

## Use case

### IAM Policy Audit Tool

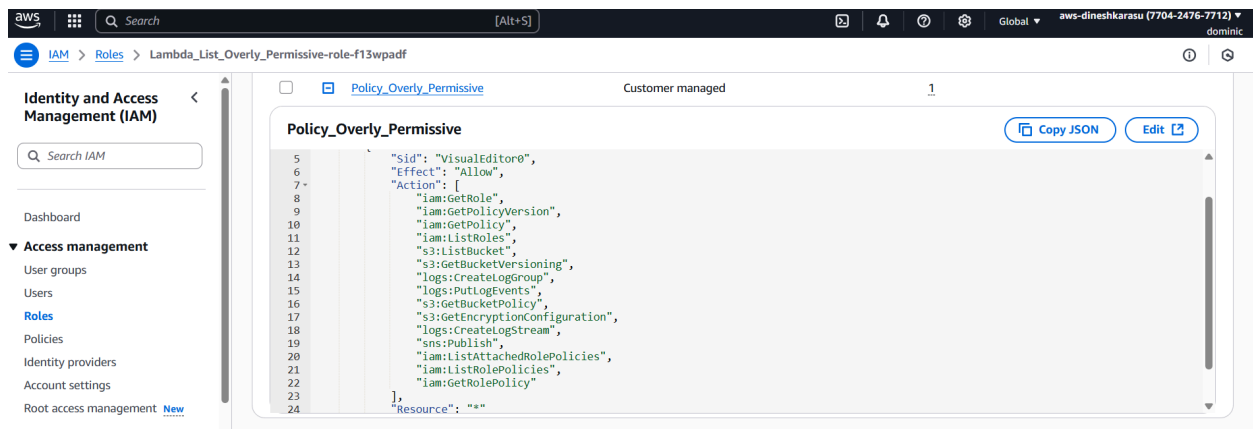
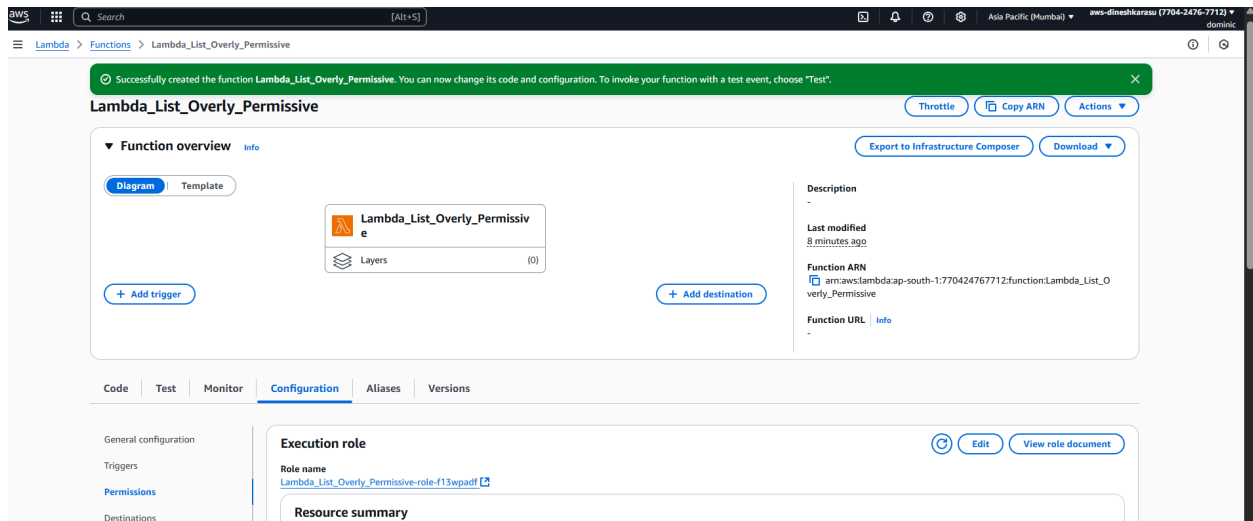
## Use case Description

List IAM roles/policies and detect overly permissive ones(e.g., Action: \*)

Check for public access, encryption, versioning.

## Approach :

1. Create a Lambda Function
2. Set IAM permissions to the Lambda to access Overly permissive resources
3. Create an SNS Topic and subscribe email to get the output over email
4. Write python code in Lambda and Integrate SNS Notification
5. Test the functionality



```
EXPLORER
└─ LAMBDA_LIST_OVERLY_PERMISSIVE
  └─ lambda_function.py

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

TEST EVENTS [NONE SELECTED]
+ Create new test event

ENVIRONMENT VARIABLES
0 Amazon Q

lambda_function.py
1 import boto3
2 import json
3
4 iam = boto3.client('iam')
5 s3 = boto3.client('s3')
6 sns = boto3.client('sns')
7
8 SNS_TOPIC_ARN = 'arn:aws:sns:ap-south-1:770424767712:SMS_SendOverlyPermissiveData'
9 print("Lambda Execution started ")
10
11 audit_report = []
12
13 def is_overly_permissive(policy_document):
14     for statement in policy_document.get('Statement', []):
15         if (statement.get('Effect') == 'Allow' and
16             (statement.get('Action') == '*' or statement.get('Resource') == '*')):
17             return True
18     return False
19
20 def check_iam_policies():
21     roles = iam.list_roles()['Roles']
22     print("Checking IAM policies")
23     for role in roles:
24         audit_report.append(f"Role: {role['RoleName']}")
25
26         # Attached Policies
27         attached = iam.list_attached_role_policies(RoleName=role['RoleName'])['AttachedPolicies']
28         print("Attached Policies:", attached)
29         for policy in attached:
30             default_version_id = iam.get_policy(PolicyArn=policy['PolicyArn'])['Policy']['DefaultVersionId']
31             version = iam.get_policy_version(PolicyArn=policy['PolicyArn'], VersionId=default_version_id)
32             if is_overly_permissive(version['PolicyVersion']['Document']):
33                 audit_report.append(f"Overly permissive attached policy: {policy['PolicyName']}")
34
35         # Inline Policies
36         inline_names = iam.list_role_policies(RoleName=role['RoleName'])['PolicyNames']
37         print("Inline Policies:", inline_names)
38         for inline in inline_names:
39             policy_doc = iam.get_role_policy(RoleName=role['RoleName'], PolicyName=inline)['PolicyDocument']
40             if is_overly_permissive(policy_doc):
```

```
20 def check_iam_policies():
21     for role in roles:
22
23         # Inline Policies
24         inline_names = iam.list_role_policies(RoleName=role['RoleName'])['PolicyNames']
25         print("Inline Policies:", inline_names)
26         for inline in inline_names:
27             policy_doc = iam.get_role_policy(RoleName=role['RoleName'], PolicyName=inline)['PolicyDocument']
28             if is_overly_permissive(policy_doc):
29                 audit_report.append(f"Overly permissive inline policy: {inline}")
30
31 def check_s3_buckets():
32     print("Checking S3 buckets")
33     buckets = s3.list_buckets()['Buckets']
34     for bucket in buckets:
35         name = bucket['Name']
36         audit_report.append(f"S3 Bucket: {name}")
37
38         # Public access check
39         print("Checking S3 Public access")
40         try:
41             policy = s3.get_bucket_policy(Bucket=name)
42             if "Allow" in policy['Policy']:
43                 audit_report.append(f"Bucket {name} might be public (Allow found)")
44         except s3.exceptions.NoSuchBucketPolicy:
45             audit_report.append(f"No bucket policy found (probably private)")
46         except Exception as e:
47             audit_report.append(f"Error getting bucket policy: {str(e)}")
48
49         # Encryption check
50         print("Checking S3 Encryption")
51         try:
52             s3.get_bucket_encryption(Bucket=name)
53             audit_report.append(f"Encryption enabled")
54         except s3.exceptions.ClientError:
55             audit_report.append(f"Encryption NOT enabled")
56
57         # Versioning check
58         print("Checking S3 Versioning")
59         versioning = s3.get_bucket_versioning(Bucket=name)
```

```
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77 def send_sns_notification():
78     print("Sending SNS notification")
79     message = "\n".join(audit_report)
80     sns.publish(
81         TopicArn=SNS_TOPIC_ARN,
82         Subject=f"IAM & S3 Audit Report",
83         Message=message
84     )
85     print("SNS notification sent")
86
87 def lambda_handler(event, context):
88     audit_report.clear()
89     audit_report.append("=== IAM Role & Policy Audit ===")
90     check_iam_policies()
91     audit_report.append("\n=== S3 Bucket Audit ===")
92     check_s3_buckets()
93     send_sns_notification()
94     return {
95         'statusCode': 200,
96         'body': json.dumps("Audit complete and report sent via SNS.")
97     }
98
```

Amazon SNS > Topics > SNS\_SendOverlyPermissiveData

**New Feature**  
Amazon SNS now supports High Throughput FIFO topics. [Learn more](#)

**SNS\_SendOverlyPermissiveData** [Edit] [Delete] [Publish message]

**Details**

<b>Name</b> SNS_SendOverlyPermissiveData	<b>Display name</b> -
<b>ARN</b> arn:aws:sns:south-1:770424767712:SNS_SendOverlyPermissiveData	<b>Topic owner</b> 770424767712
<b>Type</b> Standard	

**Subscriptions** | Access policy | Data protection policy | Delivery policy (HTTP/S) | Delivery status logging | Encryption | Tags | Integrations

**Subscriptions (1)** [Edit] [Delete] [Request confirmation] [Confirm subscription] [Create subscription]

ID	Endpoint	Status	Protocol
062d2dbe-2021-4ec8-ab29-5e1bf78b6e03	dikarasu@evoketechnologies.com	Confirmed	EMAIL

Increase the execution time for Lambda as this process takes more time and by default Lambda gets created with 3 execution time of seconds

Lambda > Functions > Lambda\_List\_Overly\_Permisive > Edit basic settings

**Edit basic settings**

**Basic settings** [Info]

Description - optional

**Memory** [Info]  
Your function is allocated CPU proportional to the memory configured.  
128 MB  
Set memory to between 128 MB and 10240 MB

**Ephemeral storage** [Info]  
You can configure up to 10 GB of ephemeral storage (/tmp) for your function. [View pricing](#)  
512 MB  
Set ephemeral storage (/tmp) to between 512 MB and 10240 MB.

**SnapStart** [Info]  
Reduce startup time by having Lambda cache a snapshot of your function after the function has initialized. To evaluate whether your function code is resilient to snapshot operations, review the [SnapStart compatibility considerations](#). For Python and .NET runtimes, view [pricing](#).  
None

**Timeout**  
2 min 30 sec

**Execution role**  
Choose a role that defines the permissions of your function. To create a custom role, go to the [IAM console](#).  
☒ Use an existing role  
☐ Create a new role from AWS policy templates

**Successfully updated the function Lambda\_List\_Overly\_Permisive.**

**Executing function: succeeded** [logs]

**Details**

```
{
  "statusCode": 200,
  "body": "\nAudit complete and report sent via SNS.\n"
}
```

**Summary**

<b>Code SHA-256</b> HAPq9EReJVECSgLavtc/gyd5v2td9eiUGF932tQJbXy=	<b>Execution time</b> 2 minutes ago
<b>Request ID</b> efcdeda7-0206-4696-aad0-4a43f11e0d04	<b>Duration</b> 42645.14 ms
<b>Billed duration</b> 42646 ms	<b>Resources configured</b> 128 MB
<b>Max memory used</b> 96 MB	

**Log output**

The area below shows the last 4 KB of the execution log. [Click here](#) to view the corresponding CloudWatch log group.

```

AWSLambdaBasicExecutionRole-3493df2-a2fa-4868-8f58-5c6abc24c162', 'PolicyArn': 'arn:aws:iam::770424767712:policy/service-role/AWSLambdaBasicExecutionRole-3d93df2-a2fa-4868-8f58-5c6abc24c162'})
Inline Policies: []
Attached Policies: [{"PolicyName": 'AWSLambdaBasicExecutionRole-d035bdc-9009-46bf-afce-a72c9238c0ff', 'PolicyArn': 'arn:aws:iam::770424767712:policy/service-role/AWSLambdaBasicExecutionRole-d035bdc-9009-46bf-afce-a72c9238c0ff'}, {'PolicyName': 'Policy_Overly_Permisive', 'PolicyArn': 'arn:aws:iam::770424767712:policy/Policy_Overly_Permisive'}]
Inline Policies: []
Attached Policies: [{"PolicyName": 'AWSLambdaBasicExecutionRole-345a8f63-4890-429f-82d3-64cae5f9333', 'PolicyArn': 'arn:aws:iam::770424767712:policy/service-role/AWSLambdaBasicExecutionRole-345a8f63-4890-429f-82d3-
```

## IAM & S3 Audit Report



AWS Notifications <no-reply@sns.amazonaws.com>




To:  Dinesh Karasu

  Reply  Reply all  Forward    ...



Mon 7/7/2025 2:18 PM

CAUTION: This email originated from outside of the organization. Do not click links or open attachments unless you recognize the sender and know the content is safe.





=== IAM Role & Policy Audit ===

- ◆ Role: Amazon\_EventBridge\_Invoke\_Lambda\_577396163
- ◆ Role: Amazon\_EventBridge\_Invoke\_Lambda\_8437325
- ◆ Role: Amazon\_EventBridge\_Invoke\_Sns\_820818589
- ◆ Role: AWSCodePipelineServiceRole-ap-south-1-CICD\_Deploy\_Lambda
- ◆ Role: AWSCodePipelineServiceRole-ap-south-1-PythonApp-CI-CD  
 Overly permissive attached policy: AWSCodePipeline\_FullAccess
- ◆ Role: AWSServiceRoleForAutoScaling  
 Overly permissive attached policy: AutoScalingServiceRolePolicy
- ◆ Role: AWSServiceRoleForRDS  
 Overly permissive attached policy: AmazonRDSServiceRolePolicy
- ◆ Role: AWSServiceRoleForSupport
- ◆ Role: AWSServiceRoleForTrustedAdvisor

## IAM & S3 Audit Report

- ◆ Role: S3-Full-Access  
 Overly permissive attached policy: S3-Full-Access
- ◆ Role: Send\_HealthCheckDetails\_to\_SQS-role-s86f0jhe
- ◆ Role: Send\_Message\_To\_SQS-role-ecgji0oj
- ◆ Role: TriggerLimitExceededSNS-role-ujt432bs  
 Overly permissive attached policy: AdministratorAccess

=== S3 Bucket Audit ===

- ◆ S3 Bucket: cf-templates-1njo811r3cjr-ap-south-1
  - ✓ No bucket policy found (probably private)
  - ✓ Encryption enabled
  -  Versioning NOT enabled
- ◆ S3 Bucket: codepipeline-ap-south-1-87272186fa9a-4a78-b3ac-3657aed161dd
  - ✓ Encryption enabled
  -  Versioning NOT enabled
- ◆ S3 Bucket: s3-bkt-img-rekognition
  - ✓ No bucket policy found (probably private)
  - ✓ Encryption enabled
  -  Versioning NOT enabled
- ◆ S3 Bucket: s3-my-codepipeline-bucket
  - ✓ No bucket policy found (probably private)
  - ✓ Encryption enabled
  -  Versioning NOT enabled