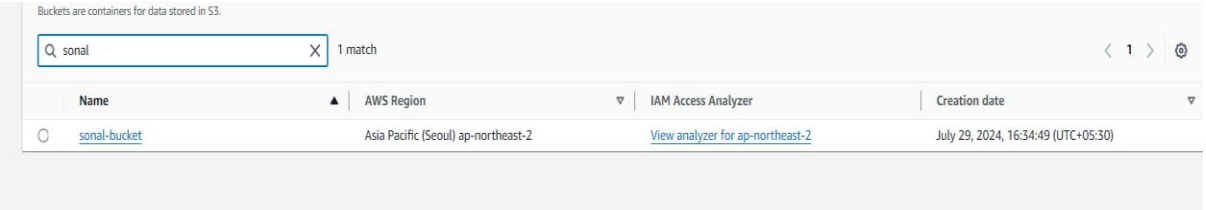
Assignment 2: Automated S3 Bucket Cleanup Using AWS Lambda and Boto3

Objective: To gain experience with AWS Lambda and Boto3 by creating a Lambda function that will automatically clean up old files in an S3 bucket.

Step 1. S3 Setup

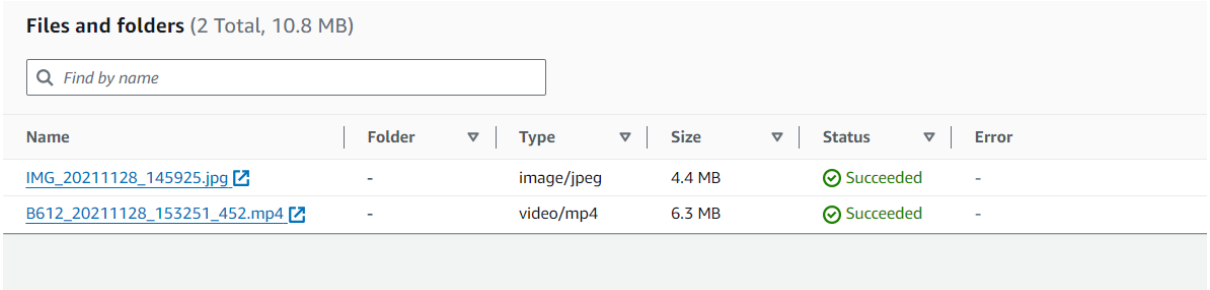
1. **Create a New S3 Bucket:**

- Go to the s3 Dashboard
- Click "Create bucket".



2. **Upload Files:**

- Navigate to your bucket.
- Click "Upload" and add multiple files to the bucket. Ensure some files have a modified date older than 30 days.

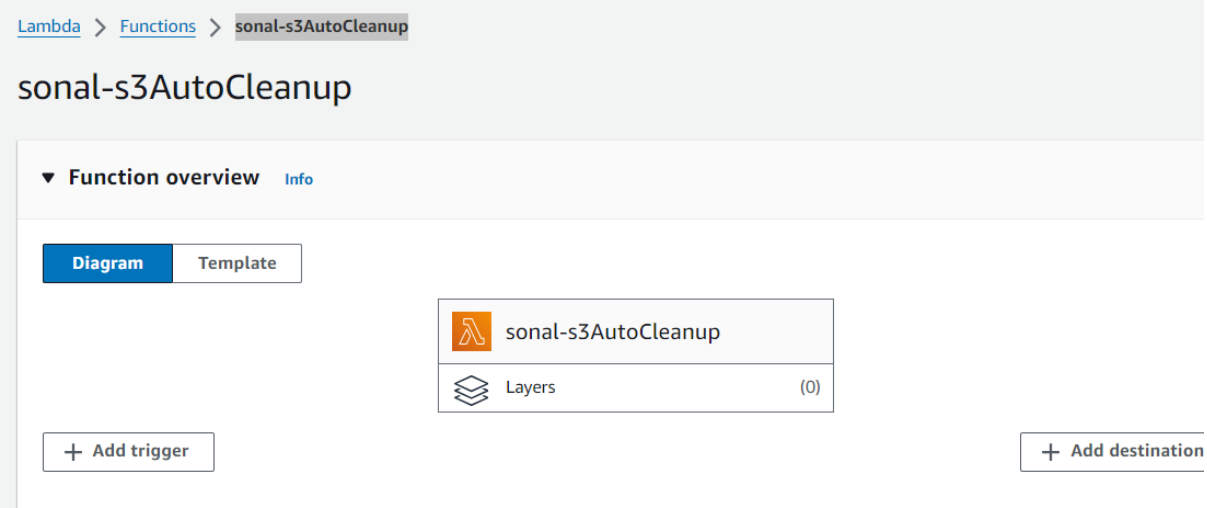


Step 2. Create IAM Role for Lambda

- Name the role '**sonal-LambdaS3CleanupRole**'
- Attach the AmazonS3FullAccess policy.
- Create the role

Step 3. Create the Lambda Function

- Give your function a name '**sonal-s3AutoCleanup'**
- Choose Python 3.11 as the runtime.
- Under "Permissions", choose "Use an existing role".
- Select the role you created earlier **'sonal-LambdaS3CleanupRole**'.
- Click "Create function".



Step 4. Write the Boto3 Python script to:

- Initialize a boto3 S3 client.
- List objects in the specified bucket.
- Delete objects older than 30 days.
- Print the names of deleted objects for logging purposes.

Python script

s3.py

```python
import boto3
from datetime import datetime, timezone, timedelta

def lambda_handler():
    s3 = boto3.client('s3')
    bucket_name = 'sonal-bucket'
    days_old = 30  # Temporary adjustment for testing
    delete_time = datetime.now(timezone.utc) - timedelta(days=days_old)

    # List objects in the specified bucket
    response = s3.list_objects_v2(Bucket=bucket_name)

    print(response)

    if 'Contents' not in response:
        print("Bucket is empty.")
        return {
            'statusCode': 200,
            'body': "Bucket is empty."
        }

    delete_keys = []
    for obj in response['Contents']:
        print(f"Object: {obj['Key']}, LastModified: {obj['LastModified']}")
        if obj['LastModified'] < delete_time:
            delete_keys.append({'Key': obj['Key']})
            print(f"Object {obj['Key']} is older than 30 days and will be deleted")

    if delete_keys:
        print(f"Deleting objects: {delete_keys}")
        s3.delete_objects(Bucket=bucket_name, Delete={'Objects': delete_keys})
        deleted_objects = [key['Key'] for key in delete_keys]
        print(f"Deleted objects: {deleted_objects}")
    else:
        print("No objects older than 30 days found.")

    return {
        'statusCode': 200,
        'body': f"Deleted objects: {delete_keys}"
    }
```

Step 4. Manual Invocation:

- After saving your function, manually trigger it.
- Go to the S3 dashboard and confirm that only files newer than 30 days remain.