



AWS Lambda

Exemplo prático de uso

`verificacao-s3-orfaos-lambda-dev`

Create 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.

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 [Info](#)

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

Durable execution - new [Info](#)

Enable durable execution to simplify building resilient multi-step applications that checkpoint progress and resume after interruptions. Supports Python and Node.js runtimes. [View pricing](#) [↗](#).

Enable

Architecture [Info](#)

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

arm64

x86_64

Create Function

☰ [Lambda](#) > [Functions](#) > Create function

Enable

Architecture [Info](#)

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

arm64

x86_64

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

To access other AWS services and resources your function needs an IAM role. Choose a role with the right permissions for your function.

Create default role

Use another role

A new role will be created named "lambdaFunction-role-iqacp1o6" to give your function permission to upload logs to Amazon CloudWatch Logs.

▶ Additional configurations

Use additional configurations to set up networking, security, and governance for your function. These settings help secure and customize your Lambda function deployment.

[Cancel](#)

[Create function](#)

Create Function - Runtime Languages

Search Runtime
Latest supported
.NET 10 (C#/F#/PowerShell)
Java 25
Node.js 24.x
Python 3.14
Ruby 3.4
Amazon Linux 2023 OS-only runtime for Go, Rust, C++, custom
Other supported
.NET 8 (C#/F#/PowerShell)
Java 11
Java 17
Java 21
Java 8 on Amazon Linux 2

Search Runtime
Other supported
.NET 8 (C#/F#/PowerShell)
Java 11
Java 17
Java 21
Java 8 on Amazon Linux 2
Node.js 20.x
Node.js 22.x
Python 3.10
Python 3.11
Python 3.12
Python 3.13
Ruby 3.2
Ruby 3.3

Function Overview

☰ [Lambda](#) > [Functions](#) > verificacao-s3-orfaos-lambda-dev



verificacao-s3-orfaos-lambda-dev

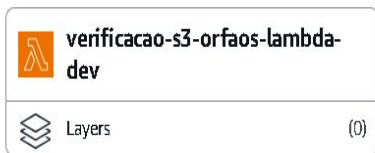
Throttle

📄 Copy ARN

Actions ▾

▼ Function overview [Info](#)

Diagram | Template



+ Add trigger

+ Add destination

Export to Infrastructure Composer

Download ▾

Description

Verificacao de s3 orfaos

Last modified

[3 days ago](#)

Function ARN

📄 `arn:aws:lambda:us-east-2:791659331772:function:verificacao-s3-orfaos-lambda-dev`

Function URL [Info](#)

-

Código – Imports & Inicialização

lambda_verificacao_s3_orfaos_function.py

The screenshot displays the AWS Lambda console interface for the `lambda_verificacao_s3_orfaos_function.py` file. The top navigation bar includes tabs for **Code**, **Test**, **Monitor**, **Configuration**, **Aliases**, and **Versions**. The **Code source** tab is selected, showing the code editor with the following Python code:

```
1 import dateutil.tz
2 import os
3 import psycopg2
4 import logging
5 import boto3
6 import base64
7 import json
8 from boto3.exceptions import ClientError
9 from datetime import datetime
10
11 saopaulo = dateutil.tz.gettz('America/Sao_Paulo')
12 date = datetime.now(saopaulo)
13 formatted_date = date.strftime("%Y-%m-%d")
14
15 _subject = os.environ['subject']
16 snsArn = os.environ['snsArn']
17 environment = os.environ['environment']
18 _secrets_name = os.environ['secrets_name']
19 _region = os.environ['region']
20 futuramanagerregion = os.environ['futuramanagerregion']
21
```

The Explorer on the left shows the file structure for **VERIFICACAO-S3-ORFAOS-LAMBDA-DEV**, including folders for `psycopg2`, `psycopg2_binary-2.9.10.dist-info`, `psycopg2_binary.libs`, and the file `lambda_verificacao_s3_orfaos_function.py`. Below the Explorer, the **DEPLOY** section shows the current deployment and buttons for **Deploy (Ctrl+Shift+U)** and **Test (Ctrl+Shift+I)**.

Código – Imports & Inicialização

lambda_verificacao_s3_orfaos_function.py

```
lambda_verificacao_s3_orfaos_function.py X
lambda_verificacao_s3_orfaos_function.py
28
29 if (environment != 'prd'):
30     payload = {"data": formatted_date}
31     invoke_response = lambda_client.invoke(FunctionName="consultarferiados-lambda",
32                                           InvocationType='RequestResponse',
33                                           Payload=json.dumps(payload))
34
35     response_payload = invoke_response['Payload'].read().decode()
36
37     diaFeriado = ""
38     descricaoFeriado = ""
39
40     if (response_payload != '{}') :
41         feriado = True
42         response_arr = json.loads(response_payload)
43
44         diaFeriado = response_arr['data']
45         descricaoFeriado = response_arr['descricao']
46
47 if(not bool(feriado)):
48     client = session.client(
49         service_name='secretsmanager',
50         region_name=_region
51     )
52
53     _secrets = client.get_secret_value(SecretId=_secrets_name)
54     secret = json.loads(_secrets['SecretString'])
55
56     # Calling SecretsManager
```

Código — Secrets Manager

lambda_verificacao_s3_orfaos_function.py

lambda_verificacao_s3_orfaos_function.py X

lambda_verificacao_s3_orfaos_function.py

```
47     if(not bool(feriado)):  
  
56         # Calling SecretsManager  
57         _bucket_name = secret[os.environ['secretsFuturaBucketName']]  
58         _host = secret[os.environ['secretsDefaultServer']]  
59         _host = _host[:_host.index(":")]  
60         _database = secret[os.environ['secretsDefaultDatabase']]  
61         _user = secret[os.environ['secretsDefaultDatabaseUser']]  
62         _password = secret[os.environ['secretsDefaultDatabasePass']]  
63         _port = secret[os.environ['secretsDefaultPort']]  
  
64  
65         db_instance = os.environ['db_instance']  
  
66  
67         rds = boto3.client('rds', region_name=_region)  
68         db = rds.describe_db_instances(DBInstanceIdentifier=db_instance)  
69
```

Código – Handler e Consulta RDS

Seleção de imagens órfãs e deleção do S3

lambda_verificacao_s3_orfaos_function.py X

lambda_verificacao_s3_orfaos_function.py

```
87 def lambda_handler(event, context):
91     if(not bool(feriado)):
92         if(bool(bancoAtivo)):
108             for row in res:
119
120                 if (lista_ids != '') :
121                     sql1 = "delete FROM upload.imagem where (imagem_id = ANY ('{"
122                     sql2 = f"\"{lista_ids}\""
123                     sql3 = "})'"
124
125                     cursor.execute(sql1+sql2+sql3)
126
127                     resultado = f"[ {origin} ] - Imagens órfãs foram removidas: \n Data: {date} \n Status: {status} \n Registros: {count}"
128                     print(resultado)
129
130                     sns.publish (
131                         TargetArn=snsArn,
132                         Subject=subject,
133                         Message=resultado,
134                         MessageStructure='string');
135                 else :
136                     sns.publish (
137                         TargetArn=snsArn,
138                         Subject=subject,
139                         Message= f"[ {origin} ] - Nenhum objeto órfão foi encontrado no momento {' if (error_details == '' ) else {error_details}"
140                         MessageStructure='string')
141
142     conn.commit()
143
```

Código – Deleção dos registros e Notificação SNS

Remoção de registros e publicação de resultado via SNS

```
lambda_verificacao_s3_orfaos_function.py X
lambda_verificacao_s3_orfaos_function.py
87 def lambda_handler(event, context):
91     if(not bool(feriado)):
92         if(bool(bancoAtivo)):
108             for row in res:
119
120                 if (lista_ids != '') :
121                     sql1 = "delete FROM upload.imagem where (imagem_id = ANY ('{"
122                     sql2 = f"\"{lista_ids}\""
123                     sql3 = "}))"
124
125                     cursor.execute(sql1+sql2+sql3)
126
127                     resultado = f"[ {origin} ] - Imagens orfãs foram removidas: \n Data: {date} \n Status: {status} \n Registros: {len(res)}"
128                     print(resultado)
129
130                     sns.publish (
131                         TargetArn=snsArn,
132                         Subject=subject,
133                         Message=resultado,
134                         MessageStructure='string');
135                 else :
136                     sns.publish (
137                         TargetArn=snsArn,
138                         Subject=subject,
139                         Message= f"[ {origin} ] - Nenhum objeto orfão foi encontrado no momento {' if (error_details == '') else error_details}"
140                         MessageStructure='string')
141
142             conn.commit()
143
```

Test Event

Configuração do evento de teste e JSON de entrada

Code **Test** Monitor Configuration Aliases Versions

Test event [Info](#)

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

CloudWatch Logs Live Tail

Save

Test

Test event action

Create new event

Edit saved event

Invocation type

Synchronous

Executes the Lambda function and blocks until receiving the function's response, with a maximum timeout of 15 minutes. Returns function output or error details directly to the calling application.

Asynchronous

Enqueues the Lambda function for execution and returns immediately with a request ID. Function processes independently, with results optionally sent to a configured destination like SQS, SNS, or EventBridge.

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

Test Event

Configuração do evento de teste e JSON de entrada

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

Hello World



Event JSON

Format JSON

```
1 {
2   "detail-type": "S3OrfaosCheck",
3   "source": "manual.test",
4   "detail": {
5     "buckets": ["bucket-nfe-dev", "bucket-docs-dev"],
6     "max_age_days": 30,
7     "delete_orphans": false,
8     "notify_sns": true
9   }
10 }
```



Configuration - General Configuration

Memória, timeout, storage e SnapStart

Code Test Monitor **Configuration** Aliases Versions

Configuration

- General configuration**
- Triggers
- Permissions
- Destinations
- Function URL
- Environment variables
- Tags
- VPC
- RDS databases
- Monitoring and operations tools
- Concurrency and recursion detection
- Asynchronous invocation

General configuration [Info](#)

Description
Verificacao de s3 orfaos

Memory
128 MB

Ephemeral storage
512 MB

Timeout
0 min 30 sec

SnapStart [Info](#)
None

[Edit](#)

Configuration - Triggers – EventBridge

Code Test Monitor **Configuration** Aliases Versions

Configuration

General configuration

Triggers

Permissions

Destinations

Function URL

Environment variables

Tags

VPC

RDS databases

Monitoring and operations tools

Concurrency and recursion
detection

Asynchronous invocation

Triggers (1) [Info](#)



Fix errors

Edit

Delete

Add trigger

Search for triggers

< 1 >

Trigger



EventBridge (CloudWatch Events): verificacao-s3-orfaos-dev

arn:aws:events:us-east-2:791659331772:rule/verificacao-s3-orfaos-dev

Rule state: **ENABLED**

▼ Details

Description: **Executa verificacao de S3 orfaos segunda a sexta as 18h40**

Event bus: **default**

Event pattern:

```
 {
  "detail-type": [
    "AWS Console Sign In via CloudTrail"
  ]
}
```

isComplexStatement: **No**

Schedule expression: **cron(40 21 ? * MON-FRI *)**

Service principal: **events.amazonaws.com**

Statement ID: **AllowExecutionVerificacaoS3OrfaosFromCloudWatchEvent**

Configuration - Destinations

Invocação assíncrona — On Failure → SNS

Code | Test | Monitor | **Configuration** | Aliases | Versions

Configuration

- General configuration
- Triggers
- Permissions
- Destinations**
- Function URL
- Environment variables
- Tags
- VPC
- RDS databases
- Monitoring and operations tools
- Concurrency and recursion detection
- Asynchronous invocation

Destinations [Info](#)

Source	Event source mapping	Condition	Destination
<input type="radio"/> Asynchronous invocation	-	On failure	arn:aws:sns:us-east-2:791659331772:futura-sns-dev

[Remove](#) [Edit](#) [Add destination](#)

Configuration - Environment Variables

Variáveis criptografadas em repouso com a chave padrão do Lambda

- General configuration
- Triggers
- Permissions
- Destinations
- Function URL
- Environment variables**
- Tags
- VPC
- RDS databases
- Monitoring and operations tools
- Concurrency and recursion detection
- Asynchronous invocation
- Code signing
- File systems
- State machines

The environment variables below are encrypted at rest with the default Lambda service key.

Configuration - VPC Configuration

VPC, Subnets e Security Groups

Code | Test | Monitor | **Configuration** | Aliases | Versions

Configuration

- General configuration
- Triggers
- Permissions
- Destinations
- Function URL
- Environment variables
- Tags
- VPC**
- RDS databases
- Monitoring and operations tools
- Concurrency and recursion detection
- Asynchronous invocation

VPC Info

VPC
[vpc-0c1801daaecdb9c21](#) (11.0.0.0/24) | futura-vpc-app-dev

Subnets

- Allow IPv6 traffic = false
- [subnet-0224cd2485922081e](#) (11.0.0.32/27) | us-east-2b, futura-pvt-subnet-app-2-dev
- [subnet-0c7a6f15fc08d18a7](#) (11.0.0.0/27) | us-east-2a, futura-pvt-subnet-app-1-dev

Security groups

- [sg-04d13f0be6cb63bc4](#) (futura-sg-lambda-v2-dev) | futura-sg-lambda-v2-dev

Security group rules (0) Last fetched 0 seconds ago

Inbound rules

Security group ID	Protocol	Ports	Source
-------------------	----------	-------	--------

Configuration - RDS Database Connections

Conexão com banco monolith via security groups

Code | Test | Monitor | **Configuration** | Aliases | Versions

Configuration

- General configuration
- Triggers
- Permissions
- Destinations
- Function URL
- Environment variables
- Tags
- VPC
- RDS databases**
- Monitoring and operations tools
- Concurrency and recursion detection
- Asynchronous invocation

RDS database connections (1) [Info](#) Ⓢ Add Proxy Connect to RDS database

The following table shows databases that match this function's VPC and security group. Review the details to confirm a connection.

	DB identifier	Proxy identifier	Database Status	Lambda security group	DB security group
<input type="radio"/>	monolith ↗	-	✔ Available	futura-sg-lambda-v2-dev (sg-0...	futura-sg-rds-v2-dev (sg-09cf8...



Obrigado!

Alguma sugestão?
Alguma dúvida?