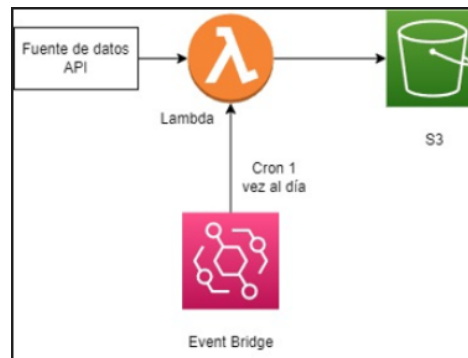


Tarea 2

José Araya y Diego Inzunza

Contexto carga de datos:

El siguiente proceso de adquisición de datos muestra la integración de 3 servicios de Amazon Aws para dar solución de información a un aplicativo para adopción de mascotas.



Los datos inicialmente son obtenidos desde la api publica <https://huachitos.cl/api/animales>. La que nos proporcionará todos los animales publicados para adopción.

```
{
  "data": [
    {
      "id": 1,
      "nombre": "Elvis",
      "tipo": "Perro",
      "edad": "4 años",
      "estado": "adopcion",
      "genero": "macho",
      "desc_fisica": "<p>Mix de frenchie de pelo corto negro, un poco jaspeado. Le cortaron su cola y solo tiene un c",
      "desc_personalidad": "<p>Es un poco baboso pero buen perro, es muy simpático, tierno, buena compañía y buen gua",
      "desc_adicional": "<p>Elvis necesita ejercicio diario para mantenerse saludable y feliz. Sería ideal tener un j",
      "esterilizado": 1,
      "vacunas": 1,
      "imagen": "https://huachitos.cl/storage/animal-images/elvis-37-1687295207.jpg",
      "equipo": "Fundación Patitas Felices",
    }
  ]
}
```

Para esto mediante AWS Event Bridge se generará un proceso batch que ejecute una vez al día una función AWS Lambda, la cual, tomará la información desde la api pública y la cargará a un bucket en S3 para posteriormente ser consumida por alguna aplicación.

1.- Llamar API y guardar información en S3.

1.- Crear función lambda y crear lambda function

Seleccione una de las siguientes opciones para crear la función.

☒ Crear desde cero

Empiece con un sencillo ejemplo "Hello World".

☐ Utilizar un proyecto

Cree una aplicación Lambda utilizando un código de muestra y los ajustes de configuración predefinidos de casos de uso comunes.

☐ Imagen del contenedor

Seleccione una imagen de contenedor para implementar para la función.

Información básica

Nombre de la función

Escriba un nombre para describir el propósito de la función.

apihuachitos_a_s3

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 (_).

Tiempo de ejecución [Información](#)

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

Python 3.10



Arquitectura [Información](#)

Elija la arquitectura del conjunto de instrucciones que desea para el código de la función.

☒ x86_64

☐ arm64

Permisos [Información](#)

De forma predeterminada, Lambda creará un rol de ejecución con permisos para cargar registros en Amazon CloudWatch Logs. Puede personalizar este rol predeterminado más adelante al agregar los disparadores.

► [Cambiar el rol de ejecución predeterminado](#)

2.- Añadir layer para trabajar con pandas

Capas Información

Editar

Añadir una capa

Orden de combinación	Nombre	Versión de la capa	Tiempos de ejecución compatibles	Arquitecturas compatibles	ARN de la versión
1	AWSSDKPandas-Python310	20	python3.10	x86_64	arn:aws:lambda:us-east-1:336392948345:layer:AWSSDKPandas-Python310:20

Lambda > Agregar capa

Agregar capa

Configuración del tiempo de ejecución de la función

Tiempo de ejecución
Python 3.10

Arquitectura
x86_64

Proporcionadas por AWS

AWS-AppConfig-Extension
Feature Flags, or other configuration data, for your AWS Lambda functions

LambdaInsightsExtension
CloudWatch Lambda Insights Extension.

AWS-Parameters-and-Secrets-Lambda-Extension
Retrieve parameters from AWS Systems Manager Parameter Store and secrets from AWS Secrets Manager

AWSSDKPandas-Python310
AWS SDK for pandas Lambda Layer - 3.9.1 (Python 3.10)

AWSLambdaPowertoolsPythonV2
Powertools for AWS Lambda (Python) [x86_64] with extra dependencies version 2.43.0

de instrucciones o especifique el nombre de recurso

○ Especificar un ARN
Proporcione el ARN y especifique una capa.

Elija

▲

Cancelar

Agregar

3.-Generar un paquete con biblioteca requests y cargar en lambda

3.1.-Crear una carpeta.

3.2.-Ejecutar en línea de comandos:
pip install requests -t (ruta de la carpeta)

3.3.-Guardar el código de la función en un archivo llamado lambda_function.py dentro de la misma carpeta.

3.4.-Comprimir los archivos dentro de la carpeta a un archivo.zip

3.5.- En aws lambda, en pestaña de código elegir Cargar desde, elegir archivo .zip y seleccionar archivo generado.

4.-En IAM generar una política para el rol asignado y así poder dar permiso para poder escribir y leer en S3.

Fecha de creación
October 06, 2024, 16:03 (UTC-03:00)

Última actividad
🟢 hace 51 minutos

ARN
arn:aws:iam::850995533867:role/service-role/apihuachitos_a_s3-role-g5vhcr4a

Duración máxima de la sesión
1 hora

Permisos

Relaciones de confianza

Etiquetas

Last Accessed

Revocar las sesiones

Políticas de permisos (2) [Información](#)

🔄

Simular

Eliminar

Agregar permisos

Buscar

Filtrar por Tipo
Todos los tipos

< 1 > ⚙

<input type="checkbox"/>	Nombre de la política	Tipo	Entidades asociadas
<input type="checkbox"/>	AmazonS3FullAccess	Administrada por AWS	2
<input type="checkbox"/>	AWSLambdaBasicExecutionRole-36dcde78-60f9-44ac-a02f-3c2083963cf8	Administrada por el cliente	1

5.- Añadir desencadenador:

Añadir un desencadenador con EventBridge para ejecutar el script una vez al día.

Agregar desencadenador

Configuración del desencadenador [Información](#)

EventBridge (CloudWatch Events)
aws asynchronous schedule management-tools

Regla

Seleccione una regla existente o cree una nueva.

☒ Cree una regla nueva

☐ Reglas existentes

Nombre de la regla

Especifique un nombre que identifique la regla de forma inequívoca.

lambda_día

Descripción de regla

Proporcione una descripción opcional para la regla.

Tipo de regla

Active el destino con arreglo a un patrón de eventos o una programación automatizada.

☐ Patrón de eventos

☒ Expresión de programación

Expresión de programación

Desencadene automáticamente el destino según una programación automatizada utilizando expresiones Cron o rate. Las expresiones Cron están en UTC.

cron(0 12 * * ? *)

por ejemplo, rate(1 day), cron(0 17 ? * MON-FRI *)

Lambda añadirá los permisos necesarios para Amazon EventBridge (CloudWatch Events) para invocar la función Lambda desde este desencadenador. [Obtenga más información](#) sobre el modelo de permisos de Lambda.

6.-Cambiar timeout a 10 segundos y memoria a 150 MB en configuración básica:

Configuración básica [Información](#)

Descripción - Opcional

Memoria [Información](#)
La CPU asignada a la función es proporcional a la memoria configurada.

MB

Establezca la memoria en un valor entre 128 MB y 10240 MB

Almacenamiento efímero [Información](#)
Puede configurar hasta 10 GB de almacenamiento efímero (/tmp) para la función. [Ver precios](#)

MB

Establezca el almacenamiento efímero (/tmp) entre 512 MB y 10240 MB.

SnapStart [Información](#)
Reduzca el tiempo de inicio haciendo que Lambda almacene en caché una instantánea de la función una vez esta se inicialice. Para evaluar si el código de la función es resistente a las operaciones de instantáneas, consulte las [consideraciones de compatibilidad de SnapStart](#).

None

Tiempos de ejecución admitidos: Java 11, Java 17, Java 21.

Tiempo de espera

min s

Rol de ejecución

7.-Añadimos el código del lambda function para llamar la información desde la API y cargarlo al bucket S3

lambda_function × Execution results × Environment Var × +

```
1 import requests
2 import pandas as pd
3 from datetime import datetime as dt
4 import boto3
5 from io import StringIO
6
7 def lambda_handler(event=None, context=None):
8     url = 'https://huachitos.cl/api/animales'
9
10    headers = {
11        'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Firefox/5.0'
12    }
13
14    try:
15        response = requests.get(url, headers=headers)
16
17        print('Lectura de API con éxito')
18        j = response.json()
19        df=pd.json_normalize(j['data'])
20        ubicacionstringio = StringIO()
21        df.to_csv(ubicacionstringio, index =False)
22        print('creacion de csv con éxito')
23
24        s3 = boto3.client('s3')
25
26        carpeta = 'apihuachitos/'
27        bucket = 'data-engineer-diplomadosach-2024'
28        nombre_en_s3 = f'{carpeta}huachitos_{dt.now().strftime("%Y%m%d_%H%M%S")}.csv'
29
30        try:
31            s3.put_object(Body=ubicacionstringio.getvalue(), Bucket=bucket, Key=nombre_en_s3)
32            print('Archivo cargado a S3 con éxito')
33        except:
34            print('Error al cargar a s3')
35            raise
36
37    except requests.exceptions.HTTPError as err:
38        if response.status_code == 403:
39            print("Error 403: Acceso denegado. Verifica tus permisos.")
40        else:
41            print(f"Error al realizar la solicitud: {err}")
```

8.- Probamos un test para verificar el funcionamiento, resultando exitoso.

Código fuente

Información

Cargar

FileEditFindViewGoToolsWindow

TestDeploy

Go to Anything (Ctrl-P)

Environment

apihuachitos_a_s3

bin

certifi

certifi-2024.8.30.dist-info

charset_normalizer

charset_normalizer-3.3.2

idna

idna-3.10.dist-info

requests

requests-2.32.3.dist-info

urllib3

urllib3-2.2.3.dist-info

lambda_function.py

Execution results

Test Event Name

prueba

Response

null

Function Logs

START RequestId: 7c3adc93-06a8-43c7-81de-88a5a07912e2 Version: \$LATEST

Lectura de API con éxito

creación de csv con éxito

Archivo cargado a S3 con éxito

END RequestId: 7c3adc93-06a8-43c7-81de-88a5a07912e2

REPORT RequestId: 7c3adc93-06a8-43c7-81de-88a5a07912e2 Duration: 3573.74 ms Billed Duration: 3574 ms Memory Size: 150 MB Max Memory Used: 150 MB Init Duration: 2854.51 ms

Request ID

7c3adc93-06a8-43c7-81de-88a5a07912e2

Status: Succeeded

Max memory used: 150 MB

apihuachitos/

Copiar URI de S3

Objetos

Propiedades

Objetos (4)

Información

Copiar URI de S3

Copiar URL

Descargar

Abrir

Eliminar

Acciones

Crear carpeta

Cargar

Los objetos son las entidades fundamentales que se almacenan en Amazon S3. Puede utilizar el [Inventario de Amazon S3](#) para obtener una lista de todos los objetos de su bucket. Para que otras personas obtengan acceso a sus objetos, tendrá que concederles permisos de forma explícita. [Más información](#)

Buscar objetos por prefijo

	Nombre	Tipo	Última modificación	Tamaño	Clase de almacenamiento
<input type="checkbox"/>	huachitos_20241006_200051.csv	csv	6 Oct 2024 5:00:52 PM -03	72.7 KB	Estándar
<input type="checkbox"/>	huachitos_20241006_201815.csv	csv	6 Oct 2024 5:18:16 PM -03	72.7 KB	Estándar
<input type="checkbox"/>	huachitos_20241006_220924.csv	csv	6 Oct 2024 7:09:25 PM -03	72.7 KB	Estándar
<input type="checkbox"/>	huachitos_20241006_223134.csv	csv	6 Oct 2024 7:31:35 PM -03	72.7 KB	Estándar