

## Project introduction

Topic:-Create a serverless image processing application that automatically resizes and optimizes images uploaded to an Amazon S3 bucket.

AWS S3 (Simple Storage Service) is a cloud data storage service. It is one of the most popular services of AWS. It has high scalability, availability, security and is cost effective. S3 has different storage tiers depending on the use case. Some common use cases of AWS S3 are:

- Storage: It can be used for storing large amounts of data.
- Backup and Archive: S3 has different storage tiers based on how frequent the data is accessed which can be used to backup critical data at low costs.
- Static website: S3 offers static website hosting through HTML files stored in S3.
- Data lakes and big data analytics: Companies can use [AWS S3](#) as a data lake and then run analytics on it for getting business insights and take critical decisions.

[AWS Lambda](#) is a serverless, event-driven compute service that lets you run code for virtually any type of application or backend

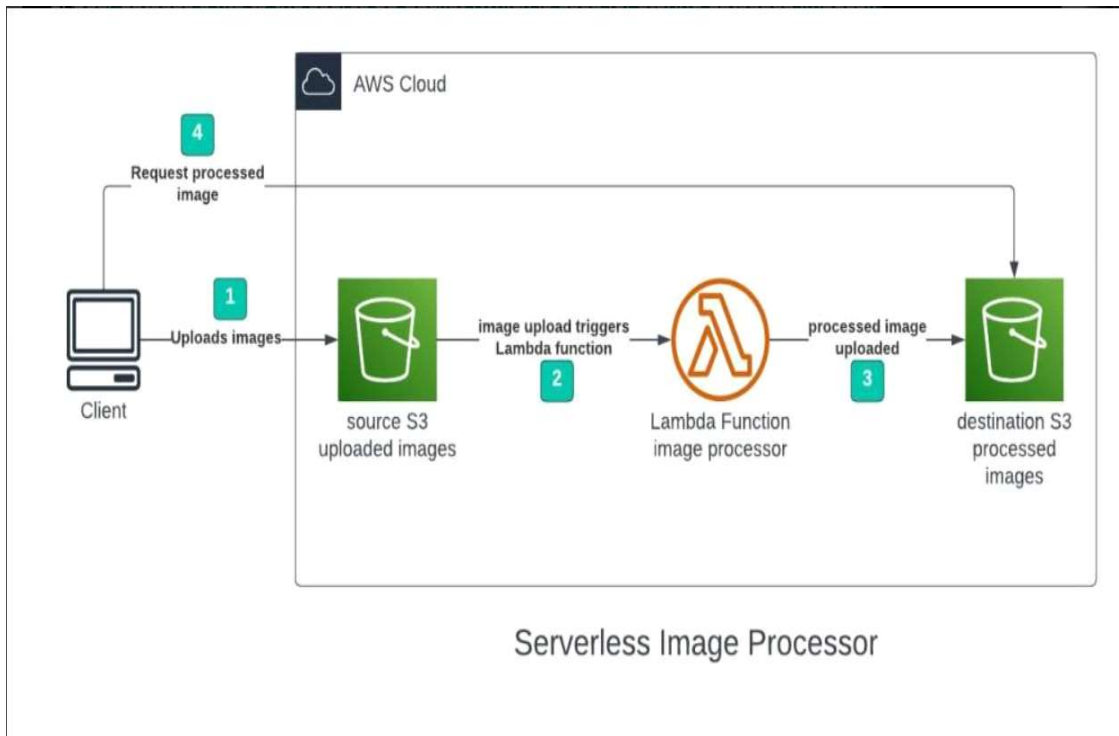
service without provisioning or managing servers. Lambda functions run on demand i.e. they execute only when needed and you pay only for what you compute. Lambda is well integrated with many other AWS services. It supports a wide variety of programming languages.

Some common use cases for AWS Lambda are:-

1. You can use Lambda for processing files as they are uploaded in an S3 bucket or whenever some event triggers the function.
2. Lambda can also be used for creating websites. This is cost effective because you are charged only for the time when the servers are running.
3. You can pass a data stream to your Lambda function and then create analysis from that.

### **Serverless Image Processing**

The Serverless Image Handler solution helps you embed images on your websites and mobile applications to drive user engagement. It uses the [sharp](#) Node.js library to provide high-speed image processing without sacrificing image quality. To minimize your costs of image optimization, manipulation, and processing, this solution automates version control and provides flexible storage and compute options for file reprocessing



## **Step 1 – Creating S3 buckets**

**We will use two S3 buckets:**

1. source Bucket: For storing uploaded images.
2. destination Bucket: For storing processed images.

## Create bucket [Info](#)

Buckets are containers for data stored in S3. [Learn more](#)

### General configuration

Bucket name

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

AWS Region

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

## **Step 2 – Configuring S3 bucket policy**

**In ‘Block Public Access settings for this bucket’ section disable “block all public access”.**

- ☐ **Block all public access**  
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.
- ☐ **Block public access to buckets and objects granted through new access control lists (ACLs)**  
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
  - ☐ **Block public access to buckets and objects granted through any access control lists (ACLs)**  
S3 will ignore all ACLs that grant public access to buckets and objects.
  - ☐ **Block public access to buckets and objects granted through new public bucket or access point policies**  
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
  - ☐ **Block public and cross-account access to buckets and objects through any public bucket or access point policies**  
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.



**Turning off block all public access might result in this bucket and the objects within becoming public**  
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

☒ I acknowledge that the current settings might result in this bucket and the objects within becoming public.

### Step 3 – Creating police in lam

**Go to AWS I am console. Navigate to policies section. Click Create policies in (JSON) and name it “ImageBucketpolicy”. Leave all other settings as default. Create the policy.**

[IAM](#) > Policies

**Policies (1202)** [Info](#)

A policy is an object in AWS that defines permissions.

Filter by Type

Search  All types

< 1 2 3 4 5 6 7 ... 61 >

Policy name	Type	Used as	Description
<a href="#">AccessAnalyzerService...</a>	AWS managed	None	Allow Access Analyzer to analyze r
<a href="#">AdministratorAccess</a>	AWS managed - job function	Permissions policy (1)	Provides full access to AWS service

## Policy:

```
"Version": "2012-10-17",
"Statement": [
{
  "Effect": "Allow",
  "Action": [
    "logs:PutLogEvents",
    "logs:CreateLogGroup",
    "logs:CreateLogStream"
  ],
  "Resource": "arn:aws:logs:*:*:*"
},
{
  "Effect": "Allow",
  "Action": ["s3:GetObject"],
  "Resource": "arn:aws:s3:::BUCKET_NAME/*"
},
{
  "Effect": "Allow",
  "Action": ["s3:PutObject"],
```

```

    "Resource": "arn:aws:s3:::DEST_BUCKET/*"
  }
]
}

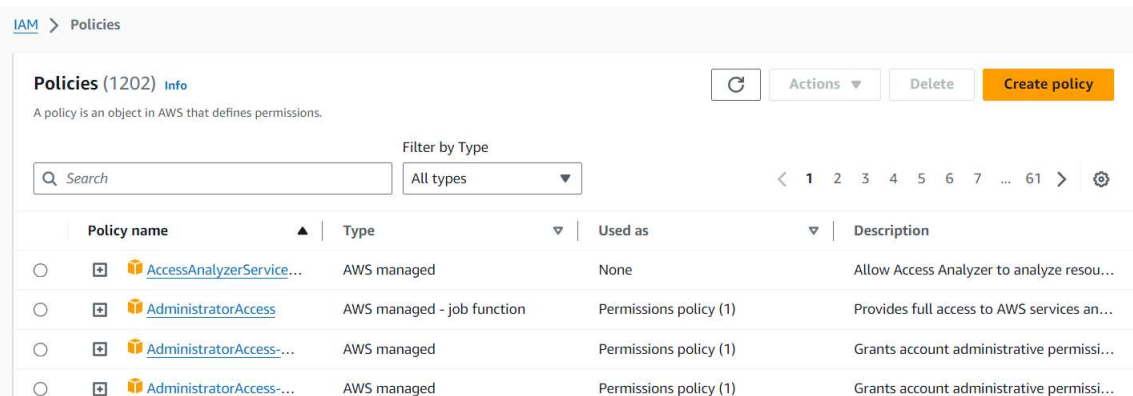
```

**\*BUCKET\_NAME = SOURCE BUCKET NAME**  
**\*DEST\_BUCKET = DESTINATION BUCKET NAME**

## Step 4 –creating role in I am

### Following Steps are Follows

- \*Go to aws I am console
- \*Create role
- \*name imageresizerlambdarole
- \*Use case – Lambda
- \*Select-ImageBucketPolicy
- \*Then create role



## Step 3 – Creating Lambda function

Go to AWS Lambda console. Navigate to Functions section. Click Create Function and name it “ImageProcessing”. Select runtime as “NodeJS 16.x” and architecture as “x86\_64”. Leave all other

settings as default. Create the 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.

demo-image-resizer-lambda

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.

Node.js 18.x 

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

☒ x86\_64  
☐ arm64

**\*Change default execution role**



▼ Change default execution role

Execution role

Choose a role that defines the permissions of your function. To create a custom role, go to the [IAM console](#).

- ☐ Create a new role with basic Lambda permissions
- ☒ Use an existing role
- ☐ Create a new role from AWS policy templates

Existing role

Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.

ImageResizerLambdaRole ▼

↻

[View the ImageResizerLambdaRole role](#) on the IAM console.

► Advanced settings

Cancel

Create function

## \*Create function

## Step 6 -upload zip file in Lambda function

Lambda > Functions > demo-image-resize-lambda

### demo-image-resize-lambda

Throttle Copy ARN Actions

Function overview Info

Export to Application Composer Download

Diagram Template

demo-image-resize-lambda

Layers (0)

S3

+ Add trigger

+ Add destination

Description

Last modified

5 hours ago

Function ARN

arn:aws:lambda:ap-south-1:992382690538:function:demo-image-resize-lambda

Function URL Info

Code Test Monitor Configuration Aliases Versions

Code source Info


Upload from

\*Zip file link-<https://github.com/OneLightWebDev/image-resizer-lambda>

## Step 7 - Edit environment variables

## Edit environment variables

### Environment variables

You can define environment variables as key-value pairs that are accessible from your function code. These are useful to store configuration settings without the need to change function code. [Learn more](#) 

Key

DEST\_BUCKET

Value

DESTINATION\_BUCKET\_NAME

Remove

Add environment variable

► Encryption configuration

Cancel

Save

## **Step 8 – Test Lambda Function**

- \*Go to AWS Lambda console. Navigate to Functions section.
- \*open function then will be created
- \*open test console
- \*template=s3-put

To invoke your function without saving an event, configure the JSON event, then choose Test.

Test event action

☒ Create new event

☐ Edit saved event

Event name

MyEventName

Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.

Event sharing settings

☒ Private

This event is only available in the Lambda console and to the event creator. You can configure a total of 10. [Learn more](#)

☐ Shareable

This event is available to IAM users within the same account who have permissions to access and use shareable events. [Learn more](#)

Template - optional

s3-put

## EVENT JSON

In event json we can change only 3 value  
\*name, arn, key

```
17     "x-amz-id-2": "EXAMPLE123/5678abcdefghijklambda
18   },
19   "s3": {
20     "s3SchemaVersion": "1.0",
21     "configurationId": "testConfigRule",
22     "bucket": {
23       "name": "example-bucket",
24       "ownerIdentity": {
25         "principalId": "EXAMPLE"
26       },
27       "arn": "arn:aws:s3:::example-bucket"
28     },
29     "object": {
30       "key": "test%2Fkey",
31       "size": 1024,
32       "eTag": "0123456789abcdef0123456789abcdef",
33       "sequencer": "0A1B2C3D4E5F678901"
34     }
35   }
36 }
37 ]
38 }
```

"name":"demo-user-images-bucket",->

"arn": "arn:aws:s3::: demo-user-images-bucket "->

"key": "golf-course.jpg",->

Amazon S3 > Buckets

**Account snapshot**  
 Storage lens provides visibility into storage usage and activity trends. [Learn more](#)

View Storage Lens dashboard

Total storage  
⌚ Pending

Object count  
⌚ Pending

Average object size  
⌚ Pending

You can enable advanced metrics in the "default-account-dashboard" configuration.

**Buckets (2)** [Info](#)  
 Buckets are containers for data stored in S3. [Learn more](#)

	Name	AWS Region	Access	Creation date
<input type="radio"/>	demo-user-images-bucket	US West (N. California) us-west-1	Bucket and objects not public	September 23, 2023, 20:53:31 (UTC-07:00)
<input type="radio"/>	demo-user-thumbnails-bucket	US West (N. California) us-west-1	Bucket and objects not public	September 23, 2023, 20:53:50 (UTC-07:00)

Amazon S3 > Buckets > demo-user-images-bucket

**demo-user-images-bucket** [Info](#)

[Objects](#) | [Properties](#) | [Permissions](#) | [Metrics](#) | [Management](#) | [Access Points](#)

**Objects (1)**  
 Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

1

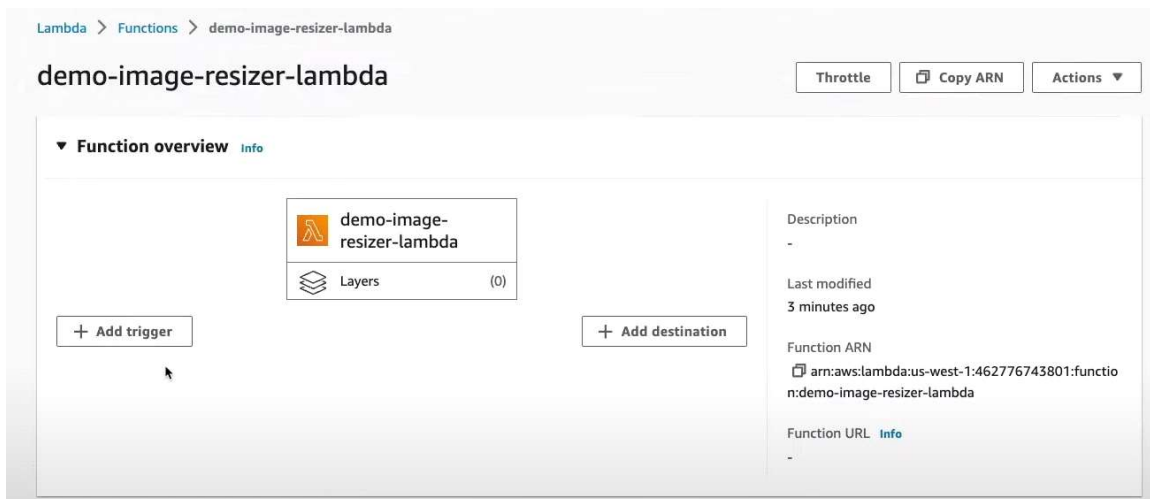
<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	golf-course.jpg	jpg	September 23, 2023, 20:54:08 (UTC-07:00)	96.2 KB	Standard

**Now we can test**



## Step 5 – Creating S3 trigger


- \*Add trigger
- \*Select s3
- \*choose source Bucket name
- \*Now Add



Lambda > Add trigger

## Add trigger

Trigger configuration [Info](#)

 **S3**  
aws   asynchronous   storage

**Bucket**  
Please select the S3 bucket that serves as the event source. The bucket must be in the same region as the function.  
 × ↺  
Bucket region: us-west-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.

**Suffix - optional**  
Enter a single optional suffix to limit the notifications to objects with keys that end with matching characters.

## Upload image in source Bucket

Amazon S3 > Buckets > demo-user-images-bucket

demo-user-images-bucket [Info](#)

Objects | Properties | Permissions | Metrics | Management | Access Points

Objects (2)  
Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

↺ 📄 Copy S3 URI 📄 Copy URL 📄 Download 🔗 Open 🗑 Delete ⌵ Actions 📁 Create folder 📤 Upload

Name

▲

Type

▼

Last modified



▼

Size

▼

Storage class

▼

<input type="checkbox"/>	 golf-course.jpg	jpg	September 23, 2023, 20:54:08 (UTC-07:00)	96.2 KB	Standard
<input checked="" type="checkbox"/>	 tiger.jpg	jpg	September 23, 2023, 22:46:56 (UTC-07:00)	280.7 KB	Standard

15

**Original image**



**Destination Bucket**



Amazon S3 > Buckets > demo-user-thumbnails-bucket

## demo-user-thumbnails-bucket info

Objects | Properties | Permissions | Metrics | Management | Access Points

**Objects (2)**

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

[Refresh](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	<a href="#">golf-course.jpg</a>	jpg	September 23, 2023, 22:27:45 (UTC-07:00)	5.9 KB	Standard
<input type="checkbox"/>	<a href="#">tiger.jpg</a>	jpg	September 23, 2023, 22:46:59 (UTC-07:00)	7.5 KB	Standard

## Resize Image

