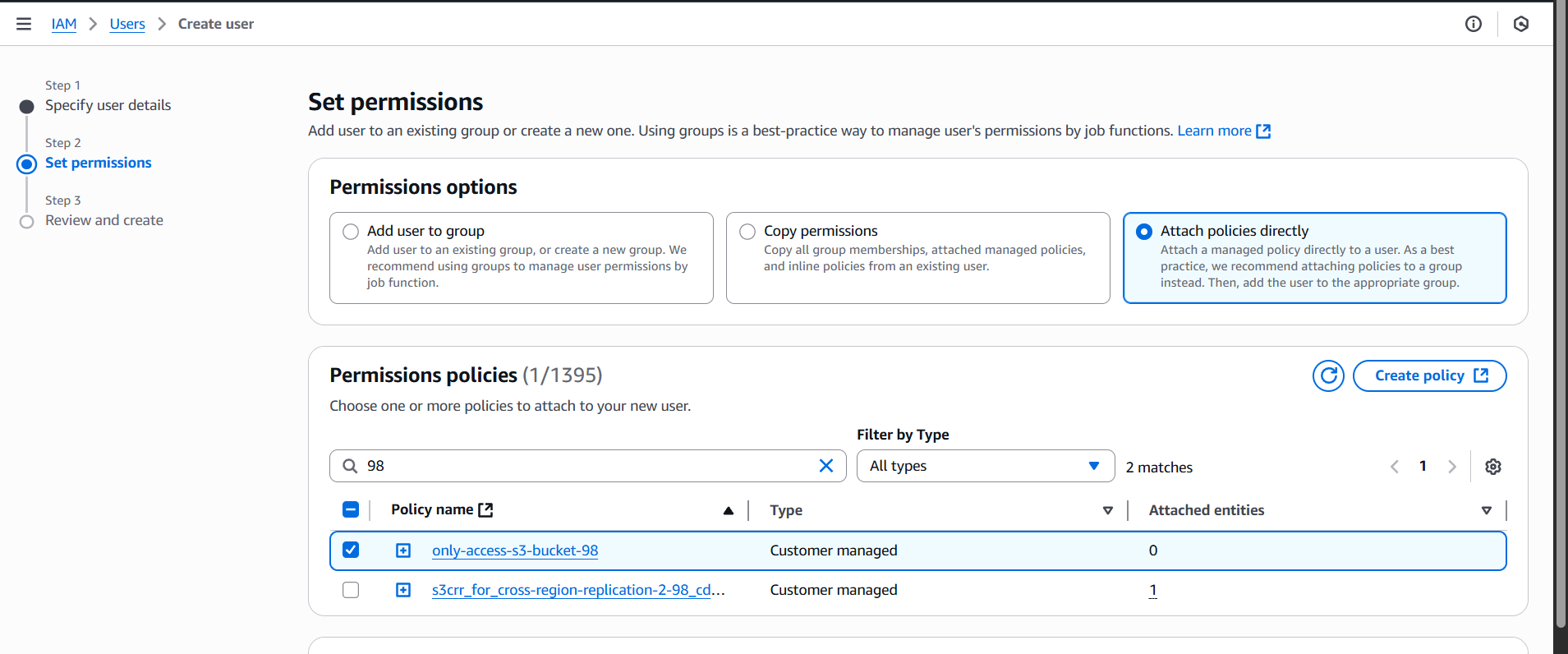
Create one policy copy and paste it



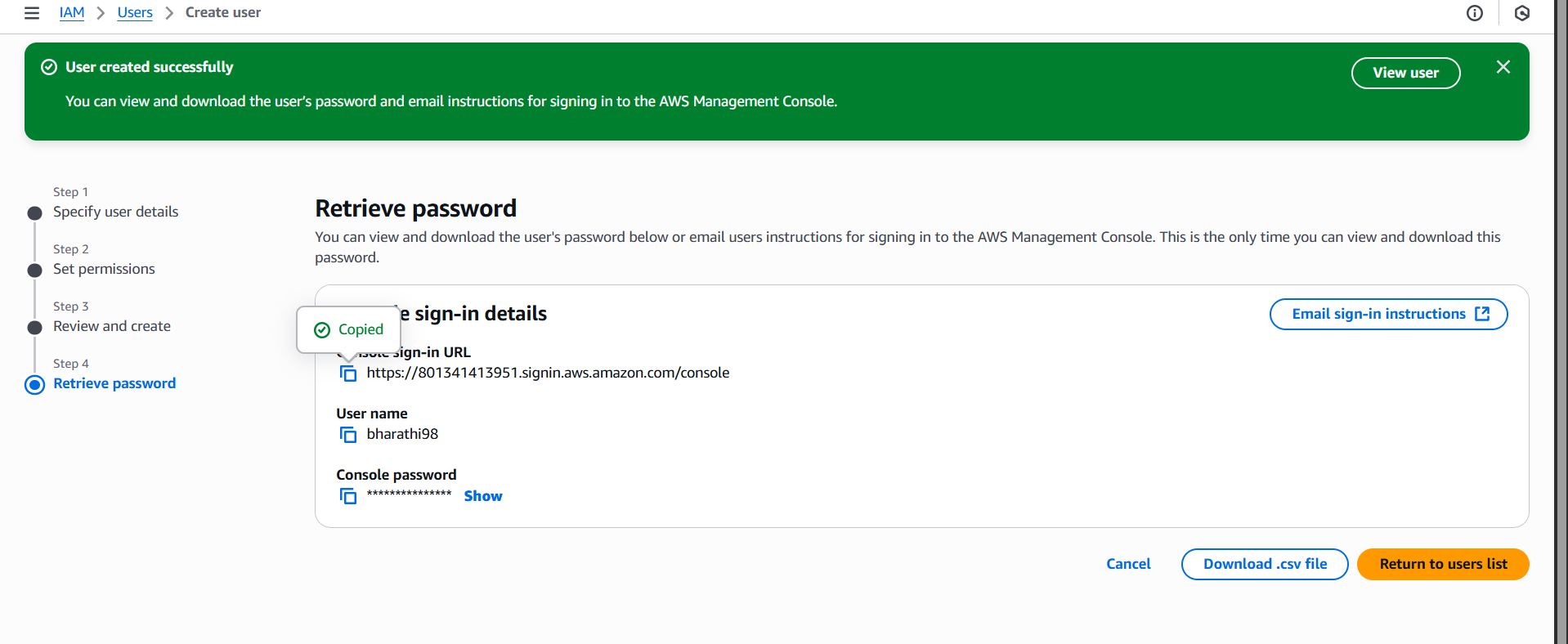
Policy has been created successfully



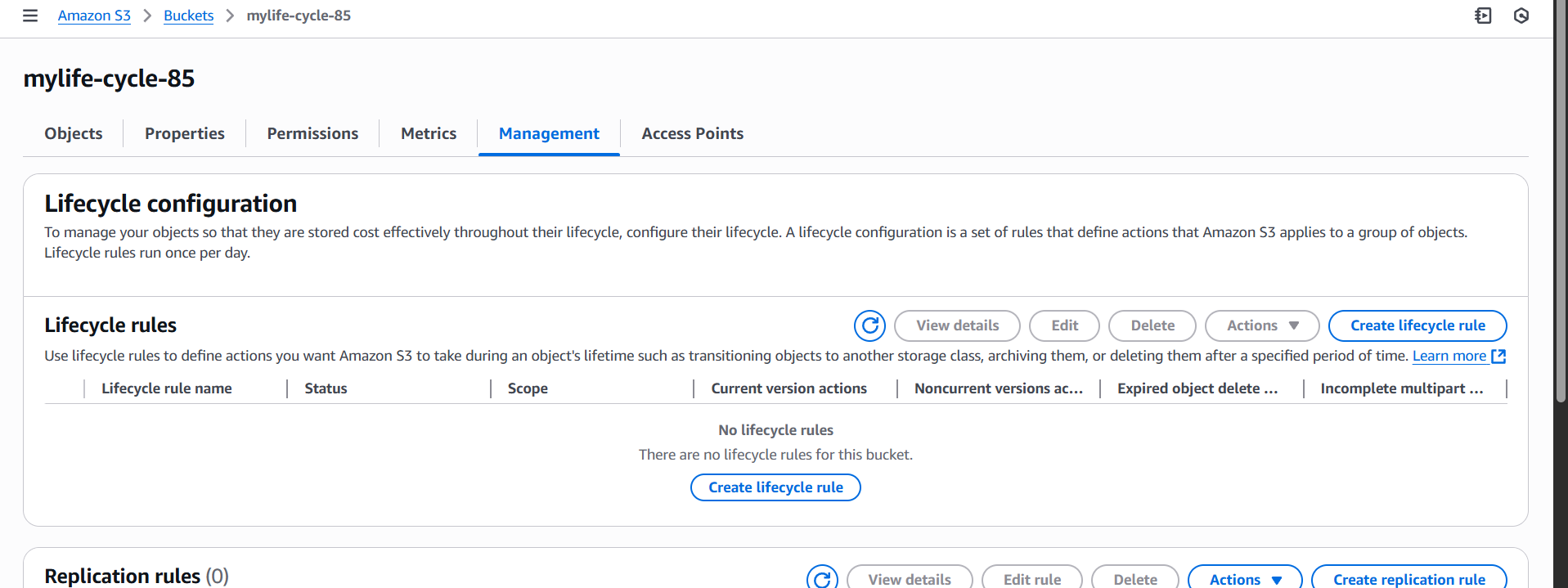
Create a user and add policy



Copy url paste in browser and give user id password

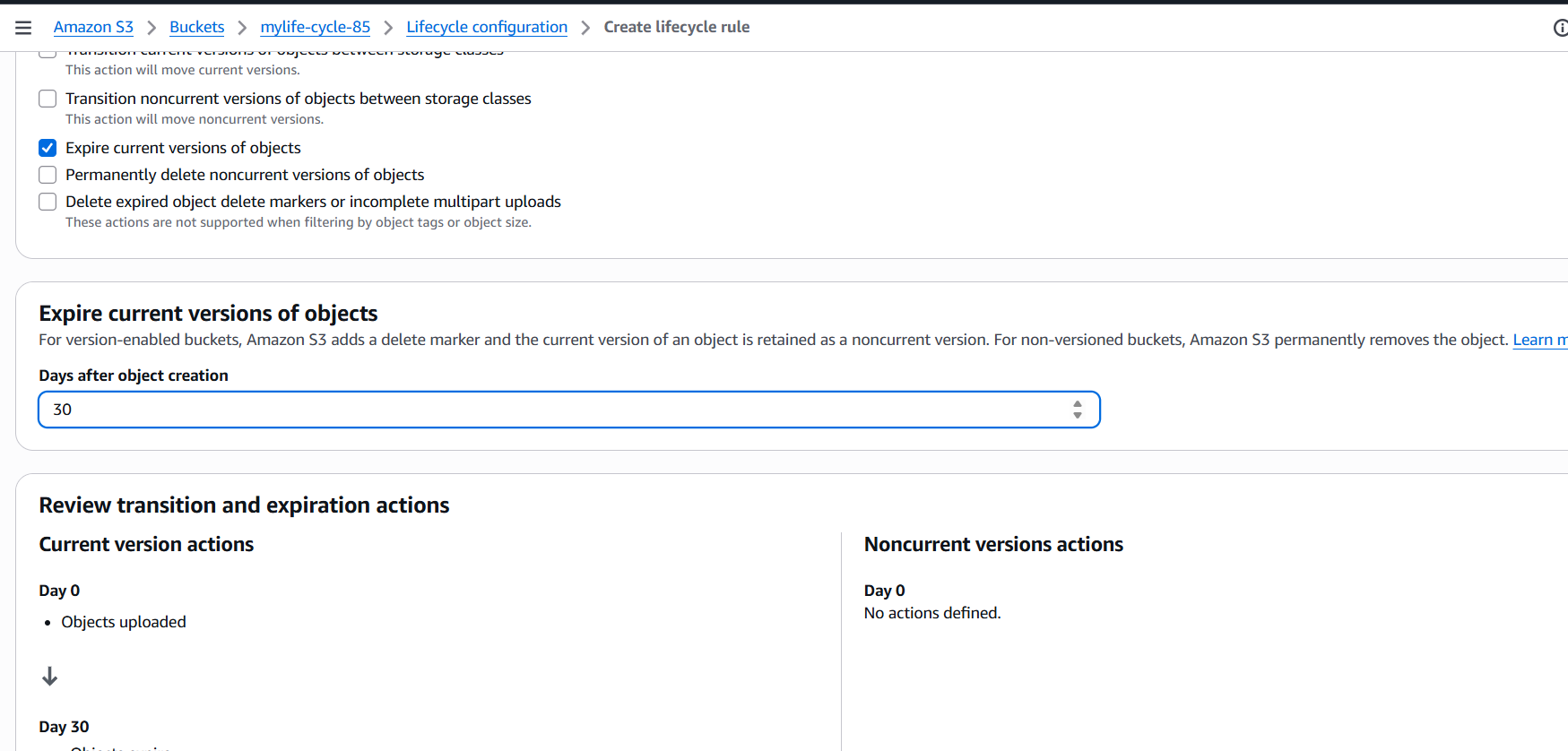


1. Set up lifecycle policies to automatically transition or delete objects based on specific criteria.

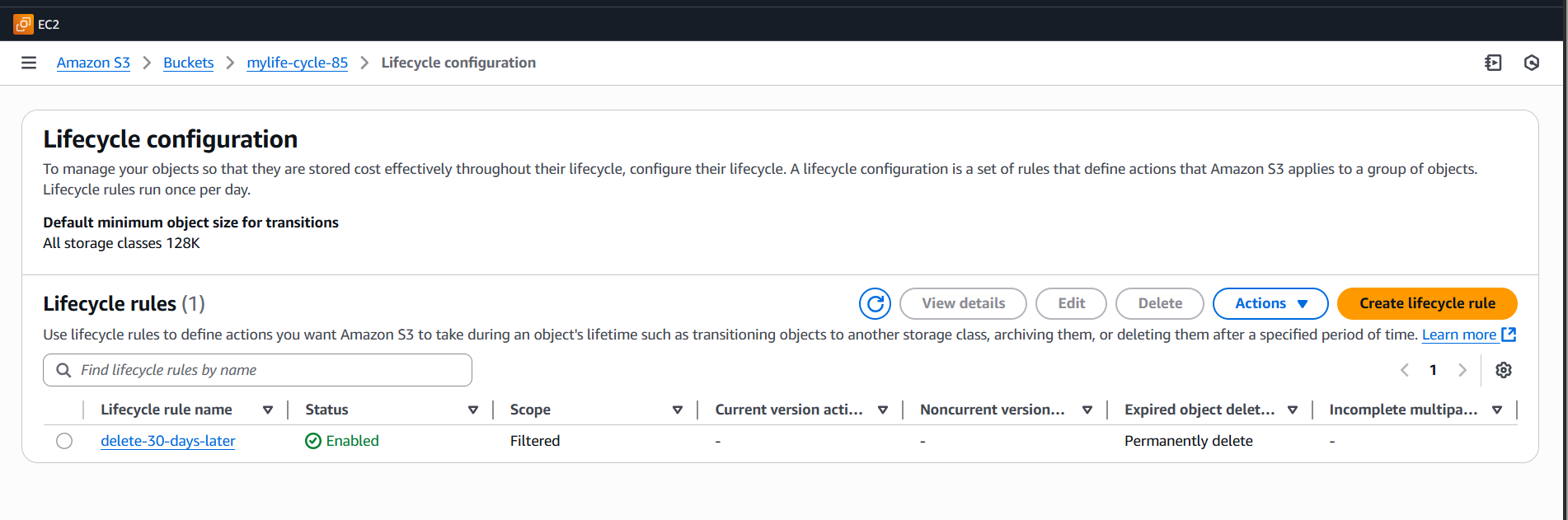


Enter the details criteria





Created a lifecycle rule



1. Push some objects to S3 using the AWS CLI.

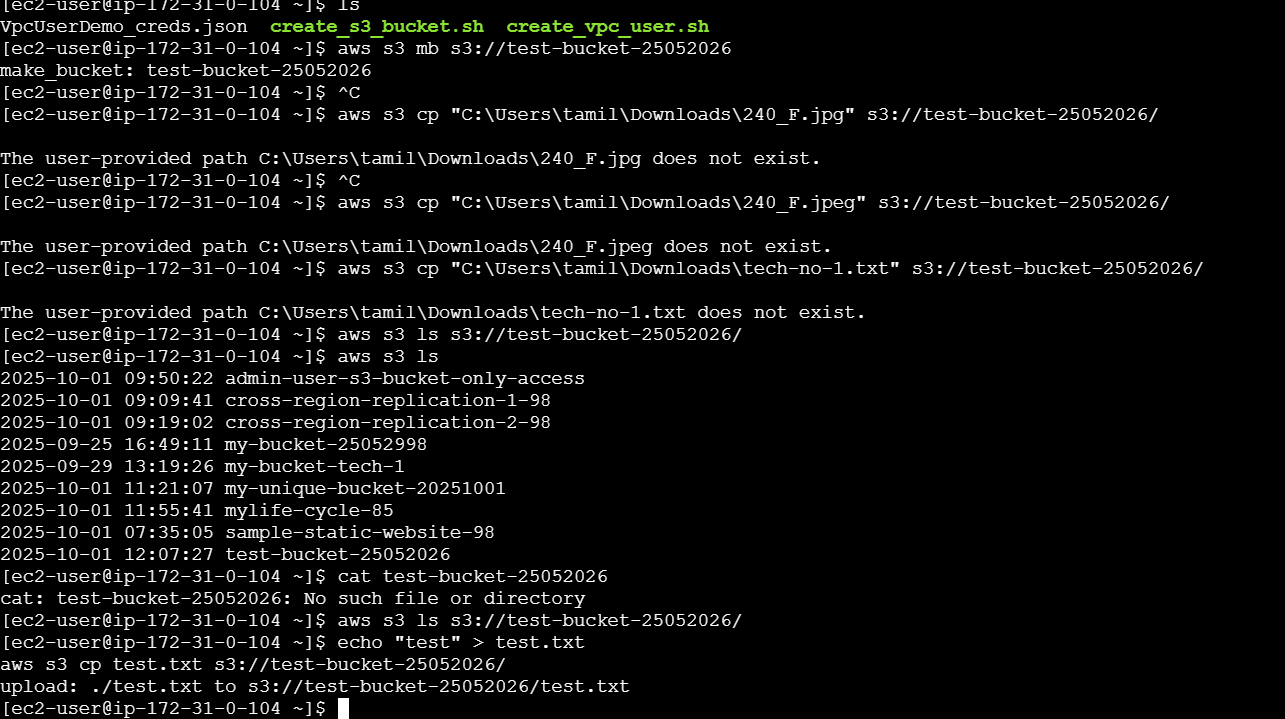
First create a buck use this command

Aws s3 ls s3://test-bucket-25052026  
aws s3 ls view bucket list

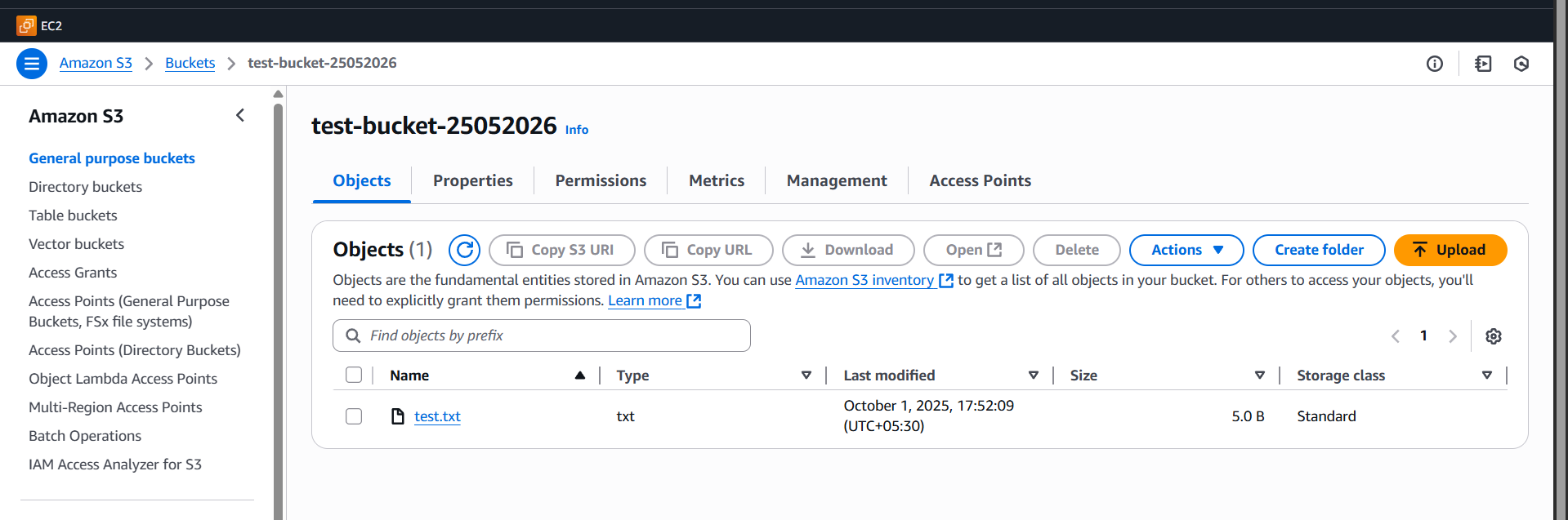
[ec2-user@ip-172-31-0-104 ~]$ echo "test" > test.txt => creta a file

aws s3 cp test.txt s3://test-bucket-25052026/ => moveing to bucket file

upload: ./test.txt to s3://test-bucket-25052026/test.txt this is acknowledges



After checking the bucket, there were files is moving



1. Write a Bash script to create an S3 bucket.

These methods are used to open or create a bucket

First, give aws configure and   
access key and secret key   
region and format   
vi create\_s3\_bucket.sh => file and open write bash scripts and execute the bash scripts

#!/bin/bash

# ========= USER INPUTS =========

BUCKET\_NAME="my-unique-bucket-20251001"

REGION="ap-south-1" # Example: Mumbai region

ACL="private" # Other options: public-read, private, etc.

# ================================

echo "Creating S3 bucket: $BUCKET\_NAME in region: $REGION ..."

# Step 1: Create the bucket

if [ "$REGION" == "us-east-1" ]; then

# In us-east-1 you don't specify LocationConstraint

aws s3api create-bucket --bucket "$BUCKET\_NAME" --acl "$ACL"

else

aws s3api create-bucket --bucket "$BUCKET\_NAME" \

--region "$REGION" \

--create-bucket-configuration LocationConstraint="$REGION" \

--acl "$ACL"

fi

# Step 2: Enable block public access (recommended for security)

aws s3api put-public-access-block \

--bucket "$BUCKET\_NAME" \

--public-access-block-configuration BlockPublicAcls=true,IgnorePublicAcls=true,BlockPublicPolicy=true,RestrictPublicBuckets=true

# Step 3: Verify bucket created

echo "Verifying bucket..."

aws s3 ls | grep "$BUCKET\_NAME"

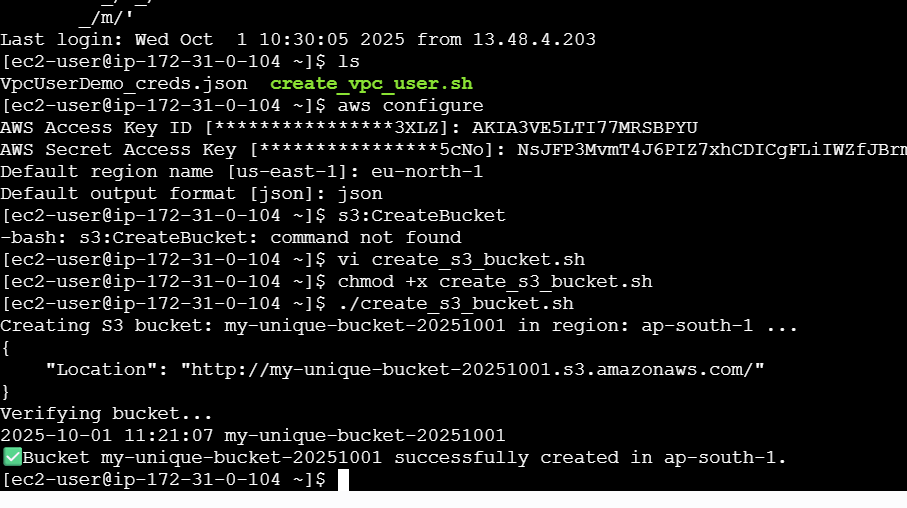
if [ $? -eq 0 ]; then

echo "✅ Bucket $BUCKET\_NAME successfully created in $REGION."

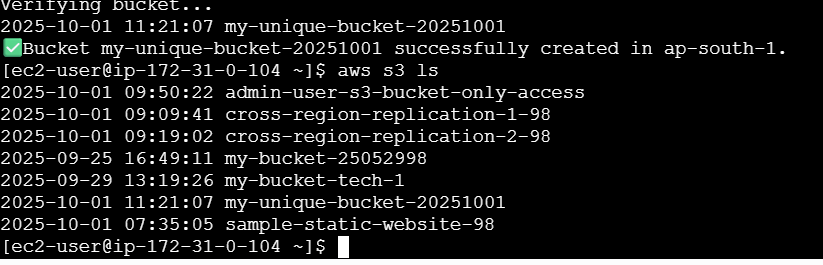
else

echo "❌ Failed to create bucket."

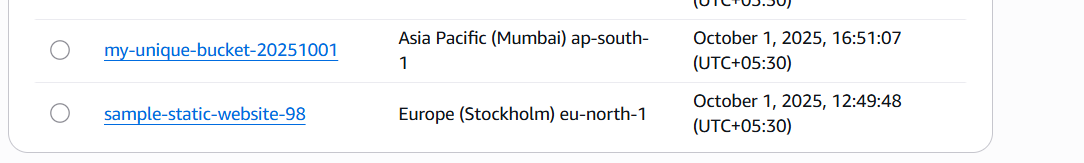
fi



Aws s3 ls => check our bucket list

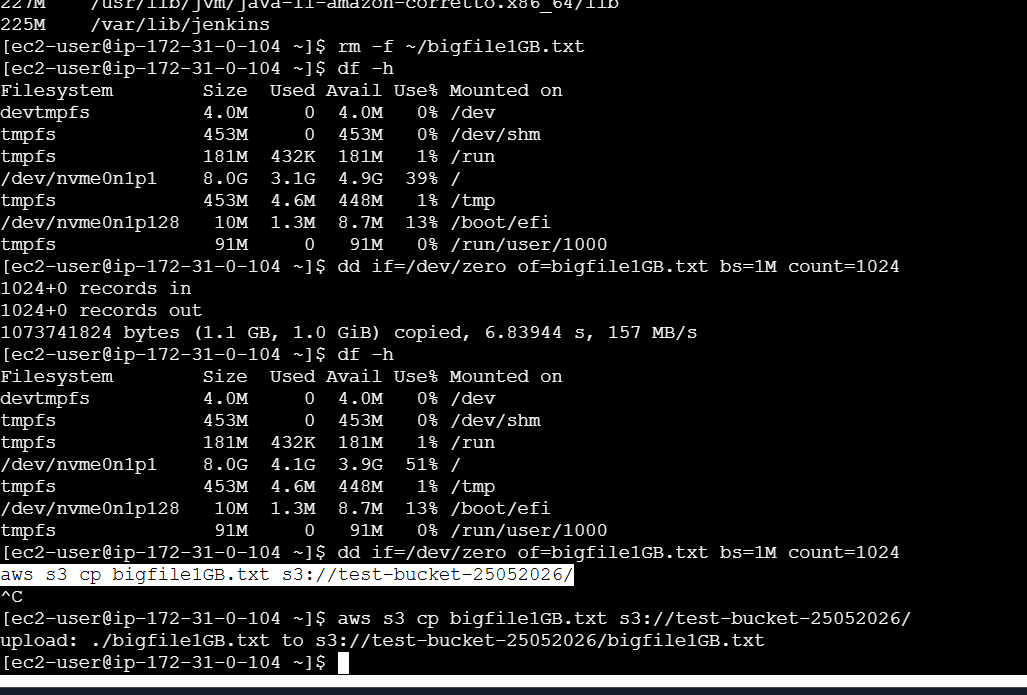


My\_unique-bucket was opened

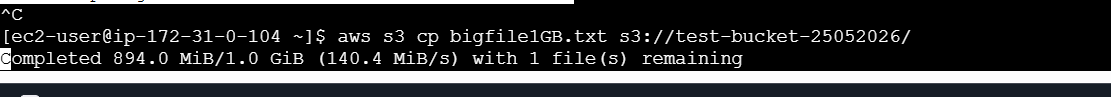


1. Upload a 1 GB file to S3 using the CLI.

go to downloads and open gita bash there.  
¬Then using $ dd if=/dev/zero of=bigfile1GB.txt bs=1M   
count=1024  
¬this command is create a on dummy file for 1 gb   
capacity.  
¬Then check ur bucket list aws s3 ls.



aws s3 cp bigfile1GB.txt s3://test-bucket-25052026 => this command go to upload or else move from here to there   
¬This for verify upload big file.  
¬Here the results are:



We can check 1 GB upload

