7/23/2022

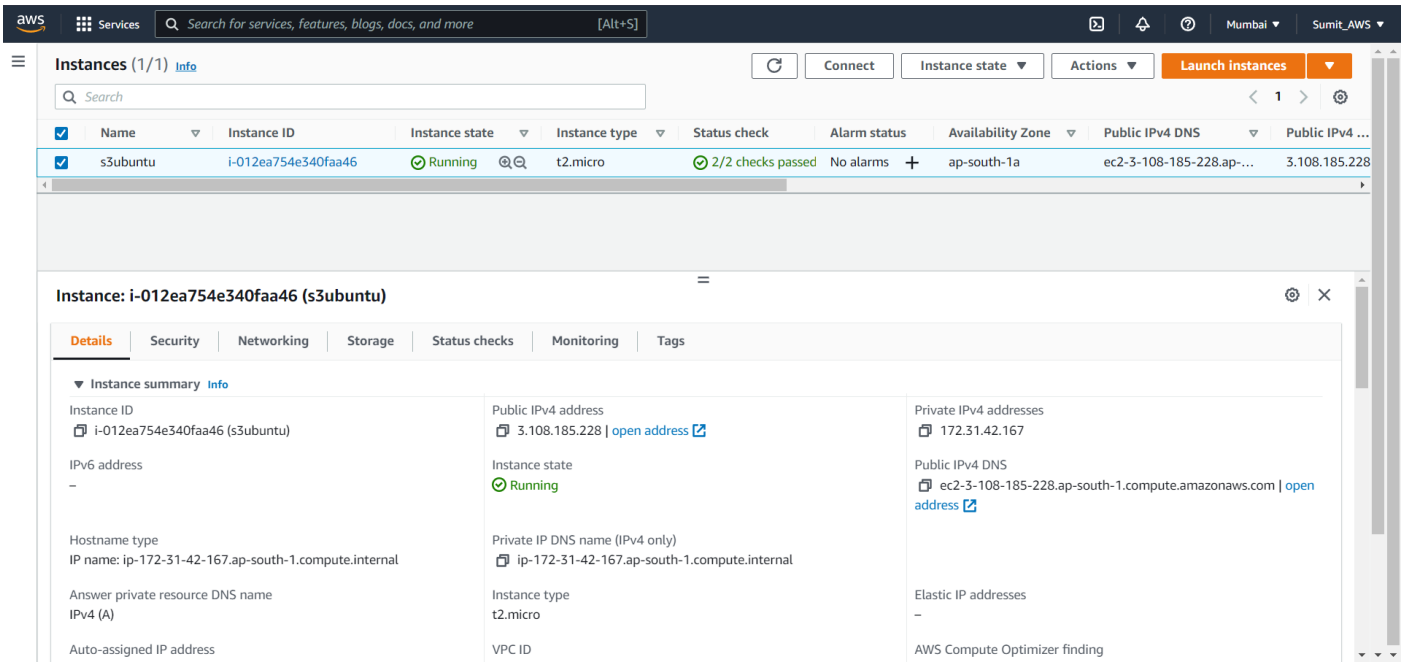
Project-05

Mount S3 bucket as a drive in both linux and windows, configure MFA delete and versioning on the bucket.

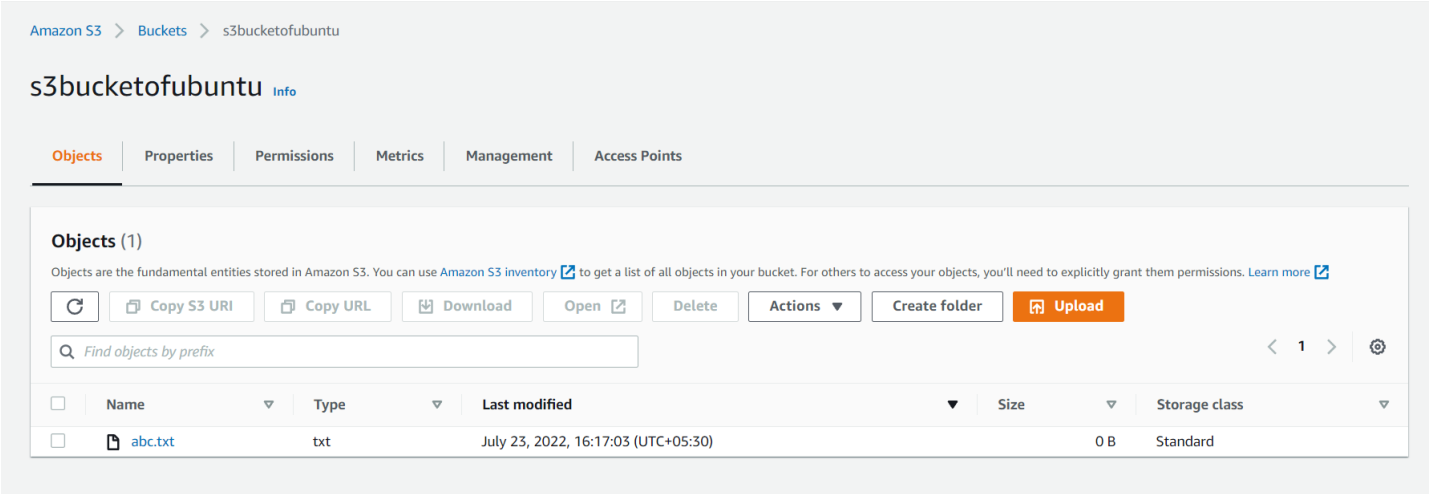
Sumit Mishra
190310286

1. Mount an S3 bucket on ubuntu AMI.

1. Created an EC2 instance to mount an S3 bucket using the Ubuntu AMI available.



2. Created an S3 bucket to mount to the EC2 instance. (Also uploaded a file named ‘abc.text’).



3. Connected to the EC2 instance using SSH client.
4. Fired the following commands to setup and mount the s3 bucket on the instance:
    a) ‘sudo apt-get update’ to update all libraries of Ubuntu.

```
root@ip-172-31-42-167:~# sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [114 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [99.8 kB]
Get:5 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [228 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [56.1 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [3564 B]
Get:9 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [203 kB]
```

b) ‘sudo apt-get install s3fs’ to install s3fs.

```
root@ip-172-31-42-167:~# sudo apt-get install s3fs
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bzip2 libfuse2 mailcap mime-support
Suggested packages:
  bzip2-doc
The following NEW packages will be installed:
```

c) ‘echo AKIATYFGC35IFYZNP7HA:5WR1bzCxVytPcfp1zRDedFJnR/JLD8Rm3zHqUxcP>~/ asswd-s3fs’ to create a configuration file for IAM user having access key and secret access key.

```
root@ip-172-31-42-167:~# echo AKIATYFGC35IFYZNP7HA:5WR1bzCxVytPcfp1zRDedFJnR/JLD8Rm3zHqUxcP>~/.passwd-s3fs
```

d) ‘cat ~/.passwd-s3fs’ to check if the file has been created and contents have been filled in it or not.

```
root@ip-172-31-42-167:~# cat ~/.passwd-s3fs
```

e) ‘chmod 600 ~/.passwd-s3fs’ to set permissions for the configuration file.

```
root@ip-172-31-42-167:~# chmod 600 `/.passwd-s3fs
```

f) ‘mkdir ~/s3-bucket-folder’ to make a directory to mount the s3 volume to.

```
root@ip-172-31-42-167:~# chmod 600 ~/.passwd-s3fs
root@ip-172-31-42-167:~# mkdir ~/s3-bucket-folder
```

g) ‘s3fs s3bucketofubuntu ~/s3-bucket-folder -o passwd_file=~/.passwd-s3fs’ to mount the s3 bucket to the specified directory using the specified configuratio file.

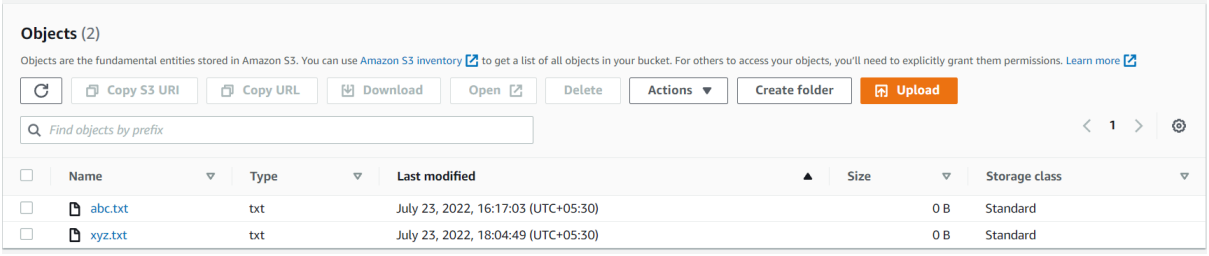
h) ‘mount’, ‘df –h’ to check if the s3 bucket has been mounted to the directory o not.

```
root@ip-172-31-42-167:~# s3fs s3bucketofubuntu ~/s3-bucket-folder -o passwd_file=~/.passwd-s3fs
root@ip-172-31-42-167:~# mount
```

i) ‘cd ~/s3-bucket-folder’ to go the directory where the volume has been mounted.

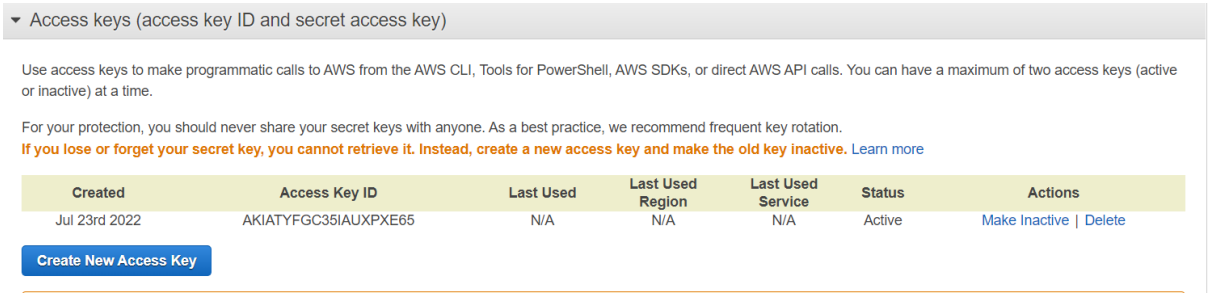
j) ‘ls’ to check if the file ‘abc.txt’ is present or not.

k) ‘touch xyz.txt’ to create a text inside that directory and check if it reflects in th s3 bucket in the console or not.



2. Enable MFA delete on s3 bucket.

1. Creating a new access key and secret access key for the root user account.



2. Configuring IAM account with the help of the generated access key and secret access key of the root user account.

```
PS C:\Users\cr7su> aws configure
AWS Access Key ID [****************P7HA]: AKIATYFGC35IAUXPXE65
AWS Secret Access Key [****************UxcP]: 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
Default region name [ap-south-1]:
Default output format [json]:
PS C:\Users\cr7su>
```

3. ‘aws s3 ls’ to display all the buckets in this region.

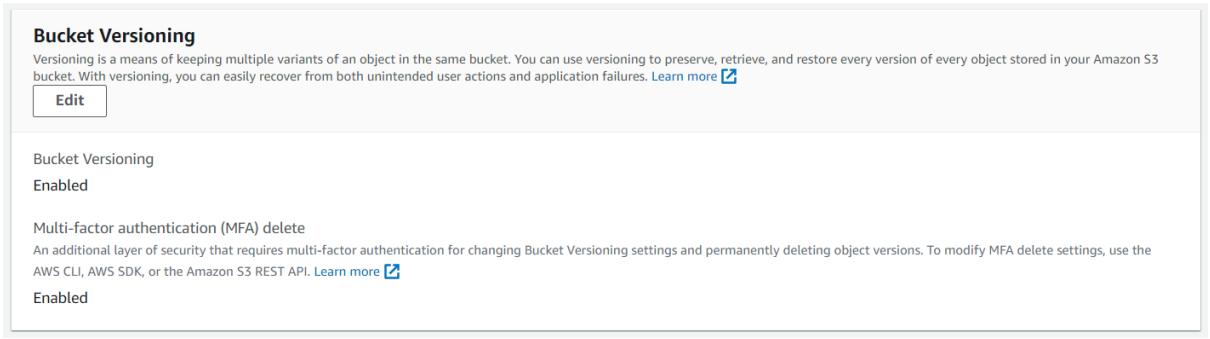
```
PS C:\Users\cr7su> aws s3 ls
2022-07-23 16:16:53 s3bucketofubuntu
```

4. ‘aws s3api put-bucket-versioning --bucket s3bucketofubuntu --versioning-configuration Status=Enabled,MFADelete=Enabled --mfa "arn:aws:iam::258046353232:mfa/root-account-mfa-device 090201"’ to enable MFA delete on the root account for s3 bucket.

```
PS C:\Users\cr7su> aws s3api put-bucket-versioning --bucket s3bucketofubuntu --versioning-configuration Status=Enabled,M
FADelete=Enabled --mfa "arn:aws:iam::258046353232:mfa/root-account-mfa-device 090201"
```

5. Checking if MFA delete and bucket versioning for the s3 bucket are enabled or not (also reflected in the console).

```
PS C:\Users\cr7su> aws s3api get-bucket-versioning --bucket s3bucketofubuntu
{
    "Status": "Enabled",
    "MFADelete": "Enabled"
}
```



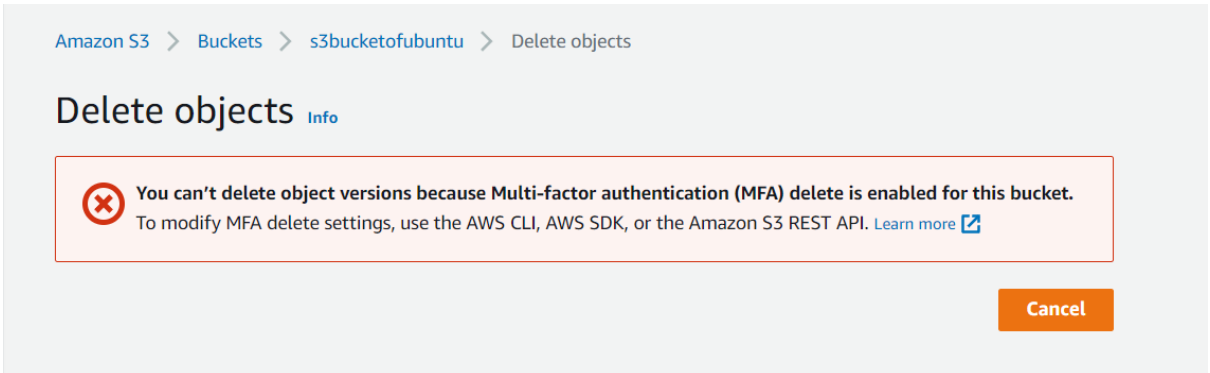
6. Deleting the 729 B abc.txt file without providing MFA key using CLI command.

```
PS C:\Users\cr7su> aws s3api delete-object --bucket s3bucketofubuntu --key abc.txt --version-id _4Ck2K8yjCwOdNf_AwntIS7qUI08GTDe

An error occurred (AccessDenied) when calling the DeleteObject operation: Mfa Authentication must be used for this request
PS C:\Users\cr7su>
```

7. Now, trying to delete the same file version using MFA code using CLI command. (We can see that the version of the abc.txt file has been deleted).

```
PS C:\Users\cr7su> aws s3api delete-object --bucket s3bucketofubuntu --key abc.txt --version-id _4Ck2K8yjCwOdNf_AwntIS7qUI08GTDe --mfa "arn:aws:iam::258046353232:mfa/root-account-mfa-device 399091"
{
    "VersionId": "_4Ck2K8yjCwOdNf_AwntIS7qUI08GTDe"
}
```

8. Now, trying to delete xyz.txt version using the s3 management console we get error message.



3. Syncing files between two s3 buckets.

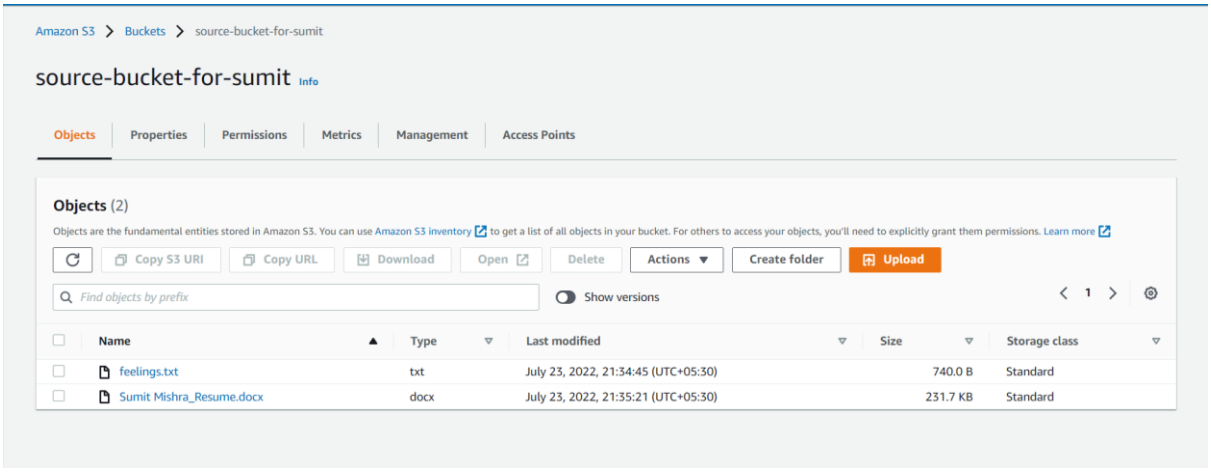
1. Creating Source bucket and enabling versioning to it.

```
PS C:\Users\cr7su> aws s3api create-bucket --bucket source-bucket-for-sumit
--region ap-south-1 --create-bucket-configuration LocationConstraint=ap-sout
h-1
{
    "Location": "http://source-bucket-for-sumit.s3.amazonaws.com/"
}

PS C:\Users\cr7su>
PS C:\Users\cr7su>
PS C:\Users\cr7su>
PS C:\Users\cr7su>
PS C:\Users\cr7su>
PS C:\Users\cr7su>
PS C:\Users\cr7su> aws s3api put-bucket-versioning --bucket source-bucket-fo
r-sumit --versioning-configuration Status=Enabled
PS C:\Users\cr7su>
```

2. Creating the destination bucket and enabling versioning to it.

```
PS C:\Users\cr7su> aws s3api create-bucket --bucket destn-bucket-for-sumit -
-region ap-south-1 --create-bucket-configuration LocationConstraint=ap-south
-1
{
    "Location": "http://destn-bucket-for-sumit.s3.amazonaws.com/"
}

PS C:\Users\cr7su> aws s3api put-bucket-versioning --bucket destn-bucket-for
-sumit --versioning-configuration Status=Enabled
PS C:\Users\cr7su>
```

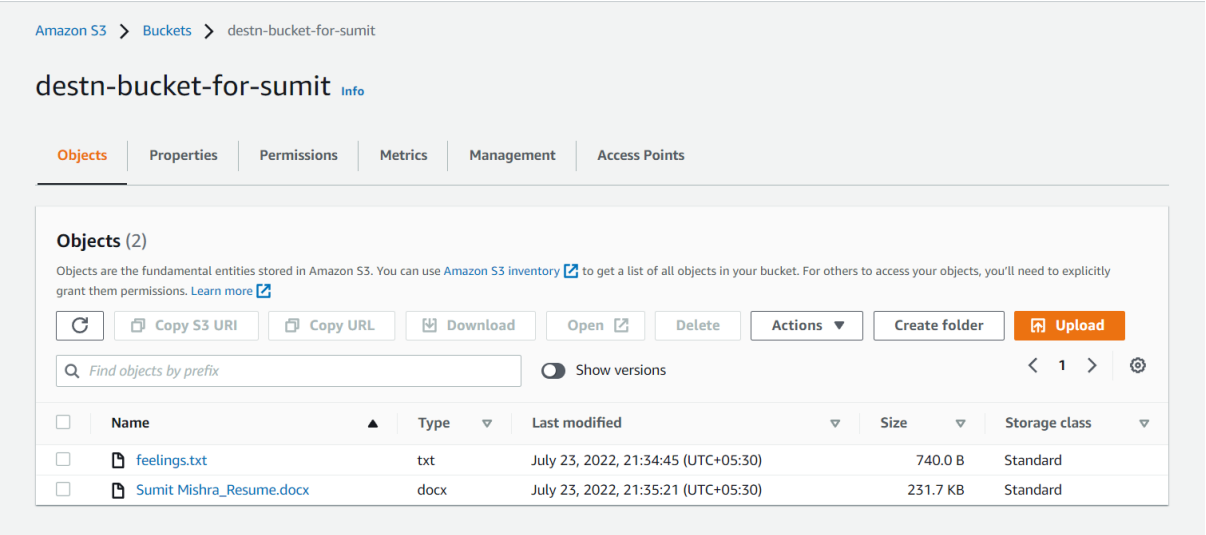
3. Uploading some files into the source bucket.



4. Syncing the files from source bucket to destination bucket using CLI.

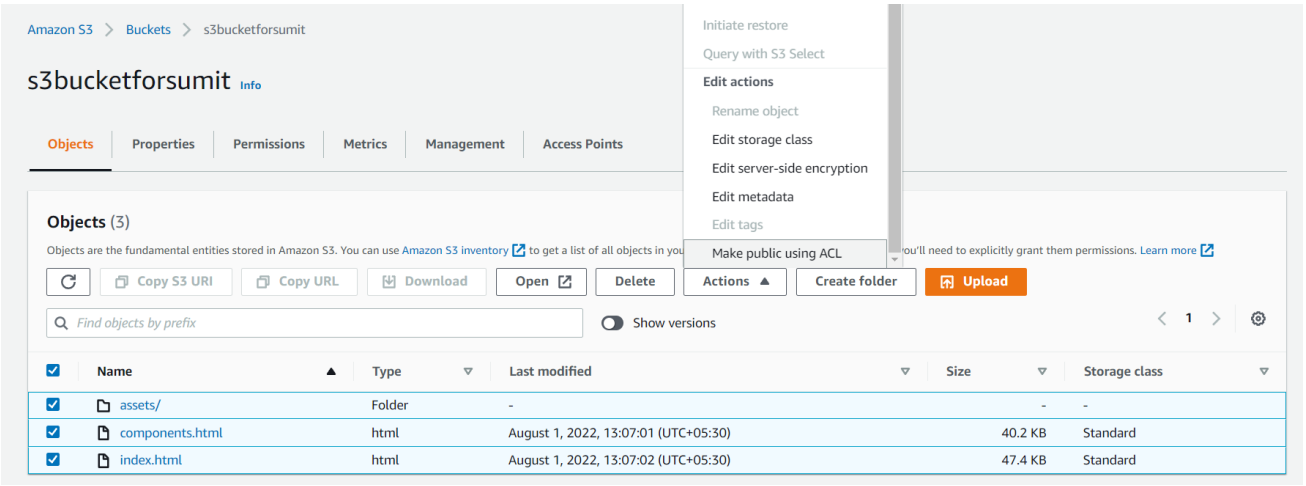
```
PS C:\Users\cr7su> aws s3 sync s3://source-bucket-for-sumit/ s3://destn-bucket-for-sumit/
PS C:\Users\cr7su>
```

5. Checking if the synced files are being reflected in the destination bucket in the s3 management console.

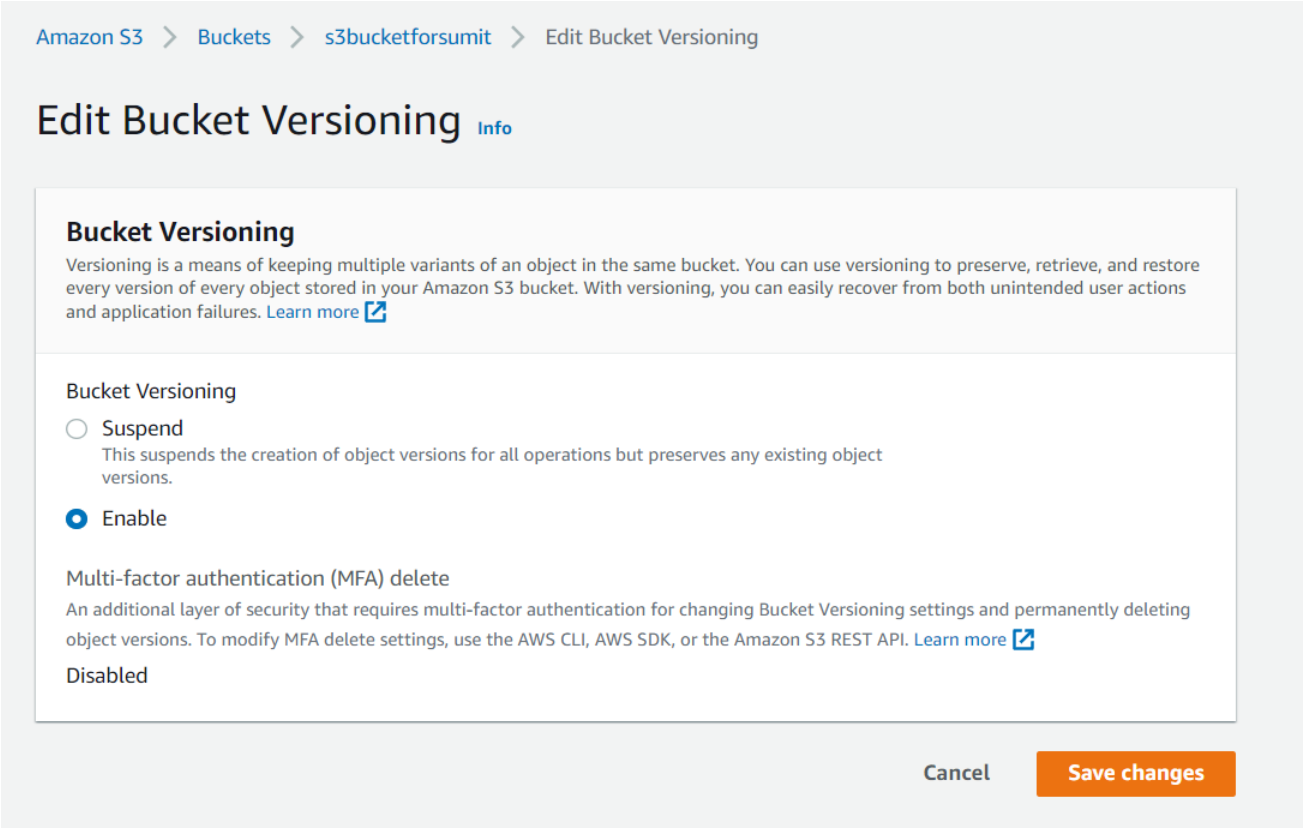


4. Mounting an s3 bucket on windows.

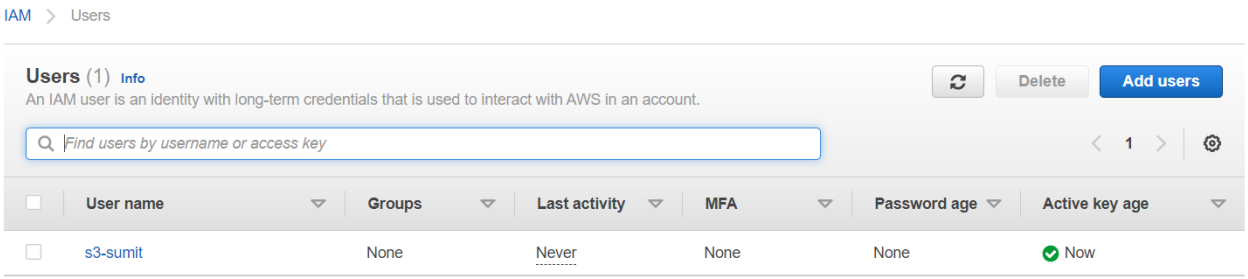
1. Create an S3 bucket on AWS console and upload some files into it.
2. Make them public using ACL.



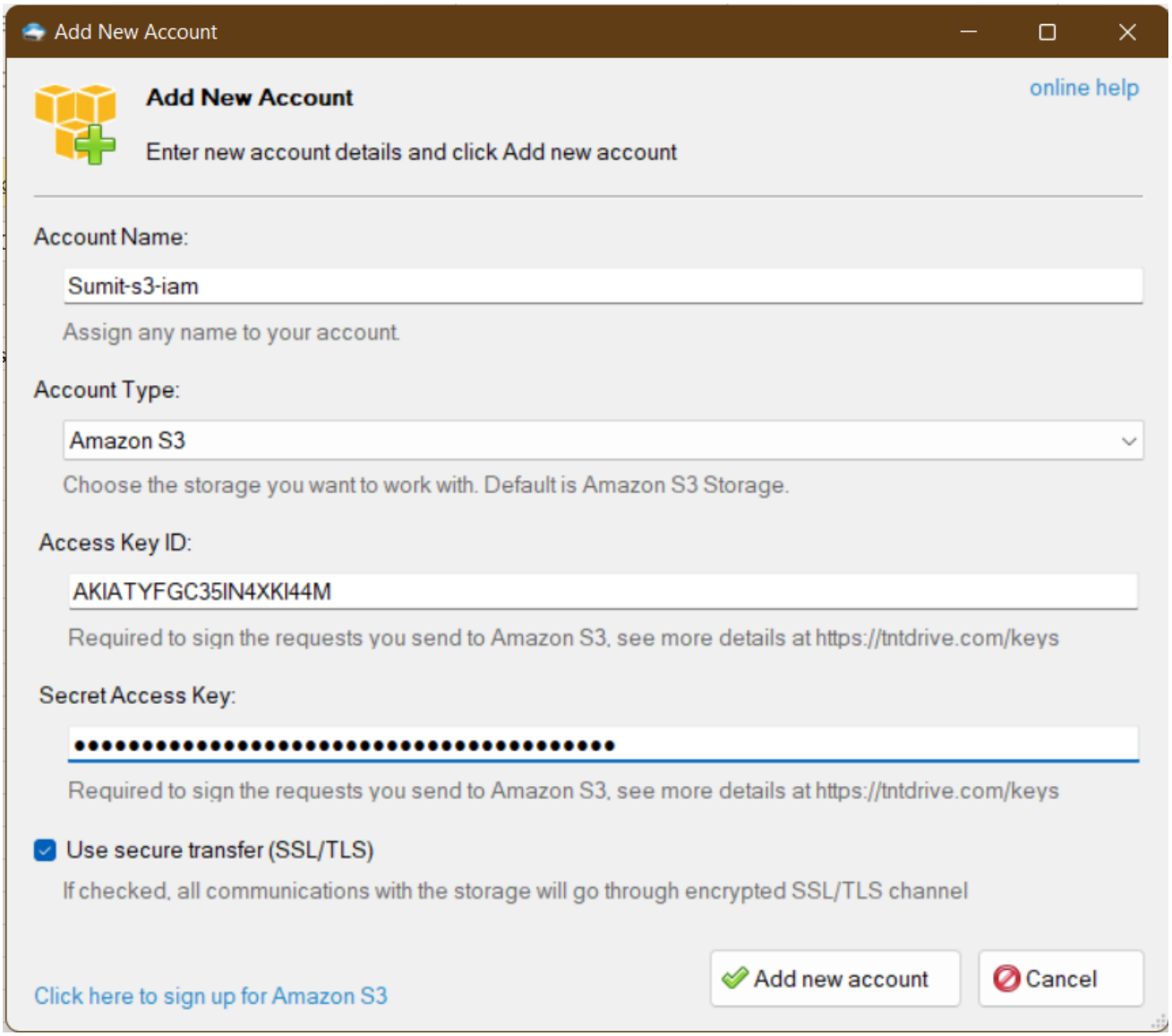
3. Allow versioning to the bucket.



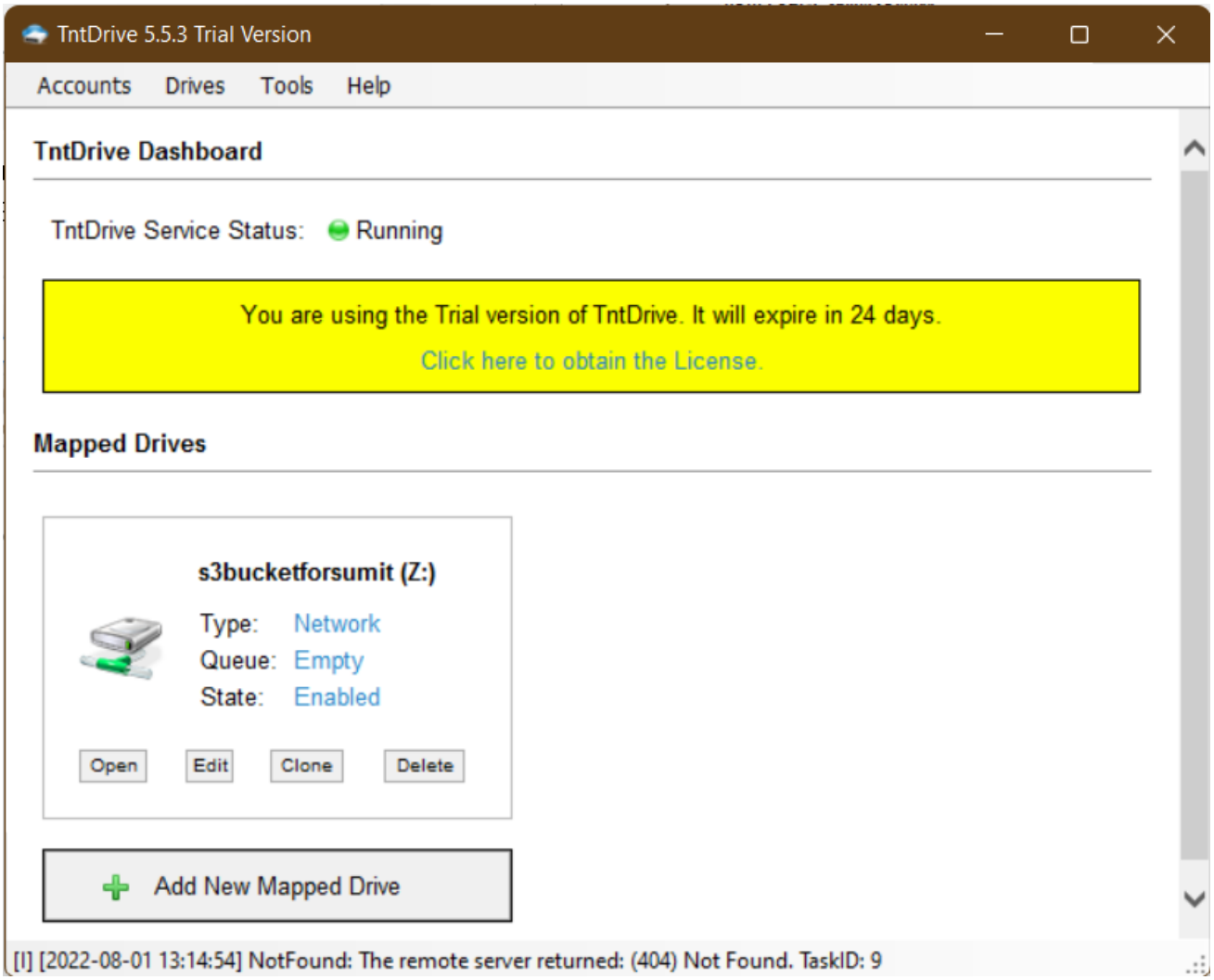
4. Download tntdrive from the internet and install it.
5. Create an IAM user for accessing the s3 bucket.



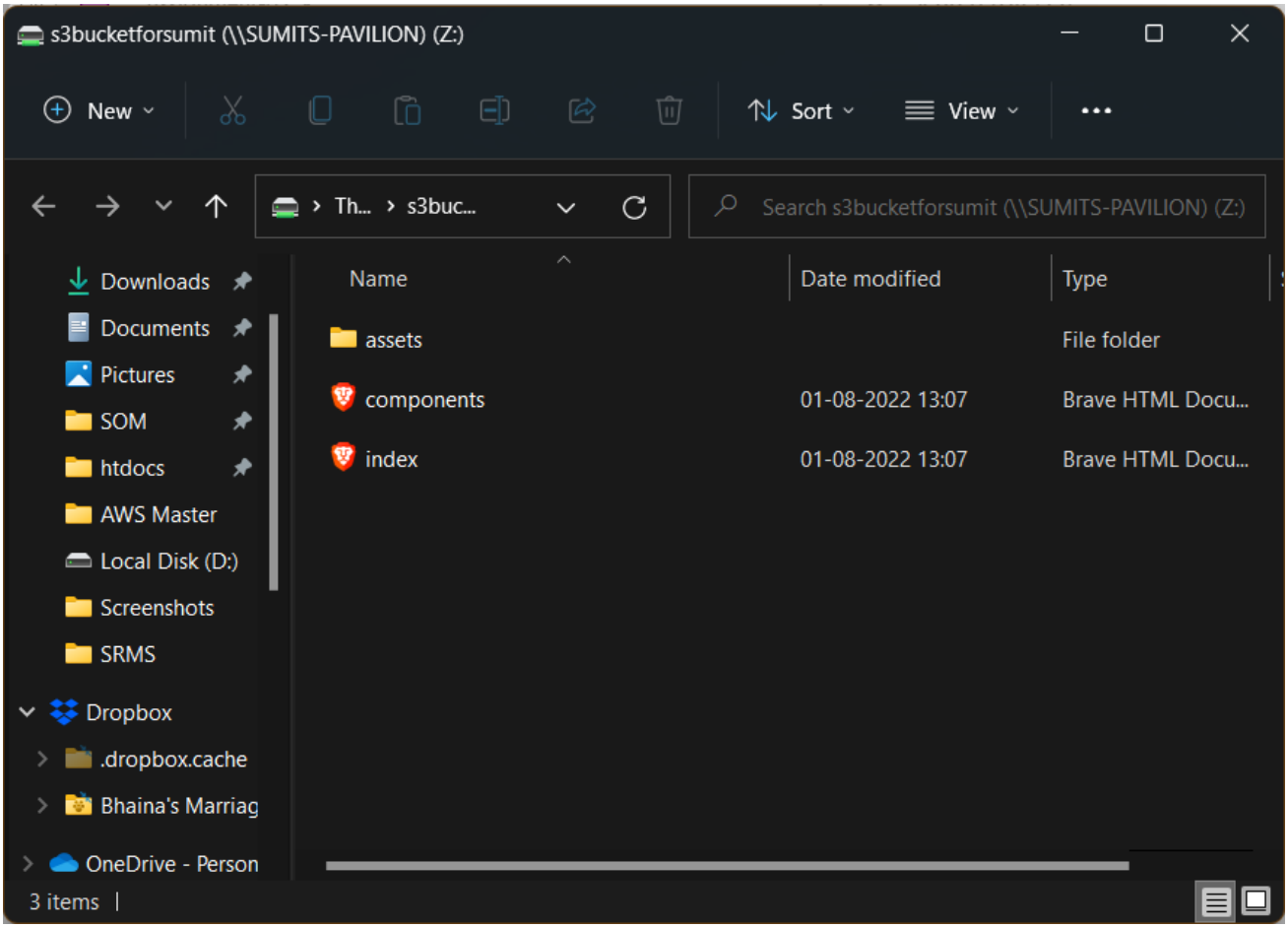
6. Open Tntdrive, select new user, provide the access key and secret access key for the IAM user created and click on add new account.



7. Then click on add new mapped drive, type the name of the s3 bucket and click on create.



8. On opening we can see the contents of the bucket on windows file explorer.



9. Adding a folder name folder_sumit, we can see that folder being reflected on the AWS S3 bucket console also.