

Crear el Bucket S3

Iremos a S3 desde la consola de AWS y haremos clic en Create bucket

Nombre del bucket: ecommerce-logs-bucket-st1630 (cambiarlo por un nombre único)

Block Public Access settings: Dejaremos marcada la opción Block all public access

Dejaremos las demás configuraciones por defecto y haremos clic en Create bucket

Crear la Tabla DynamoDB

Iremos a DynamoDB desde la consola de AWS y haremos clic en Create table.

Nombre de la tabla: EcommerceLogs

Partition key: crearemos una key con el nombre de log_id y diremos que será un String

Configuraciones de la tabla: Dejaremos la configuración por default y verificaremos que en Capacity mode esté configurado a On-demand

Haremos clic en Create table

Crear Kinesis Data Stream

Iremos a Amazon Kinesis desde la consola de AWS y haremos clic en Create data stream.

Data stream name: ecommerce-stream.

Capacity mode: On-demand

Haremos clic en Create data stream

Crear la Función Lambda

Iremos a Lambda desde la consola de AWS y haremos clic en Create function.

Function name: ProcessEcommerceLogs

Runtime: Python 3.13

Architecture: x86_64

Change default execution role: Seleccionaremos la opción Use an existing role y elegiremos el rol LabRole

Haremos clic en Create function

Copiaremos el código del archivo **lambda.py** dentro del código de la lambda y haremos clic en Deploy.

Ajustaremos la configuración general en Configuration y General Configuration.

Memory: 256MB

Timeout: 1 min 0 sec

Execution role: Seleccionaremos Use an existing role y elegiremos LabRole

Haremos clic en Save

Configuraremos el activador de la lambda.

En la página de la función, haremos clic en Add trigger

Seleccionaremos S3 como fuente

Bucket: Seleccionaremos el bucket creado ecommerce-logs-bucket-st1630 en nuestro caso

Event types: All object create events

Prefix (opcional): logs/

Suffix: .json

Marcaremos la casilla de reconocimiento sobre invocación recursiva

Haremos clic en Add

Crear y Ejecutar el Script Generador de Logs

Copiaremos el código del archivo **log_gen.py** en nuestra máquina local o una instancia EC2 cambiaremos la variable **BUCKET_NAME** por el nombre de nuestro bucket

Configuraremos las credenciales de AWS

Como es AWS Academy, obtendremos las credenciales dando clic en AWS Details en la pestaña del AWS Academy Learner Lab y haremos clic en AWS CLI

Copiaremos las credenciales en el archivo que se especifica.

Ejecutaremos el script **log_gen.py**

Consultar DynamoDB

Iremos a DynamoDB desde la consola de AWS y haremos clic en Tables > Explore table items > EcommerceLogs

Verificar métricas en CloudWatch

Iremos a CloudWatch desde la consola de AWS y haremos clic en Metrics > All metrics > EcommerceLogs. Deberíamos ver las métricas personalizadas creadas allí.