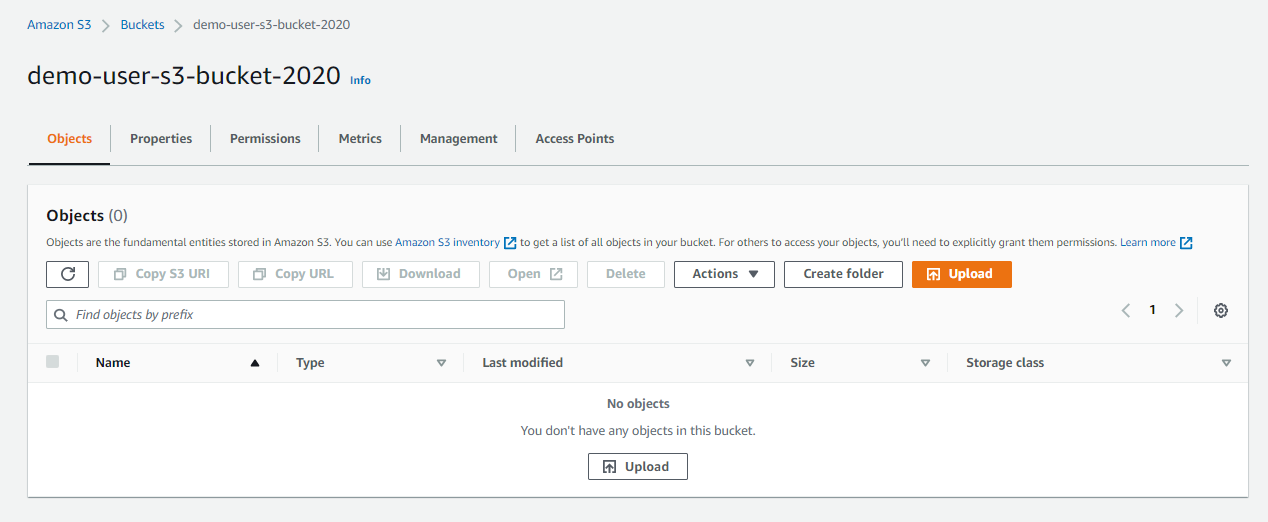
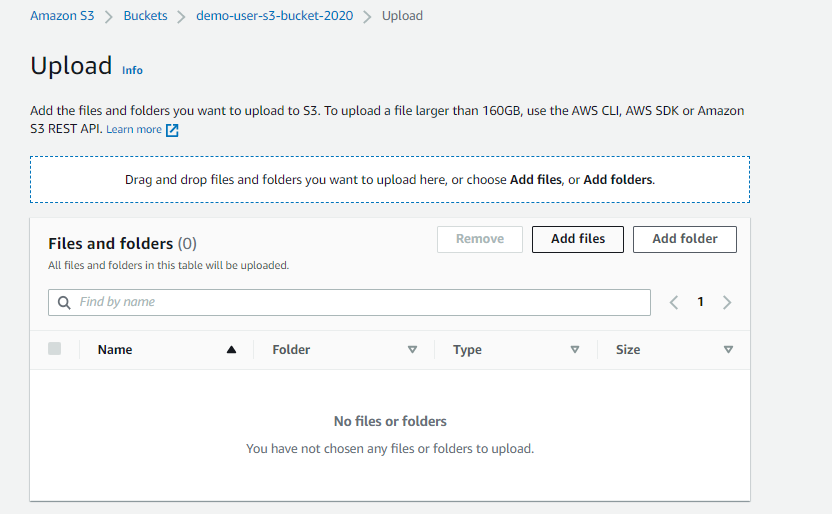
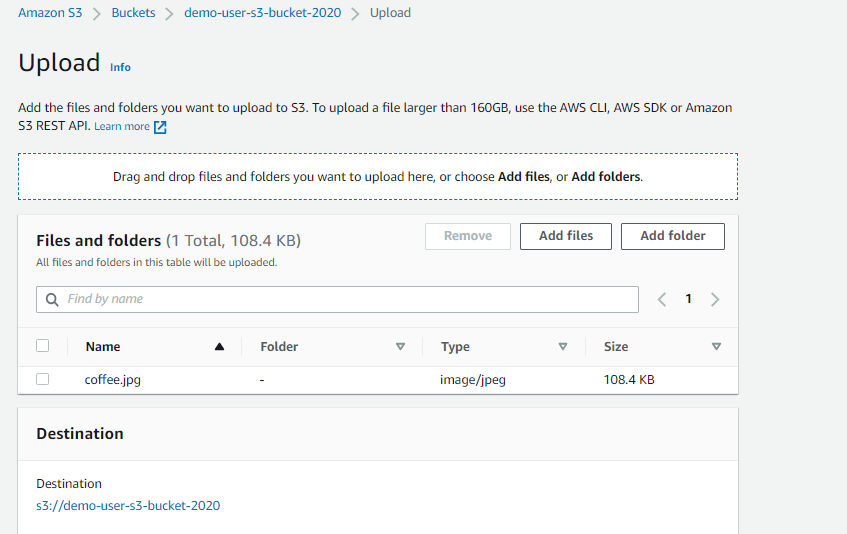
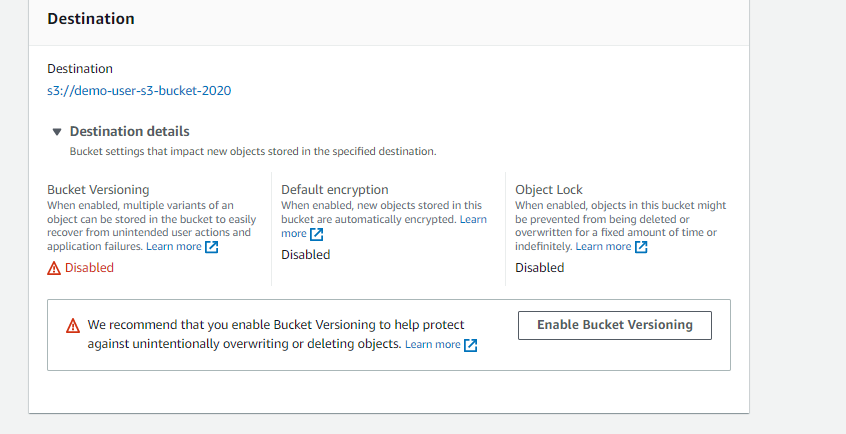


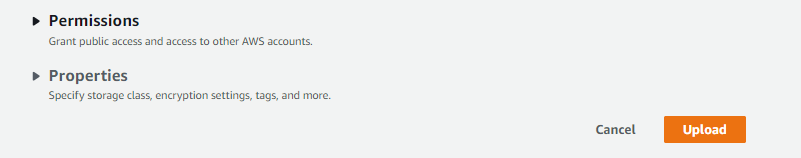
Click your bucket

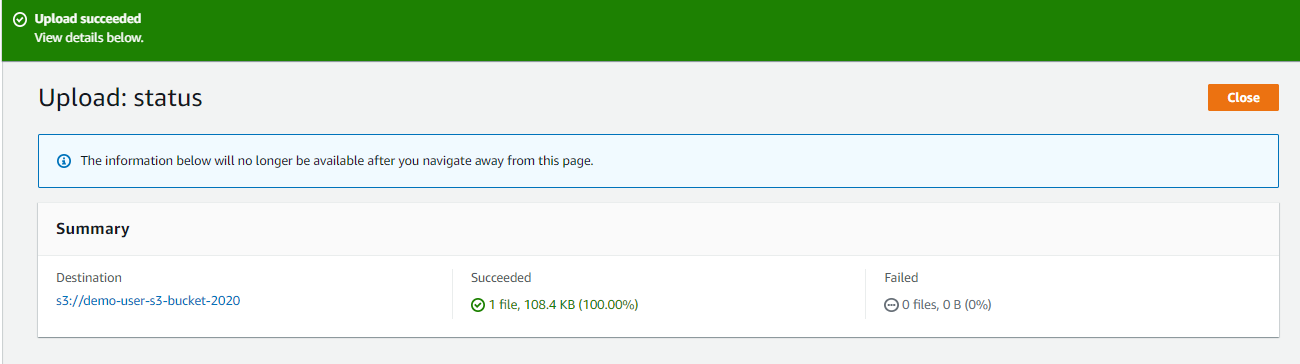


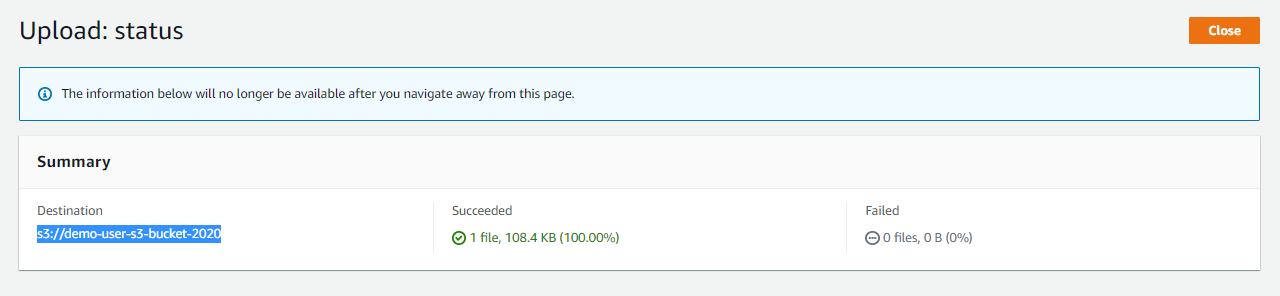


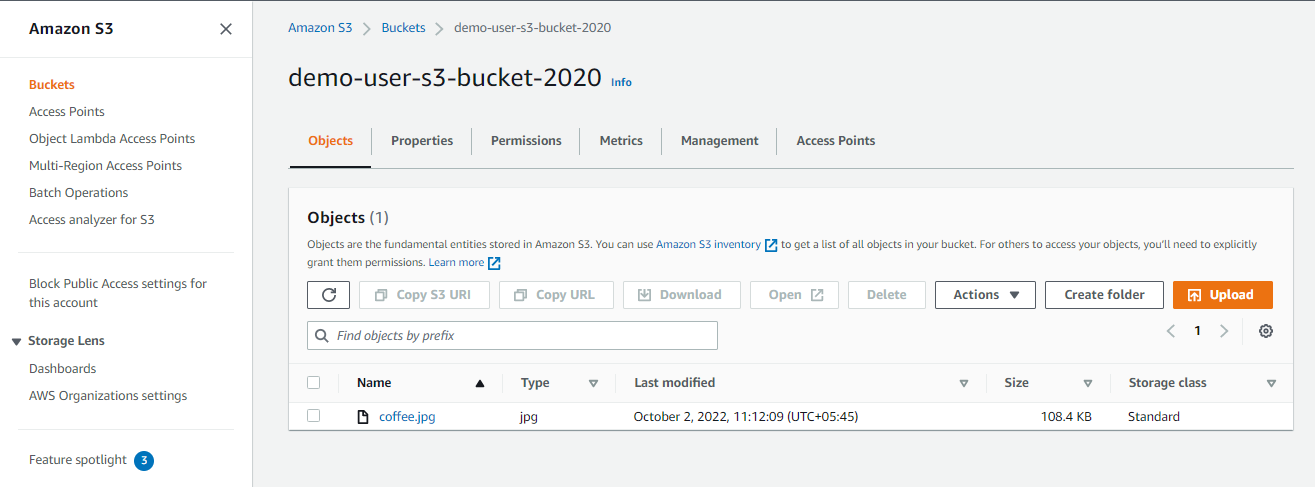




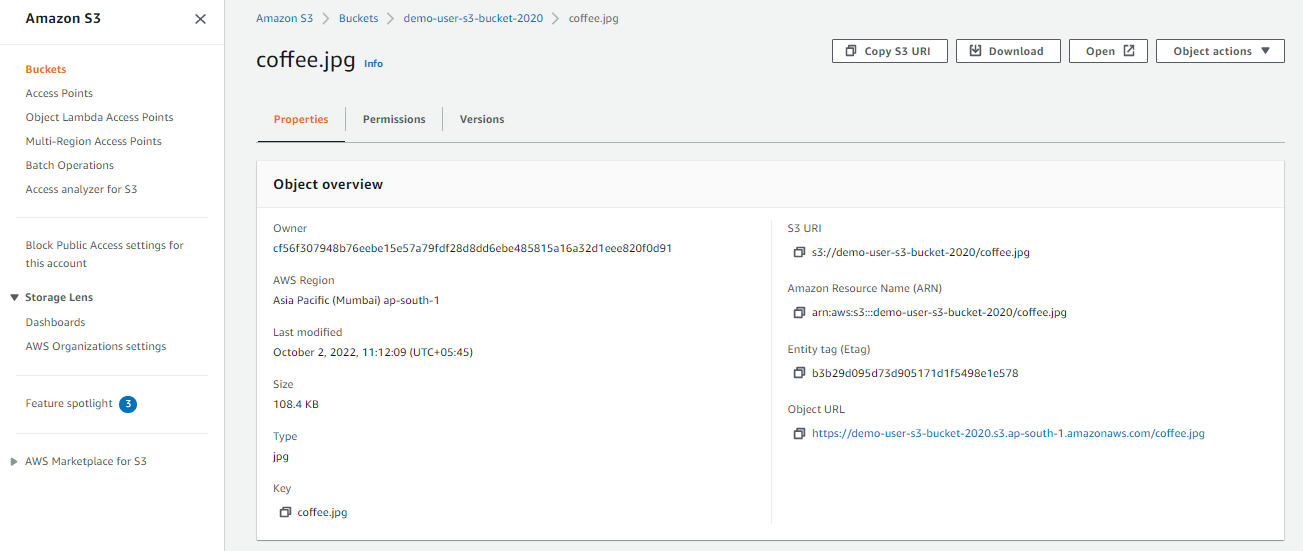


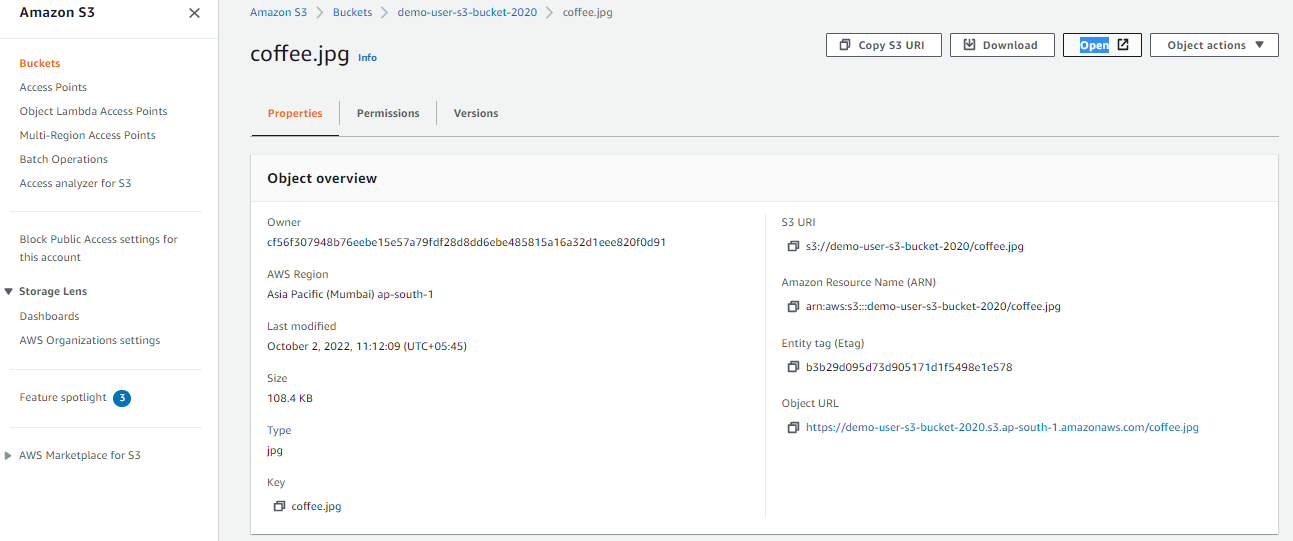




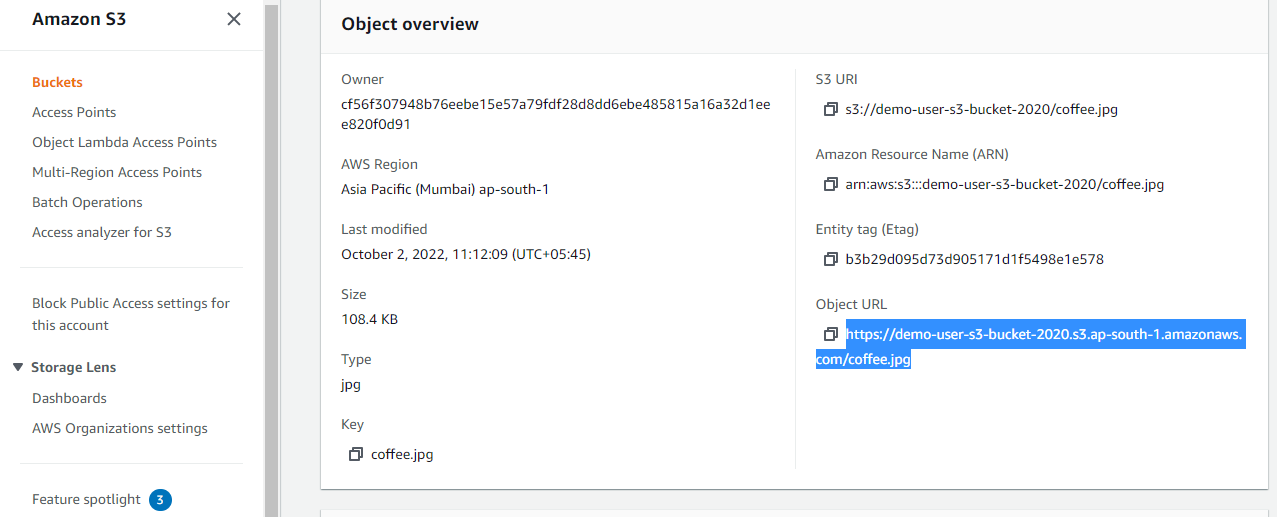


Click open your file coffee.jpg

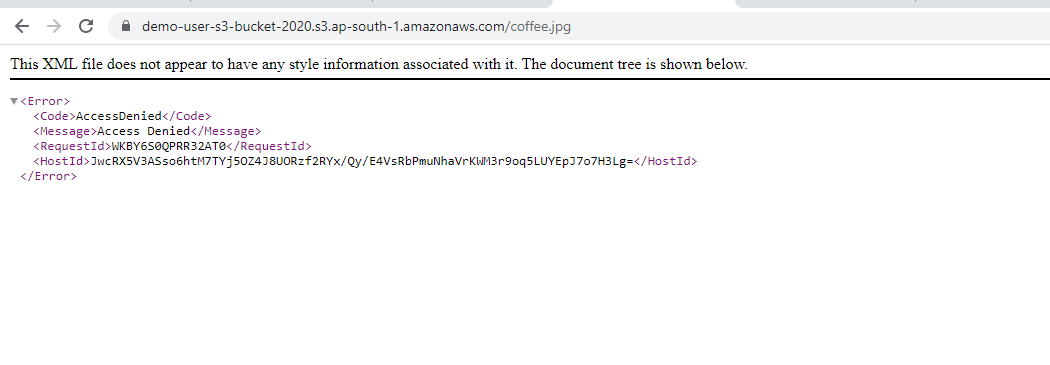




Next way to open s3 file is link object URL

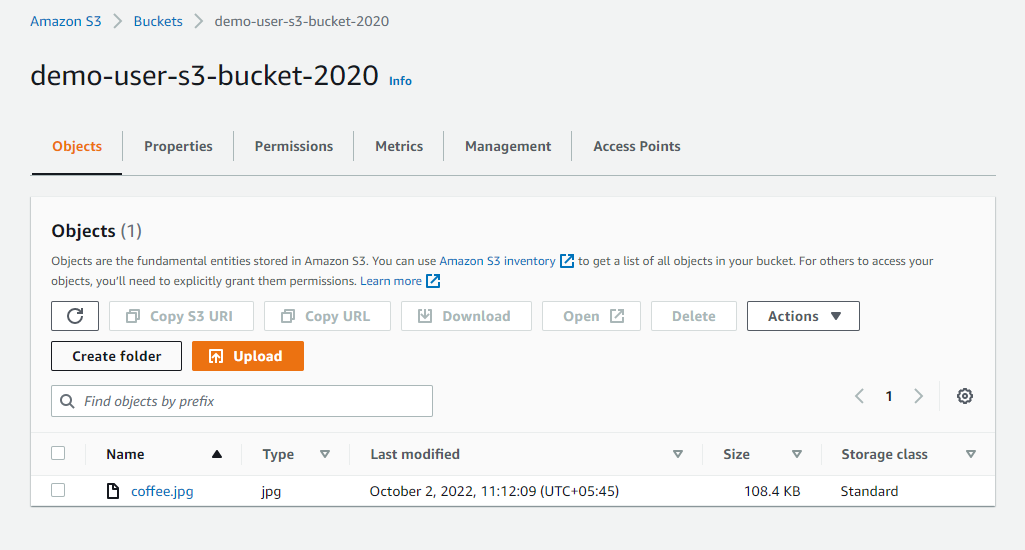


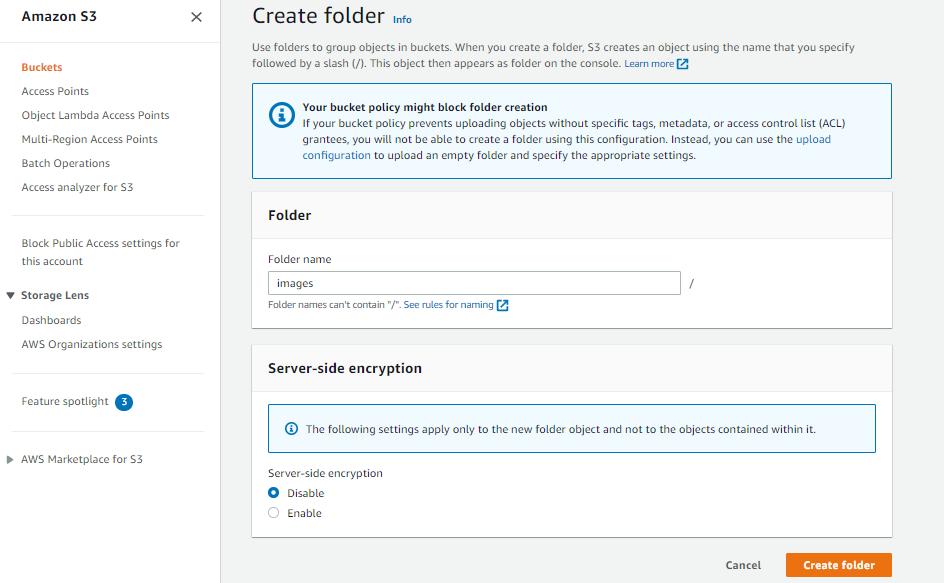
Our s3 file is not public.



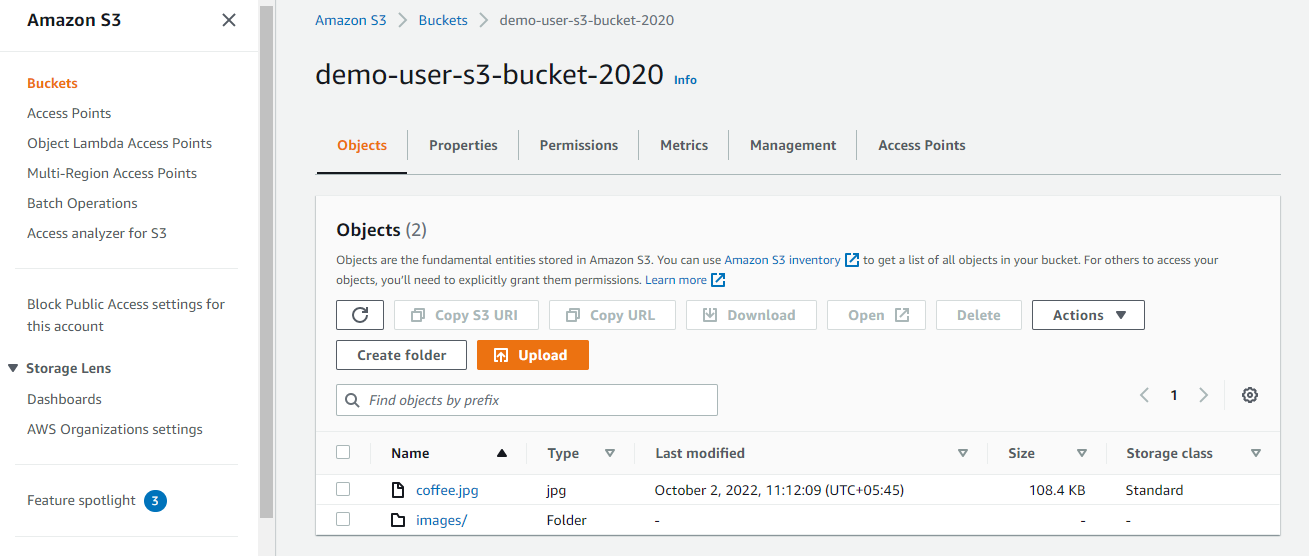
Go back to folder

Create folder



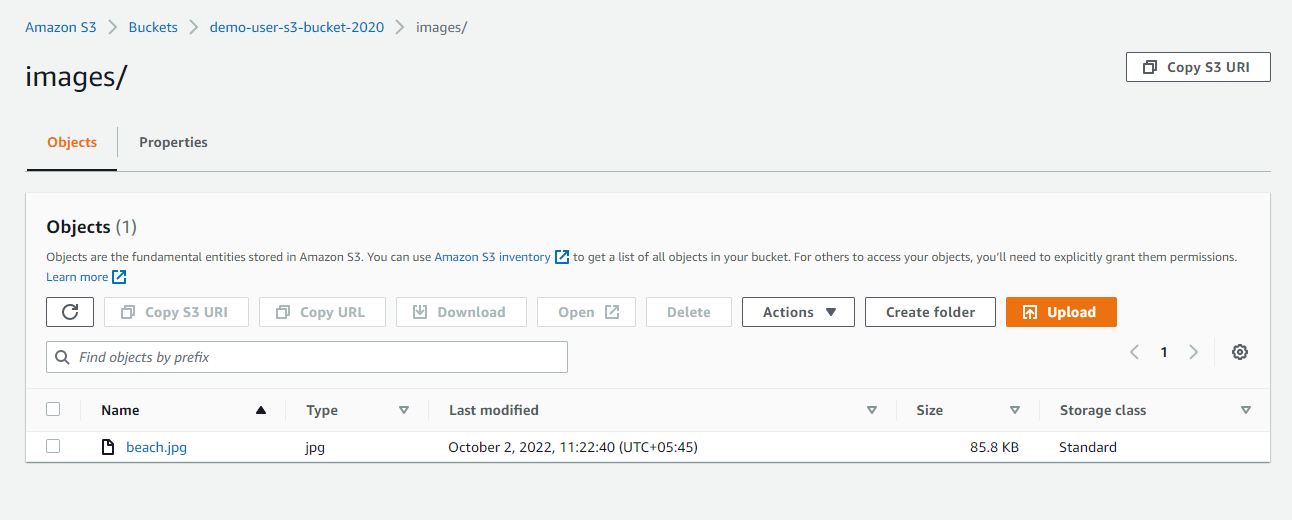


Once folder is created

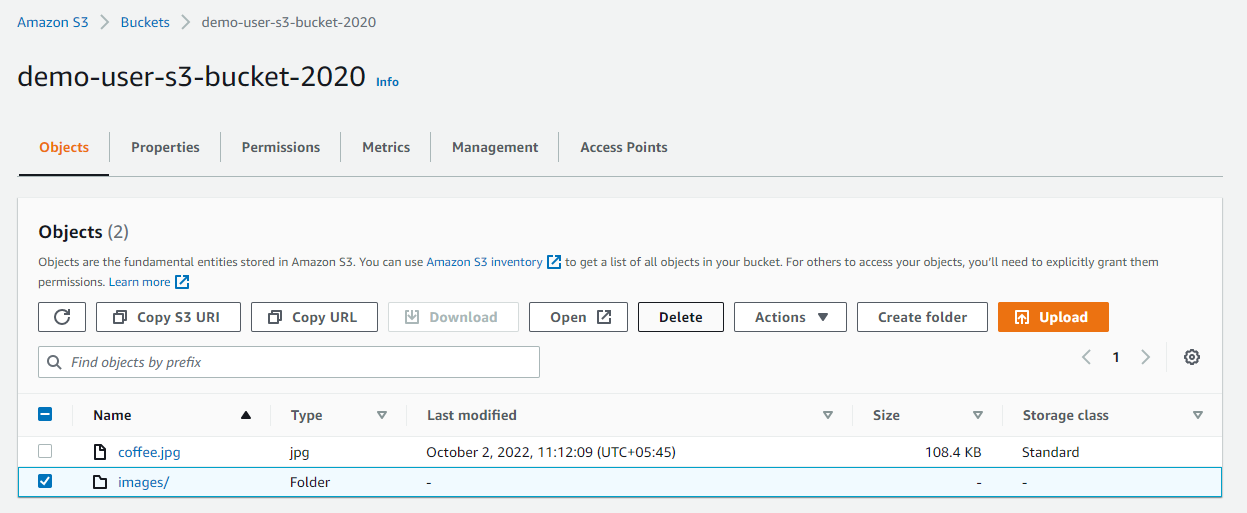


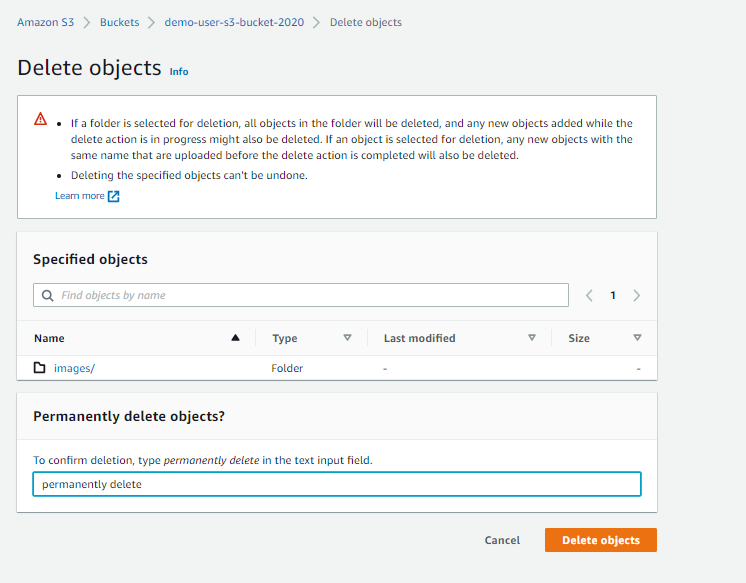
Add files

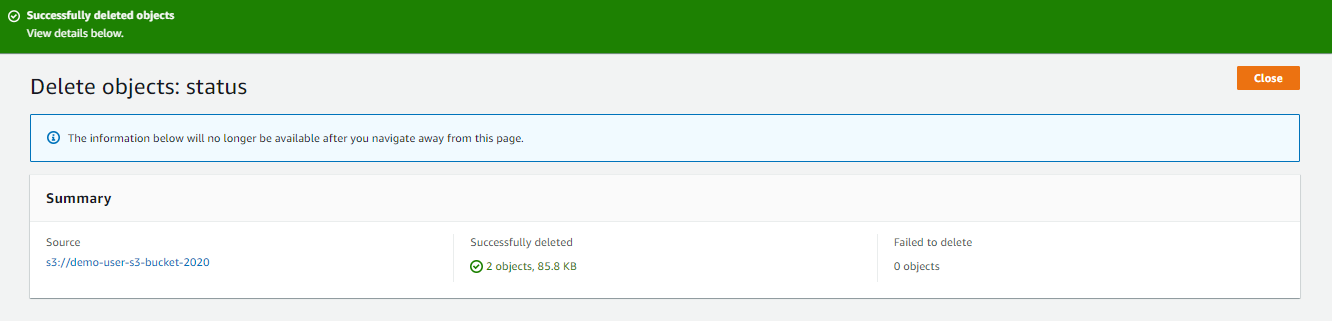




Delete

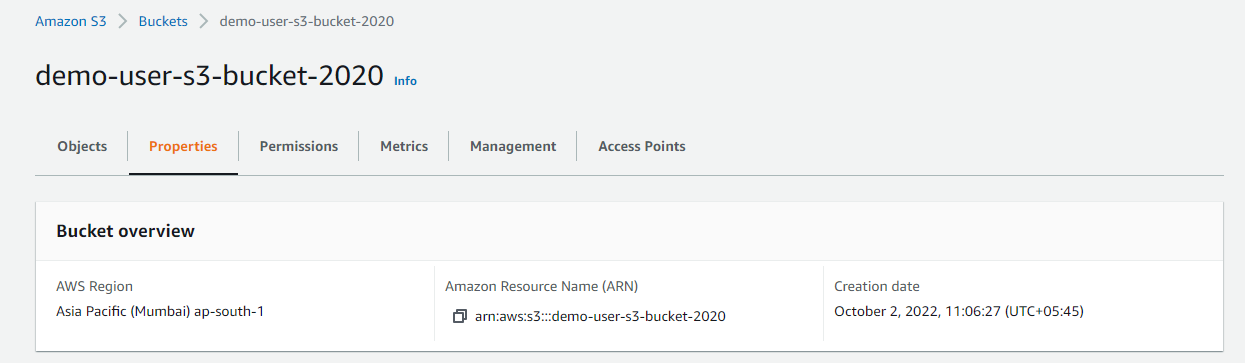


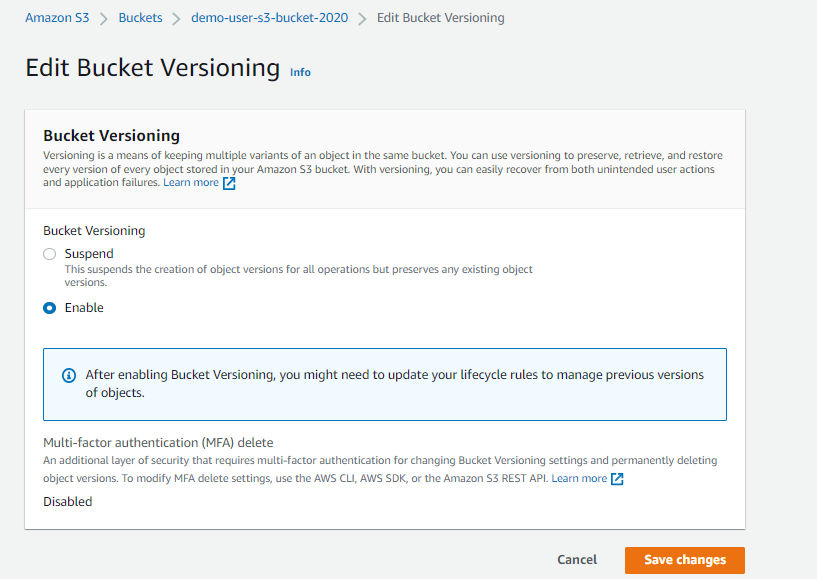


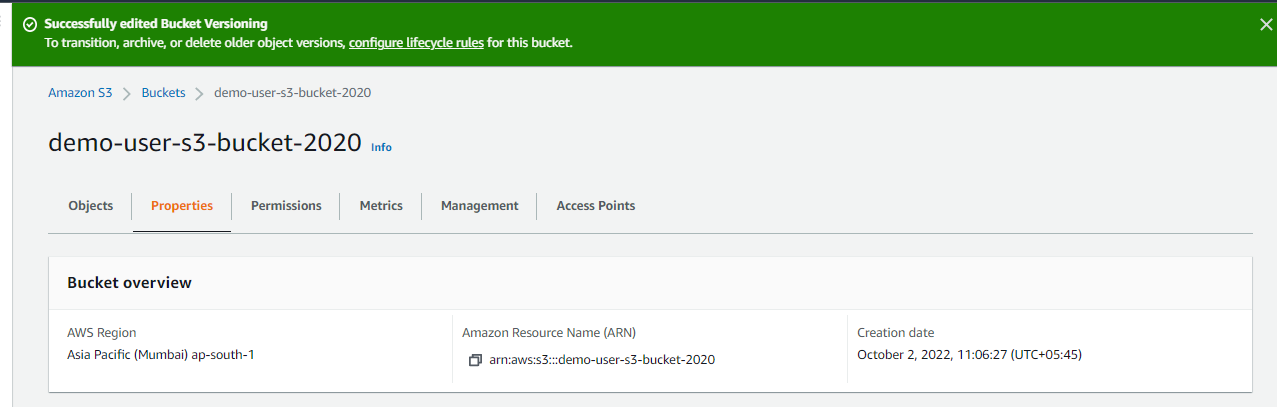


Versioning

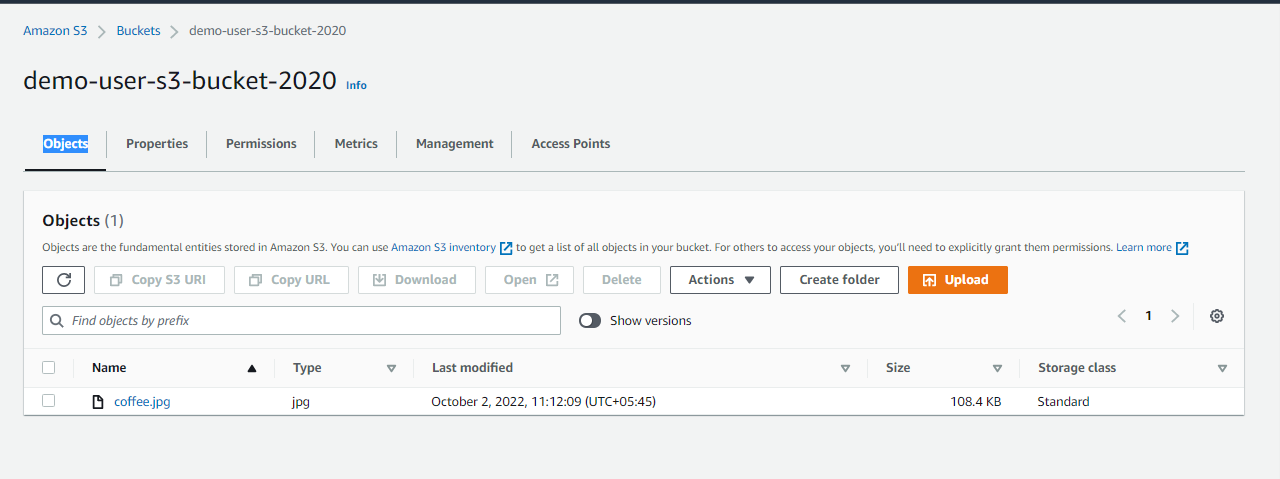
Go to properties on your s3 bucket





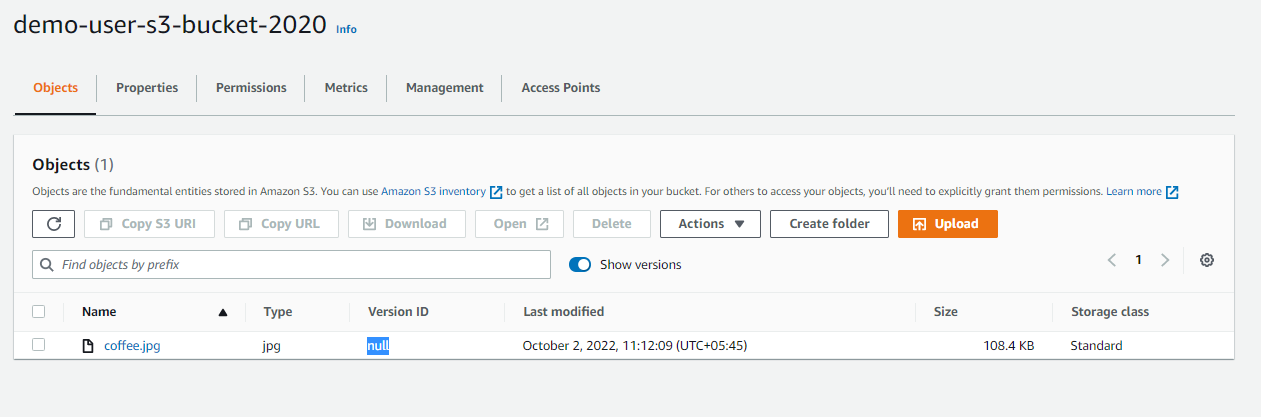


Go to Object

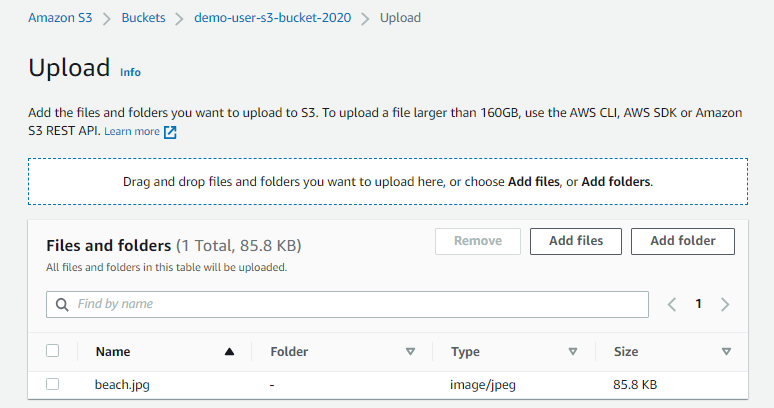


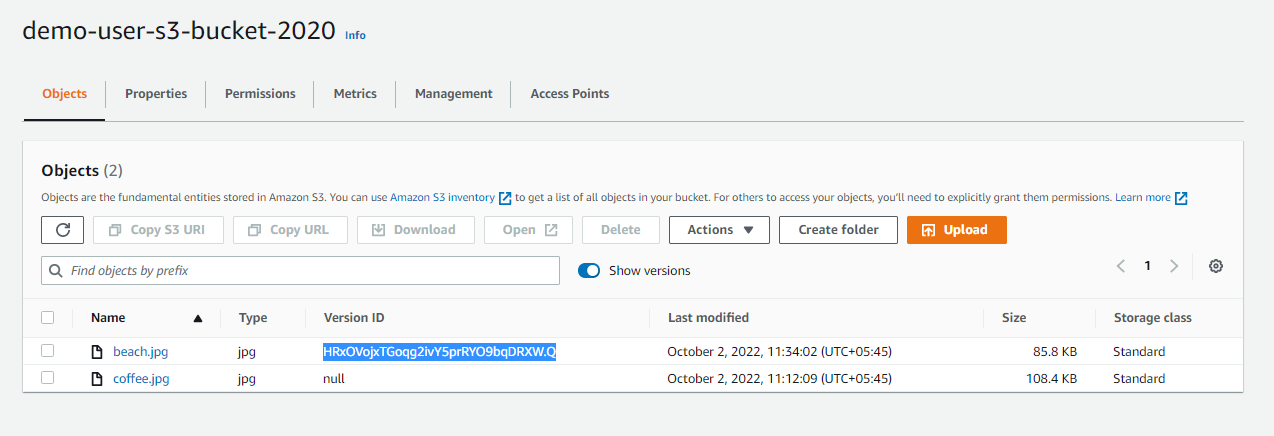
Click show version

This file is uploaded before enabling version



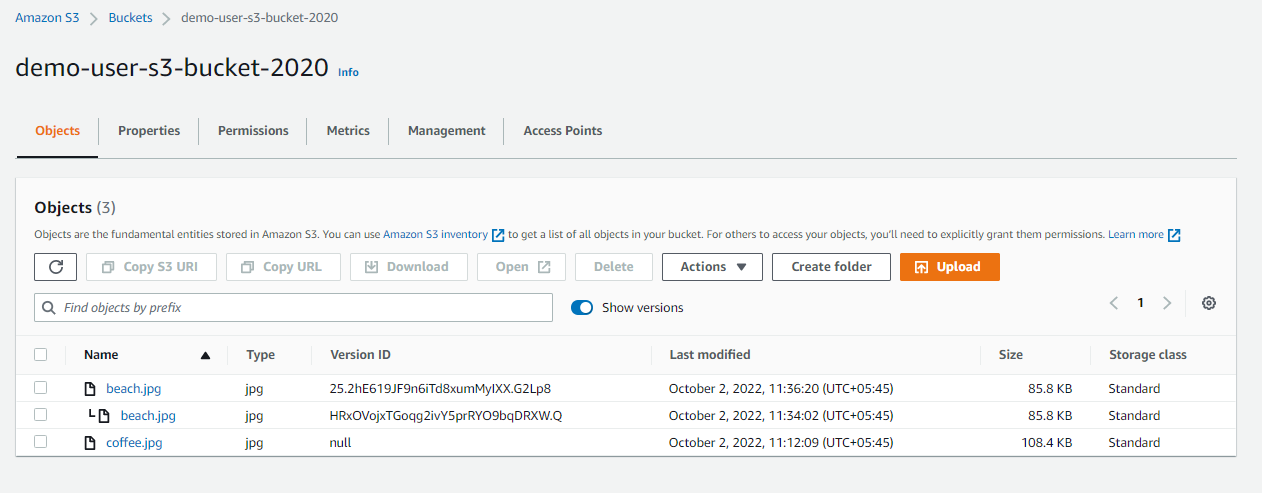
Again upload new file



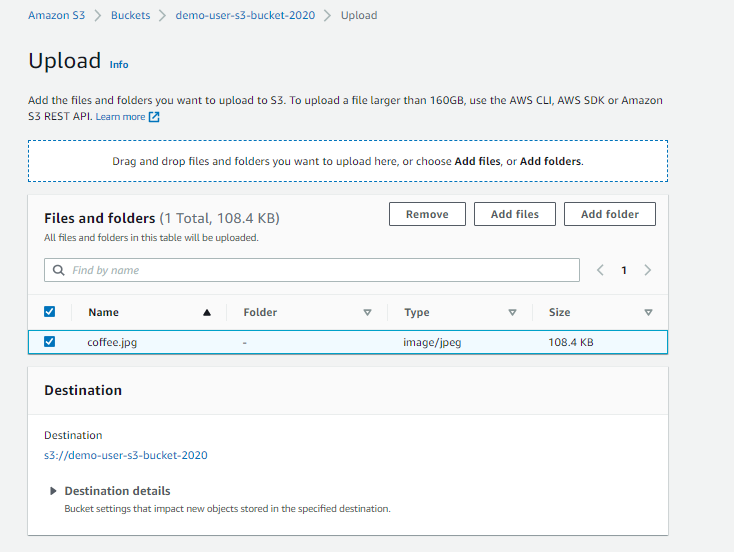


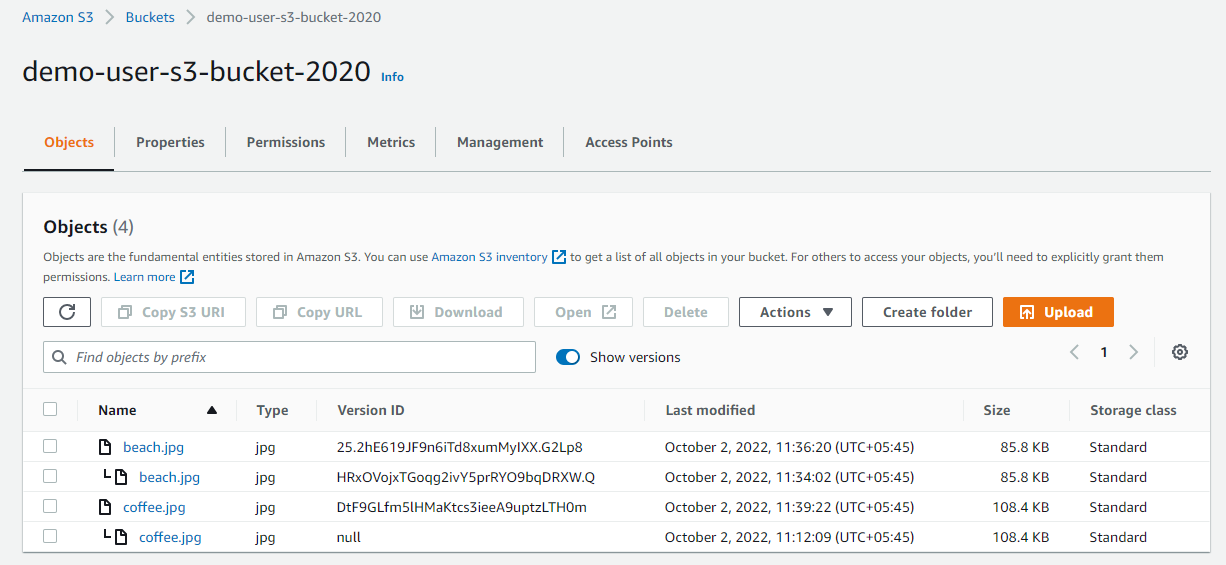
Again upload same file jpg

Once refresh

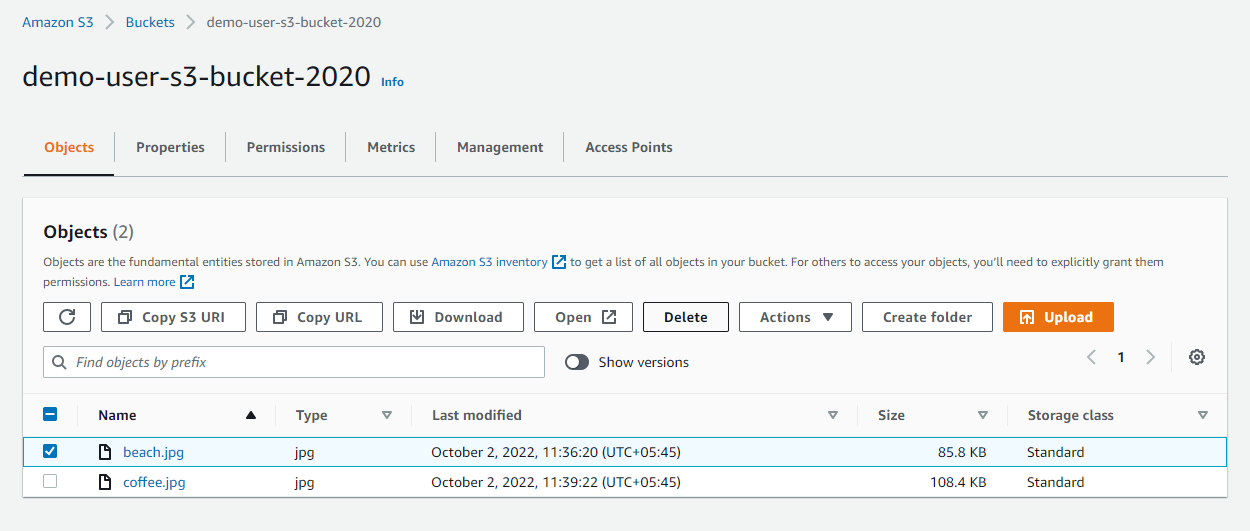


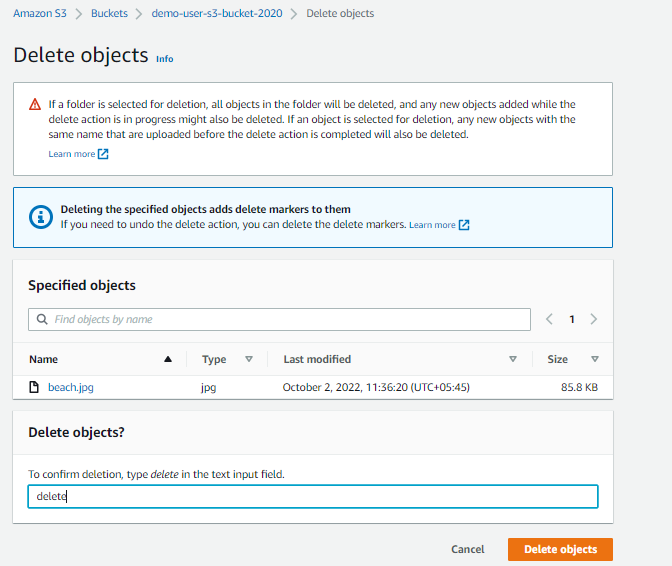
Again upload coffee.jpg

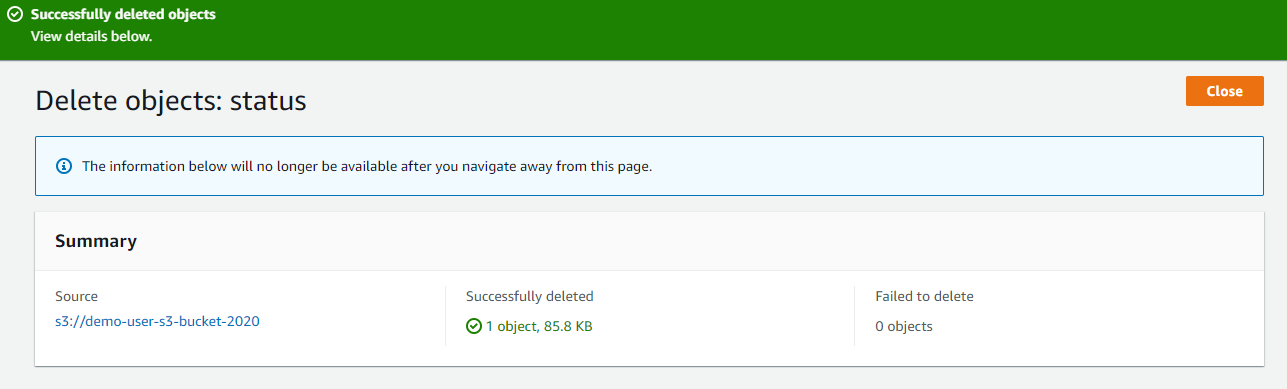




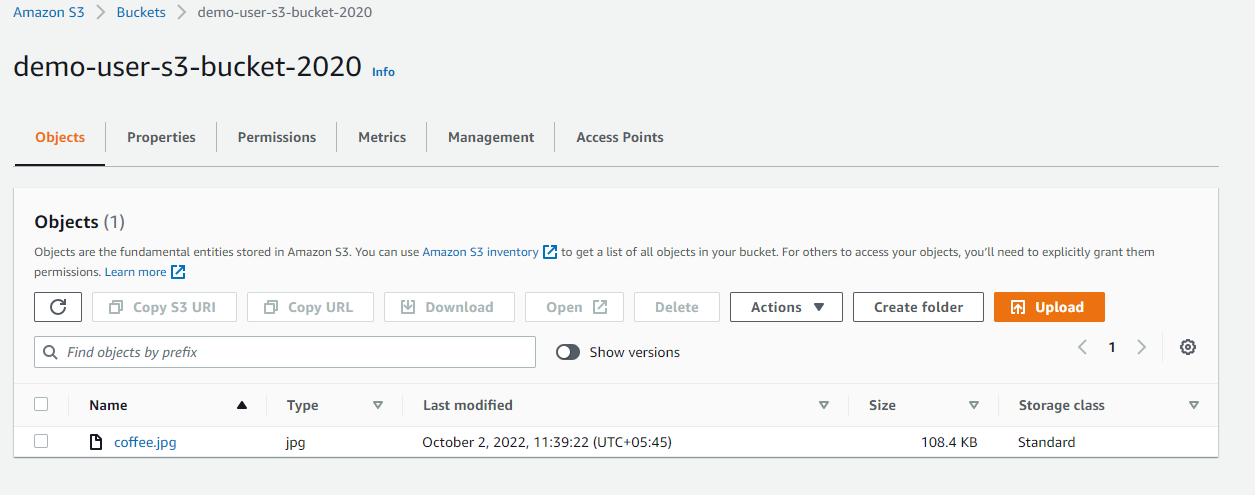
Click delete



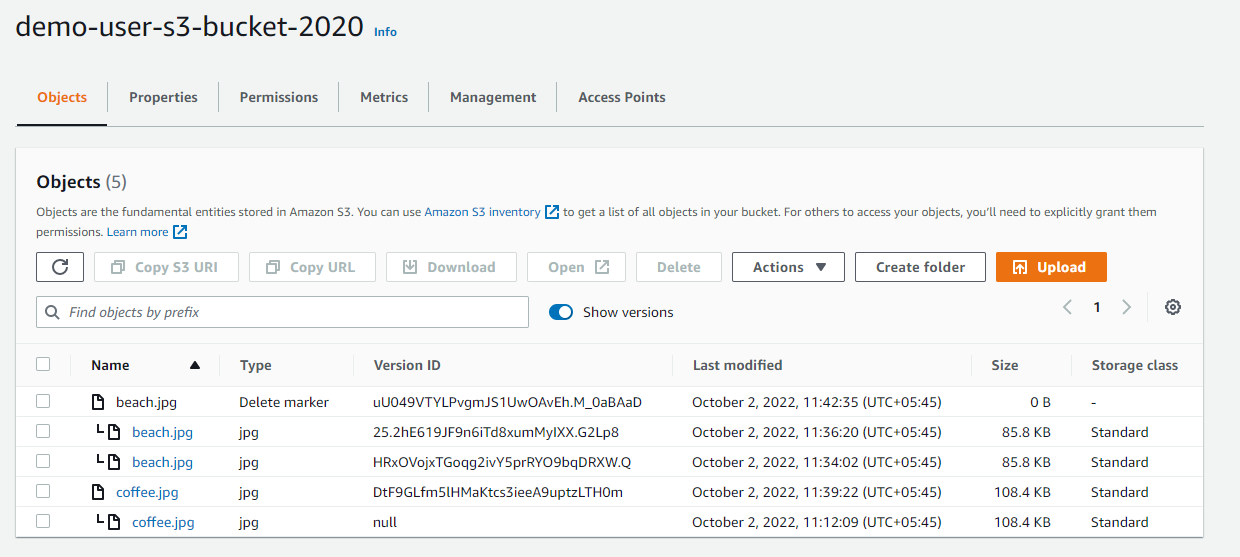




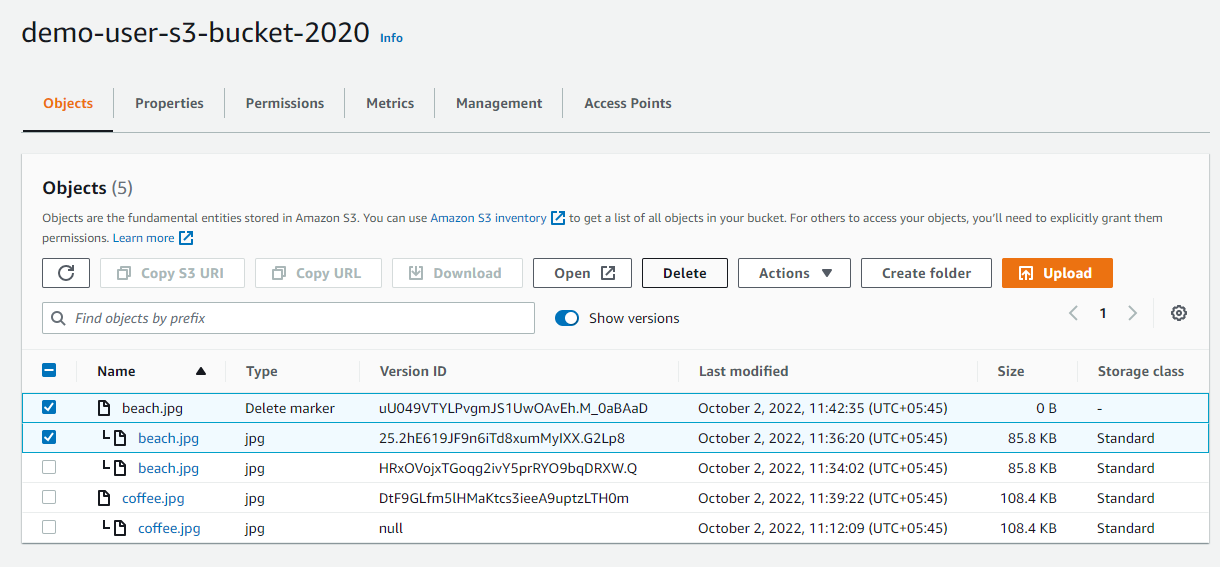
Beach file is deleted

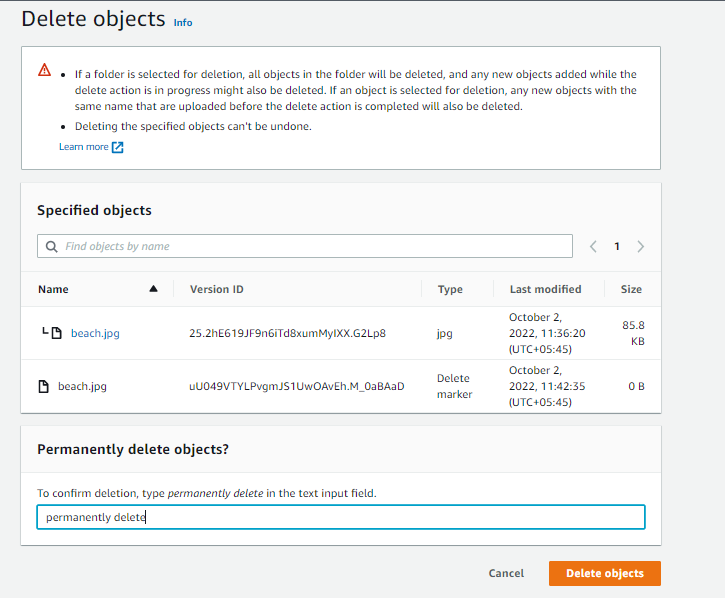


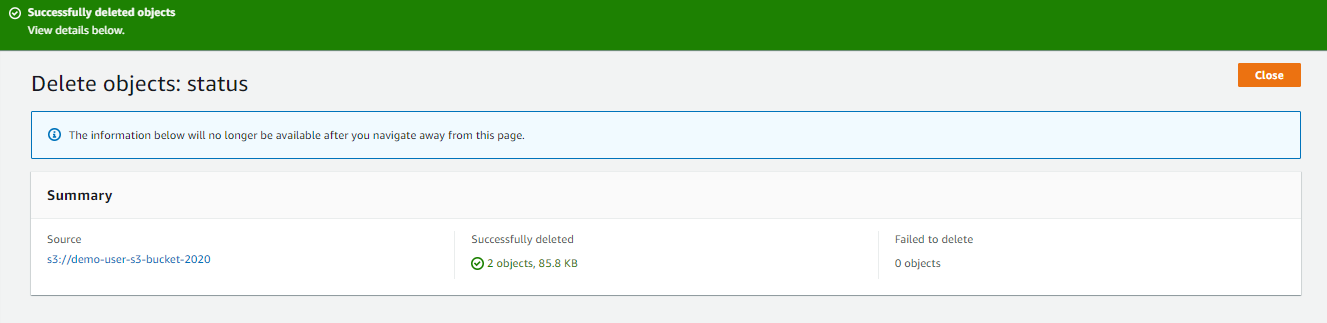
but we enable version



Select delete marker and jpg, click delete

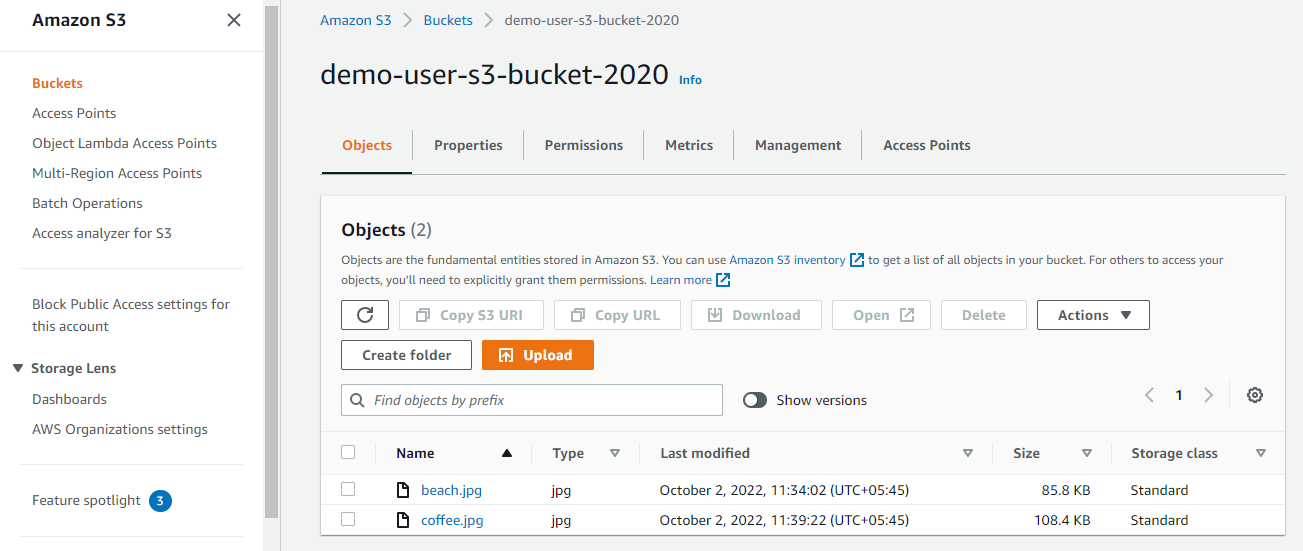




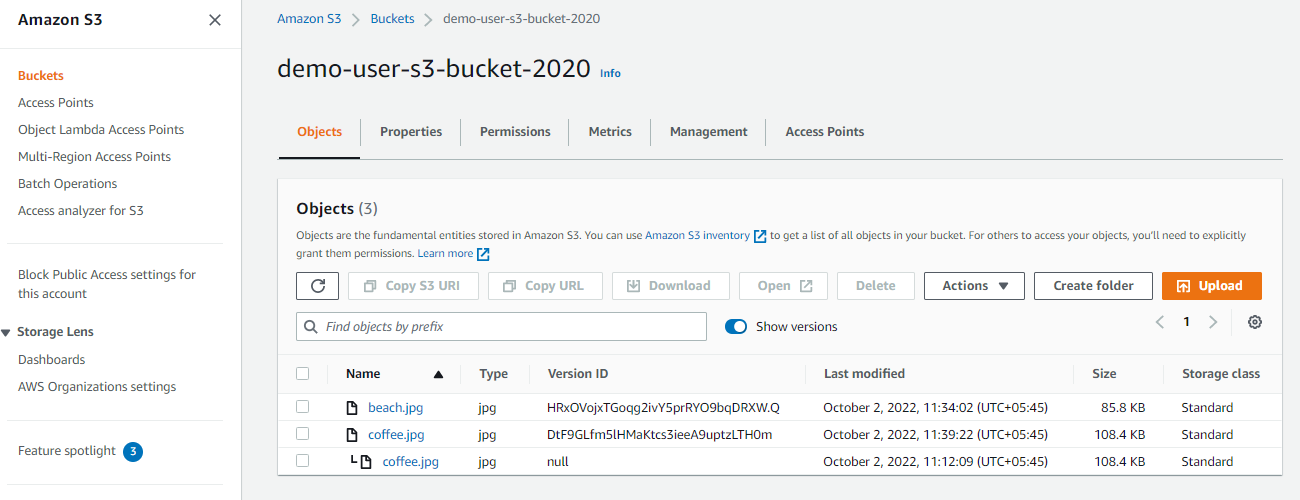


Go back to bucket

Our file is back

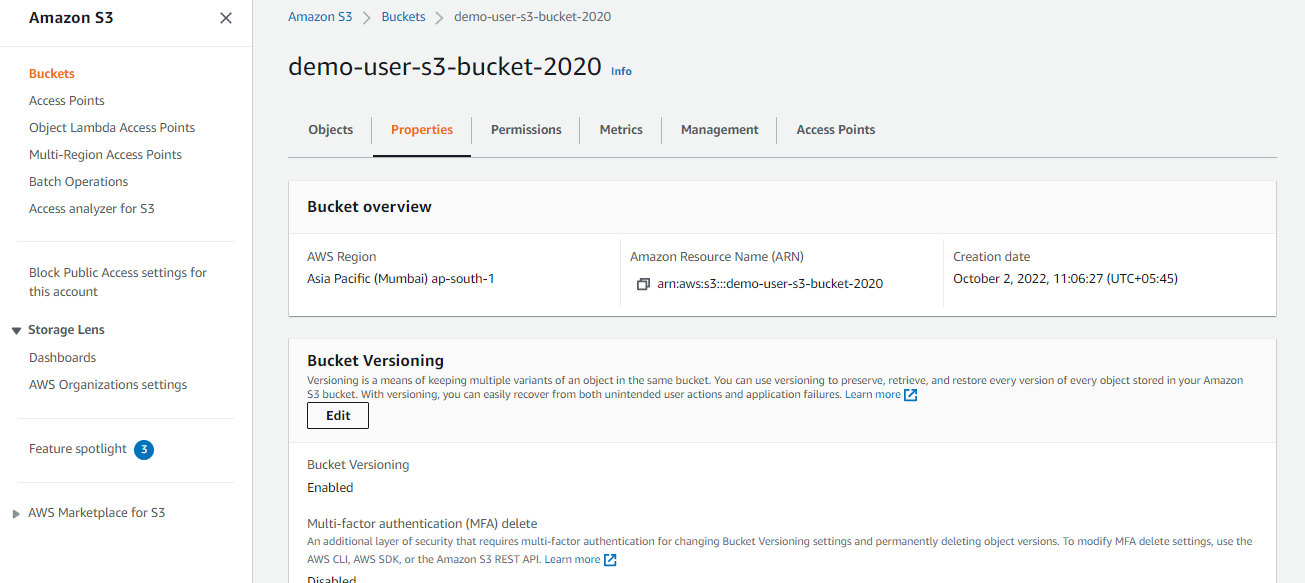


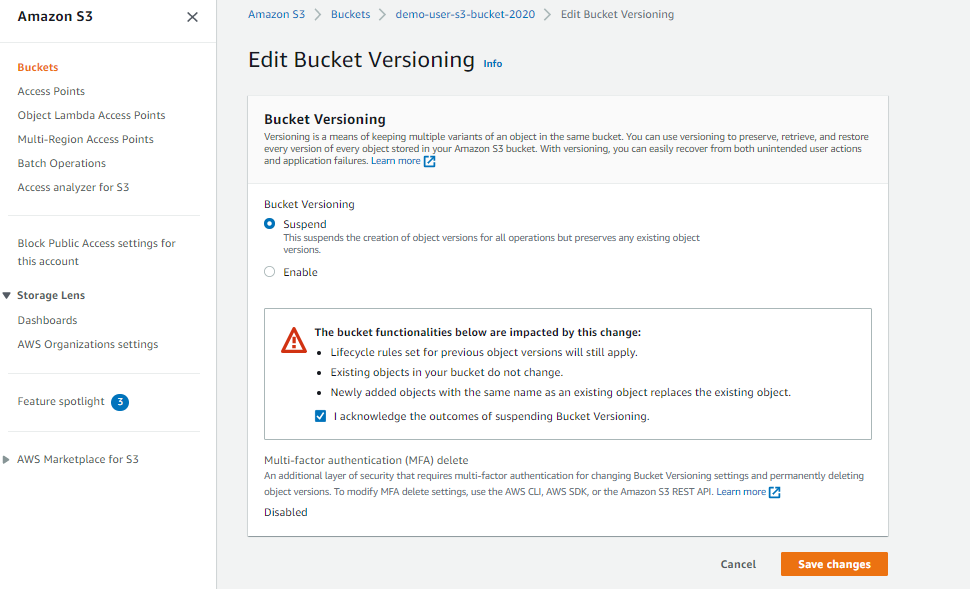
Again enable show versions



Go back to properties

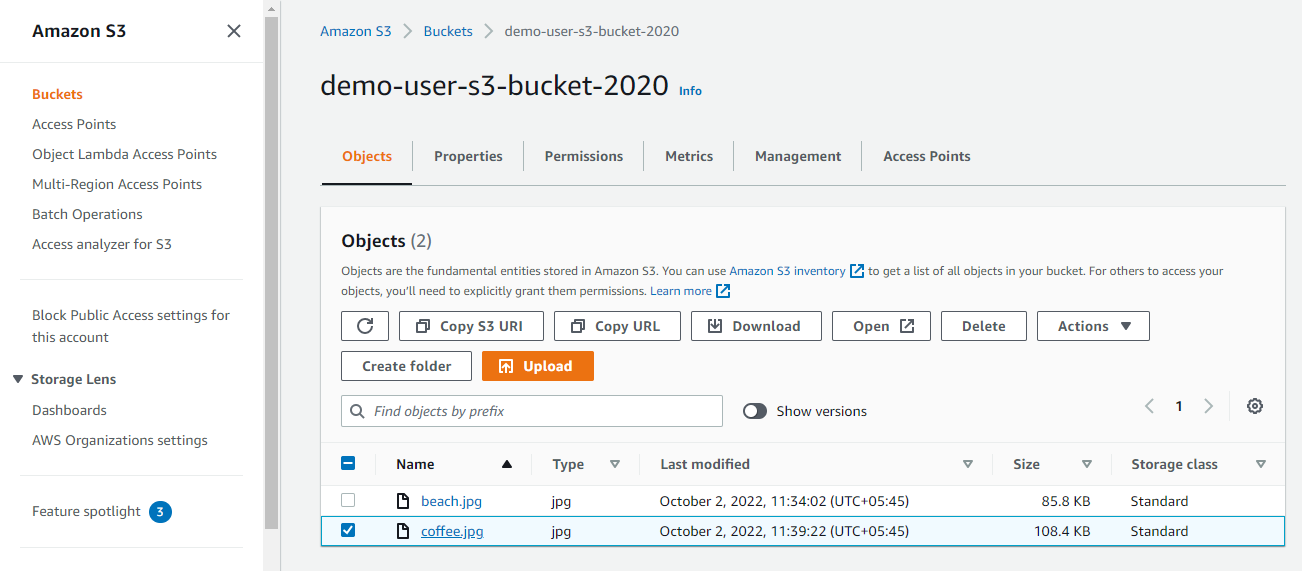
Edit bucket versioning



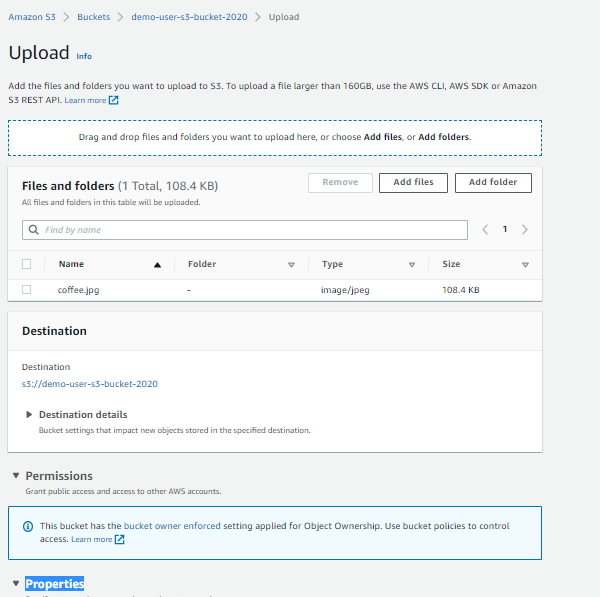


Encryption

Go back to coffee.jpg file

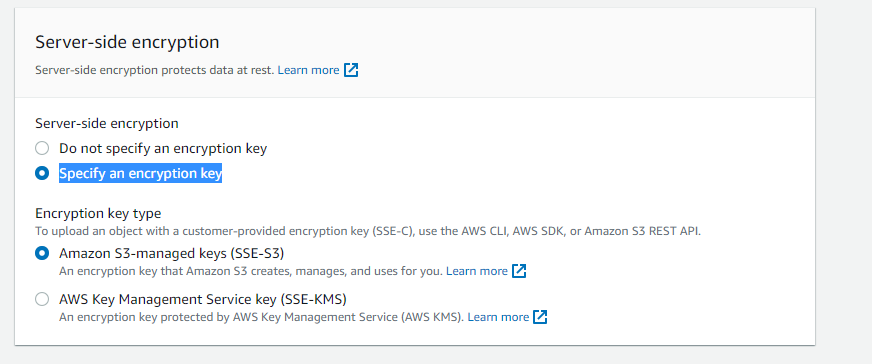


**Upload coffee.jpg file**

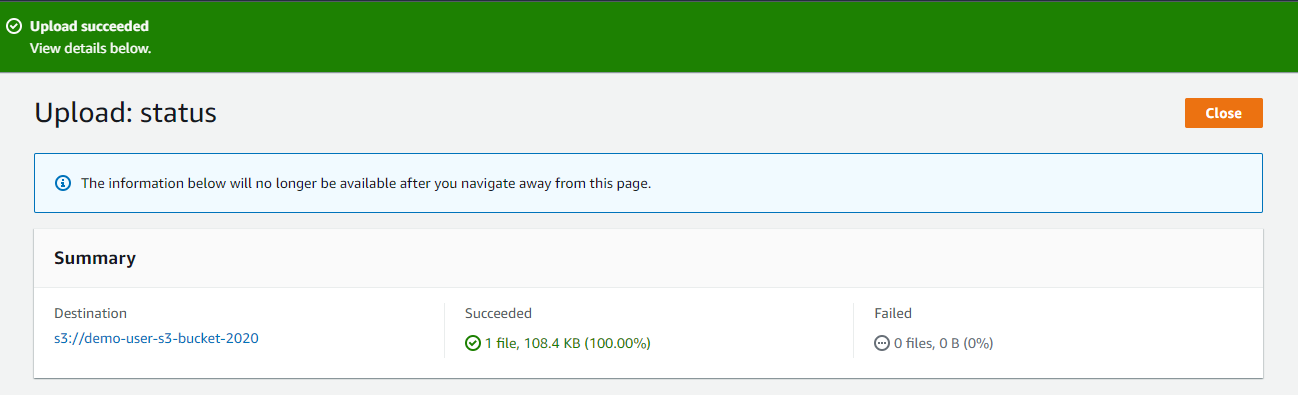


**Go to Properties**

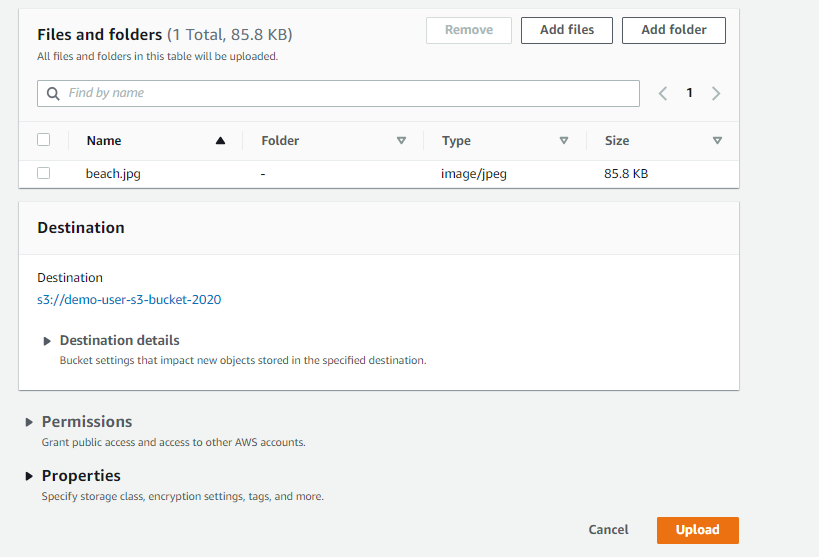
**Change server-side encryption on specify an encryption key**

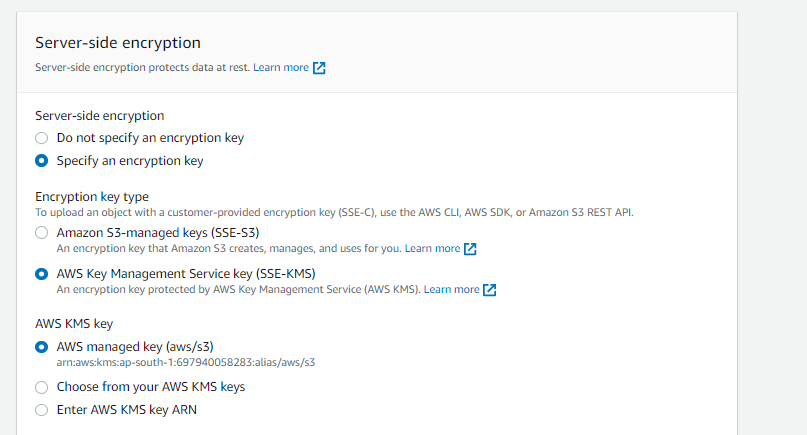


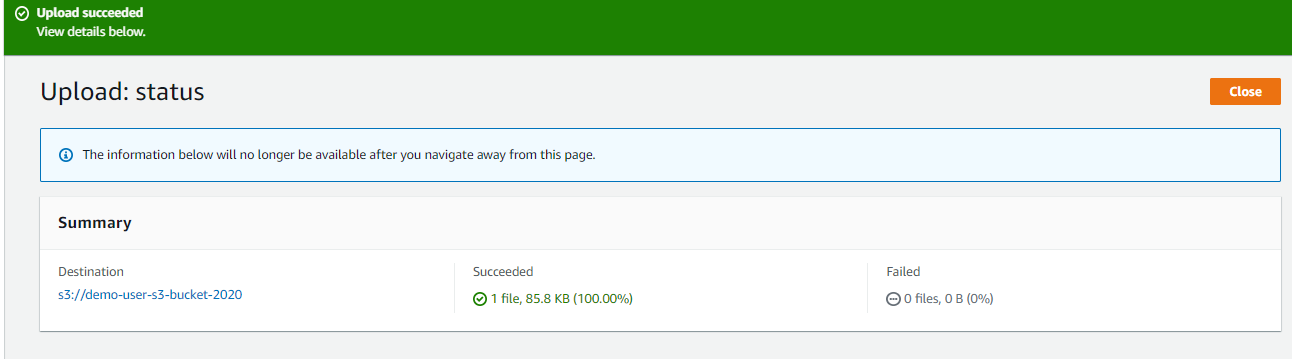
* **Click upload**

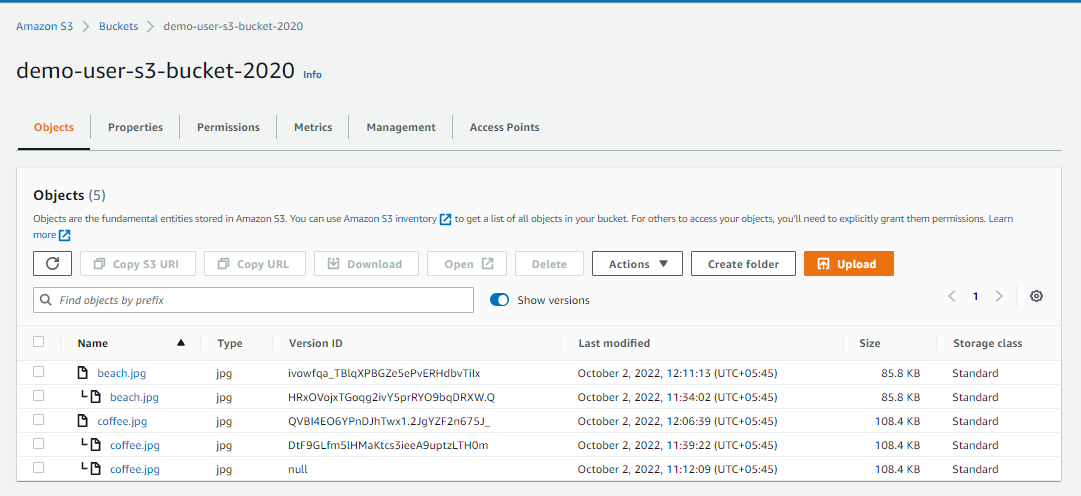


**Upload file and Go to properties**



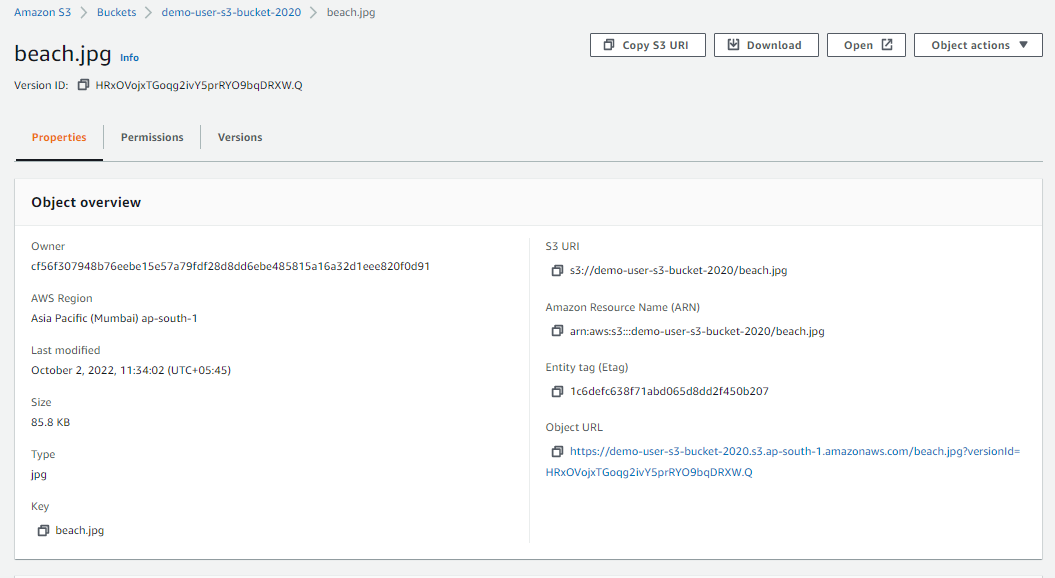


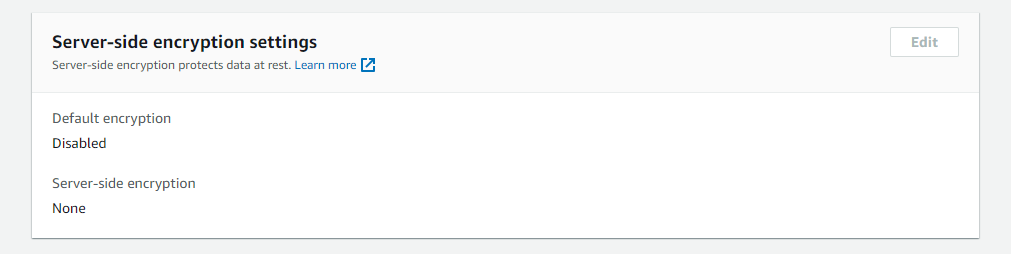




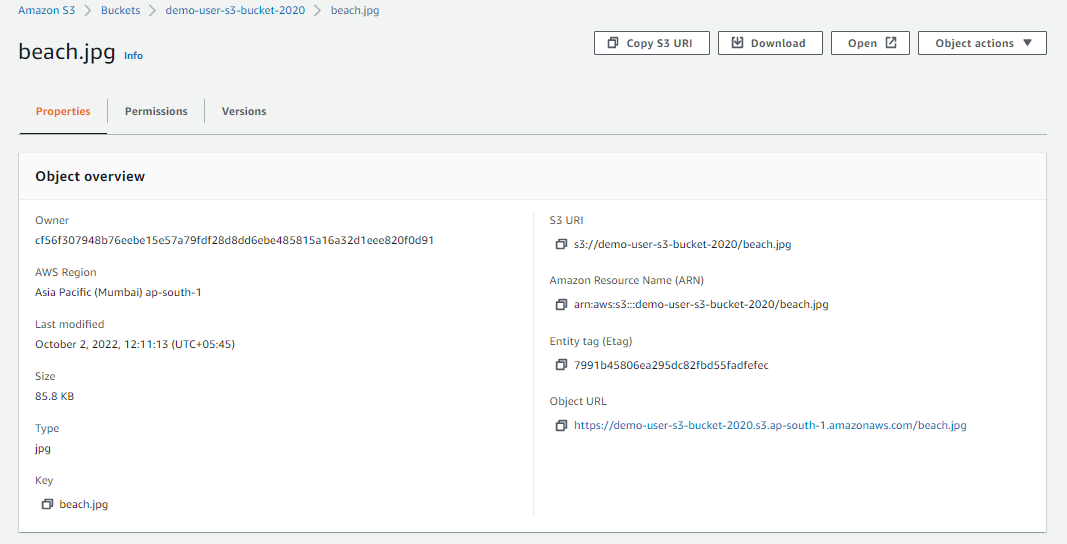
**Open two files one by one**

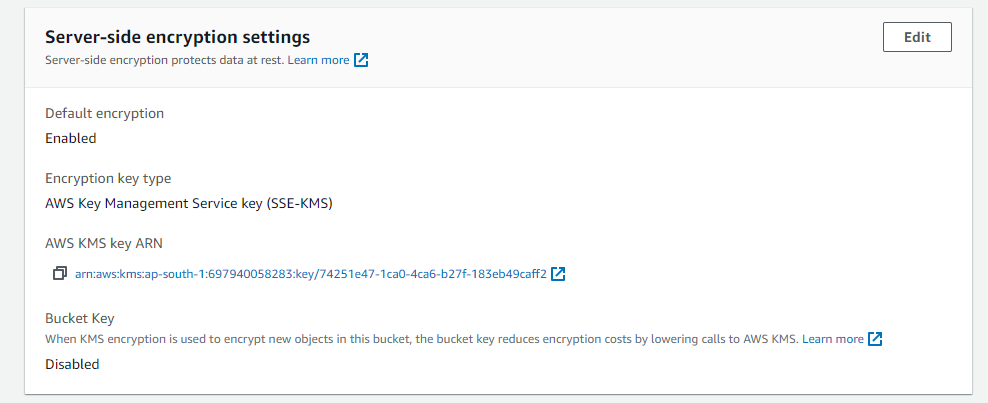
**Initial uploaded file encryption**



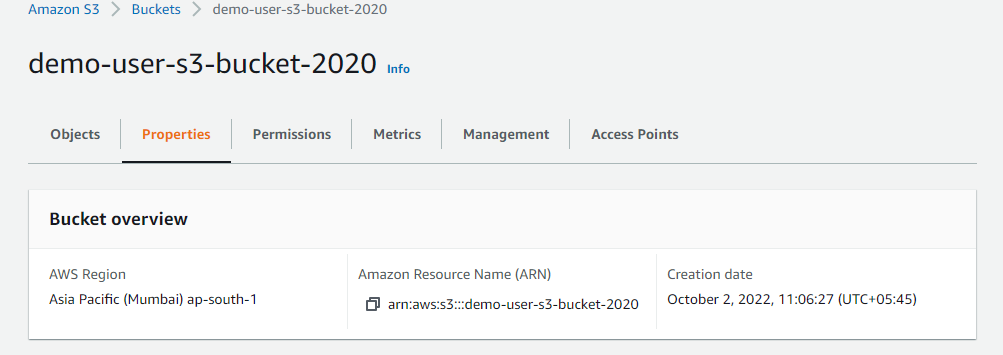


**Lastly upload encryption**

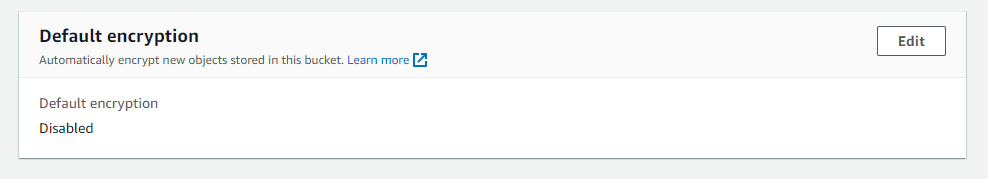


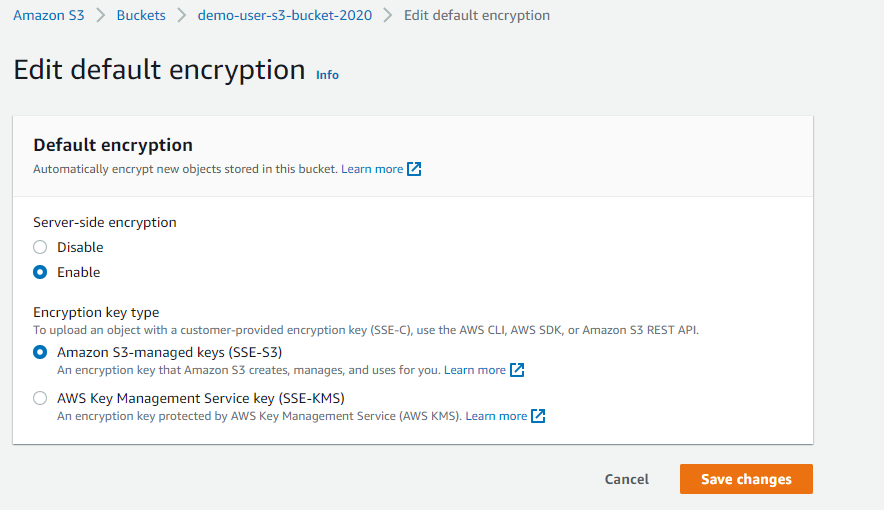


**Go to properties bucket**



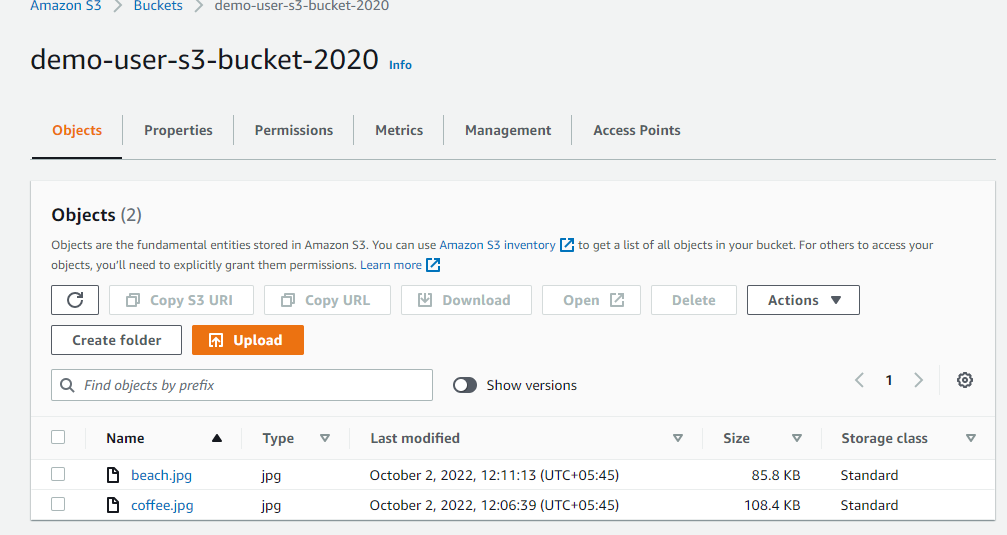
**Go to Default encryption**





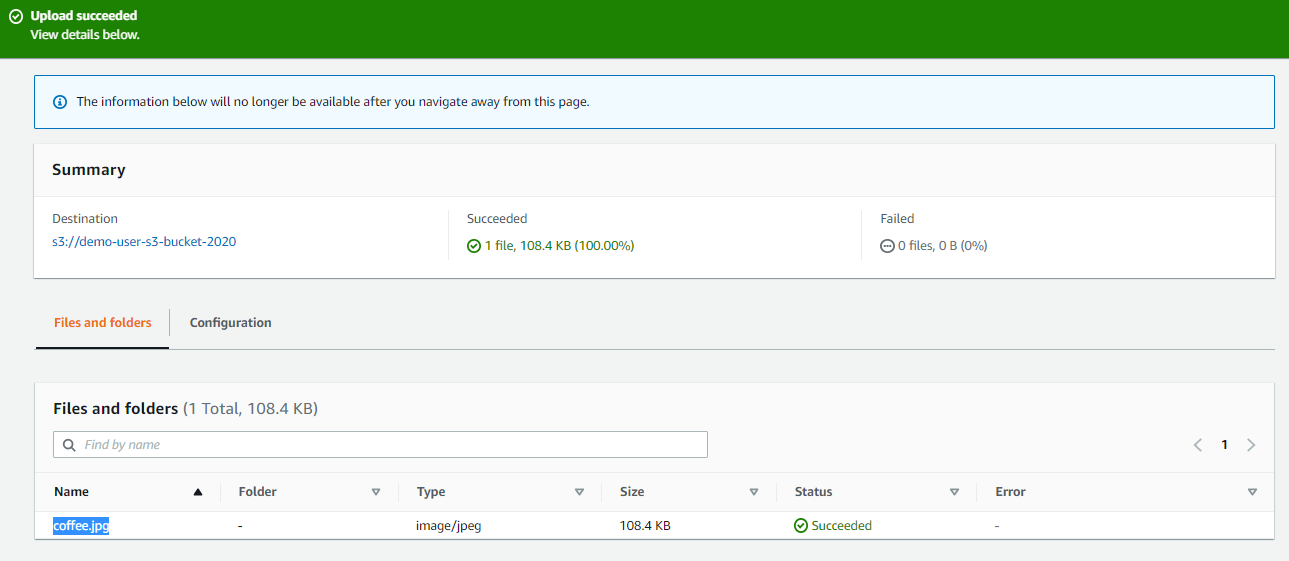
**Go to objects**

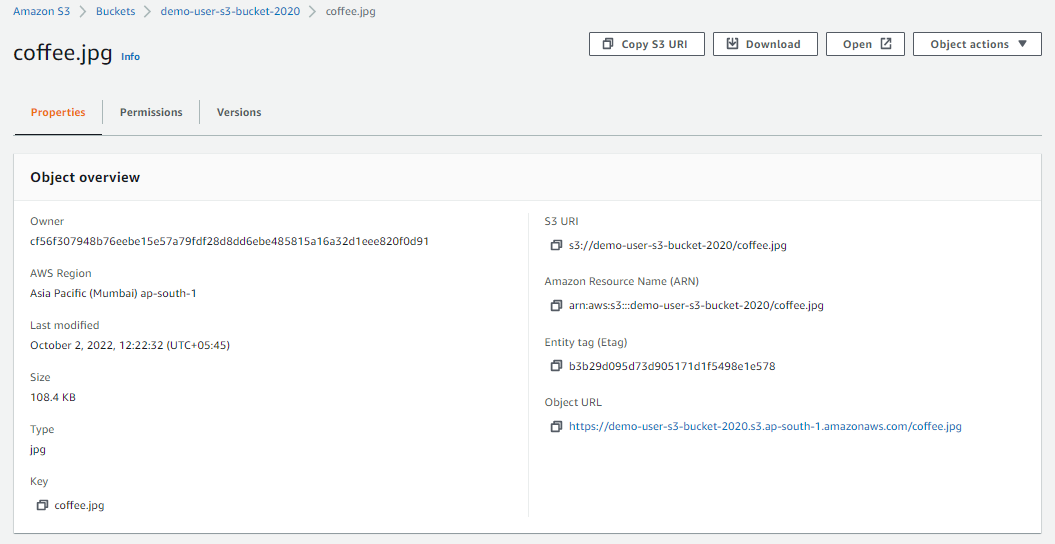
**Upload file coffee.jpg**

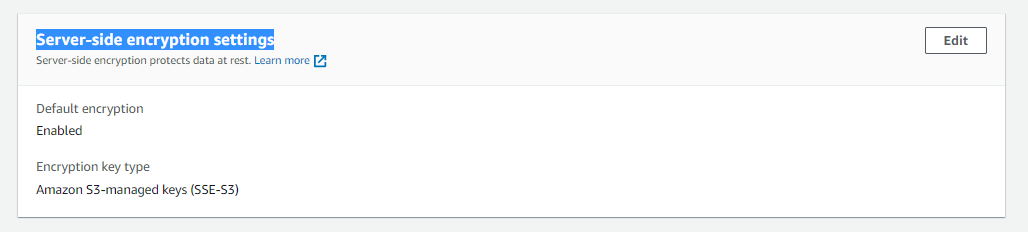


**Click upload without additional changes.**

**Open your file**

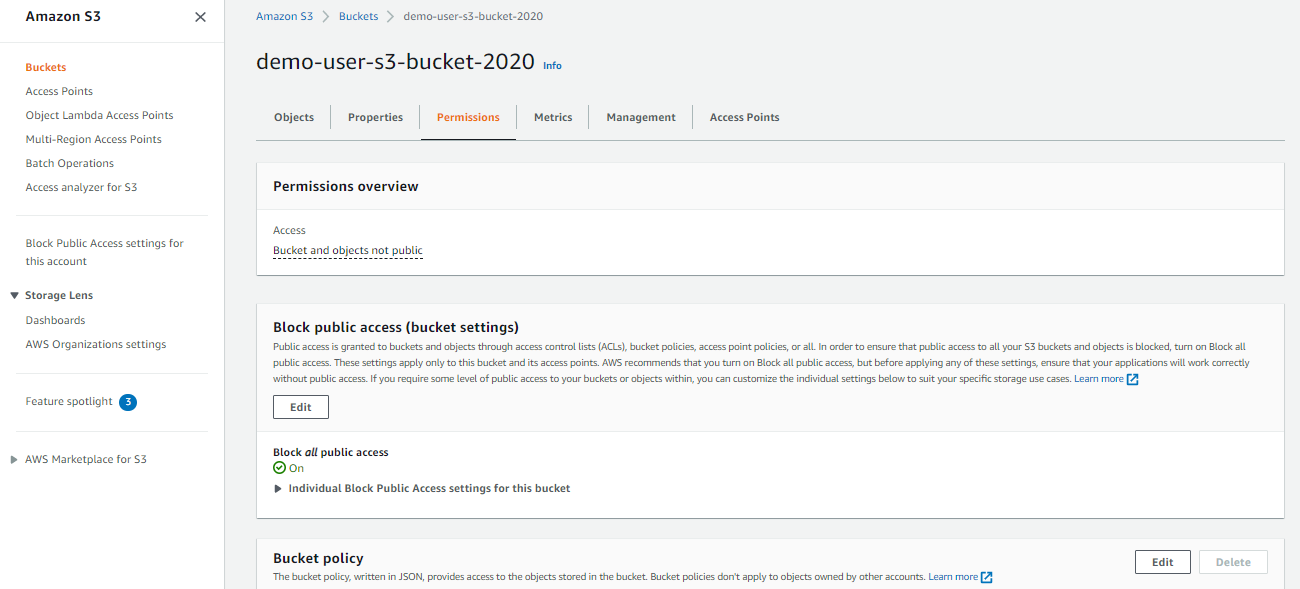






**S3 bucket Policy**

Go to permission

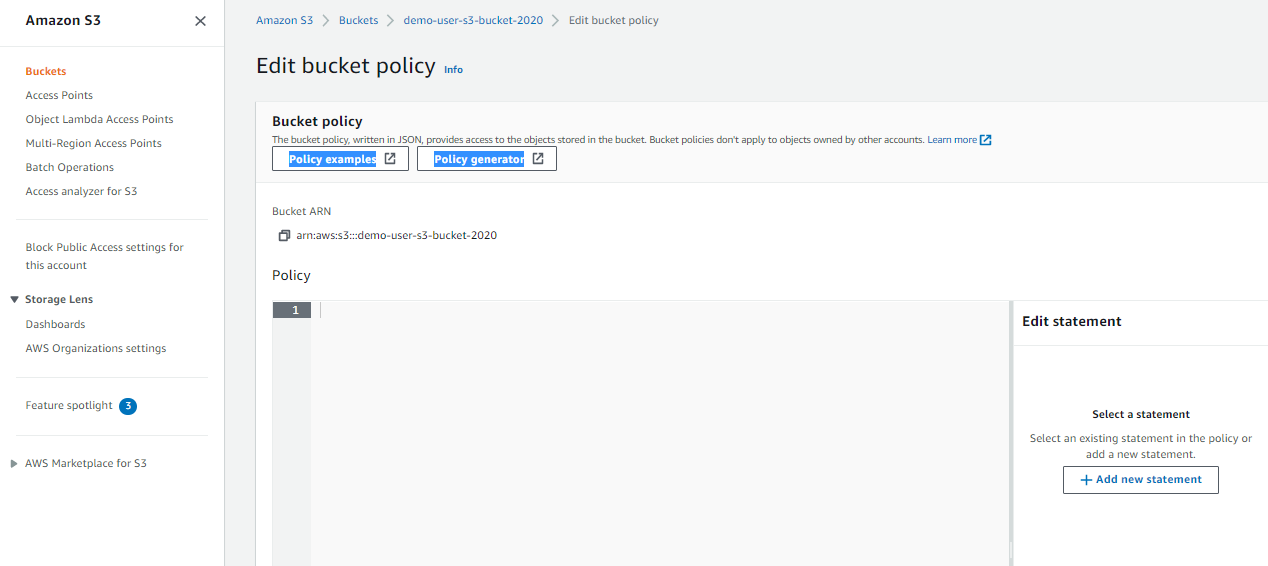


**Click edit policy**

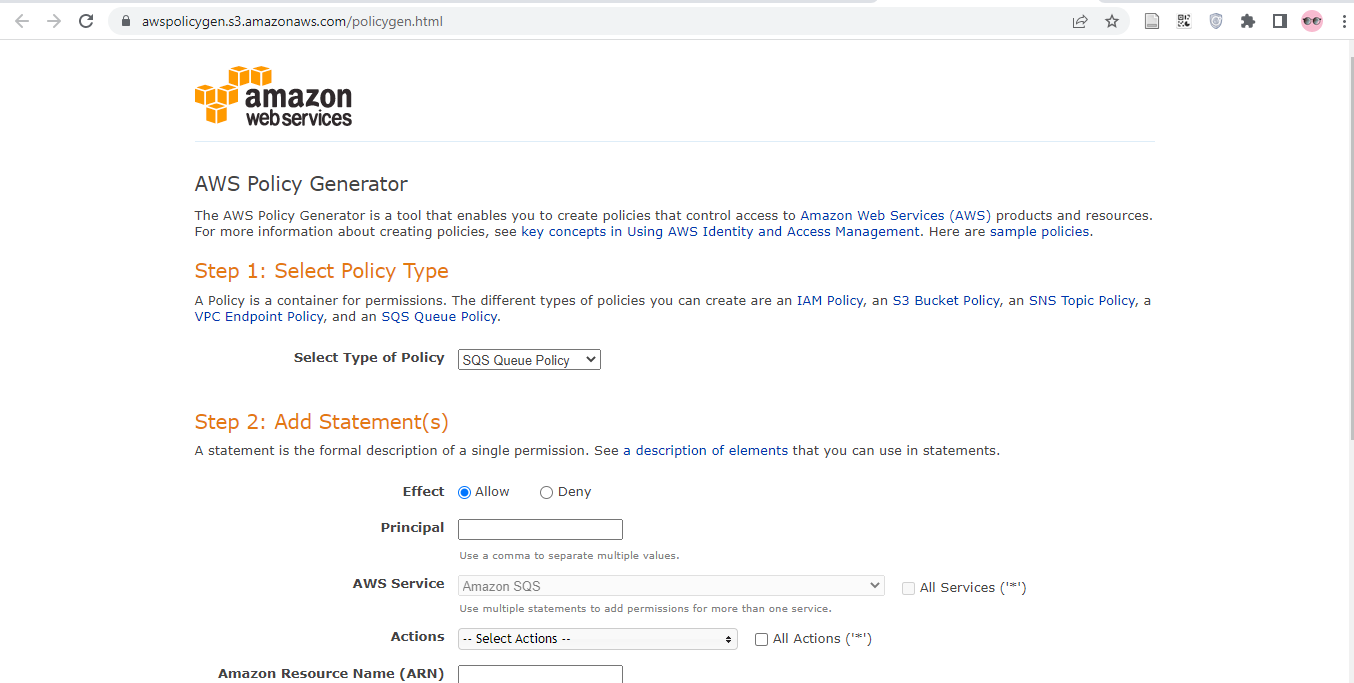


**Two ways to generate policy**

**Policy example and policy generator**



**Click policy generator**



**Follow below**



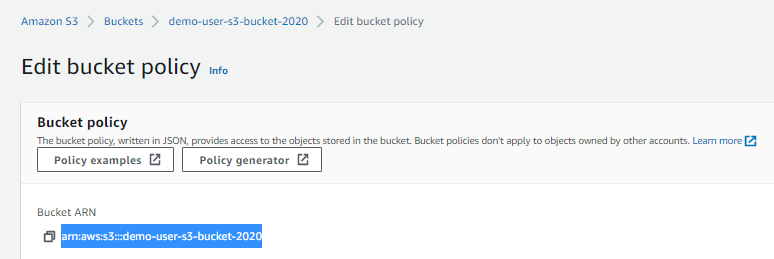
Principal: \*

Policy S3 policy

Effect : Deny

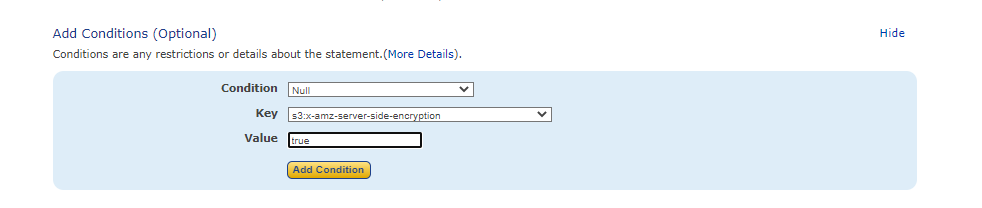
Actions: Put Object

Amazon Resource Name (ARN): arn:aws:s3:::demo-user-s3-bucket-2020/\*(donot forget to add /\*)

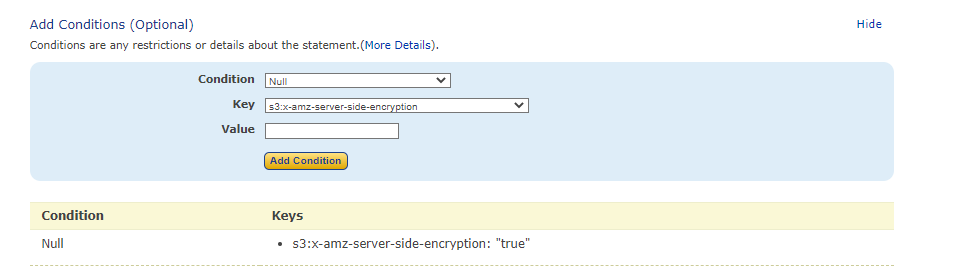


**Add condition**

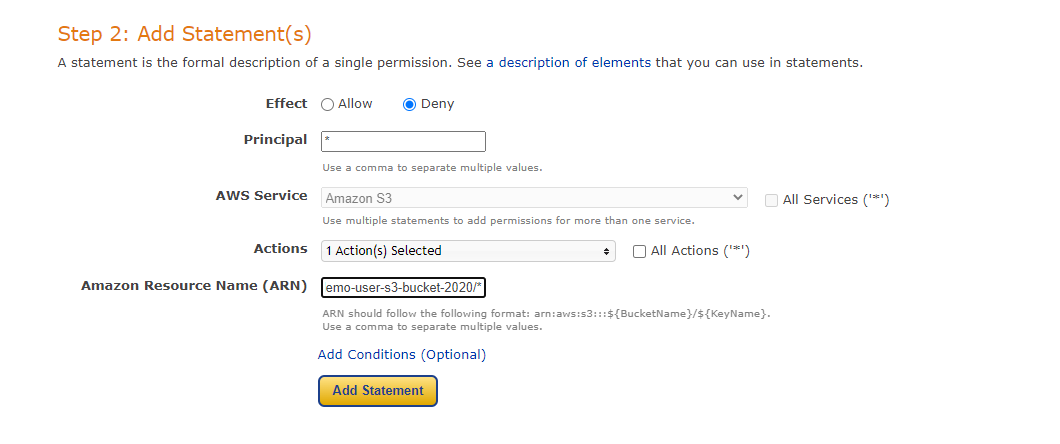
**Once completed click Add Condition**



**Output**

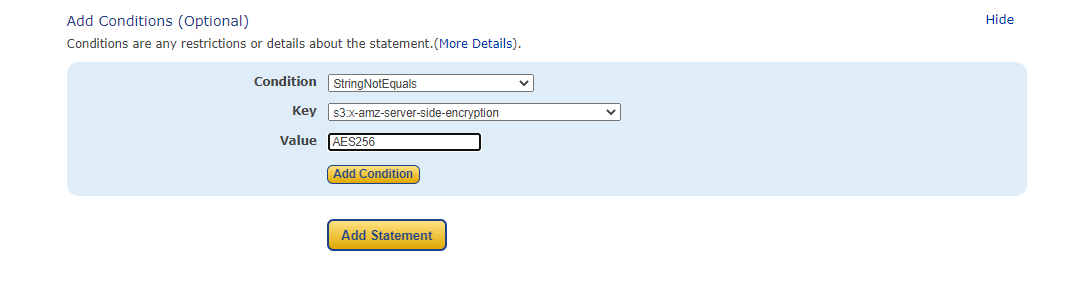


**Add statement(same as above)**



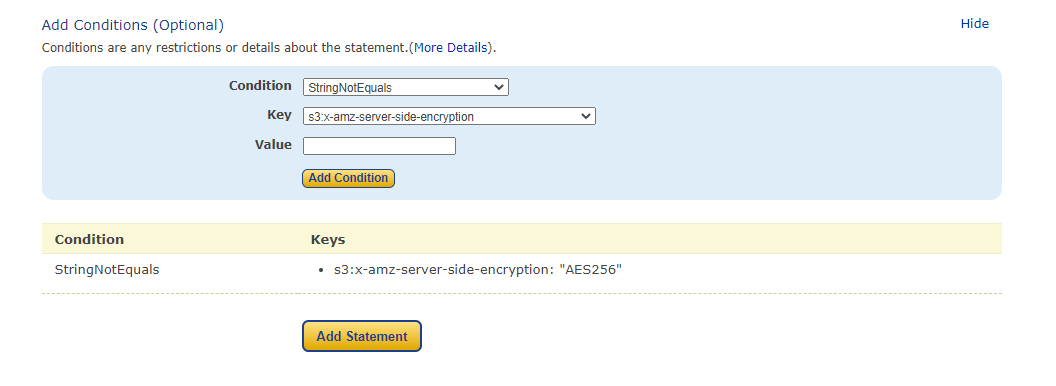
**Click add conitions**

**Then add condition**



**Once completed click Add Conditon(yellow)**

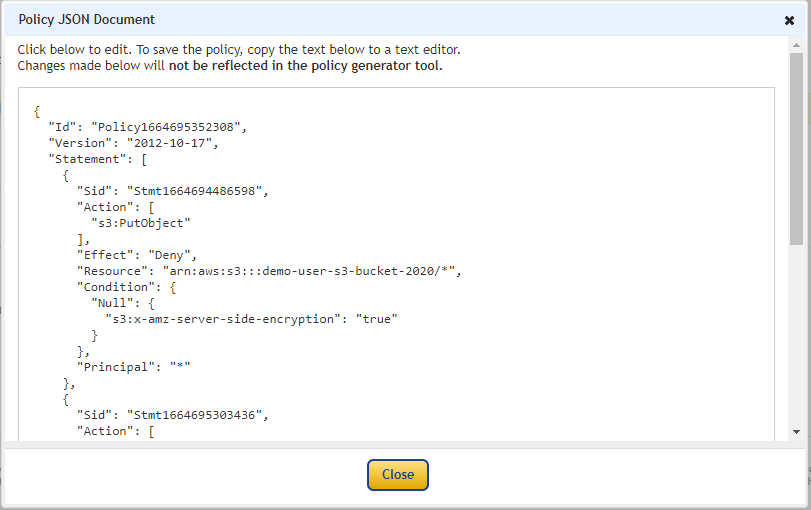
**Output**



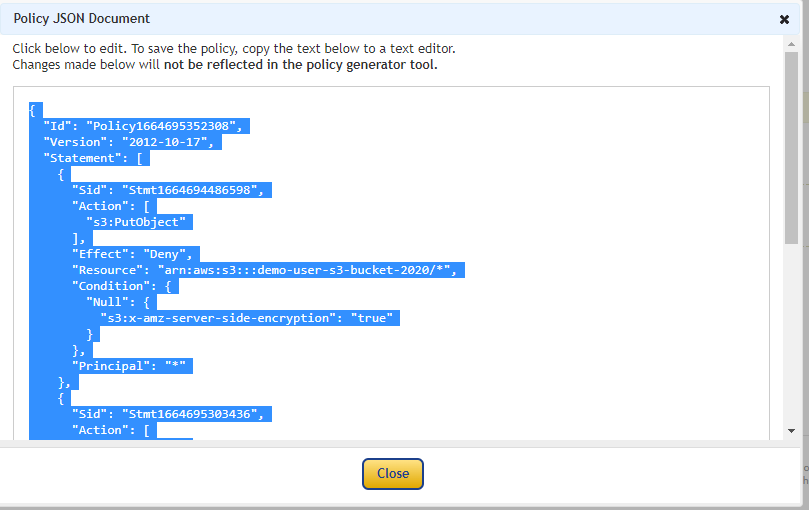
**Once completed Click Add Statement**

**Click generate policy**

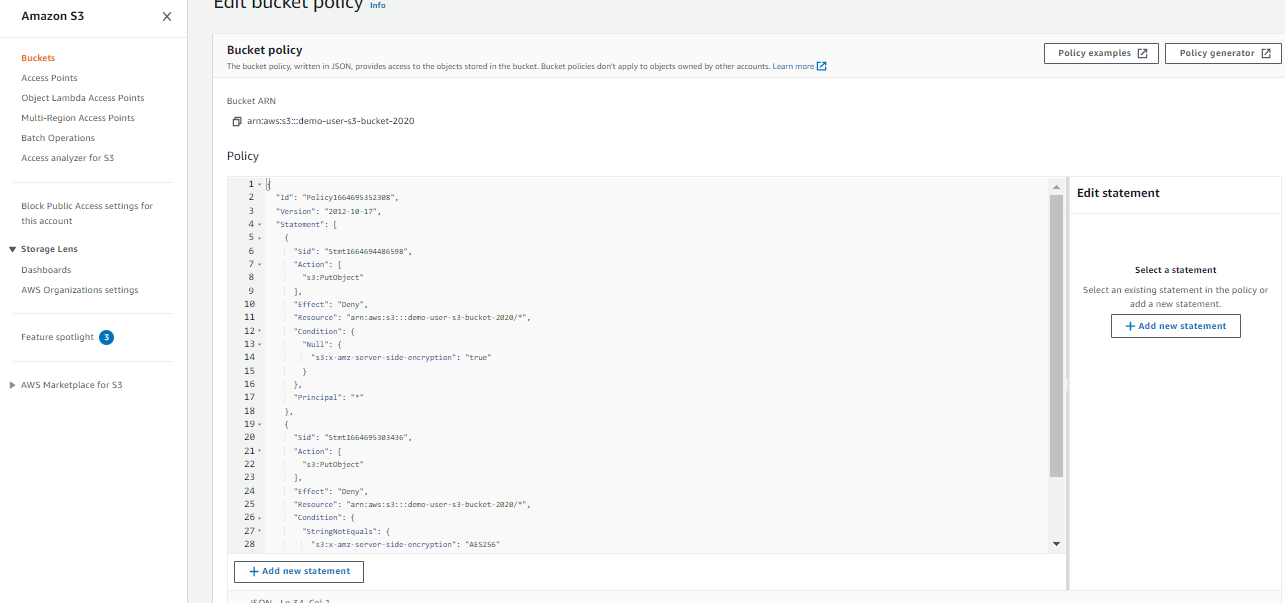




**Copy it**



**Paste it here**

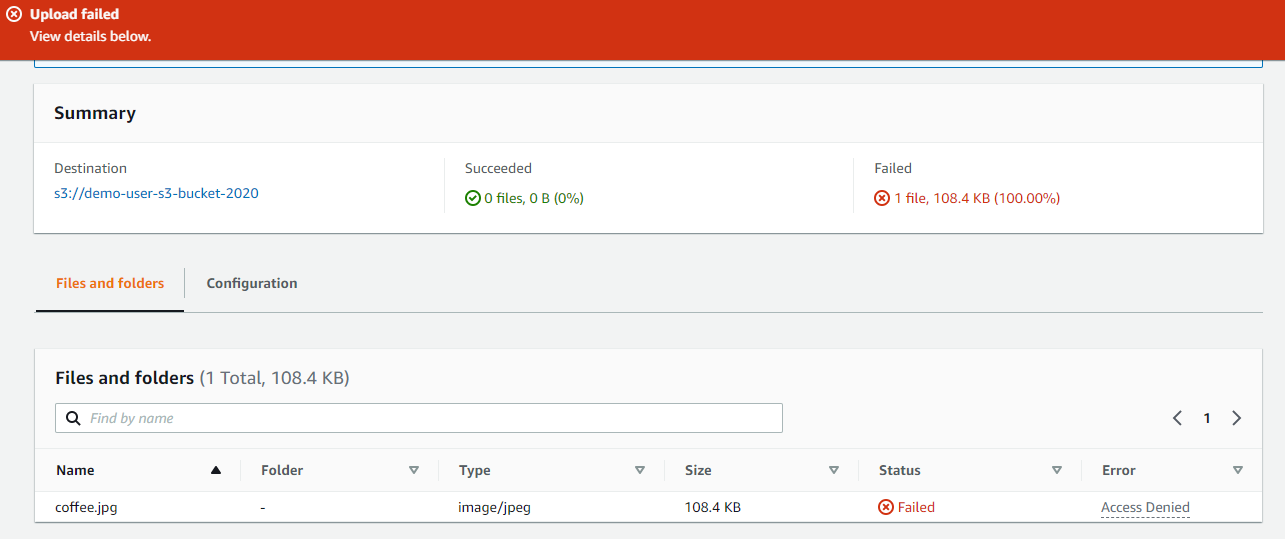


**Click save changes.**

**Go back to bucket and add file.**

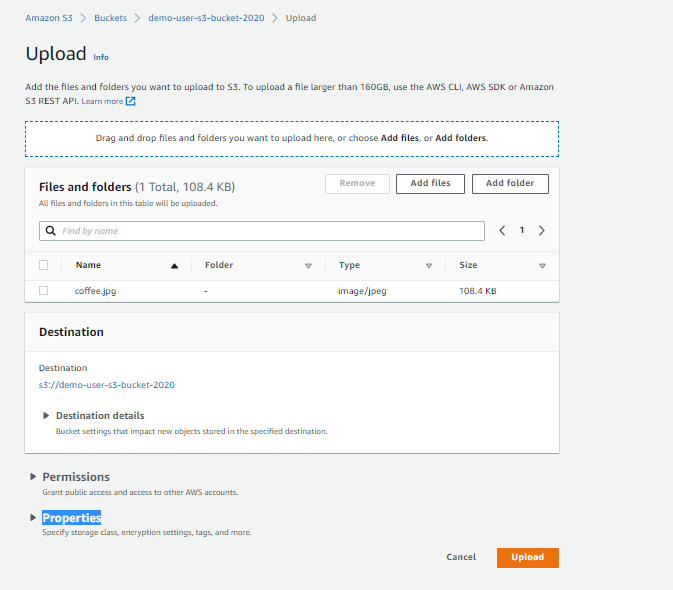


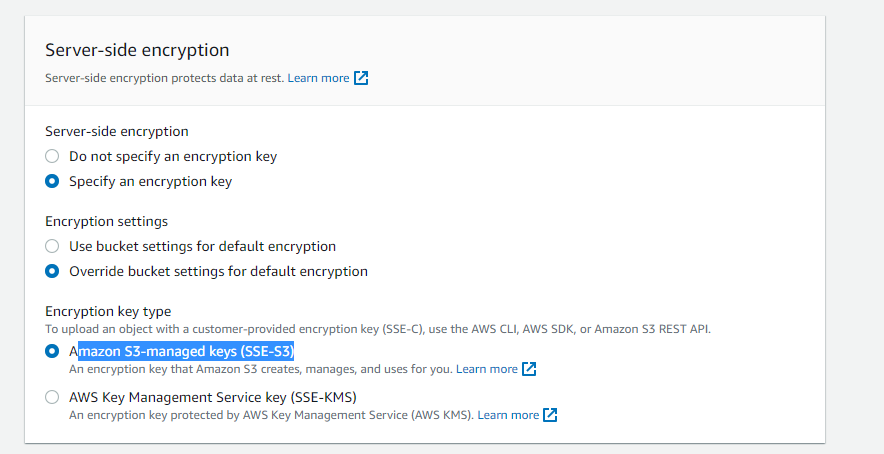
**It has been denied.**



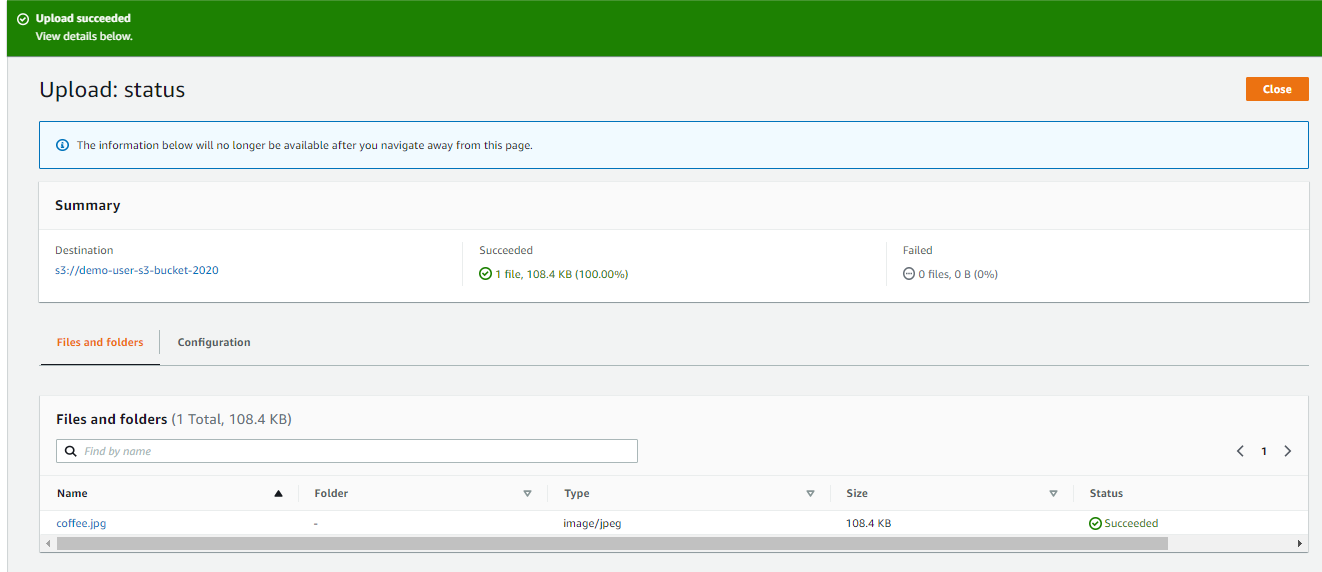
**Again if we upload file**

**Changing properties**



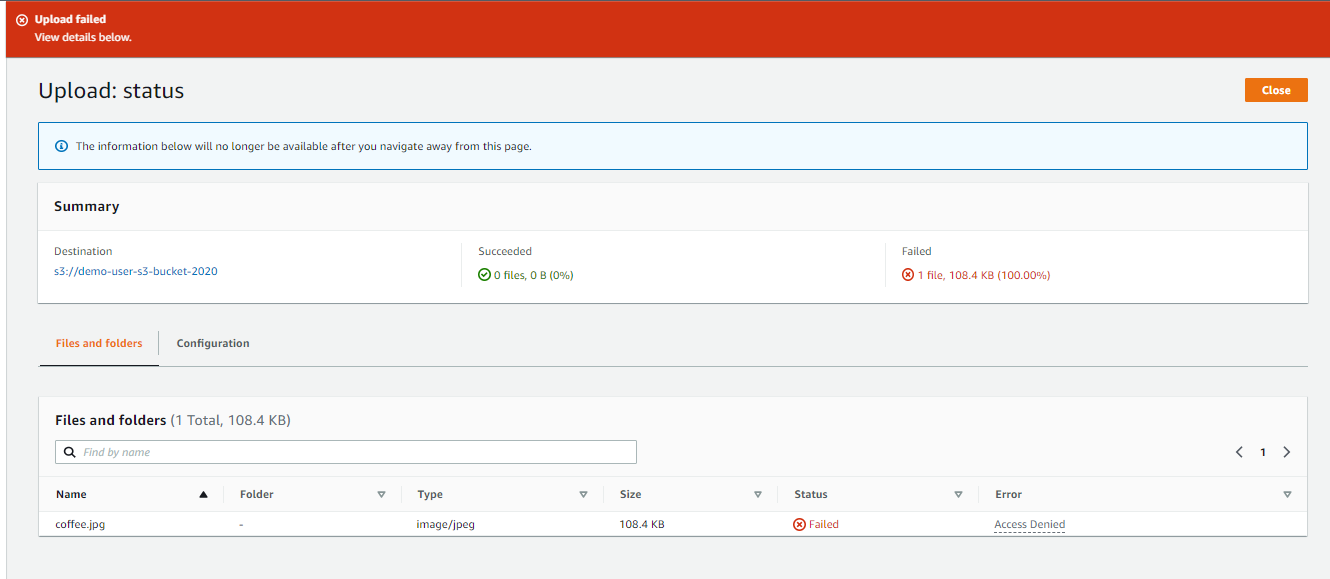


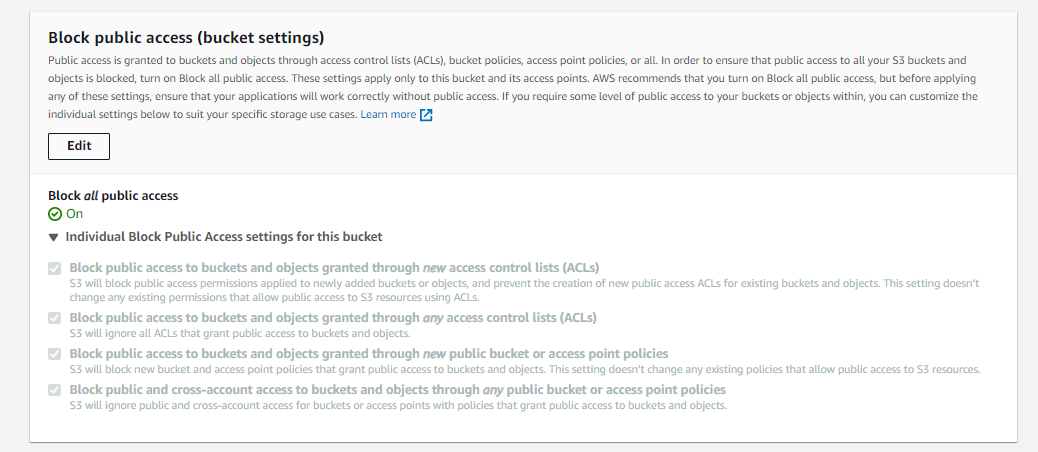
**File has successfully uploaded**



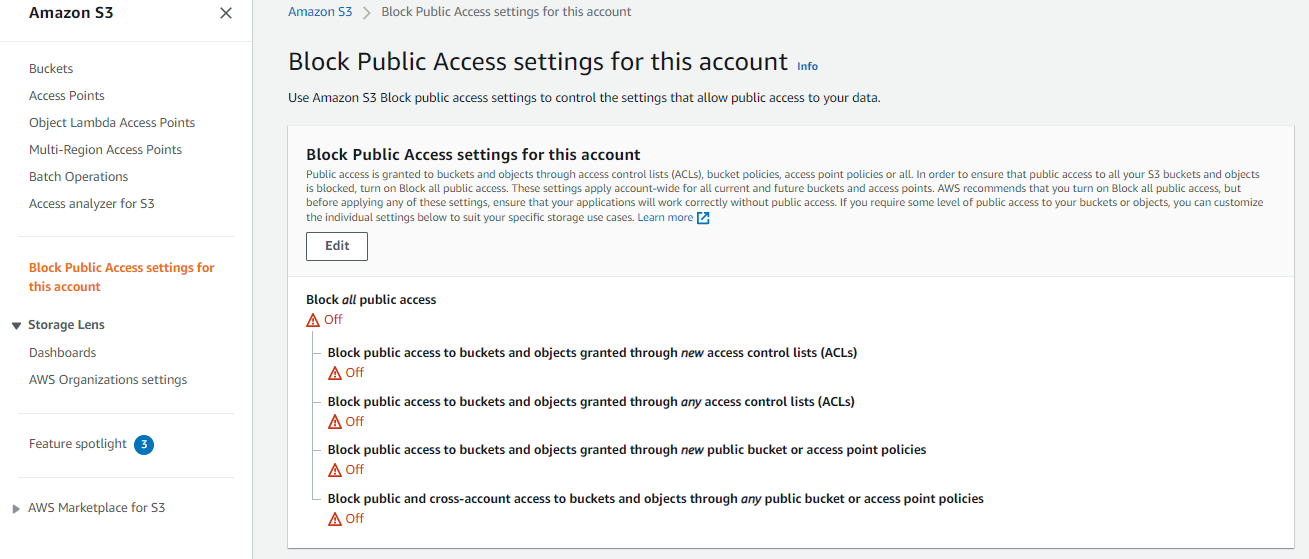
**This time upload file with SSE-KMS**







**Block all public websites**



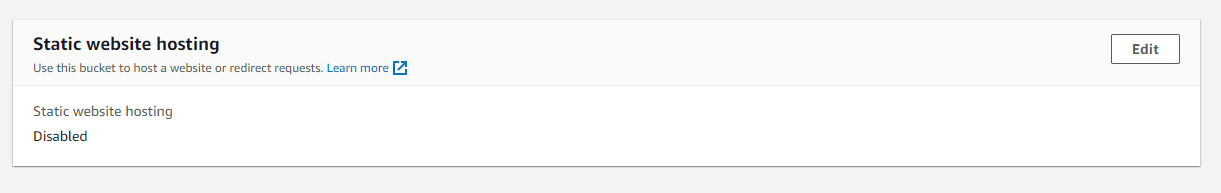
**CORS**

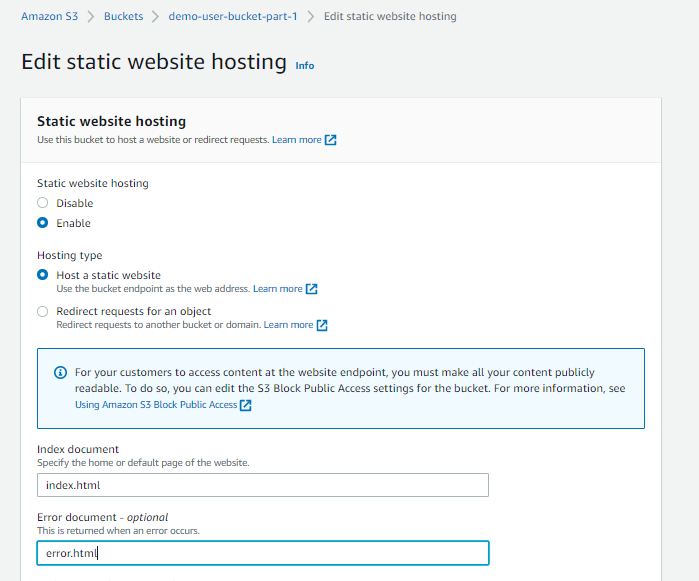
**Upload files**

**First bucket**



**Host static website**





{

"Version": "2012-10-17",

"Id": "Policy1639387301595",

"Statement": [

{

"Sid": "Stmt1639387300531",

"Effect": "Allow",

"Principal": "\*",

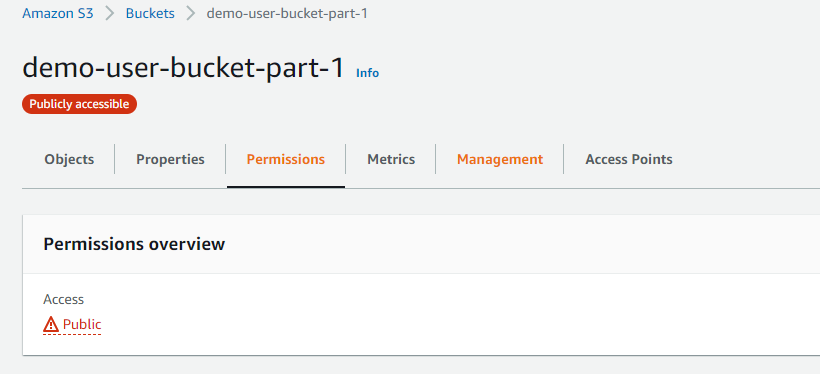
"Action": "s3:\*",

"Resource": "arn:aws:s3:::yourbucketname/\*"

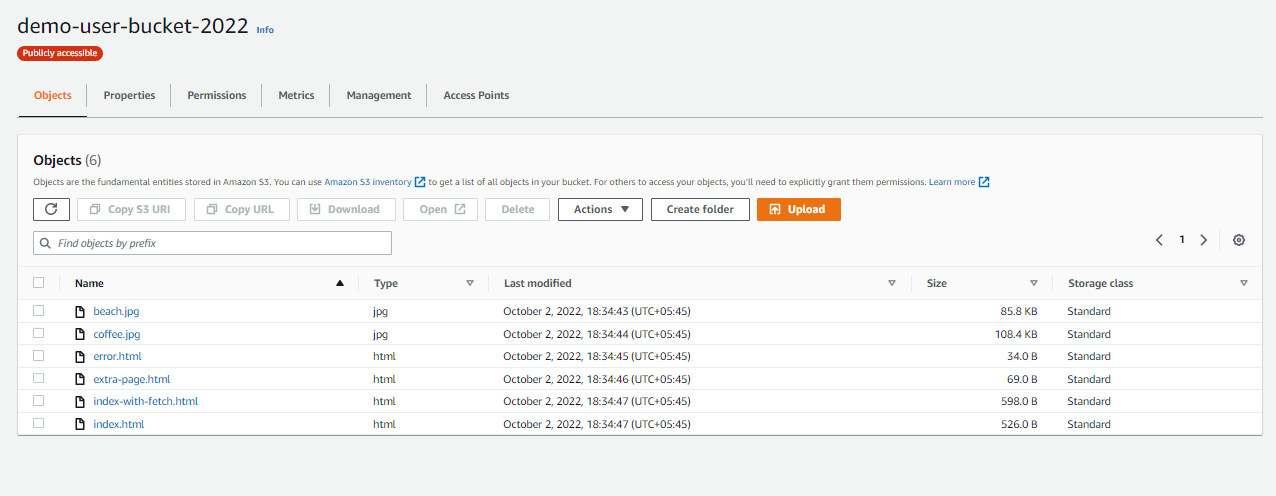
}

]

}



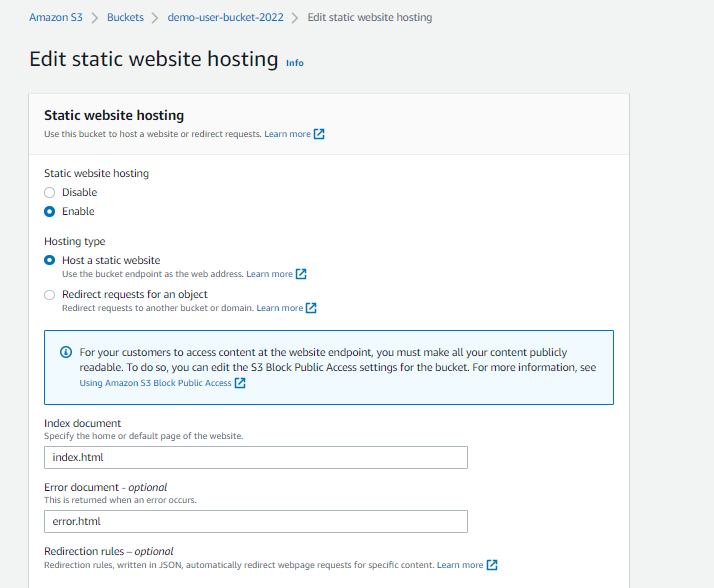
**Second bucket Add all required files**



**Host static websites**

**Go to bucket properties**

**Edit static website hosting**



Add bucket policy

{

"Version": "2012-10-17",

"Id": "Policy1639387301595",

"Statement": [

{

"Sid": "Stmt1639387300531",

"Effect": "Allow",

"Principal": "\*",

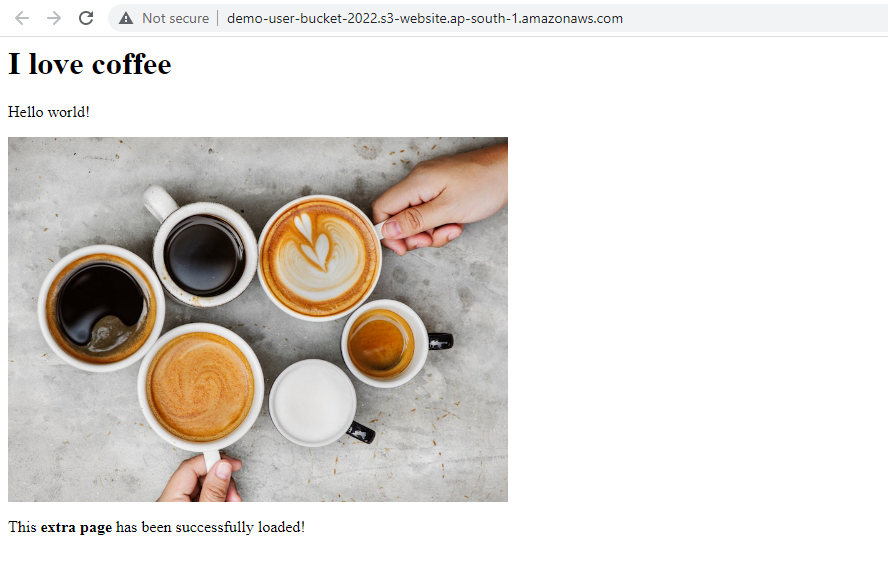
"Action": "s3:\*",

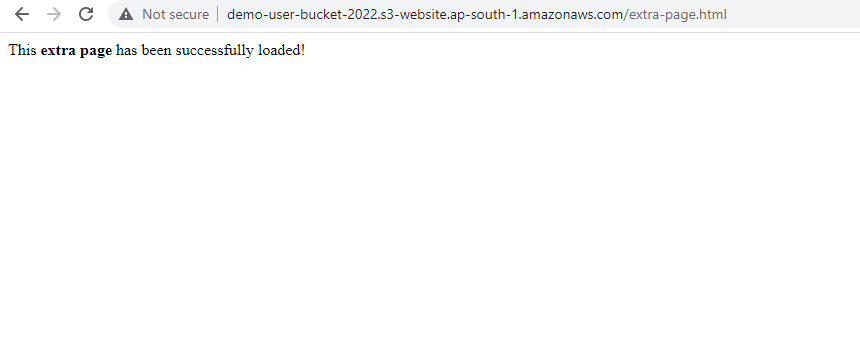
"Resource": "arn:aws:s3:::yourbucketname/\*"

}

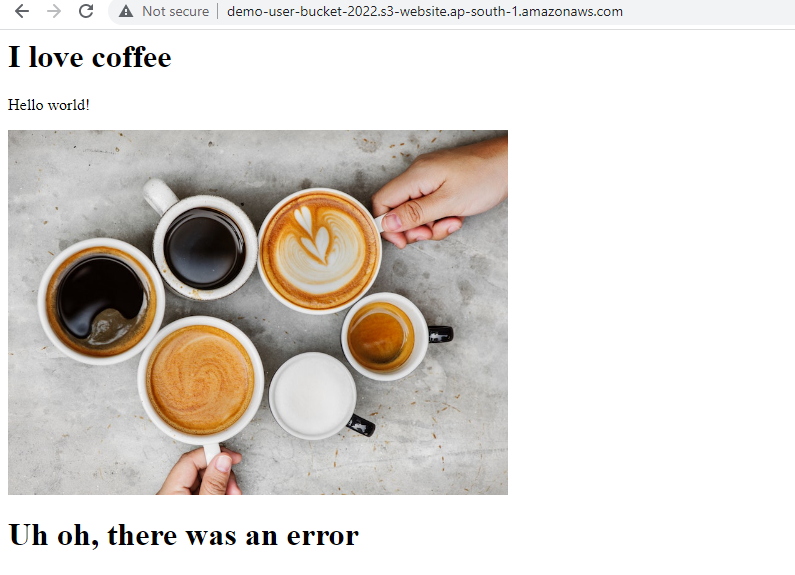
]

}





**Delete extra-page.html**



**Go to our second bucket**

**Go to permissions**

**Cross-origin resource sharing (CORS)**

[

    {

*"AllowedHeaders"*: [

            "Authorization"

        ],

*"AllowedMethods"*: [

            "GET"

        ],

*"AllowedOrigins"*: [

            "<url of first bucket with http://...without slash at the end>"

        ],

*"ExposeHeaders"*: [],

*"MaxAgeSeconds"*: 3000

    }

]

**Go back to first page**

