

Name : Nakul Lahode

Task - 04 : Implement IAM policies, secure storage, and data encryption on AWS with configured security policies and a report detailing the setup .

Overview of AWS IAM :

1. User and Access Management

- IAM allows you to create and manage users, groups, and roles, and control their access to AWS services.

2. Policy-Based Permissions

- Access is granted through JSON-based policies that define what actions are allowed or denied on specific AWS resources.

3. Least Privilege Principle

- IAM supports fine-grained control, ensuring users only have the minimum permissions needed to perform their tasks.

4. Multi-Factor Authentication (MFA)

- Enhances security by requiring users to provide two forms of authentication (e.g., password + OTP).

5. Temporary Access via Roles

- IAM roles allow secure, temporary access to AWS services for users, applications, or services (e.g., EC2 instances, Lambda functions).

6. Integration with AWS Services

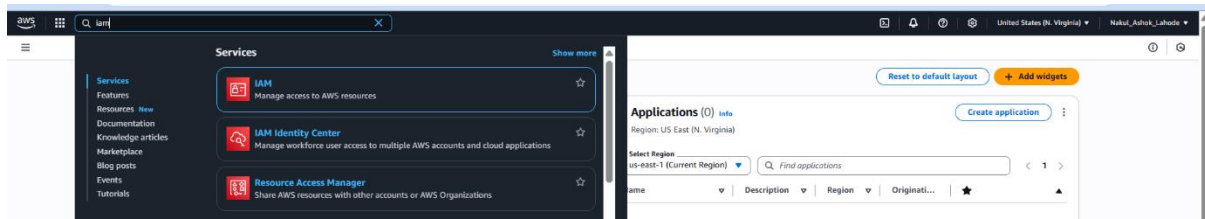
- Works seamlessly with S3, EC2, RDS, Lambda, and other AWS services to secure and manage resource access.

7. Federated Access Support

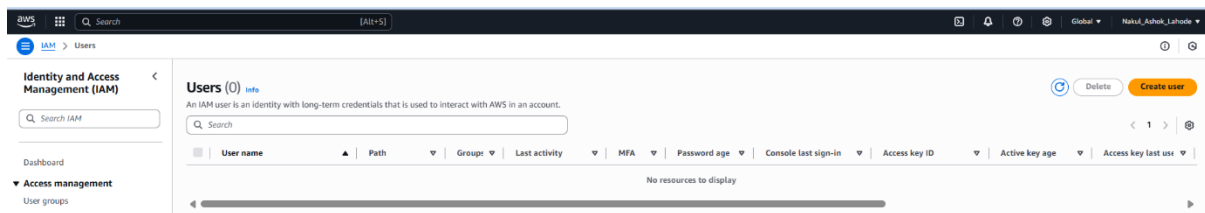
- Supports identity federation with external identity providers (like Google, Active Directory, SAML) for SSO (Single Sign-On) access.

Step 1: Create a New IAM User

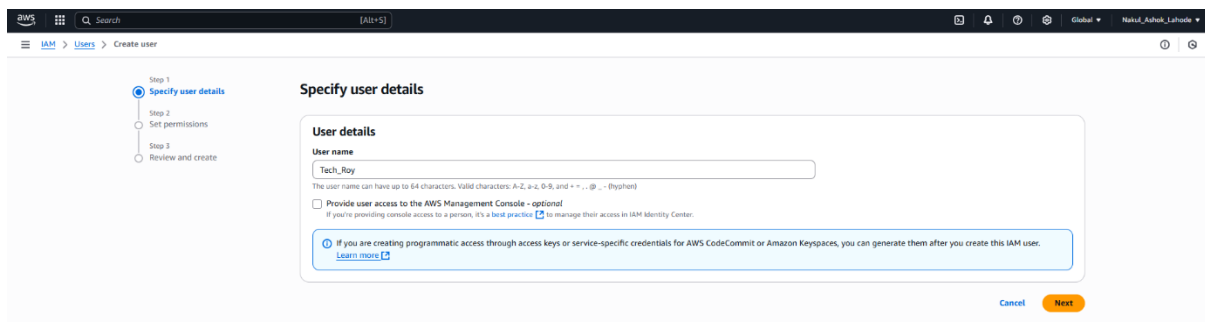
1. Navigate to the IAM service in the AWS Management Console.



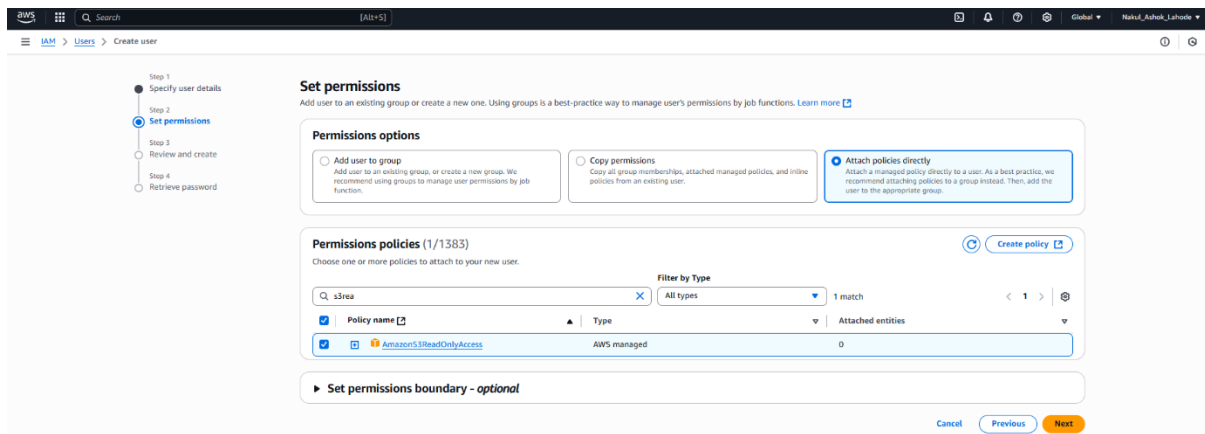
2. Click on "Users" > "Add users".



3. Enter a suitable username.

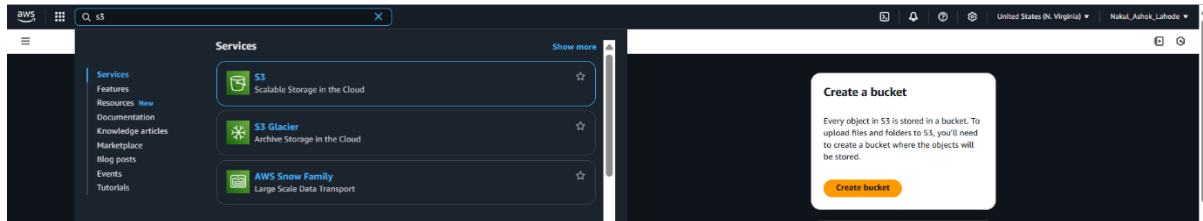


4. Choose the appropriate permissions setting (initially, no permissions or add temporary ones).
5. Review the configuration and click Create User.

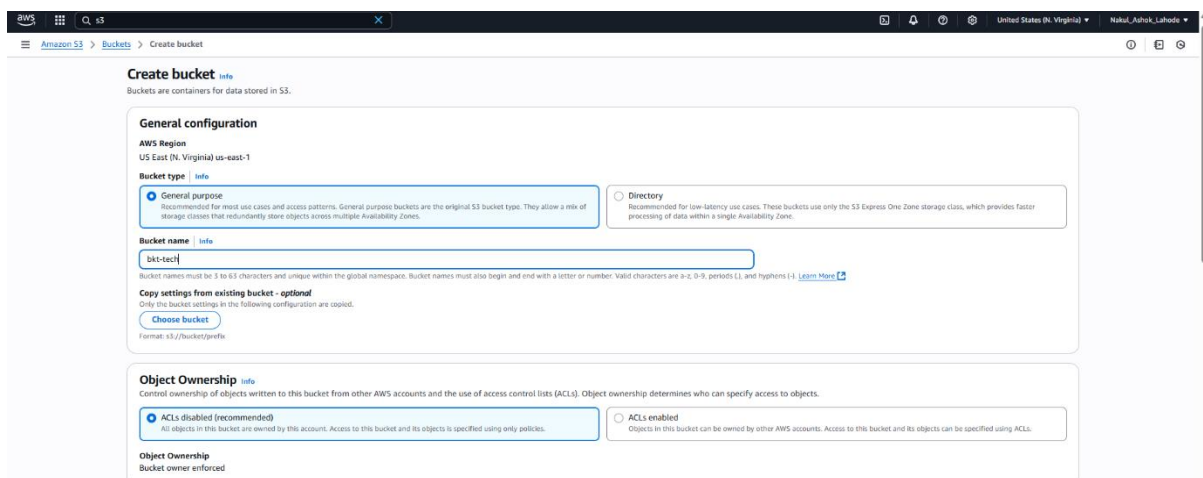


Step 2: Create an S3 Bucket with Encryption

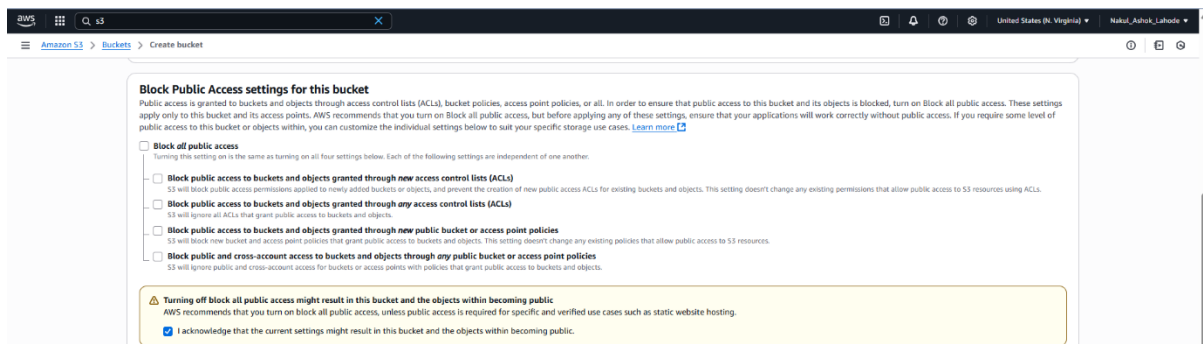
1. Go to the **S3** service.



2. Click **Create Bucket** and give it a unique name.



3. Optionally, enable or disable **Block Public Access** depending on your requirement (in this case, allow access for testing IAM).
4. Under **Bucket Settings for Default Encryption**:
 - Choose **"Enable"**.
 - Select **"Amazon S3 managed keys (SSE-S3)"** for server-side encryption.



Default encryption [info](#)
Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type [info](#)

- ☒ Server-side encryption with Amazon S3 managed keys (SSE-S3)
- ☐ Server-side encryption with AWS Key Management Service keys (SSE-KMS)
- ☐ Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)

Secure your objects with two separate layers of encryption. For details on pricing, see DSSE-KMS pricing on the Storage tab of the [Amazon S3 pricing page](#).

Bucket Key
Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#)

☐ Disable

☒ Enable

5. Click **Create Bucket**.

Step 3: Create a Custom IAM Policy

1. In the IAM dashboard, go to Policies > Create Policy.
2. Choose the JSON tab and paste your custom policy (e.g., restrict access to specific bucket).

The screenshot shows the AWS IAM console 'Specify permissions' page. The 'Policy editor' is active, displaying a JSON policy with the following content:

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "Statements",
6       "Effect": "Allow",
7       "Action": "s3:GetObject",
8       "Resource": "*"
9     }
10  ]
11 }
```

The 'Edit statement' panel on the right shows a 'Select a statement' dialog with the text: 'Select an existing statement in the policy or add a new statement.' and a '+ Add new statement' button.

3. Click Next, give the policy a proper name, and then create it.

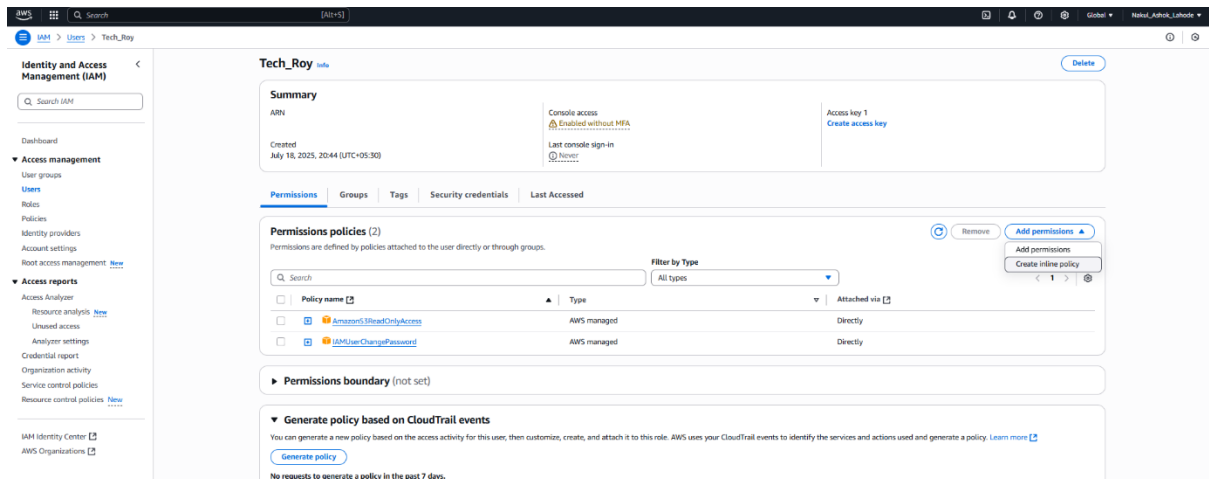
The screenshot shows the AWS IAM console 'Specify permissions' page. The 'Policy editor' is active, displaying a JSON policy with the following content:

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "Statements",
6       "Effect": "Allow",
7       "Action": "s3:PutObject",
8       "Resource": "*"
9     }
10  ]
11 }
```

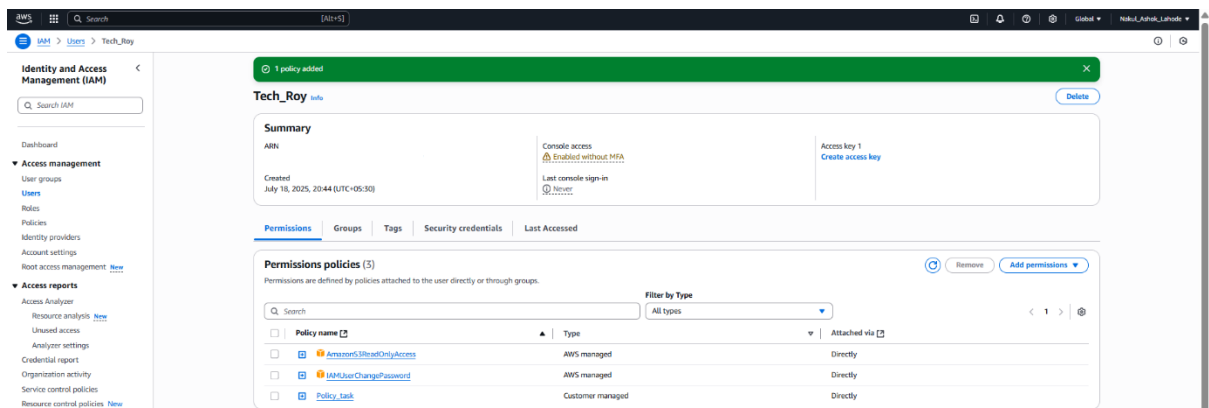
The 'Edit statement' panel on the right shows a 'Select a statement' dialog with the text: 'Select an existing statement in the policy or add a new statement.' and a '+ Add new statement' button.

Step 4: Attach the Policy to the IAM User

1. Go to **IAM > Users**, select the newly created user.
2. Click on **"Add Permissions"**.

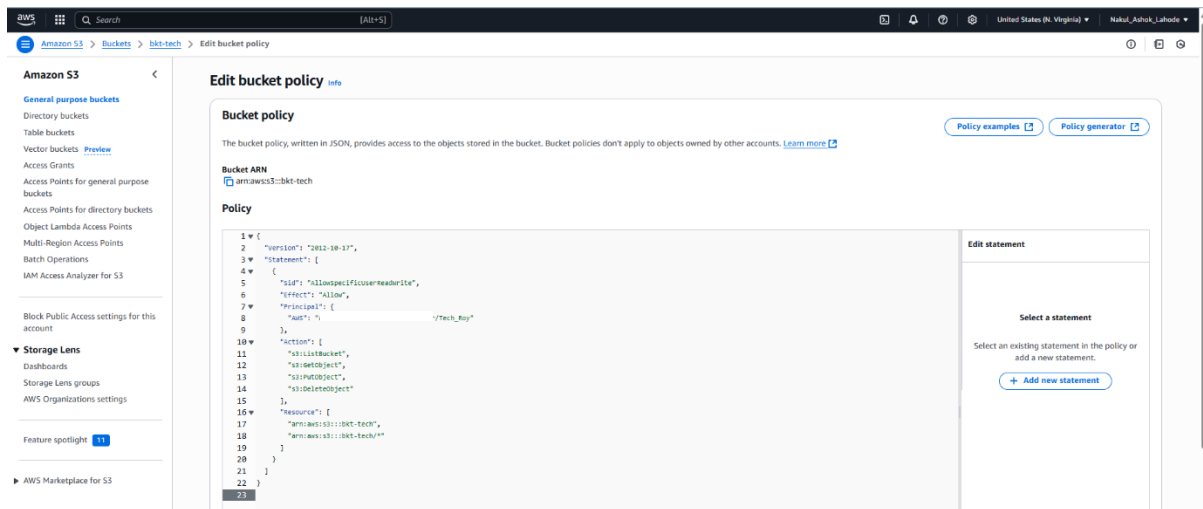


3. Choose **"Attach existing policies directly"**.
4. Select the **custom policy** you created earlier.
5. Review and click **Add Permissions**.



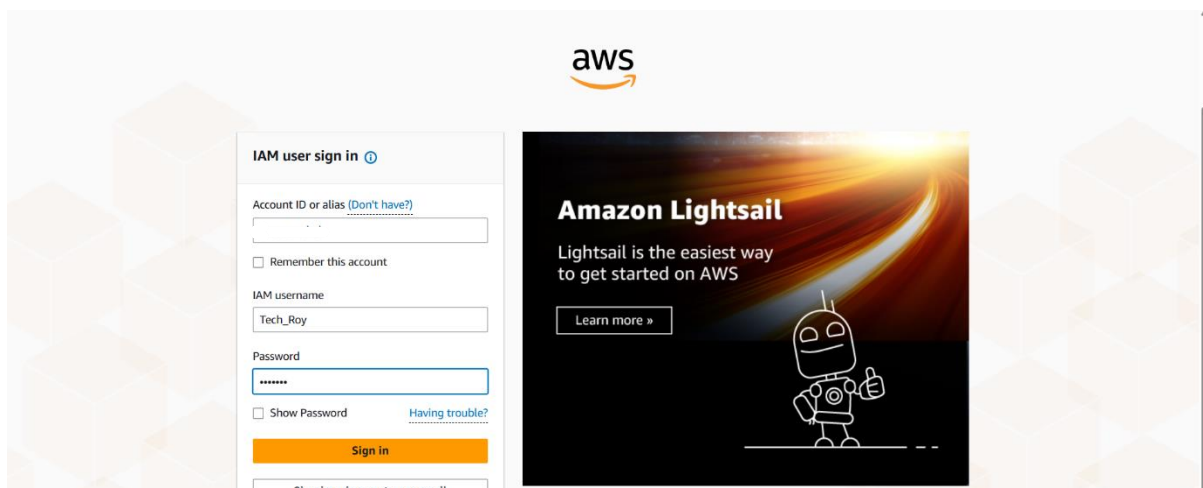
Step 5: Create & Apply Bucket Policy

1. Go to the **S3 bucket**.
2. Navigate to the **Permissions** tab.
3. Click **Bucket Policy** and paste a policy that allows only the specific IAM user to access the bucket.

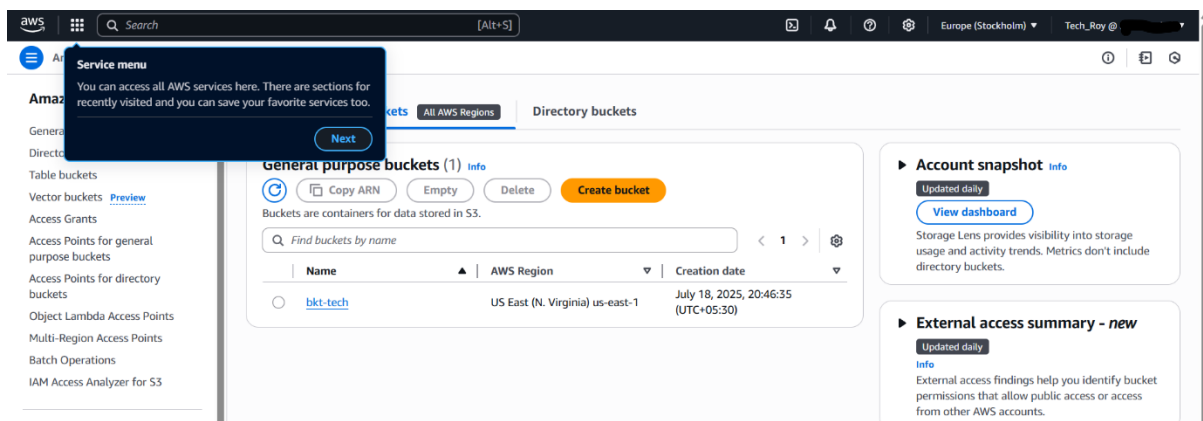


Step 6: Validate the Access Control

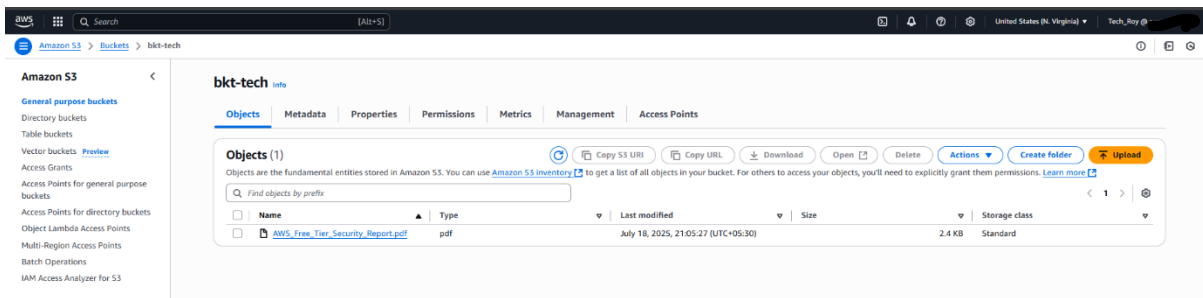
1. **Sign in** to AWS as the newly created IAM user (use the custom login link provided in IAM).



2. Navigate to **S3**, open the specific bucket.

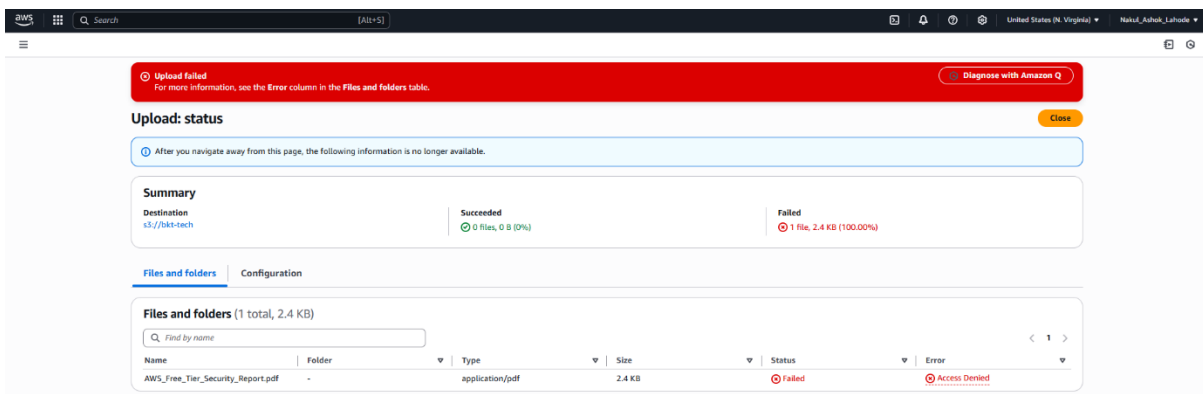


3. Try uploading a file. It should **succeed**.



4. Now switch to a **different user or root user**, and try uploading the same file.

5. This action should result in an **Access Denied** error.



6. This confirms that the policy is working as intended.