



48°

Lab - AWS re/Start

**[Desafío] Ejercicio de AWS
Lambda**



Tarea 01



Interactuando con AWS Lambda

Los objetivos son:

- Cree una función Lambda para contar el número de palabras de un archivo de texto.
- Configure un bucket de Amazon S3 para invocar una función de Lambda cuando se cargue un archivo de texto en el bucket de S3.
- Cree un tema de Amazon Simple Notification Service (Amazon SNS) para informar del recuento de palabras en un email.

Nota. El formato de envío del mensaje es el siguiente:

The word count in the <textFileName> file is nnn.

Y el asunto del e-mail a enviar es:

Word Count Result

Tarea 01



Empezaremos creando un rol, que permita a Lambda tener acceso al bucket de S3. Como en el lab no está permitido el acceso a IAM, utilizaremos el **LambdaAccessRole**. Cuyos permisos son:

- WSLambdaBasicExecutionRole
 - This is an AWS managed policy that provides write permissions to Amazon CloudWatch Logs.
- AmazonSNSFullAccess
 - This is an AWS managed policy that provides full access to Amazon SNS via the AWS Management Console.
- AmazonS3FullAccess
 - This is an AWS managed policy that provides full access to all buckets via the AWS Management Console.
- CloudWatchFullAccess
 - This is an AWS managed policy that provides full access to Amazon CloudWatch.

Tarea 01



Vamos a crear la función Lambda entonces:

☒ 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.

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 3.8

↕

↻

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

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.

LambdaAccessRole

↕

↻

Asimismo, creamos el bucket de S3 donde irán los .txt

[Amazon S3](#) > [Buckets](#) > Create bucket

Create bucket

[Info](#)

Buckets are containers for data stored in S3.

General configuration

AWS Region

US West (Oregon) us-west-2

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 - New
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](#)

mr-v-lab-files-bucket

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

New Feature

Amazon SNS now supports in-place message archiving and replay for FIFO topics. [Learn more](#)

Topic mail-count-worlds created successfully.

You can create subscriptions and send messages to them from this topic.

Amazon SNS

>

Topics

>

mail-count-worlds

mail-count-worlds

Edit

Delete

Publish message

Details

Name

mail-count-worlds

Display name

Word Count Result

ARN

arn:aws:sns:us-west-2:905418393759:mail-count-worlds

Topic owner

905418393759

Type

Standard

Subscriptions

Access policy

Data protection policy

Delivery policy (HTTP/S)

Delivery status logging


Encryption

Tags

Integrations

Asignamos nuestro trigger que nos permita activar la función lambda cada que se cree un nuevo archivo en el bucket S3

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.

✕ ↺

Bucket region: us-west-2

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.

Recursive invocation

If your function writes objects to an S3 bucket, ensure that you are using different S3 buckets for input and output. Writing to the same bucket increases the risk of creating a recursive invocation, which can result in increased Lambda usage and increased costs. [Learn more](#)

☒ I acknowledge that using the same S3 bucket for both input and output is not recommended and that this configuration can cause recursive invocations, increased Lambda usage, and increased costs.

Lambda will add the necessary permissions for AWS S3 to invoke your Lambda function from this trigger. [Learn more](#) about the Lambda permissions model.

Tarea 01



Y asimismo, debemos configurar la acción para publicar en el t pico de SNS que hemos creado. Pero antes debemos configurar quien recibe el mensaje.

Amazon SNS > Subscriptions > Create subscription

Create subscription

Details

Topic ARN

Protocol

The type of endpoint to subscribe

Email

Endpoint

An email address that can receive notifications from Amazon SNS.

After your subscription is created, you must confirm it. [Info](#)

As  quedar a nuestra funci n Lambda:

Lambda > Functions > count-words-function

count-words-function

Throttle Copy ARN Actions

Function overview Info

Diagram Template

count-words-function

Layers (0)

S3

Amazon SNS

+ Add trigger + Add destination

Export to Application Composer Download

Description

Last modified

15 minutes ago

Function ARN

arn:aws:lambda:us-west-2:905418393759:function:count-words-function

Function URL Info

Upload from

File Edit Find View Go Tools Window Test Deploy

Go to Anything (Ctrl-P)

Environment

count-words-functic

lambda_function.py

```
1 import boto3
2 import json
3 from io import BytesIO
4 from zipfile import ZipFile
5
6 def lambda_handler(event, context):
7     # Obtener informaci n del evento S3
8     s3_event = event['Records'][0]['s3']
9     bucket_name = s3_event['bucket']['name']
10    object_key = s3_event['object']['key']
11
12    # Descargar el archivo de texto desde S3
13    s3_client = boto3.client('s3')
14    response = s3_client.get_object(Bucket=bucket_name, Key=object_key)
15    file_content = response['Body'].read().decode('utf-8')
16
17    # Contar el n mero de palabras
18    word_count = len(file_content.split())
19
20    # Imprimir el resultado en CloudWatch Logs
21    print(f'The word count in the {object_key} is {word_count}')
22
23    # Retornar el resultado
24    return {
25        'statusCode': 200,
26        'body': json.dumps({'word_count': word_count})
27    }
```