

## [AWS AI S3 Security Scanner](#)

### Project Overview

I built an AI-powered S3 Security Scanner. The system uses Python and the AWS SDK ([boto3](#)) to identify unencrypted buckets, leverages Gemini AI to explain risks in plain English, and uses Amazon EventBridge to automate the audit every 12 hours.

### Phase 1: Brain & Logic (Setup)

- **Generated Gemini API Key:** I started by visiting Google AI Studio to generate a secure API key. This serves as the "analytical brain" that allows my scanner to interpret technical data.
- **Developed Scanner Logic:** Using the Cursor IDE, I implemented a Python script ([s3\\_scanner.py](#)) that uses [boto3](#) to connect to my AWS account. I programmed it to loop through all S3 buckets and check for active server-side encryption.

```
s3_scanner.py 2 •
s3_scanner.py > ...
1  import boto3
2  import json
3  import os
4  import google.generativeai as genai
5
6  def lambda_handler(event, context):
7      """Scan S3 buckets for encryption and use AI to explain risks"""
8
9      # Initialize S3 client
10     s3_client = boto3.client('s3')
11
12     print("Scanning S3 buckets for encryption...")
13
14     # Get all S3 buckets
15     response = s3_client.list_buckets()
16     buckets = response['Buckets']
17
18     print(f"Found {len(buckets)} buckets to scan\n")
19
20     # Store results
21     scan_results = []
22
23     for bucket in buckets:
24         bucket_name = bucket['Name']
```

- Integrated Gemini AI:** I drafted a specific prompt within the code that sends raw S3 configuration data to Gemini. I instructed the AI to return high-level, plain-English security insights rather than just technical error codes.

```

26     # Check if bucket has encryption enabled
27     try:
28         encryption = s3_client.get_bucket_encryption(Bucket=bucket_name)
29         encrypted = True
30         encryption_type = encryption['ServerSideEncryptionConfiguration']['Rules'][0]['ApplyServerSideEncryptionByDefault']['SSEAlgorithm']
31     except s3_client.exceptions.ServerSideEncryptionConfigurationNotFoundError:
32         encrypted = False
33         encryption_type = 'None'
34
35     # Store result
36     scan_results.append({
37         'bucket_name': bucket_name,
38         'encrypted': encrypted,
39         'encryption_type': encryption_type
40     })
41
42     status = "Encrypted" if encrypted else "Not Encrypted"
43     print(f"{status}: {bucket_name} ({encryption_type})")
44
45     # Count unencrypted buckets
46     unencrypted_count = len([r for r in scan_results if not r['encrypted']])
47     unencrypted_buckets = [r['bucket_name'] for r in scan_results if not r['encrypted']]
48
49     # Use AI to analyze security findings
50     print("\nAnalyzing security findings with Gemini AI...")
51
52     # Configure Gemini
53     api_key = os.environ.get('GOOGLE_API_KEY')
54     if not api_key:
55         ai_analysis = "AI analysis skipped: GOOGLE_API_KEY not configured"
56     else:
57         genai.configure(api_key=api_key)
58         model = genai.GenerativeModel("gemini-2.0-flash-exp")
59
60         prompt = f"""You are an AWS security expert. Analyze this S3 encryption scan and provide a brief security assessment.
61
62 Scan Results:
63
64 - Total Buckets: {len(buckets)}
65 - Encrypted: {len(buckets) - unencrypted_count}
66 - Unencrypted: {unencrypted_count}
67 - Unencrypted Bucket Names: {'', '.join(unencrypted_buckets) if unencrypted_buckets else 'None'}
68
69 Provide a 2-3 sentence analysis:
70 1. What's the security risk of unencrypted buckets?
71 2. What encryption should be enabled? (AES256 or aws:kms)
72 3. What action should the user take immediately?
73
74 Be concise and actionable."""
75
76     try:
77         response = model.generate_content(prompt)
78         ai_analysis = response.text
79     except Exception as e:
80         ai_analysis = f"AI analysis failed: {str(e)}"
81
82     # Build final result
83     result = {
84         'total_buckets': len(buckets),
85         'unencrypted_buckets': unencrypted_count,
86         'encrypted_buckets': len(buckets) - unencrypted_count,
87         'scan_results': scan_results,
88         'ai_analysis': ai_analysis,
89         'alert': unencrypted_count > 0
90     }
91
92     print(f"\nScan complete: {unencrypted_count}/{len(buckets)} buckets need encryption")
93
94     return {
95         'statusCode': 200,
96         'body': json.dumps(result)
97     }
98

```

Phase 2: Security & Permissions (IAM)

- **Created Read-Only Policy:** I logged into the IAM Console and authored a custom JSON policy. I granted the specific permissions `s3:ListAllMyBuckets` and `s3:GetEncryptionConfiguration` to ensure the principle of least privilege.

Policy ReadS3EncryptionPolicy created.

View policy

ReadS3EncryptionPolicy

Policy details

Type

Customer managed

Creation time

January 26, 2026, 18:27 (UTC-08:00)

Edited time

January 26, 2026, 18:27 (UTC-08:00)

ARN

arn:aws:iam::471744311739:policy/ReadS3EncryptionPolicy

Permissions

Entities attached

Tags

Policy versions (1)

Last Accessed

Permissions defined in this policy

Edit

Summary

JSON

Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it

Search

Allow (1 of 461 services)

Show remaining 460 services

Service	Access level	Resource	Request condition
S3	Limited: List, Read	All resources	None

- **Established Execution Role:** I created an IAM Role named `LambdaS3ScannerRole` for the Lambda service. I attached my custom S3 policy along with the `AWSLambdaBasicExecutionRole` so the function could write logs to CloudWatch.

Role S3LambdaScannerRole created.

View role

S3LambdaScannerRole

Summary

Edit

Creation date

January 26, 2026, 18:29 (UTC-08:00)

ARN

arn:aws:iam::471744311739:role/S3LambdaScannerRole

Last activity

-

Maximum session duration

1 hour

Permissions

Trust relationships

Tags

Last Accessed

Revoke sessions

Permissions policies (2)

Simulate

Remove

Add permissions

You can attach up to 10 managed policies.

Search

Filter by Type

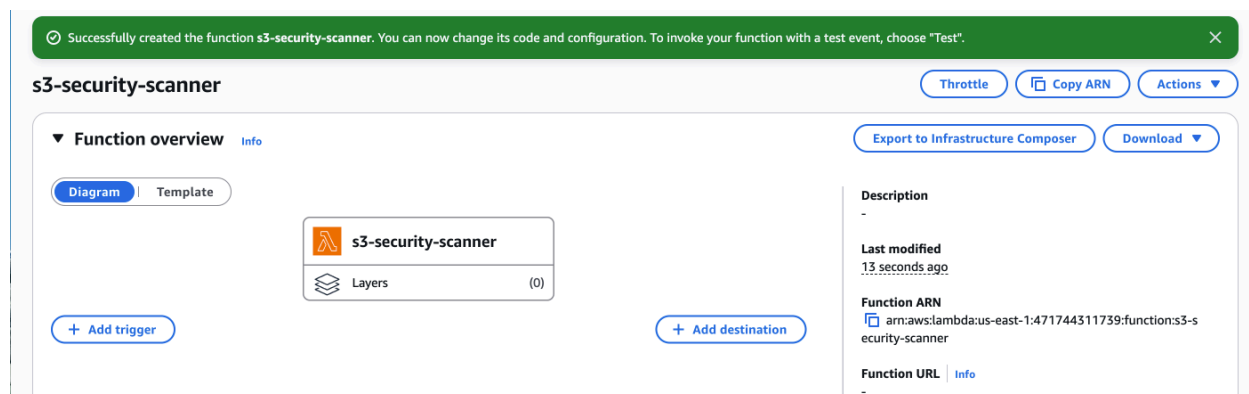
All types

Policy name	Type	Attached entities
AWSLambdaBasicExecutionRole	AWS managed	1
ReadS3EncryptionPolicy	Customer managed	1

- **Configured Trust Relationship:** I verified that the trust policy correctly allows the `lambda.amazonaws.com` service to assume this role during execution.

### Phase 3: Deployment (AWS Lambda)

- **Built Deployment Package:** I created a `requirements.txt` file and used `pip` to install dependencies into a local directory. To keep the package under the 50MB limit, I manually removed the pre-installed `boto3` and `botocore` libraries before zipping the files.
- **Initialized Lambda Function:** I created a new function in the AWS console using the **Python 3.12** runtime. I assigned the `LambdaS3ScannerRole` I built earlier to provide the necessary "identity" for the scan.



- **Configured Runtime & Secrets:** I updated the **Handler** setting to `s3_scanner.lambda_handler` to match my file name. I then added my Gemini API key as an **Environment Variable** (`GOOGLE_API_KEY`) to keep the credential out of the source code.
- **Adjusted Resources:** I increased the function **timeout to 30 seconds** to ensure the scan wouldn't cut off if I added more buckets in the future. As a result the scanned logs can be seen below in the screenshot.

CodeTestMonitorConfigurationAliasesVersions

Executing function: succeeded (logs [L2](#))

Details

```
{
  "statusCode": 200,
  "body": "{\"total_buckets\": 0, \"unencrypted_buckets\": 0, \"encrypted_buckets\": 0, \"scan_results\": [], \"ai_analysis\": \"AI analysis failed: 429 You exceeded your current quota, please check your plan and billing details. For more information on this error, head to: https://ai.google.dev/gemini-api/docs/rate-limits. To monitor your current usage, head to: https://ai.dev/rate-limit. \\n* Quota exceeded for metric: generativelanguage.googleapis.com/generate_content_free_tier_input_token_count, limit: 0, model: gemini-2.0-flash-exp\\n* Quota exceeded for metric: generativelanguage.googleapis.com/generate_content_free_tier_requests, limit: 0, model: gemini-2.0-flash-exp\\nPlease retry in 59.936989165s. [links description: \\nLearn more about Gemini API quotas\\n\\n url: \\nhttps://ai.google.dev/gemini-api/docs/rate-limits\\n\\n violations {\\n quota_metric: \\n generativelanguage.googleapis.com/generate_content_free_tier_input_token_count\\n quota_id: \\nGenerateContentInputTokensPerMinute-FreeTier\\n quota_dimensions {\\n key: \\nmodel\\n value: \\ngemini-2.0-flash-exp\\n }\\n quota_dimensions {\\n key: \\nlocation\\n value: \\nglobal\\n }\\n}\\nviolations {\\n quota_metric: \\n generativelanguage.googleapis.com/generate_content_free_tier_requests, limit: 0, model: gemini-2.0-flash-exp\\n}\\n}\\n\"}"
```

Summary

Code SHA-256

HAPq9EReJVEc5gLavtc/gyd5vZtd9eiUGF932t0jBxY=

Execution time

1 minute ago

Function version

SLATEST

Request ID

1fa7c7a-78b8-412f-b149-0a87c00e0deb

Duration

4828.62 ms

Billed duration

6926 ms

Resources configured

128 MB

Max memory used

119 MB

Init duration

2096.57 ms

Log output

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

```
/var/task/s3_scanner.py:4: FutureWarning:
All support for the 'google.generativeai' package has ended. It will no longer be receiving
updates or bug fixes. Please switch to the 'google.genai' package as soon as possible.
See README for more details:
https://github.com/google-gemini/deprecated-generative-ai-python/blob/main/README.md
import google.generativeai as genai
START RequestId: 1fa7c7a-78b8-412f-b149-0a87c00e0deb Version: SLATEST
Scanning S3 buckets for encryption...
Found 0 buckets to scan
Analyzing security findings with Gemini AI...
```

## Phase 4: Automation & Verification

- **Executed Manual Test:** I triggered the function using a blank test event. I then navigated to **CloudWatch Logs** to confirm the output. I observed that the AI successfully identified my unencrypted buckets and provided actionable remediation steps.

Rule daily-s3-security-scan was updated successfully

daily-s3-security-scan

EditDisableDeleteCloudFormation Template

Rule details

Rule name

daily-s3-security-scan

Description

Daily automated S3 encryption scan

Status

Enabled

Rule ARN

arn:aws:events:us-east-1:471744311739:rule/daily-s3-security-scan

Event bus name

default

Event bus ARN

arn:aws:events:us-east-1:471744311739:event-bus/default

Type

Scheduled Standard

Event schedule

TargetsMonitoringTags

Event schedule

Fixed rate of 24 hour

Edit

- **Scheduled Automation:** I moved to the **Amazon EventBridge** console and created a new "Scheduled Rule." I set a **rate of 12 hours** and selected my Lambda function as the target.
- **Validated Live Results:** To verify the automation, I temporarily changed the schedule to 1 minute. I watched the logs appear in real-time, confirming that my account is now being audited automatically twice a day.

Log events

Actions

Start tailing

Create metric filter

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

Filter events - press enter to search

Clear

1m

30m

1h

12h

Custom

UTC timezone

Display

	Timestamp	Message
		No older events at this moment. <a href="#">Retry</a>
▶	2026-01-27T03:09:37.152Z	INIT_START Runtime Version: python:3.12.v101 Runtime Version ARN: arn:aws:lambda:us-east-1::runtime:994aac32248ecf4d6d9d9f5e9a3a57aba3ccea19d94170a...
▶	2026-01-27T03:09:39.193Z	/var/task/s3_scanner.py:4: FutureWarning:
▶	2026-01-27T03:09:39.193Z	All support for the 'google.generativeai' package has ended. It will no longer be receiving
▶	2026-01-27T03:09:39.193Z	updates or bug fixes. Please switch to the 'google.genai' package as soon as possible.
▶	2026-01-27T03:09:39.193Z	See README for more details:
▶	2026-01-27T03:09:39.193Z	https://github.com/google-gemini/deprecated-generative-ai-python/blob/main/README.md
▶	2026-01-27T03:09:39.193Z	import google.generativeai as genai
▶	2026-01-27T03:09:39.197Z	START RequestId: 4dea182b-50aa-4a21-a52a-f272dc08d0a5 Version: \$LATEST
▶	2026-01-27T03:09:41.808Z	Scanning S3 buckets for encryption...
▶	2026-01-27T03:09:42.148Z	Found 0 buckets to scan
▶	2026-01-27T03:09:42.148Z	Analyzing security findings with Gemini AI...
▶	2026-01-27T03:09:44.089Z	Scan complete: 0/0 buckets need encryption
▶	2026-01-27T03:09:44.130Z	END RequestId: 4dea182b-50aa-4a21-a52a-f272dc08d0a5
▶	2026-01-27T03:09:44.130Z	REPORT RequestId: 4dea182b-50aa-4a21-a52a-f272dc08d0a5 Duration: 4930.79 ms Billed Duration: 6973 ms Memory Size: 128 MB Max Memory Used: 118 MB I...
		No newer events at this moment. <a href="#">Auto retry paused</a> . <a href="#">Resume</a>

**Summary of My Results:** I successfully transitioned from a manual script to a 24/7 automated security auditor. The system now identifies vulnerabilities and translates them into a format that anyone on a security team can understand instantly.