

Introducción a AWS y Configuración de Entornos

Recursos

- [Primeros pasos: ¿Es la primera vez que usa AWS?](#)
- [AWS Well-Architected Framework](#)
 - AWS Well-Architected Framework le ayuda a comprender las ventajas y desventajas de las decisiones que toma al crear sistemas en AWS. El uso del marco le permitirá conocer las prácticas recomendadas de arquitectura para diseñar y operar sistemas en la nube que sean fiables, seguros, eficaces, rentables y sostenibles.
- [This is My Architecture](#)
 - “Esta es mi arquitectura” es una serie de vídeos que muestra soluciones arquitectónicas innovadoras en la nube de AWS realizadas por clientes y socios. Cada episodio examina los elementos más interesantes y técnicamente creativos de cada arquitectura en la nube.

Preguntas del chat

- Donde se pueden encontrar los valores de AWS Access Key ID y AWS Secret Access Key?
 - “Aunque no lo recomendamos, puede crear claves de acceso para el usuario raíz para poder ejecutar comandos en AWS Command Line Interface (AWS CLI) o utilizar las operaciones de la API desde uno de los AWS SDK con las credenciales de usuario raíz.”
 - Puedes encontrar el paso a paso aquí:
 - [Creación de claves de acceso para el usuario raíz](#)

Ejercicios

Cada uno de los siguientes ejercicios tienen un entregable en **bold**. Comparte el documento con las respuestas en Telegram con tus compañeros.

1. Crea una instancia EC2 por medio del AWS CLI. **Copia en el documento el comando del AWS CLI que utilizaste.**
 - Asegúrate de crear una llave ([Amazon EC2 Key Pair](#)) antes.
 - La instancia debe tener las siguientes propiedades:
 - Region: us-east-1
 - AMI: ami-0d7a109bf30624c99

- Instance type: t2.micro
- 2. Conectate a la instancia EC2 por medio de SSH y crea un archivo en el home directory de ec2-user con el contenido "Hola, mundo!"
- 3. Utiliza boto3 para crear una instancia EC2 con las siguientes propiedades. **Copia en el documento el script de Boto3.**
 - Region: us-east-1
 - AMI: ami-0d7a109bf30624c99
 - Instance type: t2.micro
- 4. Extra: Utiliza boto3 para crear una [herramienta para CLI](#) que te permita hacer las siguientes operaciones con el servicio S3. **Copia en el documento el script de Boto3.**
 - Subir un archivo a una S3 bucket
 - Por ejemplo
 - 1. `python s3_util.py --action=upload`
`--bucket_name=devops-bootcamp-cf --file_name=mi-archivo.jpeg`
 - Borrar un archivo de una S3 bucket
 - Por ejemplo:
 - 1. `python s3_util.py --action=delete`
`--bucket_name=devops-bootcamp-cf --file_name=mi-archivo.jpeg`
 - Enlistar los objetos en una S3 bucket
 - Por ejemplo:
 - 1. `python s3_util.py --action=list`
`--bucket_name=devops-bootcamp-cf`
- 5. Utiliza AWS CloudFormation (yaml) para crear una instancia EC2 con las siguientes propiedades. **Copia en el documento el template de Cloudformation.**
 - Region: us-east-1
 - AMI: ami-0d7a109bf30624c99
 - Instance type: t2.micro

Si tienes alguna consulta de los ejercicios, te recomiendo leer esta documentación o consultar los recursos de la presentación en el Google Drive.

1. [Conéctese a la instancia de Linux desde Linux o macOS mediante SSH](#)
2. [Utilizar la AWS CLI](#)
3. [Boto3 documentation](#)
4. [Anatomía de la plantilla de AWS CloudFormation](#)