

CS2032 - Cloud Computing (Ciclo 2024-2)

Event-driven architecture

Semana 13 - Taller 2: SNS - Simple Notification Service

ELABORADO POR: GERALDO COLCHADO

Contenido

Event-driven architecture

1. **Objetivo del taller 2**
2. Ejercicio 1: Evento Nuevo Archivo en UTEC
3. Ejercicio 2: Ejercicio propuesto
4. Cierre

Event-driven architecture

Objetivo del Taller 2

- Diseño e implementación de una Arquitectura de Solución basada en eventos con el servicio “SNS - Simple Notification Service”

Contenido

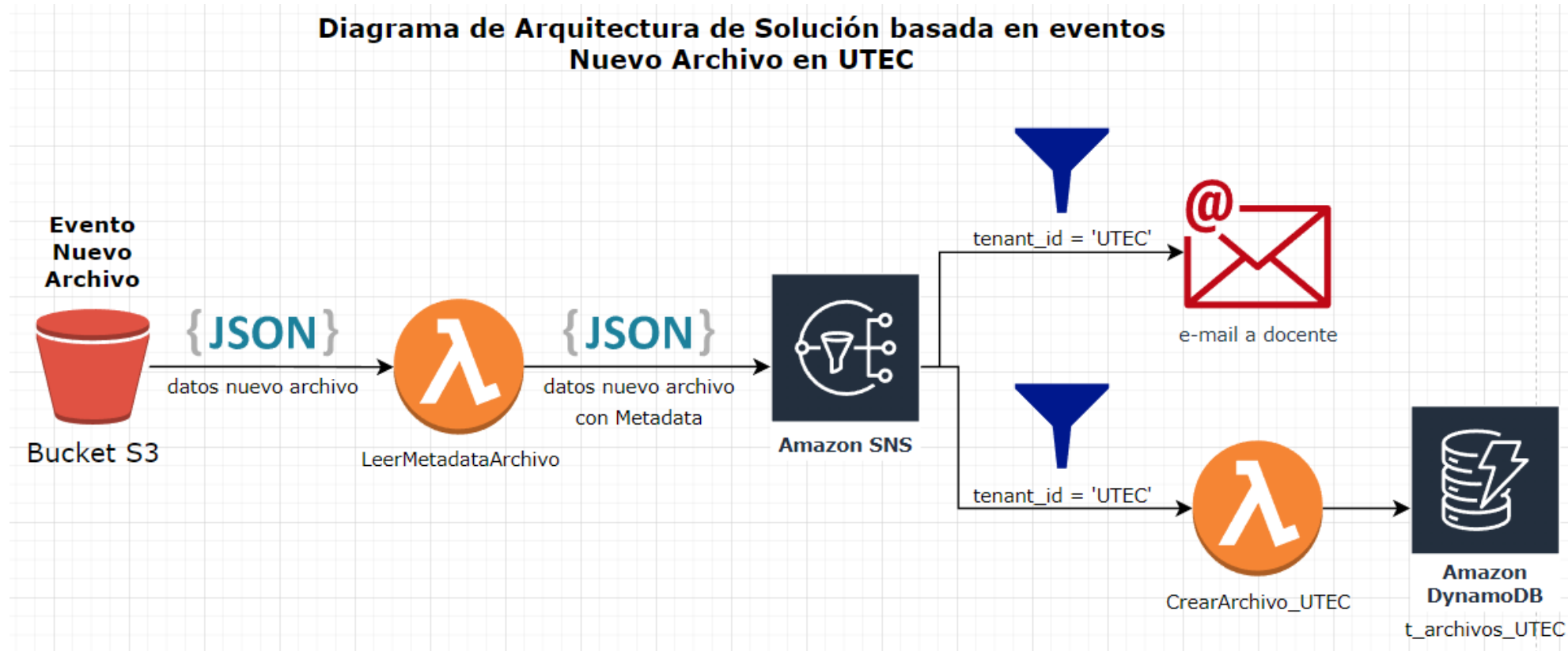
Event-driven architecture

1. Objetivo del taller 1
2. **Ejercicio 1: Evento Nuevo Archivo en UTEC**
3. Ejercicio 2: Ejercicio propuesto
4. Cierre

Event-driven architecture

Ejercicio 1 - Evento Nuevo Archivo en UTEC





Implemente la siguiente arquitectura para procesar el evento “Nuevo Archivo en UTEC”



Event-driven architecture

Ejercicio 1 - Evento Nuevo Archivo en UTEC

Paso 1: Crear un bucket S3 con nombre único con la siguiente estructura de carpetas:

Amazon S3 > Buckets > gcolchado3	
Nombre ▲	Tipo ▼
 universidades/	Carpeta
 UNIV2/	Carpeta
 UNIV3/	Carpeta
 UTEC/	Carpeta

Event-driven architecture

Ejercicio 1 - Evento Nuevo Archivo en UTEC

Paso 2: Crear un lambda “LeerMetadataArchivo” con este código fuente.

```
import json

def lambda_handler(event, context):
    print(event) # Revisar en Cloud Watch
    # TODO implement
    return {
        'statusCode': 200,
        'body': json.dumps('Hello from Lambda!')
    }
```

Paso 3: Configure en bucket una notificación de evento hacia el lambda

The screenshot shows the Amazon S3 console interface for configuring an event notification. The breadcrumb trail is 'Amazon S3 > Buckets > gcolchado3'. The bucket name 'gcolchado3' is displayed with an 'Info' link. Below the bucket name are three tabs: 'Objetos', 'Propiedades' (which is selected and highlighted in yellow), and 'Permisos'. A button labeled 'Crear notificación de eventos' is visible. Below this is a section titled 'Crear notificación de eventos' with an 'Info' link. The text below the title reads: 'Para habilitar las notificaciones, primero debe agregar una configuración que Amazon S3 publique y los destinos a los que desea que'. To the right of the main configuration area, there are several settings: 'Tipos de eventos' with a description and a checkbox for 'Todos los eventos de creación de objetos' (checked) with the event type 's3:ObjectCreated:*'; 'Destino' with a dropdown menu set to 'Función Lambda' and a description 'Ejecute un script de función Lambda basado en eventos de S3.'; 'Especificar Función Lambda' with a dropdown menu set to 'Elija uno de los Funciones de Lambda'; and 'Función Lambda' with a text input field containing 'LeerMetadataArchivo' (highlighted in yellow). At the bottom right, there is an orange button labeled 'Guardar cambios'.

Amazon S3 > Buckets > gcolchado3

gcolchado3 Info

Objetos Propiedades Permisos

Crear notificación de eventos

Crear notificación de eventos Info

Para habilitar las notificaciones, primero debe agregar una configuración que Amazon S3 publique y los destinos a los que desea que

Configuración general

Nombre del evento

NuevoArchivo

Tipos de eventos

Especifique al menos un evento sobre el que desea recibir eventos, o puede elegir uno o más eventos individuales.

☒ Todos los eventos de creación de objetos
s3:ObjectCreated:*

Destino

Elija un destino para publicar el evento. Más información

☒ Función Lambda
Ejecute un script de función Lambda basado en eventos de S3.

Especificar Función Lambda

☒ Elija uno de los Funciones de Lambda

☐ Escriba el ARN de Función Lambda

Función Lambda


LeerMetadataArchivo

Guardar cambios

Event-driven architecture

Ejercicio 1 - Evento Nuevo Archivo en UTEC

Paso 4: Suba un archivo al directorio y valide logs del lambda “LeerMetadataArchivo” en Cloud Watch

Amazon S3 > Buckets > gcolchado3 > universidades/ > UTEC/			
Nombre ▲	Tipo ▼	Última modificación ▼	Tamaño ▼
 archivo01.txt	txt	17 Oct 2022 5:28:41 PM -05	6.0 B

Cloud Watch


```
{'Records': [{'eventVersion': '2.1', 'eventSource': 'aws:s3', 'awsRegion': 'us-east-1', 'eventTime': '2022-10-17T22:28:40.885Z', 'eventName': 'ObjectCreated:Put', 'userIdentity': {'principalId': 'AWS:AROAWQSDEEHPHKPNON4UA:user2087205=Test_Student'}, 'requestParameters': {'sourceIPAddress': '148.102.115.41'}, 'responseElements': {'x-amz-request-id': 'R8TNAZMC5FZA7MC7', 'x-amz-id-2': 'GjFjKV9/SgpLNzZwVwwaUVp6/Te8waoNX6cWYal1GcekO+KcDTRWcZS2/N1b1GKqHwHlasJEEenxZloldso769X2koVSL2iv8'}, 's3': {'s3SchemaVersion': '1.0', 'configurationId': 'NuevoArchivo', 'bucket': {'name': 'gcolchado3', 'ownerIdentity': {'principalId': 'A24RZ2Q7S369PD'}}, 'arn': 'arn:aws:s3:::gcolchado3', 'object': {'key': 'universidades/UTEC/archivo01.txt', 'size': 6, 'eTag': '5bc8c567a89112d5f408a8af4f17970d', 'sequencer': '00634DD718D7C9D664'}}}]}
```


Event-driven architecture

Ejercicio 1 - Evento Nuevo Archivo en UTEC

Paso 5: Modifique lambda “LeerMetadataArchivo” para obtener metadata identificada en Cloud Watch y pruebe con archivo

Amazon S3 > Buckets > gcolchado3 > universidades/ > UTEC/

Nombre ▲	Tipo ▼	Última modificación ▼	Tamaño ▼
 archivo01.txt	txt	17 Oct 2022 6:25:29 PM -05	6.0 B

```
import json

def lambda_handler(event, context):
    print(event) # Revisar en Cloud Watch
    # Entrada (json)
    archivo_id = event['Records'][0]['s3']['object']['key']
    tenant_id = archivo_id.split('/')[1] # UTEC, UNIV1, etc.
    archivo_last_modified = event['Records'][0]['eventTime']
    archivo_size = event['Records'][0]['s3']['object']['size']
    bucket_name = event['Records'][0]['s3']['bucket']['name']
    archivo = {
        'tenant_id': tenant_id,
        'archivo_id': archivo_id,
        'archivo_datos': {
            'last_modified': archivo_last_modified,
            'size': archivo_size,
            'bucket_name': bucket_name
        }
    }
    print(archivo)
    # TODO implement
    return {
        'statusCode': 200,
        'body': json.dumps('Hello from Lambda!')
    }
```

Cloud Watch

```
{'tenant_id': 'UTEC', 'archivo_id':
'universidades/UTEC/archivo01.txt',
'archivo_datos': {'last_modified':
'2022-10-17T23:25:28.543Z', 'size':
6, 'bucket_name': 'gcolchado3'}}
```

Event-driven architecture

Ejercicio 1 - Evento Nuevo Archivo en UTEC

Paso 6: Cree un tema
“TemaNuevoArchivo”

Paso 7: Modifique el lambda
“LeerMetadataArchivo” para publicar
en el tema “TemaNuevoArchivo”

Paso 8: Cree una suscripción de
correo electrónico al tema
“TemaNuevoArchivo” con filtro
tenant_id = ‘UTEC’ y confirme el
enlace en su correo.

Paso 9: Suba un archivo al bucket
(universidades/UTEC/) y verifique si
le llegó el correo electrónico.

Paso 10: Suba un archivo al bucket
(universidades/UNIV2/) y verifique
que **no** le llegue correo electrónico.

```
import json
import boto3

def lambda_handler(event, context):
    # Entrada (json)
    archivo_id = event['Records'][0]['s3']['object']['key']
    tenant_id = archivo_id.split('/')[1] # UTEC, UNIV1, etc.
    archivo_last_modified = event['Records'][0]['eventTime']
    archivo_size = event['Records'][0]['s3']['object']['size']
    bucket_name = event['Records'][0]['s3']['bucket']['name']
    archivo = {
        'tenant_id': tenant_id,
        'archivo_id': archivo_id,
        'archivo_datos': {
            'last_modified': archivo_last_modified,
            'size': archivo_size,
            'bucket_name': bucket_name
        }
    }

    # Publicar en SNS
    sns_client = boto3.client('sns')
    response_sns = sns_client.publish(
        TopicArn = 'arn:aws:sns:us-east-1:447891120606:TemaNuevoArchivo',
        Subject = 'Nuevo Archivo',
        Message = json.dumps(archivo),
        MessageAttributes = {
            'tenant_id': {'DataType': 'String', 'StringValue': tenant_id}
        }
    )

    # TODO implement
    return {
        'statusCode': 200,
        'body': response_sns
    }
```

Nuevo Archivo Recibidos



AWS Notifications 18:48
para mí ▾



```
{"tenant_id": "UTEC", "archivo_id":
"universidades/UTEC/archivo01.txt", "archivo_datos":
{"last_modified": "2022-10-17T23:48:49.916Z", "size": 6,
"bucket_name": "gcolchado3"}}
```

Event-driven architecture

Ejercicio 1 - Evento Nuevo Archivo en UTEC

Paso 11: Crear tabla DynamoDB “t_archivos_UTEC”

Nombre	Estado	Clave de partición	Clave de ordenación
t_archivos_UTEC	Activo	tenant_id (S)	archivo_id (S)

Paso 12: Crear lambda “CrearArchivo_UTEC”

Paso 13: Cree una suscripción del lambda “CrearArchivo_UTEC” al tema “TemaNuevoArchivo” con filtro tenant_id = ‘UTEC’

Paso 14: Suba un archivo al bucket (universidades/UTEC/) y verifique si graba registro en tabla.

Paso 15: Suba un archivo al bucket (universidades/UNIV2/) y verifique que **no** grabe registro en tabla

```
import json
import boto3
```

```
def lambda_handler(event, context):
    # Entrada (json)
    print(event) # Revisar en CloudWatch
    archivo_json = json.loads(event['Records'][0]['Sns']['Message'])
    # Proceso
    dynamodb = boto3.resource('dynamodb')
    table = dynamodb.Table('t_archivos_UTEC')
    archivo = {
        'tenant_id': archivo_json['tenant_id'],
        'archivo_id': archivo_json['archivo_id'],
        'archivo_datos': archivo_json['archivo_datos']
    }
    print(archivo) # Revisar en CloudWatch
    response = table.put_item(Item=archivo)
    # Salida (json)
    return {
        'statusCode': 200,
        'response': response
    }
```

tenant_id	archivo_id	archivo_datos
UTEC	universidades/UTEC/archivo02.txt	{ "size": { "N": "6" }, "last_modified": { "S": "2022-10-18T01:11:34.552Z" }, "bucket_name": { "S": "gcolchado3" } }

Contenido

Event-driven architecture

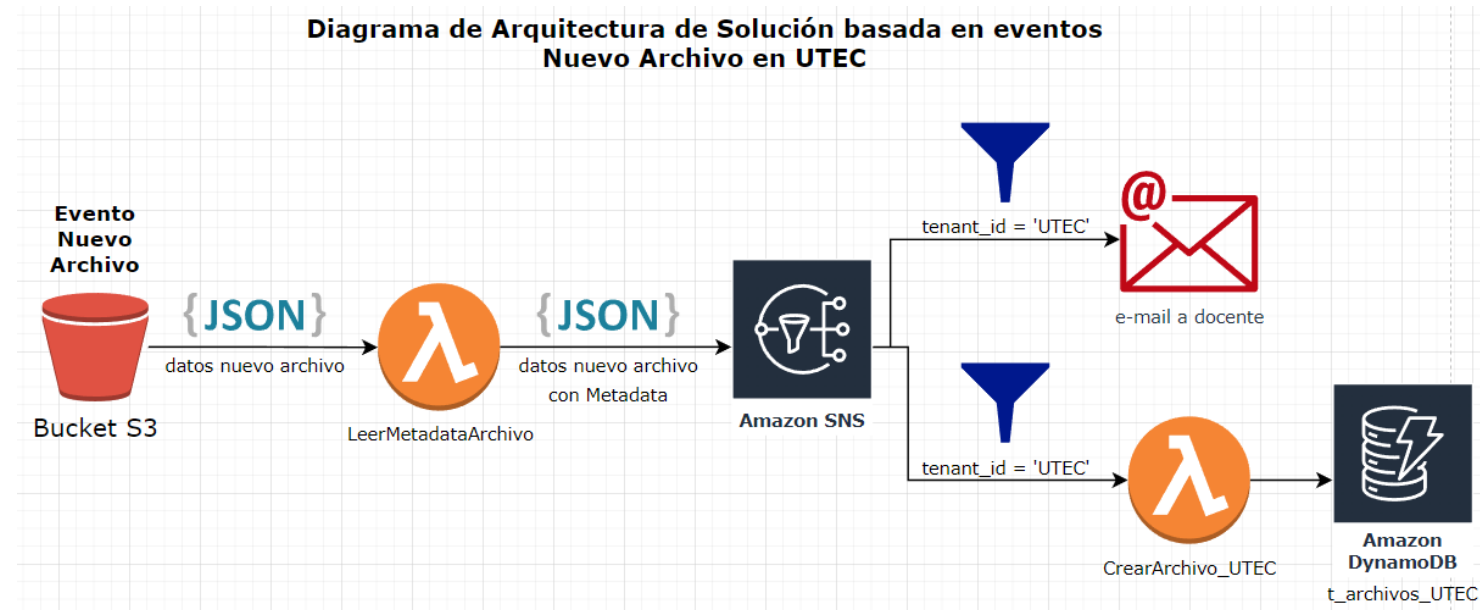
1. Objetivo del taller 1
2. Ejercicio 1: Evento Nuevo Archivo en UTEC
3. **Ejercicio 2: Ejercicio propuesto**
4. Cierre

Event-driven architecture

Ejercicio 2- Propuesto

Se le solicita que pueda identificar en la metadata del archivo el **código del curso** y el **código de alumno**. Diseñe e implemente los cambios. Se requiere que se grabe esa metadata en la tabla `t_archivos_UTE`.

Presente la solución en el padlet indicado por el docente con la evidencia.



Contenido

Event-driven architecture

1. Objetivo del taller 1
2. Ejercicio 1: Evento Nuevo Archivo en UTEC
3. Ejercicio 2: Ejercicio propuesto
4. **Cierre**

Cierre:

Event-driven architecture - Qué aprendimos?

- Diseño e implementación de una Arquitectura de Solución basada en eventos con el servicio “SNS - Simple Notification Service”

Gracias

Elaborado por docente: Geraldo Colchado