



Universidad de Guayaquil
Facultad de Ciencias Matemáticas y Físicas
Carrera de Ingeniería en Sistemas Computacionales

Asignatura:

Sistemas Operativos Distribuidos

Docente:

Ing. Cristopher Crespo León, Mgs.

Integrantes:

Arteaga Burgos Anthony
Baque Chompol Leonidas Edward
Naranjo Iñiga Pedro Anthony
Pilay Lino Jennifer Lisbeth
Ramírez Castro Arlin Damaris
Seme Marquez Jair Josue
Vaccaro Cedillo Gean Carlo
Yumaglla Guagcha Franklin David

Paralelo:

ISI-S-NO-7-5

Año Lectivo

2021-2022 CII

Tabla de Contenido

Presentación con prezi - Vínculo del enlace.....	3
Diagramas en lucid chart (Un gráfico para introducir conceptos o que explique los componentes que intervienen en una solución) - Vínculo del enlace.....	3
Implementación de colas con Amazon Kinesis – Cloud Watch.....	3
Beneficios	3
Servicios.....	4
Amazon Kinesis Video Streams	4
Funcionamiento	4
Kinesis Data Streams.....	5
Funcionamiento	5
Amazon Kinesis Data Firehose	6
Funcionamiento	7
Amazon Kinesis Data Analytics.....	8
Funcionamiento	8
Implementación	8
Implementación de Kinesis con Lambda – Cloud Watch	15
Implementation con AWS Serverless Application Model (SAM) – Cloud Watch	15
Implementación de Cognito – Api Gateway	15
Implementación de Cognito – Lambda	19
Implementación de Cognito – Cloud Watch	19
Usando Amazon Kinesis y Data Firehose	19
Buscar ejemplos en internet donde se usen implementaciones de Kinesis con Lambda	19
Realizar una estimación de costos de tener habilitado el servicio de manera mensual, usando un estimado en la calculadora.	19
Anexos.....	20

Presentación con prezi - Vínculo del enlace



Link: https://prezi.com/p/edit/_rmtxvzcbud4/

Diagramas en lucid chart (Un gráfico para introducir conceptos o que explique los componentes que intervienen en una solución) - Vínculo del enlace

Implementación de colas con Amazon Kinesis – Cloud Watch

Amazon Kinesis

Amazon Kinesis facilita la recopilación, el procesamiento y el análisis de datos de streaming en tiempo real para obtener datos de manera oportuna y reaccionar rápidamente ante información nueva. Amazon Kinesis ofrece capacidades clave para procesar de manera rentable datos de streaming a cualquier escala, además de la flexibilidad para elegir las herramientas más adecuadas para los requisitos de su aplicación. Con Amazon Kinesis, se puede incorporar datos en tiempo real, como videos, audios, registros de aplicaciones, secuencias de clics de sitios web y datos de telemetría de IoT para aprendizaje automático, análisis y otras aplicaciones. Amazon Kinesis le permite procesar y analizar datos a medida que se reciben y responder instantáneamente en vez de tener que esperar a que los datos se recopilen antes de que el procesamiento pueda comenzar.

Beneficios

Tiempo real

Amazon Kinesis le permite incorporar, guardar y procesar datos de streaming en tiempo real para que pueda obtener información en segundos o minutos, en lugar de horas o días.

Completamente administrado

Amazon Kinesis está completamente administrado y se ejecuta en sus aplicaciones de streaming sin que usted tenga que administrar infraestructura.

Escalabilidad

Amazon Kinesis puede gestionar cualquier volumen de datos de streaming y procesar datos de cientos de miles de orígenes con latencias muy bajas.

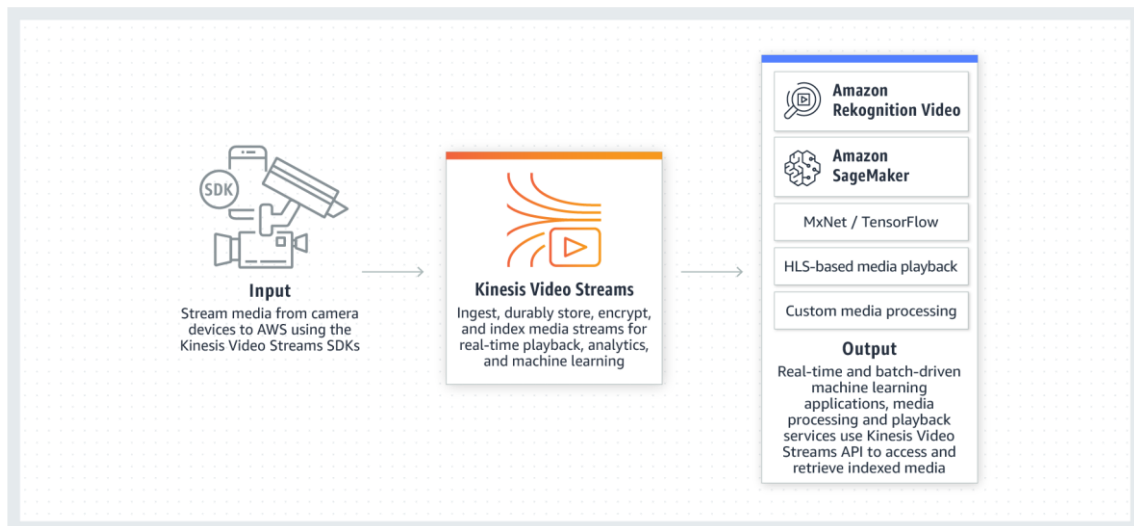
Servicios

Amazon Kinesis Video Streams

Amazon Kinesis Video Streams facilita la transmisión segura de videos desde dispositivos conectados a AWS para tareas de análisis, aprendizaje automático (ML), reproducción y otros procesos. Kinesis Video Streams aprovisiona automáticamente y escala de manera elástica toda la infraestructura necesaria para incorporar datos de las transmisiones de vídeo de millones de dispositivos. Almacena, cifra e indexa de forma duradera datos de videos en transmisiones y permite obtener acceso a los datos mediante API fáciles de usar. Kinesis Video Streams le permite reproducir videos para visualizaciones en directo y bajo demanda, y crear rápidamente aplicaciones que aprovechan la visión artificial y el análisis de videos a través de la integración con Amazon Rekognition Video, y bibliotecas para marcos de ML, como Apache MxNet, TensorFlow y OpenCV. Kinesis Video Streams también admite WebRTC, un proyecto de código abierto que permite la transmisión y la interacción de medios en tiempo real entre los navegadores web, las aplicaciones móviles y los dispositivos conectados mediante API simples. Los usos típicos incluyen videollamada y transmisión de medios entre pares.

Funcionamiento

Obtenga, procese y almacene transmisiones de medios para tareas de reproducción, analíticas y de aprendizaje automático.



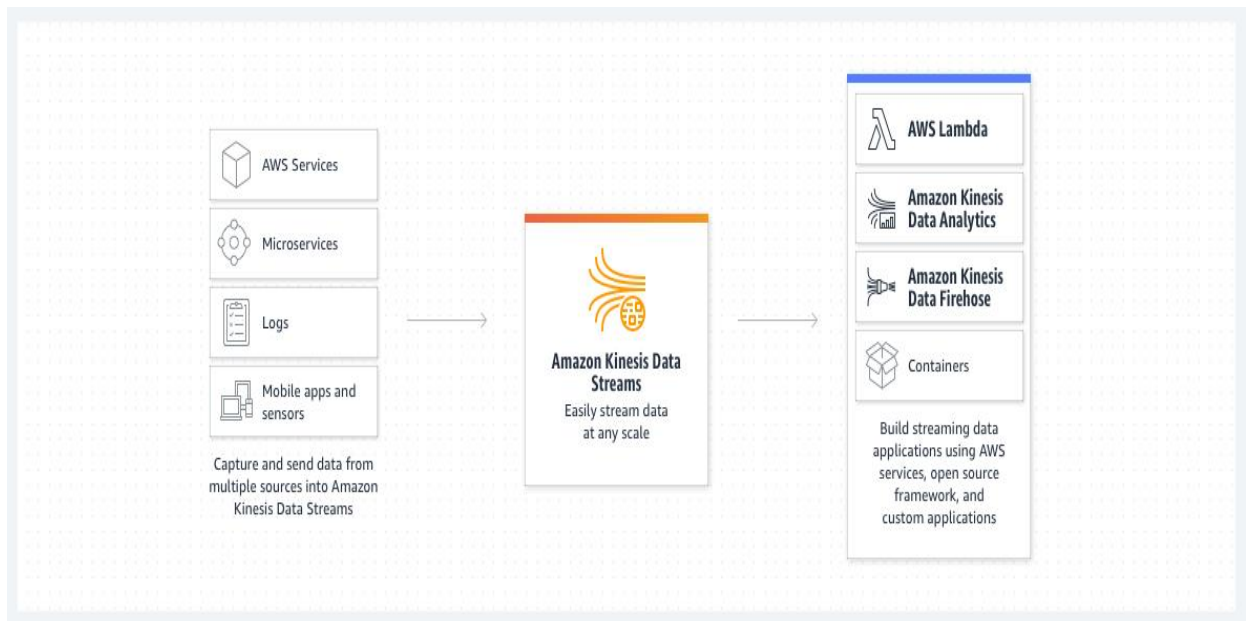
Kinesis Data Streams

Se puede utilizar Amazon Kinesis Data Streams para recopilar y procesar grandes flujos de registros de datos en tiempo real. Puede crear aplicaciones de procesamiento de datos, conocidas como aplicaciones Kinesis Data Streams. Una aplicación típica de Kinesis Data Streams lee datos de un flujo de datos como registros de datos. Estas aplicaciones pueden usar Kinesis Client Library y pueden ejecutarse en instancias de Amazon EC2. Puede enviar los registros procesados a los paneles, usarlos para generar alertas, cambiar dinámicamente las estrategias de precios y publicidad, o enviar datos a una variedad de otros servicios de AWS.

Amazon Kinesis Data Streams es un servicio de datos de *streaming* sin servidor completamente administrado que facilita elásticamente la captura y el almacenamiento de los registros, los eventos, las secuencias de clics y otras formas de datos de *streaming* en tiempo real. Kinesis Data Streams aplica un método de pago por uso muy sencillo. No hay costos iniciales o mínimos y solo paga los recursos que usa.

Funcionamiento

Amazon Kinesis Data Streams es un servicio de datos de streaming sin servidor que hace que sea fácil capturar, procesar y almacenar flujos de datos a cualquier escala.



Amazon Kinesis Data Firehose

Amazon Kinesis Data Firehose ofrece la manera más sencilla de cargar datos de streaming en almacenes de datos y herramientas de análisis. Kinesis Data Firehose es un servicio completamente administrado que facilita el registro, la transformación y la carga de volúmenes masivos, puede registrar, transformar y entregar datos de streaming desde cientos de miles de fuentes a Amazon S3, Amazon Redshift, Amazon OpenSearch Service (sucesor de Amazon Elasticsearch Service), Kinesis Data Analytics, puntos de enlace HTTP genéricos y proveedores de servicios como Datadog, New Relic, MongoDB y Splunk. Esto permite llevar a cabo análisis y obtener información casi en tiempo real. Se trata de un servicio completamente administrado cuya escala se ajusta de manera automática para adaptarse al nivel de procesamiento de sus datos. Además, no requiere administración continua. También puede procesar por lotes, comprimir, transformar y cifrar las transmisiones de datos antes de cargarlas a fin de minimizar el volumen de almacenamiento utilizado y aumentar el nivel de seguridad.

La colección de datos permite la captura de múltiples fuentes.

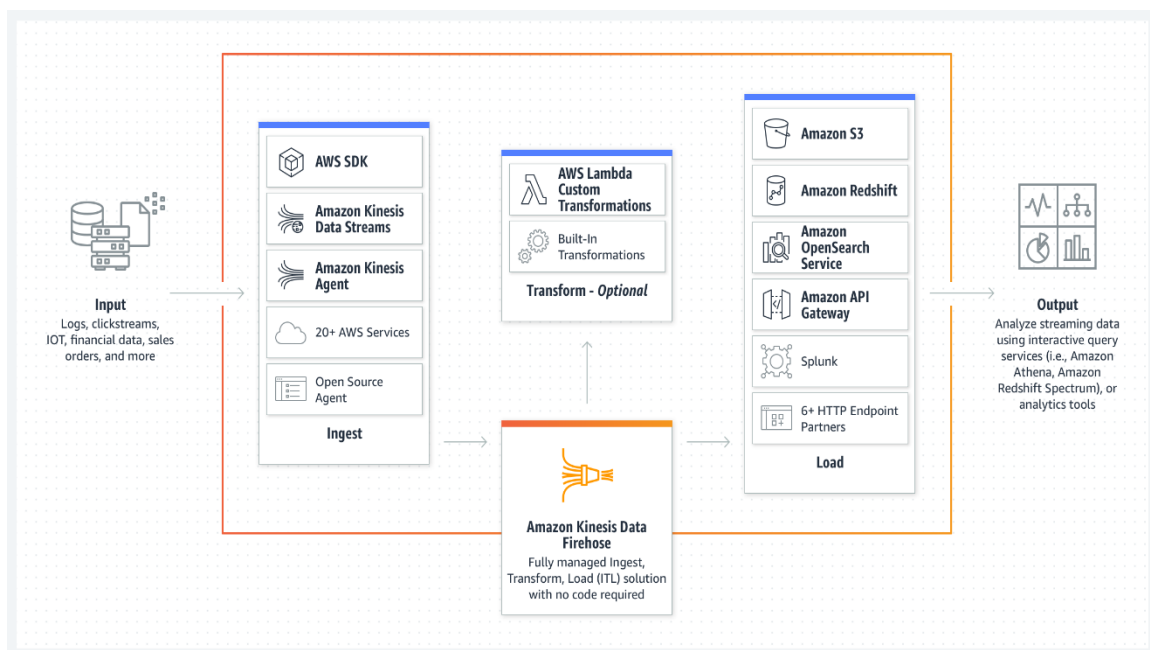


Flujo de entrega

Una transmisión de entrega es la entidad subyacente de Kinesis Data Firehose. Kinesis Data Firehose se utiliza mediante la creación de una transmisión de entrega y el posterior envío de datos a ella.

Funcionamiento

Amazon Kinesis Data Firehose es un servicio de extracción, transformación y carga (ETL) que captura, transforma y entrega de manera fiable datos de streaming en lagos y almacenes de datos y servicios de análisis.



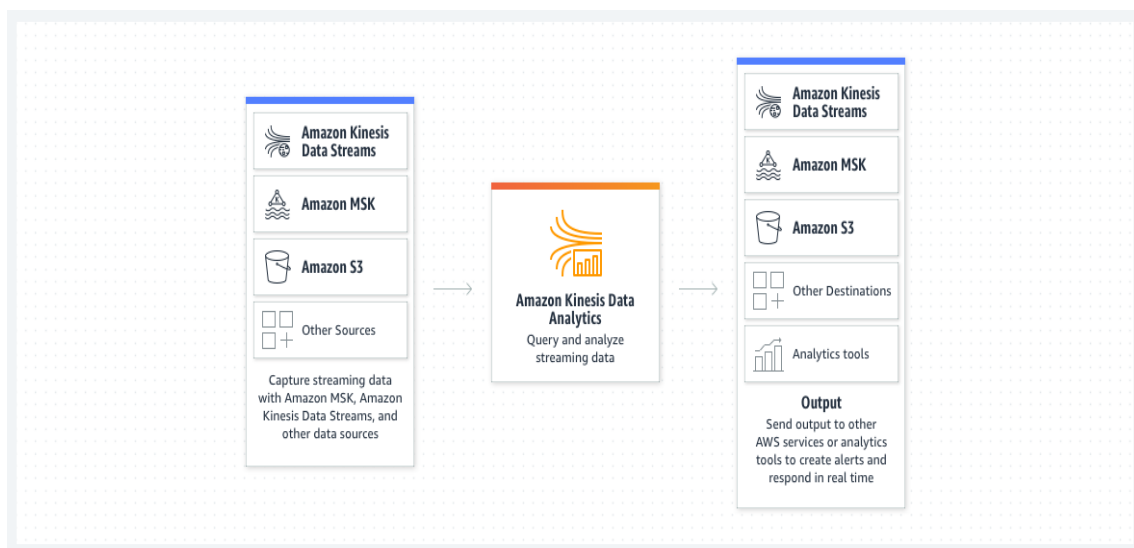
Amazon Kinesis Data Analytics

Ofrece la manera más sencilla de transformar y analizar datos de streaming en tiempo real con Apache Flink. Apache Flink es un marco de código abierto y un motor para procesar transmisiones de datos. Amazon Kinesis Data Analytics reduce la complejidad de desarrollar, administrar e integrar las aplicaciones de Apache Flink con otros servicios de AWS.

Amazon Kinesis Data Analytics se ocupa de todo lo necesario para ejecutar sus aplicaciones de streaming de forma continua y ajusta la escala automáticamente para adaptarse al volumen y rendimiento de los datos de entrada. Con Amazon Kinesis Data Analytics, no hay servidores que administrar, no hay tarifas mínimas ni costos de instalación y solo paga por los recursos que consumen sus aplicaciones de streaming.

Funcionamiento

Amazon Kinesis Data Analytics ofrece la manera más sencilla de transformar y analizar datos de streaming en tiempo real con Apache Flink.



Implementación

Inicie sesión en la consola de administración de AWS y abra la consola de Kinesis en:



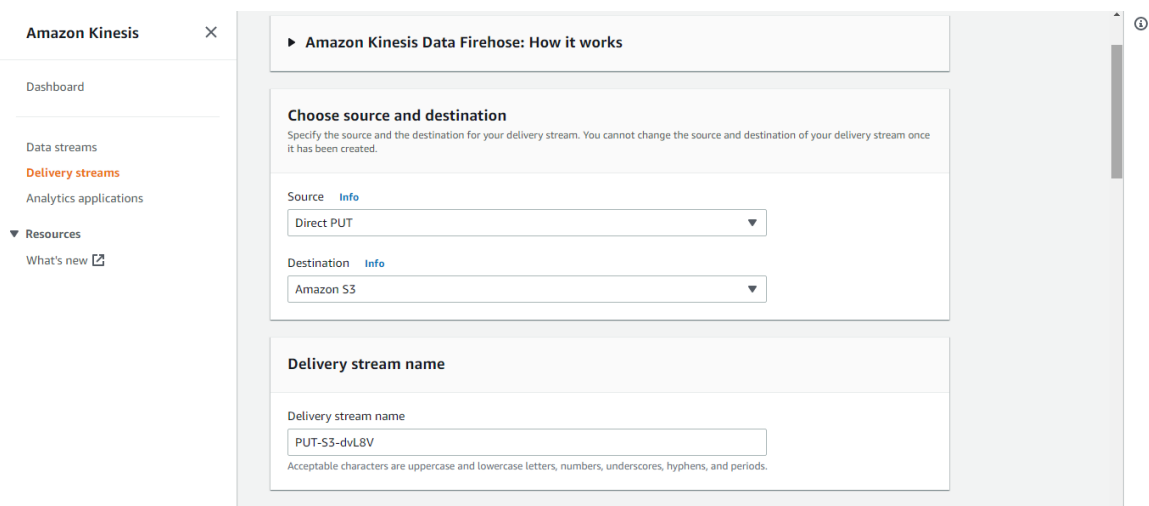
Elija Data Firehose en el panel de navegación. Elija Crear flujo de entrega.

Elija esta opción Direct Put para crear un flujo de entrega de Kinesis Data Firehose en el que las aplicaciones de productores escriban directamente.

Seleccionamos el Destino del flujo de entrega en este caso escogeremos

- Amazon S3

Indicar el nombre de su flujo de entrega de Kinesis Data Firehose.



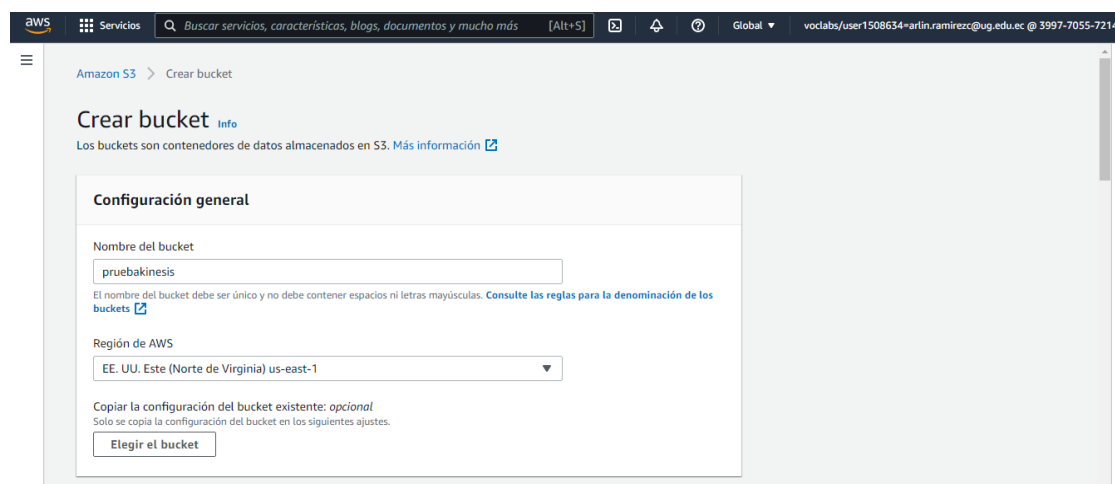
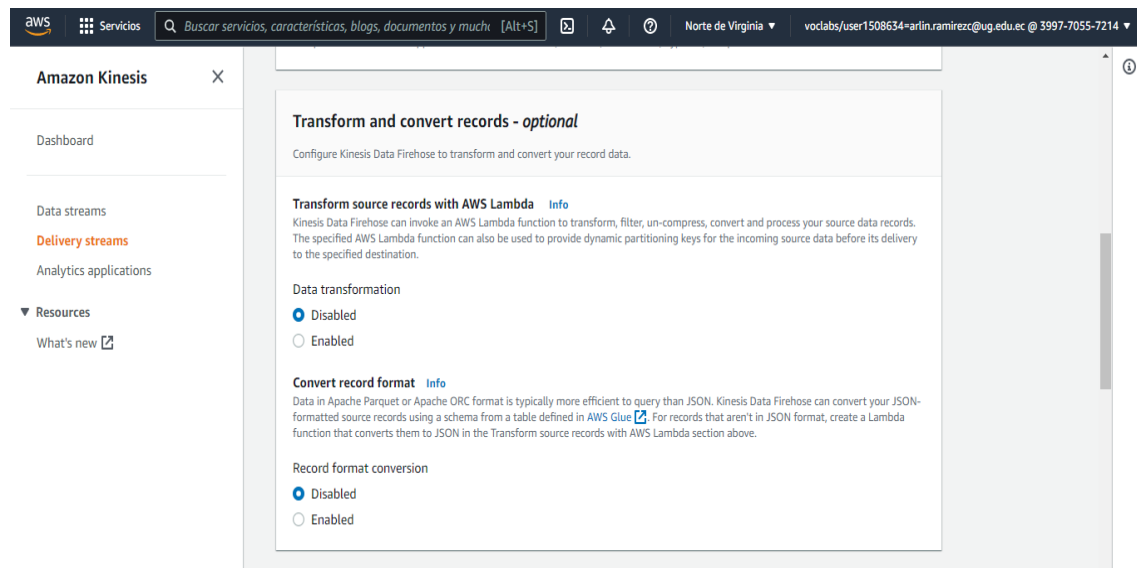
Configure Kinesis Data Firehose para transformar y convertir sus datos de registro.

En la sección Transformar registros de origen con AWS Lambda, proporcione valores para el siguiente campo: Transformación de datos.

Para crear un flujo de entrega de Kinesis Data Firehose que no transforme los datos entrantes, **elija Deshabilitado**.

En la sección Convertir formato de registro, proporcione valores para el siguiente campo: Conversión de formato de registro.

Para crear un flujo de entrega de Kinesis Data Firehose que no convierta el formato de los registros de datos entrantes, **elija Deshabilitado**.



Configuración de destino para su flujo de entrega.

Debe especificar la siguiente configuración para utilizar Amazon S3 como el destino de su flujo de entrega de Kinesis Data Firehose:

S3 bucket

Elija un depósito de S3 de su propiedad donde se deben entregar los datos de transmisión. Puede crear un nuevo depósito S3 o elegir uno existente.

The screenshot shows the 'Destination settings' page in the Amazon Kinesis console. The left sidebar contains navigation links: Dashboard, Data streams, Delivery streams (highlighted), Analytics applications, Resources, and What's new. The main content area is titled 'Destination settings' and includes an 'Info' icon. It contains the following sections:

- S3 bucket:** A text input field containing 's3://pruebakinesis', a 'Browse' button, and a 'Create' button with an external link icon. Below the input is the text 'Format: s3://bucket'.
- Dynamic partitioning:** A section with an 'Info' icon. It explains that dynamic partitioning enables creating targeted data sets by partitioning streaming S3 data based on partitioning keys. It includes a note that dynamic partitioning is only enabled when creating a new delivery stream and that enabling it incurs additional costs. There are two radio buttons: 'Disabled' (selected) and 'Enabled'.
- S3 bucket prefix - optional:** A section explaining that Kinesis Data Firehose appends the prefix 'YYYY/MM/dd/HH' (in UTC) to the data it delivers to Amazon S3. It allows overriding this default by specifying a custom prefix. There is a text input field with the placeholder 'Enter a prefix'.
- S3 bucket error output prefix - optional:** A section explaining that you can specify an S3 bucket error output prefix to be used in error conditions. This prefix can include expressions for Kinesis Data Firehose.

Kinesis Data Firehose almacena en búfer los datos entrantes antes de enviarlos al destino especificado. El tamaño de búfer recomendado para el destino varía según el proveedor de servicios.

The screenshot shows the 'Buffer hints, compression and encryption' page in the Amazon Kinesis console. The left sidebar is the same as the previous screenshot. The main content area is titled 'Buffer hints, compression and encryption' and includes an 'Info' icon. It contains the following sections:

- Buffer hints, compression and encryption:** A section explaining that the fields below are pre-populated with recommended default values for S3. Pricing may vary depending on storage and request costs.
- S3 buffer hints:** A section explaining that Kinesis Data Firehose buffers incoming records before delivering them to your S3 bucket. Record delivery is triggered once the value of either of the specified buffering hints is reached.
- Buffer size:** A section explaining that the higher buffer size may be lower in cost with higher latency. The lower buffer size will be faster in delivery with higher cost and less latency. There is a text input field containing '1' and a 'MiB' unit selector. Below the input is the text 'Minimum: 1 MiB, maximum: 128 MiB. Recommended: 5 MiB.'
- Buffer interval:** A section explaining that the higher interval allows more time to collect data and the size of data may be bigger. The lower interval sends the data more frequently and may be more advantageous when looking at shorter cycles of data activity. There is a text input field containing '60' and a 'seconds' unit selector. Below the input is the text 'Minimum: 60 seconds, maximum: 900 seconds. Recommended: 300 seconds.'
- S3 compression and encryption:** A section explaining that Kinesis Data Firehose can compress records before delivering them to your S3 bucket. Compressed records can also be encrypted in the S3 bucket using an AWS Key Management Service (KMS) master key.
- Compression for data records:** A section explaining that Kinesis Data Firehose can compress records before delivering them to your S3 bucket. There are three radio buttons: 'Disabled' (selected), 'GZIP', and 'Snappy'.

aws

Servicios

Buscar servicios, características, blogs, documentos y mucho más. [Alt+S]

Norte de Virginia

voclabs/user1508634-arlin.ramirez@ug.edu.ec @ 3997-7055-7214

Amazon Kinesis

Dashboard

Data streams

Delivery streams

Analytics applications

Resources

What's new

60 seconds

Minimum: 60 seconds, maximum: 900 seconds. Recommended: 300 seconds.

S3 compression and encryption

Kinesis Data Firehose can compress records before delivering them to your S3 bucket. Compressed records can also be encrypted in the S3 bucket using an AWS Key Management Service (KMS) master key.

Compression for data records

Kinesis Data Firehose can compress records before delivering them to your S3 bucket.

Disabled

GZIP

Snappy

Zip

Hadoop-Compatible Snappy

Encryption for data records

Compressed record gets encrypted in the S3 bucket using a KMS master key.

Disabled

Enabled

Advanced settings

Server-side encryption disabled; error logging enabled; IAM role KinesisFirehoseServiceRole-PUT-S3-dvL8V-us-east-1-1645902662747; no tags.

Comentarios

Español

© 2022, Amazon Web Services, Inc. o sus filiales.

Privacidad

Términos

Preferencias de cookies

aws

Servicios

Buscar servicios, características, blogs, documentos y mucho más. [Alt+S]

Norte de Virginia

voclabs/user1508634-arlin.ramirez@ug.edu.ec @ 3997-7055-7214

Amazon Kinesis

Dashboard

Data streams

Delivery streams

Analytics applications

Resources

What's new

Server-side encryption

You can use AWS Key Management Service (KMS) to create and manage Customer Master Keys (CMK) and to control the use of encryption across a wide range of AWS services in your applications.

Enable server-side encryption for source records in delivery stream

Amazon CloudWatch error logging

Choose Enabled if you want Kinesis Data Firehose to log record delivery errors to CloudWatch Logs.

Disabled

Enabled

Permissions

Kinesis Data Firehose uses this IAM role for all the permissions that the delivery stream needs. To specify different roles for the different permissions, use the API or the CLI.

Create or update IAM role KinesisFirehoseServiceRole-PUT-S3-dvL8V-us-east-1-1645902662747

Creates a new role or updates an existing one and adds the required policies to it, and enables Kinesis Data Firehose to assume it.

Choose existing IAM role

The role that you choose must have policies that include the permissions that Kinesis Data Firehose needs.

Tags

You can add tags to organize your AWS resources, track costs, and control access.

Key

Value - optional

desarrollo.kinesis

bolsa

Remove

Add new tag

You can add up to 49 more tags.

Amazon Kinesis

Dashboard

Data Streams

Data Firehose

Data Analytics

Video Streams

External resources

What's new

Kinesis Data Firehose delivery streams

Kinesis Data Firehose delivery streams continuously collect, transform, and load streaming data into the destinations that you specify.

Successfully created delivery stream kinesia1

Next, send records directly to the delivery stream using the Amazon Kinesis Agent or the Kinesis Data Firehose API using the AWS SDK, or send records from AWS IoT, CloudWatch Logs, or CloudWatch Events. Learn more

Test with demo data

Delete

Create delivery stream

Find delivery streams

< 1 >

Name	Status	Creation time	Source	Data transformation	Destination
kinesia1	Active	2021-07-12T13:38+0200	Direct PUT and other sources	Disabled	Amazon S3 redait.kinesis.bolsa.1

Amazon Kinesis

Kinesis Data Firehose delivery streams > kinesis1

Dashboard

Data Streams

Data Firehose

Data Analytics

Video Streams

External resources

What's new

kinesis1


Delete delivery stream

▼ Test with demo data


This test runs a script in your browser to put demo data in your Kinesis Data Firehose delivery stream, which sends to your S3 destination. The format of the demo data is {"ticker_symbol":"QKZ", "sector":"HEALTHCARE", "change":0.05, "price":84.51}

Step 1

Start sending demo data to your delivery stream. If you already have data streaming to this destination, demo data is sent along with your source records.

 Demo data is being sent to your delivery stream

If the test is successful, you will see demo records in this destination [redait.kinesis.bolsa.1](#)

 Sending demo data

Amazon S3

Buckets

Puntos de acceso

Puntos de acceso del objeto

Lambda

Operaciones por lotes

Analizador de acceso para S3

Configuración de bloqueo de acceso público correspondiente a esta cuenta

Storage Lens

Paneles

Configuración de AWS Organizations


datos/

Copiar URI de S3

Objetos Propiedades

Objetos (1)

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](#)

 Copiar URI de S3 Copiar URL Descargar Abrir Eliminar

Acciones ▼ Crear carpeta Cargar

Buscar objetos por prefijo

<input type="checkbox"/>	Nombre	Tipo	Última modificación	Tamaño	Clase de almacenamiento
<input type="checkbox"/>	2021/	Carpeta	-	-	-

Amazon S3

Buckets

Puntos de acceso

Puntos de acceso del objeto

Lambda

Operaciones por lotes

Analizador de acceso para S3

Configuración de bloqueo de acceso público correspondiente a esta cuenta

Storage Lens

Paneles

Configuración de AWS Organizations



kinesis1-1-2021-07-12-11-42-13-62e4dbd6-1140-49ab-9d96-9f793a8fd044

Info

Copiar URI de S3 Descargar Abrir Acciones de objetos

Propiedades Permisos Versiones

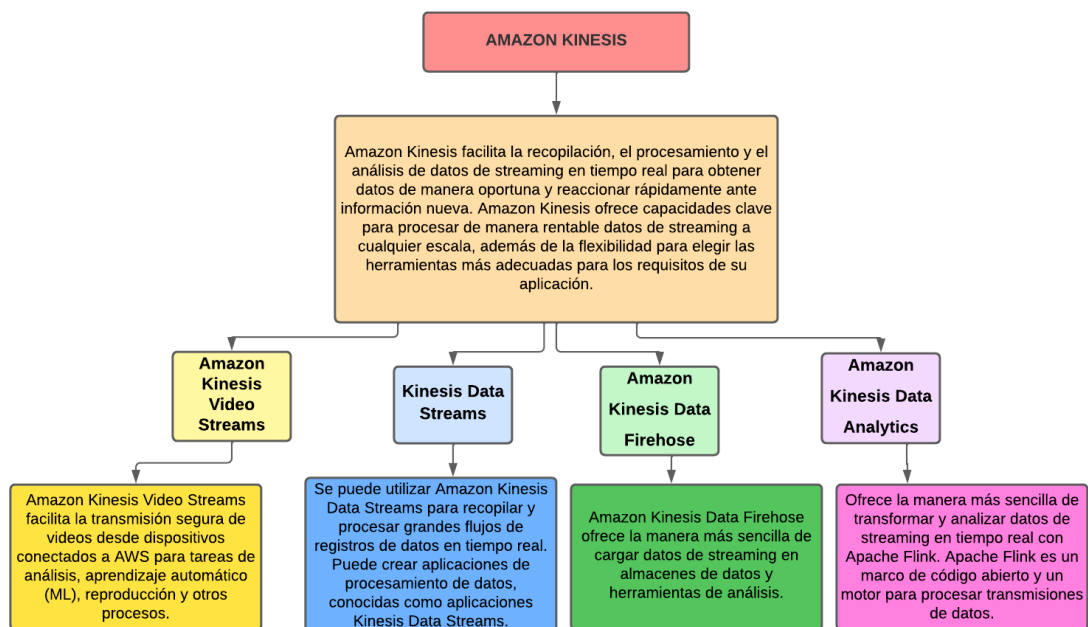
Información general sobre el objeto

Propietario	antonioruizm	URI DE S3	 s3://redait.kinesis.bolsa.1/datos/2021/07/12/11/kinesis1-1-2021-07-12-11-42-13-62e4dbd6-1140-49ab-9d96-9f793a8fd044
Región de AWS	EE. UU. Este (Norte de Virginia) us-east-1	Nombre de recurso de Amazon (ARN)	 arn:aws:s3::redait.kinesis.bolsa.1/datos/2021/07/12/11/kinesis1-1-2021-07-12-11-42-13-62e4dbd6-1140-49ab-9d96-9f793a8fd044
Última modificación	12 Jul 2021 1:43:48 PM CEST		
Tamaño			

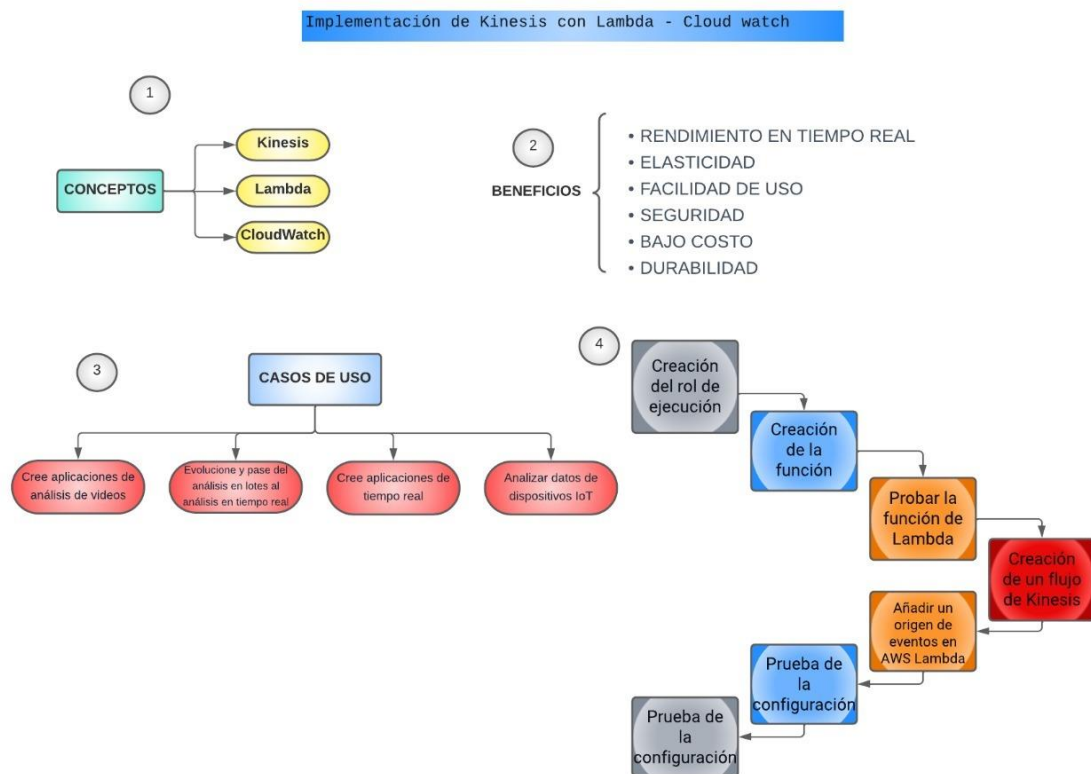
```
kinesis1-1-2021-07-12-11-42-13-62e4dbd6-1140-49ab-9d96-9f793a8fd044 -- Bloqueado

Sin título
kinesis1-1-2021-07-12-11-42-13-62e4dbd6-1140-49ab-9d96-9f793a8fd044

{"ticker_symbol": "NFLX", "sector": "TECHNOLOGY", "change": 1.86, "price": 180.86}, {"ticker_symbol": "TBV", "sector": "HEALTHCARE", "change": -0.14, "price": 190.86}, {"ticker_symbol": "AZL", "sector": "HEALTHCARE", "change": -0.51, "price": 188.61}, {"ticker_symbol": "QXZ", "sector": "FINANCIAL", "change": -4.38, "price": 124.54}, {"ticker_symbol": "WGB", "sector": "FINANCIAL", "change": -1.51, "price": 109.82}, {"ticker_symbol": "TGT", "sector": "RETAIL", "change": -0.24, "price": 65.79}, {"ticker_symbol": "ASD", "sector": "FINANCIAL", "change": -0.66, "price": 65.14}, {"ticker_symbol": "QAZ", "sector": "FINANCIAL", "change": -0.25, "price": 64.89}, {"ticker_symbol": "NGC", "sector": "HEALTHCARE", "change": 0.02, "price": 5.11}, {"ticker_symbol": "TGT", "sector": "RETAIL", "change": -0.63, "price": 67.76}, {"ticker_symbol": "VVS", "sector": "ENERGY", "change": -1.37, "price": 15.05}, {"ticker_symbol": "VVS", "sector": "ENERGY", "change": 0.47, "price": 15.74}, {"ticker_symbol": "VVS", "sector": "ENERGY", "change": 0.68, "price": 16.42}, {"ticker_symbol": "DEG", "sector": "ENERGY", "change": -0.48, "price": 15.33}, {"ticker_symbol": "WSB", "sector": "RETAIL", "change": -0.38, "price": 8.36}, {"ticker_symbol": "SAC", "sector": "ENERGY", "change": 1.87, "price": 157.44}, {"ticker_symbol": "QWE", "sector": "TECHNOLOGY", "change": -4.05, "price": 213.99}, {"ticker_symbol": "AAPL", "sector": "TECHNOLOGY", "change": 0.97, "price": 182.61}, {"ticker_symbol": "XTC", "sector": "HEALTHCARE", "change": -0.88, "price": 112.16}, {"ticker_symbol": "QWE", "sector": "TECHNOLOGY", "change": 2.22, "price": 216.21}, {"ticker_symbol": "TBV", "sector": "HEALTHCARE", "change": 0.52, "price": 189.13}, {"ticker_symbol": "UHN", "sector": "FINANCIAL", "change": 15.86, "price": 559.83}, {"ticker_symbol": "TBV", "sector": "HEALTHCARE", "change": 9.98, "price": 199.11}, {"ticker_symbol": "NGC", "sector": "HEALTHCARE", "change": -0.28, "price": 4.83}, {"ticker_symbol": "CVB", "sector": "TECHNOLOGY", "change": 0.91, "price": 53.73}, {"ticker_symbol": "QWE", "sector": "TECHNOLOGY", "change": 2.2, "price": 218.41}, {"ticker_symbol": "HJV", "sector": "ENERGY", "change": 18.43, "price": 223.43}, {"ticker_symbol": "SAC", "sector": "ENERGY", "change": -4.59, "price": 52.85}, {"ticker_symbol": "ABC", "sector": "RETAIL", "change": -1.85, "price": 23.73}, {"ticker_symbol": "MTT", "sector": "RETAIL", "change": -3.46, "price": 67.05}, {"ticker_symbol": "SLM", "sector": "ENERGY", "change": -4.11, "price": 71.69}, {"ticker_symbol": "ALY", "sector": "ENERGY", "change": 0.83, "price": 89.93}, {"ticker_symbol": "WAS", "sector": "RETAIL", "change": 0.31, "price": 12.82}, {"ticker_symbol": "KIN", "sector": "ENERGY", "change": -0.88, "price": 4.97}, {"ticker_symbol": "BFH", "sector": "RETAIL", "change": 0.25, "price": 17.33}, {"ticker_symbol": "SAC", "sector": "ENERGY", "change": 0.15, "price": 53}, {"ticker_symbol": "NFLX", "sector": "TECHNOLOGY", "change": -0.64, "price": 181.36}, {"ticker_symbol": "AAPL", "sector": "TECHNOLOGY", "change": 0.02, "price": 182.63}, {"ticker_symbol": "XTC", "sector": "HEALTHCARE", "change": -7.49, "price": 104.67}, {"ticker_symbol": "QAZ", "sector": "FINANCIAL", "change": -0.89, "price": 194.57}, {"ticker_symbol": "AZL", "sector": "HEALTHCARE", "change": -0.38, "price": 16.78}, {"ticker_symbol": "AZL", "sector": "HEALTHCARE", "change": -0.89, "price": 16.69}, {"ticker_symbol": "BAC", "sector": "FINANCIAL", "change": 0.44, "price": 15.29}, {"ticker_symbol": "NFLX", "sector": "TECHNOLOGY", "change": 1.14, "price": 182}, {"ticker_symbol": "BAC", "sector": "FINANCIAL", "change": 0.27, "price": 15.56}, {"ticker_symbol": "NFLX", "sector": "TECHNOLOGY", "change": 0.91, "price": 47.7}, {"ticker_symbol": "SLM", "sector": "ENERGY", "change": -5.95, "price": 65.74}, {"ticker_symbol": "BFH", "sector": "RETAIL", "change": -0.21, "price": 17.09}, {"ticker_symbol": "KIN", "sector": "ENERGY", "change": -0.38, "price": 4.59}, {"ticker_symbol": "DFT", "sector": "RETAIL", "change": 0.33, "price": 96.82}, {"ticker_symbol": "SLM", "sector": "ENERGY", "change": 1.65, "price": 67.39}, {"ticker_symbol": "DFT", "sector": "RETAIL", "change": 1.09, "price": 97.91}, {"ticker_symbol": "ALY", "sector": "ENERGY", "change": 5.09, "price": 95.02}, {"ticker_symbol": "SAC", "sector": "ENERGY", "change": 4.67, "price": 157.67}, {"ticker_symbol": "DEG", "sector": "ENERGY", "change": -0.43, "price": 5.43}, {"ticker_symbol": "DEG", "sector": "ENERGY", "change": 0.53, "price": 15.86}, {"ticker_symbol": "AAPL", "sector": "TECHNOLOGY", "change": 1.33, "price": 183.96}, {"ticker_symbol": "TBV", "sector": "HEALTHCARE", "change": -5.24, "price": 193.87}, {"ticker_symbol": "AAPL", "sector": "TECHNOLOGY", "change": 1.82, "price": 185.78}, {"ticker_symbol": "CRM", "sector": "HEALTHCARE", "change": -0.16, "price": 29.46}, {"ticker_symbol": "CRM", "sector": "HEALTHCARE", "change": -1.17, "price": 28.29}, {"ticker_symbol": "BFH", "sector": "RETAIL", "change": 0.45, "price": 176.61}, {"ticker_symbol": "JKL", "sector": "TECHNOLOGY", "change": -0.25, "price": 14.85}, {"ticker_symbol": "NGC", "sector": "HEALTHCARE", "change": 0.08, "price": 4.91}, {"ticker_symbol": "NGC", "sector": "HEALTHCARE", "change": 0.31, "price": 5.22}, {"ticker_symbol": "BAC", "sector": "FINANCIAL", "change": 0.25, "price": 15.81}, {"ticker_symbol": "SED", "sector": "HEALTHCARE", "change": 0.02, "price": 12.22}, {"ticker_symbol": "HJV", "sector": "ENERGY", "change": -7.24, "price": 216.19}, {"ticker_symbol": "DFT", "sector": "RETAIL", "change": -1.04, "price": 96.87}, {"ticker_symbol": "PJM", "sector": "RETAIL", "change": -0.58, "price": 133.77}
```



Implementación de Kinesis con Lambda – Cloud Watch



Link: https://lucid.app/lucidchart/6f7c023b-9b7d-4932-afa6-7218dc5937ec/edit?invitationId=inv_0c71265b-4d39-4bbd-85cd-bfa46c6fa9a6

Implementation con AWS Serverless Application Model (SAM) – Cloud Watch

Implementación de Cognito – Api Gateway

Cognito: Amazon Cognito le permite incorporar de manera rápida y sencilla el registro, inicio de sesión y control de acceso de usuarios a sus aplicaciones web y móviles.

Api Gateway: Un API Gateway es el gestor de tráfico que interactúa con los datos o el servicio backend real y aplica políticas, autenticación y control de acceso general para las llamadas de una API para proteger datos valiosos.

Asegurar AWS API Gateway usando Cognito

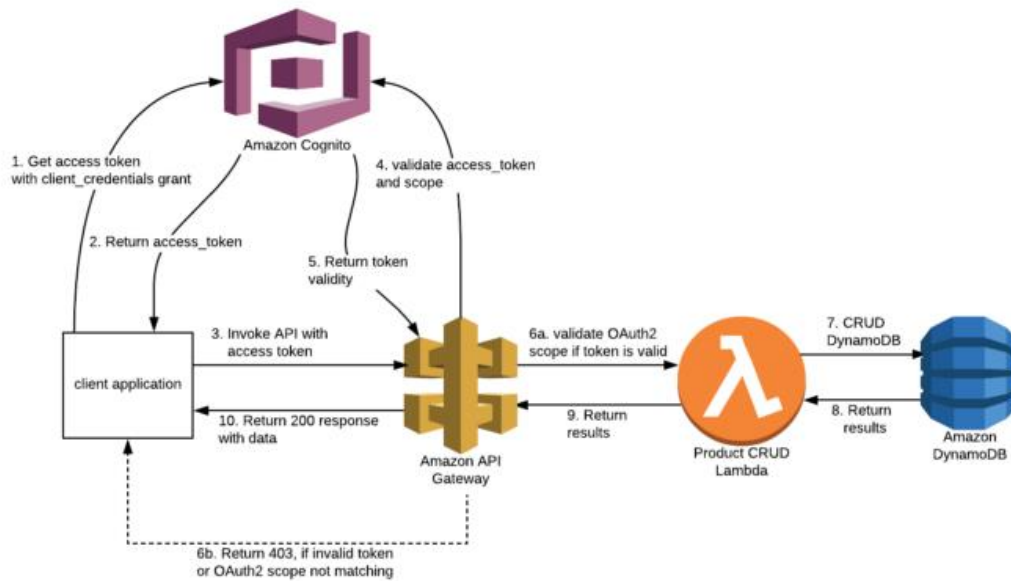


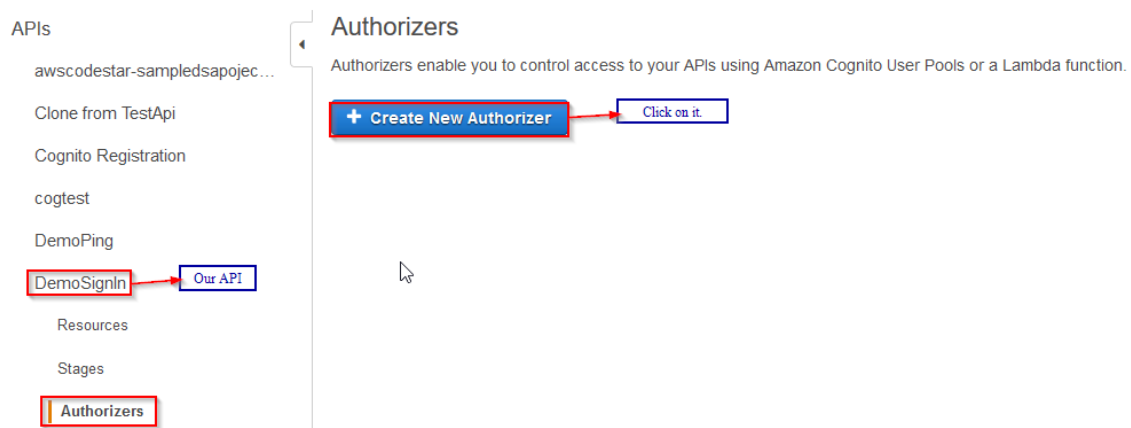
Diagrama de como funciona de manera conjunta

1. Invoque AWS Cognito / oauth2 / token endpoint con grant_type como client_credentials. Referir <https://docs.aws.amazon.com/cognito/latest/developerguide/token-endpoint.html>
2. Si la solicitud es válida, AWS Cognito devolverá un access_token con formato JWT (JSON Web Token)
3. Pase este token en el encabezado de autorización para todas las llamadas a la API
4. API Gateway realiza una llamada a AWS Cognito para validar el access_token.
5. AWS Cognito devuelve una respuesta de validación de tokens.

Configuración de Api Gateway con Congnito

Paso 1: Cree API Gateway y asígnele la función lambda correspondiente.

Paso 2: Haga clic en su API Gateway y luego en Autorizadores.



Paso 3: Configuramos la nueva Autorización indicando que vamos a usar Cognito

awscodestar-sampledapojec...

Clone from TestApi

Cognito Registration

cogtest

DemoPing

DemoSignIn

Resources

Stages

Authorizers

Gateway Responses

Models

Documentation

Dashboard

Settings

DemoUser

Authorizers enable you to control access to your APIs using Amazon Cognito User Pools or a Lambda function.

+ Create New Authorizer

Create Authorizer

Name * Name of the Authorizer

CognitoAuthorizer

Type * i

☐ Lambda ☒ Cognito

Cognito User Pool * i Select user pool you have created.

us-west-2 SampleUserPool

Token Source * i **Token Validation** i

Authorization

Create **Cancel**

Click on it.

Paso 4: Ahora debe agregar el autorizador creado a la solicitud de método en la API

Resources **Actions** Method Execution /authenticate - PUT - **Method Request**

/

/authenticate

OPTIONS

PUT

Provide information about this method's authorization settings and the parameters it can receive.

Settings +

Authorization NONE ⌵ ⓘ

Request Validator NONE

API Key Required Cognito user pool authorizers

CognitoAuthorizer Select your Authorizer in the dropdown.

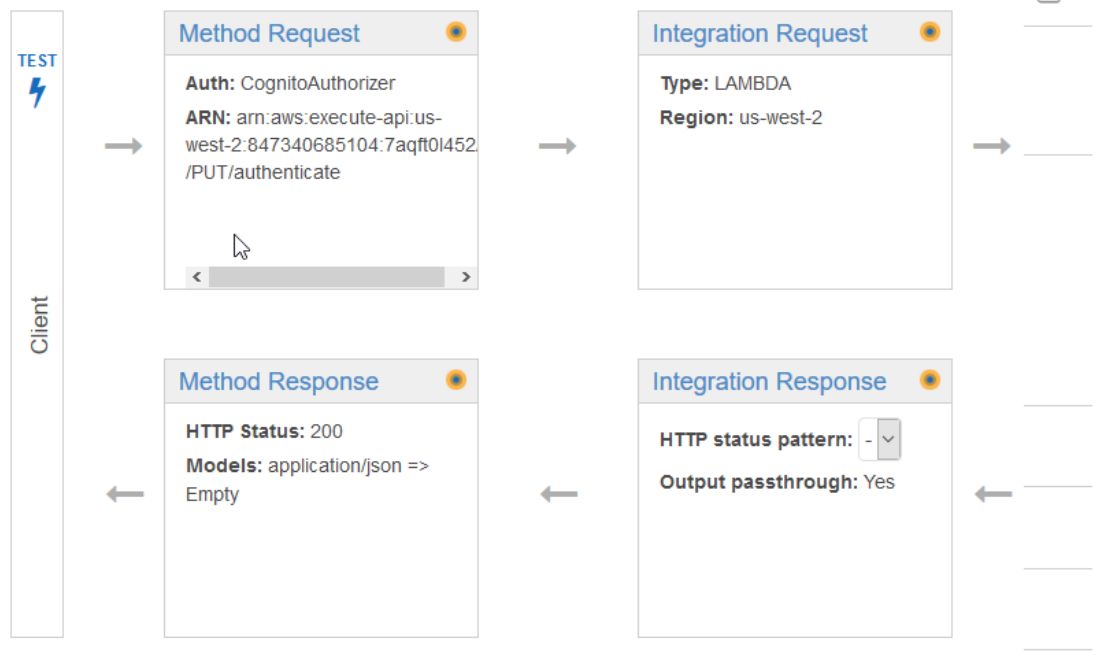
► URL Query String Parameters +

► HTTP Request Headers

► Request Body 📄 +

► SDK Settings

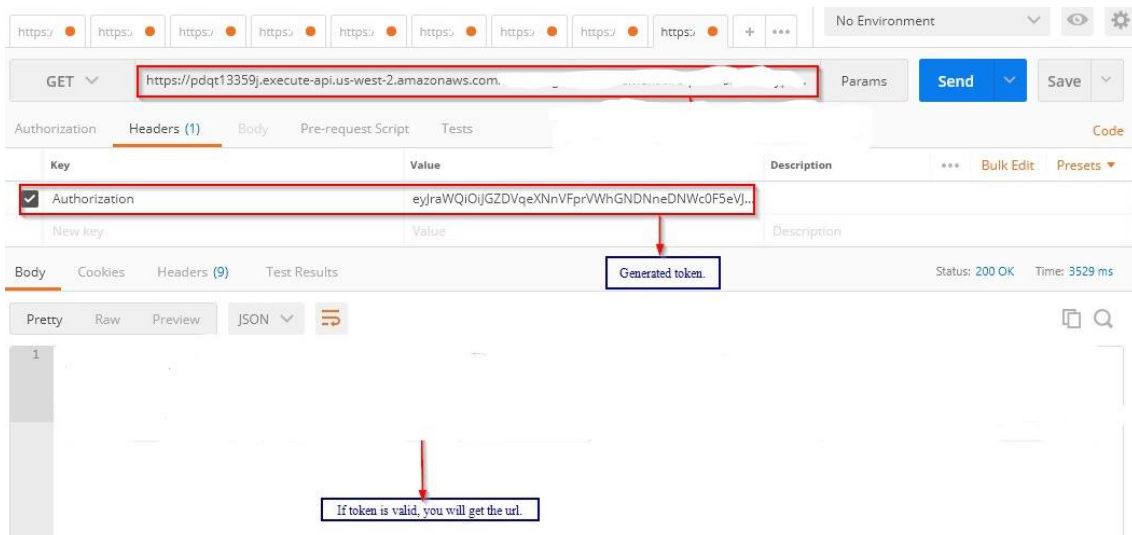
/authenticate - PUT - Method Execution

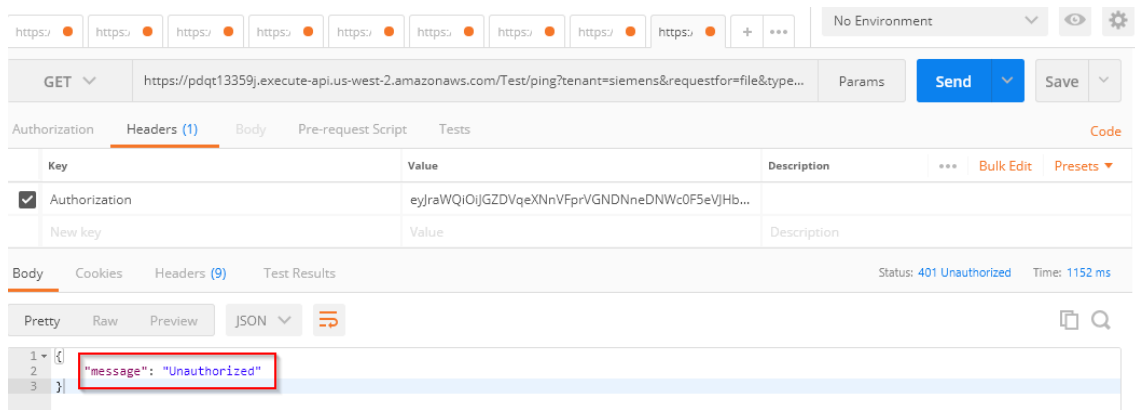


Paso 4: luego vuelva a implementar su API para guardar los cambios, obtendrá la URL para invocar la API, copie esa URL para probarla.

Paso 5: en Postman, pegue esa URL y seleccione el tipo de método que ha usado en API Gateway.

Paso 6: haga clic en Encabezados y seleccione Clave como "Autorización", Valor como "Token generado", luego haga clic en el botón Enviar para verificar si el usuario está autorizado o no.





Implementación de Cognito – Lambda



Link: https://lucid.app/lucidchart/de3fd5a1-eadd-4788-a05b-a4ce4f240388/edit?invitationId=inv_7bd081c2-de88-4edc-b01d-8c5c5f244d7c

Implementación de Cognito – Cloud Watch

Usando Amazon Kinesis y Data Firehose

Buscar ejemplos en internet donde se usen implementaciones de Kinesis con Lambda

Realizar una estimación de costos de tener habilitado el servicio de manera mensual, usando un estimado en la calculadora.

Anexos

Tema	Estudiante	LucidChard	Youtube	Enunciado pregunta	Link pregunta	Link GITHUB
Implementación de colas con Amazon Kinesis – Cloud Watch	Ramírez Castro Arlin Damaris	https://lucid.app/lucidchart/7bc94472-680b-44c4-915c-b7253880fa23/edit?invitationId=inv_03f0e3cc-e993-4a87-88c6-d49f9c12ca53		¿Qué es Amazon kinesis?	https://docs.google.com/forms/d/165T0_NX9I8Nh9uGuSy8st5SULmezn9l-Nlm-cZvLOxA/edit	
Implementación de Kinesis con Lambda – Cloud Watch	Seme Márquez Jair Josue	https://lucid.app/lucidchart/6f7c023b-9b7d-4932-afa6-7218dc5937ec/edit?invitationId=inv_0c71265b-4d39-4bbd-85cd-bfa46c6fa9a6		Indique a que tema pertenece este concepto:	https://docs.google.com/forms/d/165T0_NX9I8Nh9uGuSy8st5SULmezn9l-Nlm-cZvLOxA/edit?usp=sharing	
Implementar con AWS Serverless Application Model (SAM) - Cloud Watch	Vaccaro Cedillo Gean Carlo			¿Qué es? AWS Serverless Application Model(AWS SAM)?	https://docs.google.com/forms/d/165T0_NX9I8Nh9uGuSy8st5SULmezn9l-Nlm-cZvLOxA/edit	

Implementación de Cognito – Api Gateway	Arteaga Burgos Anthony	https://lucid.app/lucidchart/f7667af5-bf03-4076-bf4e-79f5f7c807b5/edit?invitationId=inv_442b54cb-220d-48d6-be3b-d7316f46bab1	https://youtu.be/lstajfxBn-I	Ofrece autenticación, autorización y administración de usuarios para sus aplicaciones móviles y web ¿A que herramienta pertenece este concepto?	https://docs.google.com/forms/d/165T0_NX9I8Nh9uGuSy8st5SULmezn9l-Nlm-cZvLOxA/edit	https://github.com/Anthart/Implementacion-Cognito-Api-Gateway.git
Implementación Cognito-Lambda	Baque Chompol Leonidas Edward	https://lucid.app/lucidchart/de3fd5a1-eadd-4788-a05b-a4ce4f240388/edit?invitationId=inv_7bd081c2-de88-4edc-b01d-8c5c5f244d7c		¿Amazon Cognito invoca este desencadenador cuando un usuario intenta iniciar sesión?	https://docs.google.com/forms/d/1csboBuoU0bi9CeeQHmwlwFUZZZc1C1NuDBzqeWYab6Y/edit?usp=sharing	
	Pilay Lino Jennifer Lisbeth					
	Naranjo Iñiga Pedro Anthony					

Ejemplos de implementacion en internet de AWS Kenesis con AWS lambdan	Yumaglla Guagcha Franklin David	https://lucid.app/lucidchart/7141b48b-7034-456c-a63e-b198f0c7c3da/edit?invitationId=inv_2c417b00-7e5a-4753-a14e-9ab2a292d879	¿Cuáles son los servicios de AWS Kenesis?	https://docs.google.com/forms/d/165T0_NX9I8Nh9uGuSy8st5SULmezn9l-Nlm-cZvLOxA/edit
---	---------------------------------	---	---	---

