

Serverless with python and AWS Lambda

by Luis Mejia





¿Qué es Serverless?

Serverless es un tipo de arquitectura donde los servidores (físicos o en la nube) dejan de existir para el desarrollador y en cambio el código corre en “ambientes de ejecución” que administran proveedores como Amazon, Google, IBM, etc.



¿Es mi aplicación adecuada para serverless?

Si tu aplicación es:

- en su mayoría STATELESS, como una API REST o GraphQL, un worker, entonces **serveless** puede ser un gran ajuste, ya que las solicitudes HTTP rara vez van más allá de los 30 segundos, y mucho menos de cinco minutos.

Proveedores para Serverless

Cloud providers



IBM
OpenWhisk



AWS
Lambda



Azure
Functions



Google
Cloud
Functions



Auth0
Webtask



Herramientas para Serverless

- **Serverless Framework** : es un open-source framework escrito en Node.js.
- **Zappa**: Serverless Python WSGI con AWS Lambda + API Gateway.
- **Chalice**: Python Serverless Microframework para AWS



Crear credenciales en AWS

Add user



Set user details

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name*

[+](#) Add another user

Select AWS access type

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Access type*



Programmatic access

Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.



AWS Management Console access

Enables a **password** that allows users to sign-in to the AWS Management Console.



Serverless en minutos usando Chalice

```
mkdir ~/.aws
vim ~/.aws/config
[default]
aws_access_key_id=YOUR_ACCESS_KEY_HERE
aws_secret_access_key=YOUR_SECRET_ACCESS_KEY
region=YOUR_REGION (such as us-west-2, us-west-1, etc)
```



Serverless en minutos usando Chalice

```
mkdir chalice-demo  
cd chalice-demo/  
virtualenv env  
source env/bin/activate  
pip install chalice  
chalice new-project helloworld  
cd helloworld  
chalice deploy
```

<https://aws.amazon.com/blogs/developer/build-and-deploy-a-serverless-rest-api-in-minutes-using-chalice/>





Serverless en minutos usando Serverless

```
npm install -g serverless
```

```
serverless config credentials --provider aws --key  
AKIAIOSFODNN7EXAMPLE --secret  
wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
```

<https://serverless.com/framework/docs/providers/aws/guide/installation/>



Serverless en minutos usando Serverless

```
mkdir ~/my-serverless-project
```

```
cd ~/my-serverless-project
```

```
sls create -n my-serverless-project -t aws-python3
```

```
sls deploy
```