

## Assignment 3: Monitor Unencrypted S3 Buckets Using AWS Lambda and Boto3

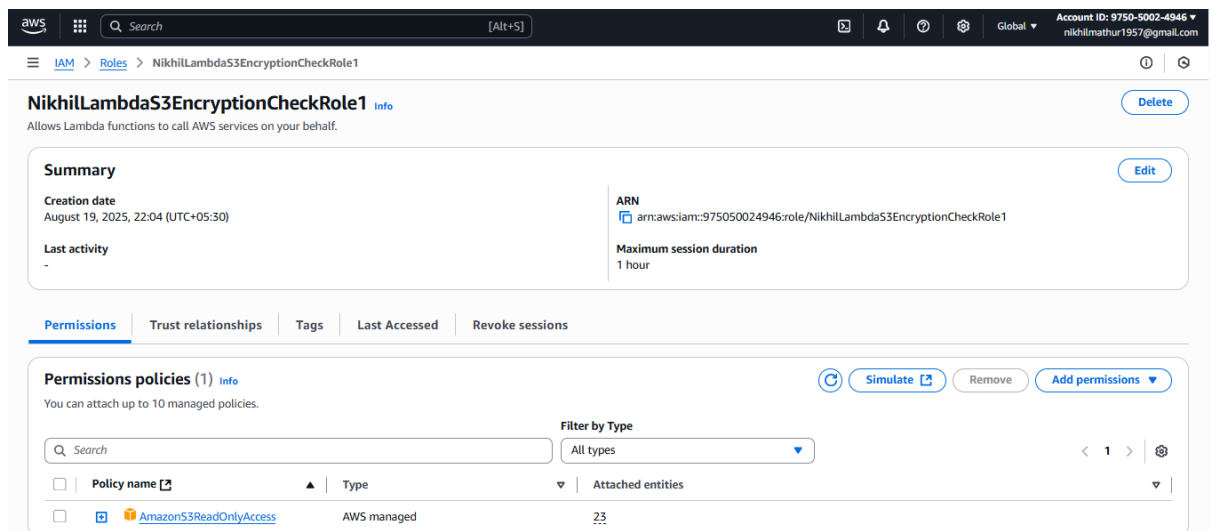
**Objective:** To enhance your AWS security posture by setting up a Lambda function that detects any S3 bucket without server-side encryption.

**Task:** Automate the detection of S3 buckets that don't have server-side encryption enabled.

### Lambda Function Creation:

#### 1. Create an IAM Role for Lambda

- AWS Console => IAM => Roles => Create role.
- Trusted Entity type: AWS Services
- Use Case: Lambda
- Click Next



**Permissions policies:** AmazonEC2FullAccess

**Role name:** NikhilLambdaEC2ControlRole1

**Click Create role.**

#### 2. Create the Lambda Function

Go to AWS Console => Lambda.

**Click Create function.**

Select Author from scratch

Function name: **NikhilS3EncryptionCheck**

Runtime: Python 3.13

Permissions:

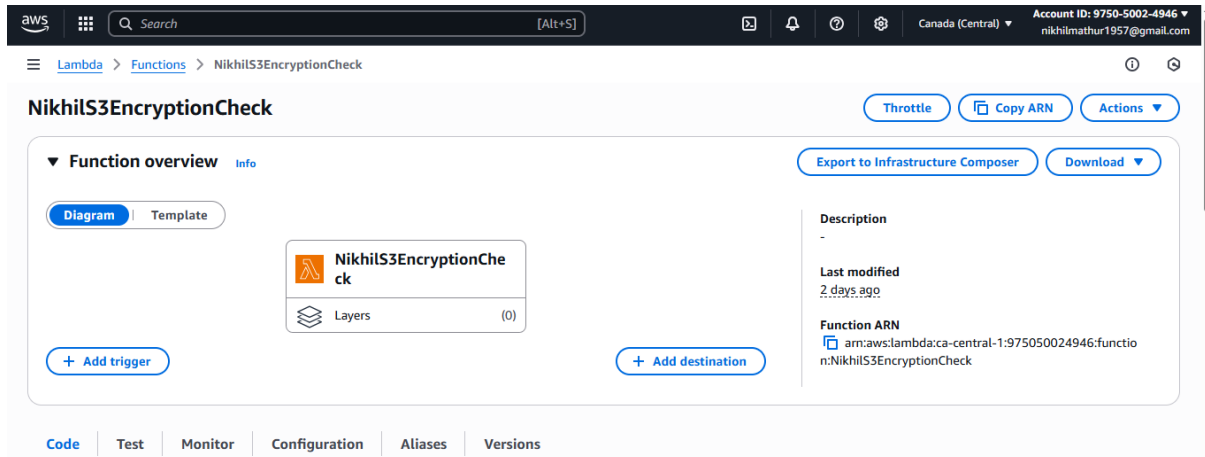
Expand Change default execution role.

Select Use an existing role.

Choose **NikhilLambdaEC2ControlRole1** from the dropdown.

**Note** => I choose the role **prashantb12-role-9p53470y** for permission access to run the code.

**Click Create function.**



### 3. Add Python Code to Control EC2

```
Assignment3.py X
Assignment-3 > Assignment3.py
1 import boto3
2 from botocore.exceptions import ClientError
3
4 def lambda_handler(event, context):
5     s3 = boto3.client("s3")
6
7     # Get all S3 buckets
8     response = s3.list_buckets()
9     buckets = response["Buckets"]
10
11     non_encrypted_buckets = []
12
13     for bucket in buckets:
14         bucket_name = bucket["Name"]
15         try:
16             # Check encryption settings
17             enc = s3.get_bucket_encryption(Bucket=bucket_name)
18             rules = enc["ServerSideEncryptionConfiguration"]["Rules"]
19
20             # If rules exist, SSE is enabled
21             print(f" {bucket_name} has encryption: {rules}")
22
23         except ClientError as e:
24             error_code = e.response["Error"]["Code"]
25             if error_code == "ServerSideEncryptionConfigurationNotFoundError":
26                 print(f" {bucket_name} does NOT have encryption enabled")
27                 non_encrypted_buckets.append(bucket_name)
28             else:
29                 print(f" Could not check bucket {bucket_name}: {e}")
30
```

```

30
31     if non_encrypted_buckets:
32         print("Buckets without encryption:", non_encrypted_buckets)
33     else:
34         print("All buckets have encryption enabled ✅")
35
36     return {"NonEncryptedBuckets": non_encrypted_buckets}
37

```

## Create the test case

[Lambda](#) > [Functions](#) > NikhilEC2TagBasedControl
 ⓘ ⓘ

Code **Test** Monitor Configuration Aliases Versions

✔ Executing function: succeeded ([logs](#))
 
[Delete](#) [CloudWatch Logs Live Tail](#) [Save](#) [Test](#)

[Details](#)

**Test event** info [Delete](#) [CloudWatch Logs Live Tail](#) [Save](#) [Test](#)

To invoke your function without saving an event, modify the event, then choose Test. Lambda uses the modified event to invoke your function, but does not overwrite the original event until you choose Save.

**Test event action**

☐ Create new event
 ☒ Edit saved event

**Event name**

NikhilMathurTestEvent 🔄

**Event JSON** [Format JSON](#)

```
1 {}
```

Click for deploy the code

Test the code

[aws](#)
🔍 Search [Alt+S]

📧 🔔 ⓘ ⚙️
Canada (Central) ▼

**Account ID: 9750-5002-4946** ▼  
 nikhilmathur1957@gmail.com

[Lambda](#) > [Functions](#) > NikhilS3EncryptionCheck
 ⓘ ⓘ

✓ NIKHILS3ENCRYPTIONCHECK  
 ✚ lambda\_function.py

✚ DEPLOY  
 Deploy (Ctrl+Shift+U)  
 Test (Ctrl+Shift+I)

✓ TEST EVENTS [SELECTED: NIKHILMATHURTESTEVE...  
 + Create new test event  
 Private saved events  
 NikhilMathurTestEvent

📄 lambda\_function.py
 

```

1 import boto3
2 from botocore.exceptions import ClientError
3
                    
```

PROBLEMS

STATUS: Succeeded

Test Event Name: NikhilMathurTestEvent

OUTPUT
CODE REFERENCE LOG
TERMINAL

Response:

```
{
  "NonEncryptedBuckets": []
}
```

Function Logs:

```

ketKeyEnabled': True}]
somu-static-website has encryption: [{'ApplyServerSideEncryptionByDefault': {'SSEAlgorithm': 'AES256'},
'BucketKeyEnabled': True}]
somumh has encryption: [{'ApplyServerSideEncryptionByDefault': {'SSEAlgorithm': 'AES256'}, 'BucketKeyEnabled': True}]
studentai-bucket has encryption: [{'ApplyServerSideEncryptionByDefault': {'SSEAlgorithm': 'AES256'},
'BucketKeyEnabled': True}]
subhadeep-bucket has encryption: [{'ApplyServerSideEncryptionByDefault': {'SSEAlgorithm': 'AES256'},
'BucketKeyEnabled': True}]
                    
```

Ln 37, Col 1 Spaces: 4 UTF-8 LF Python Lambda Layout: US