

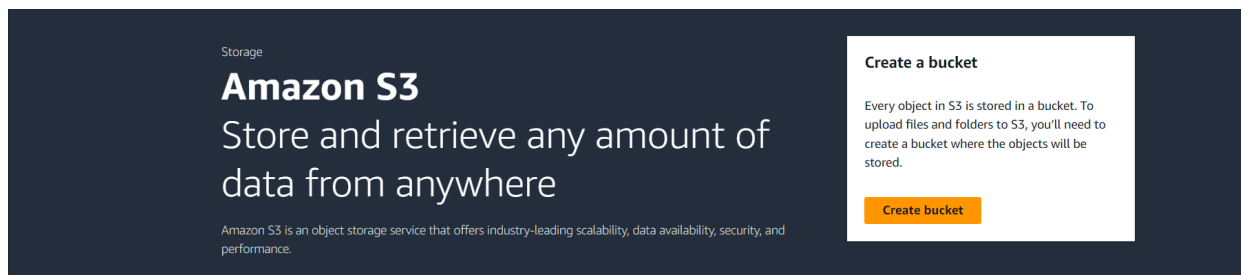
Experiment no-12

Aim: To create a Lambda function which will log "An Image has been added" once you add an object to a specific bucket in S3

Steps to Create the Setup

1. Create an S3 Bucket

- **Navigate to S3:** Go to the AWS Management Console and select the S3 service.
- **Create Bucket:** Click on "Create bucket."
- **Bucket Name and Region:** Enter a unique name for your bucket and select the desired AWS region.
- **Settings:** Configure additional settings (versioning, encryption, etc.) as needed and click "Create bucket."



Amazon S3 > Buckets > Create bucket

Create bucket [Info](#)

Buckets are containers for data stored in S3.

General configuration

AWS Region
US East (N. Virginia) us-east-1

Bucket type [Info](#)

☒ **General purpose**
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

☐ **Directory**
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name [Info](#)

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

Format: s3://bucket/prefix

General purpose buckets (1) Info All AWS Regions				Refresh	Copy ARN	Empty	Delete	Create bucket
Buckets are containers for data stored in S3.								
<input type="text" value="Find buckets by name"/>				Previous 1 Next Settings				
Name	AWS Region	IAM Access Analyzer	Creation date					
<input type="radio"/> tarunbalti	US East (N. Virginia) us-east-1	View analyzer for us-east-1	October 12, 2024, 19:10:10 (UTC+05:30)					

2. Create a Lambda Function

- **Navigate to Lambda:** From the AWS Management Console, go to the Lambda service.
- **Create Function:** Click on "Create function."
- **Author from Scratch:** Choose the "Author from scratch" option.
- **Function Name:** Enter a name for your function (e.g., **S3ImageLogger**).
- **Runtime Selection:** Select a runtime (Python 3.x or Node.js).
- **Create Function:** Click "Create function."

[Lambda](#) > [Functions](#) > Create function

Create function [Info](#)

Choose one of the following options to create your function.

☒ **Author from scratch**
 Start with a simple Hello World example.

☐ **Use a blueprint**
 Build a Lambda application from sample code and configuration presets for common use cases.

☐ **Container image**
 Select a container image to deploy for your function.

Basic information

Function name
Enter a name that describes the purpose of your function.

Function name must be 1 to 64 characters, must be unique to the Region, and can't include spaces. Valid characters are a-z, A-Z, 0-9, hyphens (-), and underscores (_).

Runtime [Info](#)
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.

[Refresh](#)

Architecture [Info](#)
Choose the instruction set architecture you want for your function code.

☒ x86_64
 ☐ arm64

3. Write the Lambda Function Code

- In the Lambda function console, scroll down to the code editor.

Replace the default code with the following code snippet (assuming you're using Python):

python

Copy code

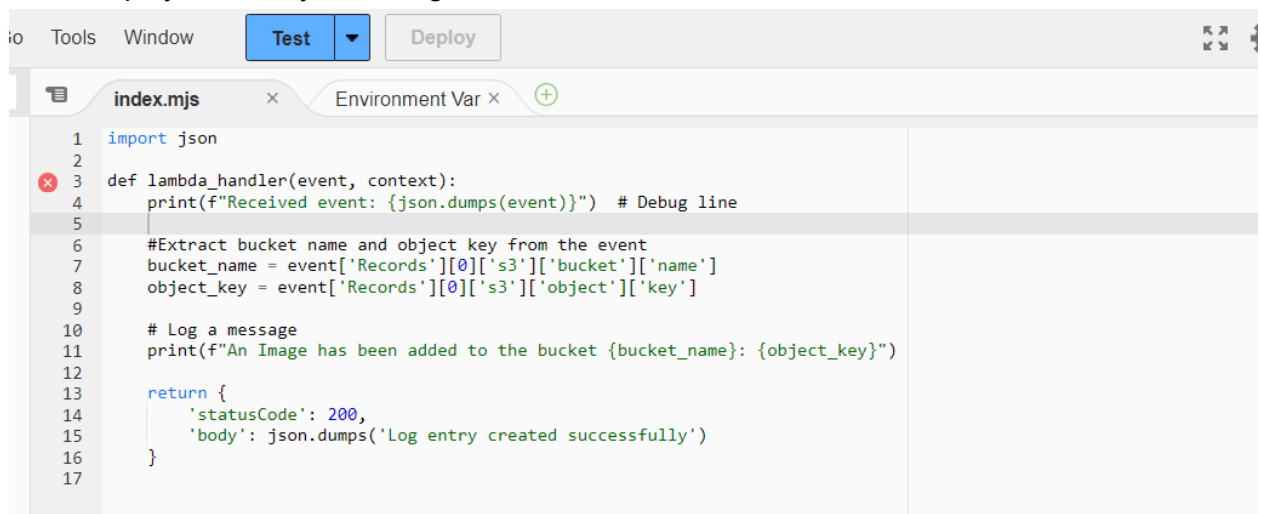
```
import json
```

```
def lambda_handler(event, context):
    # Extract bucket name and object key from the event
    bucket_name = event['Records'][0]['s3']['bucket']['name']
    object_key = event['Records'][0]['s3']['object']['key']

    # Log a message
    print(f"An Image has been added to the bucket {bucket_name}: {object_key}")


    return {
        'statusCode': 200,
        'body': json.dumps('Log entry created successfully')
    }
```

-
- Click "Deploy" to save your changes.



4. Set Up S3 Trigger for the Lambda Function

Trigger configuration [Info](#)

 **S3**
aws asynchronous storage

Bucket
Choose or enter the ARN of an S3 bucket that serves as the event source. The bucket must be in the same region as the function.

× ↺

Use: "s3/tarunbalti"

tarunbalti

Select the events that you want to have trigger the Lambda function. You can optionally set up a prefix or suffix for an event. However, for each bucket, individual events cannot have multiple configurations with overlapping prefixes or suffixes that could match the same object key.

All object create events ×

Prefix - optional
Enter a single optional prefix to limit the notifications to objects with keys that start with matching characters. Any [special characters](#) must be URL encoded.

e.g. images/

Code | Test | Monitor | **Configuration** | Aliases | Versions


General configuration
Triggers
Permissions
Destinations
Function URL
Environment variables
Tags
VPC
RDS databases

Execution role ↺ Edit View role document

Role name
myLambda-role-kia3zvn1 [↗](#)

Resource summary

To view the resources and actions that your function has permission to access, choose a service.

 Amazon CloudWatch Logs
3 actions, 2 resources

By action

By resource

5. Test the Setup

- Upload an image file to your S3 bucket.
- Go to the "Monitoring" tab in your Lambda function to check the logs.
- Alternatively, use CloudWatch Logs to view the output and confirm that the message "An Image has been added" has been logged.

[Amazon S3](#) > [Buckets](#) > [tarunbalti](#) > Upload

Upload [Info](#)

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

Files and folders (1 Total, 3.3 MB)

Remove

Add files

Add folder

All files and folders in this table will be uploaded.

< 1 >

<input type="checkbox"/>	Name	Folder	Type
<input type="checkbox"/>	early_spring_in_japan.png	-	image/png

Services

Search

[Alt+S]

N. Virginia

tarun

Upload succeeded

View details below.

Upload: status

Close

The information below will no longer be available after you navigate away from this page.

Summary

Destination

s3://tarunbalti

Succeeded

1 file, 3.3 MB (100.00%)

Failed

0 files, 0 B (0%)

Files and folders

Configuration

Files and folders (1 Total, 3.3 MB)

< 1 >

Name	Folder	Type	Size	Status	Error
early_spring...	-	image/png	3.3 MB	Succeeded	-

CloudWatch

CloudWatch > Log groups > /aws/lambda/RomaS3ImageLogger > 2024/08/29/[\$LATEST]eba8a380c545407894cbfb69b304ac0c

Log events [Refresh](#) [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](#)

1m 1h UTC timezone Display

Timestamp	Message
No older events at this moment. Retry	
2024-08-29T09:08:12.325Z	INIT_START Runtime Version: python:3.12.v30 Runtime Version ARN: arn:aws:lambda:ap-south-1::runtime:a...
2024-08-29T09:08:12.413Z	START RequestId: 94a4ae5b-dcb0-4a98-a846-486483808ff5 Version: \$LATEST
2024-08-29T09:08:12.414Z	An Image has been added to the bucket romabucket29: cat-image.jpg
2024-08-29T09:08:12.419Z	END RequestId: 94a4ae5b-dcb0-4a98-a846-486483808ff5
2024-08-29T09:08:12.419Z	REPORT RequestId: 94a4ae5b-dcb0-4a98-a846-486483808ff5 Duration: 5.09 ms Billed Duration: 6 ms Memory...
No newer events at this moment. Auto retry paused. Resume	

Conclusion: The integration between AWS S3 and Lambda offers a seamless, event-driven solution that automates tasks like logging image uploads. This enables dynamic responses to changes in S3 without the need for continuously running servers.