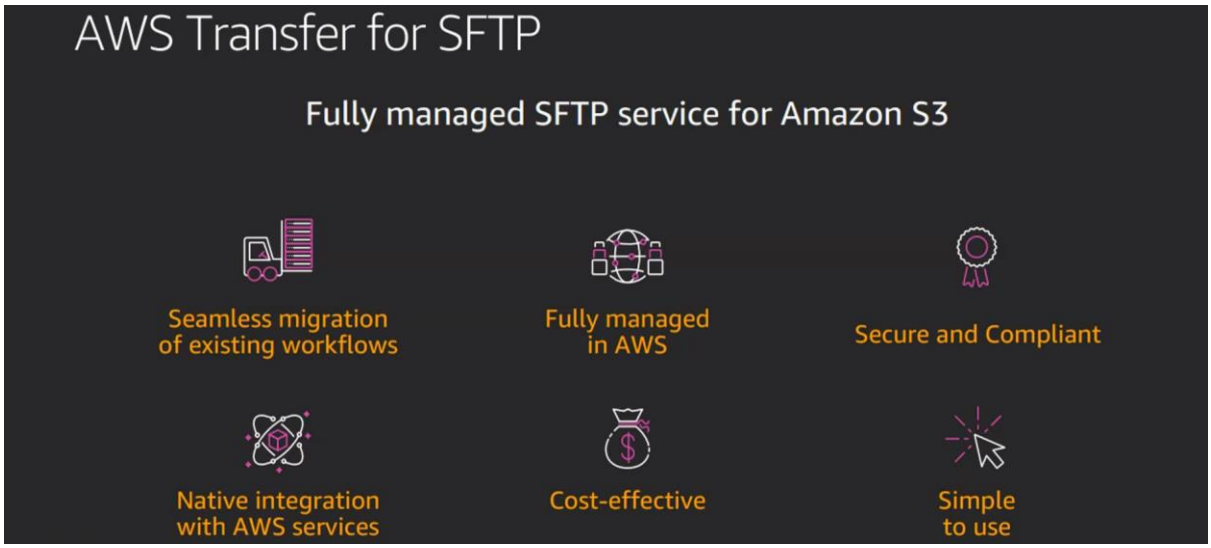
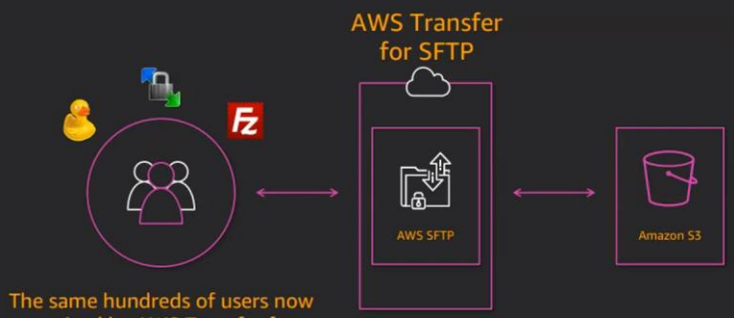
AWS - Transfer Family | Managed SFTP Service from AWS

😊 AWS Transfer Family offers fully managed support for the transfer of files over SFTP, AS2, FTPS, and FTP directly into and out of Amazon S3 or Amazon EFS. You can seamlessly migrate, automate, and monitor your file transfer workflows by maintaining existing client-side configurations for authentication, access, and firewalls — so nothing changes for your customers, partners, and internal teams, or their applications.



1. If we use this service our data actually store in S3 bucket.
2. Using S3 is reliable because multiple copies can be maintained.
3. AWS transfer service is managed we don’t need to worry about any servers.

Your SFTP architecture now



The same hundreds of users now serviced by AWS Transfer for SFTP, without requiring changes

+ Fully managed servers
+ No end-user disruption
+ Enterprise-ready
+ Simple to use
+ Pay as you use
+ Native cloud integrations

Key features

Time saved from managing SFTP servers

Automatically scales to meet your needs in real-time

Redundant across **Availability Zones** in Region

Store data in your **Amazon S3** buckets for archiving, processing, or analyzing

Automate post-upload processing with Amazon **S3 events**

Control end user access to resources by using **IAM**

Encrypt your data using server-side encryption using **Amazon S3 or AWS KMS**

HIPAA-eligible and PCI-compliant

Encryption at rest options such as SSE-S3 or SSE-KMS

End-user activity tracking in Amazon CloudWatch

SFTP endpoint fee @$0.30/hour

SFTP uploads and downloads @$0.04/GB of transfer

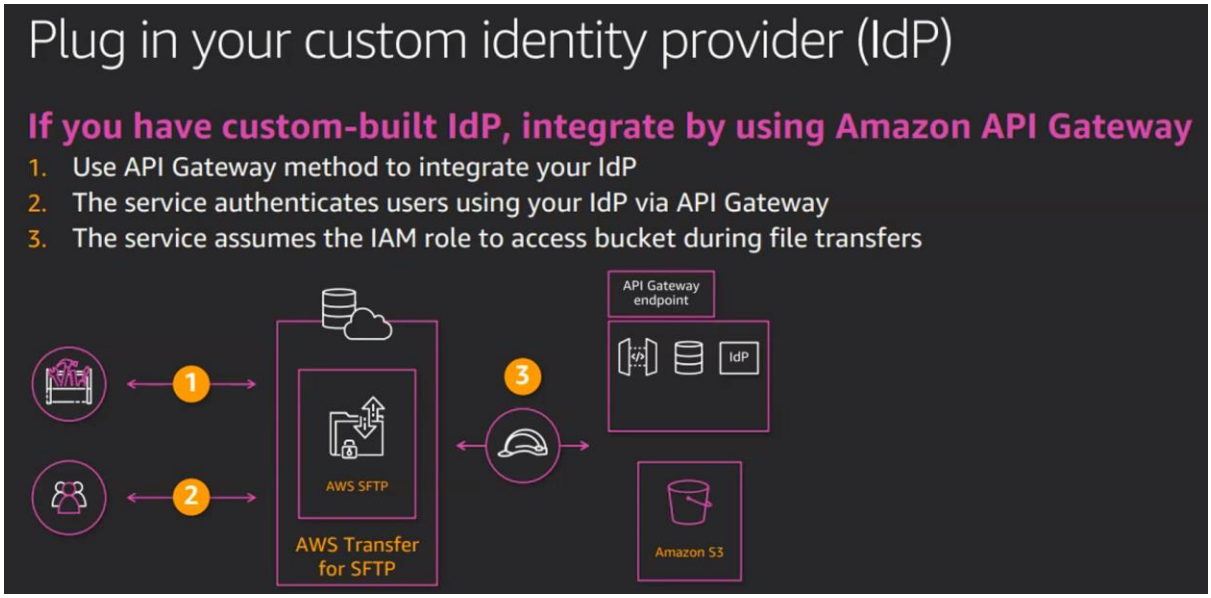
Easy to set up and configure using the AWS Management Console or service API

No IT expertise required for SFTP server or user access configuration

Service managed user access

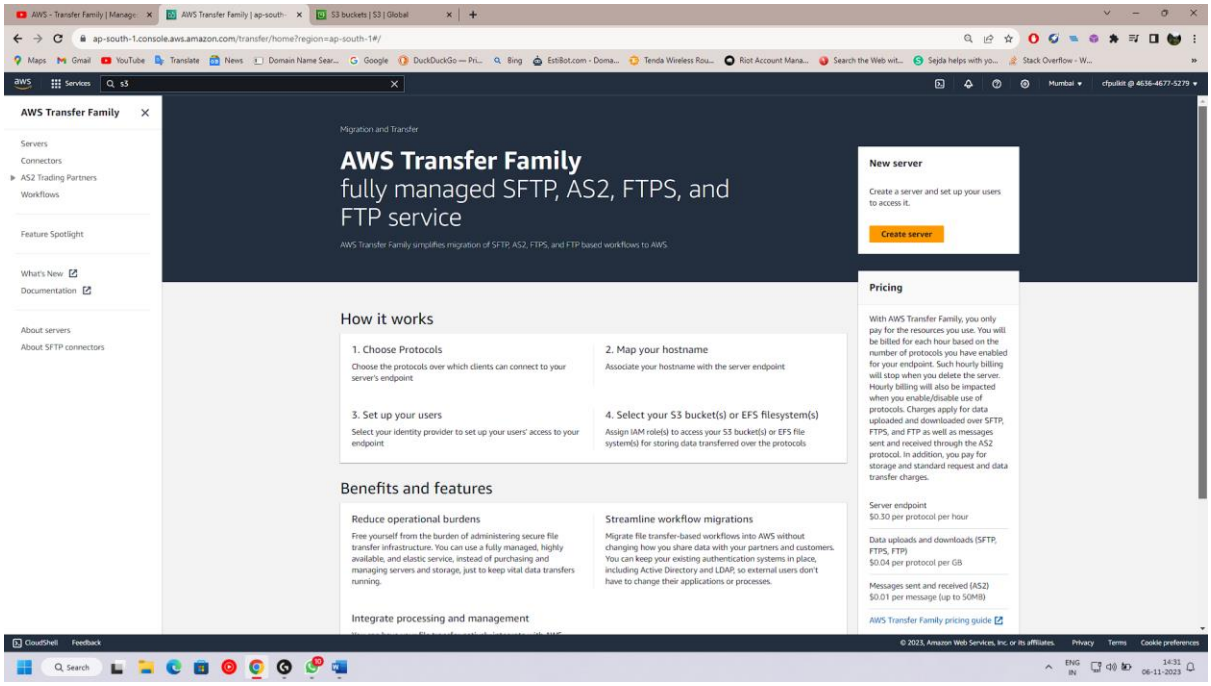
Store and manage user identities and keys inside the service

1. Configure your users credentials and keys using the AWS Management Console
2. Users serviced using their existing clients and credentials
3. Amazon S3 accessed by using IAM during file transfers

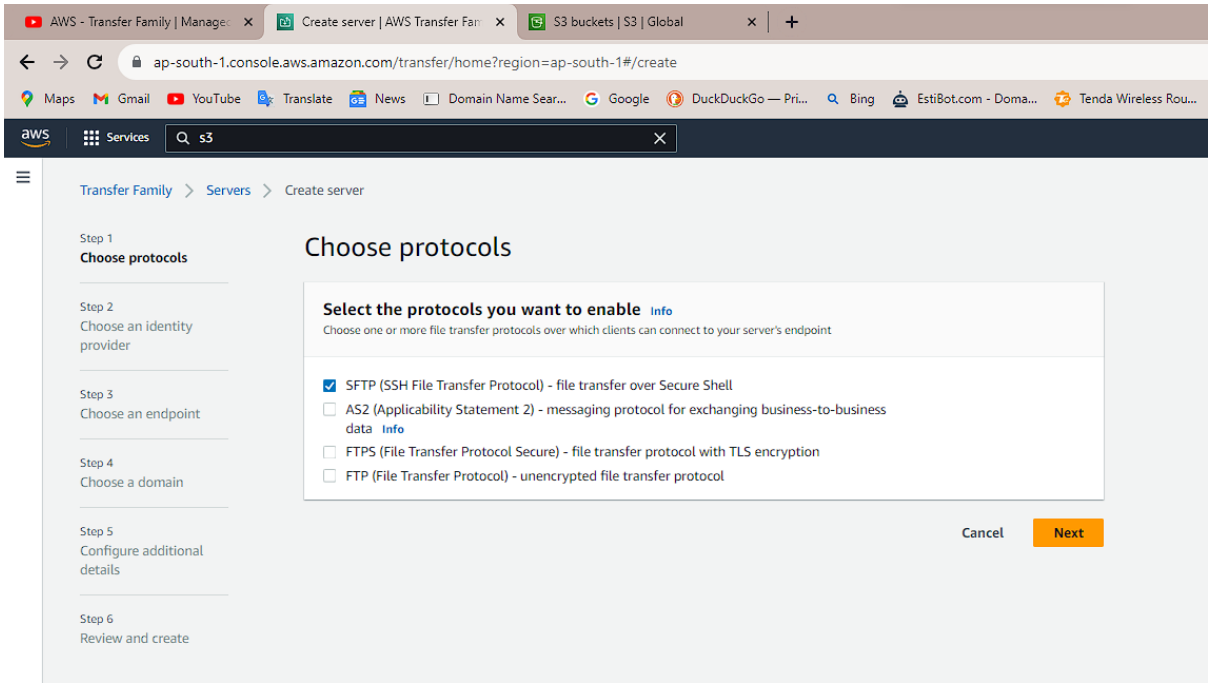


STEP 1:

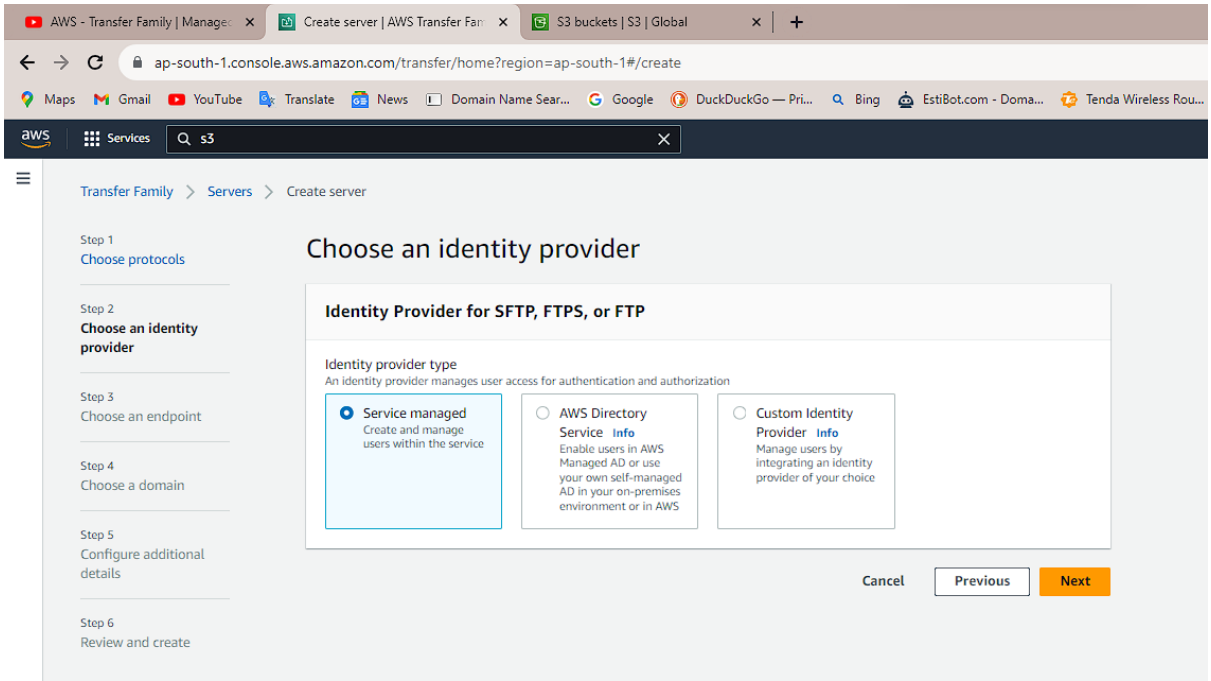
1. Log in to AWS Console.
2. Go to S3 and create 2 buckets or use any empty bucket if available.
3. Then search AWS Transfer Family and create a server.



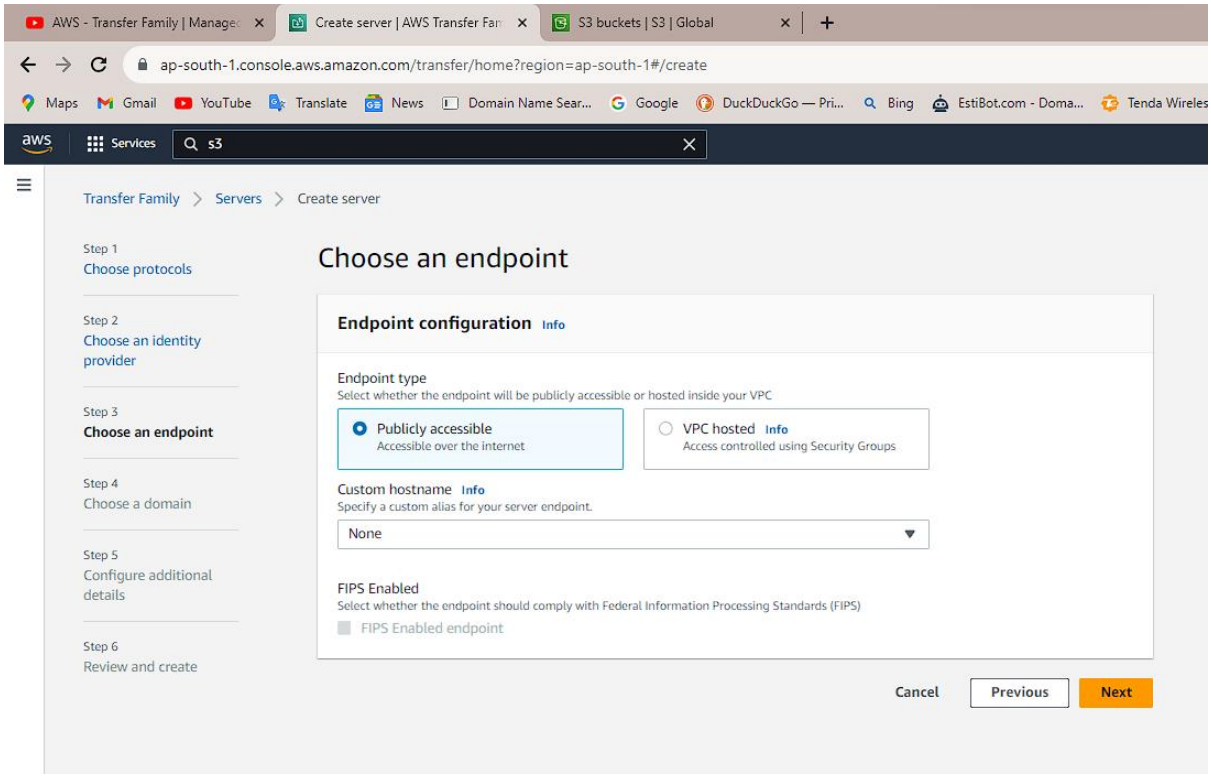
4. So here we get more than one option for the time being we'll use only SFTP.



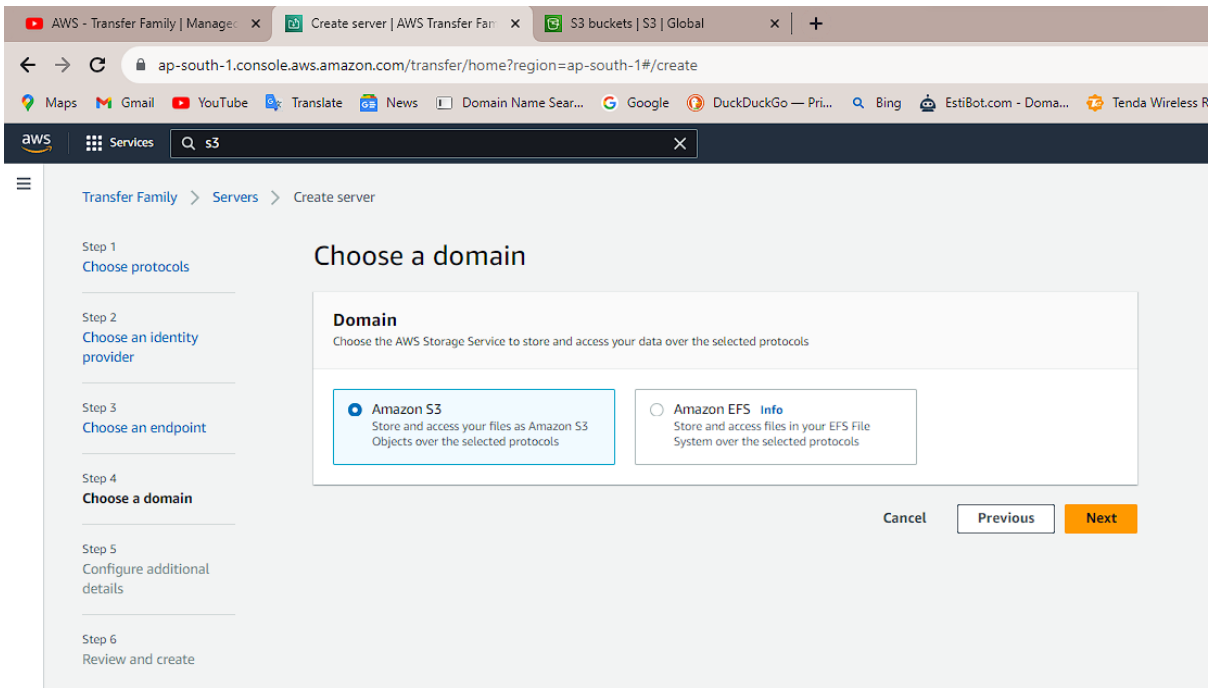
5. On the next page select Service Managed.



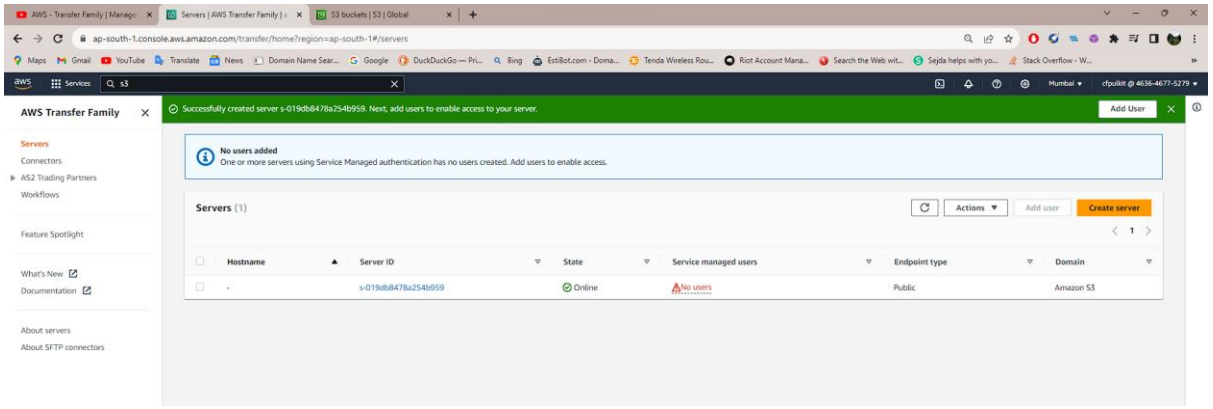
6. The next page we've two options either we can select publicly accessible or VPC hosted. We can use VPC only if we want to keep our data private but for now we'll go with publicly accessible.



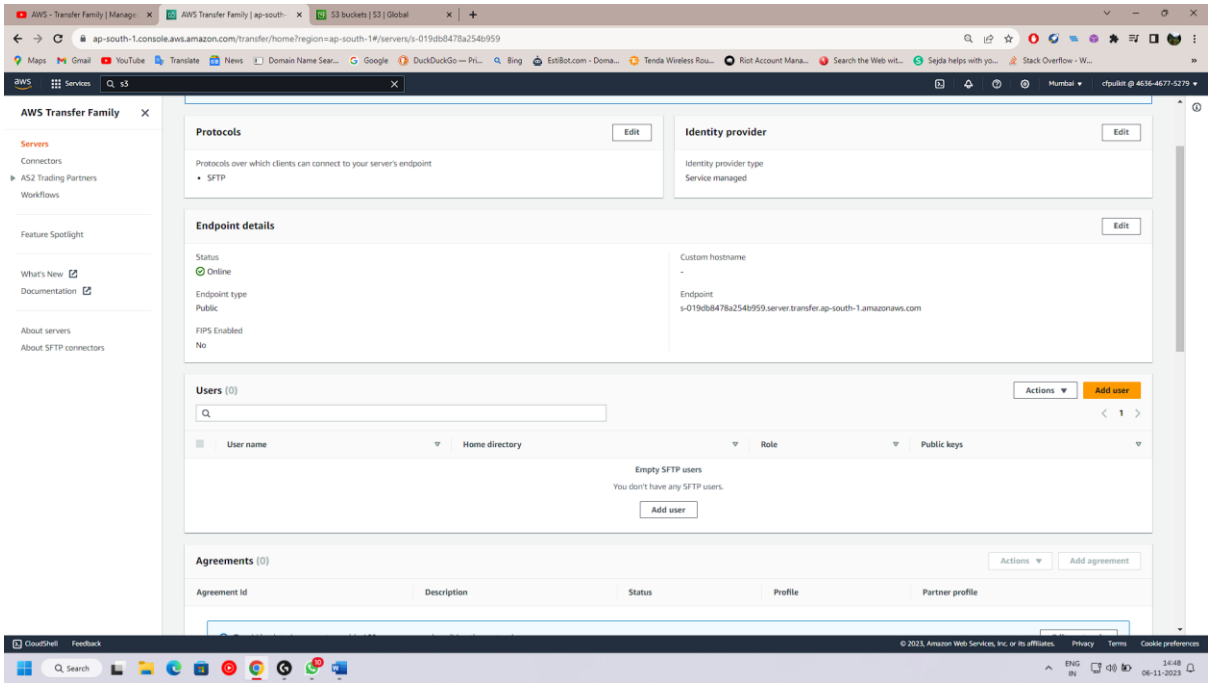
7. So, here we see two options again and we're going to select Amazon S3 because we're going to upload and download our files from here.



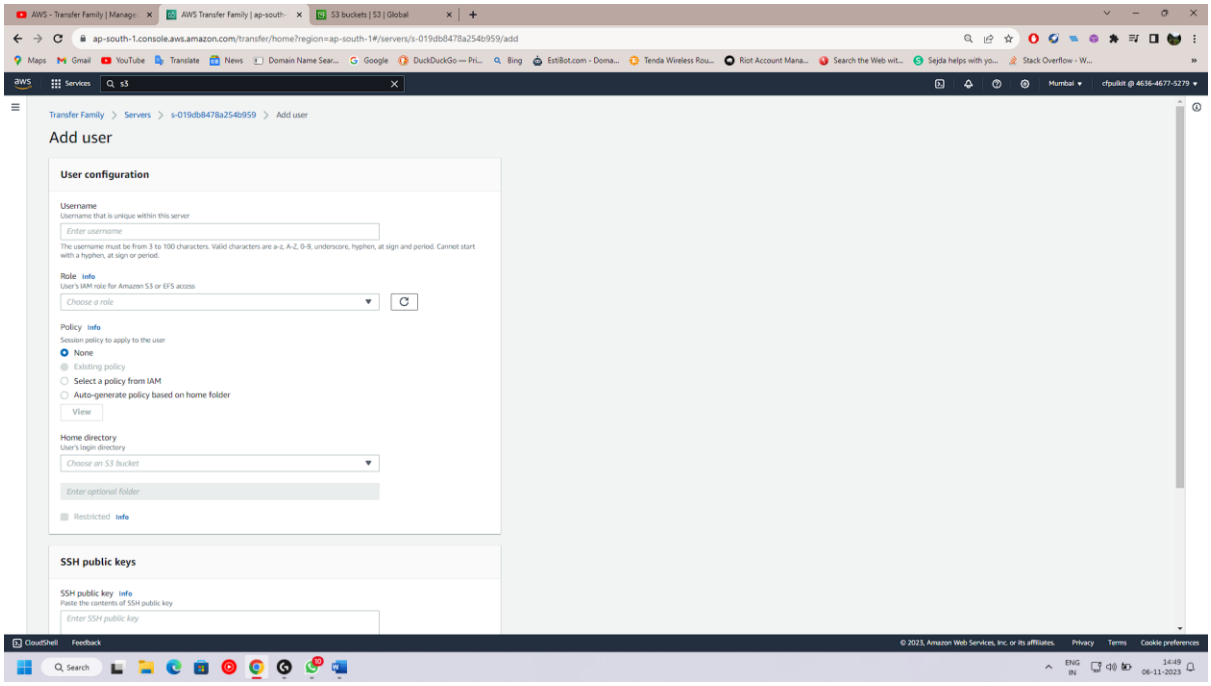
8. The next we can keep as default and move to the last page where we'll review the server details and then create server and wait for it to start and wait for the server to get online.



9. Once the server is online then we need to Add users to this.

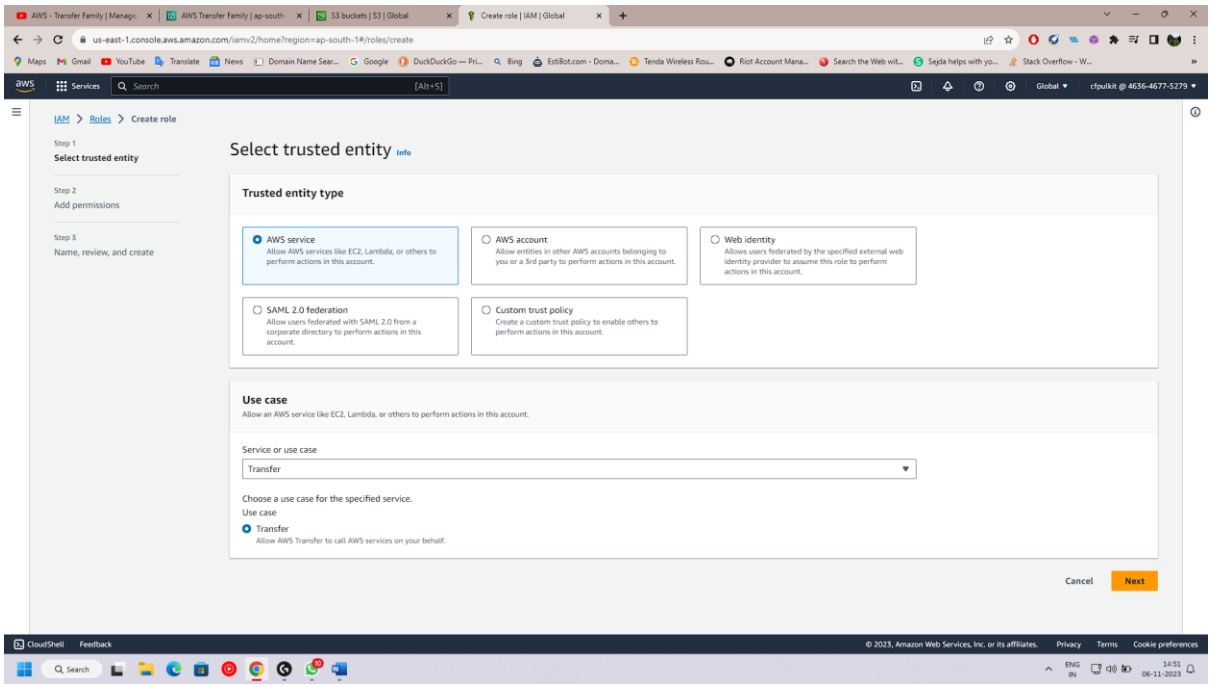


10. We'll see this page on our screen and we've to give it a name then we need to create a IAM role for this to move forward.

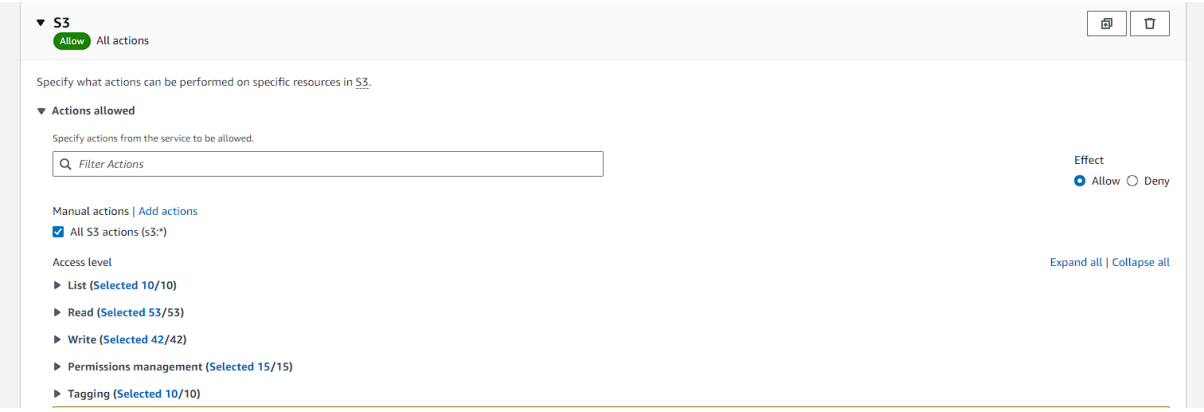


CREATE an IAM ROLE

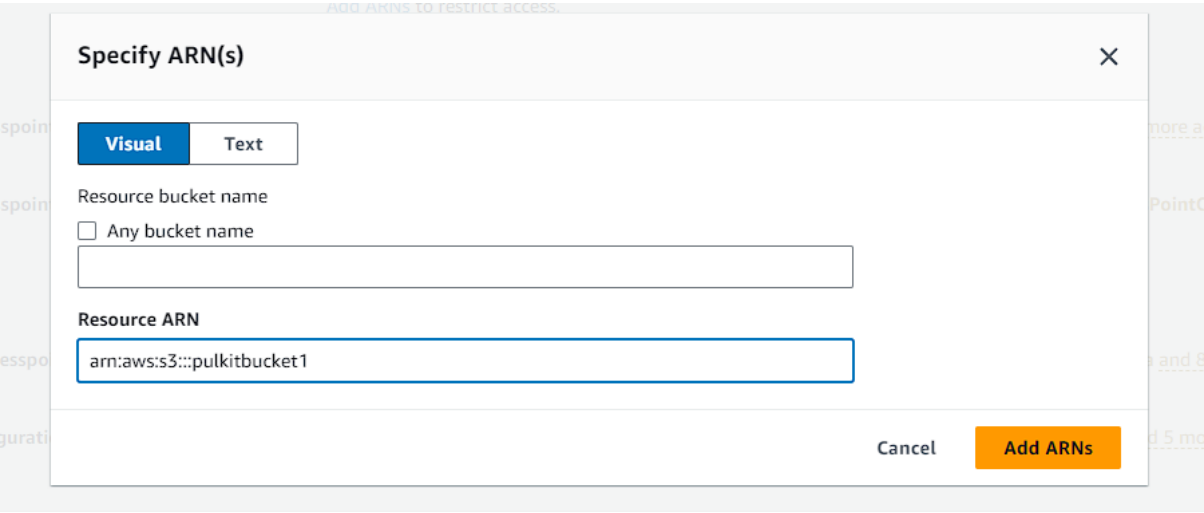
1. Go to IAM and create role select Use Case as “Transfer”



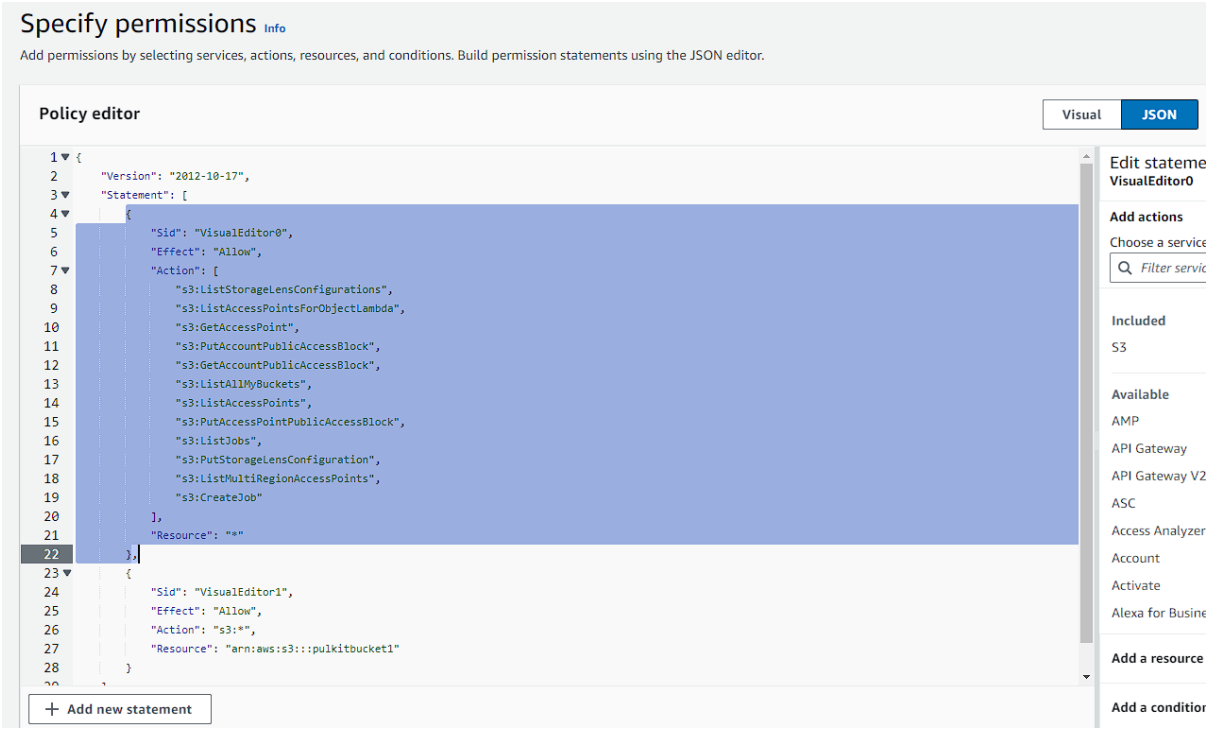
2. Before giving it the permission we need to create a policy of our own.
   a. First go to policies and click on create policy.
   b. Select service as S3 and tick on all S3 actions.



c. Then scroll down a little on the resources section select add ARN to bucket.

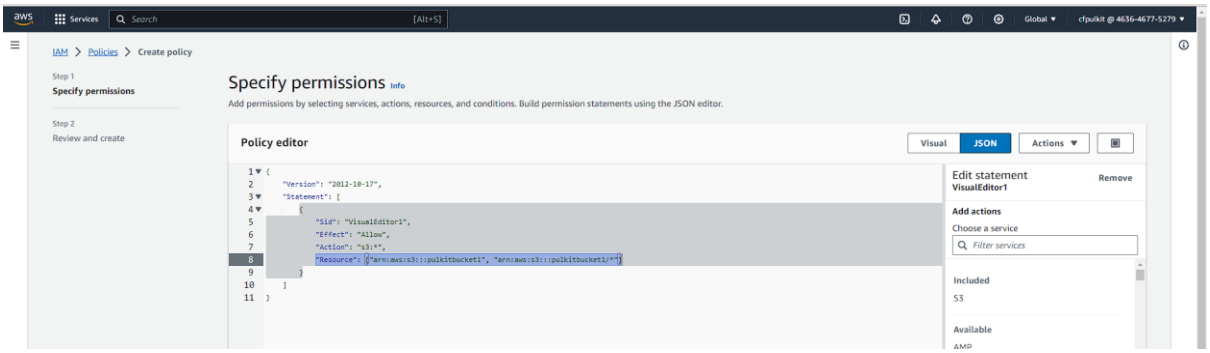


d. Then go to the top and select JSON on it to change something in the code and delete the section marked as blue below.



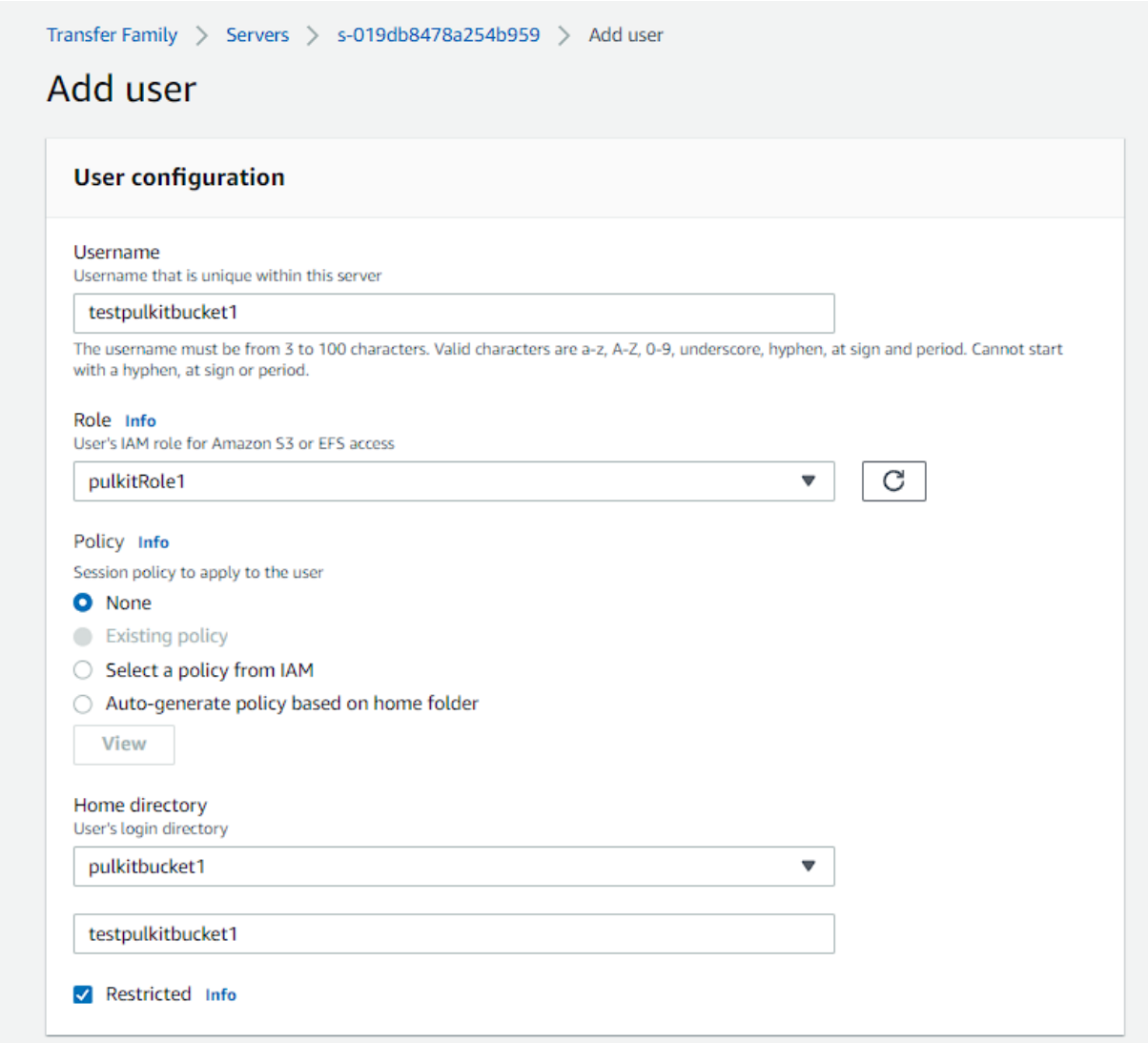
e. Then in the last add this line to the code and change the bucket name accordingly to yours.

"Resource": ["arn:aws:s3:::pulkitbucket1", "arn:aws:s3:::pulkitbucket1/*"]



f. Give the policy a name and then go back to roles then add two permissions.
g. First S3 Read only Access and the policy we created.
h. Then create role.

Go back to Amazon Transfer Family and add user now.

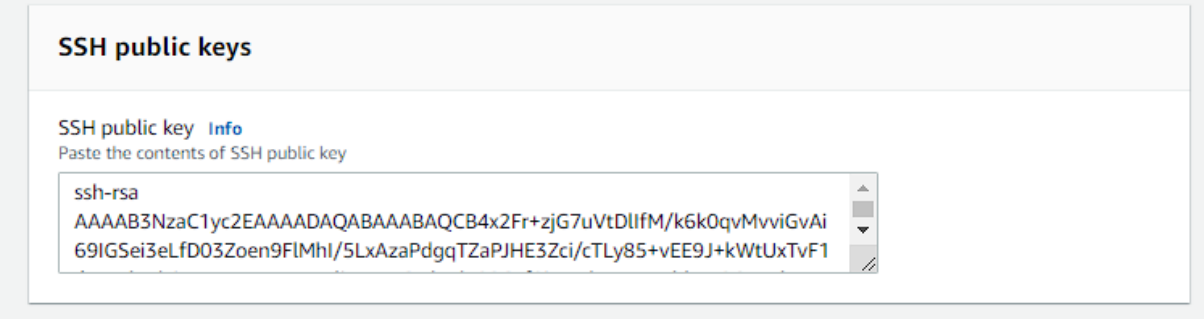


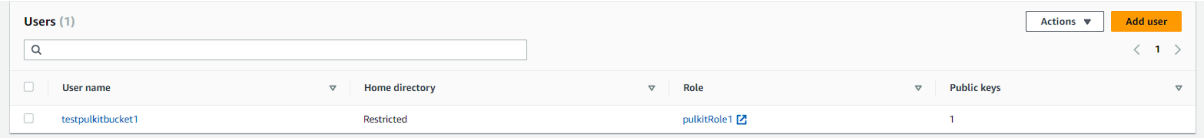
• Above we can see that we gave it a name then we selected the role for it which we created and then on the home directory select the bucket which we want to have the data for the demo.

😊 As we can see, I've selected Restricted, which implies that if we select this option, the user won't be able to navigate anywhere else.

😊 After this we need to give it a SSH Public Key which we can create through a online website. Then add user.

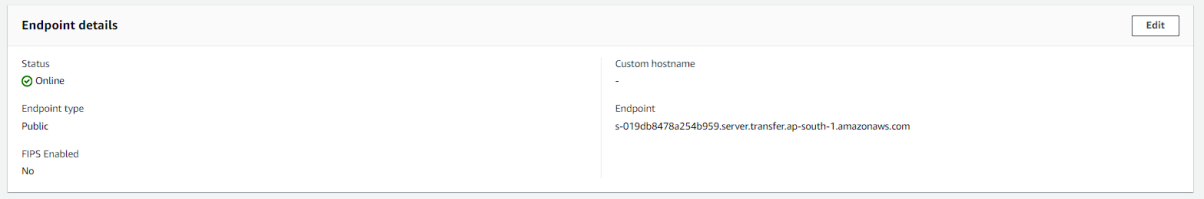
https://8gwifi.org/sshfunctions.jsp



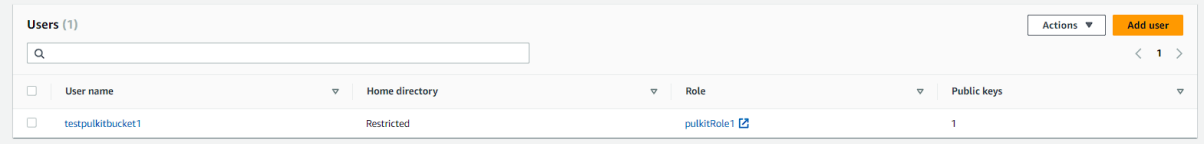


• Then we need to open Win SCP app which we can download it through the internet. In this app we need to give it the log in credentials which we can find on the server.

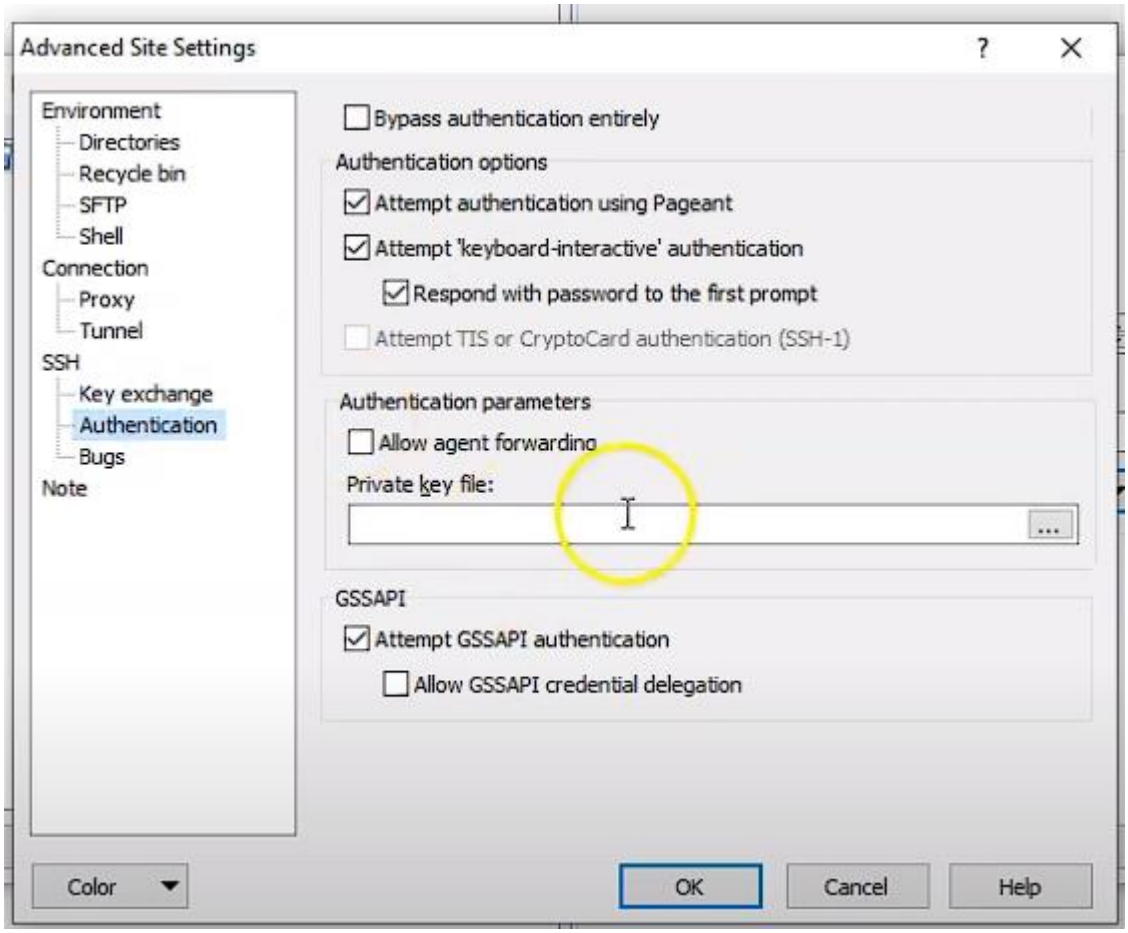
1. We need to copy the end point to the host name.



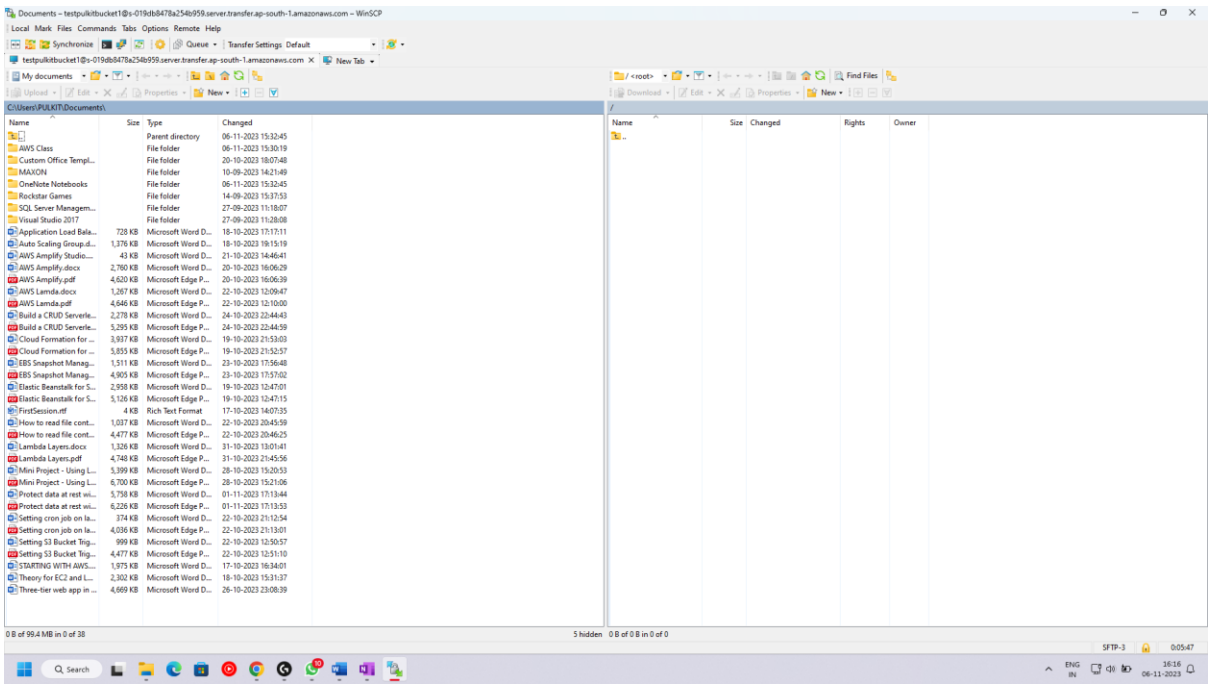
2. Then we need to write our username



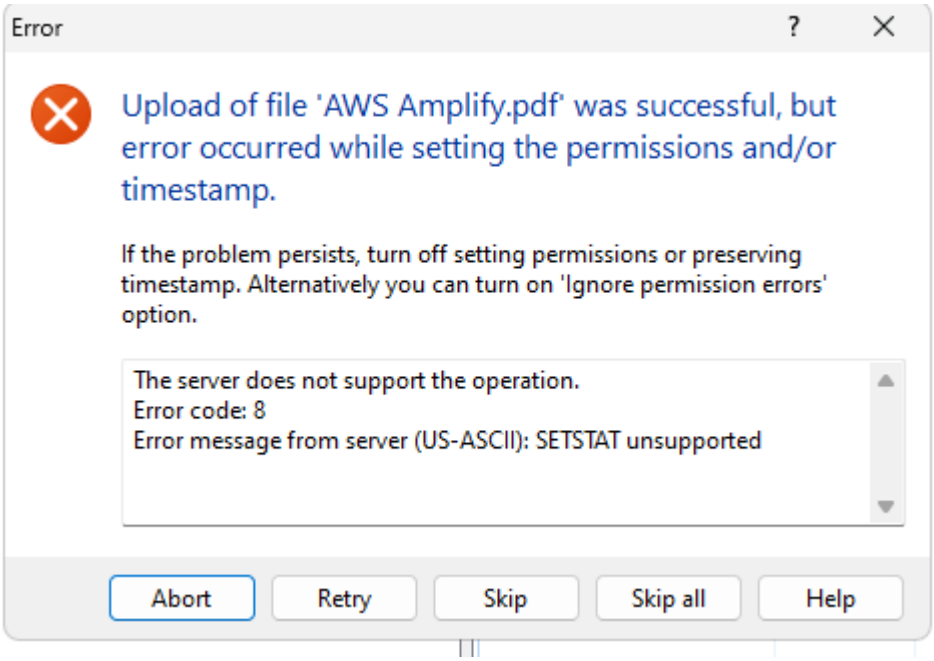
3. For the password go to advanced settings and select the private key file from the system and run it.



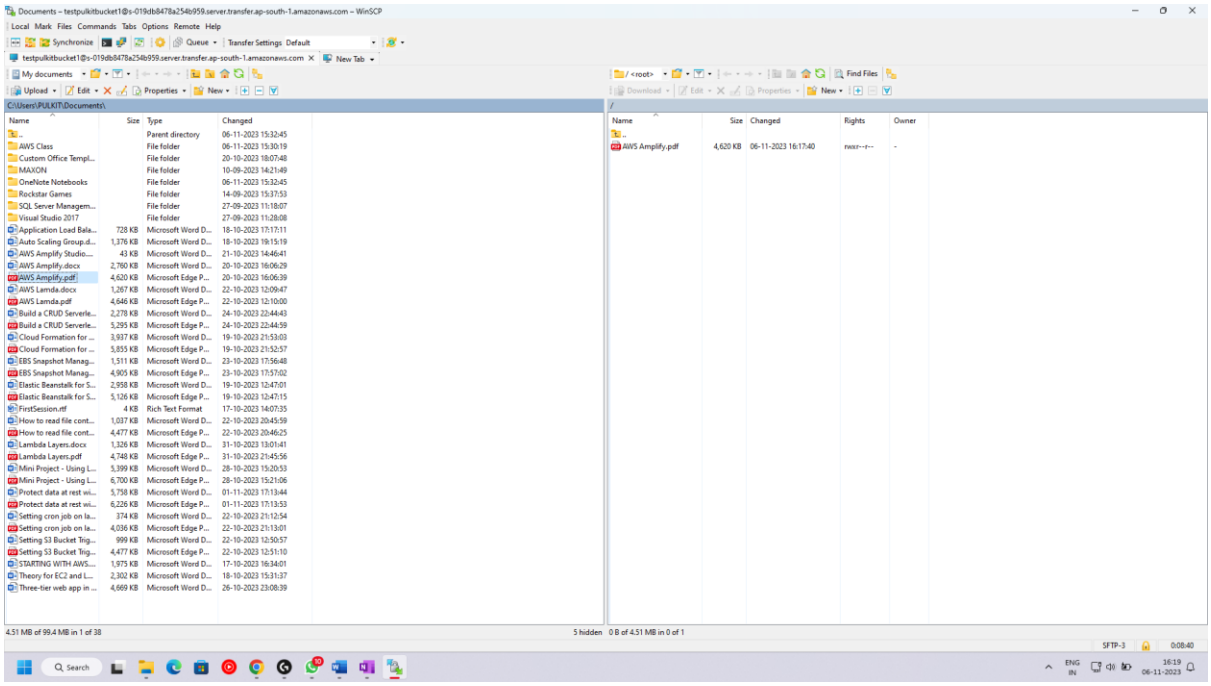
4. Here we'll see two windows side by side left side is our window and the right side is root window or S3 part.



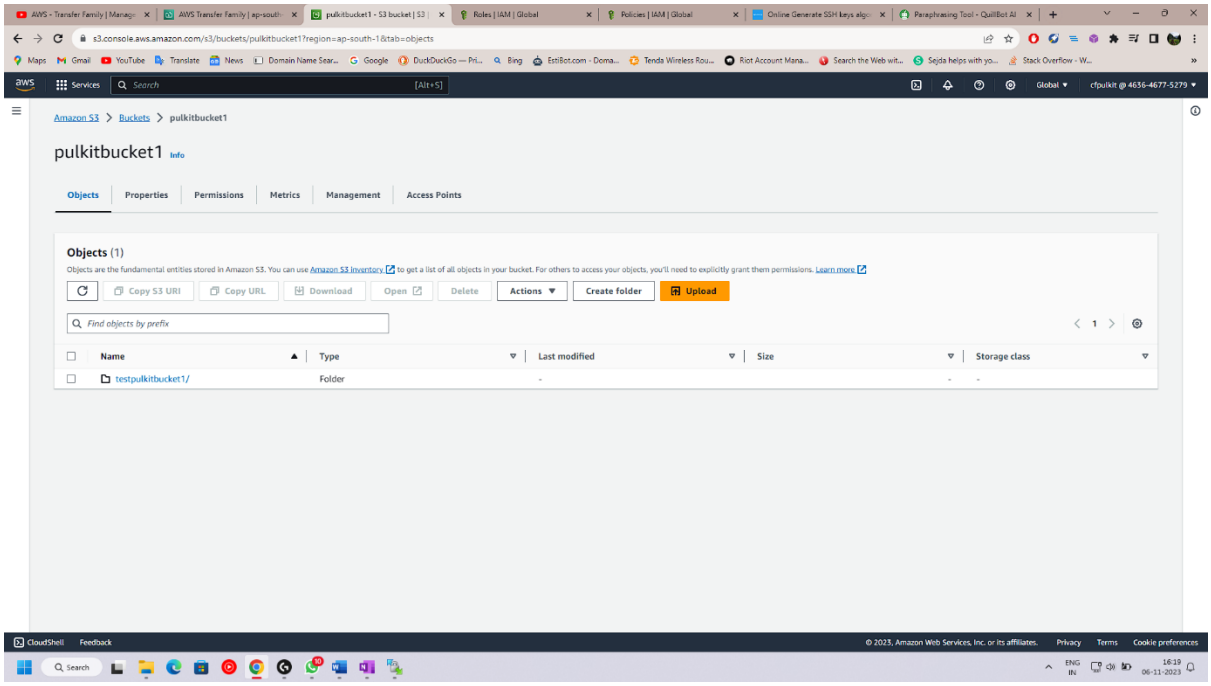
5. While uploading we'll see a error but just skip it because our file is uploaded

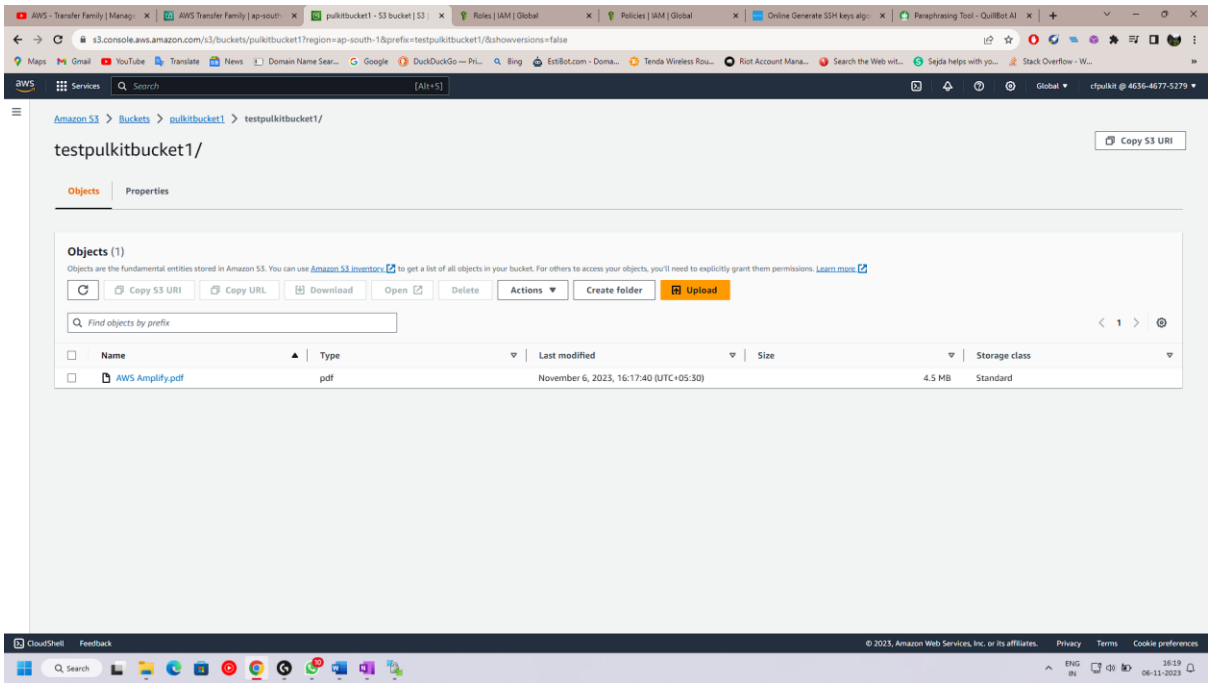


6. Hence, we can see that our file is uploaded

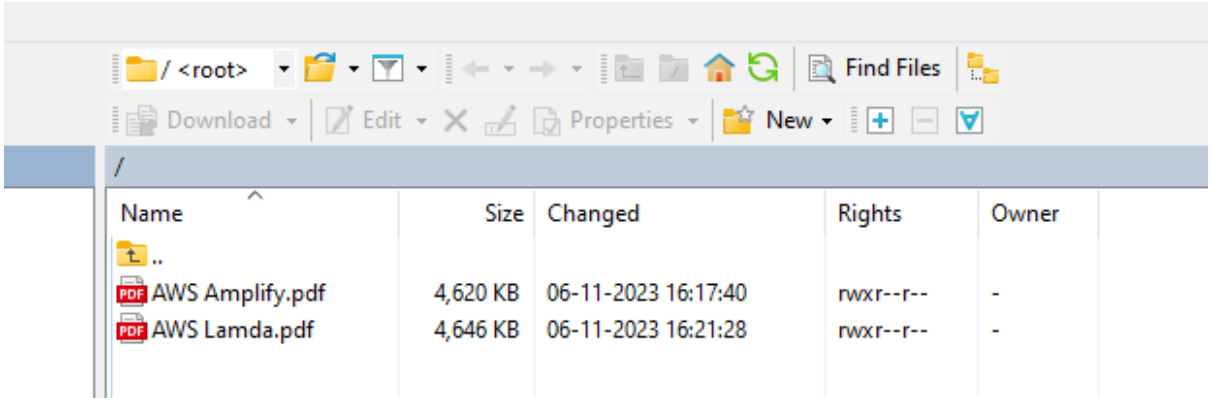


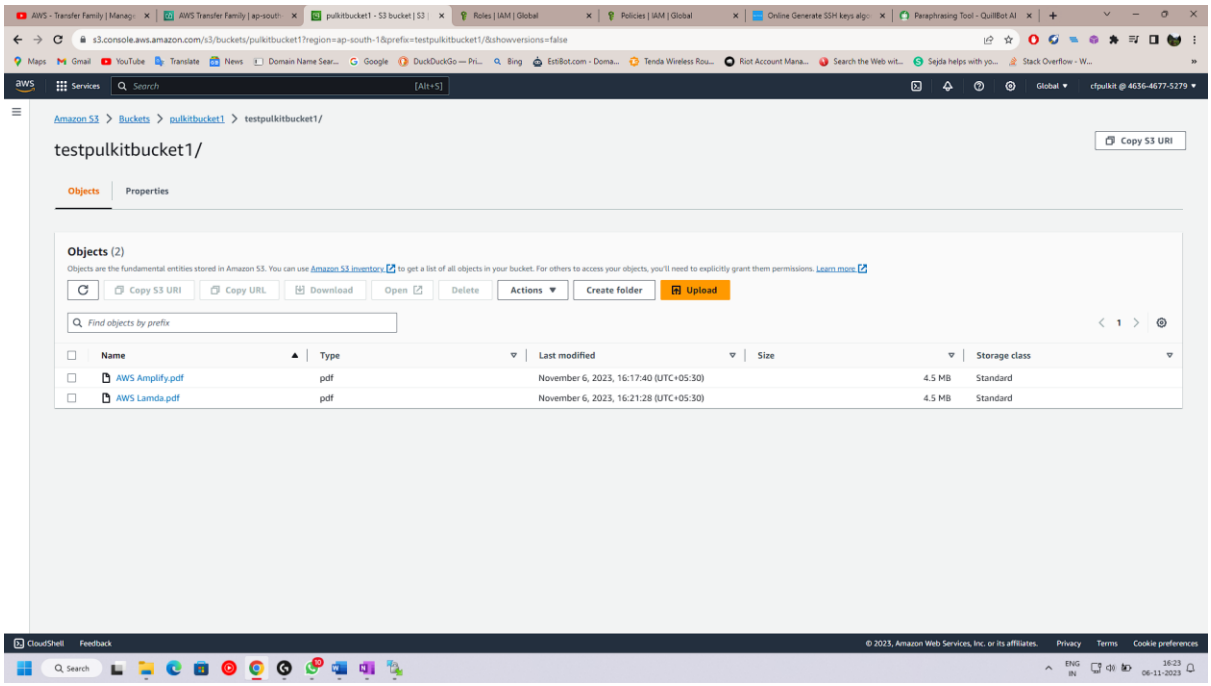
7. And now if we go to S3 we can see the same file is uploaded is here too.



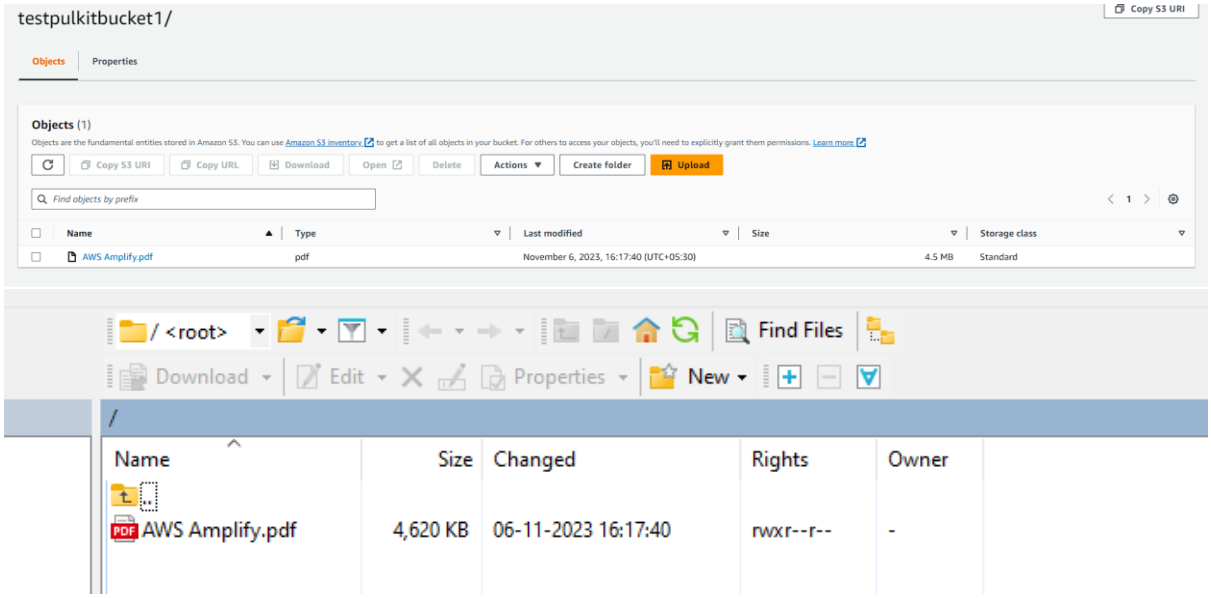


8. Now if we upload another file in Win SCP to see if it is working properly or not. If it gets uploaded then the application is working fine and we can proceed further. The same file is now available in S3 bucket too.





😊 Therefore, if we would like to verify the deletion process, we can do so by deleting a file from S3, which would also destroy it from Win SCP.



😊 The catch is that we can see every bucket that is available in the S3 if we go to the AWS transfer family, modify user configuration, deselect "Restricted," save it, and open a new Win SCP session.