

# **Despliegue de funciones lambda con Docker**

- Hasta hace poco la única manera de desplegar una función lambda estaba limitada a un paquete zip.
- Esto tenía algunas limitaciones como por ejemplo si queremos instalar alguna utilidad a nivel de sistema.
- Ahora podemos desplegar una función lambda mediante una imagen docker:

Para ello necesitamos crear los siguientes ficheros:

- `app.py`
- `dockerfile`
- `requirements.txt`

## app.py

```
def handler(event, context):  
    return 'Hola'
```

# dockerfile

```
FROM public.ecr.aws/lambda/python:3.8

# Copy function code
COPY app.py ${LAMBDA_TASK_ROOT}

# Install the function's dependencies using file requirements.txt
# from your project folder.

COPY requirements.txt .
RUN pip3 install -r requirements.txt --target "${LAMBDA_TASK_ROOT}"

# Set the CMD to your handler (could also be done as a parameter override outside of the Dockerfile)
CMD [ "app.handler" ]
```

# Test

- Podemos testear la función localmente.

```
docker build -t random-letter .
```

```
docker run -p 9000:8080 random-letter:latest
```

```
curl -XPOST "http://localhost:9000/2015-03-31/functions/function/invocations" -d '{}'
```

# Despliegue:

- Para desplegar la función primero creamos un nuevo repositorio en nuestro ECR:

```
aws ecr create-repository --repository-name repository-name --region region
```

- Tagamos nuestra imagen.

```
docker tag hello-world aws_account_id.dkr.ecr.region.amazonaws.com/repository-name
```

- Logeamos docker

```
aws ecr get-login-password | docker login --username AWS --password-stdin aws_account_id.dkr.ecr.region.amazonaws.com
```

- Hacemos un push de la imagen.

```
docker push aws_account_id.dkr.ecr.region.amazonaws.com/repository-name
```

- Para terminar creamos una nueva función lamnda desde la consola de aws usando la imagen que acabamos de subir al ECR.

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.

**Browse serverless app repository** ☐

Deploy a sample Lambda application from the AWS Serverless Application Repository.

### Basic information

**Function name**  
Enter a name that describes the purpose of your function.

Use only letters, numbers, hyphens, or underscores with no spaces.

**Container image URI** [Info](#)

The location of the container image to use for your function.

Requires a valid Amazon ECR image URI.

**Browse images**

### Container image URI

Enter the URI of your container image for this function. Choose **Browse Images** to help you find the correct image. Images are stored in the Amazon ECR repository.

[Learn more](#) [↗](#)

[Configuring functions defined as container images](#)



- Tambien se puede realizar la actualización con un comando:

```
aws lambda update-function-code --region eu-west-3 --function-name funDocker --image-uri aws_account_id.dkr.ecr.region.amazonaws.com/repository-name
```

## Ejercicio

- Usando el código de ejemplo de esta presentación crea una función lambda y despliegala usando una imagen docker.