AWS Session 7
Summary - 25-02-2023

- If we want to run the code, manage the data, and integrate the applications then we need a server and we have to manage it. But if we do all these tasks without managing the server manually then this type of concept is known as **serverless**.

- Serverless technologies feature automatic scaling, built-in high availability, and a pay-for-use billing model to increase agility and optimize costs.

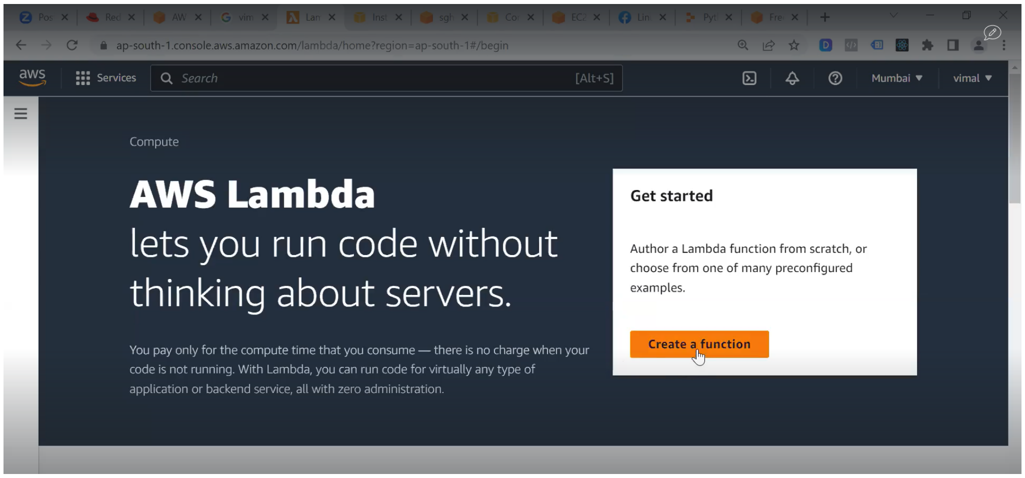
- A serverless architecture is a way to build and run applications and services without having to manage infrastructure. Your application still runs on servers, but all the server management is done by AWS. You no longer have to provision, scale, and maintain servers to run your applications, databases, and storage systems.

- AWS Lambda is a serverless, event-driven compute service that lets you run code without provisioning or managing servers.

- If we want to run the code then we can run using either manual way ie. by clicking the “Test” button or by using AWS service events.

- **Create a Lambda function** :

1) Click on “Create a function”


ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/begin
aws
Services
Search
[Alt+S]
Mumbai ▼
vimal ▼

Compute

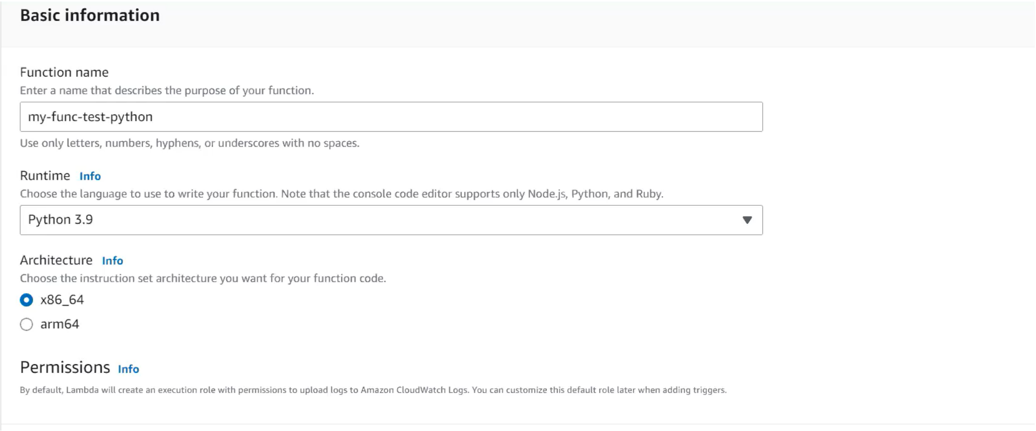
AWS Lambda
lets you run code without thinking about servers.

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.

2) Give a name of function and choose Runtime then click on create function :



Basic information

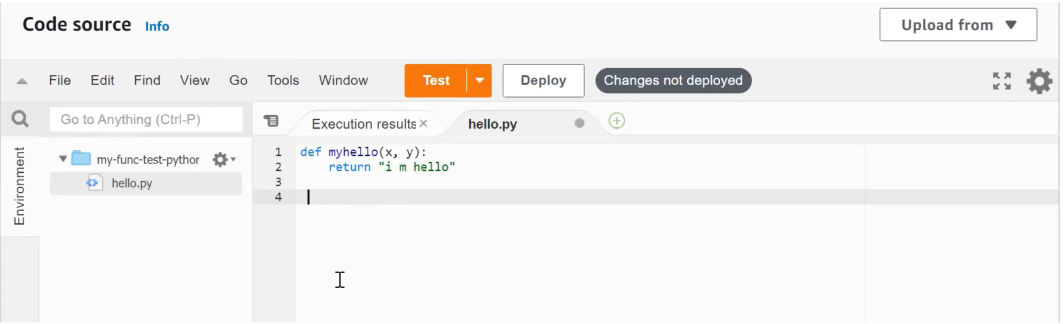
Function name
Enter a name that describes the purpose of your function.
my-func-test-python
Use only letters, numbers, hyphens, or underscores with no spaces.

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 3.9 ▼

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.

3) Write code then click on Test :

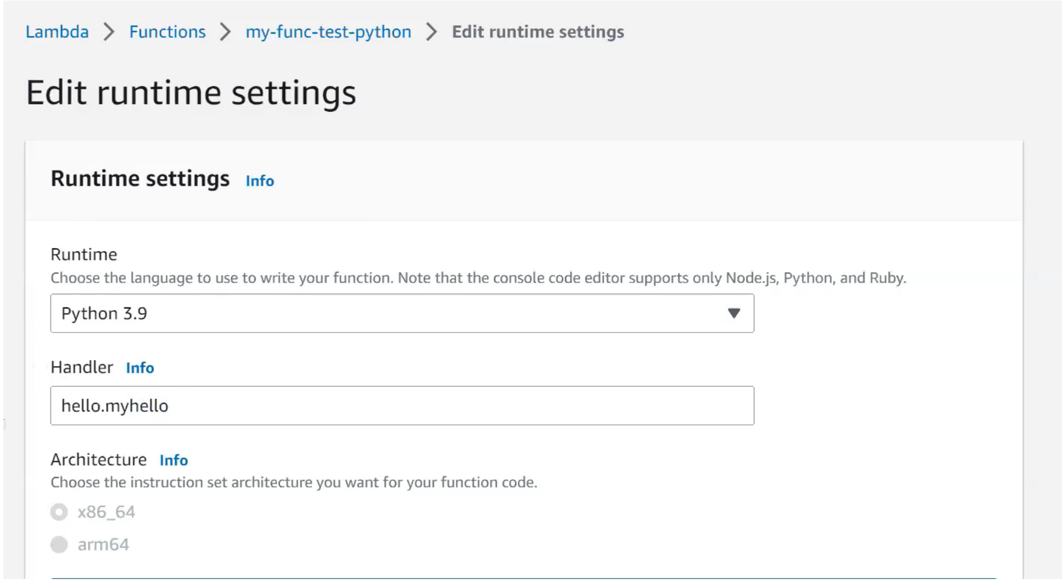


```
def myhello(x, y):
    return "i m hello"

```

4) After clicking on the test they ask about the event name just type name of eventname.

5) We have to tell which function we want to run through which file using the handler. For this go to **Runtime Setting -> Edit**


Lambda > Functions > my-func-test-python > Edit runtime settings

Edit runtime settings

Runtime settings Info

Runtime

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

Python 3.9

Handler Info

hello.myhello

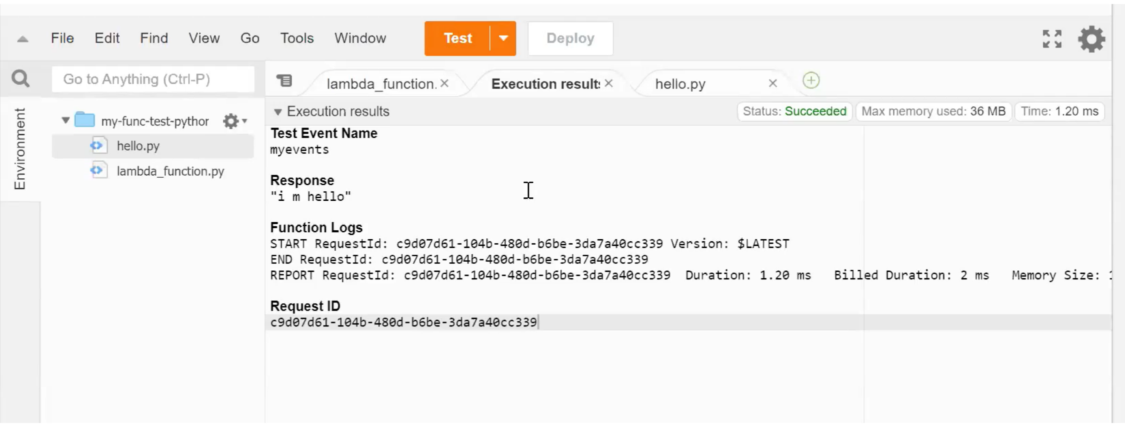
Architecture Info

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

○ x86_64

○ arm64

6) Click on "Deploy" then for testing click on "Test"



Execution results   Status: Succeeded   Max memory used: 36 MB   Time: 1.20 ms

Test Event Name

```
myevents
```

Response

```
"i m hello"
```

Function Logs

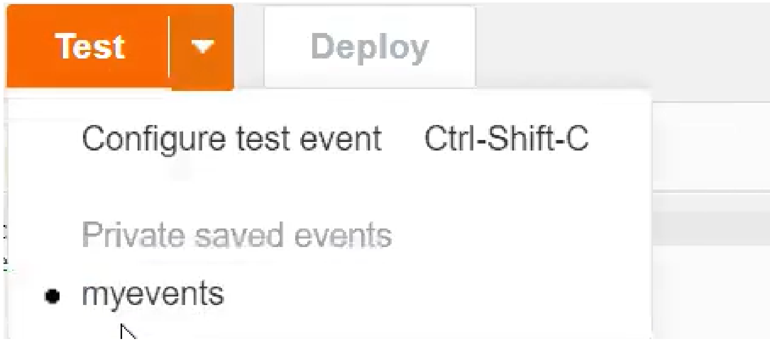
```
START RequestId: c9d07d61-104b-480d-b6be-3da7a40cc339 Version: $LATEST
END RequestId: c9d07d61-104b-480d-b6be-3da7a40cc339
REPORT RequestId: c9d07d61-104b-480d-b6be-3da7a40cc339  Duration: 1.20 ms   Billed Duration: 2 ms   Memory Size: 1
```

Request ID

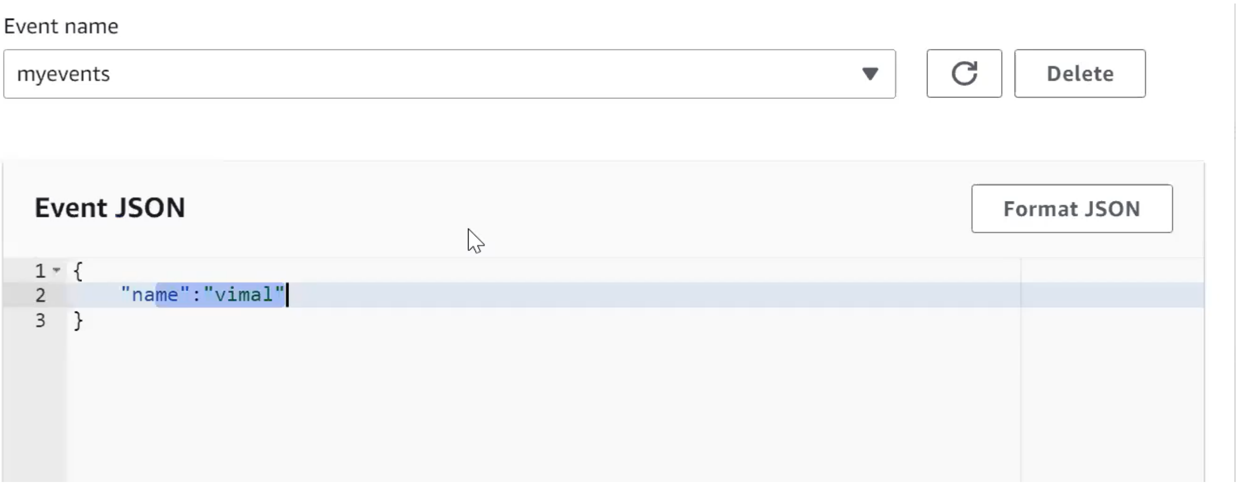
```
c9d07d61-104b-480d-b6be-3da7a40cc339
```

- Every lambda function takes 2 inputs from the user. 1) event 2) context

- We can configure test event manually :



- Give event in JSON format :


Event name

myevents

Event JSON

```
{
    "name":"vimal"
}
```

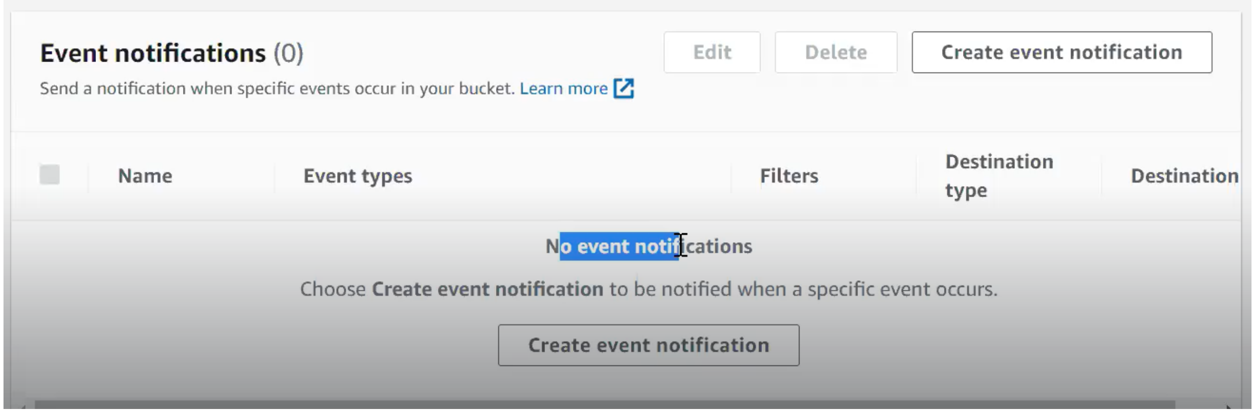
- If we want to monitor something in AWS then they provide **CloudWatch** service. By default CloudWatch service monitors logs, metrics of Lambda service.

- We can trigger the lambda function by occurring some events.

- **Integrate S3 service with Lambda** : As soon as some event occurs in S3 we want the lambda function will run.

We can integrate by two ways: either add the trigger from lambda service or add event notification from S3 service.

1) Create bucket in S3 service and after go inside that bucket, go to properties section then we can create event notification :



Event notifications (0)

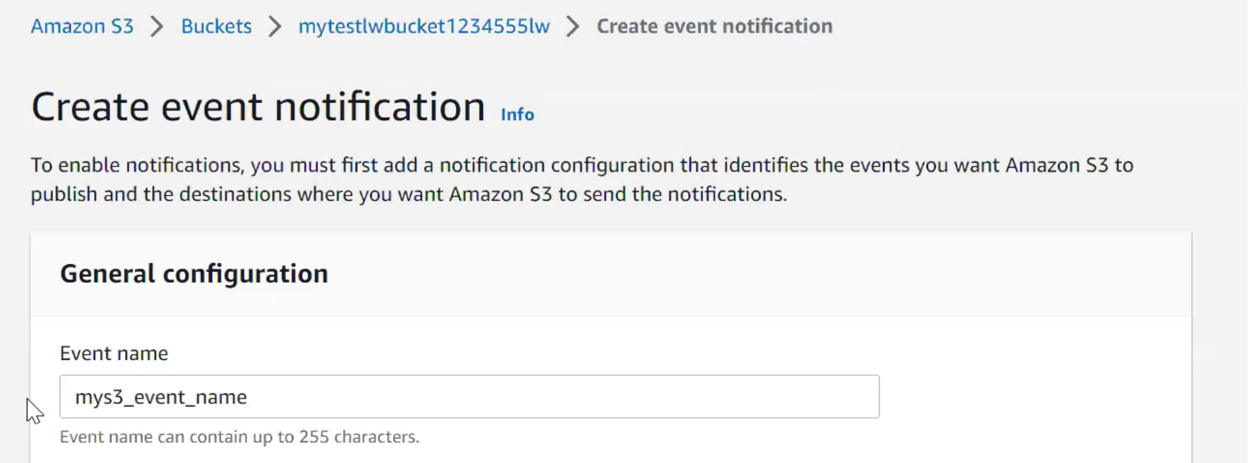
Send a notification when specific events occur in your bucket. Learn more

	Name	Event types	Filters	Destination type	Destination

No event notifications

Choose **Create event notification** to be notified when a specific event occurs.

2) Type event name :



Amazon S3 > Buckets > mytestlwbucket1234555lw > Create event notification

Create event notification Info

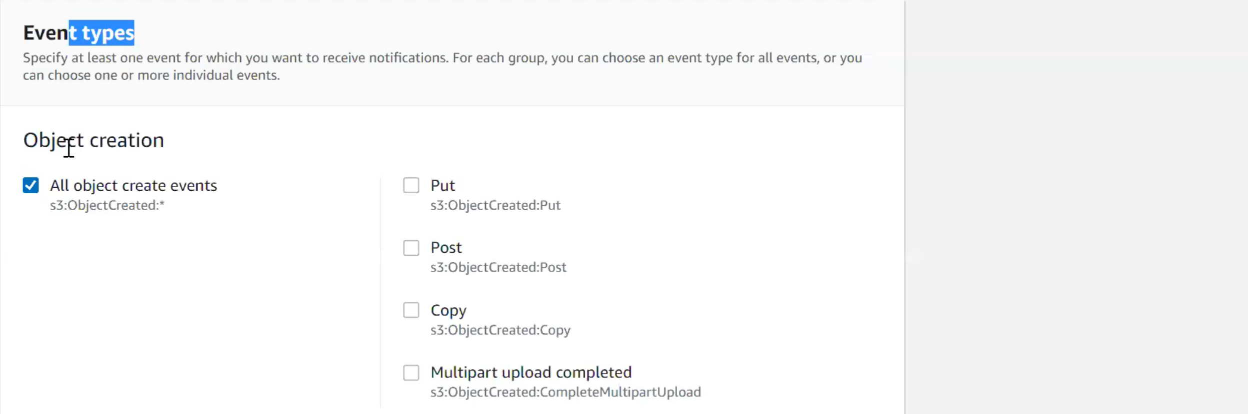
To enable notifications, you must first add a notification configuration that identifies the events you want Amazon S3 to publish and the destinations where you want Amazon S3 to send the notifications.

General configuration

Event name: mys3_event_name

Event name can contain up to 255 characters.

3) Choose Event types :



Event types

Specify at least one event for which you want to receive notifications. For each group, you can choose an event type for all events, or you can choose one or more individual events.

Object creation

☑ All object create events
s3:ObjectCreated:*

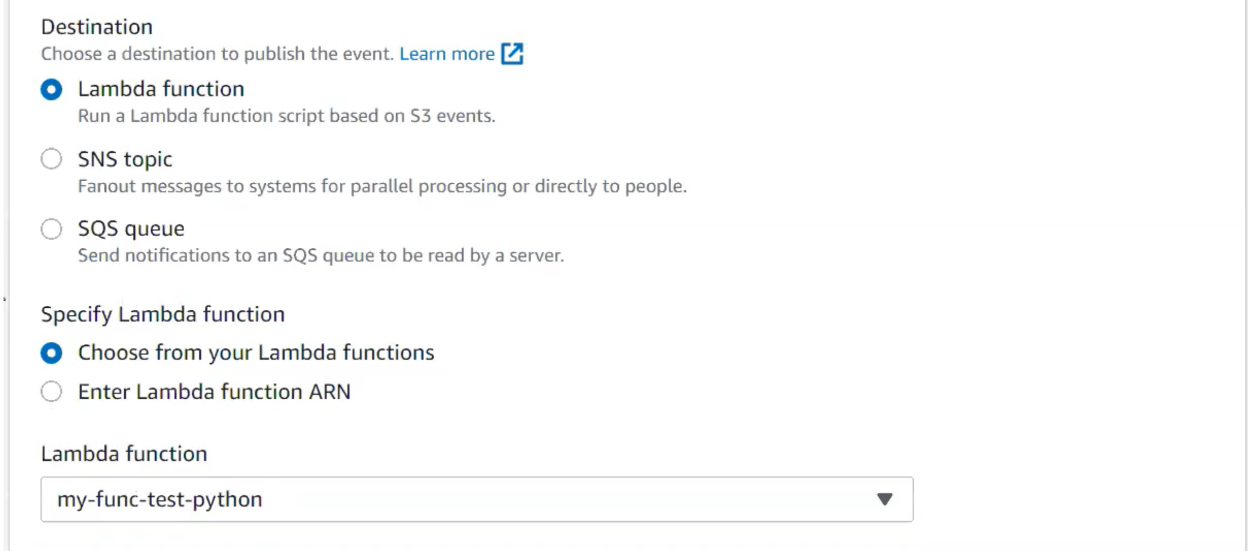
□ Put
s3:ObjectCreated:Put

□ Post
s3:ObjectCreated:Post

□ Copy
s3:ObjectCreated:Copy

□ Multipart upload completed
s3:ObjectCreated:CompleteMultipartUpload

4) Choose Destination & Select which Lambda function we want to run after occur a event from above selected event types



Destination
Choose a destination to publish the event. Learn more

● Lambda function
Run a Lambda function script based on S3 events.

○ SNS topic
Fanout messages to systems for parallel processing or directly to people.

○ SQS queue
Send notifications to an SQS queue to be read by a server.

Specify Lambda function

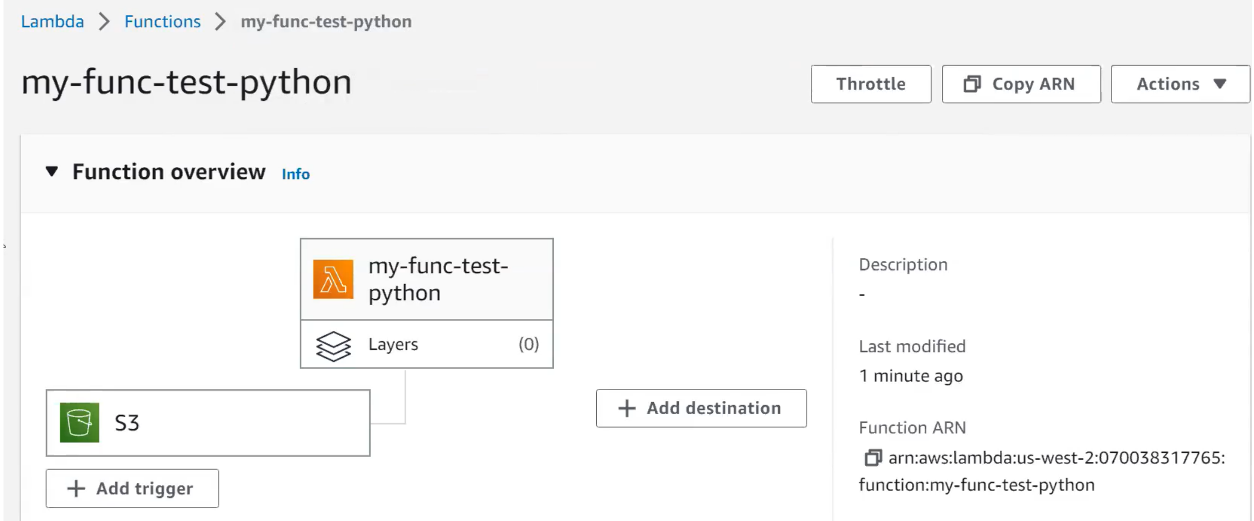
● Choose from your Lambda functions

○ Enter Lambda function ARN

Lambda function

my-func-test-python ▼

5) After above set up is done we can see in Lambda service :



Description
-

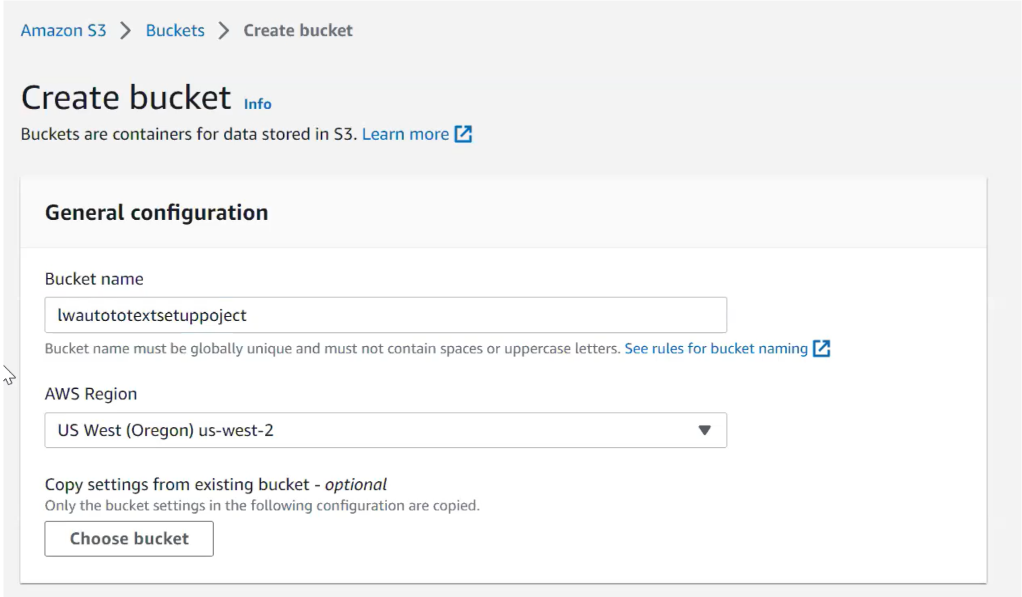
Last modified
1 minute ago

Function ARN
arn:aws:lambda:us-west-2:070038317765:function:my-func-test-python

- AWS provides one service which converts video or audio into text and that service is "**Amazon Transcribe**".

- **Demo exercise for integrating Lambda, S3, Transcribe** : Write a code in Lambda service which triggers as soon as new audio file comes in S3 service and using transcribe service that audio file will convert into text.

1) Create bucket in S3 service :



Create bucket Info

Buckets are containers for data stored in S3. Learn more

General configuration

Bucket name

lwautototextsetuppoject

Bucket name must be globally unique and must not contain spaces or uppercase letters. See rules for bucket naming

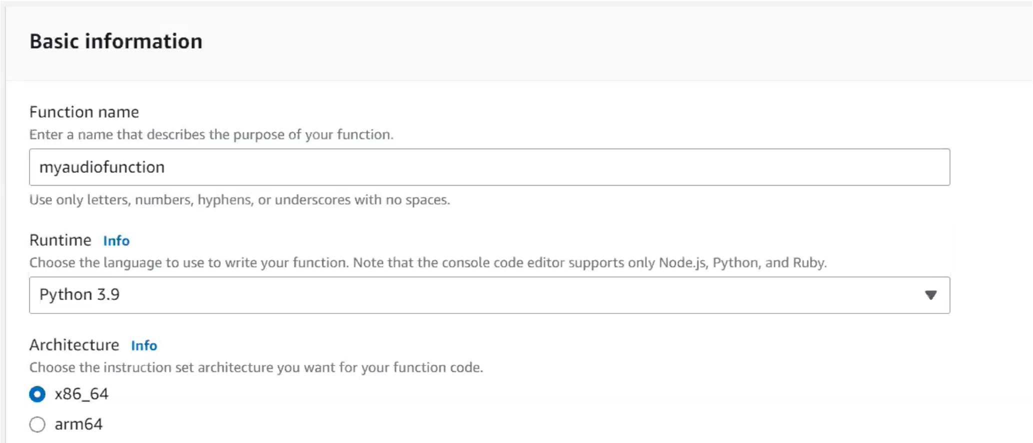
AWS Region

US West (Oregon) us-west-2

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

2) Create Lambda function :



Basic information

Function name

Enter a name that describes the purpose of your function.

myaudiofunction

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 3.9

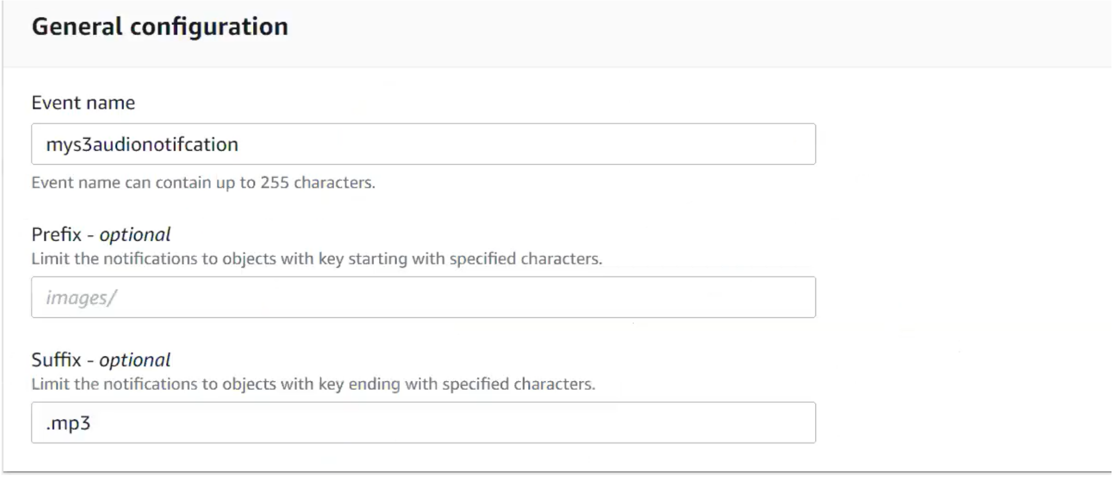
Architecture Info

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

● x86_64

○ arm64

3) Connect S3 with lambda : Select event type which occurs as soon as something uploads. Event only occur when “.mp3” file uploaded.



General configuration

Event name

mys3audionotifcation

Event name can contain up to 255 characters.

Prefix - *optional*

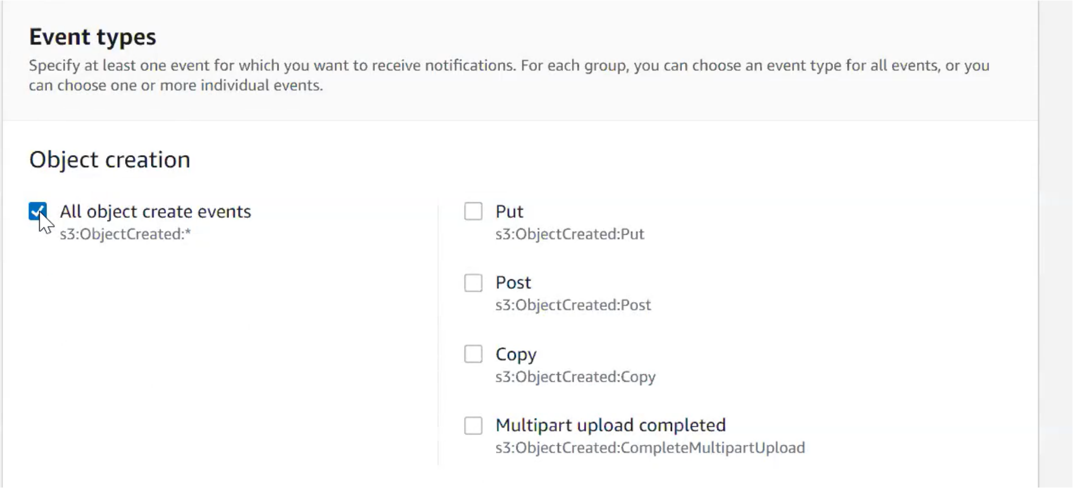
Limit the notifications to objects with key starting with specified characters.

images/

Suffix - *optional*

Limit the notifications to objects with key ending with specified characters.

.mp3



Event types

Specify at least one event for which you want to receive notifications. For each group, you can choose an event type for all events, or you can choose one or more individual events.

Object creation

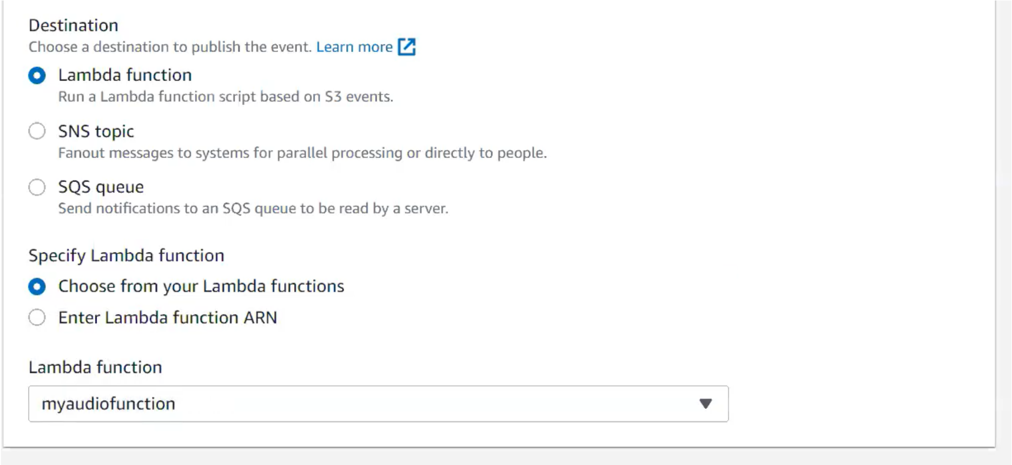
☑ All object create events
s3:ObjectCreated:*

□ Put
s3:ObjectCreated:Put

□ Post
s3:ObjectCreated:Post

□ Copy
s3:ObjectCreated:Copy

□ Multipart upload completed
s3:ObjectCreated:CompleteMultipartUpload



Destination

Choose a destination to publish the event. Learn more

● Lambda function
Run a Lambda function script based on S3 events.

○ SNS topic
Fanout messages to systems for parallel processing or directly to people.

○ SQS queue
Send notifications to an SQS queue to be read by a server.

Specify Lambda function

● Choose from your Lambda functions

○ Enter Lambda function ARN

Lambda function

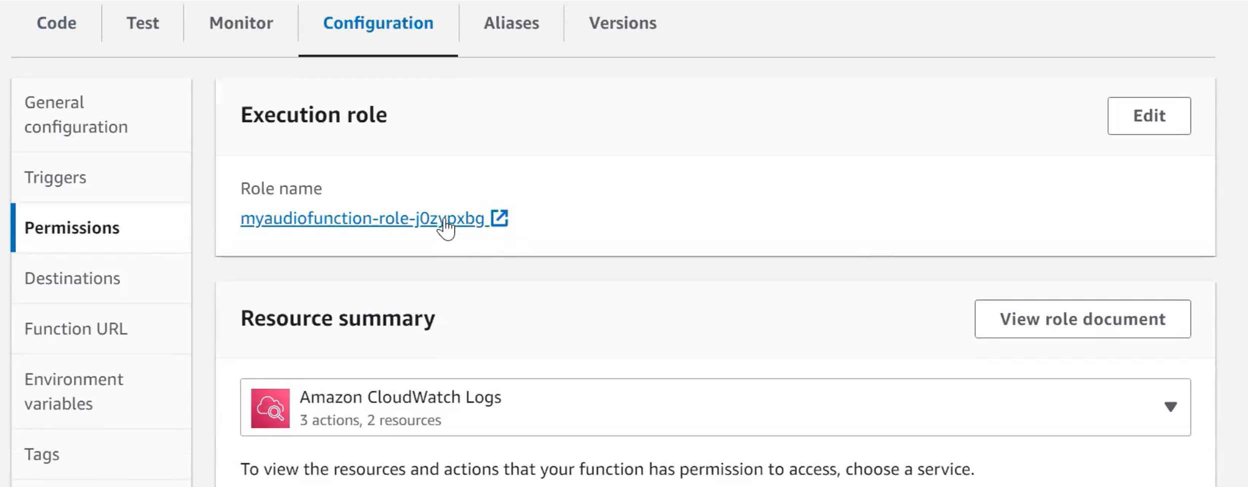
myaudiofunction ▼

4) Write code in lambda function which contacts to transcribe service then convert provided mp3 audio file into text.

Github code file url : https://github.com/vimallinuxworld13/aws_lambda_transcribe_audio.git

- If we want to connect from one service to another service in aws then by default they don’t allow it. We have to create a role for this.

5) Go to lambda function -> Configuration -> Permissions then click on role.



Execution role

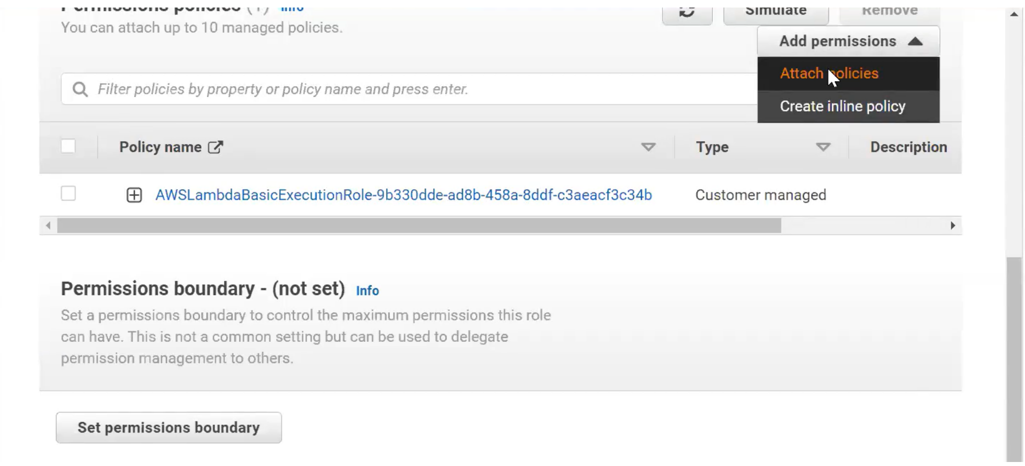
Role name
myaudiofunction-role-j0zypxbg

Resource summary

Amazon CloudWatch Logs
3 actions, 2 resources

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

6) Attach policies for different purposes such as Amazon Transcribe, Amazon S3, CloudWatch.



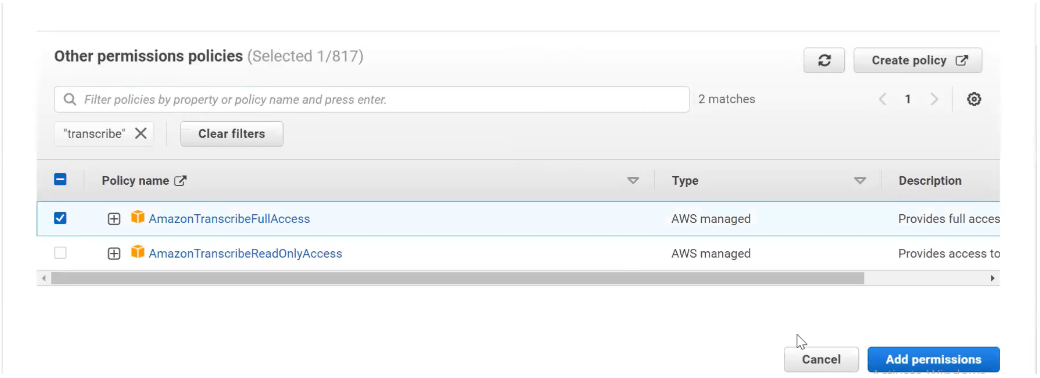
Permissions policies (1) Info
Simulate
Remove
Add permissions
Attach policies
Create inline policy
Filter policies by property or policy name and press enter.
Set permissions boundary

You can attach up to 10 managed policies.

	Policy name	Type	Description
□	AWSLambdaBasicExecutionRole-9b330dde-ad8b-458a-8ddf-c3aeacf3c34b	Customer managed	

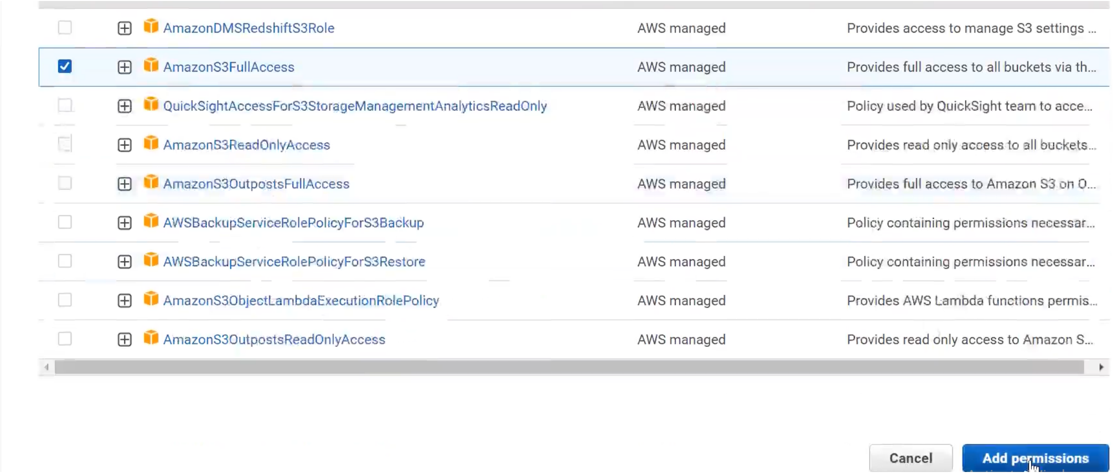
Permissions boundary - (not set) Info

Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting but can be used to delegate permission management to others.

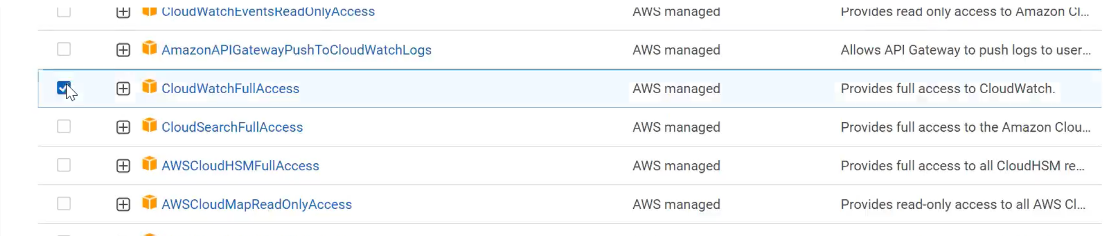


Other permissions policies (Selected 1/817)

	Policy name	Type	Description
☑	AmazonTranscribeFullAccess	AWS managed	Provides full acces
□	AmazonTranscribeReadOnlyAccess	AWS managed	Provides access to



	Policy name	Type	Description
□	AmazonDMSRedshiftS3Role	AWS managed	Provides access to manage S3 settings ...
☑	AmazonS3FullAccess	AWS managed	Provides full access to all buckets via th...
□	QuickSightAccessForS3StorageManagementAnalyticsReadOnly	AWS managed	Policy used by QuickSight team to acce...
□	AmazonS3ReadOnlyAccess	AWS managed	Provides read only access to all buckets...
□	AmazonS3OutpostsFullAccess	AWS managed	Provides full access to Amazon S3 on O...
□	AWSBackupServiceRolePolicyForS3Backup	AWS managed	Policy containing permissions necessar...
□	AWSBackupServiceRolePolicyForS3Restore	AWS managed	Policy containing permissions necessar...
□	AmazonS3ObjectLambdaExecutionRolePolicy	AWS managed	Provides AWS Lambda functions permis...
□	AmazonS3OutpostsReadOnlyAccess	AWS managed	Provides read only access to Amazon S...



□	CloudWatchEventsReadOnlyAccess	AWS managed	Provides read only access to Amazon Cl…
□	AmazonAPIGatewayPushToCloudWatchLogs	AWS managed	Allows API Gateway to push logs to user…
☑	CloudWatchFullAccess	AWS managed	Provides full access to CloudWatch.
□	CloudSearchFullAccess	AWS managed	Provides full access to the Amazon Clou…
□	AWSCloudHSMFullAccess	AWS managed	Provides full access to all CloudHSM re…
□	AWSCloudMapReadOnlyAccess	AWS managed	Provides read-only access to all AWS Cl…

Finally as soon as we upload an mp3 file in S3 bucket, lambda function will run and it will contact the transcribe service then it will convert text from the provided mp3 file in S3.