Experiment 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.

Theory:

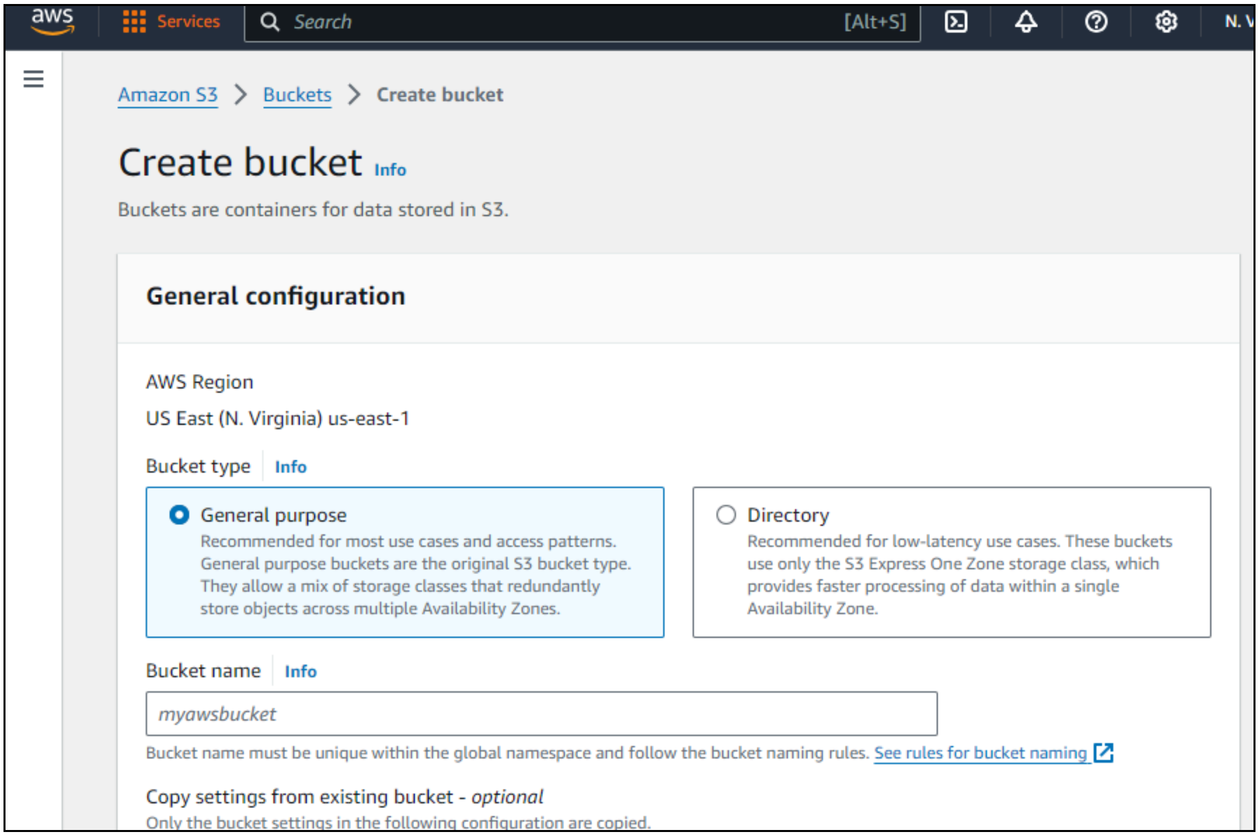
Creating a system to log activities when an image is added to an S3 bucket involves integrating Amazon S3 with AWS Lambda.

Amazon S3:
Amazon S3 (Simple Storage Service) is a service offered by AWS that provides object storage through a web service interface. It's designed to store and retrieve any amount of data from anywhere. You can use S3 to store images, videos, backups, data logs, and more.

AWS Lambda:
AWS Lambda is a serverless compute service that allows you to run code in response to events without provisioning or managing servers. You write your code and set up a trigger, and Lambda takes care of the rest. This means that when a specified event occurs, such as an object being added to an S3 bucket, the Lambda function is automatically invoked.
By setting up a Lambda function to trigger on new uploads to a specific S3 bucket, you can automate logging activities. This function will capture the event, process it, and log the message "An Image has been added." It ensures that every new upload is tracked
efficiently and that you have a record of these actions.
This kind of setup is highly scalable, reliable, and costeffective, leveraging AWS's robust infrastructure. It's particularly useful for applications that require automated monitoring
and logging of uploads for auditing or notification purposes.

Implementation:

1. Create an S3 Bucket: First, create an S3 bucket that will store the objects. This bucket will act as the trigger source for the Lambda function.



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.

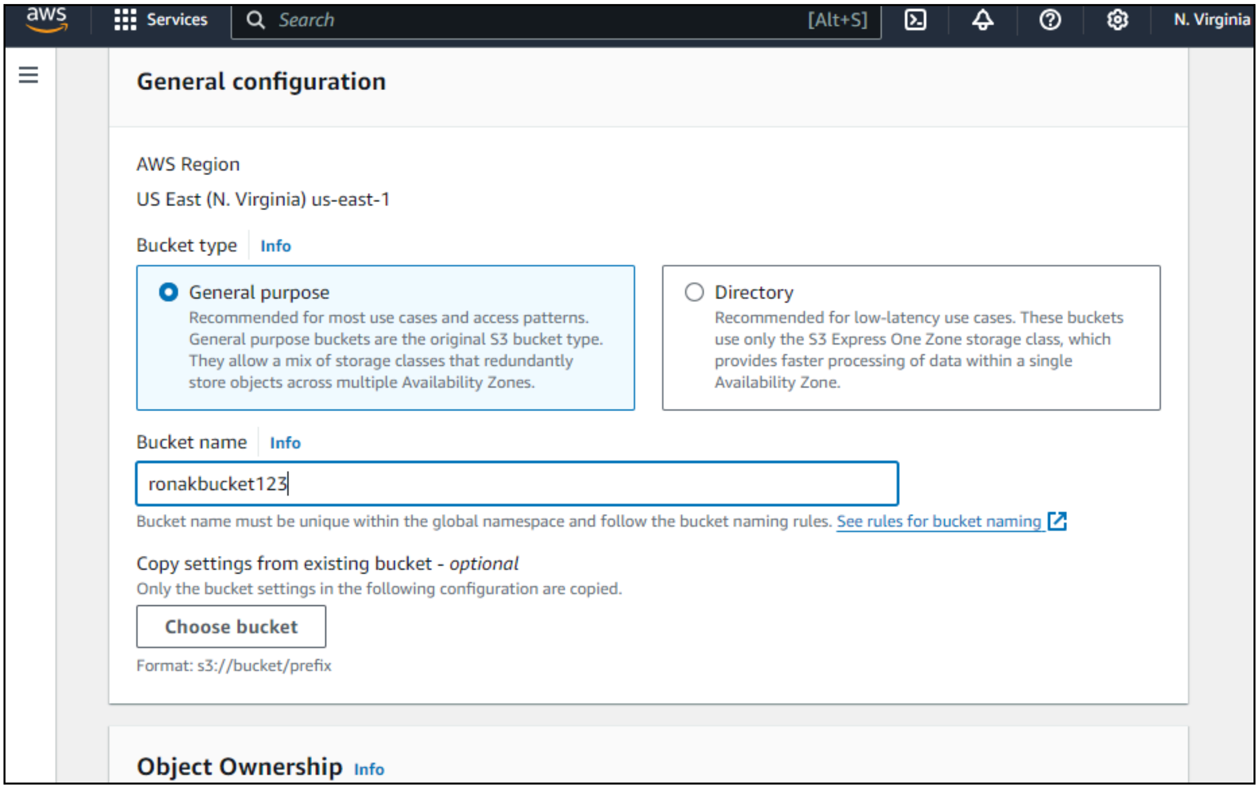
Bucket name Info

myawsbucket

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.



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

ronakbucket123

Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming

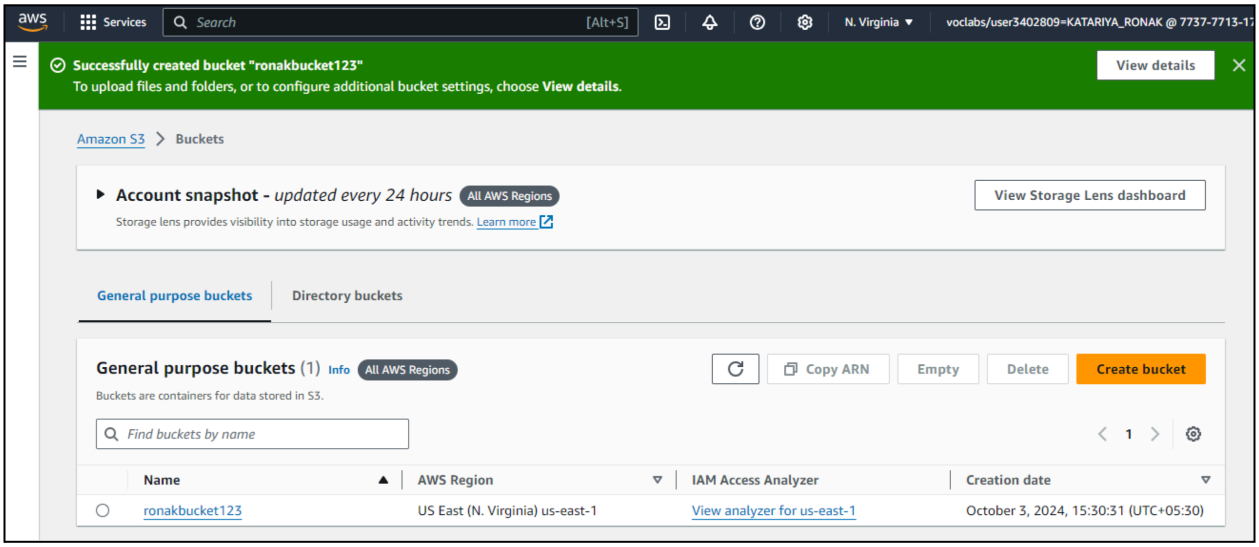
Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Format: s3://bucket/prefix

Object Ownership Info

Bucket created successfully:



Successfully created bucket "ronakbucket123"
To upload files and folders, or to configure additional bucket settings, choose **View details**.

Account snapshot - *updated every 24 hours* All AWS Regions

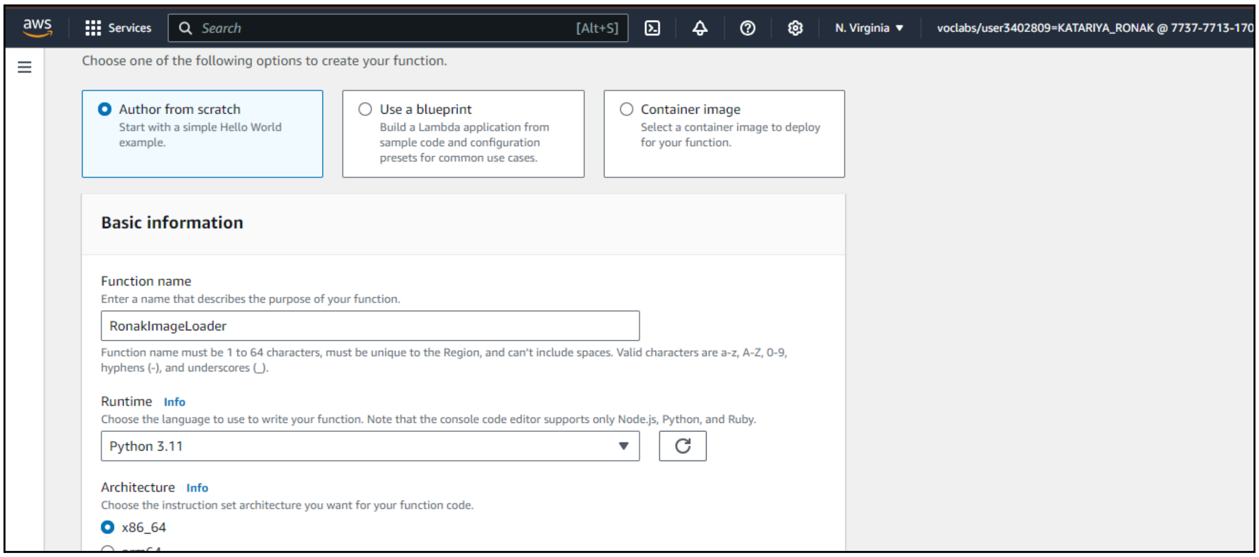
Storage lens provides visibility into storage usage and activity trends. Learn more

General purpose buckets (1) Info All AWS Regions

Buckets are containers for data stored in S3.

	Name	AWS Region	IAM Access Analyzer	Creation date
○	ronakbucket123	US East (N. Virginia) us-east-1	View analyzer for us-east-1	October 3, 2024, 15:30:31 (UTC+05:30)

2. Create the Lambda Function: Set up a new Lambda function using AWS Lambda’s console. You can choose a runtime environment like Python, Node.js, or Java.



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.
RonakImageLoader
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 (_).

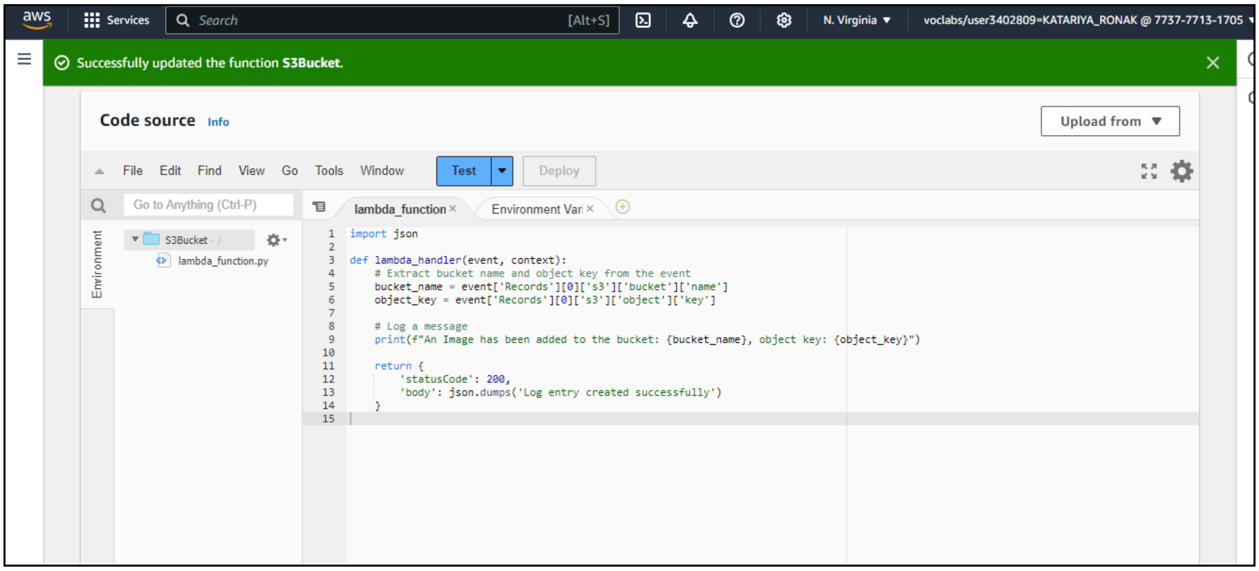
Runtime Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Python 3.11

Architecture Info
Choose the instruction set architecture you want for your function code.
● x86_64
○ arm64

Write code that logs a
message like “An Image has been added” when triggered
This is the code we have to add:

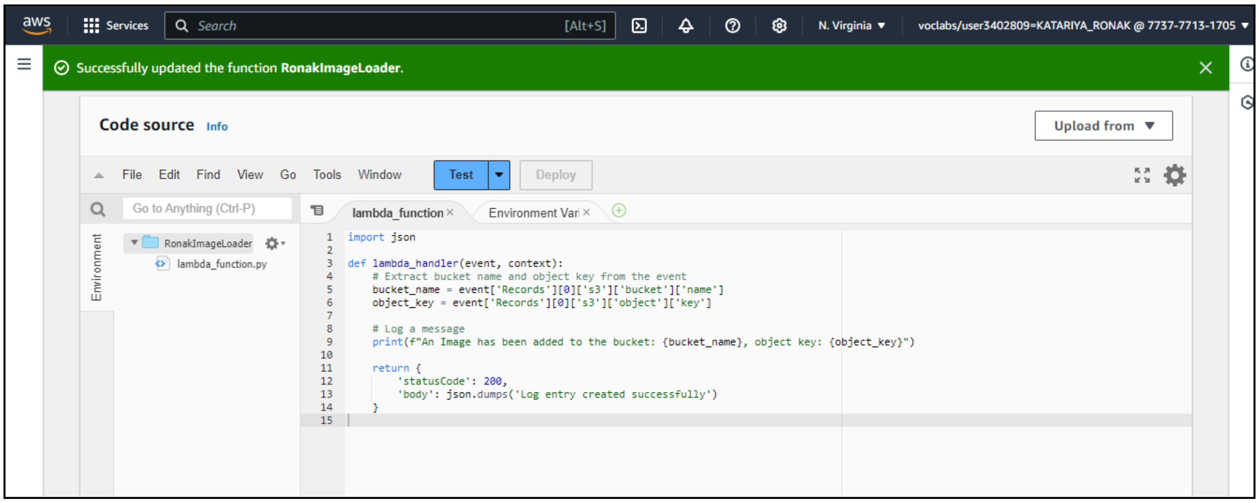
```
import json
import
logging
# Set up logging
logger = logging.getLogger()
logger.setLevel(logging.INFO)
def lambda_handler(event, context):
```

```
# Extract bucket name and object key from the S3
event for record in event['Records']:
bucket = record['s3']['bucket']['name']
key = record['s3']['object']['key']
# Log a message
logger.info(f"An image has been added to bucket {bucket}, object key: {key}")
# Check if the uploaded file is an image (you can adjust the file types
here) if key.lower().endswith(('.png', '.jpg', '.jpeg', '.gif')):
logger.info("Image Uploaded
successfully") else:
logger.info("A non-image file has been added")
return {
'statusCode': 200,
'body': json.dumps('Event processed')
}
```



Successfully updated the function **S3Bucket**.

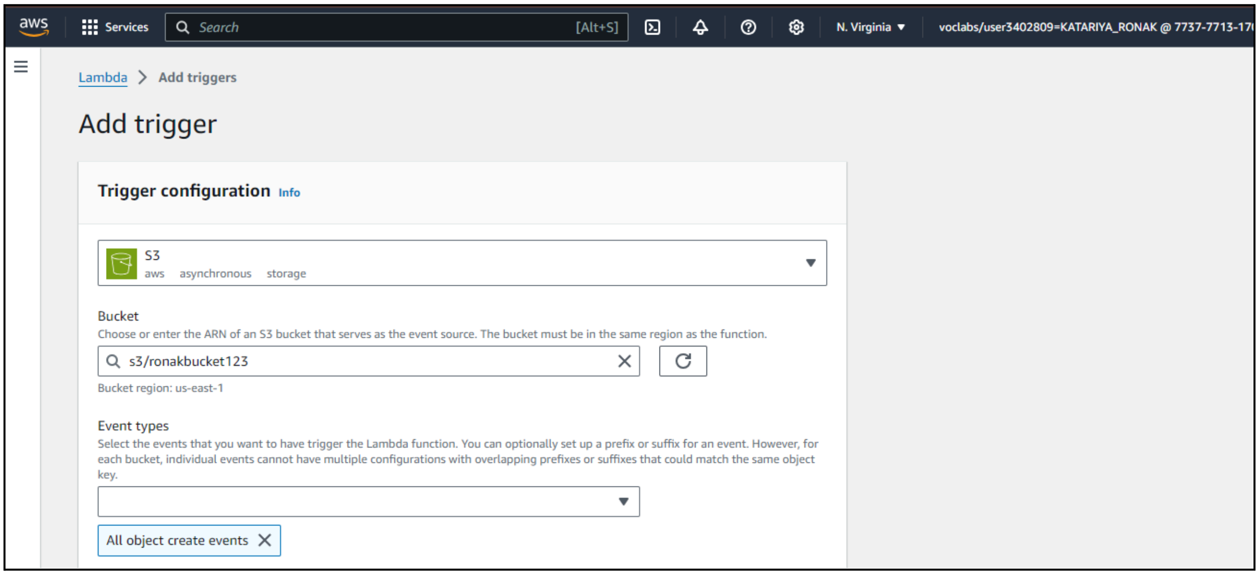
```python
import json

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: {object_key}")

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

```



Successfully updated the function **RonakImageLoader**.

```python
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: {object_key}")

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

```

Configure S3 Trigger: Link the S3 bucket to the Lambda function by setting up a trigger. Specify that the function should be triggered when an object is created in the bucket (e.g., when an image is uploaded).



Lambda > Add triggers

Add trigger

Trigger configuration Info

S3
aws asynchronous storage

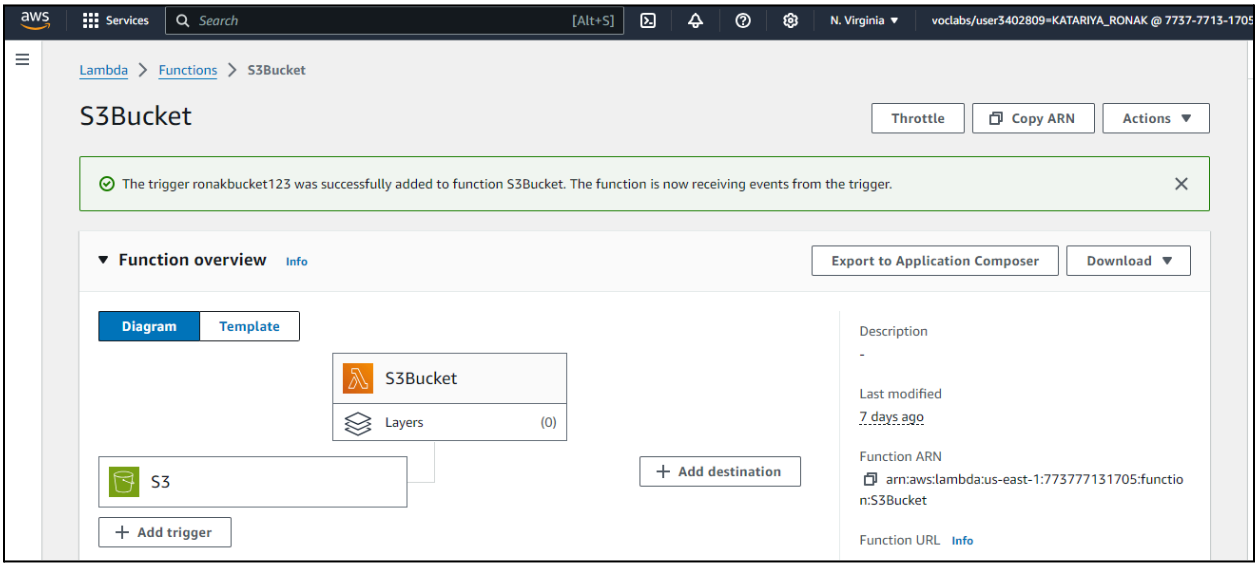
Bucket
Choose or enter the ARN of an S3 bucket that serves as the event source. The bucket must be in the same region as the function.

s3/ronakbucket123

Bucket region: us-east-1

Event types
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



Lambda > Functions > S3Bucket

S3Bucket

The trigger ronakbucket123 was successfully added to function S3Bucket. The function is now receiving events from the trigger.

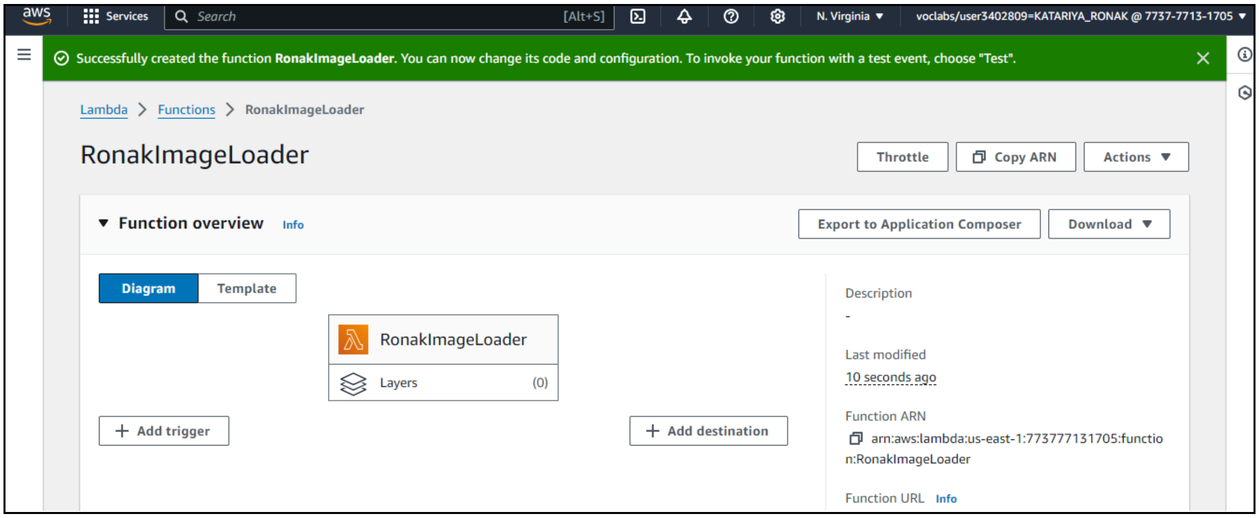
Function overview Info

Description
-

Last modified
7 days ago

Function ARN
arn:aws:lambda:us-east-1:773777131705:function:S3Bucket

Function URL Info



Successfully created the function **RonakImageLoader**. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

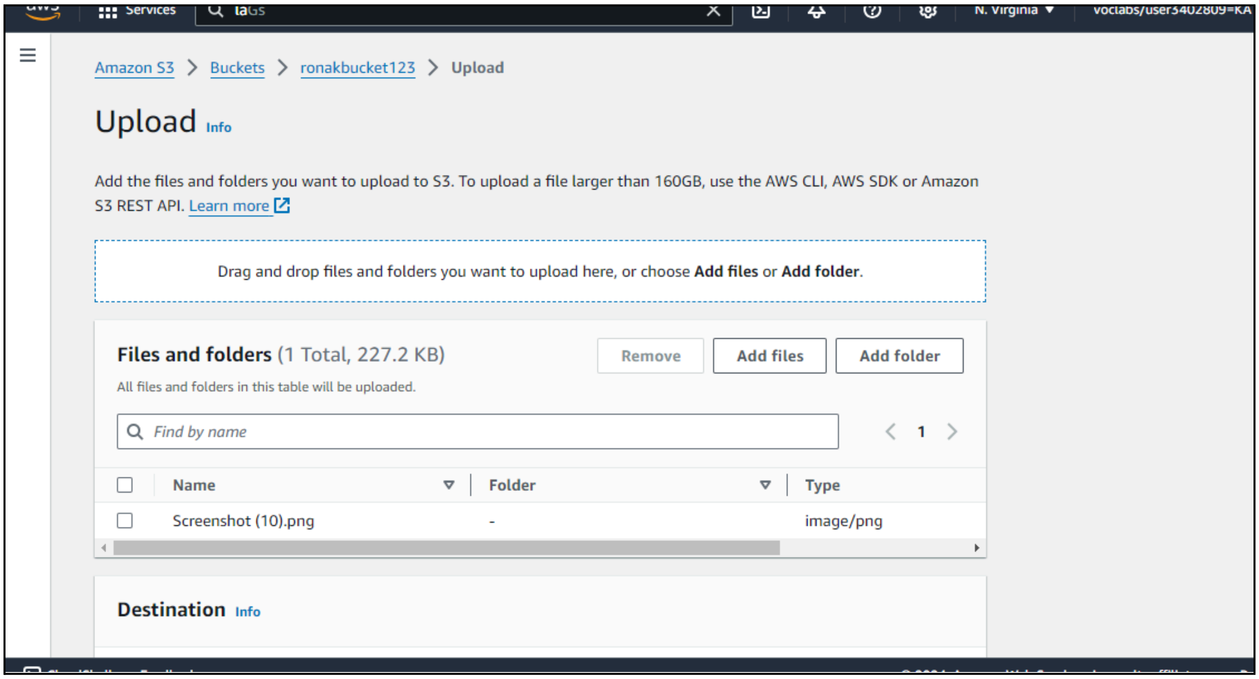
RonakImageLoader

Description
-

Last modified
10 seconds ago

Function ARN
arn:aws:lambda:us-east-1:773777131705:function:RonakImageLoader

Upload an object (e.g., an image) to the S3 bucket to test the trigger



Upload

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

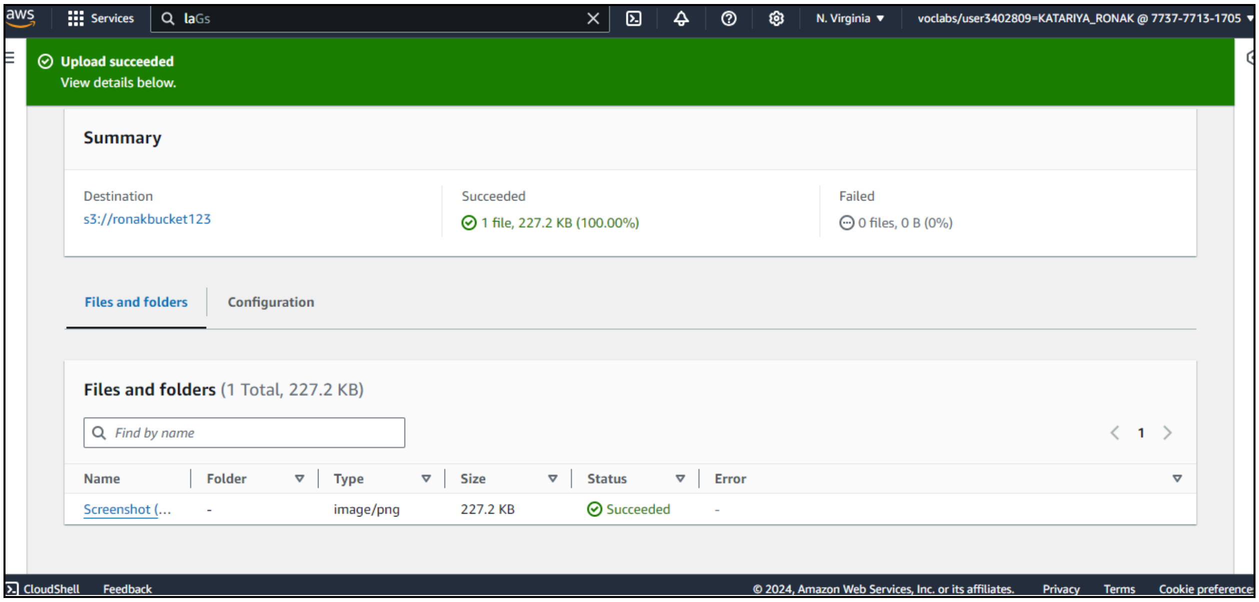
Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

Files and folders (1 Total, 227.2 KB)

All files and folders in this table will be uploaded.

Name	Folder	Type
Screenshot (10).png	-	image/png

Destination


aws
Services
laGs
N. Virginia
voclabs/user3402809=KATARIYA_RONAK @ 7737-7713-1705
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preference

Upload succeeded
View details below.

Summary

Destination	Succeeded	Failed
s3://ronakbucket123	✓ 1 file, 227.2 KB (100.00%)	0 files, 0 B (0%)

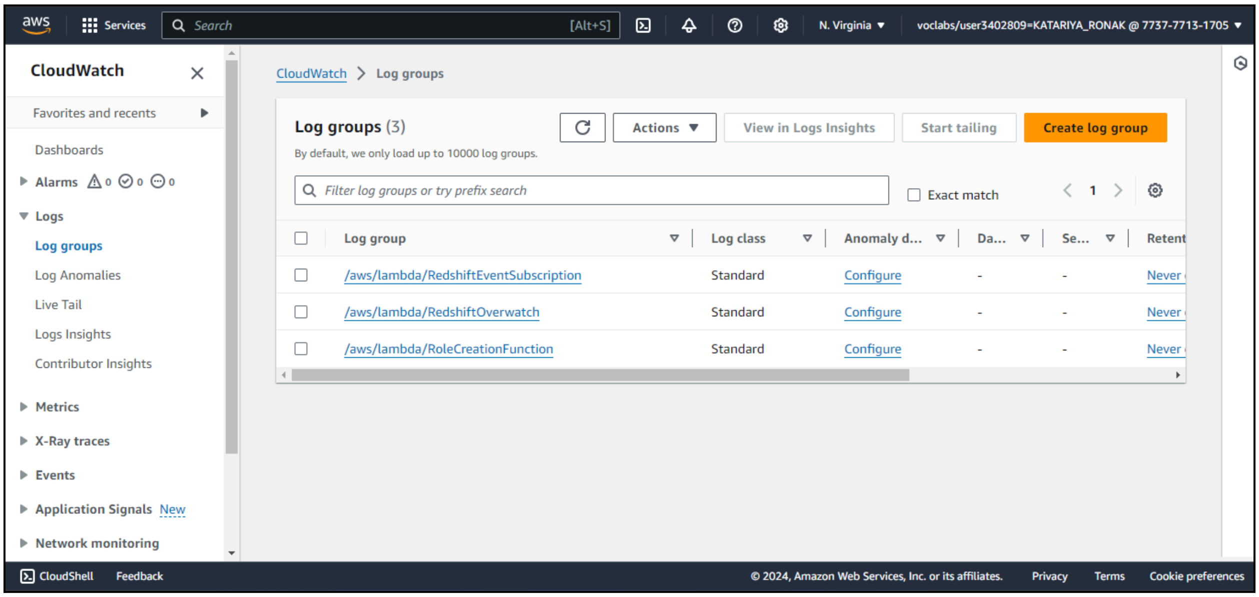
Files and folders | Configuration

Files and folders (1 Total, 227.2 KB)

Find by name

Name	Folder	Type	Size	Status	Error
Screenshot (...	-	image/png	227.2 KB	✓ Succeeded	-

Test the Setup: Upload an object (e.g., an image) to the S3 bucket to test the trigger. The Lambda function should execute and log the message “An Image has been added” in AWS CloudWatch Logs

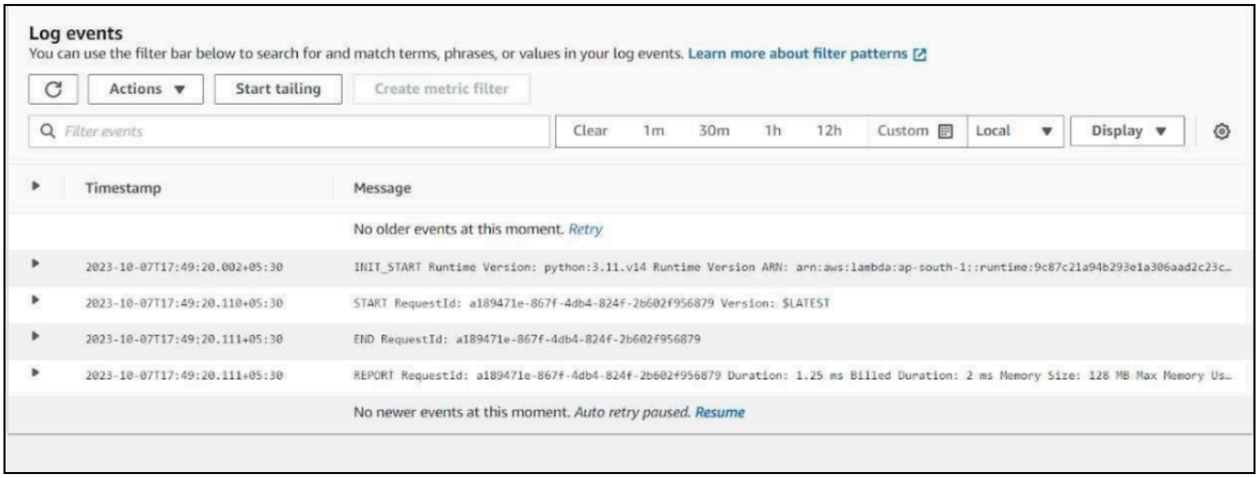


Log groups (3)

By default, we only load up to 10000 log groups.

Filter log groups or try prefix search □ Exact match

□	Log group	Log class	Anomaly d...	Da...	Se...	Retent
□	/aws/lambda/RedshiftEventSubscription	Standard	Configure	-	-	Never
□	/aws/lambda/RedshiftOverwatch	Standard	Configure	-	-	Never
□	/aws/lambda/RoleCreationFunction	Standard	Configure	-	-	Never


Log events
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
Actions
Start tailing
Create metric filter
Filter events
Clear
1m
30m
1h
12h
Custom
Local
Display

Timestamp	Message
	No older events at this moment. *Retry*
2023-10-07T17:49:20.002+05:30	INIT_START Runtime Version: python:3.11.v14 Runtime Version ARN: arn:aws:lambda:ap-south-1::runtime:9c87c21a94b293e1a306aad2c23c…
2023-10-07T17:49:20.110+05:30	START RequestId: a189471e-867f-4db4-824f-2b602f956879 Version: $LATEST
2023-10-07T17:49:20.111+05:30	END RequestId: a189471e-867f-4db4-824f-2b602f956879
2023-10-07T17:49:20.111+05:30	REPORT RequestId: a189471e-867f-4db4-824f-2b602f956879 Duration: 1.25 ms Billed Duration: 2 ms Memory Size: 128 MB Max Memory Us…
	No newer events at this moment. *Auto retry paused.* Resume

Conclusion:

Creating a Lambda function to log “An Image has been added” upon S3 uploads showcases AWS's serverless capabilities. This integration automates event handling, enhances operational efficiency, and reduces manual oversight, allowing developers to focus on application logic while ensuring responsive and scalable cloud applications.