Name: Snehal A. Patil D15A 39

Advanced Devops Lab
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

Theory:

AWS Lambda and S3 Integration: AWS Lambda allows you to execute code in response to various events, including those triggered by Amazon S3. When an object is added to an S3 bucket, it can trigger a Lambda function to execute, allowing for event-driven processing without managing servers.

Workflow:

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.

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.

○ Write code that logs a message like “An Image has been added” when triggered.

3. Set Up Permissions:

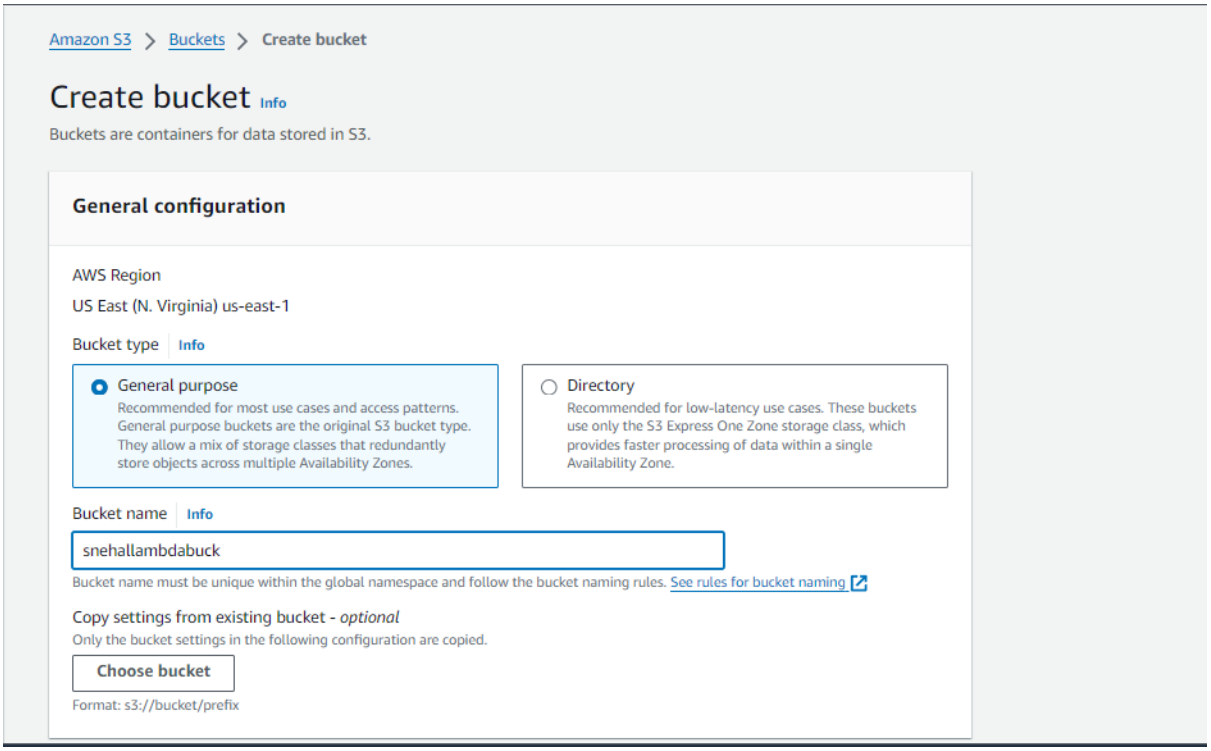
○ Ensure that the Lambda function has the necessary permissions to access S3. You can do this by attaching an IAM role with policies that allow reading from the bucket and writing logs to CloudWatch.

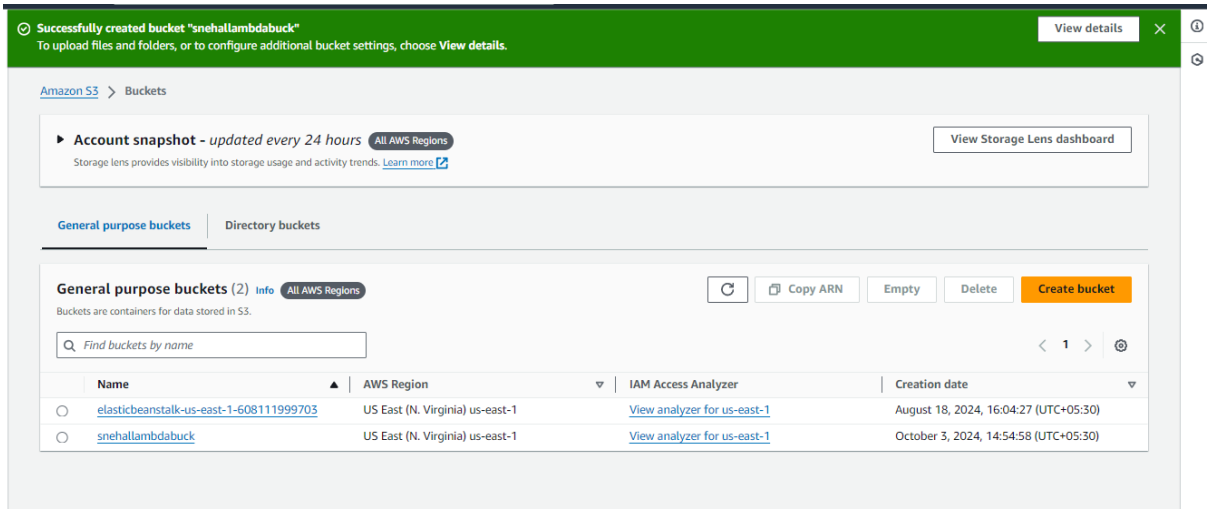
4. Configure S3 Trigger:

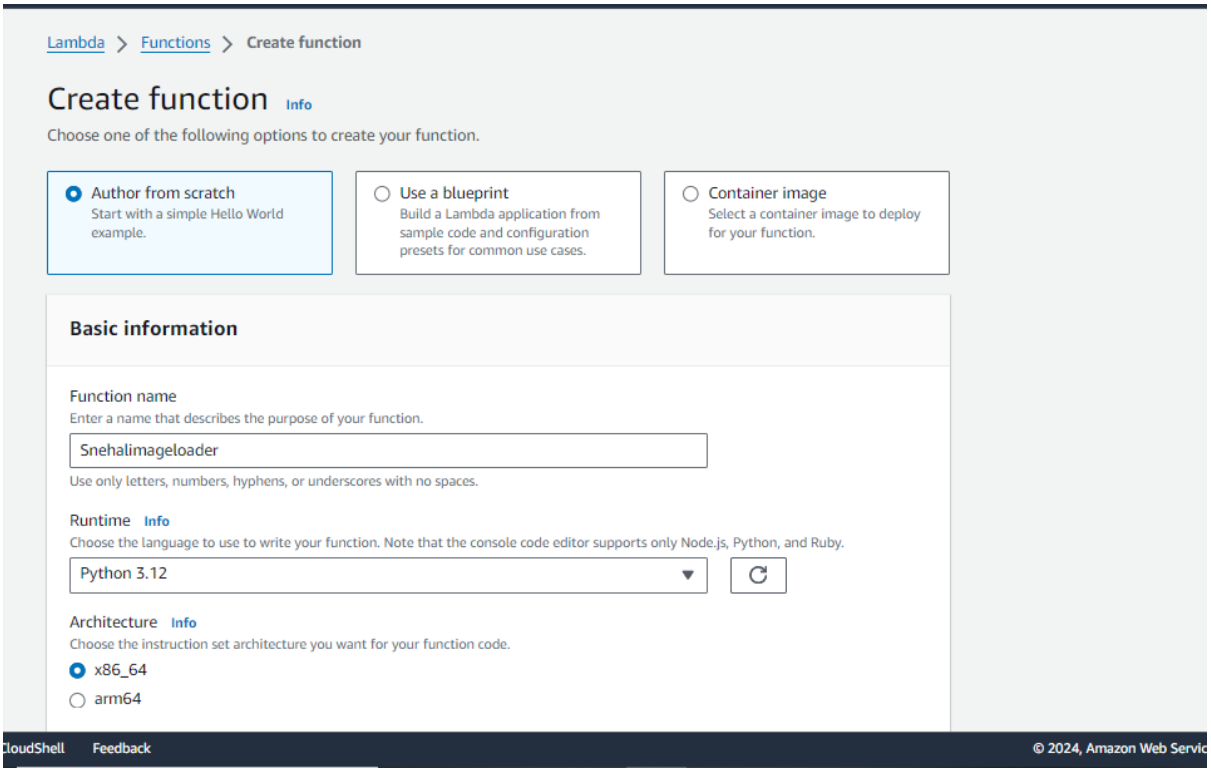
○ Link the S3 bucket to the Lambda function by setting up a trigger. Specify that the function should be triggered when an object is created in the bucket (e.g., when an image is uploaded).

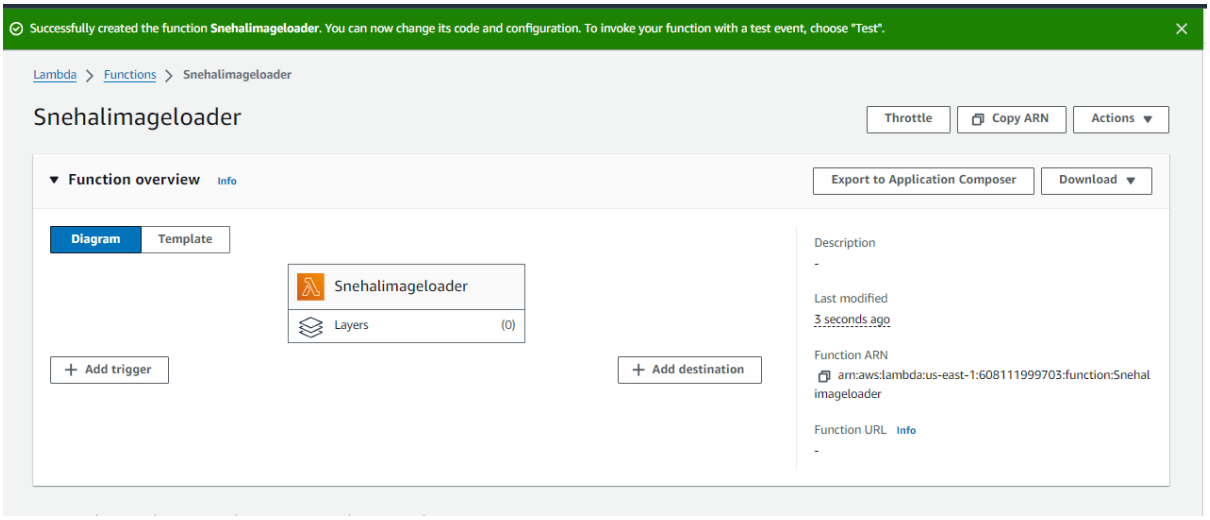
5. Test the Setup:

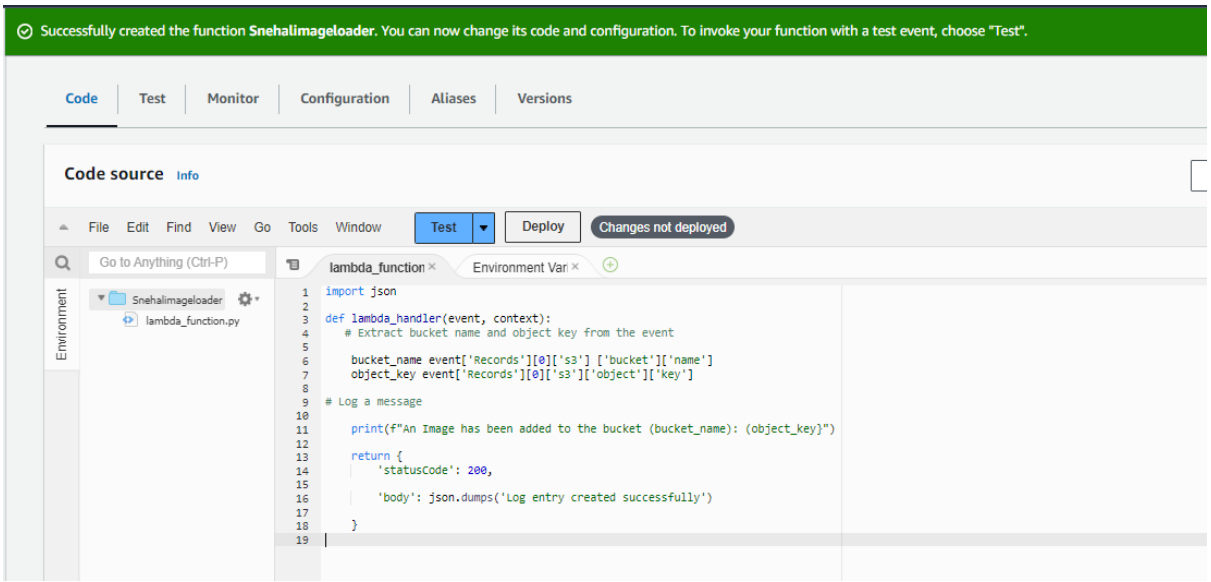
○ Upload an object (e.g., an image) to the S3 bucket to test the trigger. The Lambda function should execute and log the message “An Image has been added” in AWS CloudWatch

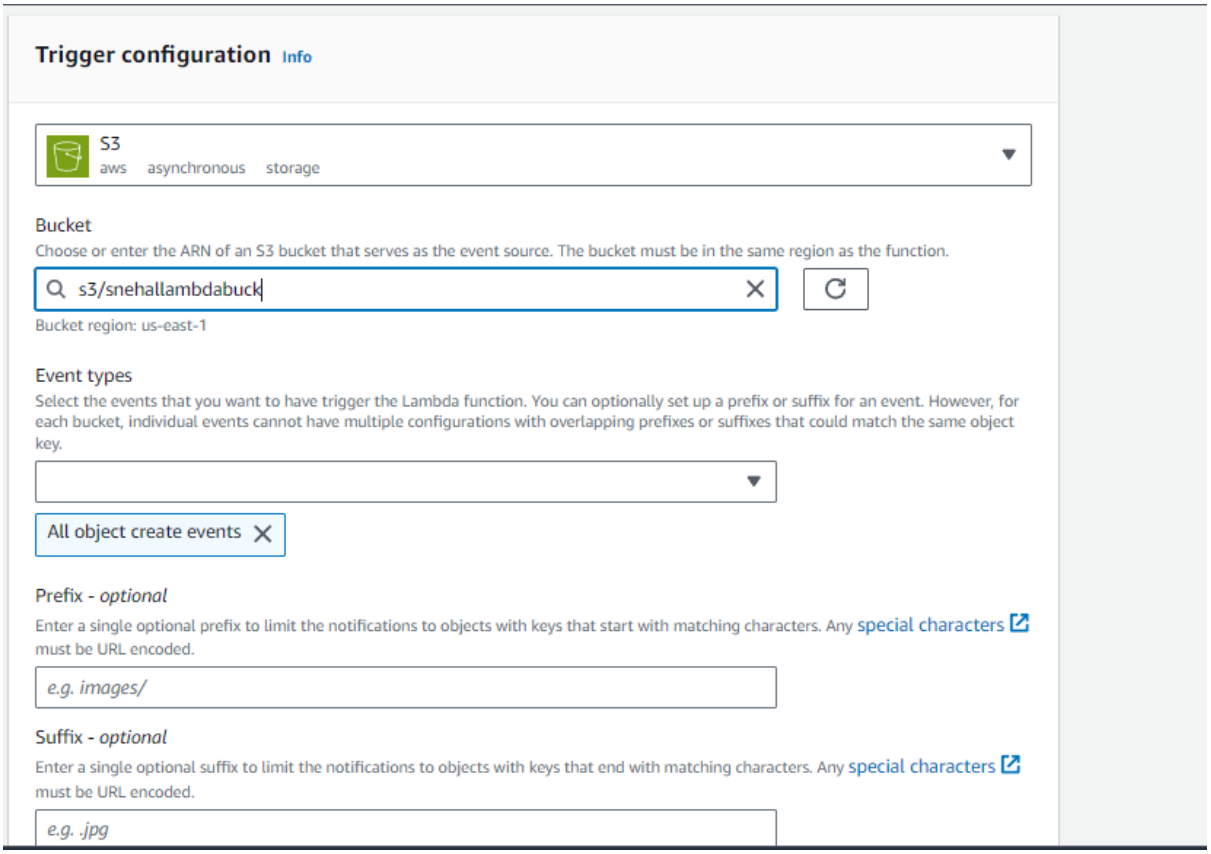

Amazon S3 > Buckets > Create bucket
Create bucket Info
Buckets are containers for data stored in S3.
General configuration
AWS Region
US East (N. Virginia) us-east-1
Bucket type Info
General purpose
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket 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
snehallambdabuck
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.
Choose bucket
Format: s3://bucket/prefix

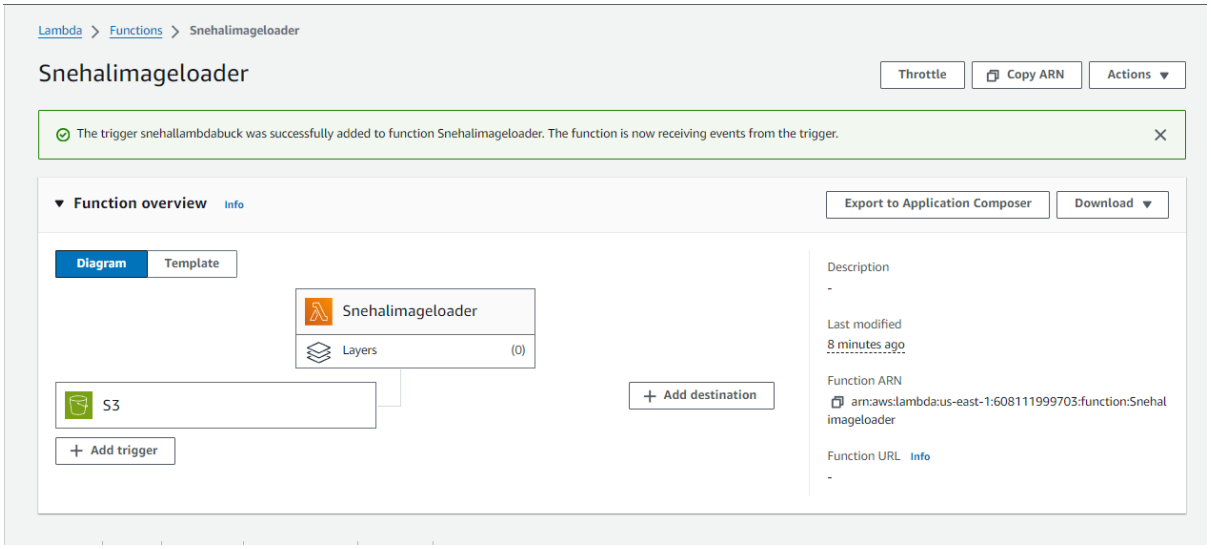

Successfully created bucket "snehallambdabuck"
To upload files and folders, or to configure additional bucket settings, choose View details.
View details
Amazon S3 > Buckets
Account snapshot - updated every 24 hours
All AWS Regions
Storage lens provides visibility into storage usage and activity trends. Learn more
View Storage Lens dashboard
General purpose buckets
Directory buckets
General purpose buckets (2) Info
All AWS Regions
Buckets are containers for data stored in S3.
Copy ARN
Empty
Delete
Create bucket
Find buckets by name
Name
AWS Region
IAM Access Analyzer
Creation date
elasticbeanstalk-us-east-1-608111999703
US East (N. Virginia) us-east-1
View analyzer for us-east-1
August 18, 2024, 16:04:27 (UTC+05:30)
snehallambdabuck
US East (N. Virginia) us-east-1
View analyzer for us-east-1
October 3, 2024, 14:54:58 (UTC+05:30)

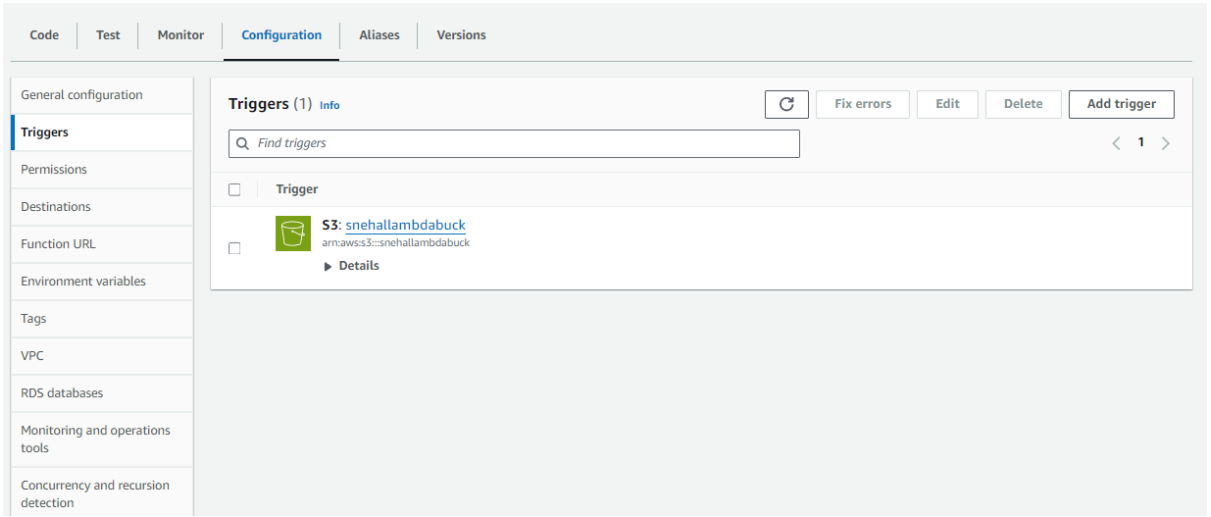

Lambda > Functions > Create function
Create function Info
Choose one of the following options to create your function.
Author from scratch
Start with a simple Hello World example.
Use a blueprint
Build a Lambda application from sample code and configuration presets for common use cases.
Container image
Select a container image to deploy for your function.
Basic information
Function name
Enter a name that describes the purpose of your function.
Snehalimageloader
Use only letters, numbers, hyphens, or underscores with no spaces.
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 3.12
Architecture Info
Choose the instruction set architecture you want for your function code.
x86_64
arm64
CloudShell
Feedback
© 2024, Amazon Web Servic

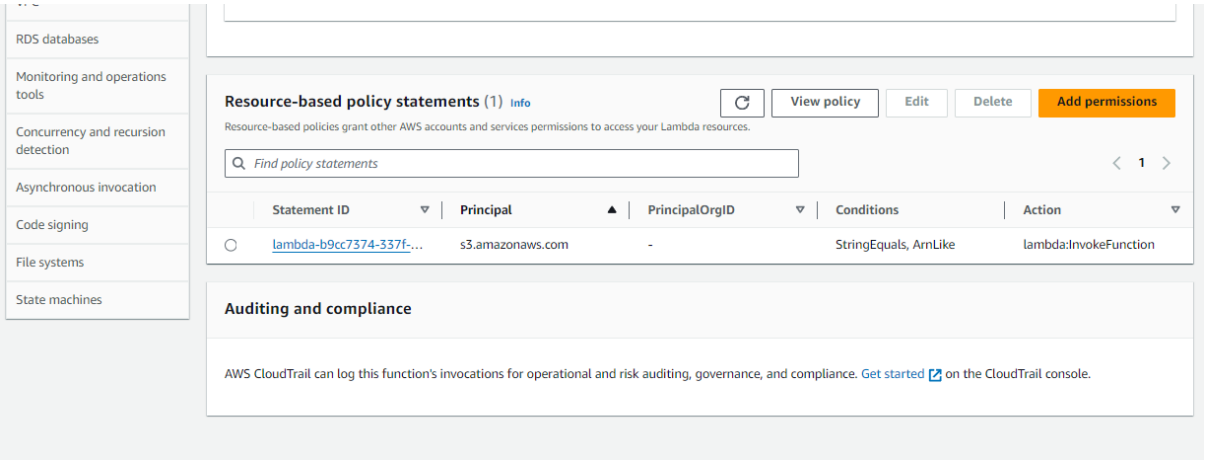

Successfully created the function Snehalimageloader. You can now change its code and configuration. To invoke your function with a test event, choose "Test".
Lambda > Functions > Snehalimageloader
Snehalimageloader
Throttle
Copy ARN
Actions
Function overview Info
Export to Application Composer
Download
Diagram
Template
Snehalimageloader
Layers (0)
Add trigger
Add destination
Description
-
Last modified
3 seconds ago
Function ARN
arn:aws:lambda:us-east-1:608111999703:function:Snehalimageloader
Function URL Info
-

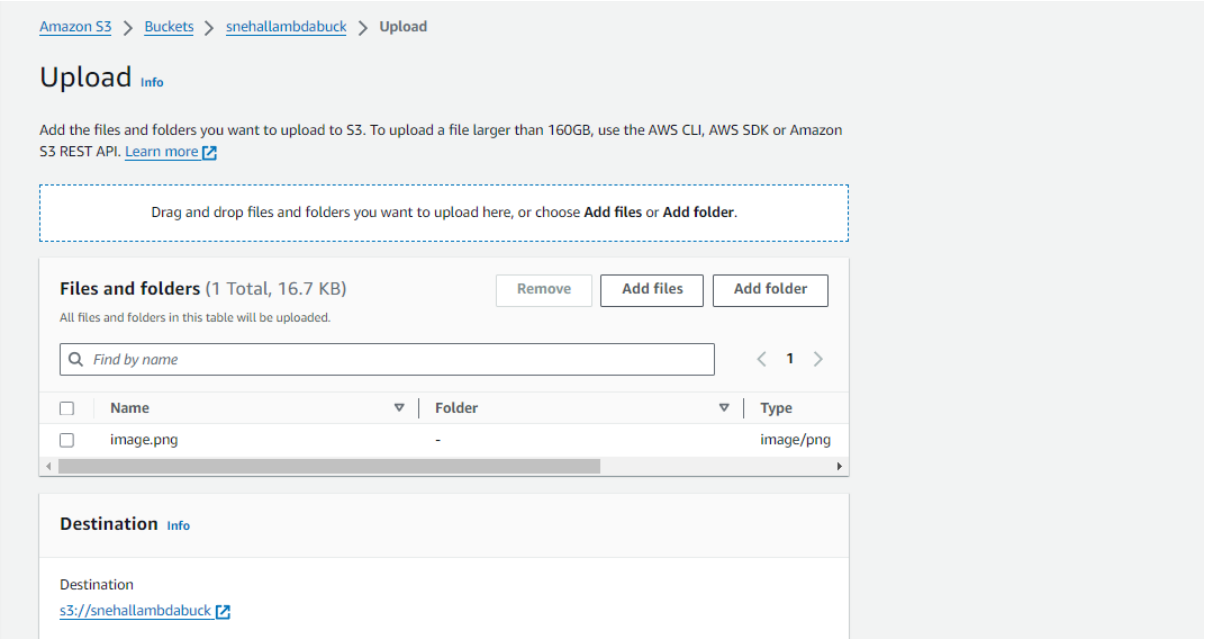

Successfully created the function Snehalimageloader. You can now change its code and configuration. To invoke your function with a test event, choose "Test".
Code
Test
Monitor
Configuration
Aliases
Versions
Code source Info
File Edit Find View Go Tools Window
Test
Deploy
Changes not deployed
Go to Anything (Ctrl-P)
lambda_function
Environment Var
Environment
Snehalimageloader
lambda_function.py
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')
}

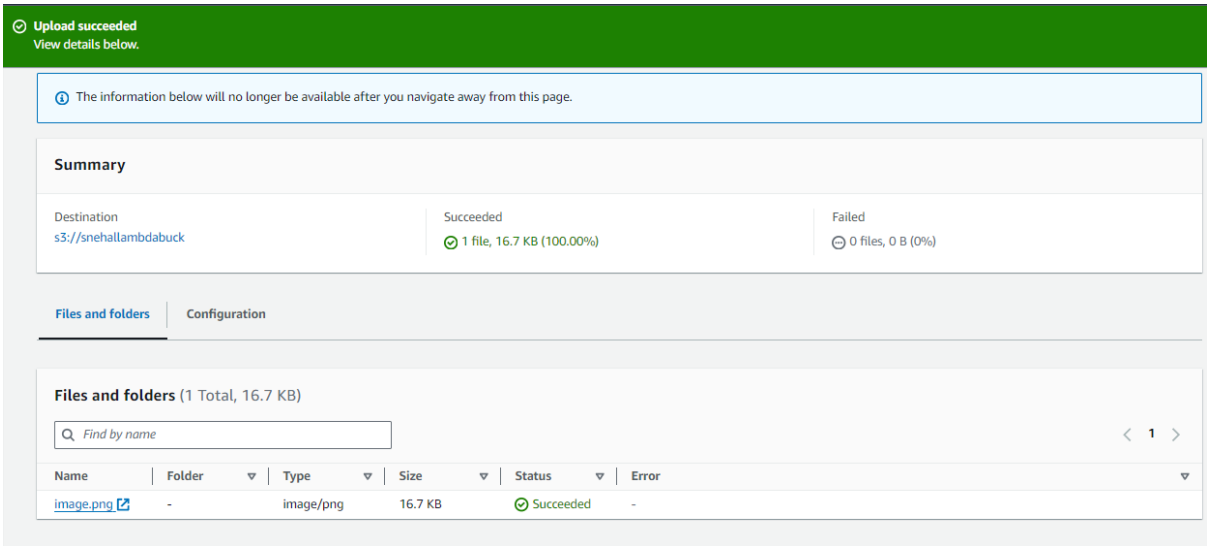

Trigger configuration Info
S3
aws asynchronous storage
Bucket
Choose or enter the ARN of an S3 bucket that serves as the event source. The bucket must be in the same region as the function.
s3/snehallambdabuck
Bucket region: us-east-1
Event types
Select the events that you want to have trigger the Lambda 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
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/
Suffix - optional
Enter a single optional suffix to limit the notifications to objects with keys that end with matching characters. Any special characters must be URL encoded.
e.g. .jpg

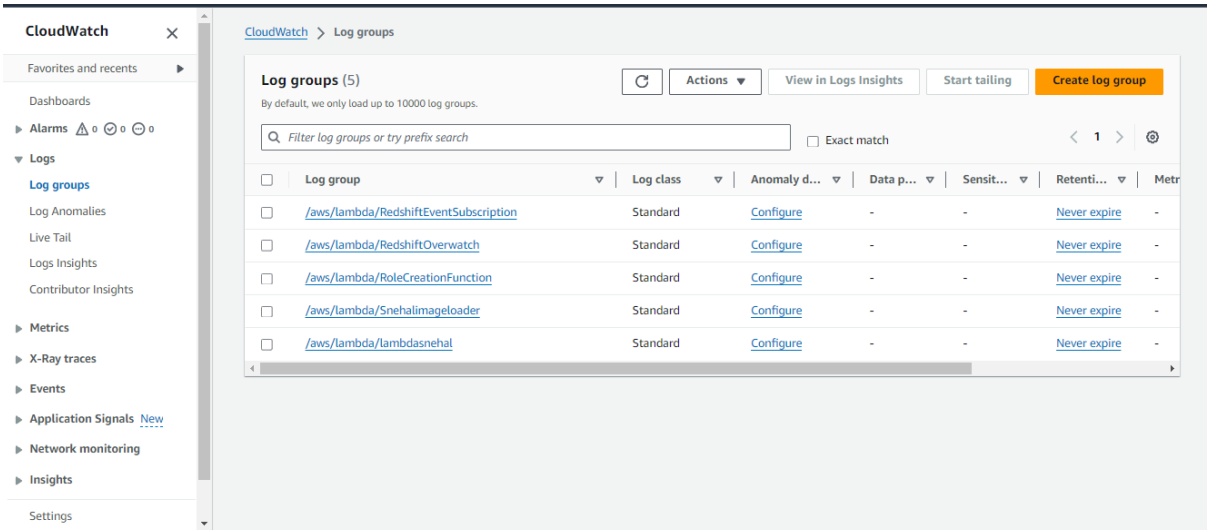

Lambda > Functions > Snehalimageloader
Snehalimageloader
Throttle
Copy ARN
Actions
The trigger snehallambdabuck was successfully added to function Snehalimageloader. The function is now receiving events from the trigger.
Function overview Info
Export to Application Composer
Download
Diagram
Template
Snehalimageloader
Layers (0)
S3
Add trigger
Add destination
Description
-
Last modified
8 minutes ago
Function ARN
arn:aws:lambda:us-east-1:608111999703:function:Snehalimageloader
Function URL Info
-

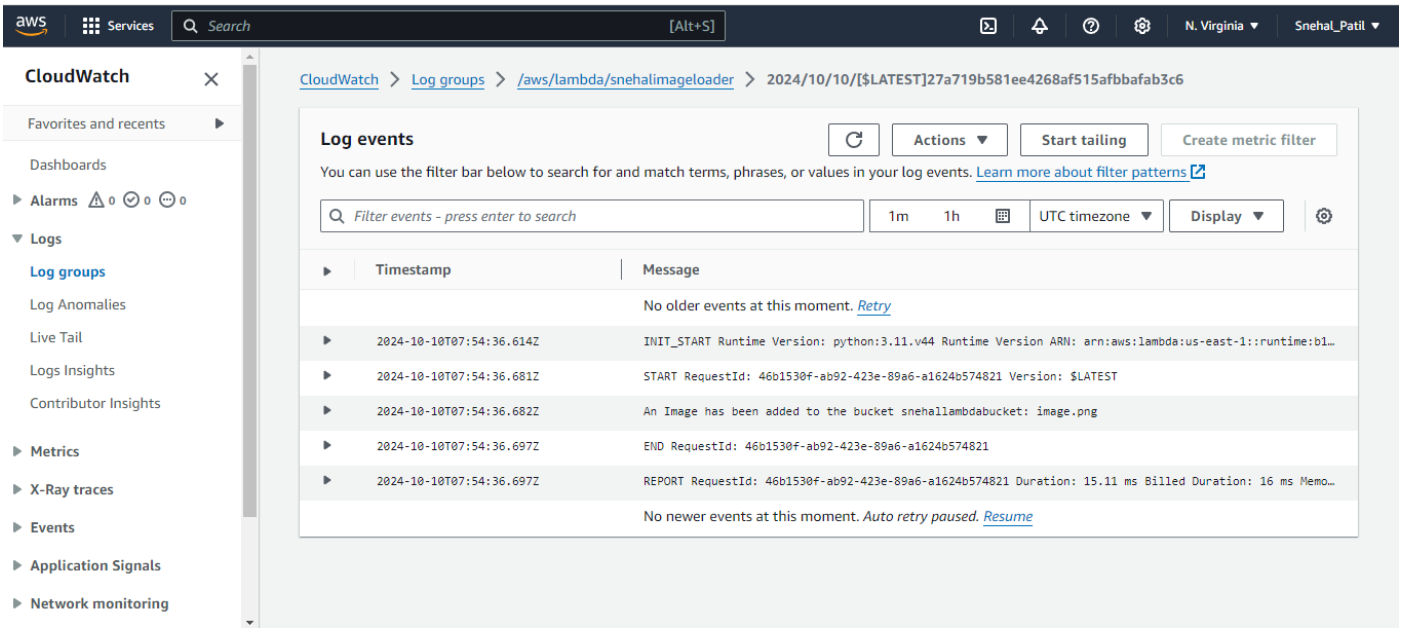

Code
Test
Monitor
Configuration
Aliases
Versions
General configuration
Triggers
Permissions
Destinations
Function URL
Environment variables
Tags
VPC
RDS databases
Monitoring and operations tools
Concurrency and recursion detection
Triggers (1) Info
Fix errors
Edit
Delete
Add trigger
Find triggers
Trigger
S3: snehallambdabuck
arn:aws:s3:::snehallambdabuck
Details


RDS databases
Monitoring and operations tools
Concurrency and recursion detection
Asynchronous invocation
Code signing
File systems
State machines
Resource-based policy statements (1) Info
View policy
Edit
Delete
Add permissions
Resource-based policies grant other AWS accounts and services permissions to access your Lambda resources.
Find policy statements
Statement ID
Principal
PrincipalOrgID
Conditions
Action
lambda-b9cc7374-337f-...
s3.amazonaws.com
-
StringEquals, ArnLike
lambda:InvokeFunction
Auditing and compliance
AWS CloudTrail can log this function's invocations for operational and risk auditing, governance, and compliance. Get started on the CloudTrail console.


Amazon S3 > Buckets > snehallambdabuck > 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, 16.7 KB)
Remove
Add files
Add folder
All files and folders in this table will be uploaded.
Find by name
Name
Folder
Type
image.png
-
image/png
Destination Info
Destination
s3://snehallambdabuck


Upload succeeded
View details below.
The information below will no longer be available after you navigate away from this page.
Summary
Destination
s3://snehallambdabuck
Succeeded
1 file, 16.7 KB (100.00%)
Failed
0 files, 0 B (0%)
Files and folders
Configuration
Files and folders (1 Total, 16.7 KB)
Find by name
Name
Folder
Type
Size
Status
Error
image.png
-
image/png
16.7 KB
Succeeded
-





Conclusion:

Integrating AWS Lambda with S3 allows for real-time, automated processing of events such as file uploads. In this example, a Lambda function is configured to log a message whenever an image is added to a specific S3 bucket. This setup demonstrates the power and flexibility of serverless computing by automating tasks without requiring manual intervention or server management. By leveraging AWS Lambda, developers can efficiently handle event-driven workflows, reduce operational overhead, and quickly deploy scalable solutions that respond to specific actions within cloud environments.