MANAGING CLOUD SOLUTIONS PROJECT

Designing a Simple AWS Architecture.

Description:

The goal of this project is to design and implement a **serverless image resizer** using AWS services. The architecture will be cost-effective, scalable, and reliable, demonstrating key AWS services for image processing. This project will provide hands-on experience in building event-driven serverless applications, leveraging AWS Lambda, Amazon S3, and other cloud services.

The system automatically resizes uploaded images and stores them in an optimized format for efficient delivery, ensuring minimal latency and seamless performance.

Essential AWS Services used:

Basic AWS Services:

- **Amazon S3** – For storing original and resized images.
- **AWS Lambda** – To process and resize images dynamically.
- **AWS IdentityandAccessManagement(IAM)** – To control access and permissions.
- **Amazon CloudWatch** – For monitoring and logging function execution.

EXAMPLE

1. Scenario:

Acontent management system requires automatic image resizing to optimize storage and improve website performance. Users upload images in high resolution, but different formats (thumbnail, medium, large) are needed for web display. The goal is to build a serverless image resizer that automatically processes uploaded images and stores the resized versions for optimized delivery

2. Problem Statement:

Design anAWS-based serverless image resizer that ensures automatic, scalable, and cost-efficient image processing.

3. Objectives:

- To create a fully automated image resizing solution using AWS serverless services.
- To ensure high availability and scalability without managing servers.
- To optimize costs by using AWS Lambda and pay-per-use services.
- To secure the system using AWS Identity and Access Management (IAM) policies.
- To enable real-time image processing with minimal latency.

4. Outcomes:

- **Automated imageprocessing:** Images are resized immediately upon upload.
- **Scalability:** The system handles varying image upload rates without downtime. **Cost**
- **optimization:** **Fast** Uses AWS Lambda and S3 to reduce operational costs.
- **performance:** Images are available with low latency via CloudFront.
- **Security:** Fine-grained IAM policies ensure restricted access.

5. Proposed AWS Components

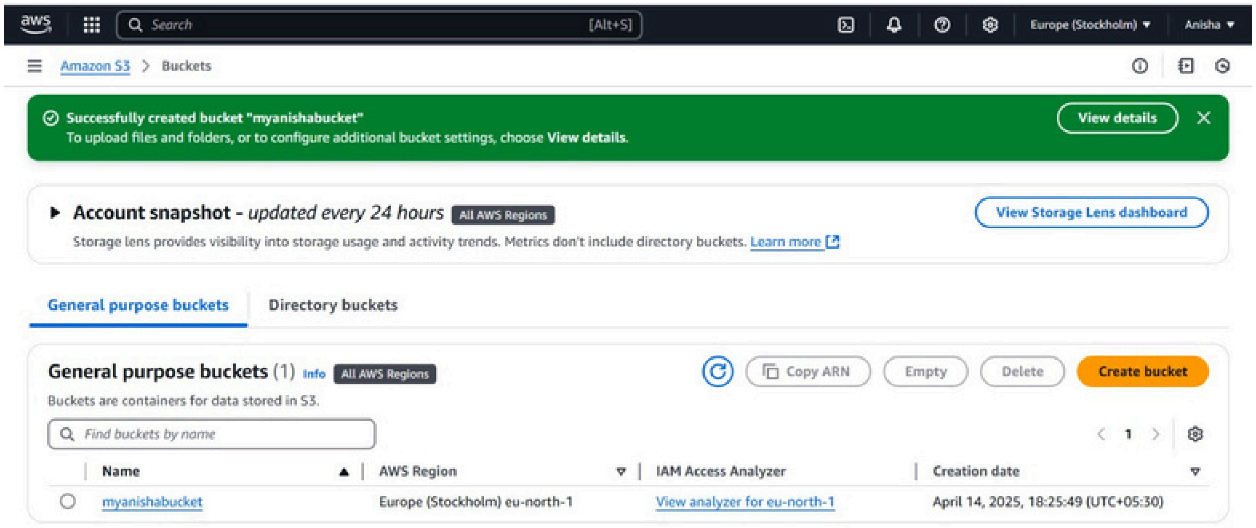
1. **Amazon S3:** Stores original andresized images.
2. **AWS Lambda:** Resizes imageswhen uploaded to S3.
3. **Amazon CloudWatch:** MonitorsLambda execution and logs errors.

4. **AWS IAM:** Manages permissions for secure access.

Implementation 1: Traditional Server-Based Approach:

1. Setup S3 Buckets

- Goto **AWS S3 Console**.
- Create two buckets:
  - o original-images-bucket (for storing uploaded original images).
  - o After creating upload an image(JPG or JPEG).
  - o resized-images-bucket (for storing resized images).

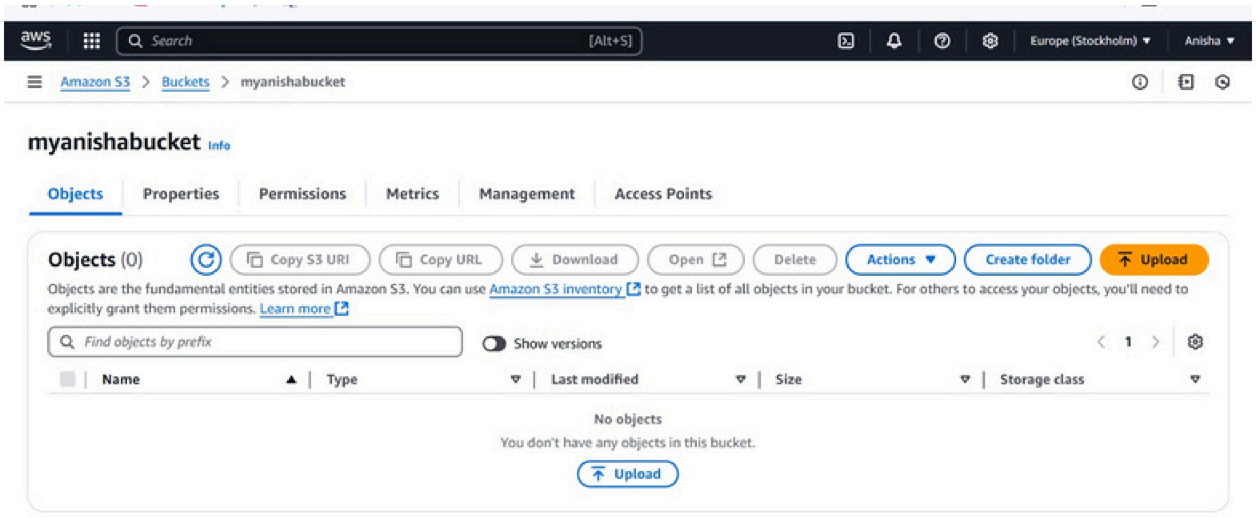


Successfully created bucket "myanishabucket"
To upload files and folders, or to configure additional bucket settings, choose **View details**.

Account snapshot - *updated every 24 hours* All AWS Regions
Storage lens provides visibility into storage usage and activity trends. Metrics don't include directory buckets. 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
myanishabucket	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	April 14, 2025, 18:25:49 (UTC+05:30)



myanishabucket Info

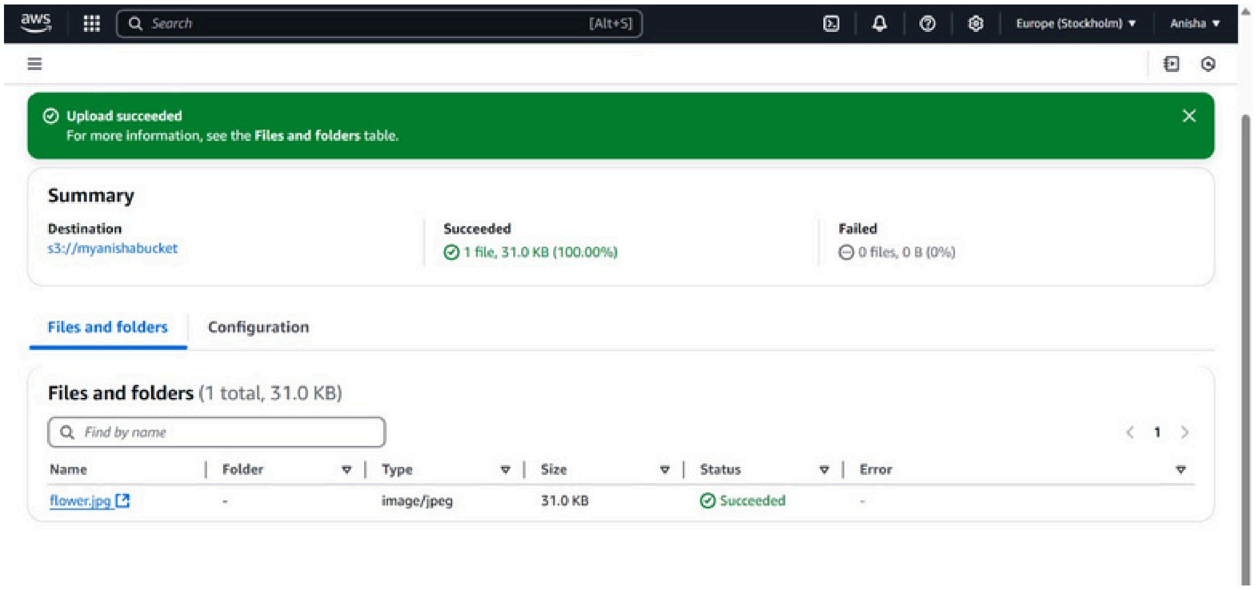
Objects (0)

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

Name	Type	Last modified	Size	Storage class

No objects

You don't have any objects in this bucket.



Upload succeeded

For more information, see the **Files and folders** table.

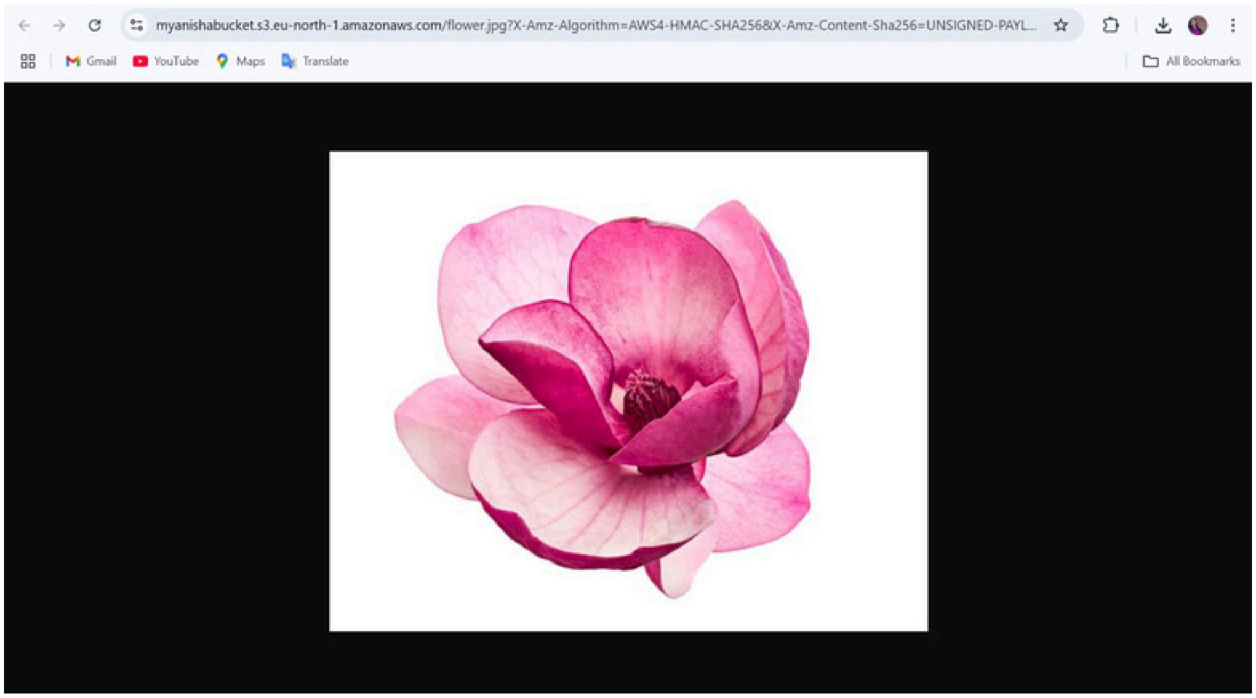
Summary

Destination	Succeeded	Failed
s3://myanishabucket	1 file, 31.0 KB (100.00%)	0 files, 0 B (0%)

Files and folders (1 total, 31.0 KB)

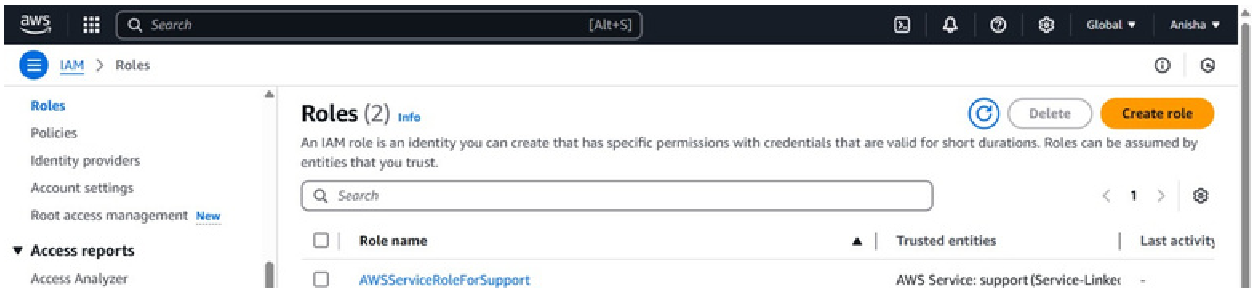
Name	Folder	Type	Size	Status	Error
flower.jpg	-	image/jpeg	31.0 KB	Succeeded	-

Result


myanishabucket.s3.eu-north-1.amazonaws.com/flower.jpg?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Content-Sha256=UNSIGNED-PAYL...
Gmail
YouTube
Maps
Translate
All Bookmarks

2. Create a New Role in IAM:

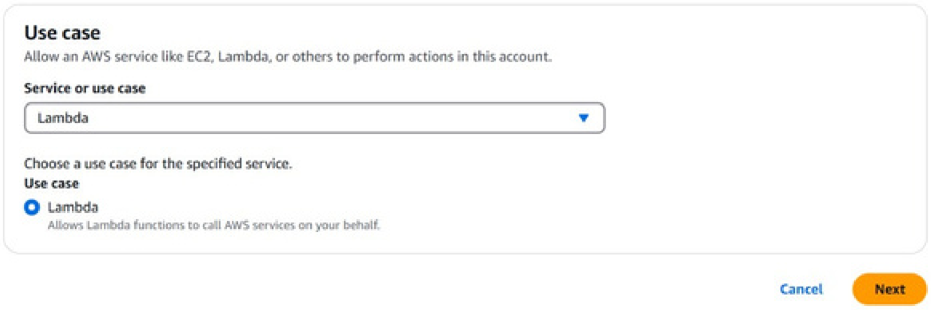
- Goto **AWS IAM ROLES** and then create a new role.
- Choose existing permissions like **AWSS3FullAccess** and **AWSLambdaBasicExecutionRole**.
- Then create the role.



Roles (2) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Role name	Trusted entities	Last activity
AWSServiceRoleForSupport	AWS Service: support (Service-Linke	-



Use case

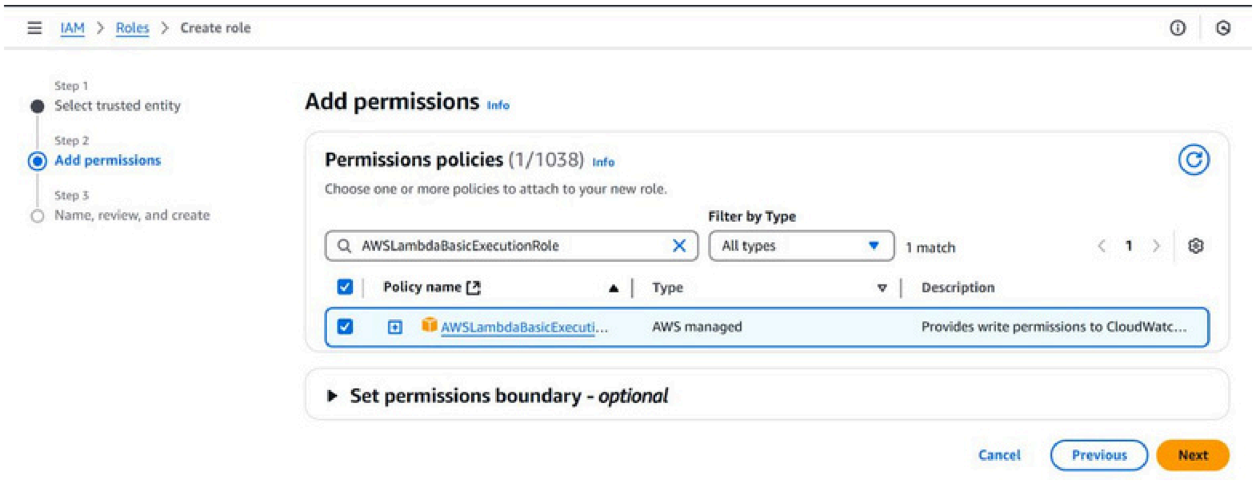
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case: Lambda

Choose a use case for the specified service.

Use case

[✓ Lambda] Allows Lambda functions to call AWS services on your behalf.


IAM > Roles > Create role

Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Add permissions Info

Permissions policies (1/1038) Info

Choose one or more policies to attach to your new role.

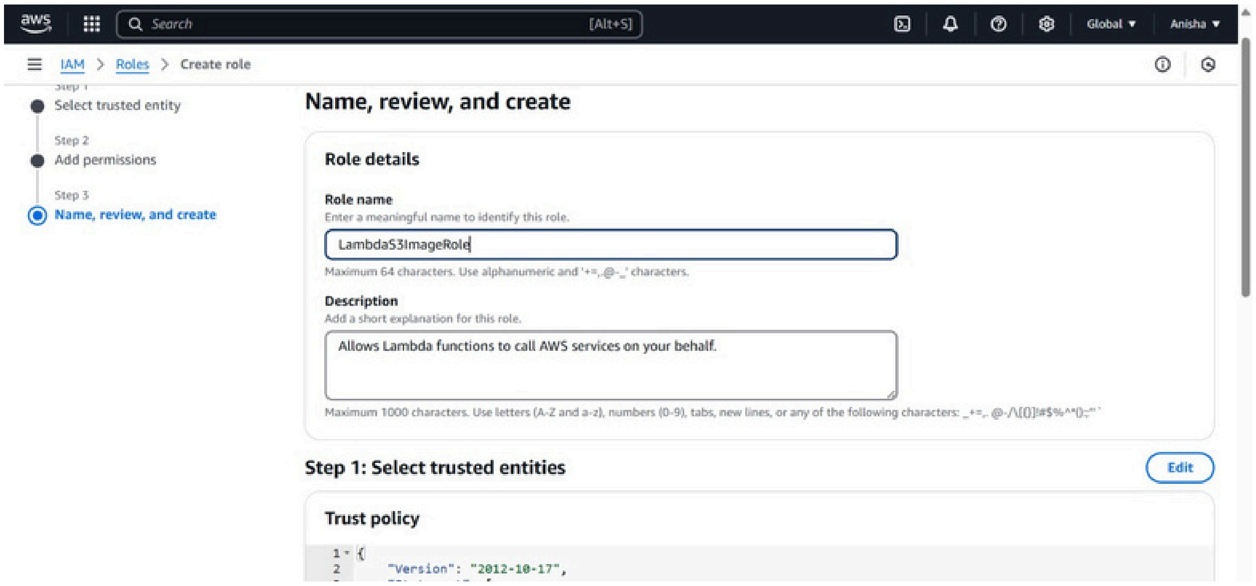
Filter by Type

AWSLambdaBasicExecutionRole | All types | 1 match

	Policy name	Type	Description
✓	AWSLambdaBasicExecuti...	AWS managed	Provides write permissions to CloudWatc...

▶ Set permissions boundary - *optional*

Cancel | Previous | Next



IAM > Roles > Create role

Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Name, review, and create

Role details

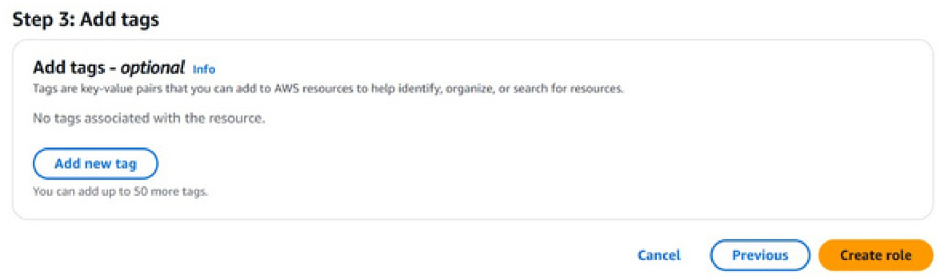
Role name
Enter a meaningful name to identify this role.
LambdaS3ImageRole
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.

Description
Add a short explanation for this role.
Allows Lambda functions to call AWS services on your behalf.
Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: _+=,.@-/\[{()}]!#$%^*();"'`

Step 1: Select trusted entities Edit

Trust policy

```
{
    "Version": "2012-10-17",
```


Step 3: Add tags

Add tags - *optional* Info

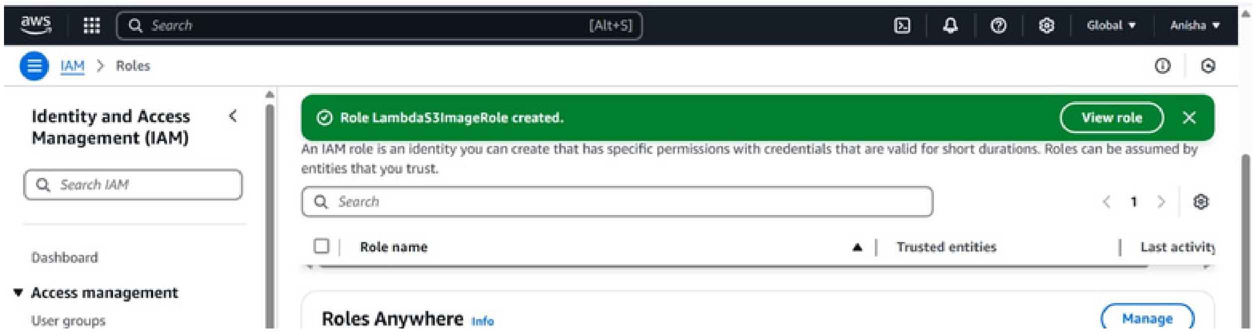
Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

Cancel | Previous | Create role



IAM > Roles

Identity and Access Management (IAM)

Search IAM

Dashboard

▼ Access management

User groups

Role LambdaS3ImageRole created. View role

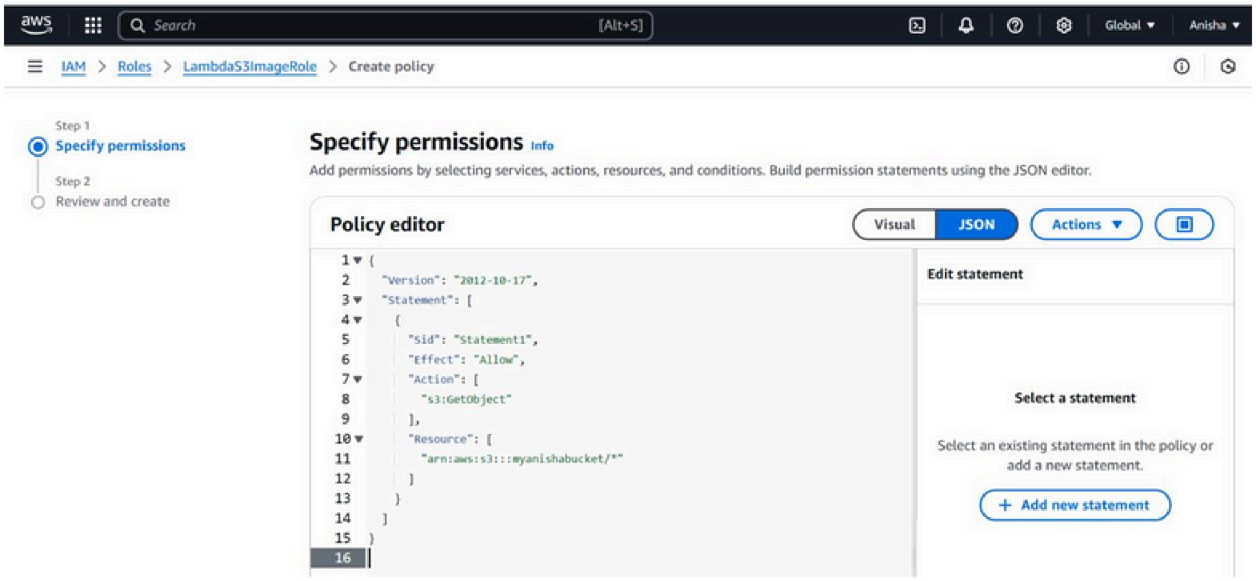
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Search

	Role name	Trusted entities	Last activity

Roles Anywhere Info | Manage

Add Inline Policy to the Role

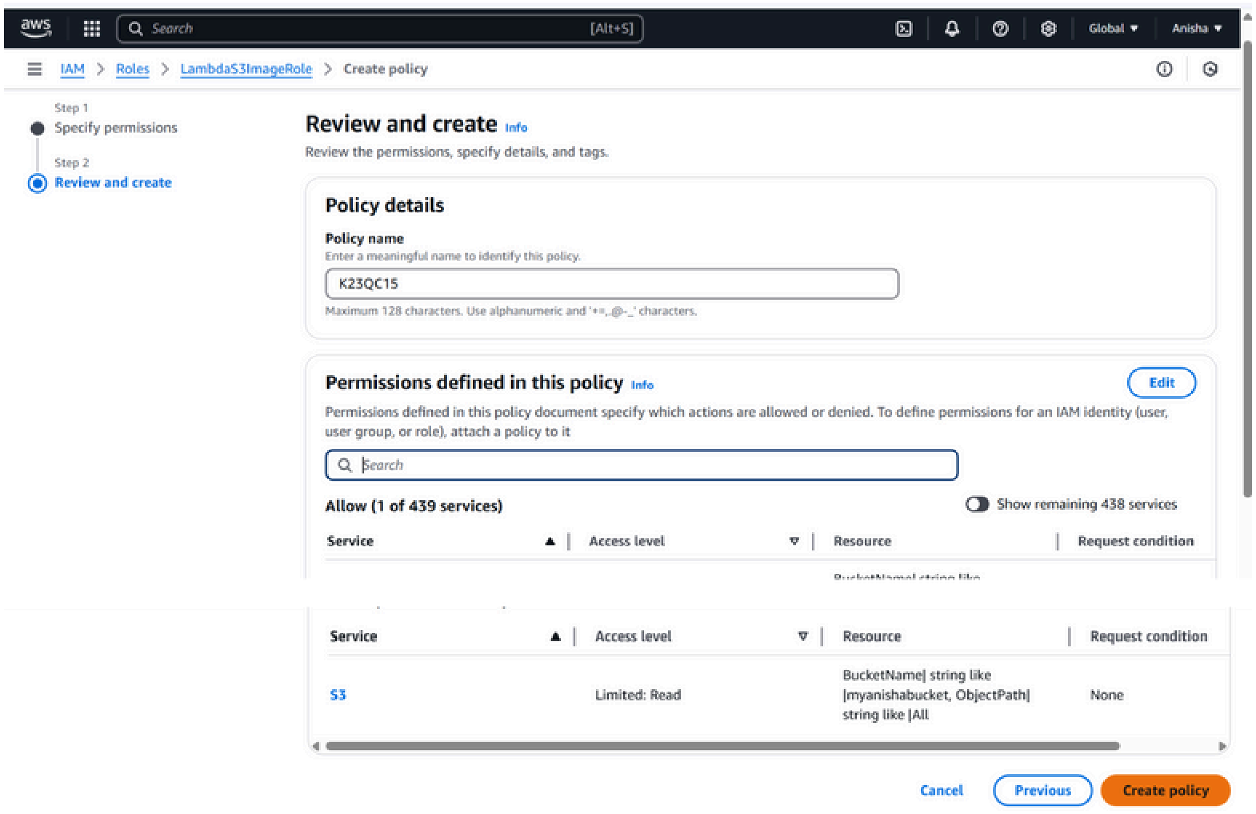


Specify permissions

Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Statement1",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3:::myanishabucket/*"
      ]
    }
  ]
}
```

Edit statement
Select a statement
Select an existing statement in the policy or add a new statement.
+ Add new statement



Review and create

Review the permissions, specify details, and tags.

Policy details

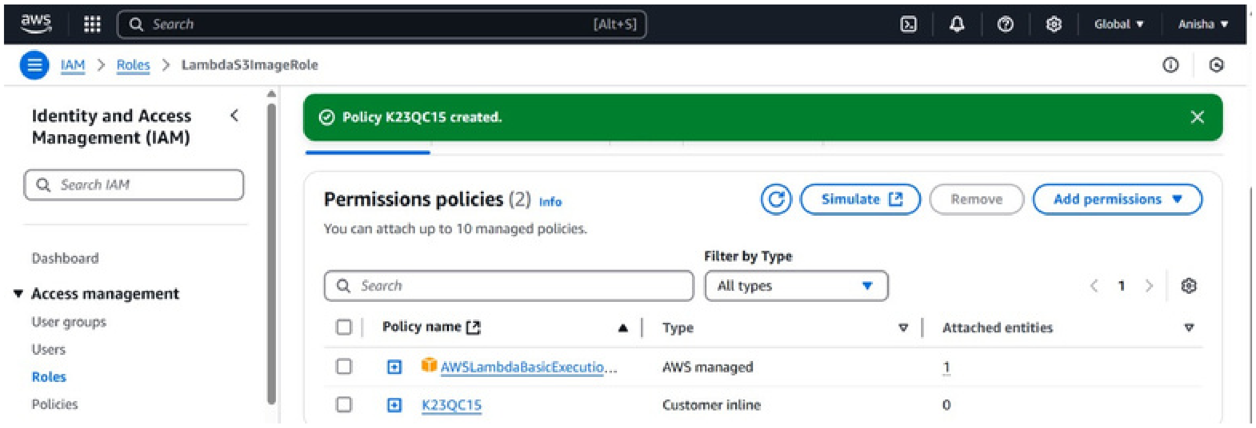
Policy name
Enter a meaningful name to identify this policy.
K23QC15
Maximum 128 characters. Use alphanumeric and '+=,.@-_' characters.

Permissions defined in this policy

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

Allow (1 of 439 services)

Service	Access level	Resource	Request condition
S3	Limited: Read	BucketName| string like |myanishabucket, ObjectPath| string like |All	None


IAM > Roles > LambdaS3ImageRole

Policy K23QC15 created.

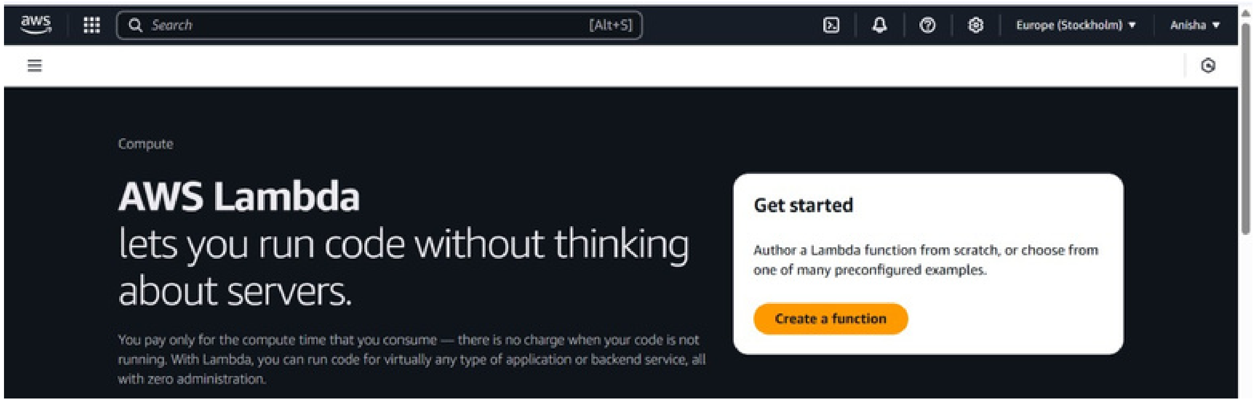
Permissions policies (2) Info

You can attach up to 10 managed policies.

Policy name	Type	Attached entities
AWSLambdaBasicExecutio...	AWS managed	1
K23QC15	Customer inline	0

3. Create Lambda Function

- Goto **AWS Lambda Console** andcreate a new function.
- Choose runtime (e.g., **Python** or **Node.js**).
- Addor upload **image resizing code** (use libraries like Pillow or Sharp).
- Choose the role as existing role and then add the **lambda-img-resize role**.
- Configure **memory and timeout** for performance.

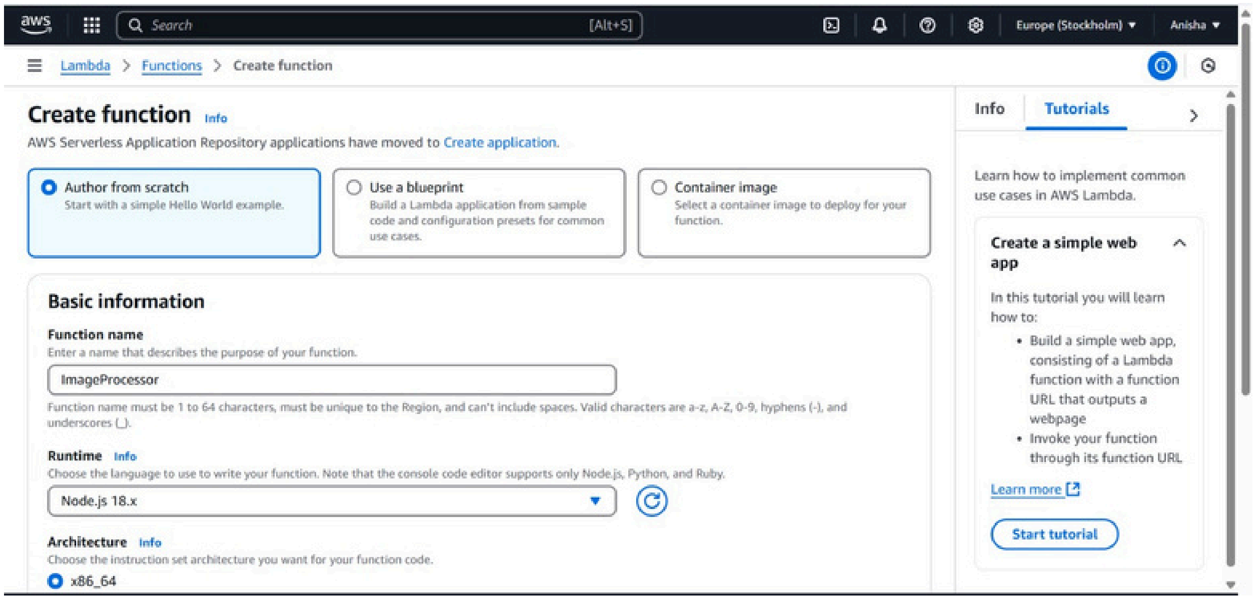

Compute

AWS Lambda lets you run code without thinking about servers.

You pay only for the compute time that you consume — there is no charge when your code is not running. With Lambda, you can run code for virtually any type of application or backend service, all with zero administration.

Get started

Author a Lambda function from scratch, or choose from one of many preconfigured examples.


Lambda > Functions > Create function

Create function Info

AWS Serverless Application Repository applications have moved to Create application.

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

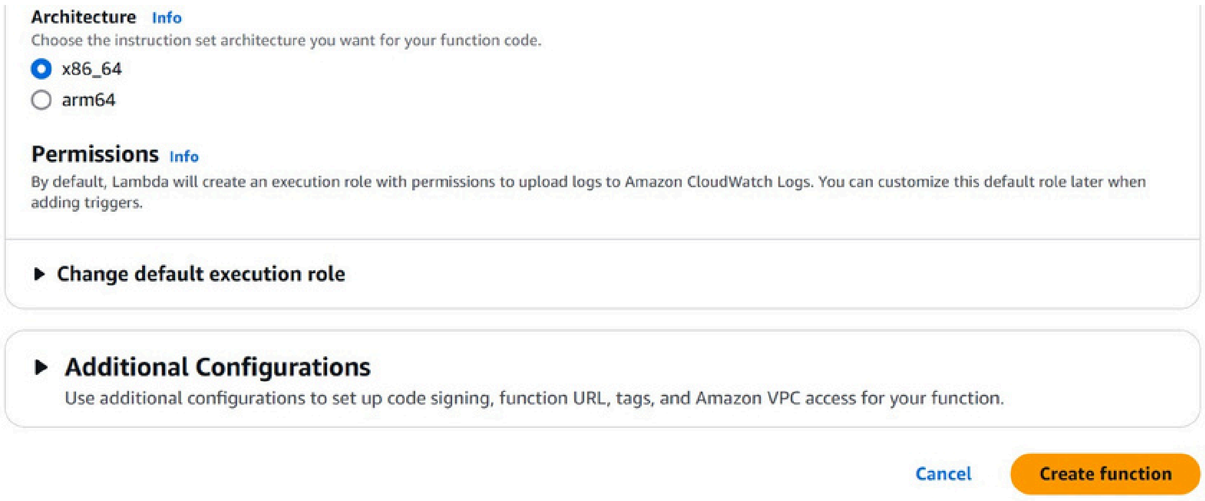
Function name: ImageProcessor

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: Node.js 18.x

Architecture: x86_64 [✓]

Create a simple web app — In this tutorial you will learn how to: Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage; Invoke your function through its function URL.



Architecture Info

Choose the instruction set architecture you want for your function code.

[✓ x86_64]
arm64

Permissions Info

By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

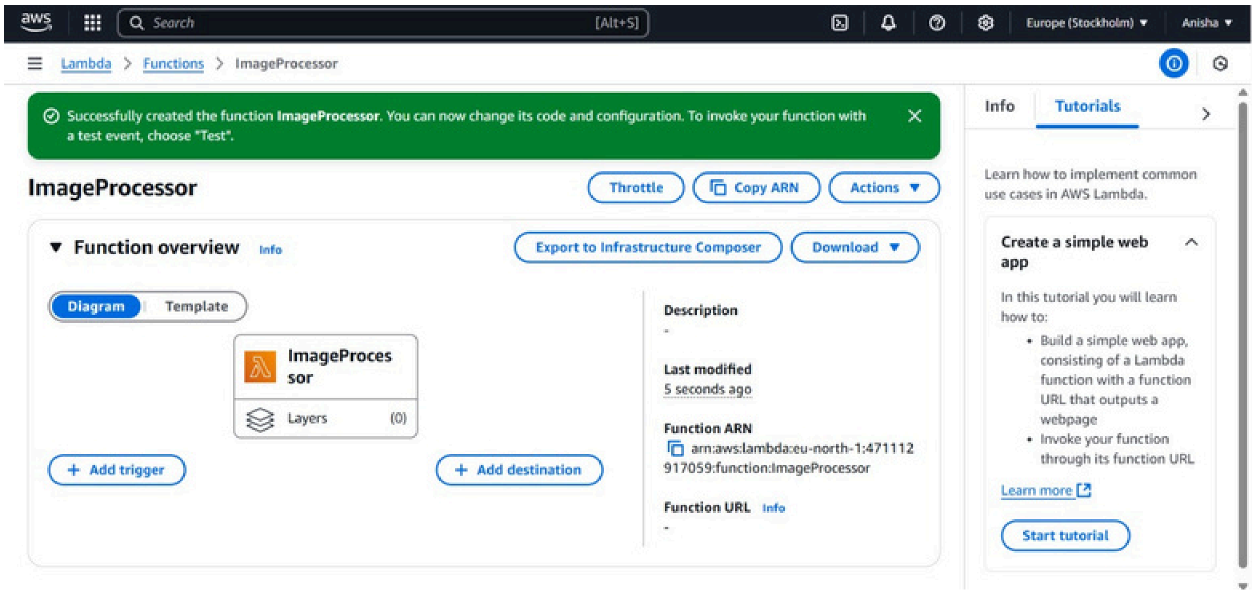
▶ Change default execution role

▶ Additional Configurations

Use additional configurations to set up code signing, function URL, tags, and Amazon VPC access for your function.

4.Connect S3 to Lambda

- InLambda, **add trigger** fromS3bucket(original-images bucket).
- Give Lambda **permission to read S3 bucket**.
- Save and test the trigger setup.



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

ImageProcessor

Description
-

Last modified
5 seconds ago

Function ARN
arn:aws:lambda:eu-north-1:471112917059:function:ImageProcessor

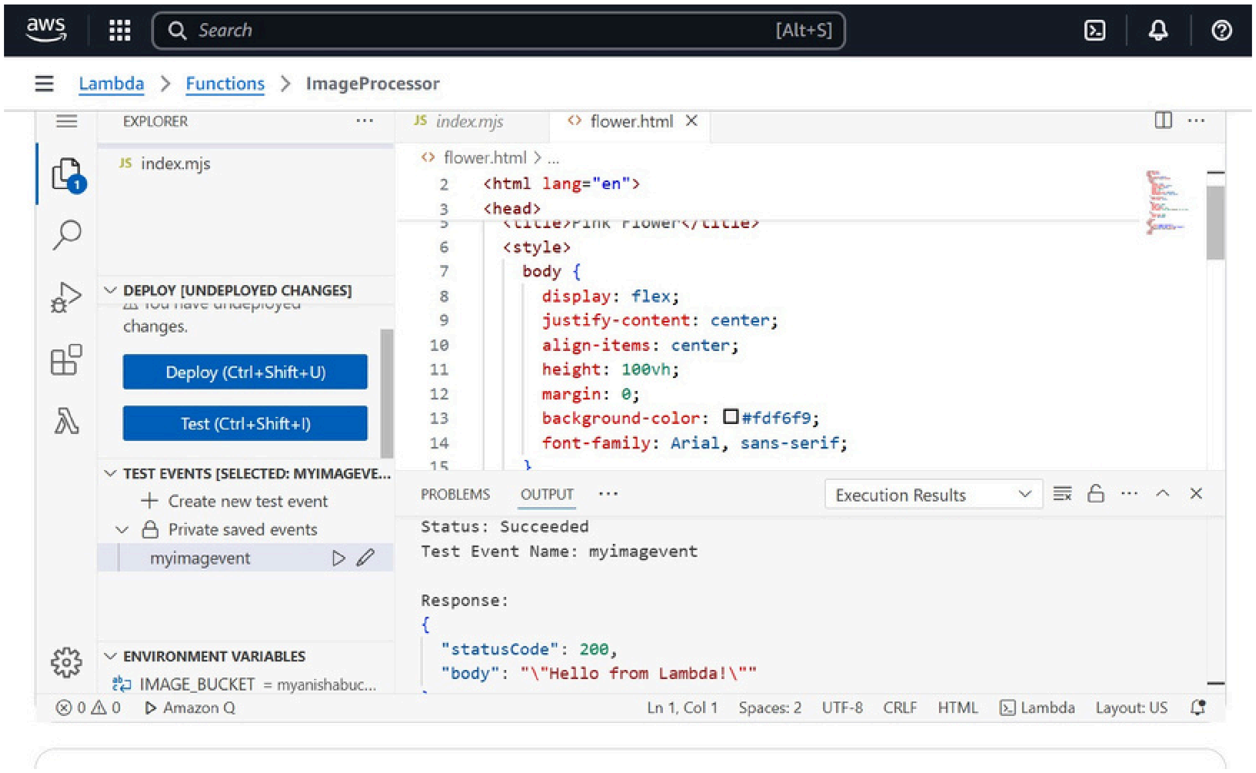
Function URL Info
-

Learn how to implement common use cases in AWS Lambda.

Create a simple web app

In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

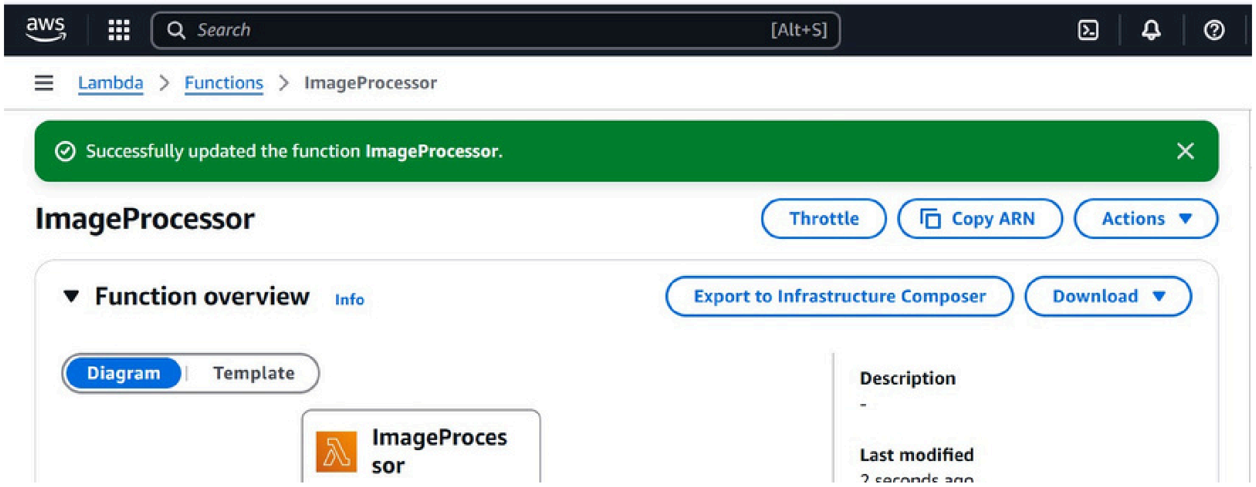


aws
Search [Alt+S]
Lambda > Functions > ImageProcessor
EXPLORER
index.mjs
DEPLOY [UNDEPLOYED CHANGES]
You have undeployed changes.
Deploy (Ctrl+Shift+U)
Test (Ctrl+Shift+I)
TEST EVENTS [SELECTED: MYIMAGEVE...
Create new test event
Private saved events
myimagevent
ENVIRONMENT VARIABLES
IMAGE_BUCKET = myanishabuc...
index.mjs
flower.html
PROBLEMS
OUTPUT
Execution Results
Ln 1, Col 1 Spaces: 2 UTF-8 CRLF HTML Lambda Layout: US
Amazon Q

```
<html lang="en">
<head>
  <title>Pink Flower</title>
  <style>
    body {
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
      margin: 0;
      background-color: #fdf6f9;
      font-family: Arial, sans-serif;
    }
```

```
Status: Succeeded
Test Event Name: myimagevent

Response:
{
  "statusCode": 200,
  "body": "\"Hello from Lambda!\""
```

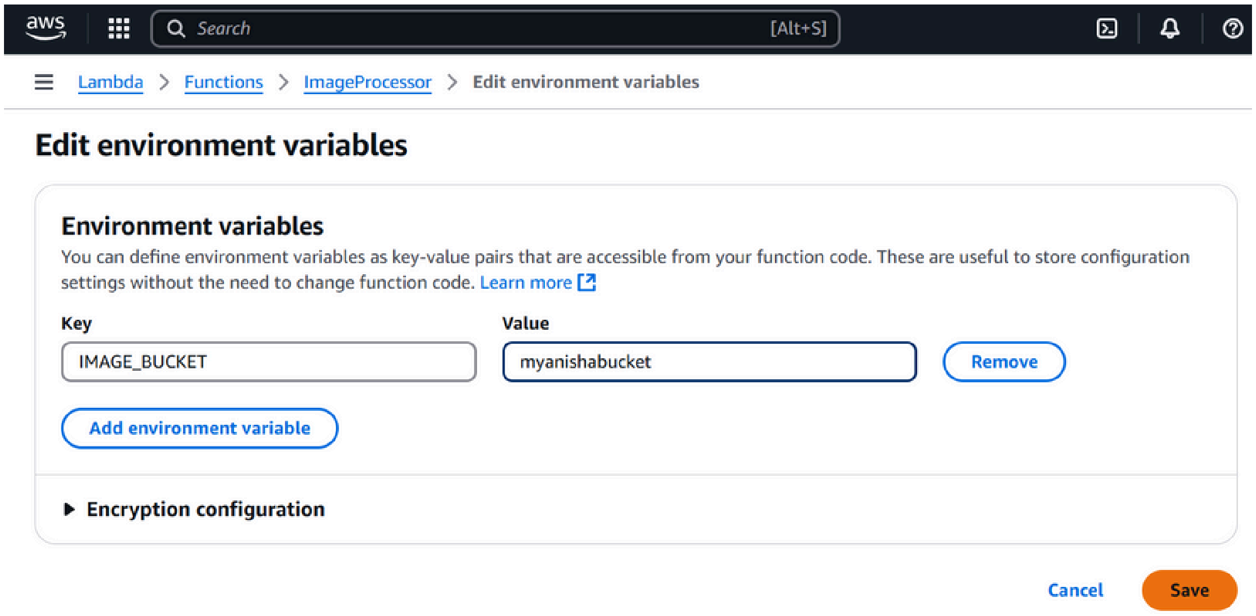


Successfully updated the function **ImageProcessor**.

ImageProcessor

Description
-

Last modified
2 seconds ago

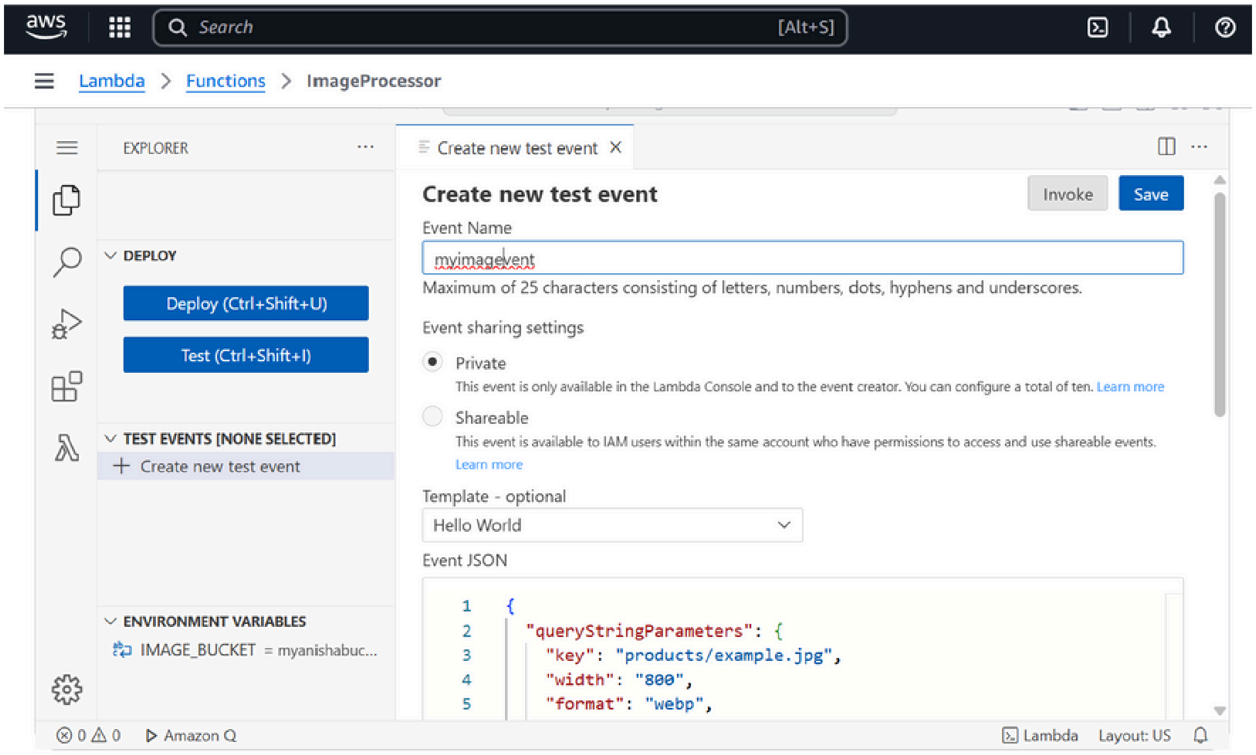

aws
Search [Alt+S]
Lambda > Functions > ImageProcessor > Edit environment variables

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	Value
IMAGE_BUCKET	myanishabucket



Create new test event

Event Name: myimagevent

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 ten. Learn more

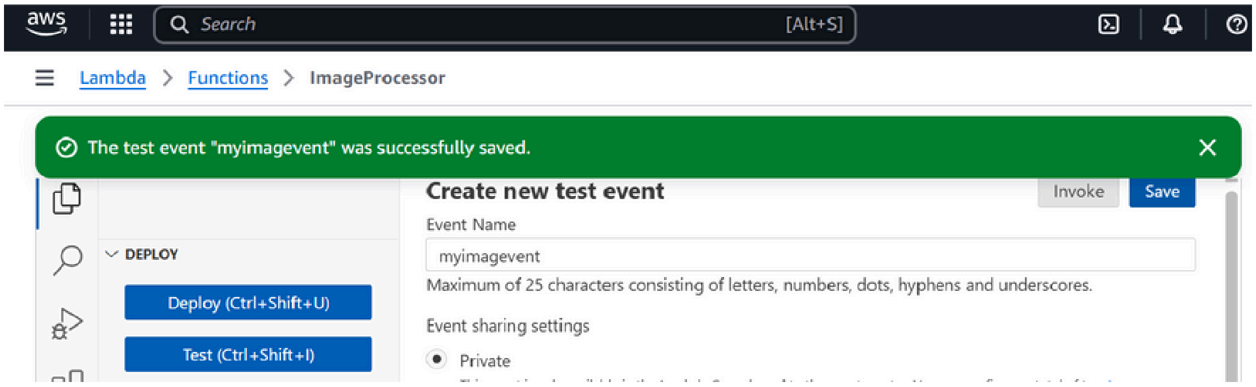
○ Shareable

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

Template - optional: Hello World

Event JSON

```
{
  "queryStringParameters": {
    "key": "products/example.jpg",
    "width": "800",
    "format": "webp",
```



The test event "myimagevent" was successfully saved.

Create new test event

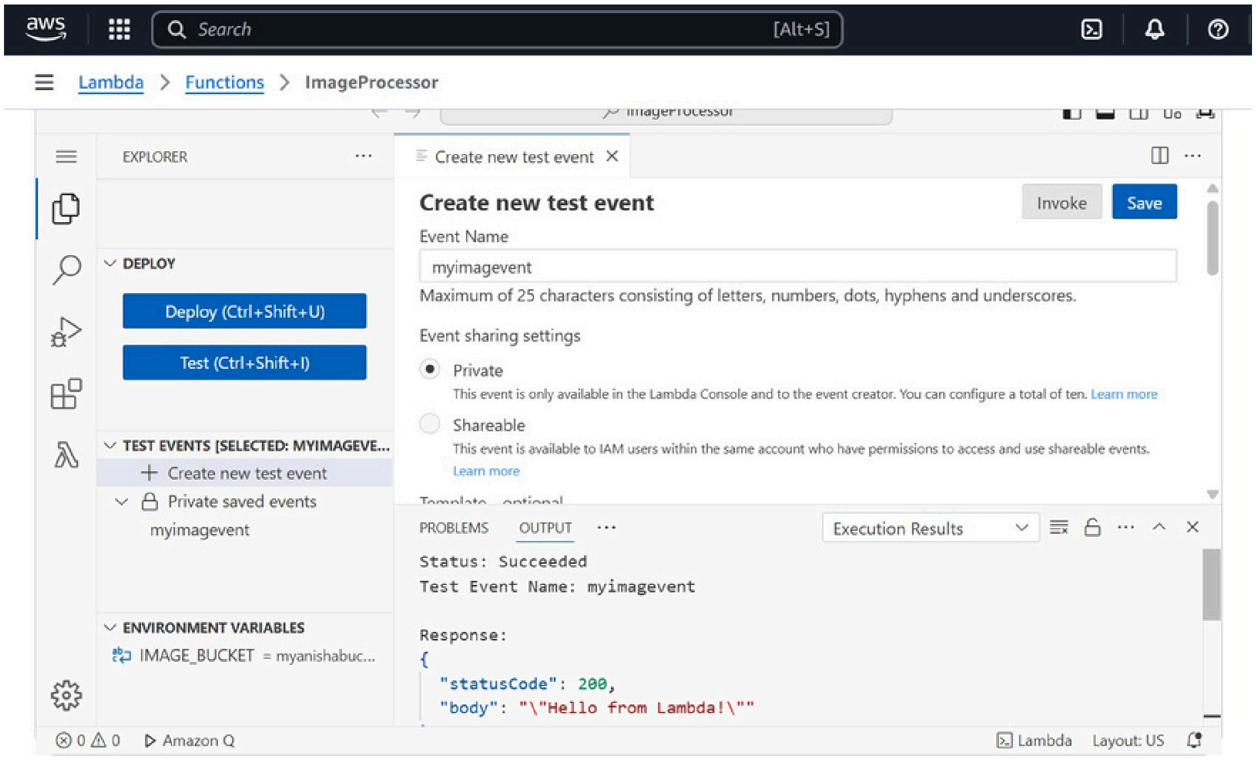
Event Name: myimagevent

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

Event sharing settings

● Private

Result:



Create new test event

Event Name: myimagevent

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 ten. Learn more

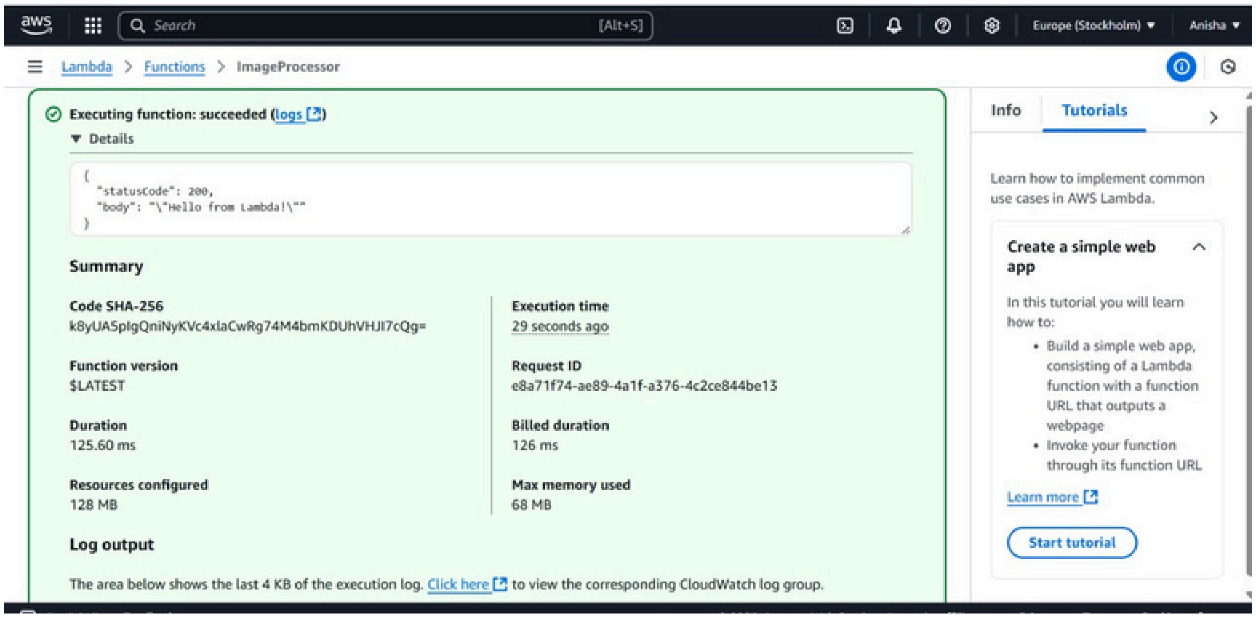
Shareable — This event is available to IAM users within the same account who have permissions to access and use shareable events. Learn more

IMAGE_BUCKET = myanishabuc...

```
Status: Succeeded
Test Event Name: myimagevent

Response:
{
  "statusCode": 200,
  "body": "\"Hello from Lambda!\""
```

5. Test An Event



Executing function: succeeded (logs)

Details

```
{
  "statusCode": 200,
  "body": "\"Hello from Lambda!\""
}
```

Summary

Code SHA-256: k8yUA5plgQniNyKVc4xlaCwRg74M4bmKDUhVHJI7cQg=	Execution time: 29 seconds ago
Function version: $LATEST	Request ID: e8a71f74-ae89-4a1f-a376-4c2ce844be13
Duration: 125.60 ms	Billed duration: 126 ms
Resources configured: 128 MB	Max memory used: 68 MB

Log output

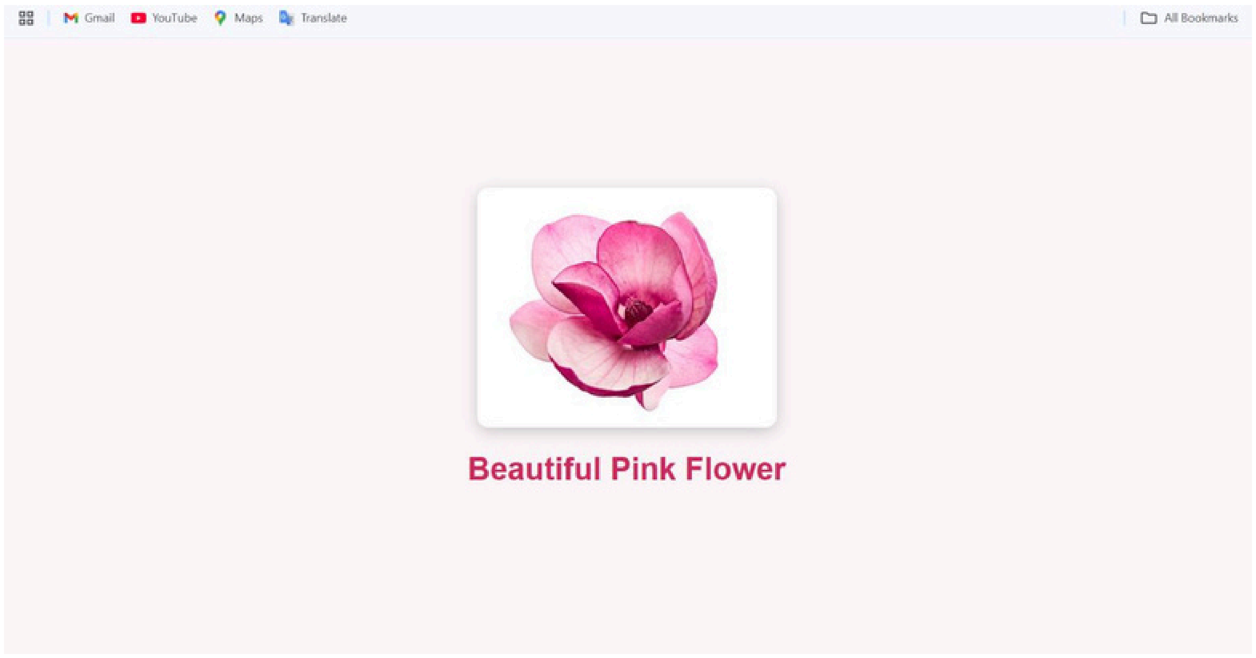
The area below shows the last 4 KB of the execution log. Click here to view the corresponding CloudWatch log group.

Learn how to implement common use cases in AWS Lambda.

In this tutorial you will learn how to:
- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

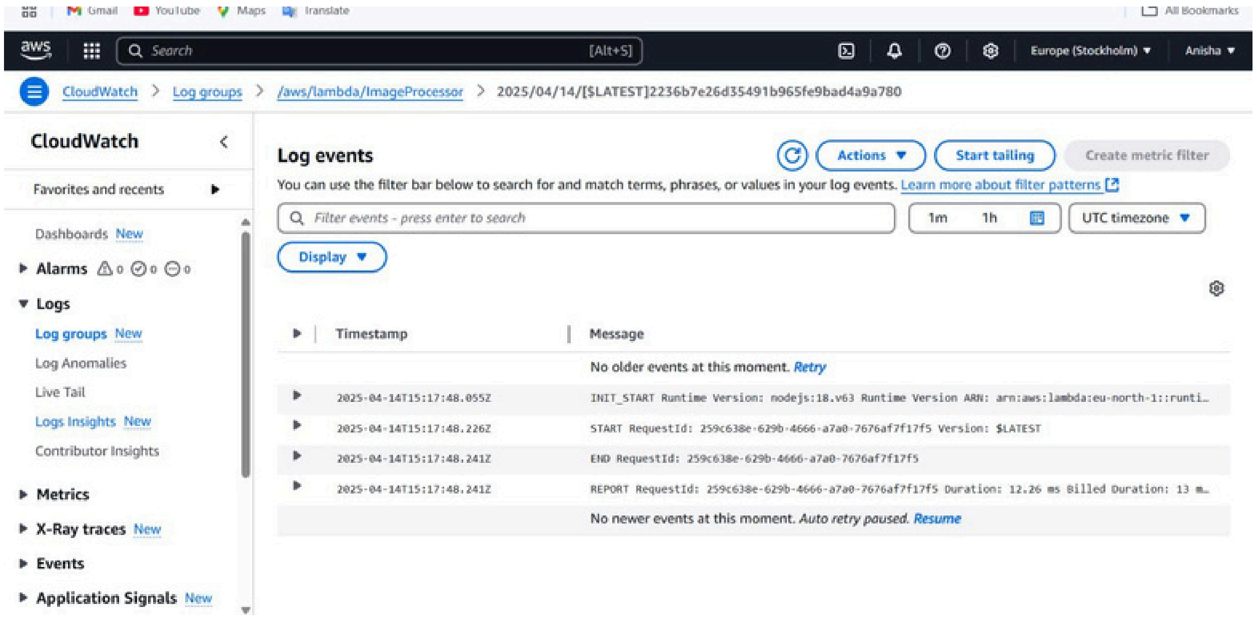
6. Test Upload

- Now check the destination bucket for the image.
- The image will be stored in resized form.
- Test with different image formats (JPG, PNG).



6. Monitor with CloudWatch

- Go to **CloudWatch Console** .
- Check **Lambda logs** forsuccessfulexecutions.
- Optionally, create **custom dashboards** for monitoring.
- Set up **CloudWatch Alarms** forerrorsor performance issues.

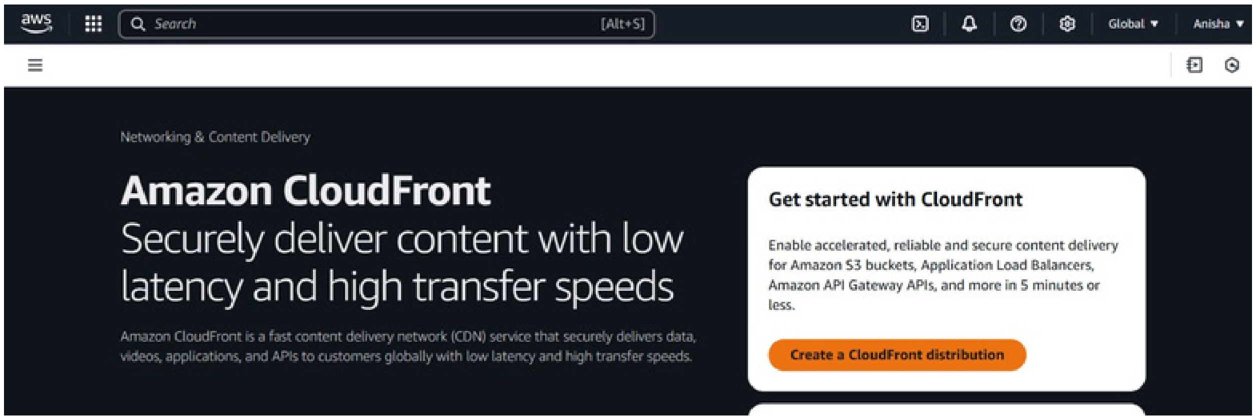


Log events

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

```
Timestamp                  Message
                           No older events at this moment. Retry
2025-04-14T15:17:48.055Z   INIT_START Runtime Version: nodejs:18.v63 Runtime Version ARN: arn:aws:lambda:eu-north-1::runti…
2025-04-14T15:17:48.226Z   START RequestId: 259c638e-629b-4666-a7a0-7676af7f17f5 Version: $LATEST
2025-04-14T15:17:48.241Z   END RequestId: 259c638e-629b-4666-a7a0-7676af7f17f5
2025-04-14T15:17:48.241Z   REPORT RequestId: 259c638e-629b-4666-a7a0-7676af7f17f5 Duration: 12.26 ms Billed Duration: 13 m…
                           No newer events at this moment. Auto retry paused. Resume
```

<u>**Create CloudFront Distribution**</u>

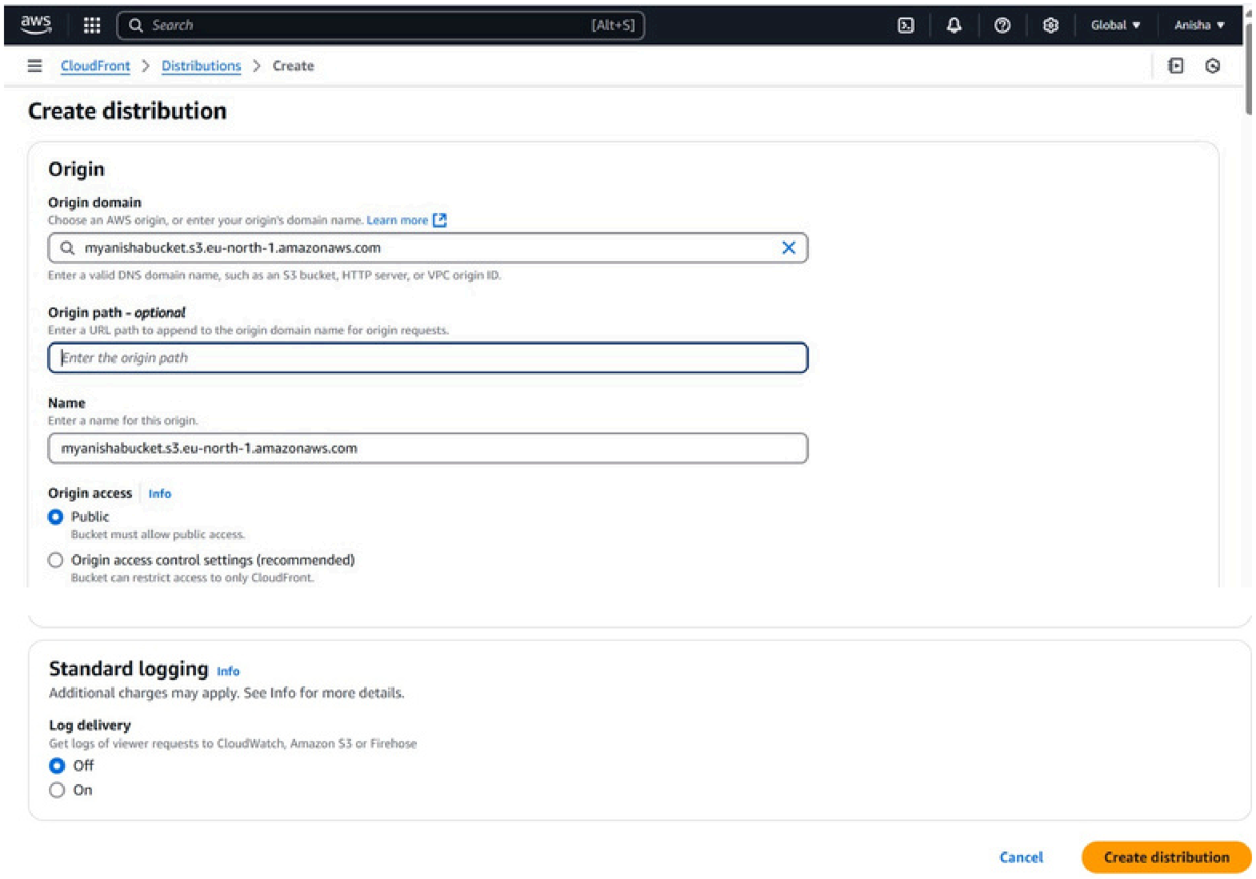

Networking & Content Delivery

Amazon CloudFront
Securely deliver content with low latency and high transfer speeds

Amazon CloudFront is a fast content delivery network (CDN) service that securely delivers data, videos, applications, and APIs to customers globally with low latency and high transfer speeds.

Get started with CloudFront

Enable accelerated, reliable and secure content delivery for Amazon S3 buckets, Application Load Balancers, Amazon API Gateway APIs, and more in 5 minutes or less.


CloudFront > Distributions > Create

Create distribution

Origin

Origin domain
Choose an AWS origin, or enter your origin's domain name. Learn more
myanishabucket.s3.eu-north-1.amazonaws.com
Enter a valid DNS domain name, such as an S3 bucket, HTTP server, or VPC origin ID.

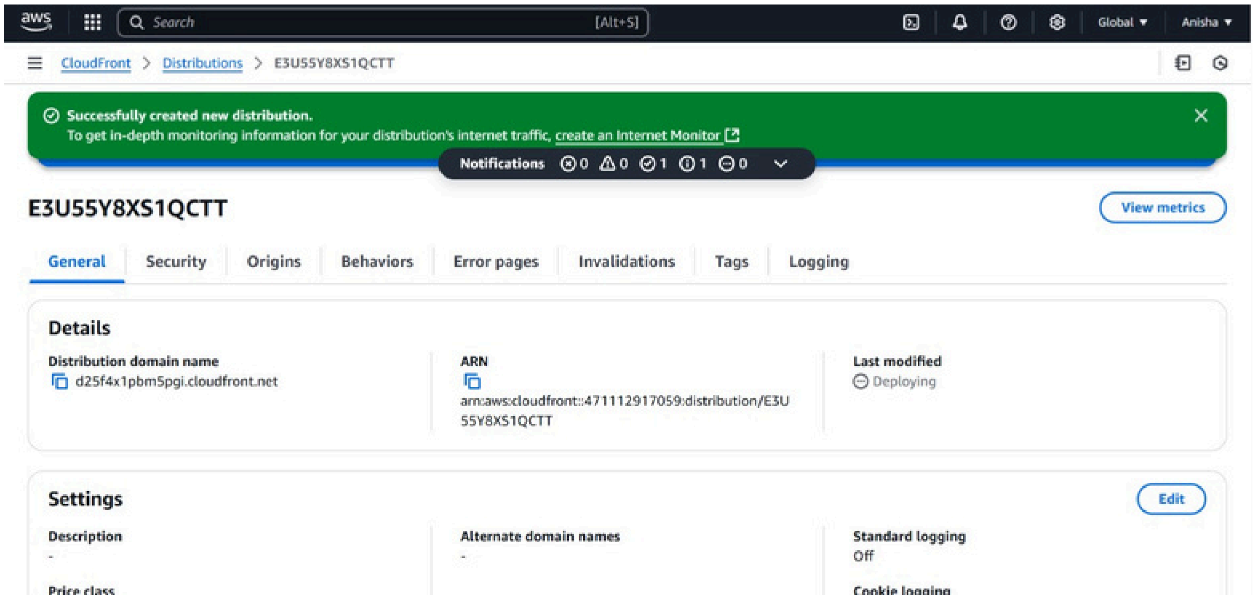
Origin path - *optional*
Enter a URL path to append to the origin domain name for origin requests.
Enter the origin path

Name
Enter a name for this origin.
myanishabucket.s3.eu-north-1.amazonaws.com

Origin access Info
[✓ Public]
Bucket must allow public access.
Origin access control settings (recommended)
Bucket can restrict access to only CloudFront.

Standard logging Info
Additional charges may apply. See Info for more details.

Log delivery
Get logs of viewer requests to CloudWatch, Amazon S3 or Firehose
[✓ Off]
On


aws
Search [Alt+S]
Global
Anisha
CloudFront > Distributions > E3U55Y8XS1QCTT

Successfully created new distribution.
To get in-depth monitoring information for your distribution's internet traffic, create an Internet Monitor

Notifications 0 0 1 1 0
View metrics
General Security Origins Behaviors Error pages Invalidations Tags Logging
Edit

E3U55Y8XS1QCTT

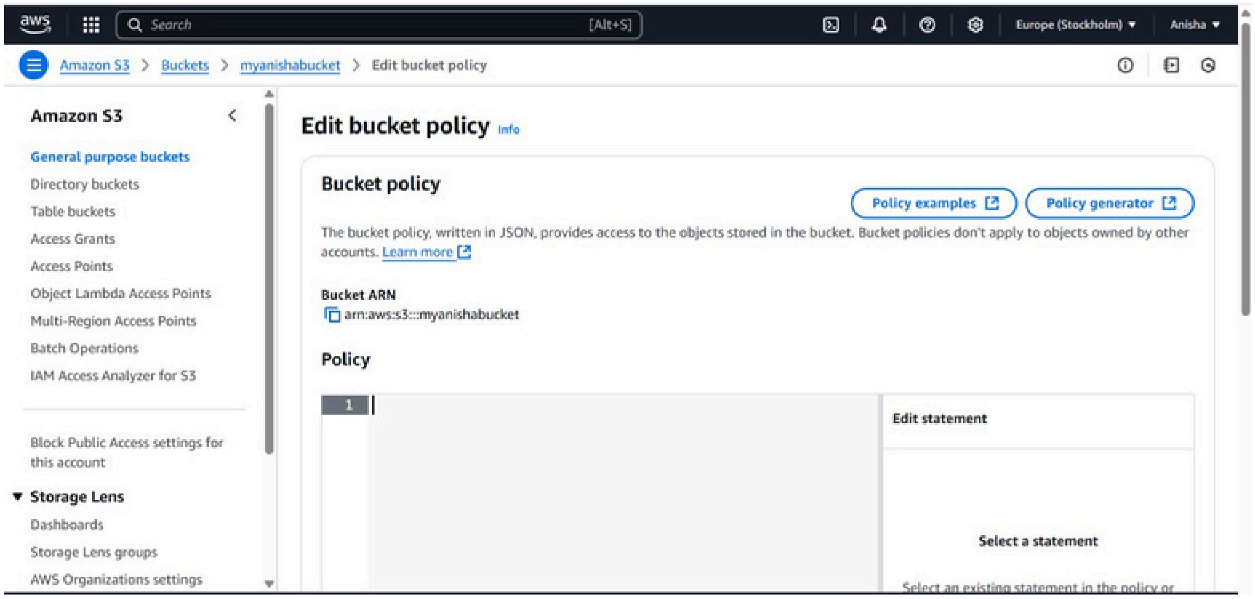
Details

Distribution domain name	ARN	Last modified
d25f4x1pbm5pgi.cloudfront.net	arn:aws:cloudfront::471112917059:distribution/E3U55Y8XS1QCTT	Deploying

Settings

Description	Alternate domain names	Standard logging
-	-	Off
Price class		Cookie logging

Lock Down the S3 Bucket



Edit bucket policy

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

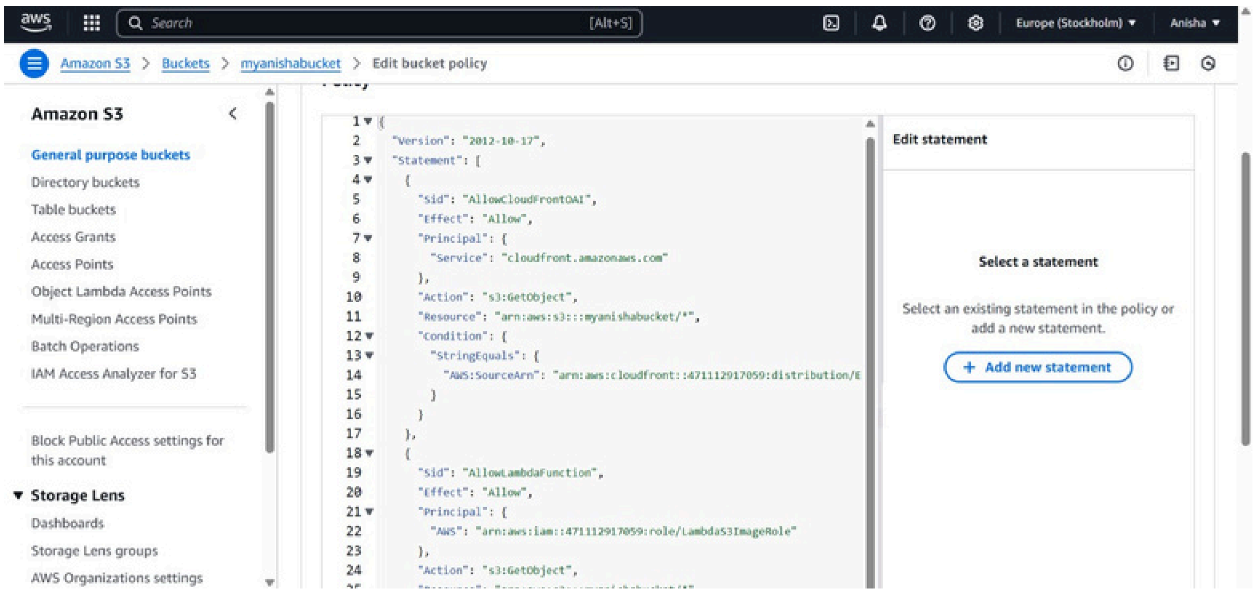
Bucket ARN
arn:aws:s3:::myanishabucket

Policy

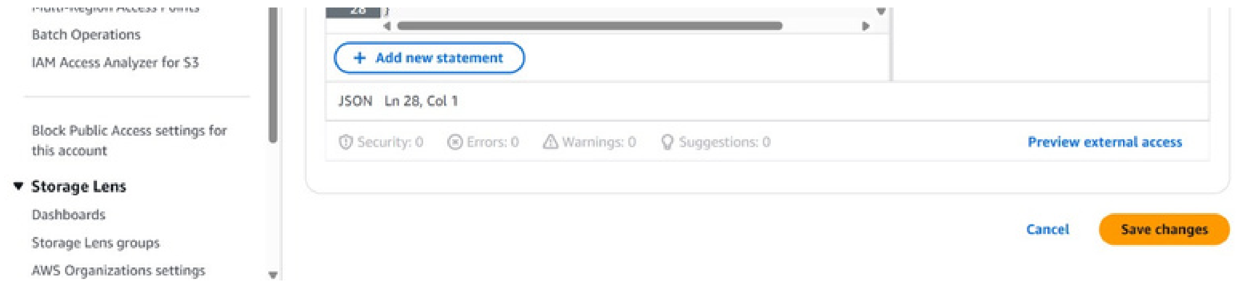
Edit statement

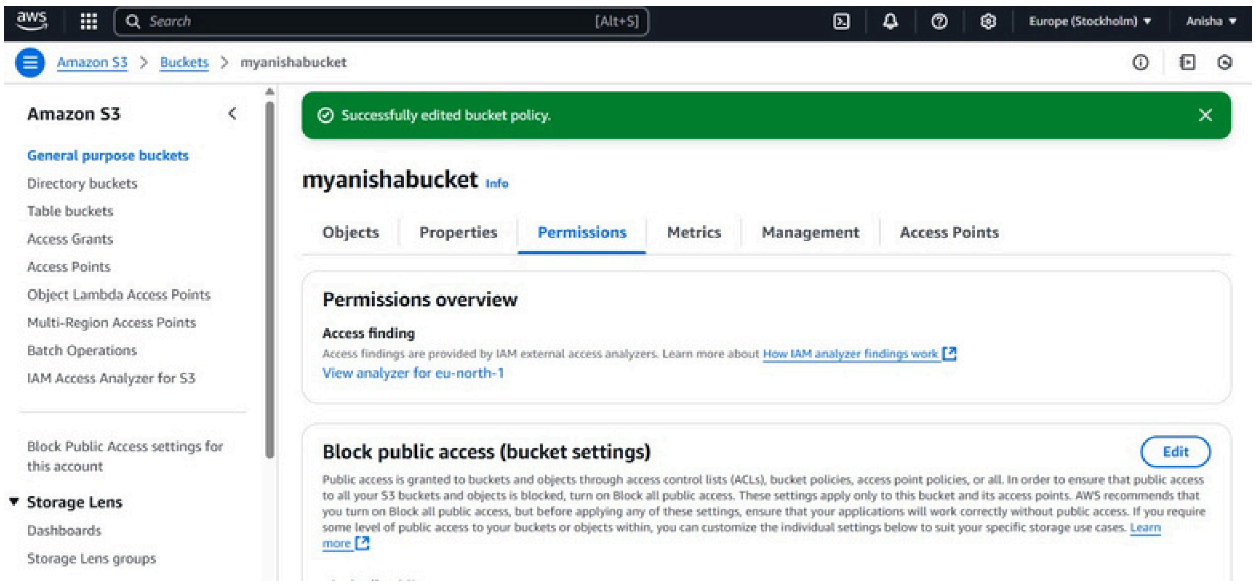
Select a statement

Select an existing statement in the policy or


aws
Search [Alt+S]
Europe (Stockholm) ▾
Anisha ▾
Amazon S3 > Buckets > myanishabucket > Edit bucket policy
Amazon S3
General purpose buckets
Directory buckets
Table buckets
Access Grants
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
▼ Storage Lens
Dashboards
Storage Lens groups
AWS Organizations settings
Edit statement
Select a statement
Select an existing statement in the policy or add a new statement.
+ Add new statement

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCloudFrontOAI",
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudfront.amazonaws.com"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::myanishabucket/*",
      "Condition": {
        "StringEquals": {
          "AWS:SourceArn": "arn:aws:cloudfront::471112917059:distribution/E
        }
      }
    },
    {
      "Sid": "AllowLambdaFunction",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::471112917059:role/LambdaS3ImageRole"
      },
      "Action": "s3:GetObject",
```





Successfully edited bucket policy.

myanishabucket Info

Permissions overview

Access finding

Access findings are provided by IAM external access analyzers. Learn more about How IAM analyzer findings work

View analyzer for eu-north-1

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

END