aws

Guía para desarrolladores

Amazon Data Firehose

Amazon Data Firehose: Guía para desarrolladores

Table of Contents

..... ix
¿Qué es Amazon Data Firehose? 1
Conozca los conceptos clave 1
Comprenda el flujo de datos en Amazon Data Firehose 2
Configuración 5
Inscríbase en AWS 5
(Opcional) Descarga bibliotecas y herramientas 5
Creación de una transmisión de Firehose 7
Configurar el origen y el destino 7
Configurar la transformación de registros y la conversión de formato 10
Configurar los ajustes de destino 12
Configurar los ajustes de destino para Amazon S3 12
Configurar los ajustes de destino de Amazon Redshift 16
OpenSearch Configure los ajustes de destino del servicio 23
Configure los ajustes de destino para Serverless OpenSearch 25
Configure los ajustes de destino para el punto final HTTP 26
Configura los ajustes de destino para Datadog 28
Configure los ajustes de destino para Honeycomb 31
Configure los ajustes de destino para Coralogix 32
Configure los ajustes de destino para Dynatrace 35
Configure los ajustes de destino para LogicMonitor 37
Configure los ajustes de destino para Logz.io 38
Configurar los ajustes de destino para MongoDB Cloud 40
Configure los ajustes de destino de New Relic 42
Configure los ajustes de destino para Snowflake 44
Configure los ajustes de destino para Splunk 47
Configure los ajustes de destino para Splunk Observability Cloud 49
Configure los ajustes de destino para Sumo Logic 51
Configura los ajustes de destino para Elastic 52
Configurar la copia de seguridad y los ajustes avanzados 54
Configura los ajustes de respaldo 54
Configuración de opciones avanzadas 56
Comprenda las sugerencias de almacenamiento en búfer 57
Probando tu transmisión de Firehose 61

Requisitos previos ... 61
Pruebas con Amazon S3 como destino ... 61
Pruebas con Amazon Redshift como destino ... 62
Pruebe utilizando el OpenSearch servicio como destino ... 63
Prueba con Splunk como destino ... 63
Envío de datos a una transmisión de Firehose ... 65
Escritura mediante Kinesis Data Streams ... 65
Escritura mediante Amazon MSK ... 67
Escritura con Amazon Data Firehose Agent ... 69
Requisitos previos ... 70
Credenciales ... 70
Proveedores de credenciales personalizados ... 71
Descargar e instalar el agente ... 72
Configuración e inicio del agente ... 74
Ajustes de la configuración del agente ... 75
Monitoreo de varios directorios de archivos y escritura en varias secuencias ... 79
Uso del agente para preprocesar los datos ... 80
Comandos del agente de la CLI ... 84
Preguntas frecuentes ... 85
Envíe datos mediante el SDK AWS ... 86
Operaciones de escritura única mediante PutRecord ... 87
Operaciones de escritura por lotes mediante PutRecordBatch ... 88
Escribir usando registros CloudWatch ... 88
Descompresión de registros CloudWatch ... 89
Extracción de mensajes tras la descompresión de los registros CloudWatch ... 89
Activación y desactivación de la descompresión ... 90
Preguntas frecuentes ... 85
Escribir usando eventos CloudWatch ... 93
Escritura mediante AWS IoT ... 94
Seguridad ... 95
Protección de los datos ... 96
Cifrado del servidor con Kinesis Data Streams como origen de datos ... 96
Cifrado del lado del servidor con Direct PUT u otros orígenes de datos ... 96
Control de acceso ... 98
Conceda a su aplicación acceso a sus recursos de Amazon Data Firehose ... 99
Conceda a Amazon Data Firehose acceso a su clúster privado de Amazon MSK ... 99

Permita que Amazon Data Firehose asuma una función de IAM 100
Conceda acceso a Amazon Data Firehose AWS Glue para la conversión de formatos de datos 102
Conceda a Amazon Data Firehose acceso a un destino de Amazon S3 103
Conceda a Amazon Data Firehose acceso a un destino de Amazon Redshift 106
Conceda a Amazon Data Firehose acceso a un destino de servicio público OpenSearch 110
Otorgue a Amazon Data Firehose acceso a un destino de OpenSearch servicio en una VPC 114
Conceda a Amazon Data Firehose acceso a un destino público sin servidor OpenSearch ... 115
Otorgue a Amazon Data Firehose acceso a un destino OpenSearch sin servidor en una VPC 118
Conceda a Amazon Data Firehose acceso a un destino de Splunk 119
Acceso a Splunk en VPC 121
Acceso a Snowflake o al punto final HTTP 123
Conceda a Amazon Data Firehose acceso a un destino con copos de nieve 123
Acceso a Snowflake en VPC 125
Conceda a Amazon Data Firehose acceso a un destino de punto final HTTP 129
Entrega entre cuentas desde Amazon MSK 132
Entrega entre cuentas en un destino de Amazon S3 135
Entrega multicuenta a un destino de servicio OpenSearch 136
Uso de etiquetas para controlar el acceso 138
Autenticate con AWS Secrets Manager 140
Comprenda los secretos 141
Creación de un secreto 142
Usa el secreto 142
Rota el secreto 144
Administre las funciones de IAM a través de la consola 144
Elija un rol de IAM existente 145
Cree un nuevo rol de IAM desde la consola 145
Edita el rol de IAM desde la consola 147
Supervisión 148
Validación de la conformidad 148
Resiliencia 149
Recuperación de desastres 150
Seguridad de infraestructuras 150
Puntos de enlace de la VPC (PrivateLink) 151

Prácticas recomendadas de seguridad 151
Implementación del acceso a los privilegios mínimos 151
Uso de roles de IAM 151
Implementación del cifrado en el servidor en recursos dependientes 152
Se usa CloudTrail para monitorear las llamadas a la API 152
Transformación de datos 153
Flujo de transformación de datos 153
Modelo de estados y transformación de datos 153
Esquemas de Lambda 155
Gestión de errores de transformación de datos 156
Duración de una invocación de Lambda 158
Backup de registros de origen 158
Particionamiento dinámico 159
Claves de particionamiento 160
Creación de claves de particionamiento con análisis en línea 160
Creación de claves de particionamiento con una función de AWS Lambda 161
Prefijo de bucket de Amazon S3 para particionamiento dinámico 165
Particionamiento dinámico de datos agregados 166
Adición de un delimitador de nueva línea al entregar datos en S3 167
Cómo habilitar el particionamiento dinámico 167
Gestión de errores de particionamiento dinámico 168
Almacenamiento en búfer y particionamiento dinámico de datos 169
Conversión del formato de registros 171
Requisitos de la conversión del formato de registro 171
Elección del deserializador JSON 172
Elección del serializador 173
Conversión del formato de registro de entrada (consola) 174
Conversión del formato de registro de entrada (API) 174
Control de errores de la conversión del formato de registro 175
Ejemplo de conversión del formato de registros 176
Integración con Managed Service para Apache Flink 177
Entrega de datos 178
Configure el formato de entrega de datos 178
Comprenda la frecuencia de entrega de datos 180
Gestione los errores en la entrega de datos 180
Configurar el formato de nombre de objeto de Amazon S3 184

Configura la rotación del índice para el servicio OpenSearch 194
Comprenda la entrega en todas las AWS cuentas y regiones 195
Registros duplicados 195
Pausa y reanuda una transmisión de Firehose 195
Entender cómo Firehose gestiona los fallos de entrega 196
Pausar una transmisión de Firehose 196
Reanudación de una transmisión de Firehose 197
Supervisión 198
Prácticas recomendadas con alarmas de CloudWatch 198
Monitorización con métricas CloudWatch 199
Métricas de particionamiento CloudWatch dinámico 200
CloudWatch Métricas de entrega de datos 201
Métricas de adquisición de datos 214
Métricas a nivel de API CloudWatch 222
CloudWatch Métricas de transformación de datos 225
CloudWatch Registra las métricas de descompresión 225
Métricas de conversión de CloudWatch formato 226
Métricas de cifrado del lado del servidor (SSE) CloudWatch 227
Dimensiones de Amazon Data Firehose 227
Métricas de uso de Amazon Data Firehose 228
Acceso a CloudWatch las métricas de Amazon Data Firehose 229
Monitorización con registros CloudWatch 230
Errores de entrega de datos 231
Acceso a CloudWatch los registros de Amazon Data Firehose 267
Monitorización del estado del agente 268
Monitorear con CloudWatch 269
Registro de llamadas a la API Firehose de Amazon Data con AWS CloudTrail 270
Información sobre Amazon Data Firehose en CloudTrail 270
Ejemplo: entradas del archivo de registro de Amazon Data Firehose 271
Prefijos personalizados de Amazon S3 277
El espacio de nombres `timestamp` 277
El espacio de nombres `firehose` 278
Espacios de nombres `partitionKeyFromLambda` y `partitionKeyFromQuery` 279
Reglas semánticas 280
Ejemplos de prefijos 281
Uso de Amazon Data Firehose con AWS PrivateLink 283

Puntos de enlace de VPC de interfaz ()AWS PrivateLink para Amazon Data Firehose 283
Uso de puntos de enlace de VPC de interfaz ()AWS PrivateLink para Amazon Data Firehose .. 283
Disponibilidad 287
Cómo etiquetar tus transmisiones de Firehose 289
Conceptos básicos de etiquetas 289
Seguimiento de costos utilizando el etiquetado 290
Restricciones de las etiquetas 291
Etiquetado de transmisiones de Firehose mediante la API Firehose de Amazon Data 292
Tutorial: Ingiera los registros de flujo de VPC en Splunk mediante Amazon Data Firehose 293
Resolución de problemas 294
Problemas comunes 294
Solución de problemas de Amazon S3 295
Solución de problemas de Amazon Redshift 296
Solución de problemas de Amazon OpenSearch Service 297
Solución de problemas de Splunk 298
Solución de problemas de Snowflake 300
La creación de la transmisión de Firehose falla 300
Solución de problemas de accesibilidad a los puntos finales de Firehose 302
Solución de problemas de puntos de conexión HTTP 303
CloudWatch Registros 303
Solución de problemas de MSK como origen 307
Error de creación de conductos 307
Conducto suspendido 307
Conducto contrapresurizado 308
Actualización incorrecta de los datos 308
Problemas de conexión al clúster de MSK 308
La métrica de actualización de los datos aumenta o no se emite 311
La conversión del formato de registro a Apache Parquet falla 313
Cuota 314
Apéndice: especificaciones de solicitudes y respuestas de entrega de puntos de conexión HTTP 318
Formato de las solicitudes 318
Formato de respuesta 322
Ejemplos 325
Historial de documentos 326
Glosario de AWS 330

Amazon Data Firehose se conocía anteriormente como Amazon Kinesis Data Firehose

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la version original de inglés, prevalecerá la version en inglés.

¿Qué es Amazon Data Firehose?

Amazon Data Firehose es un servicio totalmente gestionado para entregar datos de streaming en tiempo real a destinos como Amazon Simple Storage Service (Amazon S3), Amazon Redshift, Amazon Service, OpenSearch Amazon Serverless, Splunk y cualquier punto de enlace HTTP personalizado o punto de enlace HTTP propiedad de proveedores de servicios externos compatibles, incluidos Datadog, LogicMonitor Dynatrace, MongoDB, New Relic, Coralogix y Elastic. OpenSearch Con Amazon Data Firehose, no necesita escribir aplicaciones ni administrar recursos. Usted configura sus generadores de datos para que envíen datos a Amazon Data Firehose, que los entrega automáticamente al destino que haya especificado. También puede configurar Amazon Data Firehose para que transforme los datos antes de entregarlos.

Para obtener más información sobre las soluciones de AWS big data, consulte Big Data en AWS. Para obtener más información sobre las soluciones de datos de streaming de AWS , consulte ¿Qué son los datos de streaming?.

> **Note**
>
> Tenga en cuenta la última solución de datos de AWS streaming para Amazon MSK, que proporciona AWS CloudFormation plantillas en las que los datos fluyen a través de los productores, el almacenamiento de streaming, los consumidores y los destinos.

Conozca los conceptos clave

Al empezar a utilizar Amazon Data Firehose, podrá beneficiarse de la comprensión de los siguientes conceptos:

Firehose Stream

La entidad subyacente de Amazon Data Firehose. Para utilizar Amazon Data Firehose, debe crear una transmisión de Firehose y, a continuación, enviarle datos. Para obtener más información, consulte Crea una transmisión de Firehose y Enviar datos a una transmisión de Firehose.

record

Los datos de interés que su productor de datos envía a una transmisión de Firehose. Cada registro puede pesar hasta 1 000 KB.

productor de datos

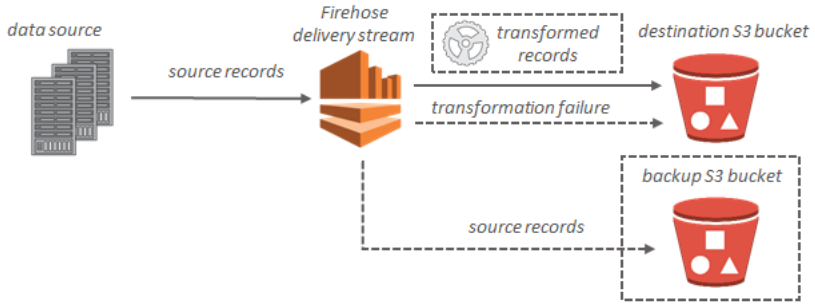
Los productores envían los discos a Firehose Streams. Por ejemplo, un servidor web que envía datos de registro a una transmisión de Firehose es un productor de datos. También puede configurar la transmisión de Firehose para que lea automáticamente los datos de una transmisión de datos de Kinesis existente y los cargue en los destinos. Para obtener más información, consulte Enviar datos a una transmisión de Firehose.

tamaño e intervalo del búfer

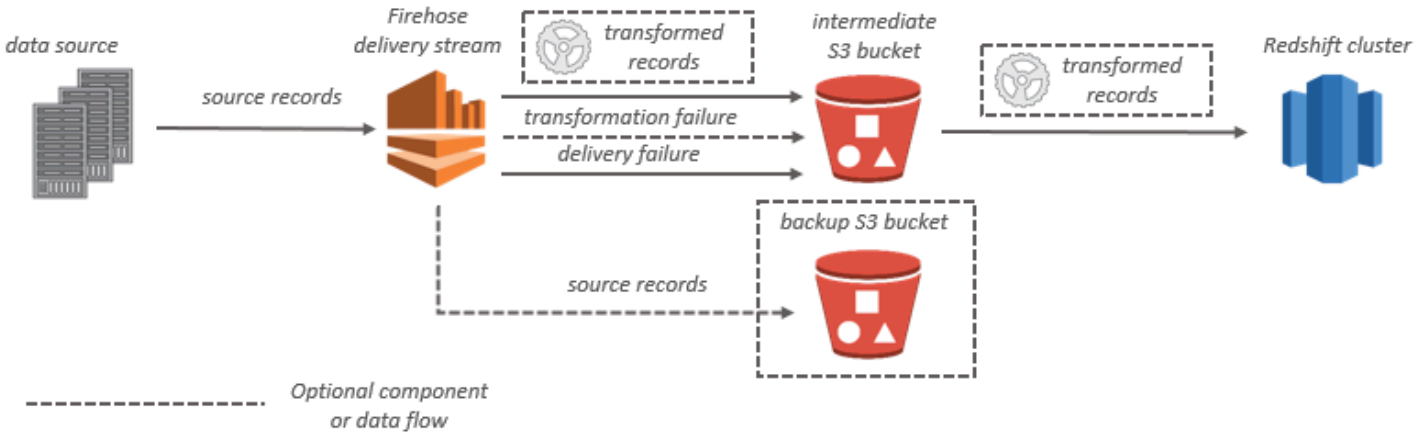
Amazon Data Firehose almacena en búfer los datos de streaming entrantes hasta un tamaño determinado o durante un período de tiempo determinado antes de entregarlos a los destinos. Buffer Sizeestá en MB y en Buffer Interval segundos.

Comprenda el flujo de datos en Amazon Data Firehose

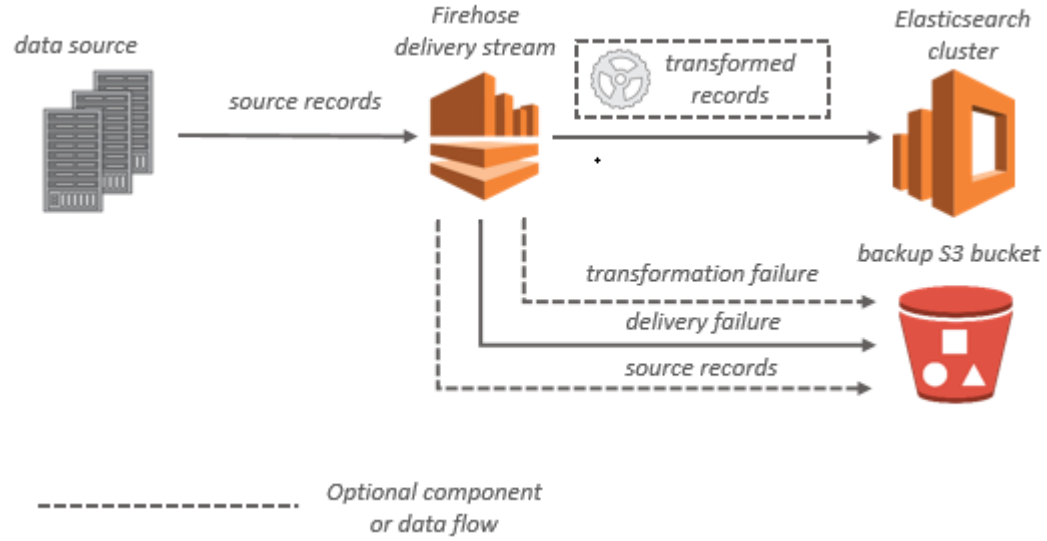
En el caso de los destinos de Amazon S3, los datos de streaming se entregan en el bucket de S3. Si habilita la transformación de datos, puede realizar una copia de seguridad de los datos de origen en otro bucket de Amazon S3.



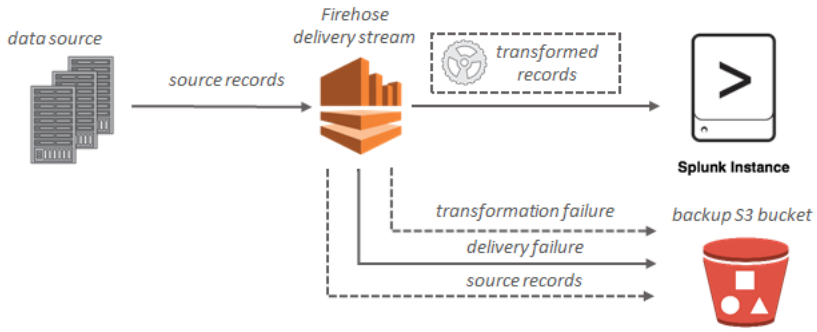
En el caso de los destinos de Amazon Redshift, los datos de streaming se entregan primero en el bucket de S3. A continuación, Amazon Data Firehose emite un comando de Amazon COPY Redshift para cargar los datos del bucket de S3 al clúster de Amazon Redshift. Si habilita la transformación de datos, puede realizar una copia de seguridad de los datos de origen en otro bucket de Amazon S3.



En el OpenSearch caso de los destinos de servicio, los datos de streaming se envían a su clúster de OpenSearch servicios y, si lo prefiere, se puede hacer una copia de seguridad de los mismos en su bucket de S3 de forma simultánea.



Si el destino es Splunk, los datos de streaming se entregan a Splunk y se puede hacer un backup de ellos en el bucket de S3 simultáneamente.


data source
source records
Firehose
delivery stream
transformed
records
Splunk Instance
transformation failure
delivery failure
source records
backup S3 bucket

Configuración de Amazon Data Firehose

Antes de utilizar Amazon Data Firehose por primera vez, complete las siguientes tareas.

Tareas

- Inscríbase en AWS
- (Opcional) Descarga bibliotecas y herramientas

Inscríbase en AWS

Cuando te registras en Amazon Web Services (AWS), tu AWS cuenta se registra automáticamente en todos los servicios de Amazon Data Firehose AWS, incluido Amazon. Solo se le cobrará por los servicios que utilice.

Si ya tienes una AWS cuenta, pasa a la siguiente tarea. Si no dispone de una cuenta de AWS , utilice el siguiente procedimiento para crear una.

Para crear una AWS cuenta

1. Abra https://portal.aws.amazon.com/billing/signup.
2. Siga las instrucciones que se le indiquen.

   Parte del procedimiento de registro consiste en recibir una llamada telefónica e indicar un código de verificación en el teclado del teléfono.

   Cuando te registras en una Cuenta de AWS, Usuario raíz de la cuenta de AWSse crea una. El usuario raíz tendrá acceso a todos los Servicios de AWS y recursos de esa cuenta. Como práctica recomendada de seguridad, asigne acceso administrativo a un usuario y utilice únicamente el usuario raíz para realizar tareas que requieren acceso de usuario raíz.

(Opcional) Descarga bibliotecas y herramientas

Las siguientes bibliotecas y herramientas le ayudarán a trabajar con Amazon Data Firehose mediante programación y desde la línea de comandos:

- Las operaciones de la API Firehose son el conjunto básico de operaciones que admite Amazon Data Firehose.

- Los AWS SDK para Go, Java, .NET, Node.js, Python y Ruby incluyen soporte y ejemplos para Amazon Data Firehose.

  Si su versión AWS SDK for Java no incluye muestras de Amazon Data Firehose, también puede descargar el AWS SDK más reciente desde. GitHub
- AWS Command Line InterfaceEs compatible con Amazon Data Firehose. AWS CLI Esto le permite controlar varios AWS servicios desde la línea de comandos y automatizarlos mediante scripts.

Crea una transmisión de Firehose

Puedes usar el AWS Management Console o un AWS SDK para crear una transmisión de Firehose con destino al destino que elijas.

Puedes actualizar la configuración de tu transmisión de Firehose en cualquier momento después de crearla, mediante la consola Amazon Data Firehose o. UpdateDestination La transmisión de Firehose permanece en ese `Active` estado mientras se actualiza la configuración y puedes seguir enviando datos. La configuración actualizada suele entrar en vigor transcurridos unos minutos. El número de versión de una transmisión de Firehose aumenta en un valor de 1 después de actualizar la configuración. Se refleja en el nombre del objeto de Amazon S3 entregado. Para obtener más información, consulte Configurar el formato de nombre de objeto de Amazon S3.

En los siguientes temas se describe cómo crear una transmisión Firehose.

Temas

- Configurar el origen y el destino
- Configurar la transformación de registros y la conversión de formato
- Configurar los ajustes de destino
- Configurar la copia de seguridad y los ajustes avanzados
- Comprenda las sugerencias de almacenamiento en búfer

Configurar el origen y el destino

1. Inicie sesión en la consola Amazon Data Firehose AWS Management Console y ábrala en https://console.aws.amazon.com/firehose
2. Selecciona Crear transmisión de Firehose.
3. Escriba valores en los siguientes campos:

   Origen

   - Direct PUT: elija esta opción para crear una transmisión Firehose en la que las aplicaciones del productor escriban directamente. Actualmente, los siguientes son AWS los servicios, agentes y servicios de código abierto que se integran con Direct PUT en Amazon Data Firehose:

- AWS SDK
- AWS Lambda
- AWS CloudWatch Registros
- AWS CloudWatch Eventos
- AWS Flujos de métricas en la nube
- AWS IOT
- AWS Eventbridge
- Amazon Simple Email Service
- Amazon SNS
- AWS Registros de ACL web de WAF
- Amazon API Gateway: registros de acceso
- Amazon Pinpoint
- Registro de agente de Amazon MSK
- Registros de consultas de Amazon Route 53 Resolver
- AWS Registros de alertas de Network Firewall
- AWS Registros de flujo de Network Firewall
- Amazon Elasticache Redis SLOWLOG
- Agente de Kinesis (Linux)
- Kinesis Tap (Windows)
- Fluentbit
- Fluentd
- Apache Nifi
- Snowflake

- Transmisión de Kinesis: elija esta opción para configurar una transmisión de Firehose que utilice una transmisión de datos de Kinesis como fuente de datos. A continuación, puede utilizar Amazon Data Firehose para leer fácilmente los datos de una transmisión de datos de Kinesis existente y cargarlos en los destinos. Para obtener más información sobre el uso de Kinesis Data Streams como fuente de datos, consulte Cómo escribir en Amazon Data Firehose mediante Kinesis Data Streams.
- Amazon MSK: elija esta opción para configurar una transmisión de Firehose que utilice Amazon MSK como fuente de datos. A continuación, puede usar Firehose para leer

fácilmente los datos de un clúster de Amazon MSK existente y cargarlos en buckets S3 específicos. Para obtener más información sobre el uso de Amazon MSK como fuente de datos, consulte Cómo escribir en Amazon Data Firehose con Amazon MSK.

Destino del arroyo Firehose

El destino de tu transmisión Firehose. Amazon Data Firehose puede enviar registros de datos a varios destinos, incluidos Amazon Simple Storage Service (Amazon S3), Amazon Redshift, OpenSearch Amazon Service y cualquier punto de enlace HTTP que sea propiedad suya o de alguno de sus proveedores de servicios externos. A continuación se indican los destinos admitidos:

- OpenSearch Servicio Amazon
- Amazon OpenSearch Serverless
- Amazon Redshift
- Amazon S3
- Coralogix
- Datadog
- Dynatrace
- Elastic
- Punto de conexión HTTP
- Honeycomb
- Logic Monitor
- Logz.io
- MongoDB Cloud
- New Relic
- Splunk
- Splunk Observability Cloud
- Sumo Logic
- Snowflake

Nombre de la transmisión Firehose

El nombre de tu transmisión de Firehose.

Configurar la transformación de registros y la conversión de formato

Configure Amazon Data Firehose para transformar y convertir los datos de sus registros.

- Si eliges Amazon MSK como fuente de tu transmisión de Firehose.

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

     Transformación de datos

     Para crear una transmisión Firehose que no transforme los datos entrantes, no active la casilla Habilitar la transformación de datos.

     Para especificar una función Lambda para que Firehose invoque y utilice para transformar los datos entrantes antes de entregarlos, active la casilla Habilitar la transformación de datos. Puede configurar una nueva función de Lambda con uno de los esquemas de Lambda o puede elegir una función de Lambda existente. La función Lambda debe contener el modelo de estado que requiere Firehose. Para obtener más información, consulte Transformación de datos en Amazon Data Firehose.
  2. En la sección Convert record format (Convertir formato de registros), proporcione valores para el siguiente campo:

     Record format conversion (Conversión del formato de registros)

     Para crear una transmisión de Firehose que no convierta el formato de los registros de datos entrantes, selecciona Desactivado.

     Para convertir el formato de los registros entrantes, elija Enabled (Habilitada) y, a continuación, especifique el formato de salida que desee. Debe especificar una AWS Glue tabla que contenga el esquema que desea que Firehose utilice para convertir el formato de registro. Para obtener más información, consulte Conversión del formato de registros.

     Para ver un ejemplo de cómo configurar la conversión de formatos de registro AWS CloudFormation, consulta AWS::KinesisFirehose: DeliveryStream.

- Si eliges Managed Service for Apache Flink o Direct PUT como fuente para tu transmisión de Firehose, en la sección Configuración de código fuente:

1. En Transformar registros, elige una de las siguientes opciones:

   a. Si tu destino es Amazon S3 o Splunk, en la sección Descomprimir CloudWatch registros fuente de Amazon Logs, selecciona Activar la descompresión.

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

      Transformación de datos

      Para crear una transmisión Firehose que no transforme los datos entrantes, no active la casilla Habilitar la transformación de datos.

      Para especificar una función Lambda para que Amazon Data Firehose la invoque y utilice para transformar los datos entrantes antes de entregarlos, active la casilla Habilitar la transformación de datos. Puede configurar una nueva función de Lambda con uno de los esquemas de Lambda o puede elegir una función de Lambda existente. La función Lambda debe contener el modelo de estado que exige Amazon Data Firehose. Para obtener más información, consulte Transformación de datos en Amazon Data Firehose.

2. En la sección Convert record format (Convertir formato de registros), proporcione valores para el siguiente campo:

   Record format conversion (Conversión del formato de registros)

   Para crear una transmisión de Firehose que no convierta el formato de los registros de datos entrantes, selecciona Desactivado.

   Para convertir el formato de los registros entrantes, elija Enabled (Habilitada) y, a continuación, especifique el formato de salida que desee. Debe especificar una AWS Glue tabla que contenga el esquema que desea que Amazon Data Firehose utilice para convertir el formato de registro. Para obtener más información, consulte Conversión del formato de registros.

   Para ver un ejemplo de cómo configurar la conversión de formatos de registro AWS CloudFormation, consulte AWS::KinesisFirehose: DeliveryStream.

Configurar los ajustes de destino

En este tema se describe la configuración de destino de tu transmisión Firehose en función del destino que selecciones. Para obtener más información sobre las sugerencias de almacenamiento en búfer, consulte. Comprenda las sugerencias de almacenamiento en búfer

Temas

- Configurar los ajustes de destino para Amazon S3
- Configurar los ajustes de destino de Amazon Redshift
- OpenSearch Configure los ajustes de destino del servicio
- Configure los ajustes de destino para Serverless OpenSearch
- Configure los ajustes de destino para el punto final HTTP
- Configura los ajustes de destino para Datadog
- Configure los ajustes de destino para Honeycomb
- Configure los ajustes de destino para Coralogix
- Configure los ajustes de destino para Dynatrace
- Configure los ajustes de destino para LogicMonitor
- Configure los ajustes de destino para Logz.io
- Configurar los ajustes de destino para MongoDB Cloud
- Configure los ajustes de destino de New Relic
- Configure los ajustes de destino para Snowflake
- Configure los ajustes de destino para Splunk
- Configure los ajustes de destino para Splunk Observability Cloud
- Configure los ajustes de destino para Sumo Logic
- Configura los ajustes de destino para Elastic

Configurar los ajustes de destino para Amazon S3

Debe especificar los siguientes ajustes para poder utilizar Amazon S3 como destino de su transmisión de Firehose.

- Escriba valores en los siguientes campos:

S3 bucket

Seleccione el bucket de S3 de su propiedad adonde se deben entregar los datos de streaming. Puede crear un bucket de S3 nuevo o elegir uno disponible.

Delimitador de nueva línea

Puede configurar su transmisión Firehose para añadir un nuevo delimitador de líneas entre los registros de los objetos que se envían a Amazon S3. Para ello, elija Habilitado. Para no agregar un delimitador de nueva línea entre los registros de los objetos que se entregan en Amazon S3, seleccione Deshabilitado. Si planea usar Athena para consultar objetos de S3 con registros agregados, active esta opción.

Particionamiento dinámico

Seleccione Habilitado para habilitar y configurar el particionamiento dinámico.

Desagregación de varios registros

Este es el proceso de analizar los registros de la secuencia Firehose y separarlos en función de un JSON válido o del nuevo delimitador de línea especificado.

Si agregas varios eventos, registros o registros en una sola PutRecord llamada a la PutRecordBatch API, aún puedes habilitar y configurar la partición dinámica. Con los datos agregados, cuando habilita la partición dinámica, Amazon Data Firehose analiza los registros y busca varios objetos JSON válidos en cada llamada a la API. Cuando la transmisión de Firehose está configurada con Kinesis Data Stream como fuente, también puede utilizar la agregación integrada en la biblioteca de productores de Kinesis (KPL). La funcionalidad de partición de datos se ejecuta después de desagregar los datos. Por lo tanto, cada registro de cada llamada a la API se puede entregar en distintos prefijos de Amazon S3. También puede aprovechar la integración de la función Lambda para realizar cualquier otra desagregación o transformación antes de la funcionalidad de particionamiento de datos.

> **Important**
>
> Si sus datos están agregados, el particionamiento dinámico solo se puede aplicar una vez completada la desagregación de los datos. Por lo tanto, si habilita el particionamiento dinámico en sus datos agregados, debe seleccionar Habilitado para habilitar la desagregación de varios registros.

Firehose Stream realiza los siguientes pasos de procesamiento en el siguiente orden: desagregación KPL (protobuf), desagregación de JSON o delimitadores, procesamiento Lambda, particionamiento de datos, conversión de formatos de datos y entrega en Amazon S3.

Tipo de desagregación de varios registros

Si has activado la desagregación de varios registros, debes especificar el método para que Firehose desagregue los datos. Utilice el menú desplegable para elegir JSON o Delimitado.

Análisis en línea

Este es uno de los mecanismos admitidos para particionar dinámicamente los datos vinculados a Amazon S3. A fin de usar el análisis en línea para el particionamiento dinámico de sus datos, debe especificar los parámetros de registro de datos que se utilizarán como claves de particionamiento y proporcionar un valor para cada clave de particionamiento especificada. Seleccione Habilitado para habilitar y configurar el análisis en línea.

> **Important**
>
> Si especificó una función AWS Lambda en los pasos anteriores para transformar los registros fuente, puede utilizarla para particionar dinámicamente los datos enlazados a S3 y seguir creando las claves de partición con el análisis en línea. Con el particionamiento dinámico, puede utilizar el análisis en línea o la función AWS Lambda para crear las claves de particionamiento. O bien, puede usar el análisis en línea y la función AWS Lambda al mismo tiempo para crear las claves de partición.

Claves de particionamiento dinámico

Puede usar los campos Clave y Valor para especificar los parámetros de registro de datos que se utilizarán como claves de particionamiento dinámico y las consultas jq para generar valores de claves de particionamiento dinámico. Firehose solo es compatible con jq 1.6. Puede especificar hasta 50 claves de particionamiento dinámico. Debes introducir expresiones jq válidas para tus valores clave de particionamiento dinámico a fin de configurar correctamente el particionamiento dinámico para tu transmisión de Firehose.

Prefijo de bucket de S3

Al habilitar y configurar el particionamiento dinámico, debe especificar los prefijos del bucket S3 a los que Amazon Data Firehose entregará los datos particionados.

Para que el particionamiento dinámico se configure correctamente, el número de prefijos de bucket de S3 debe ser idéntico al número de claves de particionamiento especificadas.

Puede particionar los datos de origen con el análisis en línea o con la función Lambda AWS que especifique. Si especificó una función de AWS Lambda para crear claves de partición para los datos de origen, debe escribir manualmente los valores del prefijo del bucket S3 con el siguiente formato: "Lambda:keyID». partitionKeyFrom Si utiliza el análisis en línea para especificar las claves de partición para sus datos de origen, puede escribir manualmente los valores de vista previa del bucket de S3 con el siguiente formato: «partitionKeyFromquery:keyID» o puede elegir el botón Aplicar claves de partición dinámica para utilizar sus pares clave/valor de particionamiento dinámico para generar automáticamente los prefijos de su bucket de S3. Al particionar sus datos con análisis en línea o AWS Lambda, también puede usar las siguientes formas de expresión en el prefijo de su bucket de S3:! {namespace:value}, donde el espacio de nombres puede ser Query o Lambda. partitionKeyFrom partitionKeyFrom

El bucket S3 y el error S3 generan una zona horaria con prefijo

Elija la zona horaria que desee utilizar para la fecha y la hora en los prefijos personalizados de Amazon Simple Storage Service Objects. De forma predeterminada, Firehose añade un prefijo de hora en UTC. Puede cambiar la zona horaria utilizada en los prefijos S3 si desea utilizar una zona horaria diferente.

Sugerencias de almacenamiento en búfer

Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino varía de un proveedor de servicios a otro.

Compresión S3

Elija la compresión de datos GZIP, Snappy, Zip o Snappy compatible con Hadoop, o sin compresión de datos. La compresión Snappy compatible con Snappy, Zip y Hadoop no está disponible para las transmisiones de Firehose con Amazon Redshift como destino.

Formato de extensión de archivo S3 (opcional)

Especifique un formato de extensión de archivo para los objetos entregados al bucket de destino de Amazon S3. Si habilita esta función, la extensión de archivo especificada anulará las extensiones de archivo predeterminadas incorporadas por las funciones de conversión de formato de datos o de compresión de S3, como .parquet o .gz. Asegúrese de haber configurado la extensión de archivo correcta cuando utilice esta función con la conversión de formato de datos o la compresión S3. La extensión del archivo debe empezar con un punto (.) y puede contener los caracteres permitidos: 0-9a-z! -_.*' (). La extensión del archivo no puede superar los 128 caracteres.

Cifrado S3

Firehose admite el cifrado del lado del servidor de Amazon S3 con AWS Key Management Service (SSE-KMS) para cifrar los datos entregados en Amazon S3. Puede optar por utilizar el tipo de cifrado predeterminado especificado en el depósito S3 de destino o cifrar con una clave de la lista de claves de su propiedad. AWS KMS Si cifra los datos con AWS KMS claves, puede usar la clave AWS administrada predeterminada (aws/s3) o una clave administrada por el cliente. Para obtener más información, consulte Protección de datos mediante el cifrado del lado del servidor con claves administradas por AWS KMS (SSE-KMS).

Configurar los ajustes de destino de Amazon Redshift

En esta sección se describe la configuración para usar Amazon Redshift como destino de transmisión de Firehose.

Elija uno de los siguientes procedimientos en función de si tiene un clúster aprovisionado de Amazon Redshift o un grupo de trabajo de Amazon Redshift sin servidor.

- Clúster aprovisionado de Amazon Redshift
- Configurar los ajustes de destino para el grupo de trabajo Amazon Redshift Serverless

Clúster aprovisionado de Amazon Redshift

En esta sección se describe la configuración para usar el clúster aprovisionado de Amazon Redshift como destino de transmisión de Firehose.

- Escriba valores en los siguientes campos:

  Clúster

  Clúster de Amazon Redshift en el que se copian los datos del bucket de S3. Configure el clúster de Amazon Redshift para que sea de acceso público y desbloquee las direcciones IP de Amazon Data Firehose. Para obtener más información, consulte Conceda a Amazon Data Firehose acceso a un destino de Amazon Redshift .

  Autenticación

  Puede elegir entre introducir el nombre de usuario y la contraseña directamente o recuperar el secreto AWS Secrets Manager para acceder al clúster de Amazon Redshift.

  - Nombre de usuario

    Especifique un usuario de Amazon Redshift con permisos para acceder al clúster de Amazon Redshift. Este usuario debe tener el permiso `INSERT` de Amazon Redshift para copiar datos del bucket de S3 en el clúster de Amazon Redshift.

  - Contraseña

    Especifique la contraseña del usuario que tiene permisos para acceder al clúster.

  - Secret

    Seleccione un secreto AWS Secrets Manager que contenga las credenciales del clúster de Amazon Redshift. Si no ve su secreto en la lista desplegable, cree uno AWS Secrets Manager para sus credenciales de Amazon Redshift. Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

  Base de datos

  Base de datos de Amazon Redshift en la que se copian los datos.

  Tabla

  Tabla de Amazon Redshift en la que se copian los datos.

  Columnas

  Las columnas específicas de la tabla donde se copian los datos (opcional). Utilice esta opción si la cantidad de columnas definida en los objetos de Amazon S3 es menor que la cantidad de columnas en la tabla Amazon Redshift.

Destino de S3 intermedio

Firehose envía primero los datos a su bucket de S3 y, a continuación, emite un comando de Amazon COPY Redshift para cargar los datos en su clúster de Amazon Redshift. Especifique el bucket de S3 de su propiedad adonde se deben entregar los datos de streaming. Cree un nuevo bucket de S3 o elija uno que le pertenezca.

Firehose no elimina los datos del bucket de S3 después de cargarlos en el clúster de Amazon Redshift. Puede administrar los datos en el bucket de S3 utilizando una configuración del ciclo de vida. Para obtener más información, consulte Administración del ciclo de vida de los objetos en la Guía del usuario de Amazon Simple Storage Service.

Prefijo de S3 intermedio

(Opcional) Para utilizar el prefijo predeterminado de los objetos de Amazon S3, deje esta opción en blanco. Firehose utiliza automáticamente un prefijo en formato de hora UTC «YYYY/MM/dd/HH» para los objetos Amazon S3 entregados. Puede añadir más elementos al comienzo de este prefijo. Para obtener más información, consulte Configurar el formato de nombre de objeto de Amazon S3.

Opciones de COPY

Parámetros que puede especificar en el comando COPY de Amazon Redshift. Podrían ser necesarios para la configuración. Por ejemplo, se requiere `GZIP` «» si la compresión de datos de Amazon S3 está habilitada. Se requiere `REGION` «» si el bucket de S3 no se encuentra en la misma AWS región que el clúster de Amazon Redshift. Para más información, consulte COPY en la Guía para desarrolladores de bases de datos de Amazon Redshift.

COPY command

Comando COPY de Amazon Redshift. Para más información, consulte COPY en la Guía para desarrolladores de bases de datos de Amazon Redshift.

Retry duration

Tiempo (0—7200 segundos) para que Firehose vuelva a intentarlo si fallan los datos del clúster de COPY Amazon Redshift. Firehose lo vuelve a intentar cada 5 minutos hasta que finalice la duración del reintento. Si estableces la duración del reintento en 0 (cero) segundos, Firehose no volverá a intentarlo si se produce un error en COPY el comando.

Sugerencias de almacenamiento en búfer

Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino varía de un proveedor de servicios a otro.

Compresión S3

Elija la compresión de datos GZIP, Snappy, Zip o Snappy compatible con Hadoop, o sin compresión de datos. La compresión Snappy compatible con Snappy, Zip y Hadoop no está disponible para las transmisiones de Firehose con Amazon Redshift como destino.

Formato de extensión de archivo S3 (opcional)

Formato de extensión de archivo S3 (opcional): especifique un formato de extensión de archivo para los objetos entregados al bucket de destino de Amazon S3. Si habilita esta función, la extensión de archivo especificada anulará las extensiones de archivo predeterminadas incorporadas por las funciones de conversión de formato de datos o de compresión de S3, como .parquet o .gz. Asegúrese de haber configurado la extensión de archivo correcta cuando utilice esta función con la conversión de formato de datos o la compresión S3. La extensión del archivo debe empezar con un punto (.) y puede contener los caracteres permitidos: 0-9a-z! -_.*' (). La extensión del archivo no puede superar los 128 caracteres.

Cifrado S3

Firehose admite el cifrado del lado del servidor de Amazon S3 con AWS Key Management Service (SSE-KMS) para cifrar los datos entregados en Amazon S3. Puede optar por utilizar el tipo de cifrado predeterminado especificado en el depósito S3 de destino o cifrar con una clave de la lista de claves de su propiedad. AWS KMS Si cifra los datos con AWS KMS claves, puede usar la clave AWS administrada predeterminada (aws/s3) o una clave administrada por el cliente. Para obtener más información, consulte Protección de datos mediante el cifrado del lado del servidor con claves administradas por AWS KMS (SSE-KMS).

Configurar los ajustes de destino para el grupo de trabajo Amazon Redshift Serverless

En esta sección se describe la configuración para usar el grupo de trabajo Amazon Redshift Serverless como destino de transmisión de Firehose.

- Escriba valores en los siguientes campos:

Nombre del grupo de trabajo

El grupo de trabajo de Amazon Redshift sin servidor en el que se copian los datos del bucket de S3. Configure el grupo de trabajo Amazon Redshift Serverless para que sea de acceso público y desbloquee las direcciones IP de Firehose. Para obtener más información, consulte la sección Conectarse a una instancia de Amazon Redshift sin servidor accesible públicamente en Conexión a Amazon Redshift sin servidor y también Conceda a Amazon Data Firehose acceso a un destino de Amazon Redshift .

Autenticación

Puede elegir entre introducir el nombre de usuario y la contraseña directamente o recuperar el secreto para acceder AWS Secrets Manager al grupo de trabajo Amazon Redshift Serverless.

- Nombre de usuario

  Especifique un usuario de Amazon Redshift con permisos para acceder al grupo de trabajo Amazon Redshift Serverless. Este usuario debe tener el permiso `INSERT` de Amazon Redshift para copiar datos del bucket de S3 en el grupo de trabajo de Amazon Redshift sin servidor.

- Contraseña

  Especifique la contraseña del usuario que tiene permisos para acceder al grupo de trabajo Amazon Redshift Serverless.

- Secret

  Seleccione un secreto AWS Secrets Manager que contenga las credenciales del grupo de trabajo Amazon Redshift Serverless. Si no ve su secreto en la lista desplegable, cree uno AWS Secrets Manager para sus credenciales de Amazon Redshift. Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

Base de datos

Base de datos de Amazon Redshift en la que se copian los datos.

Tabla

Tabla de Amazon Redshift en la que se copian los datos.

Columnas

Las columnas específicas de la tabla donde se copian los datos (opcional). Utilice esta opción si la cantidad de columnas definida en los objetos de Amazon S3 es menor que la cantidad de columnas en la tabla Amazon Redshift.

Destino de S3 intermedio

Amazon Data Firehose entrega primero los datos a su bucket de S3 y, a continuación, emite un comando de Amazon COPY Redshift para cargar los datos en su grupo de trabajo Amazon Redshift Serverless. Especifique el bucket de S3 de su propiedad adonde se deben entregar los datos de streaming. Cree un nuevo bucket de S3 o elija uno que le pertenezca.

Firehose no elimina los datos del bucket de S3 después de cargarlos en el grupo de trabajo Amazon Redshift Serverless. Puede administrar los datos en el bucket de S3 utilizando una configuración del ciclo de vida. Para obtener más información, consulte Administración del ciclo de vida de los objetos en la Guía del usuario de Amazon Simple Storage Service.

Prefijo de S3 intermedio

(Opcional) Para utilizar el prefijo predeterminado de los objetos de Amazon S3, deje esta opción en blanco. Firehose utiliza automáticamente un prefijo en formato de hora UTC «`YYYY/MM/dd/HH`» para los objetos Amazon S3 entregados. Puede añadir más elementos al comienzo de este prefijo. Para obtener más información, consulte Configurar el formato de nombre de objeto de Amazon S3.

Opciones de COPY

Parámetros que puede especificar en el comando COPY de Amazon Redshift. Podrían ser necesarios para la configuración. Por ejemplo, se requiere `GZIP` «» si la compresión de datos de Amazon S3 está habilitada. Se requiere `REGION` «» si su bucket de S3 no está en la misma AWS región que su grupo de trabajo Amazon Redshift Serverless. Para más información, consulte COPY en la Guía para desarrolladores de bases de datos de Amazon Redshift.

COPY command

Comando COPY de Amazon Redshift. Para más información, consulte COPY en la Guía para desarrolladores de bases de datos de Amazon Redshift.

Retry duration

Tiempo (0—7200 segundos) para que Firehose vuelva a intentarlo si fallan los datos de su grupo de trabajo COPY Amazon Redshift Serverless. Firehose lo vuelve a intentar cada 5 minutos hasta que finalice la duración del reintento. Si estableces la duración del reintento en 0 (cero) segundos, Firehose no volverá a intentarlo si se produce un error en COPY el comando.

Sugerencias de almacenamiento en búfer

Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino varía de un proveedor de servicios a otro.

Compresión S3

Elija la compresión de datos GZIP, Snappy, Zip o Snappy compatible con Hadoop, o sin compresión de datos. La compresión Snappy compatible con Snappy, Zip y Hadoop no está disponible para las transmisiones de Firehose con Amazon Redshift como destino.

Formato de extensión de archivo S3 (opcional)

Formato de extensión de archivo S3 (opcional): especifique un formato de extensión de archivo para los objetos entregados al bucket de destino de Amazon S3. Si habilita esta función, la extensión de archivo especificada anulará las extensiones de archivo predeterminadas incorporadas por las funciones de conversión de formato de datos o de compresión de S3, como .parquet o .gz. Asegúrese de haber configurado la extensión de archivo correcta cuando utilice esta función con la conversión de formato de datos o la compresión S3. La extensión del archivo debe empezar con un punto (.) y puede contener los caracteres permitidos: 0-9a-z! -_.*' (). La extensión del archivo no puede superar los 128 caracteres.

Cifrado S3

Firehose admite el cifrado del lado del servidor de Amazon S3 con AWS Key Management Service (SSE-KMS) para cifrar los datos entregados en Amazon S3. Puede optar por utilizar el tipo de cifrado predeterminado especificado en el depósito S3 de destino o cifrar con una clave de la lista de claves de su propiedad. AWS KMS Si cifra los datos con AWS KMS claves, puede usar la clave AWS administrada predeterminada (aws/s3) o una clave administrada por el cliente. Para obtener más información, consulte Protección de datos mediante el cifrado del lado del servidor con claves administradas por AWS KMS (SSE-KMS).

OpenSearch Configure los ajustes de destino del servicio

En esta sección se describen las opciones para usar el OpenSearch Servicio en su destino.

- Escriba valores en los siguientes campos:

  OpenSearch Dominio de servicio

  El dominio de OpenSearch servicio al que se envían sus datos.

  Índice

  El nombre del índice de OpenSearch servicios que se utilizará al indexar los datos en su clúster OpenSearch de servicios.

  Index rotation

  Elija si se debe rotar el índice OpenSearch de servicios y con qué frecuencia. Si la rotación de índices está habilitada, Amazon Data Firehose añade la marca de tiempo correspondiente al nombre del índice especificado y rota. Para obtener más información, consulte Configura la rotación del índice para el servicio OpenSearch.

  Tipo

  El nombre del tipo de OpenSearch servicio que se utilizará al indexar los datos en su clúster de servicios. OpenSearch Para Elasticsearch 7.x y OpenSearch 1.x, solo puede haber un tipo por índice. Si intentas especificar un tipo nuevo para un índice existente que ya tiene otro tipo, Firehose devuelve un error durante el tiempo de ejecución.

  Para Elasticsearch 7.x, deje este campo vacío.

  Retry duration

  Tiempo que tarda Firehose en volver a intentarlo si se produce un error en una solicitud de indexación. OpenSearch En este caso, Firehose lo vuelve a intentar cada 5 minutos hasta que caduque la duración del reintento. Para la duración del reintento, puedes establecer cualquier valor entre 0 y 7200 segundos.

  Una vez transcurrido el tiempo de reintento, Firehose envía los datos a Dead Letter Queue (DLQ), un depósito de errores de S3 configurado. En el caso de los datos que se envían a DLQ, hay que volver a llevarlos del depósito de errores de S3 configurado al destino. OpenSearch

Si quieres impedir que Firehose Stream entregue datos a DLQ debido a un tiempo de inactividad o al mantenimiento de OpenSearch los clústeres, puedes configurar la duración del reintento con un valor superior en segundos. Para aumentar el valor de duración del reintento por encima de los 7200 segundos, ponte en contacto con el servicio de asistencia.AWS

Tipo de DocumentID

Indica el método para configurar el ID de documento. Los métodos admitidos son el ID de documento generado por FireHose y el ID de documento generado por el OpenSearch servicio. El identificador del documento generado por Firehose es la opción predeterminada cuando el valor del identificador del documento no está establecido. OpenSearch La opción recomendada es el identificador de documento generado por el servicio, ya que permite realizar operaciones de escritura intensiva, como el análisis y la observabilidad de los registros, por lo que consume menos recursos de CPU en el dominio del OpenSearch servicio y, por lo tanto, mejora el rendimiento.

Destination VPC connectivity (Conectividad de VPC de destino)

Si su dominio OpenSearch de servicio está en una VPC privada, utilice esta sección para especificar esa VPC. Especifique también las subredes y subgrupos que desea que Amazon Data Firehose utilice cuando envíe datos a su dominio de servicio. OpenSearch Puede usar los mismos grupos de seguridad que usa el dominio del OpenSearch servicio. Si especifica diferentes grupos de seguridad, asegúrese de que permitan el tráfico HTTPS saliente al grupo de seguridad del dominio del OpenSearch servicio. Asegúrese también de que el grupo de seguridad del dominio de OpenSearch servicio permita el tráfico HTTPS desde los grupos de seguridad que especificó al configurar la transmisión de Firehose. Si utilizas el mismo grupo de seguridad tanto para la transmisión de Firehose como para el dominio de OpenSearch servicio, asegúrate de que la regla de entrada del grupo de seguridad permita el tráfico HTTPS. Para obtener más información acerca de las reglas de los grupos de seguridad, consulte Reglas del grupo de seguridad en la documentación de Amazon VPC.

> **Important**
>
> Cuando especifique subredes para entregar datos al destino en una VPC privada, asegúrese de tener una cantidad suficiente de direcciones IP libres en las subredes elegidas. Si no hay una dirección IP libre disponible en una subred específica,

Firehose no puede crear ni añadir ENI para la entrega de datos en la VPC privada y la entrega se degradará o fallará.

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino varía de un proveedor de servicios a otro.

Configure los ajustes de destino para Serverless OpenSearch

En esta sección se describen las opciones para usar OpenSearch Serverless como destino.

- Escriba valores en los siguientes campos:

  OpenSearch Colección Serverless

  El punto final de un grupo de índices OpenSearch sin servidor al que se envían los datos.

  Índice

  El nombre del índice OpenSearch Serverless que se utilizará al indexar los datos para su colección Serverless. OpenSearch

  Destination VPC connectivity (Conectividad de VPC de destino)

  Si su colección OpenSearch sin servidor está en una VPC privada, utilice esta sección para especificar esa VPC. Especifique también las subredes y subgrupos que desea que Amazon Data Firehose utilice cuando envíe datos a su colección Serverless. OpenSearch

  > **Important**
  >
  > Cuando especifique subredes para entregar datos al destino en una VPC privada, asegúrese de tener una cantidad suficiente de direcciones IP libres en las subredes elegidas. Si no hay una dirección IP libre disponible en una subred específica, Firehose no puede crear ni añadir ENI para la entrega de datos en la VPC privada y la entrega se degradará o fallará.

Retry duration

Tiempo que tarda Firehose en volver a intentarlo si se produce un error en una solicitud de índice a OpenSearch Serverless. En este caso, Firehose lo vuelve a intentar cada 5 minutos hasta que caduque la duración del reintento. Para la duración del reintento, puedes establecer cualquier valor entre 0 y 7200 segundos.

Una vez transcurrido el tiempo de reintento, Firehose envía los datos a Dead Letter Queue (DLQ), un depósito de errores de S3 configurado. En el caso de los datos que se envían a DLQ, hay que volver a llevarlos del depósito de errores de S3 configurado a un destino sin servidor. OpenSearch

Si quieres impedir que Firehose Stream entregue datos a DLQ debido a un tiempo de inactividad o al mantenimiento de los clústeres OpenSearch sin servidor, puedes configurar la duración del reintento con un valor superior en segundos. Para aumentar el valor de la duración del reintento por encima de los 7200 segundos, ponte en contacto con el servicio de asistencia.AWS

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino varía de un proveedor de servicios a otro.

Configure los ajustes de destino para el punto final HTTP

En esta sección se describen las opciones de uso de un punto de conexión HTTP como destino.

> **Important**
>
> Si elige un punto de conexión HTTP como destino, revise y siga las instrucciones que se ofrecen en Apéndice: especificaciones de solicitudes y respuestas de entrega de puntos de conexión HTTP.

- Proporcione valores para los siguientes campos:

Nombre del punto de conexión HTTP: opcional

Especifique un nombre fácil de recordar para el punto de conexión HTTP. Por ejemplo, `My HTTP Endpoint Destination`.

URL del punto de conexión HTTP

Especifique la URL del punto de conexión HTTP en el siguiente formato: `https://xyz.httpendpoint.com`. La URL debe ser una URL HTTPS.

Autenticación

Puede elegir entre introducir la clave de acceso directamente o recuperar el secreto AWS Secrets Manager para acceder al punto final HTTP.

- (Opcional) Clave de acceso

  Ponte en contacto con el propietario del terminal si necesitas obtener la clave de acceso para permitir la entrega de datos a su punto final desde Firehose.

- Secret

  Seleccione un secreto AWS Secrets Manager que contenga la clave de acceso para el punto final HTTP. Si no ve su secreto en la lista desplegable, cree uno AWS Secrets Manager para la clave de acceso. Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

Codificación de contenidos

Amazon Data Firehose utiliza la codificación de contenido para comprimir el cuerpo de una solicitud antes de enviarla al destino. Seleccione GZIP o Deshabilitada para habilitar o deshabilitar la codificación de contenidos de su solicitud.

Retry duration

Especifique durante cuánto tiempo Amazon Data Firehose volverá a intentar enviar datos al punto de enlace HTTP seleccionado.

Tras enviar los datos, Amazon Data Firehose espera primero una confirmación del punto de conexión HTTP. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso,

Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Amazon Data Firehose envía datos al punto de enlace HTTP (ya sea el intento inicial o un reintento), reinicia el contador de tiempo de espera de confirmación y espera un acuse de recibo del punto de enlace HTTP.

Incluso si la duración del reintento vence, Amazon Data Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota el tiempo de espera de la confirmación, Amazon Data Firehose determina si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Si no quieres que Amazon Data Firehose vuelva a intentar enviar datos, establece este valor en 0.

Parámetros: opcional

Amazon Data Firehose incluye estos pares clave-valor en cada llamada HTTP. Estos parámetros pueden ayudarlo a identificar y organizar sus destinos.

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino varía de un proveedor de servicios a otro.

> **Important**
>
> Para los destinos de punto final HTTP, si ves 413 códigos de respuesta del punto final de destino en CloudWatch Logs, reduce el tamaño de la sugerencia de almacenamiento en búfer en la transmisión de Firehose e inténtalo de nuevo.

Configura los ajustes de destino para Datadog

En esta sección se describen las opciones de uso de Datadog como destino. Para obtener más información sobre Datadog, consulte https://docs.datadoghq.com/integrations/amazon_web_services/.

- Proporcione valores para los siguientes campos.

  URL del punto de conexión HTTP

  Elija dónde quiere enviar los datos desde una de las siguientes opciones del menú desplegable.

  - Registros de Datadog: US1
  - Registros de Datadog: US3
  - Registros de Datadog: US5
  - Registros de Datadog: AP1
  - Registros de Datadog: UE
  - Registros de Datadog: GOV
  - Métricas de Datadog: EE. UU.
  - Métricas de Datadog: US5
  - Métricas de Datadog: AP1
  - Métricas de Datadog: UE
  - Configuraciones de Datadog: US1
  - Configuraciones de Datadog: US3
  - Configuraciones de Datadog: US5
  - Configuraciones de Datadog: AP1
  - Configuraciones de Datadog: UE
  - Configuraciones de Datadog - US GOV

  Autenticación

  Puede elegir entre introducir la clave de API directamente o recuperar el secreto para acceder AWS Secrets Manager a Datadog.

  - Clave de API

    Póngase en contacto con Datadog para obtener la clave de API que necesita para permitir la entrega de datos a este punto final desde Firehose.

  - Secret

Seleccione un secreto AWS Secrets Manager que contenga la clave de API de Datadog. Si no ve su secreto en la lista desplegable, cree uno en. AWS Secrets Manager Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

Codificación de contenidos

Amazon Data Firehose utiliza la codificación de contenido para comprimir el cuerpo de una solicitud antes de enviarla al destino. Seleccione GZIP o Deshabilitada para habilitar o deshabilitar la codificación de contenidos de su solicitud.

Retry duration

Especifique durante cuánto tiempo Amazon Data Firehose volverá a intentar enviar datos al punto de enlace HTTP seleccionado.

Tras enviar los datos, Amazon Data Firehose espera primero una confirmación del punto de conexión HTTP. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Amazon Data Firehose envía datos al punto de enlace HTTP (ya sea el intento inicial o un reintento), reinicia el contador de tiempo de espera de confirmación y espera un acuse de recibo del punto de enlace HTTP.

Incluso si la duración del reintento vence, Amazon Data Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota el tiempo de espera de la confirmación, Amazon Data Firehose determina si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Si no quieres que Amazon Data Firehose vuelva a intentar enviar datos, establece este valor en 0.

Parámetros: opcional

Amazon Data Firehose incluye estos pares clave-valor en cada llamada HTTP. Estos parámetros pueden ayudarlo a identificar y organizar sus destinos.

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino varía de un proveedor de servicios a otro.

Configure los ajustes de destino para Honeycomb

En esta sección se describen las opciones de uso de Honeycomb como destino. Para obtener más información sobre Honeycomb, consulte https://docs.honeycomb.io/ getting-data-in aws-cloudwatch-metrics /metrics/.

- Proporcione valores para los siguientes campos:

  Punto de conexión entre Honeycomb y Kinesis

  Especifique la URL del punto de conexión HTTP en el siguiente formato: https://api.honeycomb.io/1/kinesis_events/{{dataset}}

  Autenticación

  Puede elegir entre introducir la clave de API directamente o recuperar el secreto para acceder a Honeycomb. AWS Secrets Manager

  - Clave de API

    Póngase en contacto con Honeycomb para obtener la clave de API que necesita para permitir la entrega de datos a este punto final desde Firehose.

  - Secret

    Seleccione un secreto AWS Secrets Manager que contenga la clave de API de Honeycomb. Si no ve su secreto en la lista desplegable, cree uno en AWS Secrets Manager. Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

  Codificación de contenidos

  Amazon Data Firehose utiliza la codificación de contenido para comprimir el cuerpo de una solicitud antes de enviarla al destino. Elija GZIP para habilitar la codificación del contenido de su solicitud. Esta es la opción recomendada para el destino de Honeycomb.

Retry duration

Especifique durante cuánto tiempo Amazon Data Firehose volverá a intentar enviar datos al punto de enlace HTTP seleccionado.

Tras enviar los datos, Amazon Data Firehose espera primero una confirmación del punto de conexión HTTP. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Amazon Data Firehose envía datos al punto de enlace HTTP (ya sea el intento inicial o un reintento), reinicia el contador de tiempo de espera de confirmación y espera un acuse de recibo del punto de enlace HTTP.

Incluso si la duración del reintento vence, Amazon Data Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota el tiempo de espera de la confirmación, Amazon Data Firehose determina si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Si no quieres que Amazon Data Firehose vuelva a intentar enviar datos, establece este valor en 0.

Parámetros: opcional

Amazon Data Firehose incluye estos pares clave-valor en cada llamada HTTP. Estos parámetros pueden ayudarlo a identificar y organizar sus destinos.

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino varía de un proveedor de servicios a otro.

Configure los ajustes de destino para Coralogix

En esta sección se describen las opciones de uso de Coralogix como destino. Para obtener más información sobre Coralogix, consulte https://coralogix.com/integrations/aws-firehose.

- Proporcione valores para los siguientes campos:

  URL del punto de conexión HTTP

  Elija la URL del punto de conexión HTTP entre las siguientes opciones del menú desplegable:

  - Coralogix: EE. UU.
  - Coralogix: SINGAPUR
  - Coralogix: IRLANDA
  - Coralogix: INDIA
  - Coralogix: ESTOCOLMO

  Autenticación

  Puede elegir entre introducir la clave privada directamente o recuperar el secreto para acceder AWS Secrets Manager a Coralogix.

  - Clave privada

    Póngase en contacto con Coralogix para obtener la clave privada que necesita para permitir la entrega de datos a este punto final desde Firehose.

  - Secret

    Seleccione un secreto AWS Secrets Manager que contenga la clave privada de Coralogix. Si no ve su secreto en la lista desplegable, cree uno en. AWS Secrets Manager Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

  Codificación de contenidos

  Amazon Data Firehose utiliza la codificación de contenido para comprimir el cuerpo de una solicitud antes de enviarla al destino. Elija GZIP para habilitar la codificación del contenido de su solicitud. Esta es la opción recomendada para el destino de Coralogix.

  Retry duration

  Especifique durante cuánto tiempo Amazon Data Firehose volverá a intentar enviar datos al punto de enlace HTTP seleccionado.

Tras enviar los datos, Amazon Data Firehose espera primero una confirmación del punto de conexión HTTP. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Amazon Data Firehose envía datos al punto de enlace HTTP (ya sea el intento inicial o un reintento), reinicia el contador de tiempo de espera de confirmación y espera un acuse de recibo del punto de enlace HTTP.

Incluso si la duración del reintento vence, Amazon Data Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota el tiempo de espera de la confirmación, Amazon Data Firehose determina si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Si no quieres que Amazon Data Firehose vuelva a intentar enviar datos, establece este valor en 0.

Parámetros: opcional

Amazon Data Firehose incluye estos pares clave-valor en cada llamada HTTP. Estos parámetros pueden ayudarlo a identificar y organizar sus destinos.

- applicationName: entorno en el que se ejecuta Data Firehose
- subsystemName: nombre de la integración de Data Firehose
- ComputerName: el nombre de la transmisión Firehose en uso

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño de búfer recomendado para el destino varía en función del proveedor de servicios.

Configure los ajustes de destino para Dynatrace

En esta sección se describen las opciones de uso de Dynatrace como destino. Para obtener más información, consulte https://www.dynatrace.com/support/help/technology-support/cloud-platforms/amazon-web-services /integrations//cloudwatch-metric-streams.

- Elige opciones para usar Dynatrace como destino para tu transmisión de Firehose.

  Tipo de ingestión

  Elija si quiere entregar métricas o registros (por defecto) en Dynatrace para su posterior análisis y procesamiento.

  URL del punto de conexión HTTP

  Elija la URL del punto de conexión HTTP (Dynatrace US, Dynatrace EU o Dynatrace Global) en el menú desplegable.

  Autenticación

  Puedes elegir entre introducir el token de la API directamente o recuperar el secreto para acceder a Dynatrace. AWS Secrets Manager

  - Token de la API

    Genera el token de la API de Dynatrace que necesitas para habilitar la entrega de datos a este punto final desde Firehose. Para obtener más información, consulte API de Dynatrace: Tokens y autenticación.

  - Secret

    Seleccione un secreto AWS Secrets Manager que contenga el token de API de Dynatrace. Si no ve su secreto en la lista desplegable, cree uno en. AWS Secrets Manager Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

  URL de la API

  Proporcione la URL de la API de su entorno de Dynatrace.

  Codificación de contenidos

  Elija si desea habilitar la codificación de contenido para comprimir el cuerpo de la solicitud. Amazon Data Firehose utiliza la codificación de contenido para comprimir el cuerpo de una

solicitud antes de enviarla al destino. Cuando está habilitada, el contenido se comprime en formato GZIP.

Retry duration

Especifique durante cuánto tiempo Firehose volverá a intentar enviar datos al punto final HTTP seleccionado.

Tras enviar los datos, Firehose espera primero una confirmación del punto final HTTP. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Firehose envía datos al punto final HTTP, ya sea durante el intento inicial o después de volver a intentarlo, reinicia el contador de tiempo de espera de confirmación y espera un acuse de recibo del punto final HTTP.

Incluso si la duración del reintento vence, Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota el tiempo de espera de la confirmación, Firehose determina si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Si no quieres que Firehose vuelva a intentar enviar datos, establece este valor en 0.

Parámetros: opcional

Amazon Data Firehose incluye estos pares clave-valor en cada llamada HTTP. Estos parámetros pueden ayudarlo a identificar y organizar sus destinos.

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. Las sugerencias del búfer incluyen el tamaño y el intervalo del búfer de las transmisiones. El tamaño de búfer recomendado para el destino varía según el proveedor de servicios.

Configure los ajustes de destino para LogicMonitor

En esta sección se describen las opciones que puede utilizar LogicMonitorpara su destino. Para obtener más información, consulte https://www.logicmonitor.com.

- Proporcione valores para los siguientes campos:

  URL del punto de conexión HTTP

  Especifique la URL del punto final HTTP en el siguiente formato.

  ```
  https://ACCOUNT.logicmonitor.com
  ```

  Autenticación

  Puedes elegir entre introducir la clave de API directamente o recuperar el secreto desde donde AWS Secrets Manager accedes LogicMonitor.

  - Clave de API

    Póngase en contacto LogicMonitor para obtener la clave de API que necesita para habilitar la entrega de datos a este punto final desde Firehose.

  - Secret

    Seleccione un secreto AWS Secrets Manager que contenga la clave de API. LogicMonitor Si no ves tu secreto en la lista desplegable, crea uno en AWS Secrets Manager. Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

  Codificación de contenidos

  Amazon Data Firehose utiliza la codificación de contenido para comprimir el cuerpo de una solicitud antes de enviarla al destino. Seleccione GZIP o Deshabilitada para habilitar o deshabilitar la codificación de contenidos de su solicitud.

  Retry duration

  Especifique durante cuánto tiempo Amazon Data Firehose volverá a intentar enviar datos al punto de enlace HTTP seleccionado.

Tras enviar los datos, Amazon Data Firehose espera primero una confirmación del punto de conexión HTTP. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Amazon Data Firehose envía datos al punto de enlace HTTP (ya sea el intento inicial o un reintento), reinicia el contador de tiempo de espera de confirmación y espera un acuse de recibo del punto de enlace HTTP.

Incluso si la duración del reintento vence, Amazon Data Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota el tiempo de espera de la confirmación, Amazon Data Firehose determina si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Si no quieres que Amazon Data Firehose vuelva a intentar enviar datos, establece este valor en 0.

Parámetros: opcional

Amazon Data Firehose incluye estos pares clave-valor en cada llamada HTTP. Estos parámetros pueden ayudarlo a identificar y organizar sus destinos.

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino varía de un proveedor de servicios a otro.

Configure los ajustes de destino para Logz.io

En esta sección se describen las opciones de uso de Logz.io como destino. Para obtener más información, consulte https://logz.io/.

> **Note**
>
> En la región de Europa (Milán), Logz.io no es compatible como destino de Amazon Data Firehose.

- Proporcione valores para los siguientes campos:

  URL del punto de conexión HTTP

  Especifique la URL del punto final HTTP en el siguiente formato. La URL debe ser una HTTPS URL.

  ```
  https://listener-aws-metrics-stream-<region>.logz.io/
  ```

  Por ejemplo

  ```
  https://listener-aws-metrics-stream-us.logz.io/
  ```

  Autenticación

  Puedes elegir entre introducir el token de envío directamente o recuperar el secreto AWS Secrets Manager para acceder a Logz.io.

  - Token de envío

    Póngase en contacto con Logz.io para obtener el token de envío que necesita para permitir la entrega de datos a este punto final desde Firehose.

  - Secret

    Selecciona un secreto AWS Secrets Manager que contenga el token de envío de Logz.io. Si no ves tu secreto en la lista desplegable, crea uno en. AWS Secrets Manager Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

  Retry duration

  Especifica durante cuánto tiempo Amazon Data Firehose vuelve a intentar enviar datos a Logz.io.

Tras enviar los datos, Amazon Data Firehose espera primero una confirmación del punto de conexión HTTP. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Amazon Data Firehose envía datos al punto de enlace HTTP (ya sea el intento inicial o un reintento), reinicia el contador de tiempo de espera de confirmación y espera un acuse de recibo del punto de enlace HTTP.

Incluso si la duración del reintento vence, Amazon Data Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota el tiempo de espera de la confirmación, Amazon Data Firehose determina si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Si no quieres que Amazon Data Firehose vuelva a intentar enviar datos, establece este valor en 0.

Parámetros: opcional

Amazon Data Firehose incluye estos pares clave-valor en cada llamada HTTP. Estos parámetros pueden ayudarlo a identificar y organizar sus destinos.

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino varía de un proveedor de servicios a otro.

Configurar los ajustes de destino para MongoDB Cloud

En esta sección se describen las opciones de uso de MongoDB Cloud como destino. Para obtener más información, consulte https://www.mongodb.com.

- Proporcione valores para los siguientes campos:

URL del webhook de MongoDB Realm

Especifique la URL del punto final HTTP en el siguiente formato.

```
https://webhooks.mongodb-realm.com
```

La URL debe ser una HTTPS URL.

Autenticación

Puede elegir entre introducir la clave de API directamente o recuperar el secreto AWS Secrets Manager para acceder a MongoDB Cloud.

- Clave de API

  Póngase en contacto con MongoDB Cloud para obtener la clave de API que necesita para permitir la entrega de datos a este punto final desde Firehose.

- Secret

  Seleccione un secreto AWS Secrets Manager que contenga la clave de API para MongoDB Cloud. Si no ve su secreto en la lista desplegable, cree uno en. AWS Secrets Manager Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

Codificación de contenidos

Amazon Data Firehose utiliza la codificación de contenido para comprimir el cuerpo de una solicitud antes de enviarla al destino. Seleccione GZIP o Deshabilitada para habilitar o deshabilitar la codificación de contenidos de su solicitud.

Retry duration

Especifique durante cuánto tiempo Amazon Data Firehose volverá a intentar enviar datos al proveedor externo seleccionado.

Tras enviar los datos, Amazon Data Firehose espera primero una confirmación del punto de conexión HTTP. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Amazon Data Firehose envía datos al punto de enlace HTTP (ya sea el intento inicial o un reintento), reinicia el contador de tiempo de espera de confirmación y espera un acuse de recibo del punto de enlace HTTP.

Incluso si la duración del reintento vence, Amazon Data Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota el tiempo de espera de la confirmación, Amazon Data Firehose determina si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Si no quieres que Amazon Data Firehose vuelva a intentar enviar datos, establece este valor en 0.

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino varía de un proveedor de servicios a otro.

Parámetros: opcional

Amazon Data Firehose incluye estos pares clave-valor en cada llamada HTTP. Estos parámetros pueden ayudarlo a identificar y organizar sus destinos.

Configure los ajustes de destino de New Relic

En esta sección se describen las opciones de uso de New Relic como destino. Para obtener más información, consulte https://newrelic.com.

- Proporcione valores para los siguientes campos:

  URL del punto de conexión HTTP

  Elija la URL del punto de conexión HTTP entre las siguientes opciones de la lista desplegable.

  - Registros de New Relic: EE. UU.
  - Métricas de New Relic: EE. UU.
  - Métricas de New Relic: UE

Autenticación

Puedes elegir entre introducir la clave de API directamente o recuperar el secreto AWS Secrets Manager para acceder a New Relic.

- Clave de API

  Introduce tu clave de licencia, que es una cadena hexadecimal de 40 caracteres, desde la configuración de tu cuenta de New Relic One. Necesitas esta clave de API para permitir la entrega de datos a este punto final desde Firehose.

- Secret

  Seleccione un secreto AWS Secrets Manager que contenga la clave de API de New Relic. Si no ves tu secreto en la lista desplegable, crea uno en AWS Secrets Manager. Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

Codificación de contenidos

Amazon Data Firehose utiliza la codificación de contenido para comprimir el cuerpo de una solicitud antes de enviarla al destino. Seleccione GZIP o Deshabilitada para habilitar o deshabilitar la codificación de contenidos de su solicitud.

Retry duration

Especifique durante cuánto tiempo Amazon Data Firehose vuelve a intentar enviar datos al punto de enlace HTTP de New Relic.

Tras enviar los datos, Amazon Data Firehose espera primero una confirmación del punto de conexión HTTP. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Amazon Data Firehose envía datos al punto de enlace HTTP (ya sea el intento inicial o un reintento), reinicia el contador de tiempo de espera de confirmación y espera un acuse de recibo del punto de enlace HTTP.

Incluso si la duración del reintento vence, Amazon Data Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota

el tiempo de espera de la confirmación, Amazon Data Firehose determina si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Si no quieres que Amazon Data Firehose vuelva a intentar enviar datos, establece este valor en 0.

Parámetros: opcional

Amazon Data Firehose incluye estos pares clave-valor en cada llamada HTTP. Estos parámetros pueden ayudarlo a identificar y organizar sus destinos.

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino varía de un proveedor de servicios a otro.

Configure los ajustes de destino para Snowflake

En esta sección se describen las opciones para usar Snowflake como destino.

> **Note**
>
> La integración de Firehose con Snowflake está disponible en EE. UU. Este (Norte de Virginia), EE. UU. Oeste (Oregón), Europa (Irlanda), EE. UU. Este (Ohio), Asia Pacífico (Tokio), Europa (Fráncfort), Asia Pacífico (Singapur), Asia Pacífico (Seúl) y Asia Pacífico (Sídney). Regiones de AWS

Ajustes de conexión

- Proporcione valores para los siguientes campos:

  URL de la cuenta de Snowflake

  Especifique la URL de una cuenta de Snowflake. Por ejemplo: `xy12345.us-east-1.aws.snowflakecomputing.com`. Consulte la documentación de Snowflake para saber cómo determinar la URL de su cuenta. Tenga en cuenta que no debe especificar el número de puerto, mientras que el protocolo (https://) es opcional.

Autenticación

Puede elegir entre introducir el nombre de usuario, la clave privada y la frase de contraseña manualmente o recuperar el secreto para acceder a Snowflake. AWS Secrets Manager

- Inicio de sesión de usuario

  Especifique el usuario de Snowflake que se utilizará para cargar los datos. Asegúrese de que el usuario tiene acceso para insertar datos en la tabla Snowflake.

- Clave privada

  Especifique la clave privada del usuario utilizada para la autenticación con Snowflake. Asegúrese de que la clave privada esté en `PKCS8` formato. No incluya el encabezado y el pie de página PEM como parte de esta clave. Si la clave está dividida en varias líneas, elimine los saltos de línea.

- Contraseña

  Especifique la contraseña para descifrar la clave privada cifrada. Puede dejar este campo vacío si la clave privada no está cifrada. Para obtener más información, consulte Uso de la autenticación de pares de claves y la rotación de claves.

- Secret

  Seleccione un secreto AWS Secrets Manager que contenga las credenciales de Snowflake. Si no ve su secreto en la lista desplegable, cree uno en. AWS Secrets Manager Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

Configuración de roles

Usar el rol de Snowflake predeterminado: si se selecciona esta opción, Firehose no pasará ningún rol a Snowflake. Se asume que el rol predeterminado carga los datos. Asegúrese de que el rol predeterminado tenga permiso para insertar datos en la tabla de Snowflake.

Usa un rol de copo de nieve personalizado: introduce un rol de copo de nieve no predeterminado para que lo asuma Firehose al cargar datos en la tabla de copos de nieve.

Conectividad con Snowflake

Las opciones son privadas o públicas.

ID de VPCE privado (opcional)

El identificador VPCE para que Firehose se conecte de forma privada con Snowflake. El formato de ID es com.amazonaws.vpce. *[región] .vpce-svc- [id].* Para obtener más información, consulte & Snowflake.AWS PrivateLink

> **Note**
>
> Asegúrese de que su red Snowflake permita el acceso a Firehose. Para obtener una lista de los ID de VPCE que puede utilizar, consulte la. Acceso a Snowflake en VPC

Configuración de las bases de datos

- Debes especificar los siguientes ajustes para usar Snowflake como destino de tu transmisión Firehose.
  - Base de datos de Snowflake: todos los datos de Snowflake se guardan en bases de datos.
  - Esquema de Snowflake: cada base de datos consta de uno o más esquemas, que son agrupaciones lógicas de objetos de base de datos, como tablas y vistas
  - Tabla Snowflake: todos los datos de Snowflake se almacenan en tablas de bases de datos, estructuradas de forma lógica como conjuntos de columnas y filas.

Opciones de carga de datos para la tabla de Snowflake

- Utilice claves JSON como nombres de columnas
- Usa columnas VARIANT
  - Nombre de la columna de contenido: especifique un nombre de columna en la tabla, donde se deben cargar los datos sin procesar.
  - Nombre de la columna de metadatos (opcional): especifique un nombre de columna en la tabla, donde se debe cargar la información de los metadatos.

Retry duration

Tiempo (0—7200 segundos) para que Firehose vuelva a intentarlo si la apertura del canal o la entrega a Snowflake fallan debido a problemas con el servicio de Snowflake. Firehose lo vuelve a intentar con un retraso exponencial hasta que finalice la duración del reintento. Si estableces

la duración del reintento en 0 (cero) segundos, Firehose no volverá a intentarlo en caso de que Snowflake falle y enrutará los datos al depósito de errores de Amazon S3.

Configure los ajustes de destino para Splunk

Esta sección describe las opciones de uso de Splunk como destino.

> **Note**
>
> Firehose entrega los datos a los clústeres de Splunk configurados con Classic Load Balancer o Application Load Balancer.

- Proporcione valores para los siguientes campos:

  Splunk cluster endpoint

  Para determinar el punto final, consulte Configurar Amazon Data Firehose para enviar datos a la plataforma Splunk en la documentación de Splunk.

  Splunk endpoint type

  Elija `Raw endpoint` en la mayoría de los casos. Elija `Event endpoint` si ha preprocesado sus datos AWS Lambda para enviarlos a diferentes índices por tipo de evento. Para obtener información sobre qué punto de conexión usar, consulte Configurar Amazon Data Firehose para enviar datos a la plataforma Splunk en la documentación de Splunk.

  Autenticación

  Puede elegir entre introducir el token de autenticación directamente o recuperar el secreto para acceder AWS Secrets Manager a Splunk.

  - Authentication token

    Para configurar un punto final de Splunk que pueda recibir datos de Amazon Data Firehose, consulte la descripción general de instalación y configuración del complemento de Splunk para Amazon Data Firehose en la documentación de Splunk. Guarda el token que obtengas de Splunk al configurar el punto final para esta transmisión de Firehose y agrégalo aquí.

  - Secret

Selecciona un secreto AWS Secrets Manager que contenga el token de autenticación de Splunk. Si no ve su secreto en la lista desplegable, cree uno en AWS Secrets Manager. Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

HEC acknowledgement timeout

Especifique cuánto tiempo espera Amazon Data Firehose a que Splunk acuse de recibo del índice. Si Splunk no envía el acuse de recibo antes de que se agote el tiempo de espera, Amazon Data Firehose considerará que se trata de un error en la entrega de datos. A continuación, Amazon Data Firehose vuelve a intentarlo o hace una copia de seguridad de los datos en su bucket de Amazon S3, en función del valor de duración del reintento que haya establecido.

Retry duration

Especifique durante cuánto tiempo Amazon Data Firehose volverá a intentar enviar datos a Splunk.

Tras enviar los datos, Amazon Data Firehose espera primero el acuse de recibo de Splunk. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Amazon Data Firehose envía datos a Splunk (ya sea en el intento inicial o en un reintento), reinicia el contador de tiempo de espera de confirmación y espera una confirmación de Splunk.

Incluso si la duración del reintento vence, Amazon Data Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota el tiempo de espera de la confirmación, Amazon Data Firehose determina si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Si no quieres que Amazon Data Firehose vuelva a intentar enviar datos, establece este valor en 0.

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño de búfer recomendado para el destino varía en función del proveedor de servicios.

Configure los ajustes de destino para Splunk Observability Cloud

En esta sección se describen las opciones de uso de Splunk Observability Cloud como destino. Para obtener más información, consulte https://docs.splunk.com/observability/en/gdi/ get-data-in /connect/aws/aws-apiconfig.html # connect-to-aws-using - the-splunk-observability-cloud -api.

- Proporcione valores para los siguientes campos:

  URL del punto de conexión de ingesta de la nube

  Puede encontrar la URL de ingesta de datos en tiempo real de Splunk Observability Cloud en Profile > Organizations > Real-time Data Ingest Endpoint, en la consola de Splunk Observability.

  Autenticación

  Puede elegir entre introducir el token de acceso directamente o recuperar el secreto para acceder AWS Secrets Manager a Splunk Observability Cloud.

  - Token de acceso

    Copia tu token de acceso a Splunk Observability con el ámbito de autorización de INGEST desde los Tokens de acceso, en la sección Configuración de la consola de Splunk Observability.

  - Secret

    Selecciona un secreto AWS Secrets Manager que contenga el token de acceso a Splunk Observability Cloud. Si no ve su secreto en la lista desplegable, cree uno en. AWS Secrets Manager Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

Codificación de contenidos

Amazon Data Firehose utiliza la codificación de contenido para comprimir el cuerpo de una solicitud antes de enviarla al destino. Seleccione GZIP o Deshabilitada para habilitar o deshabilitar la codificación de contenidos de su solicitud.

Retry duration

Especifique durante cuánto tiempo Amazon Data Firehose volverá a intentar enviar datos al punto de enlace HTTP seleccionado.

Tras enviar los datos, Amazon Data Firehose espera primero una confirmación del punto de conexión HTTP. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Amazon Data Firehose envía datos al punto de enlace HTTP (ya sea el intento inicial o un reintento), reinicia el contador de tiempo de espera de confirmación y espera un acuse de recibo del punto de enlace HTTP.

Incluso si la duración del reintento vence, Amazon Data Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota el tiempo de espera de la confirmación, Amazon Data Firehose determina si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Si no quieres que Amazon Data Firehose vuelva a intentar enviar datos, establece este valor en 0.

Parámetros: opcional

Amazon Data Firehose incluye estos pares clave-valor en cada llamada HTTP. Estos parámetros pueden ayudarlo a identificar y organizar sus destinos.

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino varía de un proveedor de servicios a otro.

Configure los ajustes de destino para Sumo Logic

En esta sección se describen las opciones de uso de Sumo Logic como destino. Para obtener más información, consulte https://www.sumologic.com.

- Proporcione valores para los siguientes campos:

  URL del punto de conexión HTTP

  Especifique la URL del punto de conexión HTTP en el siguiente formato: `https://deployment name.sumologic.net/receiver/v1/kinesis/dataType/access token`. La URL debe ser una URL HTTPS.

  Codificación de contenidos

  Amazon Data Firehose utiliza la codificación de contenido para comprimir el cuerpo de una solicitud antes de enviarla al destino. Seleccione GZIP o Deshabilitada para habilitar o deshabilitar la codificación de contenidos de su solicitud.

  Retry duration

  Especifique durante cuánto tiempo Amazon Data Firehose volverá a intentar enviar datos a Sumo Logic.

  Tras enviar los datos, Amazon Data Firehose espera primero una confirmación del punto de conexión HTTP. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

  Cada vez que Amazon Data Firehose envía datos al punto de enlace HTTP (ya sea el intento inicial o un reintento), reinicia el contador de tiempo de espera de confirmación y espera un acuse de recibo del punto de enlace HTTP.

  Incluso si la duración del reintento vence, Amazon Data Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota el tiempo de espera de la confirmación, Amazon Data Firehose determina si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Si no quieres que Amazon Data Firehose vuelva a intentar enviar datos, establece este valor en 0.

Parámetros: opcional

Amazon Data Firehose incluye estos pares clave-valor en cada llamada HTTP. Estos parámetros pueden ayudarlo a identificar y organizar sus destinos.

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño del búfer recomendado para el destino de Elastic varía de un proveedor de servicios a otro.

Configura los ajustes de destino para Elastic

En esta sección se describen las opciones de uso de Elastic como destino.

- Proporcione valores para los siguientes campos:

  URL del punto de conexión de Elastic

  Especifique la URL del punto de conexión HTTP en el siguiente formato: `https://<cluster-id>.es.<region>.aws.elastic-cloud.com`. La URL debe ser una URL HTTPS.

  Autenticación

  Puedes elegir entre introducir la clave de API directamente o recuperar el secreto AWS Secrets Manager para acceder a Elastic.

  - Clave de API

    Ponte en contacto con Elastic para obtener la clave de API que necesitas para habilitar la entrega de datos a su servicio desde Firehose.

  - Secret

    Selecciona un secreto AWS Secrets Manager que contenga la clave de API de Elastic. Si no ve su secreto en la lista desplegable, cree uno en AWS Secrets Manager. Para obtener más información, consulte Autenticate con AWS Secrets Manager Amazon Data Firehose.

Codificación de contenidos

Amazon Data Firehose utiliza la codificación de contenido para comprimir el cuerpo de una solicitud antes de enviarla al destino. Seleccione GZIP (que es la opción seleccionada de forma predeterminada) o Deshabilitada para habilitar o deshabilitar la codificación de contenidos de su solicitud.

Retry duration

Especifique durante cuánto tiempo Amazon Data Firehose volverá a intentar enviar datos a Elastic.

Tras enviar los datos, Amazon Data Firehose espera primero una confirmación del punto de conexión HTTP. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Amazon Data Firehose envía datos al punto de enlace HTTP (ya sea el intento inicial o un reintento), reinicia el contador de tiempo de espera de confirmación y espera un acuse de recibo del punto de enlace HTTP.

Incluso si la duración del reintento vence, Amazon Data Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota el tiempo de espera de la confirmación, Amazon Data Firehose determina si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Si no quieres que Amazon Data Firehose vuelva a intentar enviar datos, establece este valor en 0.

Parámetros: opcional

Amazon Data Firehose incluye estos pares clave-valor en cada llamada HTTP. Estos parámetros pueden ayudarlo a identificar y organizar sus destinos.

Sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos al destino especificado. El tamaño de búfer recomendado para el destino de Elastic es de 1 MiB.

Configurar la copia de seguridad y los ajustes avanzados

En este tema se describe cómo configurar la copia de seguridad y los ajustes avanzados de la transmisión de Firehose.

Configura los ajustes de respaldo

Amazon Data Firehose utiliza Amazon S3 para hacer copias de seguridad de todos los datos o solo los datos fallidos que intenta entregar al destino elegido.

> **Important**
>
> - La configuración de Backup solo se admite si la fuente de la transmisión de Firehose es Direct PUT o Kinesis Data Streams.
> - La función de almacenamiento en búfer cero solo está disponible para los destinos de la aplicación y no está disponible para el destino de respaldo de Amazon S3.

Puedes especificar la configuración de respaldo de S3 para tu transmisión de Firehose si has elegido una de las siguientes opciones:

- Si establece Amazon S3 como destino de la transmisión de Firehose y elige especificar una función de AWS Lambda para transformar los registros de datos o si decide convertir los formatos de registro de datos para la transmisión de Firehose.
- Si establece Amazon Redshift como destino de la transmisión de Firehose y decide especificar una función AWS Lambda para transformar los registros de datos.
- Si configuras alguno de los siguientes servicios como destino de tu transmisión de Firehose: Amazon OpenSearch Service, Datadog, Dynatrace, HTTP Endpoint, LogicMonitor MongoDB Cloud, New Relic, Splunk o Sumo Logic.

Los siguientes son los ajustes de respaldo de tu transmisión de Firehose.

- Copia de seguridad de registros de origen en Amazon S3: si el destino seleccionado es S3 o Amazon Redshift, esta configuración indica si desea habilitar la copia de seguridad de los datos de origen o mantenerla deshabilitada. Si hay algún otro servicio admitido (que no sea S3 o Amazon Redshift) como destino seleccionado, esta configuración indica si desea hacer una copia de seguridad de todos los datos de origen o solo de los datos fallidos.

- Depósito de copia de seguridad de S3: este es el depósito de S3 en el que Amazon Data Firehose hace copias de seguridad de sus datos.
- Prefijo del depósito de copia de seguridad S3: es el prefijo con el que Amazon Data Firehose hace copias de seguridad de sus datos.
- Prefijo de salida de errores del bucket de copias de seguridad de S3: se hace una copia de seguridad de todos los datos fallidos en este prefijo de salida de errores del bucket de S3.
- Consejos de almacenamiento en búfer, compresión y cifrado para la copia de seguridad: Amazon Data Firehose utiliza Amazon S3 para hacer copias de seguridad de todos los datos o solo los que han fallado y que intenta entregar al destino elegido. Amazon Data Firehose almacena en búfer los datos entrantes antes de entregarlos (realizando una copia de seguridad) en Amazon S3. Puede elegir un tamaño de búfer de 1 a 128 segundos MiBs y un intervalo de búfer de 60 a 900 segundos. La condición que primero se cumpla desencadenará la entrega de datos en Amazon S3. Si habilita la transformación de datos, el intervalo de búfer se aplica desde el momento en que Amazon Data Firehose recibe los datos transformados hasta la entrega de los datos a Amazon S3. Si la entrega de datos al destino se retrasa con respecto a la escritura de datos en la transmisión Firehose, Amazon Data Firehose aumenta el tamaño del búfer de forma dinámica para ponerse al día. Esta acción ayuda a garantizar que todos los datos se entregan en el destino.
- Compresión S3: elija compresión de datos con GZIP, Snappy, Zip o Snappy compatible con Hadoop, o sin compresión de datos. La compresión Snappy compatible con Snappy, Zip y Hadoop no está disponible para la transmisión de Firehose con Amazon Redshift como destino.
- Formato de extensión de archivo S3 (opcional): especifique un formato de extensión de archivo para los objetos entregados al bucket de destino de Amazon S3. Si habilita esta función, la extensión de archivo especificada anulará las extensiones de archivo predeterminadas incorporadas por las funciones de conversión de formato de datos o de compresión de S3, como .parquet o .gz. Asegúrese de haber configurado la extensión de archivo correcta cuando utilice esta función con la conversión de formato de datos o la compresión S3. La extensión del archivo debe empezar con un punto (.) y puede contener los caracteres permitidos: 0-9a-z! -_.*' (). La extensión del archivo no puede superar los 128 caracteres.
- Firehose admite el cifrado del lado del servidor de Amazon S3 con AWS Key Management Service (SSE-KMS) para cifrar los datos entregados en Amazon S3. Puede optar por utilizar el tipo de cifrado predeterminado especificado en el depósito S3 de destino o cifrar con una clave de la lista de claves de su propiedad. AWS KMS Si cifra los datos con AWS KMS claves, puede usar la clave AWS administrada predeterminada (aws/s3) o una clave administrada por el cliente. Para obtener más información, consulte Protección de datos mediante el cifrado del lado del servidor con claves administradas por AWS KMS (SSE-KMS).

Configuración de opciones avanzadas

La siguiente sección contiene detalles sobre la configuración avanzada de tu transmisión de Firehose.

- Cifrado del lado del servidor: Amazon Data Firehose admite el cifrado del lado del servidor de Amazon S3 con el servicio de administración de AWS claves (AWS KMS) para cifrar los datos entregados en Amazon S3. Para obtener más información, consulte Protección de datos mediante el cifrado del lado del servidor con claves administradas por KMS (SSE-KMS). AWS
- Registro de errores: Amazon Data Firehose registra los errores relacionados con el procesamiento y la entrega. Además, cuando la transformación de datos está habilitada, puede registrar las invocaciones de Lambda y enviar los errores de entrega de datos a Logs. CloudWatch Para obtener más información, consulte Supervisión de Amazon Data Firehose mediante CloudWatch registros.

  > **Important**
  >
  > Si bien es opcional, se recomienda encarecidamente habilitar el registro de errores de Amazon Data Firehose durante la creación de la transmisión de Firehose. Esta práctica garantiza que pueda acceder a los detalles de los errores en caso de que se produzcan fallas en el procesamiento o la entrega de los registros.

- Permisos: Amazon Data Firehose utiliza funciones de IAM para todos los permisos que necesita la transmisión de Firehose. Puede elegir crear un nuevo rol en el que los permisos necesarios se asignen automáticamente o elegir un rol existente creado para Amazon Data Firehose. La función se utiliza para conceder a Firehose acceso a varios servicios, como el bucket de S3, la clave de AWS KMS (si el cifrado de datos está activado) y la función Lambda (si la transformación de datos está habilitada). La consola podría crear un rol con marcadores de posición. Para obtener más información, consulte ¿Qué es IAM?.
- Etiquetas: puede añadir etiquetas para organizar sus AWS recursos, realizar un seguimiento de los costes y controlar el acceso.

  Si especifica etiquetas en la `CreateDeliveryStream` acción, Amazon Data Firehose realiza una autorización adicional sobre la `firehose:TagDeliveryStream` acción para comprobar si los usuarios tienen permisos para crear etiquetas. Si no concedes este permiso, las solicitudes para crear nuevas transmisiones de Firehose con etiquetas de recursos de IAM fallarán con un resultado `AccessDeniedException` como el siguiente.

```
AccessDeniedException
User: arn:aws:sts::x:assumed-role/x/x is not authorized to perform:
 firehose:TagDeliveryStream on resource: arn:aws:firehose:us-east-1:x:deliverystream/
x with an explicit deny in an identity-based policy.
```

El siguiente ejemplo muestra una política que permite a los usuarios crear una transmisión Firehose y aplicar etiquetas.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "firehose:CreateDeliveryStream",
            "Resource": "*",
            }
        },
        {
            "Effect": "Allow",
            "Action": "firehose:TagDeliveryStream",
            "Resource": "*",
            }
        }
    ]
}
```

Una vez que hayas elegido la configuración avanzada y de respaldo, revisa tus opciones y, a continuación, selecciona Create Firehose stream.

La nueva transmisión de Firehose tarda unos momentos en estar en estado de creación antes de que esté disponible. Una vez que tu transmisión de Firehose esté en estado activo, puedes empezar a enviarle datos desde tu productor.

Comprenda las sugerencias de almacenamiento en búfer

Amazon Data Firehose almacena en memoria los datos de streaming entrantes hasta un tamaño determinado (tamaño de búfer) y durante un período de tiempo determinado (intervalo de almacenamiento en búfer) antes de entregarlos a los destinos especificados. Debería utilizar sugerencias de almacenamiento en búfer cuando desee entregar archivos de tamaño óptimo a

Amazon S3 y obtener un mejor rendimiento de las aplicaciones de procesamiento de datos o para ajustar la tasa de entrega de Firehose para que coincida con la velocidad de destino.

Puedes configurar el tamaño del búfer y el intervalo del búfer al crear nuevos flujos de Firehose o actualizar el tamaño del búfer y el intervalo de almacenamiento en tus flujos Firehose existentes. El tamaño del búfer se mide en MB y el intervalo de almacenamiento en segundos. Sin embargo, si especifica un valor para uno de ellos, también deberá proporcionar un valor para el otro. La primera condición del búfer que se cumpla hace que Firehose entregue los datos. Si no configura los valores de almacenamiento en búfer, se utilizarán los valores predeterminados.

Puedes configurar las sugerencias de almacenamiento en búfer de Firehose a través de los AWS Management Console, AWS Command Line Interface o los SDK. AWS Para las transmisiones existentes, puedes reconfigurar las sugerencias de almacenamiento en búfer con un valor que se adapte a tus casos de uso mediante la opción Editar de la consola o mediante la API. UpdateDestination En el caso de las transmisiones nuevas, puedes configurar las sugerencias de almacenamiento en búfer como parte de la creación de nuevas transmisiones mediante la consola o la API. CreateDeliveryStream Para ajustar el tamaño del búfer, establece `SizeInMBs` y `IntervalInSeconds` en el `DestinationConfiguration` parámetro específico de destino de la API o. CreateDeliveryStreamUpdateDestination

> **Note**
>
> - Para cumplir con las latencias más bajas de los casos de uso en tiempo real, puedes usar la sugerencia de cero intervalos de almacenamiento en búfer. Al configurar el intervalo de almacenamiento en búfer como cero segundos, Firehose no almacenará los datos en búfer y los entregará en unos segundos. Antes de cambiar las sugerencias de almacenamiento en búfer por un valor inferior, consulta con el proveedor las sugerencias de almacenamiento en búfer recomendadas para Firehose para sus destinos.
> - La función de almacenamiento en búfer cero solo está disponible para los destinos de la aplicación y no está disponible para el destino de respaldo de Amazon S3.

> **Note**
>
> Firehose utiliza la carga en varias partes para el destino S3 cuando configura un intervalo de tiempo de búfer inferior a 60 segundos para ofrecer latencias más bajas. Debido a que la carga se realiza en varias partes para el destino S3, los costes de la PUT API de S3

aumentarán en cierta medida si eliges un intervalo de tiempo de almacenamiento inferior a 60 segundos.

Para ver los rangos de sugerencias de almacenamiento en búfer específicos del destino y los valores predeterminados, consulta la siguiente tabla:

Destino	Tamaño del búfer en MB (predeterminado entre paréntesis)	Intervalo de almacenamiento en segundos (predeterminado entre paréntesis)
S3	1-128 (5)	0-900 (300)
Redshift	1-128 (5)	0-900 (300)
OpenSearch Sin servidor	1-100 (5)	0-900 (300)
OpenSearch	1-100 (5)	0-900 (300)
Splunk	1-5 (5)	0-60 (60)
Datadog	1-4 (4)	0-900 (60)
Coralogix	1-64 (6)	0-900 (60)
Dynatrace	1-64 (5)	0-900 (60)
Elastic	1	0-900 (60)
Honeycomb	1-64 (15)	0-900 (60)
Punto final HTTP	1-64 (5)	0-900 (60)
LogicMonitor	1-64 (5)	0-900 (60)
Logzio	1-64 (5)	0-900 (60)

Destino	Tamaño del búfer en MB (predeterminado entre paréntesis)	Intervalo de almacenamiento en segundos (predeterminado entre paréntesis)
MongoDB	1-16 (5)	0-900 (60)
Nueva reliquia	1-64 (5)	0-900 (60)
SumaLogic	1-64 (1)	0-900 (60)
Splunk Observability Cloud	1-64 (1)	0-900 (60)

Pruebe Firehose Stream con datos de muestra

Puede usarlo AWS Management Console para ingerir datos simulados de cotizaciones bursátiles. La consola ejecuta un script en tu navegador para incluir registros de muestra en tu transmisión de Firehose. Esto le permite probar la configuración de su transmisión Firehose sin tener que generar sus propios datos de prueba.

A continuación, mostramos un ejemplo de datos simulados:

```
{"TICKER_SYMBOL":"QXZ","SECTOR":"HEALTHCARE","CHANGE":-0.05,"PRICE":84.51}
```

Tenga en cuenta que se aplican cargos estándar de Amazon Data Firehose cuando su transmisión Firehose transmite los datos, pero no hay ningún cargo cuando se generan los datos. Para detener este cargo, puede detener el flujo de muestra desde la consola en cualquier momento.

Contenido

- Requisitos previos
- Pruebas con Amazon S3 como destino
- Pruebas con Amazon Redshift como destino
- Pruebe utilizando el OpenSearch servicio como destino
- Prueba con Splunk como destino

Requisitos previos

Antes de empezar, crea una transmisión de Firehose. Para obtener más información, consulte Crea una transmisión de Firehose.

Pruebas con Amazon S3 como destino

Utilice el siguiente procedimiento para probar la transmisión de Firehose utilizando Amazon Simple Storage Service (Amazon S3) como destino.

Para probar una transmisión de Firehose con Amazon S3

1. Abre la consola Firehose en https://console.aws.amazon.com/firehose/.

2. Elige una transmisión de Firehose activa. La transmisión de Firehose debe estar en estado Activo antes de que puedas empezar a enviar datos.
3. En Test with demo data, seleccione Start sending demo data para generar datos de tableros de cotizaciones simulados.
4. Siga las instrucciones que aparecen en pantalla para verificar que los datos se están entregando al bucket de S3. Tenga en cuenta que posiblemente pasen unos minutos hasta que los objetos aparezcan en el bucket. Esto dependerá de la configuración de búfer del bucket.
5. Una vez terminada la prueba, seleccione Stop sending demo data para detener los cargos por uso.

Pruebas con Amazon Redshift como destino

Utilice el siguiente procedimiento para probar la transmisión de Firehose con Amazon Redshift como destino.

Para probar una transmisión de Firehose con Amazon Redshift

1. Su transmisión de Firehose espera que haya una tabla en su clúster de Amazon Redshift. Conéctese a Amazon Redshift a través de una interfaz de SQL y ejecute la siguiente declaración para crear una tabla que acepte los datos de muestra.

```
create table firehose_test_table
(
 TICKER_SYMBOL varchar(4),
 SECTOR varchar(16),
 CHANGE float,
 PRICE float
);
```

2. Abre la consola Firehose en https://console.aws.amazon.com/firehose/.
3. Elige una transmisión de Firehose activa. La transmisión de Firehose debe estar en estado Activo antes de que puedas empezar a enviar datos.
4. Edita los detalles de destino de tu transmisión de Firehose para que apunten a la tabla recién creada`firehose_test_table`.
5. En Test with demo data, seleccione Start sending demo data para generar datos de tableros de cotizaciones simulados.

6. Siga las instrucciones que aparecen en pantalla para verificar que los datos se están entregando a la tabla. Tenga en cuenta que posiblemente pasen unos minutos hasta que los objetos aparezcan en la tabla. Esto dependerá de la configuración de búfer.
7. Una vez terminada la prueba, seleccione Stop sending demo data para detener los cargos por uso.
8. Edita los detalles de destino de tu transmisión de Firehose para que apunten a otra tabla.
9. Elimine la `firehose_test_table` tabla (opcional).

Pruebe utilizando el OpenSearch servicio como destino

Usa el siguiente procedimiento para probar tu transmisión de Firehose utilizando Amazon OpenSearch Service como destino.

Para probar una transmisión de Firehose mediante el Servicio OpenSearch

1. Abre la consola Firehose en https://console.aws.amazon.com/firehose/.
2. Elige una transmisión de Firehose activa. La transmisión de Firehose debe estar en estado Activo antes de que puedas empezar a enviar datos.
3. En Test with demo data, seleccione Start sending demo data para generar datos de tableros de cotizaciones simulados.
4. Sigue las instrucciones que aparecen en pantalla para comprobar que los datos se están enviando a tu dominio de OpenSearch servicio. Para obtener más información, consulta Búsqueda de documentos en un dominio OpenSearch de servicio en la Guía para desarrolladores de Amazon OpenSearch Service.
5. Una vez terminada la prueba, seleccione Stop sending demo data para detener los cargos por uso.

Prueba con Splunk como destino

Usa el siguiente procedimiento para probar tu transmisión de Firehose con Splunk como destino.

Para probar una transmisión de Firehose con Splunk

1. Abre la consola Firehose en https://console.aws.amazon.com/firehose/.

2. Elige una transmisión de Firehose activa. La transmisión de Firehose debe estar en estado Activo antes de que puedas empezar a enviar datos.
3. En Test with demo data, seleccione Start sending demo data para generar datos de tableros de cotizaciones simulados.
4. Compruebe si los datos se están entregando al índice de Splunk. Ejemplos de términos de búsqueda de Splunk: `sourcetype="aws:firehose:json"` y `index="name-of-your-splunk-index"`. Para obtener más información acerca de cómo buscar eventos en Splunk, consulte Search Manual en la documentación de Splunk.

   Si los datos de prueba no aparecen en el índice de Splunk, compruebe si hay eventos de error en el bucket de Amazon S3. Consulte también Datos no entregados a Splunk.
5. Una vez terminada la prueba, seleccione Stop sending demo data para evitar incurrir en cargos por uso.

Enviar datos a una transmisión de Firehose

Puede enviar datos a su transmisión de Firehose desde fuentes como la transmisión de datos de Kinesis, Amazon MSK, el agente de Kinesis o la API de Amazon Data Firehose mediante el SDK. AWS También puedes usar Amazon CloudWatch Logs, CloudWatch Events o AWS IoT como fuente de datos. Si es la primera vez que utiliza Amazon Data Firehose, tómese un tiempo para familiarizarse con los conceptos y la terminología que se presentan en. ¿Qué es Amazon Data Firehose?

> **Note**
>
> Algunos AWS servicios solo pueden enviar mensajes y eventos a una transmisión de Firehose que se encuentre en la misma región. Si tu transmisión de Firehose no aparece como opción al configurar un destino para Amazon CloudWatch Logs, CloudWatch Events o AWS IoT, verifica que tu transmisión de Firehose esté en la misma región que tus otros servicios.

Temas

- Escribir en Amazon Data Firehose mediante Kinesis Data Streams
- Escribir en Amazon Data Firehose mediante Amazon MSK
- Escribir en Amazon Data Firehose mediante Kinesis Agent
- Escribir en Amazon Data Firehose con el SDK AWS
- Escribir en Amazon Data Firehose mediante registros CloudWatch
- Escribir en Amazon Data Firehose mediante eventos CloudWatch
- Escribir en Amazon Data Firehose mediante AWS IoT

Escribir en Amazon Data Firehose mediante Kinesis Data Streams

Puede configurar Amazon Kinesis Data Streams para enviar información a una transmisión de Firehose.

> **Important**
>
> Si utiliza Kinesis Producer Library (KPL) para escribir datos en un flujo de datos de Kinesis, puede utilizar la agregación para combinar los registros que escriba en ese flujo de datos de Kinesis. Si, a continuación, utiliza esa transmisión de datos como fuente para la transmisión de Firehose, Amazon Data Firehose desagrega los registros antes de entregarlos al destino. Si configura la transmisión de Firehose para transformar los datos, Amazon Data Firehose desagrega los registros antes de enviarlos a. AWS Lambda Para obtener más información, consulte Desarrollar productores en Amazon Kinesis Data Streams con la Kinesis Producer Library y Agregación.

1. Inicie sesión en la consola Amazon Data Firehose AWS Management Console y ábrala en https://console.aws.amazon.com/firehose/.
2. Selecciona Crear transmisión de Firehose. En la página Name and source (Nombre y origen), proporcione valores para los siguientes campos:

   Nombre de la transmisión Firehose

   El nombre de tu transmisión de Firehose.

   Origen

   Elija la transmisión de Kinesis para configurar una transmisión de Firehose que utilice una transmisión de datos de Kinesis como fuente de datos. A continuación, puede utilizar Amazon Data Firehose para leer fácilmente los datos de un flujo de datos existente y cargarlos en los destinos.

   Para utilizar un flujo de datos de Kinesis como origen, elija un flujo de la lista Flujo de Kinesis o elija Crear nuevo para crear un nuevo flujo de datos de Kinesis. Después de crear un nuevo flujo, elija Actualizar para actualizar la lista Flujo de Kinesis. Si tiene un gran número de flujos, filtre la lista con Filter by name.

   > **Note**
   >
   > Al configurar una transmisión de datos de Kinesis como origen de una transmisión de Firehose, Amazon Data `PutRecord` Firehose y sus operaciones se deshabilitan.

`PutRecordBatch` En este caso, para añadir datos a la transmisión de Firehose, utilice las operaciones y los flujos de datos de Kinesis. `PutRecord` `PutRecords`

Amazon Data Firehose comienza a leer los datos desde la `LATEST` posición de la transmisión de Kinesis. Para obtener más información sobre las posiciones de Kinesis Data Streams, GetShardIteratorconsulte.

Amazon Data Firehose llama a la GetRecordsoperación de Kinesis Data Streams una vez por segundo para cada fragmento. Sin embargo, cuando la copia de seguridad completa está habilitada, Firehose llama a la `GetRecords` operación de Kinesis Data Streams dos veces por segundo para cada fragmento, una para el destino de entrega principal y otra para la copia de seguridad completa.

Se puede leer más de una transmisión de Firehose de la misma transmisión de Kinesis. Otras aplicaciones de Kinesis (consumidores) pueden leer también datos del mismo flujo. Cada llamada desde cualquier transmisión de Firehose u otra aplicación de consumo se tiene en cuenta para el límite de aceleración general del fragmento. Para evitar las limitaciones, planee sus aplicaciones con especial cuidado. Para obtener más información sobre los límites de Kinesis Data Streams, consulte Amazon Kinesis Streams Limits.

3. Seleccione Next para pasar a la página Configurar la transformación de registros y la conversión de formato.

Escribir en Amazon Data Firehose mediante Amazon MSK

Puede configurar Amazon MSK para que envíe información a una transmisión de Firehose.

1. Inicie sesión en la consola Amazon Data Firehose AWS Management Console y ábrala en https://console.aws.amazon.com/firehose/.
2. Selecciona Crear transmisión de Firehose.

   En la sección Elija un origen y un destino de la página, proporcione valores para los siguientes campos:

   Origen

   Elija Amazon MSK para configurar una transmisión Firehose que utilice Amazon MSK como fuente de datos. Puede elegir entre clústeres aprovisionados por MSK y clústeres sin servidor

de MSK. A continuación, puede utilizar Amazon Data Firehose para leer fácilmente los datos de un tema y un clúster de Amazon MSK específicos y cargarlos en el destino S3 especificado.

Destino

Elija Amazon S3 como destino de la transmisión de Firehose.

En la sección Configuración de origen de la página, proporcione valores para los siguientes campos:

Conectividad de clústeres de Amazon MSK

Elija la opción Agentes de arranque privados (recomendada) o Agentes de arranque públicos en función de la configuración del clúster. Los agentes de arranque es lo que el cliente de Apache Kafka utiliza como punto de partida para conectarse al clúster. Los corredores de bootstrap públicos están diseñados para el acceso público desde fuera AWS, mientras que los corredores de bootstrap privados están destinados al acceso desde dentro. AWS Para obtener más información sobre Amazon MSK, consulte Amazon Managed Streaming para Apache Kafka.

Para conectarse a un clúster de Amazon MSK aprovisionado o sin servidor a través de agentes de arranque privados, el clúster debe cumplir todos los requisitos siguientes.

- El clúster debe estar activo.
- El clúster debe tener IAM como uno de sus métodos de control de acceso.
- La conectividad privada de múltiples VPC debe estar habilitada para el método de control de acceso de IAM.
- Debe añadir a este clúster una política basada en recursos que conceda al director del servicio Amazon Data Firehose el permiso para invocar la API de Amazon MSK. CreateVpcConnection

Para conectarse a un clúster de Amazon MSK aprovisionado a través de agentes de arranque públicos, el clúster debe cumplir todos los requisitos siguientes.

- El clúster debe estar activo.
- El clúster debe tener IAM como uno de sus métodos de control de acceso.
- El clúster debe ser de acceso público.

Clúster de Amazon RDS

Para el mismo escenario de cuenta, especifique el ARN del clúster de Amazon MSK desde donde leerá los datos la transmisión de Firehose.

En el caso de un escenario entre cuentas, consulte Entrega entre cuentas desde Amazon MSK.

Tema

Especifique el tema de Apache Kafka del que quiere que su transmisión de Firehose ingiera datos. Una vez creada la transmisión de Firehose, no podrás actualizar este tema.

En la sección de nombres de transmisión de Firehose de la página, proporciona valores para los siguientes campos:

Nombre de la transmisión Firehose

Especifica el nombre de tu transmisión de Firehose.

3. Luego, puede completar el paso opcional de configurar la transformación de registros y la conversión del formato de registros. Para obtener más información, consulte Configurar la transformación de registros y la conversión de formato.

Escribir en Amazon Data Firehose mediante Kinesis Agent

El agente Amazon Kinesis es una aplicación de software Java independiente que sirve como implementación de referencia para mostrar cómo puede recopilar y enviar datos a Firehose. El agente supervisa continuamente un conjunto de archivos y envía nuevos datos a la transmisión de Firehose. El agente muestra cómo gestionar la rotación de archivos, los puntos de control y los reintentos en caso de error. Muestra cómo puede entregar sus datos de forma fiable, puntual y sencilla. También muestra cómo se pueden emitir CloudWatch métricas para supervisar y solucionar mejor los problemas del proceso de streaming. Para obtener más información, amazon-kinesis-agentawslabs/.

De forma predeterminada, los registros de cada archivo se analizan en función del carácter de nueva línea (`'\n'`). Sin embargo, el agente también se puede configurar para analizar registros multilínea (consulte Ajustes de la configuración del agente).

Puede instalar el agente en entornos de servidor basados en Linux, como servidores web, de registro o de base de datos. Tras instalar el agente, configúrelo especificando los archivos que se van a supervisar y el flujo Firehose de los datos. Una vez configurado el agente, recopila datos de los archivos de forma duradera y los envía de forma fiable a la transmisión Firehose.

Temas

- Requisitos previos
- Credenciales
- Proveedores de credenciales personalizados
- Descargar e instalar el agente
- Configuración e inicio del agente
- Ajustes de la configuración del agente
- Monitoreo de varios directorios de archivos y escritura en varias secuencias
- Uso del agente para preprocesar los datos
- Comandos del agente de la CLI
- Preguntas frecuentes

Requisitos previos

- Su sistema operativo debe ser Amazon Linux o Red Hat Enterprise Linux, versión 7 o posterior.
- La versión 2.0.0 o posterior del agente se ejecuta con la versión 1.8 o posterior de JRE. La versión 1.1.x del agente se ejecuta con la versión 1.7 o posterior de JRE.
- Si utiliza Amazon EC2 para ejecutar el agente, lance la instancia de EC2.
- El rol o AWS las credenciales de IAM que especifique deben tener permiso para realizar la operación Amazon Data PutRecordBatchFirehose para que el agente envíe datos a su transmisión de Firehose. Si habilita la CloudWatch supervisión del agente, también necesitará permiso para realizar la CloudWatch PutMetricDataoperación. Para obtener más información Control del acceso con Amazon Data FirehoseSupervisión del estado del agente de Kinesis, consulte y Autenticación y control de acceso para Amazon CloudWatch.

Credenciales

Gestione sus AWS credenciales mediante uno de los siguientes métodos:

- Cree un proveedor de credenciales personalizado. Para obtener más detalles, consulte the section called “Proveedores de credenciales personalizados”.
- Especifique un rol de IAM al lanzar la instancia EC2.
- Especifique AWS las credenciales al configurar el agente (consulte las entradas correspondientes `awsAccessKeyId` y `awsSecretAccessKey` en la tabla de configuración que aparece a continuaciónthe section called “Ajustes de la configuración del agente”).
- `/etc/sysconfig/aws-kinesis-agent`Edítelo para especificar su AWS región y sus claves de AWS acceso.
- Si su instancia EC2 está en una AWS cuenta diferente, cree un rol de IAM para proporcionar acceso al servicio Amazon Data Firehose. Especifique esa función al configurar el agente (consulte assumeRoleExternalAssumeroLearn and Id). Utilice uno de los métodos anteriores para especificar las AWS credenciales de un usuario de la otra cuenta que tenga permiso para asumir este rol.

Proveedores de credenciales personalizados

Puede crear un proveedor de credenciales personalizado e indicar su nombre de clase y ruta de archivo jar al agente de Kinesis en las siguientes opciones de configuración: `userDefinedCredentialsProvider.classname` y `userDefinedCredentialsProvider.location`. Para obtener las descripciones de estas dos opciones de configuración, consulte the section called “Ajustes de la configuración del agente”.

Para crear un proveedor de credenciales personalizado, defina una clase que implemente la interfaz `AWS CredentialsProvider`, como la del ejemplo siguiente.

```
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.auth.AWSCredentialsProvider;
import com.amazonaws.auth.BasicAWSCredentials;

public class YourClassName implements AWSCredentialsProvider {
    public YourClassName() {
    }

    public AWSCredentials getCredentials() {
        return new BasicAWSCredentials("key1", "key2");
    }

    public void refresh() {
    }
```

```
}
```

Su clase debe tener un constructor sin argumentos.

AWS invoca el método de actualización periódicamente para obtener las credenciales actualizadas. Si desea que el proveedor de credenciales proporcione credenciales diferentes a lo largo de su vida útil, incluya el código para actualizar las credenciales en este método. También puede dejar este método vacío si desea un proveedor de credenciales que ofrezca credenciales estáticas (no cambiantes).

Descargar e instalar el agente

Primero, conéctese a la instancia. Para obtener más información, consulte Connect to Your Instance en la Guía del usuario de Amazon EC2. Si tiene problemas para conectarse, consulte Solución de problemas de conexión a su instancia en la Guía del usuario de Amazon EC2.

A continuación, instale el agente siguiendo uno de los siguientes métodos.

- Configuración del agente desde los repositorios de Amazon Linux

  Este método solo funciona para instancias de Amazon Linux. Utilice el siguiente comando:

  ```
  sudo yum install -y aws-kinesis-agent
  ```

  La versión 2.0.0 o posterior del agente se instala en equipos con el sistema operativo Amazon Linux 2 (AL2). Esta versión del agente requiere la versión 1.8 o posterior de Java. Si la versión de Java requerida aún no está presente, el proceso de instalación del agente la instala. Para obtener más información sobre Amazon Linux 2, consulte https://aws.amazon.com/amazon-linux-2/.
- Configuración del agente desde el repositorio de Amazon S3

  Este método funciona para Red Hat Enterprise Linux, así como para las instancias de Amazon Linux 2, ya que instala el agente desde el repositorio disponible públicamente. Utilice el siguiente comando para descargar e instalar la versión más reciente de la versión 2.x.x del agente:

  ```
  sudo yum install -y https://s3.amazonaws.com/streaming-data-agent/aws-kinesis-agent-latest.amzn2.noarch.rpm
  ```

Para instalar una versión concreta del agente, especifique el número de versión en el comando. Por ejemplo, el siguiente comando instala la versión 2.0.1 del agente.

```
sudo yum install -y https://streaming-data-agent.s3.amazonaws.com/aws-kinesis-agent-2.0.1-1.amzn1.noarch.rpm
```

Si tiene Java 1.7 y no quiere actualizar la versión, puede descargar la versión 1.x.x del agente, que es compatible con Java 1.7. Por ejemplo, para descargar la versión 1.1.6 del agente, puede utilizar el comando siguiente:

```
sudo yum install -y https://s3.amazonaws.com/streaming-data-agent/aws-kinesis-agent-1.1.6-1.amzn1.noarch.rpm
```

La versión 1.x.x más reciente del agente se puede descargar mediante el siguiente comando:

```
sudo yum install -y https://s3.amazonaws.com/streaming-data-agent/aws-kinesis-agent-latest.amzn1.noarch.rpm
```

- Para configurar el agente desde el repositorio GitHub

  1. En primer lugar, asegúrese de que tiene instalada la versión de Java requerida, en función de la versión del agente.
  2. Descargue el agente del repositorio de awslabs/ amazon-kinesis-agent GitHub .
  3. Instale el agente. Para ello, diríjase al directorio de descargas y ejecute el siguiente comando:

     ```
     sudo ./setup --install
     ```

- Configuración del agente en un contenedor de Docker

  El agente de Kinesis también puede ejecutarse en un contenedor además de a través de la base de contenedores amazonlinux. Utilice el siguiente Dockerfile y ejecute `docker build`.

```
FROM amazonlinux

RUN yum install -y aws-kinesis-agent which findutils
COPY agent.json /etc/aws-kinesis/agent.json

CMD ["start-aws-kinesis-agent"]
```

Configuración e inicio del agente

Configuración e inicio del agente

1. Abra y edite el archivo de configuración (como superusuario si utiliza permisos de acceso de archivo predeterminado): `/etc/aws-kinesis/agent.json`

   En este archivo de configuración, especifique los archivos (`"filePattern"`) desde los que el agente recopila los datos y el nombre de la secuencia Firehose (`"deliveryStream"`) a la que el agente envía los datos. El nombre de archivo es un patrón y el agente reconoce las rotaciones de archivos. No puede rotar más de un archivo ni crear más de uno nuevo por segundo. El agente usa la marca de tiempo de creación del archivo para determinar qué archivos rastrear y seguir en tu transmisión de Firehose. Crear nuevos archivos o rotar los archivos más de una vez por segundo impide al agente diferenciarlos correctamente.

   ```
   {
      "flows": [
           {
               "filePattern": "/tmp/app.log*",
               "deliveryStream": "yourdeliverystream"
           }
      ]
   }
   ```

   La AWS región predeterminada es`us-east-1`. Si utiliza una región diferente, añada el ajuste `firehose.endpoint` al archivo de configuración y especifique el punto de enlace de la región. Para obtener más información, consulte Ajustes de la configuración del agente.

2. Comience el agente de forma manual:

   ```
   sudo service aws-kinesis-agent start
   ```

3. Configure el agente para iniciarse al arrancar el sistema (opcional):

```
sudo chkconfig aws-kinesis-agent on
```

El agente ya está se ejecutando como un servicio de sistema en segundo plano. Supervisa continuamente los archivos especificados y envía datos a la secuencia Firehose especificada. La auditoría de actividad se registra en `/var/log/aws-kinesis-agent/aws-kinesis-agent.log`.

Ajustes de la configuración del agente

El agente admite dos opciones de configuración obligatorias, `filePattern` y `deliveryStream`, además de configuraciones opcionales para activar características adicionales. Las opciones de configuración obligatorias y opcionales se especifican en `/etc/aws-kinesis/agent.json`.

Cada vez que cambie el archivo de configuración, debe detener y comenzar el agente con los siguientes comandos:

```
sudo service aws-kinesis-agent stop
sudo service aws-kinesis-agent start
```

También puede hacerlo con el siguiente comando:

```
sudo service aws-kinesis-agent restart
```

Las opciones de configuración generales son las siguientes.

Opción de configuración	Descripción
`assumeRoleARN`	El Nombre de recurso de Amazon (ARN) de la función que debe asumir el usuario. Para obtener más información, consulte Delegar el acceso a todas AWS las cuentas mediante funciones de IAM en la Guía del usuario de IAM.
`assumeRoleExternalId`	Un identificador opcional que determina quién puede asumir el rol. Para obtener más información, consulte Cómo utilizar un ID externo en la Guía del usuario de IAM.

Opción de configuración	Descripción
`awsAccessKeyId`	AWS ID de clave de acceso que anula las credenciales predeterminadas. Este ajuste tiene prioridad sobre los demás proveedores de credenciales.
`awsSecretAccessKey`	AWS clave secreta que anula las credenciales predeterminadas. Este ajuste tiene prioridad sobre los demás proveedores de credenciales.
`cloudwatch.emitMetrics`	Permite que el agente emita métricas CloudWatch si se ha establecido (true). Predeterminado: true
`cloudwatch.endpoint`	El punto final regional de CloudWatch. Valor predeterminado: `monitoring.us-east-1.amazonaws.com`
`firehose.endpoint`	El punto final regional de Amazon Data Firehose. Valor predeterminado: `firehose.us-east-1.amazonaws.com`
`sts.endpoint`	El punto de enlace regional del servicio de token AWS de seguridad. Valor predeterminado: `https://sts.amazonaws.com`
`userDefinedCredentialsProvider.classname`	Si define un proveedor de credenciales personalizado, proporcione su nombre de clase completo mediante esta configuración. No incluya `.class` al final del nombre de la clase.
`userDefinedCredentialsProvider.location`	Si define un proveedor de credenciales personalizado, utilice esta configuración para especificar la ruta absoluta del archivo jar que contiene el proveedor de credenciales personalizado. El agente también busca el archivo jar en la siguiente ubicación: `/usr/share/aws-kinesis-agent/lib/` .

Las opciones de configuración de flujo son las siguientes.

Opción de configuración	Descripción
`aggregatedRecordSizeBytes`	Para hacer que el agente agregue registros y, a continuación, los coloque en la transmisión Firehose en una sola operación, especifique esta configuración. Configúrelo en el tamaño que desee que tenga el registro agregado antes de que el agente lo coloque en la transmisión Firehose. Valor predeterminado: 0 (sin agregación)
`dataProcessingOptions`	La lista de opciones de procesamiento que se aplica a cada registro analizado antes de enviarlo a la transmisión Firehose. Las opciones de procesamiento se realizan en el orden especificado. Para obtener más información, consulte Uso del agente para preprocesar los datos.
`deliveryStream`	[Obligatorio] El nombre del arroyo Firehose.
`filePattern`	[Obligatorio] Un glob para los archivos que deben ser monitorizados por el agente. Cualquier archivo que coincida con este patrón es seleccionado y monitorizado automáticamente por el agente. En todos los archivos que coincidan con este patrón, conceda permisos de lectura a `aws-kinesis-agent-user` . En el directorio que contiene los archivos, conceda permisos de lectura y ejecución a `aws-kinesis-agent-user` . **Important** El agente recoge cualquier archivo que coincida con este patrón. Para asegurarse de que el agente no recoge registros no deseados, seleccione este patrón con precaución.
`initialPosition`	La posición inicial desde la que el archivo comenzó a ser analizado. Los valores válidos son `START_OF_FILE` y `END_OF_FILE` . Valor predeterminado: `END_OF_FILE`

Opción de configuración	Descripción
`maxBufferAgeMillis`	El tiempo máximo, en milisegundos, durante el que el agente almacena los datos en búfer antes de enviarlos a la transmisión Firehose. Rango de valores: 1000 - 900 000 (de 1 segundo a 15 minutos) Valor predeterminado: 60 000 (1 minuto)
`maxBufferSizeBytes`	El tamaño máximo, en bytes, para el que el agente almacena los datos en búfer antes de enviarlos a la transmisión Firehose. Rango de valores: 1 - 4 194 304 (4 MB) Valor predeterminado: 4 194 304 (4 MB)
`maxBufferSizeRecords`	El número máximo de registros para los que el agente almacena datos en búfer antes de enviarlos a la transmisión Firehose. Rango de valores: 1 - 500 Predeterminado: 500
`minTimeBetweenFilePollsMillis`	El intervalo de tiempo, en milisegundos, en el que el agente sondea y analiza los archivos monitorizados para identificar datos nuevos. Intervalo de valores: 1 o más Predeterminado: 100
`multiLineStartPattern`	El patrón para identificar el comienzo de un registro. Un registro consta de una línea que coincide con el patrón y de líneas siguientes que no coinciden con el patrón. Los valores válidos son expresiones regulares . De forma predeterminada, cada línea en los archivos de registro se analiza como un registro.

Opción de configuración	Descripción
`skipHeaderLines`	La cantidad de líneas de los archivos monitorizados, a partir de la primera, que el agente debe omitir en el momento de analizarlos. Intervalo de valores: 0 o más Cantidad predeterminada: 0 (cero)
`truncatedRecordTerminator`	La cadena que el agente utiliza para truncar un registro analizado cuando el tamaño del registro supera el límite de tamaño de registro de Amazon Data Firehose. (1000 KB) Valor predeterminado: `'\n'` (línea nueva)

Monitoreo de varios directorios de archivos y escritura en varias secuencias

Puede configurar el agente para que monitorice varios directorios de archivos y envíe datos a varias secuencias especificando varias opciones de configuración de secuencia. En el siguiente ejemplo de configuración, el agente supervisa dos directorios de archivos y envía datos a una transmisión de datos de Kinesis y a una transmisión de Firehose, respectivamente. Puede especificar puntos de enlace diferentes para Kinesis Data Streams y Amazon Data Firehose, de modo que la transmisión de datos y la transmisión de Firehose no tengan que estar en la misma región.

```
{
    "cloudwatch.emitMetrics": true,
    "kinesis.endpoint": "https://your/kinesis/endpoint",
    "firehose.endpoint": "https://your/firehose/endpoint",
    "flows": [
        {
            "filePattern": "/tmp/app1.log*",
            "kinesisStream": "yourkinesisstream"
        },
        {
            "filePattern": "/tmp/app2.log*",
            "deliveryStream": "yourfirehosedeliverystream"
        }
    ]
```

```
}
```

Para obtener más información detallada sobre el uso del agente con Amazon Kinesis Data Streams, consulte Writing to Amazon Kinesis Data Streams with Kinesis Agent.

Uso del agente para preprocesar los datos

El agente puede preprocesar los registros analizados de los archivos monitorizados antes de enviarlos a la transmisión de Firehose. Para habilitar esta característica, añada la opción de configuración `dataProcessingOptions` al flujo de archivos. Puede añadir una o más opciones de procesamiento que se ejecutarán en el orden especificado.

El agente es compatible con las siguientes opciones de procesamiento. Dado que el agente es de código abierto, el usuario puede desarrollar y ampliar sus opciones de procesamiento. Puede descargar el agente desde Kinesis Agent.

Opciones de procesamiento

`SINGLELINE`

Elimina los caracteres de nueva línea y los espacios situados al principio y al final de las líneas para convertir un registro multilínea en un registro de una sola línea.

```
{
    "optionName": "SINGLELINE"
}
```

`CSVTOJSON`

Convierte un registro con un formato separado mediante delimitadores al formato JSON.

```
{
    "optionName": "CSVTOJSON",
    "customFieldNames": [ "field1", "field2", ... ],
    "delimiter": "yourdelimiter"
}
```

`customFieldNames`

[Obligatorio] Los nombres de campos utilizados como claves en cada par de valores de clave JSON. Por ejemplo, si especifica `["f1", "f2"]`, el registro "v1, v2" se convierte en `{"f1":"v1","f2":"v2"}`.

`delimiter`

La cadena utilizada como delimitador en el registro. El valor predeterminado es una coma (,).

`LOGTOJSON`

Convierte un registro con un formato de registro en un registro con formato JSON. Los formatos de registro admitidos son Apache Common Log, Apache Combined Log, Apache Error Log y RFC3164 Syslog.

```
{
    "optionName": "LOGTOJSON",
    "logFormat": "logformat",
    "matchPattern": "yourregexpattern",
    "customFieldNames": [ "field1", "field2", … ]
}
```

`logFormat`

[Obligatorio] El formato de entrada del registro. Los valores posibles son los siguientes:

- `COMMONAPACHELOG`: formato común de registro de Apache. Cada entrada de registro sigue el siguiente patrón de forma predeterminada: "`%{host} %{ident} %{authuser} [%{datetime}] \"%{request}\" %{response} %{bytes}`".
- `COMBINEDAPACHELOG`: formato combinado de registro de Apache. Cada entrada de registro sigue el siguiente patrón de forma predeterminada: "`%{host} %{ident} %{authuser} [%{datetime}] \"%{request}\" %{response} %{bytes} %{referrer} %{agent}`".
- `APACHEERRORLOG`: formato de registro de errores de Apache. Cada entrada de registro sigue el siguiente patrón de forma predeterminada: "`[%{timestamp}] [%{module}: %{severity}] [pid %{processid}:tid %{threadid}] [client: %{client}] %{message}`".
- `SYSLOG`: formato RFC3164 de Syslog. Cada entrada de registro sigue el siguiente patrón de forma predeterminada: "`%{timestamp} %{hostname} %{program}[%{processid}]: %{message}`".

`matchPattern`

Sobrescribe el patrón predeterminado del formato de log especificado. Utilice esta configuración para extraer valores de entradas de log si utilizan un formato personalizado. Si especifica `matchPattern`, también debe especificar `customFieldNames`.

`customFieldNames`

Los nombres de campos utilizados como claves en cada par de valores de clave JSON. Utilice esta opción para definir nombres de campos para valores extraídos de `matchPattern`, o sobrescriba los nombres de campos de los formatos de logs predefinidos.

Example : Configuración LOGTOJSON

Este es un ejemplo de configuración `LOGTOJSON` de una entrada de registro en Formato común de registro de Apache convertida a formato JSON:

```
{
    "optionName": "LOGTOJSON",
    "logFormat": "COMMONAPACHELOG"
}
```

Antes de la conversión:

```
64.242.88.10 - - [07/Mar/2004:16:10:02 -0800] "GET /mailman/listinfo/hsdivision
 HTTP/1.1" 200 6291
```

Después de la conversión:

```
{"host":"64.242.88.10","ident":null,"authuser":null,"datetime":"07/
Mar/2004:16:10:02 -0800","request":"GET /mailman/listinfo/hsdivision
 HTTP/1.1","response":"200","bytes":"6291"}
```

Example : Configuración LOGTOJSON con campos personalizados

Este es otro ejemplo de configuración `LOGTOJSON`:

```
{
    "optionName": "LOGTOJSON",
    "logFormat": "COMMONAPACHELOG",
    "customFieldNames": ["f1", "f2", "f3", "f4", "f5", "f6", "f7"]
}
```

Con esta configuración, la misma entrada de registro con Formato común de registro de Apache del ejemplo anterior se convierte a formato JSON de la siguiente manera:

```
{"f1":"64.242.88.10","f2":null,"f3":null,"f4":"07/Mar/2004:16:10:02 -0800","f5":"GET /
mailman/listinfo/hsdivision HTTP/1.1","f6":"200","f7":"6291"}
```

Example : Convertir una entrada de registro con Formato común de registro de Apache

La siguiente configuración de secuencia convierte la entrada de registro común de Apache en un registro de una línea con formato JSON:

```
{
    "flows": [
        {
            "filePattern": "/tmp/app.log*",
            "deliveryStream": "my-delivery-stream",
            "dataProcessingOptions": [
                {
                    "optionName": "LOGTOJSON",
                    "logFormat": "COMMONAPACHELOG"
                }
            ]
        }
    ]
}
```

Example : Convertir registros multilínea

La siguiente configuración de flujo analiza aquellos registros multilínea cuya primera línea comience por "`[SEQUENCE=`". Primero, cada registro se convierte en un registro de una línea. Después, se extraen los valores del registro basándose en tabulaciones delimitadoras. Finalmente, los valores extraídos se asignan a valores `customFieldNames` específicos para formar un registro de una línea en formato JSON.

```
{
    "flows": [
        {
            "filePattern": "/tmp/app.log*",
            "deliveryStream": "my-delivery-stream",
            "multiLineStartPattern": "\\[SEQUENCE=",
            "dataProcessingOptions": [
                {
                    "optionName": "SINGLELINE"
                },
```

```
                {
                    "optionName": "CSVTOJSON",
                    "customFieldNames": [ "field1", "field2", "field3" ],
                    "delimiter": "\\t"
                }
            ]
        }
    ]
}
```

Example : Configuración LOGTOJSON con patrón de coincidencia

Este es un ejemplo de una configuración de entrada de registro con Formato común de registro de Apache LOGTOJSON convertida a formato JSON con el último campo (bytes) omitido:

```
{
    "optionName": "LOGTOJSON",
    "logFormat": "COMMONAPACHELOG",
    "matchPattern": "^([\\d.]+) (\\S+) (\\S+) \\[([\\w:/]+\\s[+\\-]\\d{4})\\] \"(.
+?)\" (\\d{3})",
    "customFieldNames": ["host", "ident", "authuser", "datetime", "request",
 "response"]
}
```

Antes de la conversión:

```
123.45.67.89 - - [27/Oct/2000:09:27:09 -0400] "GET /java/javaResources.html HTTP/1.0"
 200
```

Después de la conversión:

```
{"host":"123.45.67.89","ident":null,"authuser":null,"datetime":"27/Oct/2000:09:27:09
 -0400","request":"GET /java/javaResources.html HTTP/1.0","response":"200"}
```

Comandos del agente de la CLI

Iniciar automáticamente al agente al arrancar el sistema:

```
sudo chkconfig aws-kinesis-agent on
```

Compruebe el estado del agente:

```
sudo service aws-kinesis-agent status
```

Detener el agente:

```
sudo service aws-kinesis-agent stop
```

Leer el archivo de registro del agente desde esta ubicación:

```
/var/log/aws-kinesis-agent/aws-kinesis-agent.log
```

Desinstalar el agente:

```
sudo yum remove aws-kinesis-agent
```

Preguntas frecuentes

¿Hay un agente de Kinesis para Windows?

El agente de Kinesis para Windows es un software diferente del agente de Kinesis para plataformas Linux.

¿Por qué se ralentiza el agente de Kinesis o aumenta el valor de **`RecordSendErrors`**?

Normalmente esto se debe a la limitación de Kinesis. Compruebe la `WriteProvisionedThroughputExceeded` métrica de las transmisiones de Kinesis Data Streams o `ThrottledRecords` la métrica de las transmisiones de Firehose. Cualquier aumento desde 0 en estas métricas indica que es necesario aumentar los límites de flujos. Para obtener más información, consulte Límites de transmisión de datos de Kinesis y transmisiones de Firehose.

Una vez que descarte la limitación, compruebe si el agente de Kinesis está configurado para seguir una gran cantidad de archivos pequeños. Se produce un retraso cuando el agente de Kinesis sigue un archivo nuevo, por lo que el agente de Kinesis debería seguir una pequeña cantidad de archivos de mayor tamaño. Intente consolidar los archivos de registro en archivos más grandes.

¿Por qué se producen excepciones `java.lang.OutOfMemoryError` ?

El agente de Kinesis no tiene memoria suficiente para gestionar la carga de trabajo actual. Intente aumentar `JAVA_START_HEAP` y `JAVA_MAX_HEAP` en `/usr/bin/start-aws-kinesis-agent` y reinicie el agente.

¿Por qué se producen excepciones `IllegalStateException : connection pool shut down`?

El agente de Kinesis no tiene suficientes conexiones para gestionar la carga de trabajo actual. Intente aumentar `maxConnections` y `maxSendingThreads` en los ajustes generales de configuración del agente en `/etc/aws-kinesis/agent.json`. El valor predeterminado para estos campos es 12 veces los procesadores de tiempo de ejecución disponibles. Consulte AgentConfiguration.java para obtener más información sobre las configuraciones avanzadas de los agentes.

¿Cómo puedo depurar otro problema con el agente de Kinesis?

Los registros de nivel `DEBUG` pueden habilitarse en `/etc/aws-kinesis/log4j.xml`.

¿Cómo debo configurar el agente de Kinesis?

Cuanto menor sea el valor de `maxBufferSizeBytes`, más frecuentemente enviará datos el agente de Kinesis. Esto puede ser bueno ya que disminuye el tiempo de entrega de los registros, pero también aumenta las solicitudes por segundo a Kinesis.

¿Por qué el agente de Kinesis envía registros duplicados?

Esto ocurre debido a una mala configuración en el seguimiento de archivos. Asegúrese de que cada `fileFlow's filePattern` solo coincida con un archivo. Esto también puede ocurrir si el modo `logrotate` que se está utilizando está en modo `copytruncate`. Intente cambiar al modo predeterminado o al de creación para evitar la duplicación. Para obtener más información sobre la gestión de registros duplicados, consulte Handling Duplicate Records.

Escribir en Amazon Data Firehose con el SDK AWS

Puede usar la API Amazon Data Firehose para enviar datos a una transmisión de Firehose mediante el AWS SDK para Java, .NET, Node.js,Python o Ruby. Si es la primera vez que utiliza Amazon Data

Firehose, tómese un tiempo para familiarizarse con los conceptos y la terminología que se presentan en. ¿Qué es Amazon Data Firehose? Para obtener más información, consulte Comience a crear con Amazon Web Services.

Estos ejemplos no representan códigos listos para producción, ya que no comprueban todas las excepciones posibles ni toman en cuenta todas las consideraciones de seguridad y desempeño posibles.

La API Amazon Data Firehose ofrece dos operaciones para enviar datos a su transmisión Firehose: y. PutRecordPutRecordBatch `PutRecord()`envía un registro de datos en una llamada y `PutRecordBatch()` puede enviar varios registros de datos en una sola llamada.

Temas

- Operaciones de escritura única mediante PutRecord
- Operaciones de escritura por lotes mediante PutRecordBatch

Operaciones de escritura única mediante PutRecord

Para colocar datos solo se necesita el nombre del flujo Firehose y un búfer de bytes (<=1000 KB). Dado que Amazon Data Firehose agrupa varios registros por lotes antes de cargar el archivo en Amazon S3, es posible que desee añadir un separador de registros. Para colocar los datos un registro a la vez en una transmisión de Firehose, usa el siguiente código:

```
PutRecordRequest putRecordRequest = new PutRecordRequest();
putRecordRequest.setDeliveryStreamName(deliveryStreamName);

String data = line + "\n";

Record record = new Record().withData(ByteBuffer.wrap(data.getBytes()));
putRecordRequest.setRecord(record);

// Put record into the DeliveryStream
firehoseClient.putRecord(putRecordRequest);
```

Para obtener más información sobre el contexto del código, consulta el código de ejemplo incluido en el AWS SDK. Para obtener información sobre la sintaxis de solicitud y respuesta, consulta el tema correspondiente en Firehose API Operations.

Operaciones de escritura por lotes mediante PutRecordBatch

Para colocar datos solo se necesita el nombre de la secuencia Firehose y una lista de registros. Dado que Amazon Data Firehose agrupa varios registros por lotes antes de cargar el archivo en Amazon S3, es posible que desee añadir un separador de registros. Para colocar los registros de datos en lotes en una transmisión de Firehose, utilice el siguiente código:

```
PutRecordBatchRequest putRecordBatchRequest = new PutRecordBatchRequest();
putRecordBatchRequest.setDeliveryStreamName(deliveryStreamName);
putRecordBatchRequest.setRecords(recordList);

// Put Record Batch records. Max No.Of Records we can put in a
// single put record batch request is 500
firehoseClient.putRecordBatch(putRecordBatchRequest);

recordList.clear();
```

Para obtener más información sobre el contexto del código, consulta el código de ejemplo incluido en el AWS SDK. Para obtener información sobre la sintaxis de solicitud y respuesta, consulta el tema correspondiente en Firehose API Operations.

Escribir en Amazon Data Firehose mediante registros CloudWatch

CloudWatch Los eventos de registro se pueden enviar a Firehose mediante filtros de CloudWatch suscripción. Para obtener más información, consulte Filtros de suscripción con Amazon Data Firehose.

CloudWatch Los eventos de registro se envían a Firehose en formato gzip comprimido. Si quieres enviar eventos de registro descomprimidos a los destinos de Firehose, puedes usar la función de descompresión de Firehose para descomprimir los registros automáticamente. CloudWatch

> **Important**
>
> Actualmente, Firehose no admite la entrega de CloudWatch registros al destino de Amazon OpenSearch Service porque Amazon CloudWatch combina varios eventos de registro en un registro de Firehose y Amazon OpenSearch Service no puede aceptar varios eventos de registro en un registro. Como alternativa, puedes considerar usar un filtro de suscripción para Amazon OpenSearch Service in CloudWatch Logs.

Descompresión de registros CloudWatch

Si utilizas Firehose para entregar CloudWatch registros y quieres entregar datos descomprimidos a tu destino de transmisión Firehose, usa la conversión de formato de datos de Firehose (Parquet, ORC) o la partición dinámica. Debes habilitar la descompresión de la transmisión de Firehose.

Puedes activar la descompresión mediante los o los SDK AWS Management Console. AWS Command Line Interface AWS

> **Note**
>
> Si habilitas la función de descompresión en una transmisión, úsala exclusivamente para los filtros de suscripciones de CloudWatch Logs y no para los de Vended Logs. Si habilitas la función de descompresión en una transmisión que se utiliza para ingerir tanto CloudWatch registros como registros vendidos, se produce un error en la ingestión de registros vendidos a Firehose. Esta función de descompresión es exclusiva para los registros. CloudWatch

Extracción de mensajes tras la descompresión de los registros CloudWatch

Cuando habilita la descompresión, tiene la opción de habilitar también la extracción de mensajes. Al utilizar la extracción de mensajes, Firehose filtra todos los metadatos, como el propietario, el grupo de registro, el flujo de registro y otros, de los CloudWatch registros descomprimidos y entrega solo el contenido de los campos de mensajes. Si envía datos a un destino de Splunk, debe activar la extracción de mensajes para que Splunk analice los datos. Los siguientes son ejemplos de resultados después de la descompresión con y sin extracción de mensajes.

Figura 1: Ejemplo de salida después de la descompresión sin extracción del mensaje:

```
{
 "owner": "111111111111",
 "logGroup": "CloudTrail/logs",
 "logStream": "111111111111_CloudTrail/logs_us-east-1",
 "subscriptionFilters": [
 "Destination"
 ],
 "messageType": "DATA_MESSAGE",
 "logEvents": [
 {
 "id": "31953106606966983378809025079804211143289615424298221568",
```

```
 "timestamp": 1432826855000,
 "message": "{\"eventVersion\":\"1.03\",\"userIdentity\":{\"type\":\"Root1\"}"
 },
 {
 "id": "31953106606966983378809025079804211143289615424298221569",
 "timestamp": 1432826855000,
 "message": "{\"eventVersion\":\"1.03\",\"userIdentity\":{\"type\":\"Root2\"}"
 },
 {
 "id": "31953106606966983378809025079804211143289615424298221570",
 "timestamp": 1432826855000,
 "message": "{\"eventVersion\":\"1.03\",\"userIdentity\":{\"type\":\"Root3\"}"
 }
 ]
}
```

Figura 2: Ejemplo de salida después de la descompresión con extracción del mensaje:

```
{"eventVersion":"1.03","userIdentity":{"type":"Root1"}
{"eventVersion":"1.03","userIdentity":{"type":"Root2"}
{"eventVersion":"1.03","userIdentity":{"type":"Root3"}
```

Activación y desactivación de la descompresión

Puedes activar y desactivar la descompresión mediante los AWS Command Line Interface o AWS los AWS Management Console SDK.

Habilitar la descompresión en un flujo de datos nuevo mediante el AWS Management Console

Para habilitar la descompresión en un flujo de datos nuevo mediante el AWS Management Console

1. Inicie sesión en la consola Kinesis AWS Management Console y ábrala en https://console.aws.amazon.com/kinesis.
2. Elija Amazon Data Firehose en el panel de navegación.
3. Selecciona Crear transmisión de Firehose.
4. En Elegir origen y destino

   Origen

   La fuente de tu transmisión de Firehose. Elige una de las siguientes fuentes:

- Direct PUT: elija esta opción para crear una transmisión Firehose en la que las aplicaciones del productor escriban directamente. Para obtener una lista de AWS los servicios, agentes y servicios de código abierto que están integrados con Direct PUT en Firehose, consulte esta sección.
- Transmisión de Kinesis: elija esta opción para configurar una transmisión de Firehose que utilice una transmisión de datos de Kinesis como fuente de datos. A continuación, puede usar Firehose para leer fácilmente los datos de una transmisión de datos de Kinesis existente y cargarlos en los destinos. Para obtener más información, consulte Cómo escribir en Firehose mediante Kinesis Data Streams

Destino

El destino de tu transmisión Firehose. Seleccione una de las siguientes opciones:

- Amazon S3
- Splunk

5. En el nombre de la transmisión de Firehose, introduce un nombre para la transmisión.
6. (Opcional) En Transformar registros:
   - En la sección Descomprimir registros fuente de Amazon CloudWatch Logs, selecciona Activar la descompresión.
   - Si quieres usar la extracción de mensajes después de la descompresión, selecciona Activar la extracción de mensajes.

Habilitar la descompresión en un flujo de datos existente mediante el AWS Management Console

Si tiene una transmisión Firehose con una función Lambda para realizar la descompresión, puede sustituirla por la función de descompresión Firehose. Antes de continuar, revise el código de la función Lambda para confirmar que solo realiza la descompresión o la extracción de mensajes. La salida de la función Lambda debería tener un aspecto similar a los ejemplos que se muestran en la figura 1 o la figura 2 de la sección anterior. Si el resultado tiene un aspecto similar, puede reemplazar la función Lambda siguiendo estos pasos.

1. Sustituya la función Lambda actual por este modelo. La nueva función Lambda de blueprint detecta automáticamente si los datos entrantes están comprimidos o descomprimidos. Solo realiza la descompresión si los datos de entrada están comprimidos.

2. Activa la descompresión con la opción Firehose integrada para la descompresión.
3. Activa CloudWatch las métricas para tu transmisión de Firehose si aún no está habilitada. Supervisa la métrica CloudWatchProcessorLambda _ IncomingCompressedData y espera a que esta métrica cambie a cero. Esto confirma que todos los datos de entrada enviados a la función Lambda están descomprimidos y que la función Lambda ya no es necesaria.
4. Elimine la transformación de datos de Lambda porque ya no la necesita para descomprimir la transmisión.

Desactivar la descompresión mediante el AWS Management Console

Para deshabilitar la descompresión de un flujo de datos mediante el AWS Management Console

1. Inicie sesión en la consola Kinesis AWS Management Console y ábrala en https://console.aws.amazon.com/kinesis.
2. Elija Amazon Data Firehose en el panel de navegación.
3. Elige la transmisión de Firehose que deseas editar.
4. En la página de detalles de la transmisión de Firehose, selecciona la pestaña Configuración.
5. En la sección Transformar y convertir registros, selecciona Editar.
6. En Descomprimir registros fuente de Amazon CloudWatch Logs, desmarca Activar la descompresión y, a continuación, selecciona Guardar cambios.

Preguntas frecuentes

¿Qué ocurre con los datos de origen en caso de que se produzca un error durante la descompresión?

Si Amazon Data Firehose no puede descomprimir el registro, el registro se entrega tal cual (en formato comprimido) al bucket de error S3 que especificó durante la creación de la transmisión de Firehose. Junto con el registro, el objeto entregado también incluye el código y el mensaje de error, y estos objetos se entregarán en un prefijo de bucket de S3 denominado. `decompression-failed` Firehose seguirá procesando otros registros después de una descompresión fallida de un registro.

¿Qué ocurre con los datos de origen en caso de que se produzca un error en el proceso de procesamiento tras una descompresión satisfactoria?

Si Amazon Data Firehose produce un error en los pasos de procesamiento posteriores a la descompresión, como la partición dinámica y la conversión de formatos de datos, el registro se entrega en formato comprimido al depósito de S3 de error que especificó durante la creación de la transmisión de Firehose. Junto con el registro, el objeto entregado también incluye el código y el mensaje de error.

¿Cómo se le informa en caso de error o excepción?

En caso de que se produzca un error o una excepción durante la descompresión, si configura CloudWatch los registros, Firehose registrará los mensajes CloudWatch de error en los registros. Además, Firehose envía las métricas a las CloudWatch métricas que puedes monitorear. Si lo desea, también puede crear alarmas en función de las métricas emitidas por Firehose.

¿Qué ocurre cuando **`put`** las operaciones no provienen de CloudWatch los registros?

Cuando los clientes `puts` no provienen de CloudWatch Logs, aparece el siguiente mensaje de error:

```
Put to Firehose failed for AccountId: <accountID>, FirehoseName: <firehosename> because
 the request is not originating from allowed source types.
```

¿Qué métricas emite Firehose para la función de descompresión?

Firehose emite métricas para la descompresión de todos los registros. Debe seleccionar el período (1 minuto), la estadística (suma) y el intervalo de fechas para obtener el número de errores, éxitos`DecompressedRecords`, errores o `DecompressedBytes` éxitos. Para obtener más información, consulte CloudWatch Registra las métricas de descompresión.

Escribir en Amazon Data Firehose mediante eventos CloudWatch

Puedes configurar Amazon CloudWatch para que envíe eventos a una transmisión de Firehose añadiendo un objetivo a una regla de CloudWatch eventos.

Para crear un objetivo para una regla de CloudWatch eventos que envíe eventos a una transmisión de Firehose existente

1. Inicia sesión AWS Management Console y abre la CloudWatch consola en https://console.aws.amazon.com/cloudwatch/.

2. Elija Crear regla.
3. En la página Paso 1: Crear regla, en Objetivos, selecciona Añadir objetivo y, a continuación, selecciona Firehose stream.
4. Elige una transmisión de Firehose existente.

Para obtener más información sobre la creación de reglas de CloudWatch eventos, consulte Introducción a Amazon CloudWatch Events.

Escribir en Amazon Data Firehose mediante AWS IoT

Puedes configurar AWS IoT el envío de información a una transmisión de Firehose añadiendo una acción.

Para crear una acción que envíe eventos a una transmisión de Firehose existente

1. Al crear una regla en la consola de AWS IoT, en la página Create a rule (Crear una regla), en Set one or more actions (Establecer una o varias acciones), elija Add action (Añadir acción).
2. Elija Enviar mensajes a un flujo de Amazon Kinesis Firehose.
3. Elija Configurar acción.
4. En Nombre de transmisión, elige una transmisión Firehose existente.
5. En Separator, seleccione un carácter de separación a insertar entre registros.
6. En Nombre del rol de IAM, elija un rol de IAM existente o elija Crear un nuevo rol.
7. Seleccione Agregar acción.

Para obtener más información acerca de la creación de reglas de AWS IoT, consulte AWS IoT Rule Tutorials.

Seguridad en Amazon Data Firehose

La seguridad en la nube AWS es la máxima prioridad. Como AWS cliente, se beneficiará de una arquitectura de centro de datos y red diseñada para cumplir con los requisitos de las organizaciones más sensibles a la seguridad.

La seguridad es una responsabilidad compartida entre usted AWS y usted. El modelo de responsabilidad compartida la describe como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube: AWS es responsable de proteger la infraestructura que ejecuta AWS los servicios en la AWS nube. AWS también le proporciona servicios que puede utilizar de forma segura. Auditores externos prueban y verifican periódicamente la eficacia de nuestra seguridad en el marco de los programas de conformidad de AWS. Para obtener más información sobre los programas de cumplimiento que se aplican a Data Firehose, consulte AWS Servicios dentro del alcance por programa de cumplimiento.
- Seguridad en la nube: su responsabilidad viene determinada por el AWS servicio que utilice. Usted también es responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y los reglamentos aplicables.

Esta documentación le ayuda a entender cómo aplicar el modelo de responsabilidad compartida al utilizar Data Firehose. En los temas siguientes se muestra cómo configurar Data Firehose para cumplir sus objetivos de seguridad y conformidad. También aprenderás a usar otros AWS servicios que pueden ayudarte a monitorear y proteger tus recursos de Data Firehose.

Temas

- Protección de datos en Amazon Data Firehose
- Control del acceso con Amazon Data Firehose
- Autenticate con AWS Secrets Manager Amazon Data Firehose
- Gestione las funciones de IAM a través de la consola Amazon Data Firehose
- Supervisión de Amazon Data Firehose
- Validación de conformidad para Amazon Data Firehose
- Resiliencia en Amazon Data Firehose
- Seguridad de la infraestructura en Amazon Data Firehose
- Mejores prácticas de seguridad para Amazon Data Firehose

Protección de datos en Amazon Data Firehose

Amazon Data Firehose cifra todos los datos en tránsito mediante el protocolo TLS. Además, para los datos almacenados en un almacenamiento provisional durante el procesamiento, Amazon Data Firehose cifra los datos AWS Key Management Servicey verifica su integridad mediante la verificación por suma de control.

Si tiene datos confidenciales, puede activar el cifrado de datos del lado del servidor cuando utilice Amazon Data Firehose. La forma de hacerlo dependerá del origen de los datos.

> **Note**
>
> Si necesita módulos criptográficos validados por FIPS 140-2 para acceder a AWS través de una interfaz de línea de comandos o una API, utilice un punto de conexión FIPS. Para obtener más información sobre los puntos de conexión de FIPS disponibles, consulte Estándar de procesamiento de la información federal (FIPS) 140-2.

Cifrado del servidor con Kinesis Data Streams como origen de datos

Cuando envía datos de sus productores de datos a su transmisión de datos, Kinesis Data Streams cifra los datos mediante AWS Key Management Service una clave AWS KMS() antes de almacenarlos en reposo. Cuando la transmisión Firehose lee los datos de la transmisión de datos, Kinesis Data Streams primero descifra los datos y, a continuación, los envía a Amazon Data Firehose. Amazon Data Firehose almacena en búfer los datos de la memoria en función de las sugerencias de almacenamiento en búfer que especifique. A continuación, los envía a los destinos sin tener que almacenar los datos no cifrados en reposo.

Para obtener información sobre cómo habilitar el cifrado del servidor para Kinesis Data Streams, consulte Using Server-Side Encryption en la Guía para desarrolladores de Amazon Kinesis Data Streams.

Cifrado del lado del servidor con Direct PUT u otros orígenes de datos

Si envías datos a tu transmisión de Firehose mediante Amazon Logs PutRecordo Events PutRecordBatch, o si envías los datos mediante AWS IoT Amazon CloudWatch Logs o CloudWatch Events, puedes activar el cifrado del lado del servidor mediante esta operación. StartDeliveryStreamEncryption

Para detenerlo server-side-encryption, usa la operación. StopDeliveryStreamEncryption

También puedes habilitar SSE al crear la transmisión Firehose. Para ello, especifica DeliveryStreamEncryptionConfigurationInputcuándo se invoca CreateDeliveryStream.

Cuando el CMK es de tipo`CUSTOMER_MANAGED_CMK`, si el servicio Amazon Data Firehose no puede descifrar los registros debido a a, `KMSNotFoundException` a, `KMSInvalidStateException` a o `KMSDisabledException` `KMSAccessDeniedException` a, el servicio espera hasta 24 horas (el período de retención) para que resuelva el problema. Si el problema continúa después del periodo de retención, el servicio omite los registros que han superado el periodo de retención y no se pudieron descifrar y, a continuación, descarta los datos. Amazon Data Firehose proporciona las siguientes cuatro CloudWatch métricas que puede utilizar para realizar un seguimiento de las cuatro AWS KMS excepciones:

- `KMSKeyAccessDenied`
- `KMSKeyDisabled`
- `KMSKeyInvalidState`
- `KMSKeyNotFound`

Para obtener más información sobre estas métricas, consulte the section called “Monitorización con métricas CloudWatch ”.

> **Important**
>
> Para cifrar tu transmisión de Firehose, usa CMK simétricas. Amazon Data Firehose no admite CMK asimétricas. Para obtener información sobre las CMK simétricas y asimétricas, consulte Acerca de las CMK simétricas y asimétricas en la guía para desarrolladores. AWS Key Management Service

> **Note**
>
> Cuando utilizas una clave gestionada por el cliente (CUSTOMER_MANAGED_CMK) para habilitar el cifrado del lado del servidor (SSE) para tu transmisión de Firehose, el servicio Firehose establece un contexto de cifrado siempre que utilice tu clave. Como este contexto de cifrado representa un caso en el que se utilizó una clave propiedad de tu AWS cuenta, se registra como parte de los registros de eventos de tu cuenta. AWS CloudTrail AWS Este

contexto de cifrado es un sistema generado por el servicio Firehose. Su aplicación no debe hacer suposiciones sobre el formato o el contenido del contexto de cifrado establecido por el servicio Firehose.

Control del acceso con Amazon Data Firehose

En las siguientes secciones se explica cómo controlar el acceso a y desde los recursos de Amazon Data Firehose. La información que cubren incluye cómo conceder acceso a tu aplicación para que pueda enviar datos a tu transmisión de Firehose. También describen cómo puede conceder a Amazon Data Firehose acceso a su bucket de Amazon Simple Storage Service (Amazon S3), clúster de Amazon Redshift o clúster de Amazon OpenSearch Service, así como los permisos de acceso que necesita si utiliza Datadog, Dynatrace, LogicMonitor MongoDB, New Relic, Splunk o Sumo Logic como destino. Por último, en este tema encontrarás instrucciones sobre cómo configurar Amazon Data Firehose para que pueda entregar datos a un destino que pertenezca a una cuenta diferente AWS . La tecnología para gestionar todas estas formas de acceso es la AWS Identity and Access Management (IAM). Para obtener más información acerca de IAM, consulte ¿Qué es IAM?.

Contenido

- Conceda a su aplicación acceso a sus recursos de Amazon Data Firehose
- Conceda a Amazon Data Firehose acceso a su clúster privado de Amazon MSK
- Permita que Amazon Data Firehose asuma una función de IAM
- Conceda acceso a Amazon Data Firehose AWS Glue para la conversión de formatos de datos
- Conceda a Amazon Data Firehose acceso a un destino de Amazon S3
- Conceda a Amazon Data Firehose acceso a un destino de Amazon Redshift
- Conceda a Amazon Data Firehose acceso a un destino de servicio público OpenSearch
- Otorgue a Amazon Data Firehose acceso a un destino de OpenSearch servicio en una VPC
- Conceda a Amazon Data Firehose acceso a un destino público sin servidor OpenSearch
- Otorgue a Amazon Data Firehose acceso a un destino OpenSearch sin servidor en una VPC
- Conceda a Amazon Data Firehose acceso a un destino de Splunk
- Acceso a Splunk en VPC
- Acceso a Snowflake o al punto final HTTP
- Conceda a Amazon Data Firehose acceso a un destino con copos de nieve
- Acceso a Snowflake en VPC

- Conceda a Amazon Data Firehose acceso a un destino de punto final HTTP
- Entrega entre cuentas desde Amazon MSK
- Entrega entre cuentas en un destino de Amazon S3
- Entrega multicuenta a un destino de servicio OpenSearch
- Uso de etiquetas para controlar el acceso

Conceda a su aplicación acceso a sus recursos de Amazon Data Firehose

Para permitir que tu aplicación acceda a tu transmisión de Firehose, usa una política similar a la de este ejemplo. Puede ajustar las operaciones individuales de las API a las que concede acceso modificando la sección `Action` o concediendo acceso a todas las operaciones `"firehose:*"`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "firehose:DeleteDeliveryStream",
                "firehose:PutRecord",
                "firehose:PutRecordBatch",
                "firehose:UpdateDestination"
            ],
            "Resource": [
                "arn:aws:firehose:region:account-id:deliverystream/delivery-stream-
name"
            ]
        }
    ]
}
```

Conceda a Amazon Data Firehose acceso a su clúster privado de Amazon MSK

Si la fuente de tu transmisión de Firehose es un clúster privado de Amazon MSK, utiliza una política similar a la de este ejemplo.

```
{
  "Version": "2012-10-17",
```

```
  "Statement": [
    {
      "Principal": {
        "Service": [
          "firehose.amazonaws.com"
        ]
    },
    "Effect": "Allow",
    "Action": [
      "kafka:CreateVpcConnection"
    ],
    "Resource": "cluster-arn"
    }
  ]
}
```

Permita que Amazon Data Firehose asuma una función de IAM

En esta sección se describen los permisos y las políticas que otorgan a Amazon Data Firehose acceso para ingerir, procesar y entregar datos desde el origen hasta el destino.

> **Note**
>
> Si utilizas la consola para crear una transmisión de Firehose y eliges la opción de crear una nueva función, AWS adjunta la política de confianza necesaria a la función. Si desea que Amazon Data Firehose utilice una función de IAM existente o si crea una función por su cuenta, adjunte las siguientes políticas de confianza a esa función para que Amazon Data Firehose pueda asumirla. Edite las políticas para sustituir el ID de *`cuenta por el ID de su cuenta`*. AWS Para obtener información acerca de cómo modificar la relación de confianza de un rol, consulte Modificación de un rol.

Amazon Data Firehose utiliza una función de IAM para todos los permisos que la transmisión de Firehose necesita para procesar y entregar datos. Asegúrese de que las siguientes políticas de confianza estén asociadas a esa función para que Amazon Data Firehose pueda asumirla.

```
{
 "Version": "2012-10-17",
 "Statement": [{
  "Sid": "",
```

```
  "Effect": "Allow",
  "Principal": {
   "Service": "firehose.amazonaws.com"
  },
  "Action": "sts:AssumeRole",
  "Condition": {
   "StringEquals": {
    "sts:ExternalId": "account-id"
   }
  }
 }]
}
```

Esta política utiliza la clave de contexto de `sts:ExternalId` condición para garantizar que solo la actividad de Amazon Data Firehose que se origine en su AWS cuenta pueda asumir esta función de IAM. Para obtener más información sobre cómo evitar el uso no autorizado de los roles de IAM, consulte Problema del suplente confuso en la Guía del usuario de IAM.

Si elige Amazon MSK como fuente para su transmisión de Firehose, debe especificar otra función de IAM que conceda permisos a Amazon Data Firehose para ingerir datos de origen del clúster de Amazon MSK especificado. Asegúrese de que las siguientes políticas de confianza estén asociadas a esa función para que Amazon Data Firehose pueda asumirla.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Principal": {
        "Service": [
          "firehose.amazonaws.com"
        ]
      },
      "Effect": "Allow",
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Asegúrese de que este rol que concede a Amazon Data Firehose permisos para ingerir datos de origen del clúster de Amazon MSK especificado conceda los siguientes permisos:

```
{
   "Version": "2012-10-17",
   "Statement": [{
        "Effect":"Allow",
        "Action": [
           "kafka:GetBootstrapBrokers",
           "kafka:DescribeCluster",
           "kafka:DescribeClusterV2",
           "kafka-cluster:Connect"
         ],
         "Resource": "CLUSTER-ARN"
       },
       {
         "Effect":"Allow",
         "Action": [
           "kafka-cluster:DescribeTopic",
           "kafka-cluster:DescribeTopicDynamicConfiguration",
           "kafka-cluster:ReadData"
         ],
         "Resource": "TOPIC-ARN"
       }]
}
```

Conceda acceso a Amazon Data Firehose AWS Glue para la conversión de formatos de datos

Si su transmisión de Firehose realiza una conversión de formato de datos, Amazon Data Firehose hace referencia a las definiciones de tablas almacenadas en. AWS Glue Para conceder a Amazon Data Firehose el acceso necesario AWS Glue, añade la siguiente declaración a tu política. Para obtener información sobre cómo encontrar el ARN de la tabla, consulte Especificación de los ARN de los recursos de AWS Glue.

```
[{
    "Effect": "Allow",
    "Action": [
        "glue:GetTable",
        "glue:GetTableVersion",
        "glue:GetTableVersions"
    ],
```

```
    "Resource": "table-arn"
}, {
    "Sid": "GetSchemaVersion",
    "Effect": "Allow",
    "Action": [
        "glue:GetSchemaVersion"
    ],
    "Resource": ["*"]
}]
```

La política recomendada para obtener esquemas del registro de esquemas no tiene restricciones de recursos. Para obtener más información, consulte los ejemplos de deserializadores de IAM en la Guía para desarrolladores. AWS Glue

> **Note**
>
> Actualmente, no AWS Glue se admite en las regiones de Israel (Tel Aviv), Asia Pacífico (Yakarta) o Oriente Medio (Emiratos Árabes Unidos). Si trabaja con Amazon Data Firehose en la región de Asia Pacífico (Yakarta) o Oriente Medio (Emiratos Árabes Unidos), asegúrese de dar acceso a su Amazon Data Firehose AWS Glue en una de las regiones en las que se AWS Glue admite actualmente. Compatible con la interoperabilidad entre regiones entre Data Firehose. AWS Glue Para obtener más información sobre las regiones en las que AWS Glue se admite, consulte https://docs.aws.amazon.com/general/latest/gr/glue.html

Conceda a Amazon Data Firehose acceso a un destino de Amazon S3

Cuando utiliza un destino de Amazon S3, Amazon Data Firehose entrega los datos a su bucket de S3 y, si lo desea, puede utilizar una AWS KMS clave de su propiedad para el cifrado de datos. Si el registro de errores está activado, Amazon Data Firehose también envía los errores de entrega de datos al grupo de CloudWatch registros y a las transmisiones. Debes tener un rol de IAM al crear una transmisión de Firehose. Amazon Data Firehose asume esa función de IAM y obtiene acceso al bucket, la clave y el grupo de CloudWatch registros y las transmisiones especificados.

Utilice la siguiente política de acceso para permitir que Amazon Data Firehose acceda a su bucket y AWS KMS clave de S3. Si no es el propietario del bucket de S3, agregue `s3:PutObjectAcl` a la lista de acciones de Amazon S3. Esto otorga al propietario del bucket acceso total a los objetos entregados por Amazon Data Firehose.

```
{
    "Version": "2012-10-17",
    "Statement":
    [
        {
            "Effect": "Allow",
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name",
                "arn:aws:s3:::bucket-name/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStream",
                "kinesis:GetShardIterator",
                "kinesis:GetRecords",
                "kinesis:ListShards"
            ],
            "Resource": "arn:aws:kinesis:region:account-id:stream/stream-name"
        },
        {
           "Effect": "Allow",
           "Action": [
               "kms:Decrypt",
               "kms:GenerateDataKey"
           ],
           "Resource": [
               "arn:aws:kms:region:account-id:key/key-id"
           ],
           "Condition": {
               "StringEquals": {
                   "kms:ViaService": "s3.region.amazonaws.com"
               },
               "StringLike": {
```

```
                    "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::bucket-name/
prefix*"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:aws:logs:region:account-id:log-group:log-group-name:log-stream:log-
stream-name"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "lambda:InvokeFunction",
                "lambda:GetFunctionConfiguration"
            ],
            "Resource": [
                "arn:aws:lambda:region:account-id:function:function-name:function-
version"
            ]
        }
    ]
}
```

La política anterior también incluye una declaración que permite el acceso a Amazon Kinesis Data Streams. Si no utiliza Kinesis Data Streams como origen de datos, puede eliminar dicha declaración. Si utilizas Amazon MSK como fuente, puedes sustituir esa afirmación por la siguiente:

```
{
   "Sid":"",
   "Effect":"Allow",
   "Action":[
      "kafka:GetBootstrapBrokers",
      "kafka:DescribeCluster",
      "kafka:DescribeClusterV2",
      "kafka-cluster:Connect"
   ],
```

```
   "Resource":"arn:aws:kafka:{{mskClusterRegion}}:{{mskClusterAccount}}:cluster/
{{mskClusterName}}/{{clusterUUID}}"
},
{
   "Sid":"",
   "Effect":"Allow",
   "Action":[
      "kafka-cluster:DescribeTopic",
      "kafka-cluster:DescribeTopicDynamicConfiguration",
      "kafka-cluster:ReadData"
   ],
   "Resource":"arn:aws:kafka:{{mskClusterRegion}}:{{mskClusterAccount}}:topic/
{{mskClusterName}}/{{clusterUUID}}/{{mskTopicName}}"
},
{
   "Sid":"",
   "Effect":"Allow",
   "Action":[
      "kafka-cluster:DescribeGroup"
   ],
   "Resource":"arn:aws:kafka:{{mskClusterRegion}}:{{mskClusterAccount}}:group/
{{mskClusterName}}/{{clusterUUID}}/*"
}
```

Para obtener más información sobre cómo permitir que otros AWS servicios accedan a sus AWS recursos, consulte Creación de un rol para delegar permisos a un AWS servicio en la Guía del usuario de IAM.

Para obtener información sobre cómo conceder a Amazon Data Firehose acceso a un destino de Amazon S3 en otra cuenta, consulte. the section called “Entrega entre cuentas en un destino de Amazon S3”

Conceda a Amazon Data Firehose acceso a un destino de Amazon Redshift

Consulte lo siguiente cuando conceda acceso a Amazon Data Firehose cuando utilice un destino de Amazon Redshift.

Temas

- Rol de IAM y política de acceso

- Acceso mediante VPC a un clúster aprovisionado de Amazon Redshift o un grupo de trabajo de Amazon Redshift sin servidor

Rol de IAM y política de acceso

Cuando utiliza un destino de Amazon Redshift, Amazon Data Firehose entrega los datos a su bucket de S3 como ubicación intermedia. Opcionalmente, puede usar una AWS KMS clave de su propiedad para el cifrado de datos. A continuación, Amazon Data Firehose carga los datos del bucket de S3 en el clúster aprovisionado de Amazon Redshift o en el grupo de trabajo Amazon Redshift Serverless. Si el registro de errores está activado, Amazon Data Firehose también envía los errores de entrega de datos al grupo de CloudWatch registros y a las transmisiones. Amazon Data Firehose utiliza el nombre de usuario y la contraseña de Amazon Redshift especificados para acceder al clúster aprovisionado o al grupo de trabajo Amazon Redshift Serverless, y utiliza una función de IAM para acceder al bucket, la clave, el grupo de registros y las transmisiones especificados. CloudWatch Debes tener un rol de IAM al crear una transmisión de Firehose.

Utilice la siguiente política de acceso para permitir que Amazon Data Firehose acceda a su bucket y AWS KMS clave de S3. Si no es propietario del bucket de S3, `s3:PutObjectAcl` añádalo a la lista de acciones de Amazon S3, que otorgan al propietario del bucket acceso total a los objetos entregados por Amazon Data Firehose. Esta política también incluye una declaración que permite el acceso a Amazon Kinesis Data Streams. Si no utiliza Kinesis Data Streams como origen de datos, puede eliminar dicha declaración.

```
{
"Version": "2012-10-17",
    "Statement":
    [
        {
            "Effect": "Allow",
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name",
                "arn:aws:s3:::bucket-name/*"
```

```
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey"
            ],
            "Resource": [
                "arn:aws:kms:region:account-id:key/key-id"
            ],
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "s3.region.amazonaws.com"
                },
                "StringLike": {
                    "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::bucket-name/
prefix*"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStream",
                "kinesis:GetShardIterator",
                "kinesis:GetRecords",
                "kinesis:ListShards"
            ],
            "Resource": "arn:aws:kinesis:region:account-id:stream/stream-name"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:aws:logs:region:account-id:log-group:log-group-name:log-stream:log-
stream-name"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
```

```
                "lambda:InvokeFunction",
                "lambda:GetFunctionConfiguration"
            ],
            "Resource": [
                "arn:aws:lambda:region:account-id:function:function-name:function-
version"
            ]
        }
    ]
}
```

Para obtener más información sobre cómo permitir que otros AWS servicios accedan a sus AWS recursos, consulte Creación de un rol para delegar permisos a un AWS servicio en la Guía del usuario de IAM.

Acceso mediante VPC a un clúster aprovisionado de Amazon Redshift o un grupo de trabajo de Amazon Redshift sin servidor

Si el clúster aprovisionado de Amazon Redshift o el grupo de trabajo de Amazon Redshift sin servidor está en una nube privada virtual (VPC), debe ser de acceso público y tener una dirección IP pública. Además, conceda a Amazon Data Firehose acceso a su clúster aprovisionado de Amazon Redshift o grupo de trabajo Amazon Redshift Serverless desbloqueando las direcciones IP de Amazon Data Firehose. Amazon Data Firehose utiliza actualmente un bloque CIDR para cada región disponible:

- `13.58.135.96/27` para Este de EE. UU. (Ohio)
- `52.70.63.192/27` para Este de EE. UU. (Norte de Virginia)
- `13.57.135.192/27` para Oeste de EE. UU. (Norte de California)
- `52.89.255.224/27` para Oeste de EE. UU. (Oregón)
- `18.253.138.96/27`para AWS GovCloud (este de EE. UU.)
- `52.61.204.160/27`para AWS GovCloud (US-West)
- `35.183.92.128/27` para Canadá (centro)
- `40.176.98.192/27`para Canada West (Calgary)
- `18.162.221.32/27` para Asia-Pacífico (Hong Kong)
- `13.232.67.32/27` para Asia-Pacífico (Bombay)
- `18.60.192.128/27` para Asia-Pacífico (Hyderabad)

- 13.209.1.64/27 para Asia Pacífico (Seúl)
- 13.228.64.192/27 para Asia Pacífico (Singapur)
- 13.210.67.224/27 para Asia Pacífico (Sídney)
- 108.136.221.64/27 para Asia-Pacífico (Yakarta)
- 13.113.196.224/27 para Asia Pacífico (Tokio)
- 13.208.177.192/27 para Asia-Pacífico (Osaka)
- 52.81.151.32/27 para China (Pekín)
- 161.189.23.64/27 para China (Ningxia)
- 16.62.183.32/27 para Europa (Zúrich)
- 35.158.127.160/27 para Europa (Fráncfort)
- 52.19.239.192/27 para Europa (Irlanda)
- 18.130.1.96/27 para Europa (Londres)
- 35.180.1.96/27 para Europa (París)
- 13.53.63.224/27 para Europa (Estocolmo)
- 15.185.91.0/27 para Medio Oriente (Baréin)
- 18.228.1.128/27 para América del Sur (São Paulo)
- 15.161.135.128/27 para Europa (Milán)
- 13.244.121.224/27 para África (Ciudad del Cabo)
- 3.28.159.32/27 para Medio Oriente (EAU)
- 51.16.102.0/27 para Israel (Tel Aviv)
- 16.50.161.128/27 para Asia-Pacífico (Melbourne)

Para obtener más información acerca de cómo desbloquear direcciones IP, consulte el paso Autorización de acceso al clúster en la Guía de introducción a Amazon Redshift.

Conceda a Amazon Data Firehose acceso a un destino de servicio público OpenSearch

Cuando utiliza un destino de OpenSearch servicio, Amazon Data Firehose envía los datos a su clúster de OpenSearch servicios y, al mismo tiempo, realiza copias de seguridad de todos los documentos fallidos o de todos los documentos en su bucket de S3. Si el registro de errores

está activado, Amazon Data Firehose también envía los errores de entrega de datos al grupo de CloudWatch registros y a las transmisiones. Amazon Data Firehose utiliza una función de IAM para acceder al dominio de OpenSearch servicio, al bucket de S3, a la AWS KMS clave y al grupo de CloudWatch registros y a las transmisiones especificados. Debes tener un rol de IAM al crear una transmisión de Firehose.

Utilice la siguiente política de acceso para permitir que Amazon Data Firehose acceda a su bucket, dominio de OpenSearch servicio y AWS KMS clave de S3. Si no es propietario del bucket de S3, `s3:PutObjectAcl` añádalo a la lista de acciones de Amazon S3, lo que otorga al propietario del bucket acceso total a los objetos entregados por Amazon Data Firehose. Esta política también incluye una declaración que permite el acceso a Amazon Kinesis Data Streams. Si no utiliza Kinesis Data Streams como origen de datos, puede eliminar dicha declaración.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name",
                "arn:aws:s3:::bucket-name/*"
            ]
        },
        {
           "Effect": "Allow",
           "Action": [
               "kms:Decrypt",
               "kms:GenerateDataKey"
           ],
           "Resource": [
               "arn:aws:kms:region:account-id:key/key-id"
           ],
           "Condition": {
               "StringEquals": {
```

```
                    "kms:ViaService": "s3.region.amazonaws.com"
                },
                "StringLike": {
                    "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::bucket-name/
prefix*"
                }
            }
        },
        {
           "Effect": "Allow",
           "Action": [
               "es:DescribeDomain",
               "es:DescribeDomains",
               "es:DescribeDomainConfig",
               "es:ESHttpPost",
               "es:ESHttpPut"
           ],
          "Resource": [
              "arn:aws:es:region:account-id:domain/domain-name",
              "arn:aws:es:region:account-id:domain/domain-name/*"
          ]
       },
       {
          "Effect": "Allow",
          "Action": [
              "es:ESHttpGet"
          ],
          "Resource": [
              "arn:aws:es:region:account-id:domain/domain-name/_all/_settings",
              "arn:aws:es:region:account-id:domain/domain-name/_cluster/stats",
              "arn:aws:es:region:account-id:domain/domain-name/index-name*/
_mapping/type-name",
              "arn:aws:es:region:account-id:domain/domain-name/_nodes",
              "arn:aws:es:region:account-id:domain/domain-name/_nodes/stats",
              "arn:aws:es:region:account-id:domain/domain-name/_nodes/*/stats",
              "arn:aws:es:region:account-id:domain/domain-name/_stats",
              "arn:aws:es:region:account-id:domain/domain-name/index-name*/_stats",
              "arn:aws:es:region:account-id:domain/domain-name/"
          ]
       },
       {
          "Effect": "Allow",
          "Action": [
              "kinesis:DescribeStream",
```

```
                "kinesis:GetShardIterator",
                "kinesis:GetRecords",
                "kinesis:ListShards"
            ],
            "Resource": "arn:aws:kinesis:region:account-id:stream/stream-name"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:aws:logs:region:account-id:log-group:log-group-name:log-stream:log-
stream-name"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "lambda:InvokeFunction",
                "lambda:GetFunctionConfiguration"
            ],
            "Resource": [
                "arn:aws:lambda:region:account-id:function:function-name:function-
version"
            ]
        }
    ]
}
```

Para obtener más información sobre cómo permitir que otros AWS servicios accedan a sus AWS recursos, consulte Creación de un rol para delegar permisos a un AWS servicio en la Guía del usuario de IAM.

Para obtener información sobre cómo conceder a Amazon Data Firehose acceso a un clúster de OpenSearch servicios de otra cuenta, consulte. the section called “Entrega multicuenta a un destino de servicio OpenSearch”

Otorgue a Amazon Data Firehose acceso a un destino de OpenSearch servicio en una VPC

Si su dominio de OpenSearch servicio está en una VPC, asegúrese de conceder a Amazon Data Firehose los permisos que se describen en la sección anterior. Además, debe conceder a Amazon Data Firehose los siguientes permisos para que pueda acceder a la VPC de su dominio OpenSearch de servicio.

- `ec2:DescribeVpcs`
- `ec2:DescribeVpcAttribute`
- `ec2:DescribeSubnets`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeNetworkInterfaces`
- `ec2:CreateNetworkInterface`
- `ec2:CreateNetworkInterfacePermission`
- `ec2:DeleteNetworkInterface`

> **Important**
>
> No revoques estos permisos después de crear la transmisión Firehose. Si revocas estos permisos, tu transmisión de Firehose se degradará o dejará de entregar datos a OpenSearch tu dominio de servicio cada vez que el servicio intente consultar o actualizar los ENI.

> **Important**
>
> Cuando especifique subredes para entregar datos al destino en una VPC privada, asegúrese de tener una cantidad suficiente de direcciones IP libres en las subredes elegidas. Si no hay una dirección IP libre disponible en una subred específica, Firehose no puede crear ni añadir ENI para la entrega de datos en la VPC privada y la entrega se degradará o fallará.

Cuando creas o actualizas tu transmisión de Firehose, especificas un grupo de seguridad para que Firehose lo utilice cuando envíe datos a tu dominio de servicio. OpenSearch Puedes usar el mismo grupo de seguridad que usa el dominio del OpenSearch servicio o uno diferente. Si especifica

un grupo de seguridad diferente, asegúrese de que permita el tráfico HTTPS saliente al grupo de seguridad del dominio del OpenSearch servicio. Asegúrese también de que el grupo de seguridad del dominio de OpenSearch servicio permita el tráfico HTTPS desde el grupo de seguridad que especificó al configurar la transmisión de Firehose. Si utilizas el mismo grupo de seguridad tanto para la transmisión de Firehose como para el dominio de OpenSearch servicio, asegúrate de que la regla de entrada del grupo de seguridad permita el tráfico HTTPS. Para obtener más información acerca de las reglas de los grupos de seguridad, consulte Reglas del grupo de seguridad en la documentación de Amazon VPC.

Conceda a Amazon Data Firehose acceso a un destino público sin servidor OpenSearch

Cuando utiliza un destino OpenSearch sin servidor, Amazon Data Firehose envía los datos a OpenSearch su colección sin servidor y, al mismo tiempo, realiza copias de seguridad de todos los documentos fallidos o de todos los documentos en su bucket de S3. Si el registro de errores está activado, Amazon Data Firehose también envía los errores de entrega de datos al grupo de CloudWatch registros y a las transmisiones. Amazon Data Firehose utiliza una función de IAM para acceder a la colección OpenSearch Serverless, el bucket de S3, la AWS KMS clave y el grupo de CloudWatch registros y las transmisiones especificados. Debes tener un rol de IAM al crear una transmisión de Firehose.

Utilice la siguiente política de acceso para permitir que Amazon Data Firehose acceda al bucket de S3, al dominio OpenSearch sin servidor y a la clave. AWS KMS Si no es propietario del bucket de S3, `s3:PutObjectAcl` añádalo a la lista de acciones de Amazon S3, lo que otorga al propietario del bucket acceso total a los objetos entregados por Amazon Data Firehose. Esta política también incluye una declaración que permite el acceso a Amazon Kinesis Data Streams. Si no utiliza Kinesis Data Streams como origen de datos, puede eliminar dicha declaración.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
```

```
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name",
                "arn:aws:s3:::bucket-name/*"
            ]
        },
        {
           "Effect": "Allow",
           "Action": [
               "kms:Decrypt",
               "kms:GenerateDataKey"
           ],
           "Resource": [
               "arn:aws:kms:region:account-id:key/key-id"
           ],
           "Condition": {
               "StringEquals": {
                   "kms:ViaService": "s3.region.amazonaws.com"
               },
               "StringLike": {
                   "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::bucket-name/
prefix*"
               }
           }
        },
       {
          "Effect": "Allow",
          "Action": [
              "kinesis:DescribeStream",
              "kinesis:GetShardIterator",
              "kinesis:GetRecords",
              "kinesis:ListShards"
          ],
          "Resource": "arn:aws:kinesis:region:account-id:stream/stream-name"
       },
       {
          "Effect": "Allow",
          "Action": [
              "logs:PutLogEvents"
          ],
          "Resource": [
              "arn:aws:logs:region:account-id:log-group:log-group-name:log-stream:log-
stream-name"
```

```
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "lambda:InvokeFunction",
                "lambda:GetFunctionConfiguration"
            ],
            "Resource": [
                "arn:aws:lambda:region:account-id:function:function-name:function-
version"
            ]
        },
        {
         "Effect": "Allow",
         "Action": "aoss:APIAccessAll",
         "Resource": "arn:aws:aoss:region:account-id:collection/collection-id"
       }
     ]
}
```

Además de la política anterior, también debe configurar Amazon Data Firehose para que tenga los siguientes permisos mínimos asignados en una política de acceso a datos:

```
[
   {
      "Rules":[
         {
            "ResourceType":"index",
            "Resource":[
               "index/target-collection/target-index"
            ],
            "Permission":[
               "aoss:WriteDocument",
               "aoss:UpdateIndex",
               "aoss:CreateIndex"
            ]
         }
      ],
      "Principal":[
         "arn:aws:sts::account-id:assumed-role/firehose-delivery-role-name/*"
      ]
```

```
    }
]
```

Para obtener más información sobre cómo permitir que otros AWS servicios accedan a sus AWS recursos, consulte Creación de un rol para delegar permisos a un AWS servicio en la Guía del usuario de IAM.

Otorgue a Amazon Data Firehose acceso a un destino OpenSearch sin servidor en una VPC

Si su colección OpenSearch sin servidor está en una VPC, asegúrese de conceder a Amazon Data Firehose los permisos que se describen en la sección anterior. Además, debe conceder a Amazon Data Firehose los siguientes permisos para que pueda acceder a la VPC de su colección OpenSearch Serverless.

- `ec2:DescribeVpcs`
- `ec2:DescribeVpcAttribute`
- `ec2:DescribeSubnets`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeNetworkInterfaces`
- `ec2:CreateNetworkInterface`
- `ec2:CreateNetworkInterfacePermission`
- `ec2:DeleteNetworkInterface`

> **Important**
>
> No revoques estos permisos después de crear la transmisión Firehose. Si revocas estos permisos, tu transmisión de Firehose se degradará o dejará de entregar datos a OpenSearch tu dominio de servicio cada vez que el servicio intente consultar o actualizar los ENI.

> **Important**
>
> Cuando especifique subredes para entregar datos al destino en una VPC privada, asegúrese de tener una cantidad suficiente de direcciones IP libres en las subredes elegidas. Si no hay

una dirección IP libre disponible en una subred específica, Firehose no puede crear ni añadir ENI para la entrega de datos en la VPC privada y la entrega se degradará o fallará.

Cuando creas o actualizas tu transmisión de Firehose, especificas un grupo de seguridad para que Firehose lo use cuando envíe datos a tu colección Serverless. OpenSearch Puedes usar el mismo grupo de seguridad que usa la colección OpenSearch Serverless o uno diferente. Si especifica un grupo de seguridad diferente, asegúrese de que permita el tráfico HTTPS saliente al grupo de seguridad de la colección OpenSearch Serverless. Asegúrese también de que el grupo de seguridad de la colección OpenSearch Serverless permita el tráfico HTTPS desde el grupo de seguridad que especificó al configurar la transmisión Firehose. Si utilizas el mismo grupo de seguridad tanto para la transmisión de Firehose como para la colección OpenSearch Serverless, asegúrate de que la regla de entrada del grupo de seguridad permita el tráfico HTTPS. Para obtener más información acerca de las reglas de los grupos de seguridad, consulte Reglas del grupo de seguridad en la documentación de Amazon VPC.

Conceda a Amazon Data Firehose acceso a un destino de Splunk

Cuando utiliza un destino de Splunk, Amazon Data Firehose envía los datos a su punto de conexión HTTP Event Collector (HEC) de Splunk. También hace una copia de seguridad de esos datos en el depósito de Amazon S3 que especifique y, si lo desea, puede utilizar una AWS KMS clave de su propiedad para el cifrado del lado del servidor de Amazon S3. Si el registro de errores está activado, Firehose envía los errores de entrega de datos a sus flujos de CloudWatch registro. También se puede utilizar AWS Lambda para la transformación de datos.

Si utilizas un balanceador de AWS cargas, asegúrate de que sea un Classic Load Balancer o un Application Load Balancer. Además, habilite las sesiones fijas basadas en la duración con la caducidad de las cookies deshabilitada para Classic Load Balancer y la caducidad establecida en el máximo (7 días) para Application Load Balancer. Para obtener información sobre cómo hacerlo, consulte Duración de la sesión basada en la duración para Classic Load Balancer o Application Load Balancer.

Debes tener un rol de IAM al crear una transmisión de Firehose. Firehose asume esa función de IAM y obtiene acceso al bucket, la clave y el grupo de CloudWatch registros y las transmisiones especificados.

Utilice la siguiente política de acceso para permitir que Amazon Data Firehose acceda a su bucket de S3. Si no es propietario del bucket de S3, `s3:PutObjectAcl` añádalo a la lista de acciones de

Amazon S3, que otorgan al propietario del bucket acceso total a los objetos entregados por Amazon Data Firehose. Esta política también otorga a Amazon Data Firehose acceso para el registro de errores y CloudWatch AWS Lambda para la transformación de datos. La política también incluye una declaración que permite el acceso a Amazon Kinesis Data Streams. Si no utiliza Kinesis Data Streams como origen de datos, puede eliminar dicha declaración. Amazon Data Firehose no utiliza IAM para acceder a Splunk. Para tener acceso a Splunk, utiliza el token de HEC.

```
{
    "Version": "2012-10-17",
    "Statement":
    [
        {
            "Effect": "Allow",
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name",
                "arn:aws:s3:::bucket-name/*"
            ]
        },
        {
           "Effect": "Allow",
           "Action": [
               "kms:Decrypt",
               "kms:GenerateDataKey"
           ],
           "Resource": [
               "arn:aws:kms:region:account-id:key/key-id"
           ],
           "Condition": {
               "StringEquals": {
                   "kms:ViaService": "s3.region.amazonaws.com"
               },
               "StringLike": {
                   "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::bucket-name/
prefix*"
               }
```

```
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStream",
                "kinesis:GetShardIterator",
                "kinesis:GetRecords",
                "kinesis:ListShards"
            ],
            "Resource": "arn:aws:kinesis:region:account-id:stream/stream-name"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:aws:logs:region:account-id:log-group:log-group-name:log-stream:*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "lambda:InvokeFunction",
                "lambda:GetFunctionConfiguration"
            ],
            "Resource": [
                "arn:aws:lambda:region:account-id:function:function-name:function-
version"
            ]
        }
    ]
}
```

Para obtener más información sobre cómo permitir que otros AWS servicios accedan a sus AWS recursos, consulte Crear un rol para delegar permisos a un AWS servicio en la Guía del usuario de IAM.

Acceso a Splunk en VPC

Si la plataforma Splunk está en una VPC, debe ser de acceso público y tener una dirección IP pública. Además, conceda a Amazon Data Firehose acceso a su plataforma Splunk desbloqueando

las direcciones IP de Amazon Data Firehose. Amazon Data Firehose utiliza actualmente los siguientes bloques CIDR.

- `18.216.68.160/27, 18.216.170.64/27, 18.216.170.96/27` para Este de EE. UU. (Ohio)
- `34.238.188.128/26, 34.238.188.192/26, 34.238.195.0/26` para Este de EE. UU. (Norte de Virginia)
- `13.57.180.0/26` para Oeste de EE. UU. (Norte de California)
- `34.216.24.32/27, 34.216.24.192/27, 34.216.24.224/27` para Oeste de EE. UU. (Oregón)
- `18.253.138.192/26`para AWS GovCloud (US-East)
- `52.61.204.192/26`para AWS GovCloud (US-West)
- `18.162.221.64/26` para Asia-Pacífico (Hong Kong)
- `13.232.67.64/26` para Asia-Pacífico (Bombay)
- `13.209.71.0/26` para Asia Pacífico (Seúl)
- `13.229.187.128/26` para Asia Pacífico (Singapur)
- `13.211.12.0/26` para Asia Pacífico (Sídney)
- `13.230.21.0/27, 13.230.21.32/27` para Asia Pacífico (Tokio)
- `51.16.102.64/26` para Israel (Tel Aviv)
- `35.183.92.64/26` para Canadá (centro)
- `40.176.98.128/26`para Canada West (Calgary)
- `18.194.95.192/27, 18.194.95.224/27, 18.195.48.0/27` para Europa (Fráncfort)
- `34.241.197.32/27, 34.241.197.64/27, 34.241.197.96/27` para Europa (Irlanda)
- `18.130.91.0/26` para Europa (Londres)
- `35.180.112.0/26` para Europa (París)
- `13.53.191.0/26` para Europa (Estocolmo)
- `15.185.91.64/26` para Medio Oriente (Baréin)
- `18.228.1.192/26` para América del Sur (São Paulo)
- `15.161.135.192/26` para Europa (Milán)
- `13.244.165.128/26` para África (Ciudad del Cabo)
- `13.208.217.0/26` para Asia-Pacífico (Osaka)
- `52.81.151.64/26` para China (Pekín)

- `161.189.23.128/26` para China (Ningxia)
- `108.136.221.128/26` para Asia-Pacífico (Yakarta)
- `3.28.159.64/26` para Medio Oriente (EAU)
- `51.16.102.64/26` para Israel (Tel Aviv)
- `16.62.183.64/26` para Europa (Zúrich)
- `18.60.192.192/26` para Asia-Pacífico (Hyderabad)
- `16.50.161.192/26` para Asia-Pacífico (Melbourne)

Acceso a Snowflake o al punto final HTTP

No hay ningún subconjunto de rangos de direcciones AWS IP específico de Amazon Data Firehose cuando el destino es un punto final HTTP o clústeres públicos de Snowflake.

Para añadir Firehose a una lista de permitidos para clústeres públicos de Snowflake o a tus puntos de conexión HTTP o HTTPS públicos, añade todos los rangos de direcciones AWS IP actuales a tus reglas de entrada.

> **Note**
>
> Las notificaciones no siempre provienen de direcciones IP de la misma AWS región que el tema asociado. Debes incluir el rango de direcciones AWS IP de todas las regiones.

Conceda a Amazon Data Firehose acceso a un destino con copos de nieve

Cuando utilizas Snowflake como destino, Firehose envía los datos a una cuenta de Snowflake mediante la URL de tu cuenta de Snowflake. También hace copias de seguridad de los datos de error en el depósito de Amazon Simple Storage Service que especifique y, si lo desea, puede utilizar una AWS Key Management Service clave de su propiedad para el cifrado del lado del servidor de Amazon S3. Si el registro de errores está activado, Firehose envía los errores de entrega de datos a tus flujos de CloudWatch Logs.

Debes tener un rol de IAM antes de crear una transmisión de Firehose. Firehose asume esa función de IAM y obtiene acceso al bucket, la clave y el grupo de CloudWatch registros y las transmisiones especificados. Usa la siguiente política de acceso para permitir que Firehose acceda a tu bucket de S3. Si no eres propietario del bucket de S3, añádelo `s3:PutObjectAcl` a la lista de acciones

de Amazon Simple Storage Service, que otorgan al propietario del bucket acceso total a los objetos entregados por Firehose. Esta política también otorga a Firehose acceso CloudWatch para el registro de errores. La política también incluye una declaración que permite el acceso a Amazon Kinesis Data Streams. Si no utiliza Kinesis Data Streams como origen de datos, puede eliminar dicha declaración. Firehose no usa IAM para acceder a Snowflake. Para acceder a Snowflake, utiliza la URL de su cuenta de Snowflake y el ID de PrivateLink Vpce en el caso de un clúster privado.

```
{
"Version": "2012-10-17",
    "Statement":
    [
        {
"Effect": "Allow",
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name",
                "arn:aws:s3:::bucket-name/*"
            ]
        },
        {
"Effect": "Allow",
           "Action": [
               "kms:Decrypt",
               "kms:GenerateDataKey"
           ],
           "Resource": [
               "arn:aws:kms:region:account-id:key/key-id"
           ],
           "Condition": {
"StringEquals": {
"kms:ViaService": "s3.region.amazonaws.com"
               },
               "StringLike": {
"kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::bucket-name/prefix*"
               }
           }
```

```
        },
        {
"Effect": "Allow",
            "Action": [
                "kinesis:DescribeStream",
                "kinesis:GetShardIterator",
                "kinesis:GetRecords",
                "kinesis:ListShards"
            ],
            "Resource": "arn:aws:kinesis:region:account-id:stream/stream-name"
        },
        {
"Effect": "Allow",
            "Action": [
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:aws:logs:region:account-id:log-group:log-group-name:log-stream:*"
            ]
        }
    ]
}
```

Para obtener más información sobre cómo permitir que otros AWS servicios accedan a sus AWS recursos, consulte Creación de un rol para delegar permisos a un AWS servicio en la Guía del usuario de IAM.

Acceso a Snowflake en VPC

Si su clúster de Snowflake tiene habilitados los enlaces privados, Firehose utiliza puntos de enlace de VPC para entregar los datos a su clúster privado sin tener que pasar por la Internet pública. Para ello, crea reglas de red de Snowflake que permitan la entrada desde los siguientes puntos para el clúster en el que se encuentra. `AwsVpceIds` Región de AWS Para obtener más información, consulte Crear una regla de red en la Guía del usuario de Snowflake.

ID de punto de conexión de VPC que se utilizarán en función de las regiones en las que se encuentre el clúster

Región de AWS	**VPCE IDs**
Este de EE. UU. (Ohio)	vpce-0d96cafcd96a50aeb

Región de AWS	**VPCE IDs**
	vpce-0cec34343d48f537b

Región de AWS	**VPCE IDs**
Este de EE. UU. (Norte de Virginia)	vpce-0b4d7e8478e141ba8 vpce-0b75cd681fb507352 vpce-01c03e63820ec00d8 vpce-0c2cfc51dc2882422 vpce-06ca862f019e4e056 vpce-020cda0cfa63f8d1c vpce-0b80504a1a783cd70 vpce-0289b9ff0b5259a96 vpce-0d7add8628bd69a12 vpce-02bfb5966cc59b2af vpce-09e707674af878bf2 vpce-049b52e96cc1a2165 vpce-0bb6c7b7a8a86cdbb vpce-03b22d599f51e80f3 vpce-01d60dc60fc106fe1 vpce-0186d20a4b24ecbef vpce-0533906401a36e416 vpce-05111fb13d396710e vpce-0694613f4fbd6f514 vpce-09b21cb25fe4cc4f4 vpce-06029c3550e4d2399

Región de AWS	**VPCE IDs**
	vpce-00961862a21b033da vpce-01620b9ae33273587 vpce-078cf4ec226880ac9 vpce-0d711bf076ce56381 vpce-066b7e13cbfca6f6e vpce-0674541252d9ccc26 vpce-03540b88dedb4b000 vpce-0b1828e79ad394b95 vpce-0dc0e6f001fb1a60d vpce-0d8f82e71a244098a vpce-00e374d9e3f1af5ce vpce-0c1e3d6631ddb442f
Oeste de EE. UU. (Oregón)	vpce-0f60f72da4cd1e4e7 vpce-0c60d21eb8b1669fd vpce-01c4e3e29afdafbef vpce-0cc6bf2a88da139de vpce-0797e08e169e50662 vpce-033cbe480381b5c0e vpce-00debbdd8f9eb10a5 vpce-08ec2f386c809e889 vpce-0856d14310857b545

Región de AWS	**VPCE IDs**
Europa (Fráncfort)	vpce-068dbb7d71c9460fb vpce-0a7a7f095942d4ec9
Europa (Irlanda)	vpce-06857e59c005a6276 vpce-04390f4f8778b75f2 vpce-011fd2b1f0aa172fd
Asia-Pacífico (Tokio)	vpce-06369e5258144e68a vpce-0f2363cdb8926fbe8
Asia-Pacífico (Singapur)	vpce-049cd46cce7a12d52 vpce-0e8965a1a4bdb8941
Asia-Pacífico (Seúl)	vpce-0aa444d9001e1faa1 vpce-04a49d4dcfd02b884
Asia-Pacífico (Sídney)	vpce-048a60a182c52be63 vpce-03c19949787fd1859

Conceda a Amazon Data Firehose acceso a un destino de punto final HTTP

Puede usar Amazon Data Firehose para entregar datos a cualquier destino de punto final HTTP. Amazon Data Firehose también hace copias de seguridad de esos datos en el bucket de Amazon S3 que especifique y, si lo desea, puede utilizar una AWS KMS clave de su propiedad para el cifrado del lado del servidor de Amazon S3. Si el registro de errores está activado, Amazon Data Firehose envía los errores de entrega de datos a sus flujos de CloudWatch registro. También se puede utilizar AWS Lambda para la transformación de datos.

Debes tener un rol de IAM al crear una transmisión de Firehose. Amazon Data Firehose asume esa función de IAM y obtiene acceso al bucket, la clave y el grupo de CloudWatch registros y las transmisiones especificados.

Utilice la siguiente política de acceso para permitir que Amazon Data Firehose acceda al bucket de S3 que especificó para la copia de seguridad de los datos. Si no es propietario del bucket de S3, `s3:PutObjectAcl` añádalo a la lista de acciones de Amazon S3, que otorgan al propietario del bucket acceso total a los objetos entregados por Amazon Data Firehose. Esta política también otorga a Amazon Data Firehose acceso para el registro de errores y CloudWatch AWS Lambda para la transformación de datos. La política también incluye una declaración que permite el acceso a Amazon Kinesis Data Streams. Si no utiliza Kinesis Data Streams como origen de datos, puede eliminar dicha declaración.

> **Important**
>
> Amazon Data Firehose no utiliza la IAM para acceder a destinos de punto final HTTP propiedad de proveedores de servicios externos compatibles, como Datadog, Dynatrace, LogicMonitor MongoDB, New Relic, Splunk o Sumo Logic. Para acceder a un destino de punto final HTTP específico propiedad de un proveedor de servicios externo compatible, póngase en contacto con ese proveedor de servicios para obtener la clave de API o la clave de acceso necesaria para permitir la entrega de datos a ese servicio desde Amazon Data Firehose.

```
{
    "Version": "2012-10-17",
    "Statement":
    [
        {
            "Effect": "Allow",
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name",
                "arn:aws:s3:::bucket-name/*"
            ]
        },
        {
```

```
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey"
            ],
            "Resource": [
                "arn:aws:kms:region:account-id:key/key-id"
            ],
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "s3.region.amazonaws.com"
                },
                "StringLike": {
                    "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::bucket-name/
prefix*"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStream",
                "kinesis:GetShardIterator",
                "kinesis:GetRecords",
                "kinesis:ListShards"
            ],
            "Resource": "arn:aws:kinesis:region:account-id:stream/stream-name"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:aws:logs:region:account-id:log-group:log-group-name:log-stream:*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "lambda:InvokeFunction",
                "lambda:GetFunctionConfiguration"
            ],
            "Resource": [
```

```
                "arn:aws:lambda:region:account-id:function:function-name:function-
version"
            ]
        }
    ]
}
```

Para obtener más información sobre cómo permitir que otros AWS servicios accedan a sus AWS recursos, consulte Creación de un rol para delegar permisos a un AWS servicio en la Guía del usuario de IAM.

> **Important**
>
> Actualmente, Amazon Data Firehose NO admite la entrega de datos a puntos de enlace HTTP de una VPC.

Entrega entre cuentas desde Amazon MSK

Cuando creas una transmisión de Firehose desde tu cuenta de Firehose (por ejemplo, la cuenta B) y tu fuente es un clúster de MSK en otra AWS cuenta (cuenta A), debes tener implementadas las siguientes configuraciones.

Cuenta A:

1. En la consola de Amazon MSK, elija el clúster aprovisionado y, a continuación, seleccione Propiedades.
2. En Configuración de red, seleccione Editar y active Conectividad con varias VPC.
3. En Configuración de seguridad, seleccione Editar política del clúster.
   a. Si el clúster aún no tiene ninguna política configurada, marque Incluir entidad principal de servicio de Firehose y Habilitar entrega de S3 entre cuentas de Firehose. AWS Management Console Generará automáticamente una política con los permisos adecuados.
   b. Si el clúster ya tiene una política configurada, agregue los siguientes permisos a la política existente:

```
{
      "Effect": "Allow",
```

```
      "Principal": {
        "AWS": "arn:aws:iam::arn:role/mskaasTestDeliveryRole"
      },
      "Action": [
        "kafka:GetBootstrapBrokers",
        "kafka:DescribeCluster",
        "kafka:DescribeClusterV2",
        "kafka-cluster:Connect"
      ],
      "Resource": "arn:aws:kafka:us-east-1:arn:cluster/DO-NOT-TOUCH-mskaas-
provisioned-privateLink/xxxxxxxxx-2f3a-462a-ba09-xxxxxxxxxx-20"  // ARN of the
 cluster
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::arn:role/mskaasTestDeliveryRole"
      },
      "Action": [
        "kafka-cluster:DescribeTopic",
        "kafka-cluster:DescribeTopicDynamicConfiguration",
        "kafka-cluster:ReadData"
      ],
      "Resource": "arn:aws:kafka:us-east-1:arn:topic/DO-NOT-TOUCH-mskaas-
provisioned-privateLink/xxxxxxxxx-2f3a-462a-ba09-xxxxxxxxxx-20/*"//topic of the
 cluster
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::233450236687:role/mskaasTestDeliveryRole"
      },
      "Action": "kafka-cluster:DescribeGroup",
      "Resource": "arn:aws:kafka:us-east-1:arn:group/DO-NOT-TOUCH-mskaas-
provisioned-privateLink/xxxxxxxxx-2f3a-462a-ba09-xxxxxxxxxx-20/*" //topic of
 the cluster
    },
 }
```

4. En Entidad principal de AWS , ingrese el ID de la entidad principal de la cuenta B.
5. En Tema, especifica el tema de Apache Kafka del que quieres que tu transmisión de Firehose ingiera datos. Una vez creada la transmisión de Firehose, no podrás actualizar este tema.
6. Seleccione Save changes (Guardar cambios)

Cuenta B:

1. En la consola Firehose, selecciona Crear transmisión de Firehose con la cuenta B.
2. En Origen, elija Amazon Managed Streaming para Apache Kafka.
3. En Configuración de origen, en Clúster de Amazon Managed Streaming para Apache, ingrese el ARN del clúster de Amazon MSK de la cuenta A.
4. En Tema, especifica el tema de Apache Kafka del que quieres que tu transmisión de Firehose ingiera datos. Una vez creada la transmisión de Firehose, no podrás actualizar este tema.
5. En Nombre de flujo de entrega, especifique el nombre de su transmisión Firehose.

En la cuenta B, cuando cree su transmisión de Firehose, debe tener una función de IAM (que se crea de forma predeterminada al usar la AWS Management Console) que conceda a la transmisión de Firehose acceso de «lectura» al clúster multicuenta de Amazon MSK para el tema configurado.

A continuación se indica lo que configura la AWS Management Console:

```
{
    "Sid": "",
    "Effect": "Allow",
    "Action": [
        "kafka:GetBootstrapBrokers",
        "kafka:DescribeCluster",
        "kafka:DescribeClusterV2",
        "kafka-cluster:Connect"
        ],
    "Resource": "arn:aws:kafka:us-east-1:arn:cluster/DO-NOT-TOUCH-mskaas-provisioned-
privateLink/xxxxxxxxx-2f3a-462a-ba09-xxxxxxxxxx-20/*" //topic of the cluster
    },
    {
    "Sid": "",
    "Effect": "Allow",
    "Action": [
        "kafka-cluster:DescribeTopic",
        "kafka-cluster:DescribeTopicDynamicConfiguration",
        "kafka-cluster:ReadData"
    ],
    "Resource": "arn:aws:kafka:us-east-1:arn:topic/DO-NOT-TOUCH-mskaas-provisioned-
privateLink/xxxxxxxxx-2f3a-462a-ba09-xxxxxxxxxx-20/mskaas_test_topic" //topic of the
 cluster
    },
```

```
    {
    "Sid": "",
    "Effect": "Allow",
    "Action": [
        "kafka-cluster:DescribeGroup"
    ],
    "Resource": "arn:aws:kafka:us-east-1:arn:group/DO-NOT-TOUCH-mskaas-provisioned-
privateLink/xxxxxxxxx-2f3a-462a-ba09-xxxxxxxxxx-20/*" //topic of the cluster
    },
 }
```

Luego, puede completar el paso opcional de configurar la transformación de registros y la conversión del formato de registros. Para obtener más información, consulte Configurar la transformación de registros y la conversión de formato.

Entrega entre cuentas en un destino de Amazon S3

Puede utilizar las API Amazon Data Firehose AWS CLI o las API para crear una transmisión de Firehose en una AWS cuenta con un destino de Amazon S3 en otra cuenta. El siguiente procedimiento muestra un ejemplo de configuración de una transmisión Firehose propiedad de la cuenta A para entregar datos a un bucket de Amazon S3 propiedad de la cuenta B.

1. Cree un rol de IAM en la cuenta A siguiendo los pasos descritos en Conceder a Firehose acceso a un destino de Amazon S3.

   > **Note**
   >
   > En este caso, el bucket de Amazon S3 especificado en la política de acceso es propiedad de la cuenta B. Asegúrese de añadir acciones `s3:PutObjectAcl` a la lista de acciones de Amazon S3 de la política de acceso, que otorgan a la cuenta B acceso total a los objetos entregados por Amazon Data Firehose. Este permiso es necesario para la entrega entre cuentas. Amazon Data Firehose establece el encabezado "x-amz-acl" de la solicitud en "»bucket-owner-full-control.

2. Para permitir el acceso desde el rol de IAM creado anteriormente, cree una política de bucket de S3 en la cuenta B. El código siguiente es un ejemplo de la política del bucket. Para obtener más información, consulte Uso de políticas de bucket y usuario.

   ```
   {
   ```

```
    "Version": "2012-10-17",
    "Id": "PolicyID",
    "Statement": [
        {
            "Sid": "StmtID",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::accountA-id:role/iam-role-name"
            },
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:PutObject",
                "s3:PutObjectAcl"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name",
                "arn:aws:s3:::bucket-name/*"
            ]
        }
    ]
}
```

3. Crea una transmisión de Firehose en la cuenta A con el rol de IAM que creaste en el paso 1.

Entrega multicuenta a un destino de servicio OpenSearch

Puede utilizar las API de Amazon Data Firehose AWS CLI o las API para crear una transmisión de Firehose en una AWS cuenta con un destino de OpenSearch servicio en otra cuenta. El siguiente procedimiento muestra un ejemplo de cómo se puede crear una transmisión Firehose en la cuenta A y configurarla para que entregue datos a un destino de OpenSearch servicio propiedad de la cuenta B.

1. Cree un rol de IAM en la cuenta A siguiendo los pasos que se describen en the section called “Conceda a Amazon Data Firehose acceso a un destino de servicio público OpenSearch ”.
2. Para permitir el acceso desde el rol de IAM que creó en el paso anterior, cree una política de OpenSearch servicio en la cuenta B. El siguiente JSON es un ejemplo.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::Account-A-ID:role/firehose_delivery_role "
      },
      "Action": "es:ESHttpGet",
      "Resource": [
        "arn:aws:es:us-east-1:Account-B-ID:domain/cross-account-cluster/_all/
_settings",
        "arn:aws:es:us-east-1:Account-B-ID:domain/cross-account-cluster/_cluster/
stats",
        "arn:aws:es:us-east-1:Account-B-ID:domain/cross-account-cluster/roletest*/
_mapping/roletest",
        "arn:aws:es:us-east-1:Account-B-ID:domain/cross-account-cluster/_nodes",
        "arn:aws:es:us-east-1:Account-B-ID:domain/cross-account-cluster/_nodes/
stats",
        "arn:aws:es:us-east-1:Account-B-ID:domain/cross-account-cluster/_nodes/*/
stats",
        "arn:aws:es:us-east-1:Account-B-ID:domain/cross-account-cluster/_stats",
        "arn:aws:es:us-east-1:Account-B-ID:domain/cross-account-cluster/roletest*/
_stats",
        "arn:aws:es:us-east-1:Account-B-ID:domain/cross-account-cluster/"
      ]
    }
  ]
}
```

3. Crea una transmisión de Firehose en la cuenta A con el rol de IAM que creaste en el paso 1. Al crear la transmisión Firehose, utilice las API AWS CLI o las API de Amazon Data Firehose y especifique el `ClusterEndpoint` campo en lugar de Servicio. `DomainARN` OpenSearch

> **Note**
>
> Para crear una transmisión de Firehose en una AWS cuenta con un destino de OpenSearch servicio en otra cuenta, debe utilizar las AWS CLI API Firehose de Amazon Data. No puede utilizarlas AWS Management Console para crear este tipo de configuración multicuenta.

Uso de etiquetas para controlar el acceso

Puede utilizar el `Condition` elemento (o `Condition` bloque) opcional en una política de IAM para ajustar el acceso a las operaciones de Amazon Data Firehose en función de las claves y los valores de las etiquetas. En las siguientes subsecciones se describe cómo hacerlo para las distintas operaciones de Amazon Data Firehose. Para obtener más información sobre el uso del elemento `Condition` y los operadores que puede utilizar dentro de él, consulte Elementos de política JSON de IAM: Condition.

CreateDeliveryStream

Para la operación `CreateDeliveryStream`, utilice la clave de condición `aws:RequestTag`. En el ejemplo siguiente, `MyKey` y `MyValue` representan la clave y el valor correspondiente de una etiqueta. Para obtener más información, consulte Conceptos básicos de etiquetas.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": [
            "firehose:CreateDeliveryStream",
            "firehose:TagDeliveryStream"
        ],
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "aws:RequestTag/MyKey": "MyValue"
            }
        }
    }]
}
```

TagDeliveryStream

Para la operación `TagDeliveryStream`, utilice la clave de condición `aws:TagKeys`. En el ejemplo siguiente, `MyKey` es un ejemplo de clave de etiqueta.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Allow",
            "Action": "firehose:TagDeliveryStream",
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": "MyKey"
                }
            }
        }
    ]
}
```

UntagDeliveryStream

Para la operación `UntagDeliveryStream`, utilice la clave de condición `aws:TagKeys`. En el ejemplo siguiente, `MyKey` es un ejemplo de clave de etiqueta.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "firehose:UntagDeliveryStream",
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": "MyKey"
                }
            }
        }
    ]
}
```

ListDeliveryStreams

No se puede utilizar el control de acceso basado en etiquetas con `ListDeliveryStreams`.

Otras operaciones de Amazon Data Firehose

Para todas las operaciones de Amazon Data Firehose que no sean`CreateDeliveryStream`,`TagDeliveryStream`, y `UntagDeliveryStream``ListDeliveryStreams`, utilice la clave de `aws:RequestTag` condición.

En el ejemplo siguiente, `MyKey` y `MyValue` representan la clave y el valor correspondiente de una etiqueta.

`ListDeliveryStreams`, usa la clave de `firehose:ResourceTag` condición para controlar el acceso en función de las etiquetas de esa transmisión de Firehose.

En el ejemplo siguiente, `MyKey` y `MyValue` representan la clave y el valor correspondiente de una etiqueta. La política solo se aplicaría a las transmisiones de Data Firehose que tengan una etiqueta denominada `MyKey` con un valor de. `MyValue` Para obtener más información sobre cómo controlar el acceso en función de las etiquetas de los recursos, consulte Controlar el acceso a AWS los recursos mediante etiquetas en la Guía del usuario de IAM.

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
            "Effect": "Deny",
            "Action": "firehose:DescribeDeliveryStream",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                      "firehose:ResourceTag/MyKey": "MyValue"
                  }
            }
        }
    ]
}
```

Autenticate con AWS Secrets Manager Amazon Data Firehose

Amazon Data Firehose se integra AWS Secrets Manager para proporcionar un acceso seguro a sus datos secretos y automatizar la rotación de credenciales. Esta integración permite a Firehose recuperar un secreto de Secrets Manager en tiempo de ejecución para conectarse a los destinos de streaming mencionados anteriormente y entregar sus flujos de datos. De este modo, tus secretos no están visibles en texto plano durante el flujo de trabajo de creación de transmisiones AWS Management Console ni en los parámetros de la API. Proporciona una práctica segura para administrar sus secretos y lo libera de actividades complejas de administración de credenciales, como la configuración de funciones Lambda personalizadas para administrar la rotación de contraseñas.

Para obtener más información, consulte la Guía del usuario de AWS Secrets Manager.

Comprenda los secretos

Un secreto puede ser una contraseña, un conjunto de credenciales, como un nombre de usuario y una contraseña, un token de OAuth u otra información secreta que se almacene de forma cifrada en Secrets Manager.

Para cada destino, debe especificar el par clave-valor secreto en el formato JSON correcto, como se muestra en la siguiente sección. Amazon Data Firehose no podrá conectarse a tu destino si tu secreto no tiene el formato JSON correcto según el destino.

Formato de secreto para el clúster Amazon Redshift Provisioned y el grupo de trabajo Amazon Redshift Serverless

```
{
    "username":  "<username>",
    "password":  "<password>"
}
```

Formato de secreto para Splunk

```
{
    "hec_token":  "<hec token>"
}
```

Formato del secreto de Snowflake

```
{
    "user":  "<user>",
    "private_key":  "<private_key>",
    "key_passphrase":  "<passphrase>" // optional
}
```

Formato de secreto para el punto final HTTP, Coralogix, Datadog, Dynatrace, Elastic, Honeycomb, LogicMonitor Logz.io, MongoDB Cloud y New Relic

```
{
    "api_key":  "<apikey>"
}
```

Creación de un secreto

Para crear un secreto, siga los pasos que se indican en la sección Crear un AWS Secrets Manager secreto de la Guía del AWS Secrets Manager usuario.

Usa el secreto

Le recomendamos que las utilice AWS Secrets Manager para almacenar sus credenciales o claves para conectarse a destinos de streaming como Amazon Redshift, HTTP Endpoint, Snowflake, Splunk, Coralogix, Datadog, Dynatrace, Elastic, Honeycomb, Logz.io, MongoDB Cloud y New Relic. LogicMonitor

Puede configurar la autenticación con Secrets Manager para estos destinos a través de la consola AWS de administración en el momento de crear la transmisión de Firehose. Para obtener más información, consulte Configurar los ajustes de destino. Como alternativa, también puede usar las operaciones CreateDeliveryStreamy UpdateDestinationAPI para configurar la autenticación con Secrets Manager.

Firehose guarda en caché los secretos con un cifrado y los utiliza para cada conexión a destinos. Actualiza la memoria caché cada 10 minutos para garantizar que se utilicen las credenciales más recientes.

Puedes optar por desactivar la capacidad de recuperar datos secretos de Secrets Manager en cualquier momento durante el ciclo de vida de la transmisión. Si no quieres usar Secrets Manager para recuperar secretos, puedes usar el nombre de usuario/contraseña o la clave de API en su lugar.

> **Note**
>
> Aunque esta función no conlleva ningún coste adicional en Firehose, se te facturará el acceso y el mantenimiento de Secrets Manager. Para obtener más información, consulta la página de AWS Secrets Managerprecios.

Concede acceso a Firehose para recuperar el secreto

Para que Firehose recupere un secreto AWS Secrets Manager, debes proporcionar a Firehose los permisos necesarios para acceder al secreto y la clave que lo cifra.

Cuando se utiliza AWS Secrets Manager para almacenar y recuperar secretos, hay varias opciones de configuración diferentes en función de dónde esté almacenado el secreto y de cómo esté cifrado.

- Si el secreto está almacenado en la misma AWS cuenta que tu función de IAM y está cifrado con la clave AWS gestionada predeterminada (`aws/secretsmanager`), la función de IAM que Firehose asume solo necesita `secretsmanager:GetSecretValue` permiso sobre el secreto.

```
// secret role policy
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "secretsmanager:GetSecretValue",
            "Resource": "Secret ARN"
        }
    ]
}
```

  Para obtener más información sobre las políticas de IAM, consulte los ejemplos de políticas de permisos para. AWS Secrets Manager

- Si el secreto se almacena en la misma cuenta que el rol, pero se cifra con una clave gestionada por el cliente (CMK), el rol necesita ambos permisos`secretsmanager:GetSecretValue`. `kms:Decrypt` La política de CMK también debe permitir el desempeño de la función de IAM. `kms:Decrypt`

```
{
    "Version": "2012-10-17",
    "Statement": [{
            "Effect": "Allow",
            "Action": "secretsmanager:GetSecretValue",
            "Resource": "Secret ARN"
        },
        {
            "Effect": "Allow",
            "Action": "kms:Decrypt",
            "Resource": "KMSKeyARN"
        }
    ]
}
```

- Si el secreto se almacena en una AWS cuenta diferente a la de su rol y se cifra con la clave AWS administrada predeterminada, esta configuración no es posible, ya que Secrets Manager no permite el acceso entre cuentas cuando el secreto está cifrado con la clave AWS administrada.

- Si el secreto se almacena en una cuenta diferente y se cifra con una CMK, el rol de IAM necesita tener `secretsmanager:GetSecretValue` permisos sobre el secreto y `kms:Decrypt` sobre la CMK. La política de recursos del secreto y la política de CMK de la otra cuenta también deben conceder al rol de IAM los permisos necesarios. Para obtener más información, consulta Acceso entre cuentas.

Rota el secreto

La rotación consiste en actualizar periódicamente un secreto. Puede configurarlo AWS Secrets Manager para que rote automáticamente el secreto según el cronograma que especifique. De esta forma, puede reemplazar los secretos a largo plazo por secretos a corto plazo. Esto ayuda a reducir el riesgo de compromiso. Para obtener más información, consulte Rotar AWS Secrets Manager los secretos en la Guía del AWS Secrets Manager usuario.

Gestione las funciones de IAM a través de la consola Amazon Data Firehose

Amazon Data Firehose es un servicio totalmente gestionado que ofrece datos de streaming en tiempo real a los destinos. También puedes configurar Firehose para transformar y convertir el formato de tus datos antes de entregarlos. Para usar estas funciones, primero debes proporcionar funciones de IAM para conceder permisos a Firehose cuando crees o edites una transmisión de Firehose. Firehose usa esta función de IAM para todos los permisos que necesita la transmisión de Firehose.

Por ejemplo, considere un escenario en el que crea una transmisión de Firehose que entrega datos a Amazon S3 y esta transmisión de Firehose tiene los registros de origen de Transform con la función habilitada. AWS Lambda En este caso, debes proporcionar funciones de IAM para conceder a Firehose permisos de acceso al bucket de S3 e invocar la función Lambda, como se muestra a continuación.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "lambdaProcessing",
        "Effect": "Allow",
        "Action": ["lambda:InvokeFunction", "lambda:GetFunctionConfiguration"],
        "Resource": "arn:aws:lambda:us-east-1:<account id>:function:<lambda function
 name>:<lambda function version>"
```

```
    }, {
        "Sid": "s3Permissions",
        "Effect": "Allow",
        "Action": ["s3:AbortMultipartUpload", "s3:GetBucketLocation", "s3:GetObject",
 "s3:ListBucket", "s3:ListBucketMultipartUploads", "s3:PutObject"],
        "Resource": ["arn:aws:s3:::<bucket name>", "arn:aws:s3:::<bucket name>/*"]
    }]
}
```

La consola Firehose te permite elegir cómo quieres proporcionar estas funciones. Puedes elegir una de las siguientes opciones.

- Elija un rol de IAM existente
- Cree un nuevo rol de IAM desde la consola

Elija un rol de IAM existente

Puede elegir uno de los roles de IAM existentes. Con esta opción, asegúrese de que el rol de IAM que elija tenga una política de confianza adecuada y los permisos necesarios para su origen y destino. Para obtener más información, consulte Control del acceso con Amazon Data Firehose.

Cree un nuevo rol de IAM desde la consola

Como alternativa, también puedes usar la consola Firehose para crear un nuevo rol en tu nombre.

Cuando Firehose crea un rol de IAM en su nombre, el rol incluye automáticamente todas las políticas de permisos y confianza que otorgan los permisos necesarios en función de la configuración de transmisión de Firehose.

Por ejemplo, si no habilitaste la función Transformar los registros fuente con la AWS Lambda función, la consola generará la siguiente declaración en la política de permisos.

```
{
  "Sid": "lambdaProcessing",
  "Effect": "Allow",
   "Action": [
     "lambda:InvokeFunction",
     "lambda:GetFunctionConfiguration"
   ],
   "Resource": "arn:aws:lambda:us-east-1:<account id>:function:
%FIREHOSE_POLICY_TEMPLATE_PLACEHOLDER%"
```

```
}
```

> **Note**
>
> Es seguro ignorar las declaraciones de política que contienen`%FIREHOSE_POLICY_TEMPLATE_PLACEHOLDER%`, ya que no otorgan permisos sobre ningún recurso.

La consola crea y edita los flujos de trabajo de Firehose Stream también crea una política de confianza y la adjunta a la función de IAM. La política de confianza permite a Firehose asumir la función de IAM. A continuación se muestra un ejemplo de política de confianza.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "firehoseAssume",
        "Effect": "Allow",
        "Principal": {
            "Service": "firehose.amazonaws.com"
        },
        "Action": "sts:AssumeRole"
    }]
}
```

> **Important**
>
> - Deberías evitar usar la misma función de IAM administrada desde la consola para múltiples transmisiones de Firehose. De lo contrario, la función de IAM podría volverse demasiado permisiva o provocar errores.
> - Para utilizar diferentes declaraciones de política dentro de una política de permisos de una función de IAM gestionada desde una consola, puede crear su propia función de IAM y copiar las declaraciones de política en una política de permisos adjunta a la nueva función. Para adjuntar el rol a la transmisión de Firehose, seleccione la opción Elegir el rol de IAM existente en el acceso al servicio.
> - La consola administra cualquier función de IAM que contenga la cadena service-role en su ARN. Al elegir la opción de rol de IAM existente, asegúrese de seleccionar un rol de IAM

sin la cadena de rol de servicio en su ARN para que la consola no realice ningún cambio en él.

Pasos para crear un rol de IAM desde la consola

1. Abre la consola Firehose en https://console.aws.amazon.com/firehose/.
2. Selecciona Crear transmisión de Firehose.
3. Elige un origen y un destino. Para obtener más información, consulte Crea una transmisión de Firehose.
4. Elija la configuración de destino. Para obtener más información, consulte Configurar los ajustes de destino.
5. En Configuración avanzada, para Acceso al servicio, selecciona Crear o actualizar el rol de IAM.

> **Note**
>
> Se trata de una opción predeterminada. Para usar un rol existente, seleccione la opción Elegir un rol de IAM existente. La consola Firehose no modificará tu rol.

6. Selecciona Crear transmisión de Firehose.

Edita el rol de IAM desde la consola

Al editar una transmisión de Firehose, Firehose actualiza la política de permisos correspondiente en consecuencia para reflejar los cambios de configuración y permisos.

Por ejemplo, si editas la transmisión de Firehose y habilitas la AWS Lambda función Transformar registros fuente con la última versión de la función Lambda a`exampleLambdaFunction`, obtienes la siguiente declaración de política en la política de permisos.

```
{
  "Sid": "lambdaProcessing",
  "Effect": "Allow",
  "Action": [
    "lambda:InvokeFunction",
    "lambda:GetFunctionConfiguration"
  ],
```

```
  "Resource": "arn:aws:lambda:us-east-1:<account id>:function:exampleLambdaFunction:
$LATEST"
}
```

> **Important**
>
> Un rol de IAM administrado desde una consola está diseñado para ser autónomo. No se recomienda modificar la política de permisos o la política de confianza fuera de la consola.

Edita el rol de IAM desde la consola

1. Abre la consola Firehose en https://console.aws.amazon.com/firehose/.
2. Elige las transmisiones de Firehose y elige el nombre de la transmisión de Firehose que quieras actualizar.
3. En la pestaña Configuración, en la sección Acceso al servidor, selecciona Editar.
4. Actualice la opción de rol de IAM.

   > **Note**
   >
   > De forma predeterminada, la consola siempre actualiza un rol de IAM con el patrón service-role en su ARN. Al elegir la opción de rol de IAM existente, asegúrese de seleccionar un rol de IAM sin la cadena de rol de servicio en su ARN para que la consola no realice ningún cambio en él.

5. Elija Guardar cambios.

Supervisión de Amazon Data Firehose

Amazon Data Firehose proporciona funciones de supervisión para las transmisiones de Firehose. Para obtener más información, consulte Supervisión.

Validación de conformidad para Amazon Data Firehose

Los auditores externos evalúan la seguridad y la conformidad de Amazon Data Firehose como parte de varios programas de AWS conformidad. Estos incluyen SOC, PCI, FedRAMP, HIPAA y otros.

Para ver una lista de AWS los servicios incluidos en el ámbito de los programas de conformidad específicos, consulte AWS Servicios incluidos en el ámbito de aplicación por programa de conformidad. Para obtener información general, consulte Programas de conformidad de AWS.

Puede descargar informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulta Descarga de informes en AWS Artifact.

Su responsabilidad de cumplimiento al utilizar Data Firehose viene determinada por la confidencialidad de sus datos, los objetivos de cumplimiento de su empresa y las leyes y reglamentos aplicables. Si su uso de Data Firehose está sujeto al cumplimiento de estándares como HIPAA, PCI o FedRAMP, proporciona recursos para ayudarlo a: AWS

- Guías de inicio rápido sobre seguridad y cumplimiento: estas guías de implementación analizan las consideraciones arquitectónicas y proporcionan los pasos para implementar entornos básicos centrados en la seguridad y el cumplimiento. AWS
- Documento técnico sobre cómo diseñar una arquitectura para la seguridad y el cumplimiento de la HIPAA: este documento técnico describe cómo las empresas pueden utilizar para crear aplicaciones que cumplan con la HIPAA. AWS
- AWS Recursos de cumplimiento: esta colección de libros de trabajo y guías puede aplicarse a su sector y ubicación.
- AWS Config— Este AWS servicio evalúa en qué medida las configuraciones de sus recursos cumplen con las prácticas internas, las directrices del sector y las normativas.
- AWS Security Hub— Este AWS servicio proporciona una visión integral del estado de su seguridad AWS que le ayuda a comprobar el cumplimiento de los estándares y las mejores prácticas del sector de la seguridad.

Resiliencia en Amazon Data Firehose

La infraestructura AWS global se basa en AWS regiones y zonas de disponibilidad. AWS Las regiones proporcionan varias zonas de disponibilidad aisladas y separadas físicamente, que están conectadas mediante redes de baja latencia, alto rendimiento y alta redundancia. Con las zonas de disponibilidad, puede diseñar y utilizar aplicaciones y bases de datos que realizan una conmutación por error automática entre zonas de disponibilidad sin interrupciones. Las zonas de disponibilidad tienen una mayor disponibilidad, tolerancia a errores y escalabilidad que las infraestructuras tradicionales de centros de datos únicos o múltiples.

Para obtener más información sobre AWS las regiones y las zonas de disponibilidad, consulte Infraestructura global.AWS

Además de la infraestructura AWS global, Data Firehose ofrece varias funciones para ayudar a respaldar sus necesidades de respaldo y resiliencia de datos.

Recuperación de desastres

Amazon Data Firehose se ejecuta en modo sin servidor y se ocupa de las degradaciones de los hosts, la disponibilidad de las zonas de disponibilidad y otros problemas relacionados con la infraestructura mediante la migración automática. Cuando esto ocurre, Amazon Data Firehose se asegura de que la transmisión de Firehose se migre sin pérdida de datos.

Seguridad de la infraestructura en Amazon Data Firehose

Como servicio gestionado, Amazon Data Firehose está protegido por la seguridad de la red AWS global. Para obtener información sobre los servicios AWS de seguridad y cómo se AWS protege la infraestructura, consulte Seguridad AWS en la nube. Para diseñar su AWS entorno utilizando las mejores prácticas de seguridad de la infraestructura, consulte Protección de infraestructuras en un marco de buena AWS arquitectura basado en el pilar de la seguridad.

Utiliza las llamadas a la API AWS publicadas para acceder a Firehose a través de la red. Los clientes deben admitir lo siguiente:

- Seguridad de la capa de transporte (TLS). Exigimos TLS 1.2 y recomendamos TLS 1.3.
- Conjuntos de cifrado con confidencialidad directa total (PFS) como DHE (Ephemeral Diffie-Hellman) o ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La mayoría de los sistemas modernos como Java 7 y posteriores son compatibles con estos modos.

Además, las solicitudes deben estar firmadas mediante un ID de clave de acceso y una clave de acceso secreta que esté asociada a una entidad de seguridad de IAM principal. También puede utilizar AWS Security Token Service (AWS STS) para generar credenciales de seguridad temporales para firmar solicitudes.

> **Note**
>
> Para las solicitudes HTTPS salientes, Amazon Data Firehose utiliza una biblioteca HTTP que selecciona automáticamente la versión más alta del protocolo TLS admitida en el lado de destino.

Puntos de enlace de la VPC (PrivateLink)

Amazon Data Firehose admite puntos de enlace de VPC (). PrivateLink Para obtener más información, consulte Uso de Amazon Data Firehose con AWS PrivateLink.

Mejores prácticas de seguridad para Amazon Data Firehose

Amazon Data Firehose proporciona una serie de características de seguridad que debe tener en cuenta a la hora de desarrollar e implementar sus propias políticas de seguridad. Las siguientes prácticas recomendadas son directrices generales y no constituyen una solución de seguridad completa. Puesto que es posible que estas prácticas recomendadas no sean adecuadas o suficientes para el entorno, considérelas como consideraciones útiles en lugar de como normas.

Implementación del acceso a los privilegios mínimos

Al conceder permisos, usted decide quién obtiene qué permisos y qué recursos de Amazon Data Firehose. Habilite las acciones específicas que desea permitir en dichos recursos. Por lo tanto, debe conceder únicamente los permisos obligatorios para realizar una tarea. La implementación del acceso con privilegios mínimos es esencial a la hora de reducir los riesgos de seguridad y el impacto que podrían causar los errores o los intentos malintencionados.

Uso de roles de IAM

Las aplicaciones de productores y clientes deben tener credenciales válidas para acceder a las transmisiones de Firehose, y su transmisión de Firehose debe tener credenciales válidas para acceder a los destinos. No debe almacenar AWS las credenciales directamente en una aplicación cliente o en un bucket de Amazon S3. Estas son las credenciales a largo plazo que no rotan automáticamente y que podrían tener un impacto empresarial significativo si se comprometen.

En su lugar, deberías usar un rol de IAM para administrar las credenciales temporales de tus aplicaciones de productor y cliente para acceder a las transmisiones de Firehose. Al utilizar un rol, no

tiene que utilizar credenciales a largo plazo (como un nombre de usuario y una contraseña o claves de acceso) para acceder a otros recursos.

Para obtener más información, consulte los siguientes temas de la guía del usuario de IAM:

- Roles de IAM
- Situaciones habituales con los roles: usuarios, aplicaciones y servicios

Implementación del cifrado en el servidor en recursos dependientes

Los datos en reposo y los datos en tránsito se pueden cifrar en Amazon Data Firehose. Para obtener más información, consulte Protección de datos en Amazon Amazon Data Firehose.

Se usa CloudTrail para monitorear las llamadas a la API

Amazon Data Firehose está integrado con AWS CloudTrail un servicio que proporciona un registro de las acciones realizadas por un usuario, un rol o un AWS servicio en Amazon Data Firehose.

Con la información recopilada por CloudTrail, puede determinar la solicitud que se realizó a Amazon Data Firehose, la dirección IP desde la que se realizó la solicitud, quién la hizo, cuándo se realizó y detalles adicionales.

Para obtener más información, consulte the section called “Registro de llamadas a la API Firehose de Amazon Data con AWS CloudTrail”.

Transformación de datos en Amazon Data Firehose

Amazon Data Firehose puede invocar la función Lambda para transformar los datos de origen entrantes y entregar los datos transformados a los destinos. Puede habilitar la transformación de datos de Amazon Data Firehose al crear su transmisión de Firehose.

Flujo de transformación de datos

Al activar la transformación de datos de Firehose, Firehose almacena en búfer los datos entrantes. La sugerencia sobre el tamaño del búfer oscila entre 0,2 MB y 3 MB. La sugerencia de tamaño de búfer Lambda predeterminada es de 1 MB para todos los destinos, excepto Splunk y Snowflake. Para Splunk y Snowflake, la sugerencia de almacenamiento en búfer predeterminada es de 256 KB. La sugerencia del intervalo de almacenamiento en búfer de Lambda oscila entre 0 y 900 segundos. La sugerencia de intervalo de almacenamiento en búfer de Lambda predeterminada es de sesenta segundos para todos los destinos excepto Snowflake. Para Snowflake, el intervalo de sugerencia de almacenamiento en búfer predeterminado es de 30 segundos. Para ajustar el tamaño del búfer, defina el ProcessingConfigurationparámetro de la API CreateDeliveryStreamo UpdateDestinationcon la llamada y. ProcessorParameter`BufferSizeInMBsIntervalInSeconds` A continuación, Firehose invoca la función Lambda especificada de forma asíncrona con cada lote almacenado en búfer mediante el modo de invocación sincrónica. AWS Lambda Los datos transformados se envían de Lambda a Firehose. A continuación, Firehose lo envía al destino cuando se alcanza el tamaño o el intervalo de almacenamiento en búfer de destino especificados, lo que ocurra primero.

> **Important**
>
> El modo de invocación sincrónica de Lambda tiene un límite de tamaño de carga de 6 MB para la solicitud y la respuesta. Asegúrese de que el tamaño de almacenamiento en búfer para enviar la solicitud a la función es inferior o igual a 6 MB. Asegúrese también de que la respuesta devuelta por la función no sea superior a 6 MB.

Modelo de estados y transformación de datos

Todos los registros transformados de Lambda deben contener los siguientes parámetros; de lo contrario, Amazon Data Firehose los rechazará y considerará que se trata de un error de transformación de datos.

En el caso de Kinesis Data Streams y Direct PUT:

recordId

El ID de registro se pasa de Amazon Data Firehose a Lambda durante la invocación. El registro transformado debe contener el mismo ID de registro. Cualquier discrepancia entre el ID del registro original y el del transformado se trata como un error de transformación de datos.

resultado

Es el estado de la transformación de los datos del registro. Los valores posibles son `Ok` si el registro se ha transformado correctamente, `Dropped` si la lógica de procesamiento ha omitido el registro intencionadamente y `ProcessingFailed` si el registro no se ha podido transformar. Si un registro tiene el estado `Ok` o`Dropped`, Amazon Data Firehose considera que se ha procesado correctamente. De lo contrario, Amazon Data Firehose considerará que no se ha procesado correctamente.

datos

Es la carga útil de datos transformados después codificarlos en base64.

A continuación se presenta un ejemplo de salida de Lambda:

```
{
   "recordId": "<recordId from the Lambda input>",
   "result": "Ok",
   "data": "<Base64 encoded Transformed data>"
}
```

En el caso de Amazon MSK:

recordId

El ID del registro se pasa de Firehose a Lambda durante la invocación. El registro transformado debe contener el mismo ID de registro. Cualquier discrepancia entre el ID del registro original y el del transformado se trata como un error de transformación de datos.

resultado

Es el estado de la transformación de los datos del registro. Los valores posibles son `Ok` si el registro se ha transformado correctamente, `Dropped` si la lógica de procesamiento ha omitido el registro intencionadamente y `ProcessingFailed` si el registro no se ha podido

transformar. Si un registro tiene el estado 0k o`Dropped`, Firehose considera que se ha procesado correctamente. De lo contrario, Firehose considerará que no se ha procesado correctamente.

KafkaRecordValue

Es la carga útil de datos transformados después codificarlos en base64.

A continuación se presenta un ejemplo de salida de Lambda:

```
 {
    "recordId": "<recordId from the Lambda input>",
    "result": "Ok",
    "kafkaRecordValue": "<Base64 encoded Transformed data>"
}
```

Esquemas de Lambda

Estos planos muestran cómo puede crear y utilizar las funciones de AWS Lambda para transformar los datos de las transmisiones de datos de Amazon Data Firehose.

Para ver los planos que están disponibles en la consola AWS Lambda

1. Inicie sesión en la AWS Lambda consola AWS Management Console y ábrala en https://console.aws.amazon.com/lambda/.
2. Elija Create function (Crear función) y, a continuación, elija Use a blueprint (Utilizar un proyecto).
3. En el campo Blueprints, busque la palabra clave `firehose` para encontrar los blueprints de Amazon Data Firehose Lambda.

Lista de esquemas:

- Registros de procesos enviados a Amazon Data Firehose stream (Node.js, Python)

  Este plano muestra un ejemplo básico de cómo procesar los datos del flujo de datos de Firehose con AWS Lambda.

  Fecha de lanzamiento más reciente: noviembre de 2016.

  Notas de la versión: ninguna.

- CloudWatch Registros del proceso enviados a Firehose

  Este modelo está obsoleto. Para obtener información sobre el procesamiento de CloudWatch los registros enviados a Firehose, consulte Cómo escribir en Firehose mediante registros. CloudWatch
- Convierta los registros de transmisión de Amazon Data Firehose en formato syslog a JSON (Node.js)

  En este esquema se muestra cómo convertir los registros de entrada en formato RFC3164 de Syslog en JSON.

  Fecha de lanzamiento más reciente: noviembre de 2016.

  Notas de la versión: ninguna.

Para ver los planos que están disponibles en el AWS Serverless Application Repository

1. Vaya a AWS Serverless Application Repository.
2. Elija Examinar todas las aplicaciones.
3. En el campo Applications (Aplicaciones) busque la palabra clave `firehose`.

También puede crear una función de Lambda sin utilizar un esquema. Consulte Introducción a AWS Lambda.

Gestión de errores de transformación de datos

Si se produce un error en la invocación de la función Lambda porque se ha agotado el tiempo de espera de la red o porque se ha alcanzado el límite de invocación de Lambda, Amazon Data Firehose vuelve a intentar la invocación tres veces de forma predeterminada. Si la invocación no se realiza correctamente, Amazon Data Firehose omite ese lote de registros. los trata como registros que no se han podido procesar. Puede especificar o anular las opciones de reintento mediante la API o. CreateDeliveryStream`UpdateDestination` Para este tipo de error, puede registrar los errores de invocación en Amazon CloudWatch Logs. Para obtener más información, consulte Supervisión de Amazon Data Firehose mediante registros CloudWatch .

Si el estado de la transformación de datos de un registro es`ProcessingFailed`, Amazon Data Firehose considera que el registro no se ha procesado correctamente. Para este tipo de error, puede enviar registros de errores a Amazon CloudWatch Logs desde su función Lambda. Para obtener

más información, consulte Acceder a Amazon CloudWatch Logs AWS Lambda en la Guía para AWS Lambda desarrolladores.

Si ocurre un error durante la transformación de datos, los registros que no se hayan podido procesar se entregan en el bucket de S3, en la carpeta `processing-failed`. Los registros tienen el siguiente formato:

```
{
    "attemptsMade": "count",
    "arrivalTimestamp": "timestamp",
    "errorCode": "code",
    "errorMessage": "message",
    "attemptEndingTimestamp": "timestamp",
    "rawData": "data",
    "lambdaArn": "arn"
}
```

`attemptsMade`

La cantidad de intentos de solicitud de invocación.

`arrivalTimestamp`

La hora a la que Amazon Data Firehose recibió el registro.

`errorCode`

Código de error HTTP devuelto por Lambda.

`errorMessage`

Mensaje de error HTTP devuelto Lambda.

`attemptEndingTimestamp`

La hora en que Amazon Data Firehose dejó de intentar invocar a Lambda.

`rawData`

Los datos de registros codificados en base64.

`lambdaArn`

El nombre de recurso de Amazon (ARN) de la función de Lambda.

Duración de una invocación de Lambda

Amazon Data Firehose admite un tiempo de invocación a Lambda de hasta 5 minutos. Si la función Lambda tarda más de 5 minutos en completarse, aparece el siguiente error: Firehose detectó errores de tiempo de espera al llamar a Lambda. AWS The maximum supported function timeout is 5 minutes.

Para obtener información sobre lo que hace Amazon Data Firehose si se produce un error de este tipo, consulte. the section called “Gestión de errores de transformación de datos”

Backup de registros de origen

Amazon Data Firehose puede realizar copias de seguridad de todos los registros no transformados en su bucket de S3 de forma simultánea y, al mismo tiempo, entregar los registros transformados al destino. Puedes habilitar la copia de seguridad de los registros de origen al crear o actualizar tu transmisión de Firehose. El backup de los registros de origen no se puede deshabilitar después de haberlo habilitado.

Particionamiento dinámico en Amazon Data Firehose

El particionamiento dinámico le permite particionar continuamente los datos de streaming en Firehose mediante claves dentro de los datos (por ejemplo`customer_id`, `transaction_id` o) y, a continuación, entregar los datos agrupados por estas claves en los prefijos correspondientes del Amazon Simple Storage Service (Amazon S3). Esto facilita la ejecución de análisis rentables y de alto rendimiento sobre datos de streaming en Amazon S3 mediante diversos servicios, como Amazon Athena, Amazon EMR, Amazon Redshift Spectrum y Amazon. QuickSight Además, AWS Glue puede realizar tareas de extracción, transformación y carga (ETL) más sofisticadas después de que los datos de streaming particionados dinámicamente se entreguen a Amazon S3, en casos de uso en los que se requiera un procesamiento adicional.

El particionamiento de los datos minimiza la cantidad de datos analizados, optimiza el rendimiento y reduce los costos de las consultas de análisis en Amazon S3. También aumenta el acceso granular a los datos. Las transmisiones Firehose se utilizan tradicionalmente para capturar y cargar datos en Amazon S3. Para particionar un conjunto de datos de streaming con el objetivo de llevar a cabo análisis basados en Amazon S3, tendría que ejecutar aplicaciones de particionamiento entre buckets de Amazon S3 antes de hacer que los datos estén disponibles para su análisis, lo que podría resultar complicado o costoso.

Con el particionamiento dinámico, Firehose agrupa continuamente los datos en tránsito mediante claves de datos definidas de forma dinámica o estática y entrega los datos a los prefijos individuales de Amazon S3 por clave. Esto se reduce time-to-insight en minutos u horas. También reduce los costos y simplifica las arquitecturas.

Temas

- Claves de particionamiento
- Prefijo de bucket de Amazon S3 para particionamiento dinámico
- Particionamiento dinámico de datos agregados
- Adición de un delimitador de nueva línea al entregar datos en S3
- Cómo habilitar el particionamiento dinámico
- Gestión de errores de particionamiento dinámico
- Almacenamiento en búfer y particionamiento dinámico de datos

Claves de particionamiento

Con el particionamiento dinámico, crea conjuntos de datos específicos a partir de los datos de S3 de streaming mediante el particionamiento de los datos basado en claves de particionamiento. Las claves de particionamiento le permiten filtrar los datos de streaming en función de valores específicos. Por ejemplo, si necesita filtrar los datos en función del ID del cliente y el país, puede especificar el campo de datos `customer_id` como clave de particionamiento y el campo de datos `country` como otra clave de particionamiento. A continuación, especifique las expresiones (con los formatos admitidos) para definir los prefijos de los buckets de S3 en los que se entregarán los registros de datos particionados de forma dinámica.

A continuación se indican los métodos admitidos para crear claves de particionamiento:

- Análisis en línea: este método utiliza el mecanismo de soporte integrado de Firehose, un analizador jq, para extraer las claves para la partición de los registros de datos que están en formato JSON. Actualmente, solo admitimos la versión. `jq 1.6`
- AWS Función Lambda: este método utiliza una función AWS Lambda específica para extraer y devolver los campos de datos necesarios para la partición.

> **Important**
>
> Al habilitar el particionamiento dinámico, debe configurar al menos uno de estos métodos para particionar los datos. Puede configurar uno de estos métodos para especificar las claves de particionamiento o ambos al mismo tiempo.

Creación de claves de particionamiento con análisis en línea

Para configurar el análisis en línea como método de particionamiento dinámico para sus datos de streaming, debe elegir los parámetros de registro de datos que se utilizarán como claves de particionamiento y proporcionar un valor para cada clave de particionamiento especificada.

En el siguiente ejemplo de registro de datos, se muestra cómo se pueden definir las claves de partición para él mediante el análisis en línea. Tenga en cuenta que los datos deben codificarse en formato Base64. También puede consultar el ejemplo de CLI.

```
{
   "type": {
```

```
    "device": "mobile",
    "event": "user_clicked_submit_button"
  },
  "customer_id": "1234567890",
  "event_timestamp": 1565382027,   #epoch timestamp
  "region": "sample_region"
}
```

Por ejemplo, puede elegir particionar los datos en función del parámetro `customer_id` o del parámetro `event_timestamp`. Esto significa que desea que el valor del parámetro `customer_id` o del parámetro `event_timestamp` de cada registro se utilice para determinar el prefijo de S3 en el que se entregará el registro. También puede elegir un parámetro anidado, como `device` con una expresión `.type.device`. La lógica de particionamiento dinámico puede depender de varios parámetros.

Tras seleccionar los parámetros de datos para las claves de particionamiento, asigne cada parámetro a una expresión jq válida. En la siguiente tabla se muestra este tipo de asignación de parámetros a expresiones jq:

Parámetro	Expresión jq
`customer_id`	.customer_id
`device`	.type.device
`year`	.event_timestamp| strftime("%Y")
`month`	.event_timestamp| strftime("%m")
`day`	.event_timestamp| strftime("%d")
`hour`	.event_timestamp| strftime("%H")

En tiempo de ejecución, Firehose utiliza la columna derecha de arriba para evaluar los parámetros en función de los datos de cada registro.

Creación de claves de particionamiento con una función de AWS Lambda

Para los registros de datos comprimidos o cifrados, o los datos que estén en cualquier formato de archivo que no sea JSON, puede utilizar la función AWS Lambda integrada con su propio código

personalizado para descomprimir, descifrar o transformar los registros con el fin de extraer y devolver los campos de datos necesarios para la partición. Se trata de una expansión de la función Lambda de transformación existente que está disponible en la actualidad con Firehose. Puede transformar, analizar y devolver los campos de datos que luego puede usar para el particionamiento dinámico con la misma función de Lambda.

El siguiente es un ejemplo de la función Lambda de procesamiento de flujos Firehose en Python que reproduce todos los registros leídos desde la entrada hasta la salida y extrae las claves de partición de los registros.

```
from __future__ import print_function
import base64
import json
import datetime

# Signature for all Lambda functions that user must implement
def lambda_handler(firehose_records_input, context):
    print("Received records for processing from DeliveryStream: " +
 firehose_records_input['deliveryStreamArn']
          + ", Region: " + firehose_records_input['region']
          + ", and InvocationId: " + firehose_records_input['invocationId'])

    # Create return value.
    firehose_records_output = {'records': []}

    # Create result object.
    # Go through records and process them

    for firehose_record_input in firehose_records_input['records']:
        # Get user payload
        payload = base64.b64decode(firehose_record_input['data'])
        json_value = json.loads(payload)

        print("Record that was received")
        print(json_value)
        print("\n")
        # Create output Firehose record and add modified payload and record ID to it.
        firehose_record_output = {}
        event_timestamp = datetime.datetime.fromtimestamp(json_value['eventTimestamp'])
        partition_keys = {"customerId": json_value['customerId'],
                          "year": event_timestamp.strftime('%Y'),
                          "month": event_timestamp.strftime('%m'),
```

```
                        "date": event_timestamp.strftime('%d'),
                        "hour": event_timestamp.strftime('%H'),
                        "minute": event_timestamp.strftime('%M')
                        }

        # Create output Firehose record and add modified payload and record ID to it.
        firehose_record_output = {'recordId': firehose_record_input['recordId'],
                                  'data': firehose_record_input['data'],
                                  'result': 'Ok',
                                  'metadata': { 'partitionKeys': partition_keys }}

        # Must set proper record ID
        # Add the record to the list of output records.

        firehose_records_output['records'].append(firehose_record_output)

    # At the end return processed records
    return firehose_records_output
```

A continuación se muestra un ejemplo de la función Lambda de procesamiento de flujos Firehose en Go que reproduce todos los registros leídos desde la entrada hasta la salida y extrae las claves de partición de los registros.

```
package main

import (
 "fmt"
 "encoding/json"
 "time"
 "strconv"

 "github.com/aws/aws-lambda-go/events"
 "github.com/aws/aws-lambda-go/lambda"
)

type DataFirehoseEventRecordData struct {
 CustomerId string `json:"customerId"`
}

func handleRequest(evnt events.DataFirehoseEvent) (events.DataFirehoseResponse, error)
 {
```

```
 fmt.Printf("InvocationID: %s\n", evnt.InvocationID)
 fmt.Printf("DeliveryStreamArn: %s\n", evnt.DeliveryStreamArn)
 fmt.Printf("Region: %s\n", evnt.Region)

 var response events.DataFirehoseResponse

 for _, record := range evnt.Records {
  fmt.Printf("RecordID: %s\n", record.RecordID)
  fmt.Printf("ApproximateArrivalTimestamp: %s\n", record.ApproximateArrivalTimestamp)

  var transformedRecord events.DataFirehoseResponseRecord
  transformedRecord.RecordID = record.RecordID
  transformedRecord.Result = events.DataFirehoseTransformedStateOk
  transformedRecord.Data = record.Data

  var metaData events.DataFirehoseResponseRecordMetadata
  var recordData DataFirehoseEventRecordData
  partitionKeys := make(map[string]string)

  currentTime := time.Now()
  json.Unmarshal(record.Data, &recordData)
  partitionKeys["customerId"] = recordData.CustomerId
  partitionKeys["year"] = strconv.Itoa(currentTime.Year())
  partitionKeys["month"] = strconv.Itoa(int(currentTime.Month()))
  partitionKeys["date"] = strconv.Itoa(currentTime.Day())
  partitionKeys["hour"] = strconv.Itoa(currentTime.Hour())
  partitionKeys["minute"] = strconv.Itoa(currentTime.Minute())
  metaData.PartitionKeys = partitionKeys
  transformedRecord.Metadata = metaData

  response.Records = append(response.Records, transformedRecord)
 }

 return response, nil
}

func main() {
 lambda.Start(handleRequest)
}
```

Prefijo de bucket de Amazon S3 para particionamiento dinámico

Cuando crea una transmisión de Firehose que utiliza Amazon S3 como destino, debe especificar un bucket de Amazon S3 al que Firehose entregará sus datos. Los prefijos de buckets de Amazon S3 se utilizan para organizar los datos que almacena en los buckets de Amazon S3. Un prefijo de bucket de Amazon S3 es similar a un directorio que permite agrupar objetos similares.

Con el particionamiento dinámico, los datos particionados se entregan en los prefijos de Amazon S3 especificados. Si no habilitas la partición dinámica, es opcional especificar un prefijo de bucket de S3 para tu transmisión de Firehose. Sin embargo, si opta por habilitar la partición dinámica, debe especificar los prefijos del bucket S3 a los que Firehose entrega los datos particionados.

En todas las transmisiones de Firehose en las que se habilita la partición dinámica, el valor del prefijo del bucket de S3 se compone de expresiones basadas en las claves de partición especificadas para esa transmisión de Firehose. Si vuelve a utilizar el ejemplo de registro de datos anterior, puede crear el siguiente valor de prefijo de S3, que consta de expresiones que se basan en las claves de particionamiento definidas anteriormente:

```
"ExtendedS3DestinationConfiguration": {
"BucketARN": "arn:aws:s3:::my-logs-prod",
"Prefix": "customer_id=!{partitionKeyFromQuery:customer_id}/
    device=!{partitionKeyFromQuery:device}/
    year=!{partitionKeyFromQuery:year}/
    month=!{partitionKeyFromQuery:month}/
    day=!{partitionKeyFromQuery:day}/
    hour=!{partitionKeyFromQuery:hour}/"
}
```

Firehose evalúa la expresión anterior en tiempo de ejecución. Agrupa los registros que coinciden con la misma expresión de prefijo de S3 evaluada en un único conjunto de datos. A continuación, Firehose envía cada conjunto de datos al prefijo S3 evaluado. La frecuencia de entrega del conjunto de datos a S3 viene determinada por la configuración del búfer de flujo Firehose. Como resultado, el registro de este ejemplo se entrega en la siguiente clave de objeto de S3:

```
s3://my-logs-prod/customer_id=1234567890/device=mobile/year=2019/month=08/day=09/
hour=20/my-delivery-stream-2019-08-09-23-55-09-a9fa96af-e4e4-409f-bac3-1f804714faaa
```

En el caso del particionamiento dinámico, debe usar el siguiente formato de expresión en el prefijo de bucket de S3: `!{namespace:value}`, donde el espacio de nombres puede ser `partitionKeyFromQuery`, `partitionKeyFromLambda` o ambos. Si utiliza el análisis en línea para crear las claves de particionamiento para sus datos de origen, debe especificar un valor de prefijo de bucket de S3 que conste de expresiones especificadas en el siguiente formato: `"partitionKeyFromQuery:keyID"`. Si utiliza una función de AWS Lambda para crear claves de particionamiento para sus datos de origen, debe especificar un valor de prefijo de bucket de S3 que conste de expresiones especificadas en el siguiente formato: `"partitionKeyFromLambda:keyID"`.

> **Note**
>
> También puedes especificar el valor del prefijo del bucket de S3 con el formato de colmena, por ejemplo customer_id=! partitionKeyFrom{query:customer_ID}.

Para obtener más información, consulte la sección «Elija Amazon S3 como destino» en Creación de una transmisión de Amazon Firehose y prefijos personalizados para objetos de Amazon S3.

Particionamiento dinámico de datos agregados

Puede aplicar el particionamiento dinámico a los datos agregados (por ejemplo, varios eventos, registros o registros agregados en una sola llamada a la API `PutRecord` y `PutRecordBatch`), pero primero se deben desagregar estos datos. Puede desagregar sus datos habilitando la desagregación de varios registros, el proceso de analizar los registros del flujo Firehose y separarlos.

La desagregación de varios registros puede ser de cualquier `JSON` tipo, lo que significa que la separación de registros se basa en objetos JSON consecutivos. La desagregación también puede ser de este tipo`Delimited`, lo que significa que la separación de registros se realiza en función de un delimitador personalizado específico. Este delimitador personalizado debe ser una cadena codificada en base64. Por ejemplo, si desea utilizar la siguiente cadena como delimitador personalizado####, debe especificarla en el formato codificado en base 64, que es lo que traduce a. `IyMjIw==`

> **Note**
>
> Al desagregar registros JSON, asegúrate de que la entrada siga presentándose en el formato JSON compatible. Los objetos JSON deben estar en una sola línea sin delimitador o estar

únicamente delimitados por líneas nuevas (JSONL). Una matriz de objetos JSON no es una entrada válida.
Estos son ejemplos de entradas correctas: `{"a":1}{"a":2}` and `{"a":1}\n{"a":2}`
Este es un ejemplo de una entrada incorrecta: `[{"a":1}, {"a":2}]`

Con los datos agregados, al habilitar el particionamiento dinámico, Firehose analiza los registros y busca objetos JSON válidos o registros delimitados dentro de cada llamada a la API en función del tipo de desagregación de registros múltiples especificado.

Important

Si sus datos están agregados, el particionamiento dinámico solo se puede aplicar si primero se desagregan los datos.

Important

Cuando utilizas la función de transformación de datos en Firehose, la desagregación se aplicará antes de la transformación de datos. Los datos que lleguen a Firehose se procesarán en el siguiente orden: Desagregación → Transformación de datos mediante Lambda → Claves de partición.

Adición de un delimitador de nueva línea al entregar datos en S3

Puede activar New Line Delimiter para añadir un nuevo delimitador de línea entre los registros de los objetos que se envían a Amazon S3. Esto puede resultar útil para analizar objetos en Amazon S3. Esto también resulta especialmente útil cuando se aplica una partición dinámica a los datos agregados, ya que la desagregación de varios registros (que debe aplicarse a los datos agregados para poder particionarlos dinámicamente) elimina las nuevas líneas de los registros como parte del proceso de análisis.

Cómo habilitar el particionamiento dinámico

Puede configurar el particionamiento dinámico para las transmisiones de Firehose a través de la consola de administración de Amazon Data Firehose, la CLI o las API.

> **Important**
>
> Solo puedes habilitar la partición dinámica cuando creas una nueva transmisión de Firehose. No puedes habilitar la partición dinámica para una transmisión de Firehose existente que no tenga habilitada la partición dinámica.

Para ver los pasos detallados sobre cómo habilitar y configurar la partición dinámica a través de la consola de administración de Firehose al crear una nueva transmisión de Firehose, consulte Creación de una transmisión de Amazon Firehose. Cuando se dedique a la tarea de especificar el destino de su transmisión de Firehose, asegúrese de seguir los pasos de la sección Elija Amazon S3 como destino, ya que, actualmente, la partición dinámica solo es compatible con las transmisiones de Firehose que utilizan Amazon S3 como destino.

Una vez que se habilita la partición dinámica en una transmisión Firehose activa, puedes actualizar la configuración añadiendo nuevas o quitando o actualizando las claves de partición existentes y las expresiones de prefijo S3. Una vez actualizada, Firehose comienza a usar las nuevas claves y las nuevas expresiones de prefijo S3.

> **Important**
>
> Una vez que habilitas la partición dinámica en una transmisión de Firehose, no se puede deshabilitar en esta transmisión de Firehose.

Gestión de errores de particionamiento dinámico

Si Amazon Data Firehose no puede analizar los registros de datos de su transmisión Firehose o no puede extraer las claves de partición especificadas ni evaluar las expresiones incluidas en el valor del prefijo S3, estos registros de datos se envían al prefijo del bucket de error S3 que debe especificar al crear la transmisión Firehose, donde habilita la partición dinámica. El prefijo del depósito de errores S3 contiene todos los registros que Firehose no puede entregar al destino S3 especificado. Estos registros se organizan en función del tipo de error. Junto con el registro, el objeto entregado también incluye información sobre el error para ayudar a comprenderlo y resolverlo.

Debe especificar un prefijo de bucket de error S3 para una transmisión de Firehose si quiere habilitar la partición dinámica para esta transmisión de Firehose. Si no quieres habilitar la partición dinámica para una transmisión de Firehose, es opcional especificar un prefijo de bucket de error S3.

Almacenamiento en búfer y particionamiento dinámico de datos

Amazon Data Firehose almacena en búfer los datos de streaming entrantes hasta un tamaño determinado y durante un período de tiempo determinado antes de entregarlos a los destinos especificados. Puedes configurar el tamaño del búfer y el intervalo del búfer al crear nuevos flujos de Firehose o actualizar el tamaño y el intervalo del búfer en tus flujos Firehose existentes. El tamaño de un búfer se mide en MB y el intervalo de un búfer, en segundos.

Cuando la partición dinámica está habilitada, Firehose almacena internamente en búfer los registros que pertenecen a una partición determinada en función de la sugerencia de almacenamiento en búfer configurada (tamaño y tiempo) antes de entregar estos registros a su bucket de Amazon S3. Para entregar objetos de tamaño máximo, Firehose utiliza internamente un búfer de varias etapas. Por lo tanto, el end-to-end retraso de un lote de registros puede ser 1,5 veces mayor que el tiempo de sugerencia de almacenamiento en búfer configurado. Esto afecta a la actualización de los datos de una transmisión de Firehose.

El recuento de particiones activas es el número total de particiones activas en el búfer de entrega. Por ejemplo, si la consulta de particionamiento dinámico crea 3 particiones por segundo y tiene una configuración de sugerencias de búfer que activa la entrega cada 60 segundos, tendrá un promedio de 180 particiones activas. Si Firehose no puede entregar los datos de una partición a un destino, esta partición se cuenta como activa en el búfer de entrega hasta que se pueda entregar.

Se crea una nueva partición cuando se evalúa un prefijo de S3 para obtener un nuevo valor en función de los campos de datos de registro y las expresiones de prefijos de S3. Se crea un búfer nuevo para cada partición activa. Todos los registros posteriores con el mismo prefijo de S3 evaluado se envían a ese búfer.

Una vez que el búfer alcanza el límite de tamaño del búfer o el intervalo de tiempo del búfer, Firehose crea un objeto con los datos del búfer y lo envía al prefijo Amazon S3 especificado. Una vez entregado el objeto, el búfer de esa partición y la propia partición se eliminan y se eliminan del recuento de particiones activas.

Firehose entrega los datos de cada búfer como un único objeto una vez que se alcanza el tamaño o el intervalo del búfer para cada partición por separado. Cuando el número de particiones activas alcanza un límite de 500 por transmisión de Firehose, el resto de los registros de la transmisión de Firehose se envían al prefijo de depósito de errores de S3 especificado (). activePartitionExceeded Puede utilizar el formulario Amazon Data Firehose Limits para solicitar un aumento de esta cuota hasta 5000 particiones activas por transmisión de Firehose determinada. Si necesitas más

particiones, puedes crear más transmisiones de Firehose y distribuir las particiones activas entre ellas.

Cómo convertir el formato de registro de entrada en Firehose

Amazon Data Firehose puede convertir el formato de los datos de entrada de JSON a Apache Parquet o Apache ORC antes de almacenar los datos en Amazon S3. Parquet y ORC son formatos de datos en columnas que ahorran espacio y permiten unas búsquedas más rápidas en comparación con los formatos orientados a filas como JSON. Si desea convertir un formato de entrada distinto de JSON, como valores separados por comas (CSV) o texto estructurado, puede transformarlo primero AWS Lambda a JSON. Para obtener más información, consulte Transformación de datos.

Temas

- Requisitos de la conversión del formato de registro
- Elección del deserializador JSON
- Elección del serializador
- Conversión del formato de registro de entrada (consola)
- Conversión del formato de registro de entrada (API)
- Control de errores de la conversión del formato de registro
- Ejemplo de conversión del formato de registros

Requisitos de la conversión del formato de registro

Amazon Data Firehose requiere los tres elementos siguientes para convertir el formato de los datos de sus registros:

- Un deserializador para leer el JSON de los datos de entrada: puede elegir uno de los dos tipos de deserializadores: Apache Hive JSON u OpenX JSON. SerDe SerDe

> **Note**
>
> Al combinar varios documentos JSON en el mismo registro, asegúrese de que la entrada siga presentándose en el formato JSON compatible. Una matriz de documentos JSON no es una entrada válida.
> Por ejemplo, esta es la entrada correcta: `{"a":1}{"a":2}`
> Y esta es la entrada incorrecta: `[{"a":1}, {"a":2}]`

- Un esquema para determinar cómo interpretar esos datos: utilice AWS Glue para crear un esquema en AWS Glue Data Catalog. A continuación, Amazon Data Firehose hace referencia a ese esquema y lo utiliza para interpretar los datos de entrada. Puede usar el mismo esquema para configurar Amazon Data Firehose y su software de análisis. Para obtener más información, consulte Rellenar el catálogo de datos de AWS Glue en la Guía para AWS Glue desarrolladores.

  > **Note**
  >
  > El esquema creado en el catálogo AWS Glue de datos debe coincidir con la estructura de datos de entrada. De lo contrario, los datos convertidos no contendrán atributos que no estén especificados en el esquema. Si utiliza un JSON anidado, utilice un tipo STRUCT en el esquema que refleje la estructura de los datos JSON. Consulte este ejemplo para ver cómo gestionar un JSON anidado con un tipo STRUCT.

- Un serializador para convertir los datos al formato de almacenamiento en columnas de destino (Parquet u ORC) : puede elegir uno de los dos tipos de serializadores: ORC o Parquet. SerDe SerDe

> **Important**
>
> Si habilitas la conversión de formatos de registro, no podrás configurar el destino de Amazon Data Firehose como Amazon OpenSearch Service, Amazon Redshift o Splunk. Con la conversión de formato habilitada, Amazon S3 es el único destino que puede usar para su transmisión de Firehose.

Puede convertir el formato de sus datos incluso si agrega sus registros antes de enviarlos a Amazon Data Firehose.

Elección del deserializador JSON

Elija el JSON de OpenX SerDe si el JSON de entrada contiene marcas de tiempo en los siguientes formatos:

- aaaa-MM-dd'T'HH:mm:ss[.S]'Z', donde la fracción puede tener hasta 9 dígitos: por ejemplo, `2017-02-07T15:13:01.39256Z`.

- aaaa-[M]M-[d]d HH:mm:ss[.S], donde la fracción puede tener hasta 9 dígitos: por ejemplo, `2017-02-07 15:13:01.14`.
- Segundos epoch: por ejemplo, `1518033528`.
- Milisegundos epoch: por ejemplo, `1518033528123`.
- Segundos epoch con número de punto flotante: por ejemplo, `1518033528.123`.

El OpenX JSON SerDe puede convertir puntos (`.`) en guiones bajos (`_`). También puede convertir claves JSON a minúsculas antes de deserializarlas. Para obtener más información sobre las opciones disponibles con este deserializador a través de Amazon Data Firehose, consulte OpenX. JsonSerDe

Si no está seguro de qué deserializador elegir, utilice OpenX JSON SerDe, a menos que tenga marcas de tiempo que no admita.

Si tiene marcas de tiempo en formatos distintos a los enumerados anteriormente, utilice el JSON de Apache Hive. SerDe Si elige este deserializador, puede especificar los formatos de marca temporal que va a utilizar. Para ello, siga la sintaxis de los patrones de las cadenas de formato `DateTimeFormat` de Joda-Time. Para obtener más información, consulte Clase DateTimeFormat.

También puede utilizar el valor especial `millis` para analizar las marcas temporales en milisegundos con formato de tiempo Unix. Si no especificas ningún formato, Amazon Data Firehose lo utilizará de forma predeterminada`java.sql.Timestamp::valueOf`.

El JSON de Hive SerDe no permite lo siguiente:

- Puntos (`.`) en los nombres de las columnas.
- Campos cuyo tipo sea `uniontype`.
- Campos que tienen tipos numéricos en el esquema, pero que son cadenas en el JSON. Por ejemplo, si el esquema es (un int) y el JSON lo es`{"a":"123"}`, Hive SerDe muestra un error.

La colmena SerDe no convierte el JSON anidado en cadenas. Por ejemplo, si se tiene `{"a": {"inner":1}}`, no trata `{"inner":1}` como una cadena.

Elección del serializador

El serializador que elija depende de sus necesidades empresariales. Para obtener más información sobre las dos opciones del serializador, consulte ORC y Parquet. SerDe SerDe

Conversión del formato de registro de entrada (consola)

Puedes habilitar la conversión de formatos de datos en la consola al crear o actualizar una transmisión de Firehose. Con la conversión de formatos de datos habilitada, Amazon S3 es el único destino que puede configurar para la transmisión de Firehose. Además, la compresión de Amazon S3 se deshabilita al habilitar la conversión de formatos. Sin embargo, la compresión Snappy se realiza automáticamente como parte del proceso de conversión. El formato de encuadre para Snappy que Amazon Data Firehose utiliza en este caso es compatible con Hadoop. Esto significa que puede utilizar los resultados de la compresión de Snappy y ejecutar consultas con estos datos en Athena. Para ver el formato de encuadre Snappy en el que se basa Hadoop, consulte .java. BlockCompressorStream

Para habilitar la conversión de formato de datos para una transmisión Firehose de datos

1. Inicie sesión en la AWS Management Console consola Amazon Data Firehose y ábrala en https://console.aws.amazon.com/firehose/.
2. Elige una transmisión de Firehose para actualizarla o crea una nueva transmisión de Firehose siguiendo los pasos que se indican. Crea una transmisión de Firehose
3. En Convert record format (Convertir formato de registro), establezca Record format conversion (Conversión del formato de registro) en Enabled (Habilitado).
4. Elija el formato de salida que desea utilizar. Para obtener más información acerca de las dos opciones, consulte Apache Parquet y Apache ORC.
5. Elija una AWS Glue tabla para especificar un esquema para sus registros fuente. Establezca la región, la base de datos, la tabla y la versión de la tabla.

Conversión del formato de registro de entrada (API)

Si desea que Amazon Data Firehose convierta el formato de los datos de entrada de JSON a Parquet u ORC, especifique el DataFormatConversionConfigurationelemento opcional en ExtendedS3 o en ExtendedS3 DestinationConfiguration. DestinationUpdate Si lo especifica, se aplicarán las siguientes restricciones: DataFormatConversionConfiguration

- En BufferingHints, no puede establecer un valor inferior `SizeInMBs` a 64 si habilita la conversión al formato de registro. Además, si la conversión de formato no está habilitada, el valor predeterminado es 5. El valor pasa a ser 128 cuando se habilita.

- Debe establecer `CompressionFormat` en ExtendeDS3 DestinationConfiguration o en ExtendeDS3 para. DestinationUpdate `UNCOMPRESSED` El valor predeterminado de `CompressionFormat` es `UNCOMPRESSED`. Por lo tanto, también puede dejarlo sin especificar en ExtendedS3. DestinationConfiguration Los datos se siguen comprimiendo como parte del proceso de serialización utilizando la compresión Snappy de forma predeterminada. El formato de encuadre para Snappy que Amazon Data Firehose utiliza en este caso es compatible con Hadoop. Esto significa que puede utilizar los resultados de la compresión de Snappy y ejecutar consultas con estos datos en Athena. Para ver el formato de encuadre Snappy en el que se basa Hadoop, consulte .java. BlockCompressorStream Al configurar el serializador, puede elegir otros tipos de compresión.

Control de errores de la conversión del formato de registro

Cuando Amazon Data Firehose no puede analizar o deserializar un registro (por ejemplo, cuando los datos no coinciden con el esquema), lo escribe en Amazon S3 con un prefijo de error. Si se produce un error en la escritura, Amazon Data Firehose lo volverá a intentar para siempre, lo que bloqueará la entrega posterior. Para cada registro fallido, Amazon Data Firehose escribe un documento JSON con el siguiente esquema:

```
{
  "attemptsMade": long,
  "arrivalTimestamp": long,
  "lastErrorCode": string,
  "lastErrorMessage": string,
  "attemptEndingTimestamp": long,
  "rawData": string,
  "sequenceNumber": string,
  "subSequenceNumber": long,
  "dataCatalogTable": {
    "catalogId": string,
    "databaseName": string,
    "tableName": string,
    "region": string,
    "versionId": string,
    "catalogArn": string
  }
}
```

Ejemplo de conversión del formato de registros

Para ver un ejemplo de cómo configurar la conversión de formatos de registro con AWS CloudFormation, consulte AWS::DataFirehose: DeliveryStream.

Uso de Amazon Managed Service para Apache Flink

Con Amazon Managed Service para Apache Flink, usted puede usar Java, Scala o SQL para procesar y analizar datos de streaming. El servicio le permite crear y ejecutar código en orígenes de streaming para realizar análisis de series temporales, alimentar paneles en tiempo real y crear métricas en tiempo real.

Para ver un ejemplo de integración con Amazon Managed Service para Apache Flink, consulte Example: Writing to Amazon Data Firehose.

En este ejercicio, creará una aplicación Apache Flink que tenga una transmisión de datos de Kinesis como fuente y una transmisión Firehose como sumidero. Con el receptor, puede verificar la salida de la aplicación en un bucket de Amazon S3.

Antes de empezar, configure los requisitos previos necesarios:

- Componentes del servicio gestionado para la aplicación Apache Flink
- Requisitos previos para completar el ejercicio

Conozca la entrega de datos de Amazon Data Firehose

Una vez que los datos se envían a tu transmisión Firehose, se envían automáticamente al destino que elijas.

> **Important**
>
> Si utiliza Kinesis Producer Library (KPL) para escribir datos en un flujo de datos de Kinesis, puede utilizar la agregación para combinar los registros que escriba en ese flujo de datos de Kinesis. Si, a continuación, utiliza esa transmisión de datos como fuente para la transmisión de Firehose, Amazon Data Firehose desagrega los registros antes de entregarlos al destino. Si configura la transmisión de Firehose para transformar los datos, Amazon Data Firehose desagrega los registros antes de entregarlos a ellos. AWS Lambda Para obtener más información, consulte Developing Amazon Kinesis Data Streams Producers Using the Kinesis Producer Library y Aggregation en la Guía para desarrolladores de Amazon Kinesis Data Streams.

Temas

- Configure el formato de entrega de datos
- Comprenda la frecuencia de entrega de datos
- Gestione los errores en la entrega de datos
- Configurar el formato de nombre de objeto de Amazon S3
- Configura la rotación del índice para el servicio OpenSearch
- Comprenda la entrega en todas las AWS cuentas y regiones
- Registros duplicados
- Pausa y reanuda una transmisión de Firehose

Configure el formato de entrega de datos

Para el envío de datos a Amazon Simple Storage Service (Amazon S3), Firehose concatena varios registros entrantes en función de la configuración de almacenamiento en búfer de la transmisión de Firehose. A continuación, entrega los registros en Amazon S3 como un objeto de Amazon S3.

De forma predeterminada, Firehose concatena los datos sin ningún delimitador. Si desea disponer de nuevos delimitadores de línea entre los registros, puede añadir nuevos delimitadores de línea activando la función en la configuración de la consola Firehose o en el parámetro de API.

Para el envío de datos a Amazon Redshift, Firehose entrega primero los datos entrantes al bucket de S3 en el formato descrito anteriormente. A continuación, Firehose emite un comando de Amazon COPY Redshift para cargar los datos del bucket de S3 en el clúster aprovisionado de Amazon Redshift o en el grupo de trabajo Amazon Redshift Serverless. Asegúrese de que, después de que Amazon Data Firehose haya concatenado varios registros entrantes a un objeto de Amazon S3, el objeto de Amazon S3 se pueda copiar en el clúster aprovisionado de Amazon Redshift o en el grupo de trabajo Amazon Redshift Serverless. Para obtener más información, consulte Amazon Redshift COPY Command Data Format Parameters.

Para la entrega de datos a OpenSearch Service y OpenSearch Serverless, Amazon Data Firehose almacena en búfer los registros entrantes en función de la configuración de almacenamiento en búfer de la transmisión de Firehose. A continuación, genera una solicitud masiva de OpenSearch Service o OpenSearch Serverless para indexar varios registros en su clúster de servicios o en su colección Serverless. OpenSearch OpenSearch Asegúrese de que el registro esté codificado en UTF-8 y aplanado en un objeto JSON de una sola línea antes de enviarlo a Amazon Data Firehose. Además, la `rest.action.multi.allow_explicit_index` opción de su clúster de OpenSearch servicios debe estar establecida en true (valor predeterminado) para recibir solicitudes masivas con un índice explícito establecido por registro. Para obtener más información, consulta las opciones avanzadas de configuración de OpenSearch servicios en la Guía para desarrolladores de Amazon OpenSearch Service.

Para la entrega de datos a Splunk, Amazon Data Firehose concatena los bytes que usted envía. Si desea delimitadores en los datos, como, por ejemplo, un carácter de nueva línea, debe insertarlos usted mismo. Asegúrese de que Splunk esté configurado para analizar dichos delimitadores.

Al entregar datos en un punto de conexión HTTP que pertenece a un proveedor de servicios de terceros admitido, puede usar el servicio Amazon Lambda integrado para crear una función que transforme los registros de entrada en el formato que coincida con el formato que espera la integración del proveedor de servicios. Póngase en contacto con el proveedor de servicios de terceros cuyo punto de conexión HTTP haya elegido como destino para obtener más información sobre el formato de registro aceptado.

Para la entrega de datos a Snowflake, Amazon Data Firehose almacena internamente los datos durante un segundo y utiliza las operaciones de la API de streaming de Snowflake para insertar

datos en Snowflake. De forma predeterminada, los registros que se insertan se vacían y se archivan en la tabla de Snowflake cada segundo. Tras realizar la llamada de inserción, Firehose emite una CloudWatch métrica que mide el tiempo que tardaron los datos en enviarse a Snowflake. Firehose actualmente solo admite un elemento JSON como carga útil de registro y no admite matrices JSON. Asegúrate de que la carga útil de entrada sea un objeto JSON válido y esté bien formada sin comillas dobles, comillas ni caracteres de escape adicionales.

Comprenda la frecuencia de entrega de datos

Cada destino de Firehose tiene su propia frecuencia de entrega de datos. Para obtener más información, consulte Comprenda las sugerencias de almacenamiento en búfer.

Gestione los errores en la entrega de datos

Cada destino de Amazon Data Firehose tiene su propia gestión de errores en la entrega de datos.

Amazon S3

La entrega de datos al bucket de S3 podría generar errores por varias razones. Por ejemplo, es posible que el bucket ya no exista, que la función de IAM que Amazon Data Firehose asume no tenga acceso al bucket, que la red haya fallado o se produzcan eventos similares. En estas condiciones, Amazon Data Firehose volverá a intentarlo durante un máximo de 24 horas hasta que la entrega se realice correctamente. El tiempo máximo de almacenamiento de datos de Amazon Data Firehose es de 24 horas. Si, una vez transcurridas esas 24 horas, no se pueden entregar los datos, estos se pierden.

Amazon Redshift

Para un destino de Amazon Redshift, puede especificar una duración de reintento (de 0 a 7200 segundos) al crear una transmisión de Firehose.

La entrega de datos en su clúster aprovisionado de Amazon Redshift o grupo de trabajo de Amazon Redshift sin servidor puede fallar por varios motivos. Por ejemplo, es posible que tengas una configuración de clúster incorrecta en tu transmisión de Firehose, un clúster o grupo de trabajo en mantenimiento o un fallo de red. En estas condiciones, Amazon Data Firehose lo vuelve a intentar durante el tiempo especificado y omite ese lote concreto de objetos de Amazon S3. La información de los objetos ignorados se entrega al bucket de S3 en forma de archivo de manifiesto, en la carpeta `errors/`, que puede utilizar para reposiciones manuales. Para obtener

más información acerca de cómo usar el comando COPY para copiar datos manualmente con archivos de manifiesto, consulte Uso de un manifiesto para especificar archivos de datos.

Amazon OpenSearch Service y OpenSearch Serverless

Para el destino OpenSearch Service y OpenSearch Serverless, puedes especificar una duración de reintento (de 0 a 7200 segundos) durante la creación de la transmisión de Firehose.

La entrega de datos al clúster de OpenSearch servicios o a la recopilación OpenSearch sin servidor puede fallar por varios motivos. Por ejemplo, es posible que tengas una configuración incorrecta de un clúster de OpenSearch servicio o colección OpenSearch sin servidor de tu transmisión de Firehose, OpenSearch un clúster de servicio OpenSearch o una colección sin servidor en mantenimiento, un fallo de red o eventos similares. En estas condiciones, Amazon Data Firehose lo vuelve a intentar durante el tiempo especificado y, a continuación, omite esa solicitud de índice concreta. Los documentos ignorados se entregan al bucket de S3 en forma de archivo de manifiesto, en la carpeta `AmazonOpenSearchService_failed/`, que puede utilizar para reposiciones manuales.

En el OpenSearch caso de Service, cada documento tiene el siguiente formato JSON:

```
{
    "attemptsMade": "(number of index requests attempted)",
    "arrivalTimestamp": "(the time when the document was received by Firehose)",
    "errorCode": "(http error code returned by OpenSearch Service)",
    "errorMessage": "(error message returned by OpenSearch Service)",
    "attemptEndingTimestamp": "(the time when Firehose stopped attempting index
 request)",
    "esDocumentId": "(intended OpenSearch Service document ID)",
    "esIndexName": "(intended OpenSearch Service index name)",
    "esTypeName": "(intended OpenSearch Service type name)",
    "rawData": "(base64-encoded document data)"
}
```

En el OpenSearch caso de Serverless, cada documento tiene el siguiente formato JSON:

```
{
    "attemptsMade": "(number of index requests attempted)",
    "arrivalTimestamp": "(the time when the document was received by Firehose)",
    "errorCode": "(http error code returned by OpenSearch Serverless)",
    "errorMessage": "(error message returned by OpenSearch Serverless)",
```

```
    "attemptEndingTimestamp": "(the time when Firehose stopped attempting index
 request)",
    "osDocumentId": "(intended OpenSearch Serverless document ID)",
    "osIndexName": "(intended OpenSearch Serverless index name)",
    "rawData": "(base64-encoded document data)"
}
```

Splunk

Cuando Amazon Data Firehose envía datos a Splunk, espera el acuse de recibo de Splunk. Si se produce un error o el acuse de recibo no llega dentro del tiempo de espera del acuse de recibo, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Amazon Data Firehose envía datos a Splunk, ya sea en el intento inicial o en un reintento, reinicia el contador de tiempo de espera de confirmación. A continuación, espera a que llegue una confirmación desde Splunk. Incluso si la duración del reintento vence, Amazon Data Firehose seguirá esperando el acuse de recibo hasta que lo reciba o se agote el tiempo de espera del acuse de recibo. Si se agota el tiempo de espera de la confirmación, Amazon Data Firehose comprueba si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una confirmación o determina que el tiempo de reintento se ha agotado.

Un error de recepción de una confirmación no es el único tipo de error de entrega de datos que puede producirse. Para obtener información sobre los demás tipos de errores de entrega de datos, consulte Errores de entrega de datos relacionados con Splunk. Cualquier error de entrega de datos desencadenará la lógica de reintentos si el tiempo de reintento es mayor que 0.

A continuación se muestra un ejemplo de registro de error.

```
{
  "attemptsMade": 0,
  "arrivalTimestamp": 1506035354675,
  "errorCode": "Splunk.AckTimeout",
  "errorMessage": "Did not receive an acknowledgement from HEC before the HEC
 acknowledgement timeout expired. Despite the acknowledgement timeout, it's possible
 the data was indexed successfully in Splunk. Amazon Data Firehose backs up in
 Amazon S3 data for which the acknowledgement timeout expired.",
```

```
  "attemptEndingTimestamp": 13626284715507,
  "rawData":
 "MiAyNTE2MjAyNzIyMDkgZW5pLTA1ZjMyMmQ1IDIxOC45Mi4xODguMjE0IDE3Mi4xNi4xLjE2NyAyNTIzMyAxNDMzID
  "EventId": "495771939281141473396007784710824923931641398772000035842.0"
}
```

Destino de punto de conexión HTTP

Cuando Amazon Data Firehose envía datos a un destino de punto final HTTP, espera una respuesta de este destino. Si se produce un error o la respuesta no llega dentro del tiempo de espera de la respuesta, Amazon Data Firehose inicia el contador de duración de los reintentos. Continúa intentándolo hasta que se agota el tiempo de reintento. Después de eso, Amazon Data Firehose considera que se trata de un error en la entrega de datos y realiza una copia de seguridad de los datos en su bucket de Amazon S3.

Cada vez que Amazon Data Firehose envía datos a un destino de punto final HTTP, ya sea el intento inicial o un reintento, reinicia el contador de tiempo de espera de respuesta. A continuación, espera a que llegue una respuesta del destino de punto de conexión HTTP. Incluso si vence la duración del reintento, Amazon Data Firehose seguirá esperando la respuesta hasta que la reciba o se agote el tiempo de espera de la respuesta. Si se agota el tiempo de espera de la respuesta, Amazon Data Firehose comprueba si queda tiempo en el contador de reintentos. Si queda tiempo, vuelve a intentarlo y repite la lógica hasta que recibe una respuesta o determina que el tiempo de reintento se ha agotado.

Un error de recepción de una respuesta no es el único tipo de error de entrega de datos que puede producirse. Para obtener información sobre los demás tipos de errores de entrega de datos, consulte HTTP Endpoint Data Delivery Errors.

A continuación se muestra un ejemplo de registro de error.

```
{
 "attemptsMade":5,
 "arrivalTimestamp":1594265943615,
 "errorCode":"HttpEndpoint.DestinationException",
 "errorMessage":"Received the following response from the endpoint destination.
 {"requestId": "109777ac-8f9b-4082-8e8d-b4f12b5fc17b", "timestamp": 1594266081268,
 "errorMessage": "Unauthorized"}",
 "attemptEndingTimestamp":1594266081318,
 "rawData":"c2FtcGxlIHJhdyBkYXRh",
 "subsequenceNumber":0,
 "dataId":"49607357361271740811418664280693044274821622880012337186.0"
```

```
}
```

Destino en forma de copo de nieve

Para el destino Snowflake, al crear una transmisión de Firehose, puedes especificar una duración de reintento opcional (de 0 a 7200 segundos). El valor predeterminado para la duración del reintento es de 60 segundos.

La entrega de datos a la tabla de Snowflake puede fallar por varios motivos, como una configuración de destino incorrecta de Snowflake, un fallo de red, etc. La política de reintentos no se aplica a los errores no recuperables. Por ejemplo, si Snowflake rechaza tu carga JSON porque hay una columna adicional que falta en la tabla, Firehose no intentará volver a entregarla. En su lugar, crea una copia de seguridad de todos los errores de inserción debidos a problemas de carga de JSON en el depósito de errores de S3.

Del mismo modo, si se produce un error en la entrega debido a un rol, tabla o base de datos incorrectos, Firehose no lo vuelve a intentar y escribe los datos en el bucket de S3. La duración del reintento solo se aplica a los errores debidos a un problema con el servicio de Snowflake, a fallos transitorios de la red, etc. En estas condiciones, Firehose lo vuelve a intentar durante el tiempo especificado antes de entregarlos a S3. Los registros fallidos se entregan en la carpeta snowflake-failed/, que puede utilizar para rellenarlos manualmente.

El siguiente es un ejemplo de JSON para cada registro que entregue a S3.

```
{
    "attemptsMade": 3,
    "arrivalTimestamp": 1594265943615,
    "errorCode": "Snowflake.InvalidColumns",
    "errorMessage": "Snowpipe Streaming does not support columns of type
 AUTOINCREMENT, IDENTITY, GEO, or columns with a default value or collation",
    "attemptEndingTimestamp": 1712937865543,
    "rawData": "c2FtcGxlIHJhdyBkYXRh"
}
```

Configurar el formato de nombre de objeto de Amazon S3

Cuando Firehose entrega datos a Amazon S3, el nombre de la clave del objeto S3 sigue el formato <evaluated prefix><suffix>, donde el sufijo tiene el formato: - - - - - - - - - <Firehose stream name><Firehose stream version><year><month><day><hour><minute><second><uuid><file

extension><Firehose stream version>comienza por 1 y aumenta en 1 por cada cambio de configuración de Firehose Stream. Puedes cambiar las configuraciones de transmisión de Firehose (por ejemplo, el nombre del bucket de S3, las sugerencias de almacenamiento en búfer, la compresión y el cifrado). Puedes hacerlo mediante la consola Firehose o la operación UpdateDestinationAPI.

Para<evaluated prefix>, Firehose añade un prefijo de hora predeterminado en el formato. `YYYY/MM/dd/HH` Este prefijo crea una jerarquía lógica en el depósito, en la que cada barra inclinada (/) crea un nivel en la jerarquía. Puede modificar esta estructura especificando un prefijo personalizado que incluya expresiones que se evalúan en tiempo de ejecución. Para obtener información sobre cómo especificar un prefijo personalizado, consulte Prefijos personalizados para objetos de Amazon Simple Storage Service.

De forma predeterminada, la zona horaria utilizada como prefijo y sufijo es UTC, pero puede cambiarla por la zona horaria que prefiera. Por ejemplo, para usar la hora estándar de Japón en lugar de UTC, puede configurar la zona horaria en Asia/Tokio en la configuración de parámetros de la AWS Management Console API (). CustomTimeZone La siguiente lista contiene las zonas horarias que Firehose admite para la configuración del prefijo S3.

Zonas horarias

A continuación se muestra una lista de las zonas horarias que Firehose admite para la configuración del prefijo S3.

Africa

```
Africa/Abidjan
Africa/Accra
Africa/Addis_Ababa
Africa/Algiers
Africa/Asmera
Africa/Bangui
Africa/Banjul
Africa/Bissau
Africa/Blantyre
Africa/Bujumbura
Africa/Cairo
Africa/Casablanca
Africa/Conakry
Africa/Dakar
Africa/Dar_es_Salaam
```

```
Africa/Djibouti
Africa/Douala
Africa/Freetown
Africa/Gaborone
Africa/Harare
Africa/Johannesburg
Africa/Kampala
Africa/Khartoum
Africa/Kigali
Africa/Kinshasa
Africa/Lagos
Africa/Libreville
Africa/Lome
Africa/Luanda
Africa/Lubumbashi
Africa/Lusaka
Africa/Malabo
Africa/Maputo
Africa/Maseru
Africa/Mbabane
Africa/Mogadishu
Africa/Monrovia
Africa/Nairobi
Africa/Ndjamena
Africa/Niamey
Africa/Nouakchott
Africa/Ouagadougou
Africa/Porto-Novo
Africa/Sao_Tome
Africa/Timbuktu
Africa/Tripoli
Africa/Tunis
Africa/Windhoek
```

America

```
America/Adak
America/Anchorage
America/Anguilla
America/Antigua
America/Aruba
America/Asuncion
America/Barbados
```

```
America/Belize
America/Bogota
America/Buenos_Aires
America/Caracas
America/Cayenne
America/Cayman
America/Chicago
America/Costa_Rica
America/Cuiaba
America/Curacao
America/Dawson_Creek
America/Denver
America/Dominica
America/Edmonton
America/El_Salvador
America/Fortaleza
America/Godthab
America/Grand_Turk
America/Grenada
America/Guadeloupe
America/Guatemala
America/Guayaquil
America/Guyana
America/Halifax
America/Havana
America/Indianapolis
America/Jamaica
America/La_Paz
America/Lima
America/Los_Angeles
America/Managua
America/Manaus
America/Martinique
America/Mazatlan
America/Mexico_City
America/Miquelon
America/Montevideo
America/Montreal
America/Montserrat
America/Nassau
America/New_York
America/Noronha
America/Panama
America/Paramaribo
```

```
America/Phoenix
America/Port_of_Spain
America/Port-au-Prince
America/Porto_Acre
America/Puerto_Rico
America/Regina
America/Rio_Branco
America/Santiago
America/Santo_Domingo
America/Sao_Paulo
America/Scoresbysund
America/St_Johns
America/St_Kitts
America/St_Lucia
America/St_Thomas
America/St_Vincent
America/Tegucigalpa
America/Thule
America/Tijuana
America/Tortola
America/Vancouver
America/Winnipeg
```

Antarctica

```
Antarctica/Casey
Antarctica/DumontDUrville
Antarctica/Mawson
Antarctica/McMurdo
Antarctica/Palmer
```

Asia

```
Asia/Aden
Asia/Almaty
Asia/Amman
Asia/Anadyr
Asia/Aqtau
Asia/Aqtobe
Asia/Ashgabat
Asia/Ashkhabad
Asia/Baghdad
Asia/Bahrain
```

```
Asia/Baku
Asia/Bangkok
Asia/Beirut
Asia/Bishkek
Asia/Brunei
Asia/Calcutta
Asia/Colombo
Asia/Dacca
Asia/Damascus
Asia/Dhaka
Asia/Dubai
Asia/Dushanbe
Asia/Hong_Kong
Asia/Irkutsk
Asia/Jakarta
Asia/Jayapura
Asia/Jerusalem
Asia/Kabul
Asia/Kamchatka
Asia/Karachi
Asia/Katmandu
Asia/Krasnoyarsk
Asia/Kuala_Lumpur
Asia/Kuwait
Asia/Macao
Asia/Magadan
Asia/Manila
Asia/Muscat
Asia/Nicosia
Asia/Novosibirsk
Asia/Phnom_Penh
Asia/Pyongyang
Asia/Qatar
Asia/Rangoon
Asia/Riyadh
Asia/Saigon
Asia/Seoul
Asia/Shanghai
Asia/Singapore
Asia/Taipei
Asia/Tashkent
Asia/Tbilisi
Asia/Tehran
Asia/Thimbu
```

```
Asia/Thimphu
Asia/Tokyo
Asia/Ujung_Pandang
Asia/Ulaanbaatar
Asia/Ulan_Bator
Asia/Vientiane
Asia/Vladivostok
Asia/Yakutsk
Asia/Yekaterinburg
Asia/Yerevan
```

Atlantic

```
Atlantic/Azores
Atlantic/Bermuda
Atlantic/Canary
Atlantic/Cape_Verde
Atlantic/Faeroe
Atlantic/Jan_Mayen
Atlantic/Reykjavik
Atlantic/South_Georgia
Atlantic/St_Helena
Atlantic/Stanley
```

Australia

```
Australia/Adelaide
Australia/Brisbane
Australia/Broken_Hill
Australia/Darwin
Australia/Hobart
Australia/Lord_Howe
Australia/Perth
Australia/Sydney
```

Europe

```
Europe/Amsterdam
Europe/Andorra
Europe/Athens
Europe/Belgrade
Europe/Berlin
```

```
Europe/Brussels
Europe/Bucharest
Europe/Budapest
Europe/Chisinau
Europe/Copenhagen
Europe/Dublin
Europe/Gibraltar
Europe/Helsinki
Europe/Istanbul
Europe/Kaliningrad
Europe/Kiev
Europe/Lisbon
Europe/London
Europe/Luxembourg
Europe/Madrid
Europe/Malta
Europe/Minsk
Europe/Monaco
Europe/Moscow
Europe/Oslo
Europe/Paris
Europe/Prague
Europe/Riga
Europe/Rome
Europe/Samara
Europe/Simferopol
Europe/Sofia
Europe/Stockholm
Europe/Tallinn
Europe/Tirane
Europe/Vaduz
Europe/Vienna
Europe/Vilnius
Europe/Warsaw
Europe/Zurich
```

Indian

```
Indian/Antananarivo
Indian/Chagos
Indian/Christmas
Indian/Cocos
Indian/Comoro
```

```
Indian/Kerguelen
Indian/Mahe
Indian/Maldives
Indian/Mauritius
Indian/Mayotte
Indian/Reunion
```

Pacific

```
Pacific/Apia
Pacific/Auckland
Pacific/Chatham
Pacific/Easter
Pacific/Efate
Pacific/Enderbury
Pacific/Fakaofo
Pacific/Fiji
Pacific/Funafuti
Pacific/Galapagos
Pacific/Gambier
Pacific/Guadalcanal
Pacific/Guam
Pacific/Honolulu
Pacific/Kiritimati
Pacific/Kosrae
Pacific/Majuro
Pacific/Marquesas
Pacific/Nauru
Pacific/Niue
Pacific/Norfolk
Pacific/Noumea
Pacific/Pago_Pago
Pacific/Palau
Pacific/Pitcairn
Pacific/Ponape
Pacific/Port_Moresby
Pacific/Rarotonga
Pacific/Saipan
Pacific/Tahiti
Pacific/Tarawa
Pacific/Tongatapu
Pacific/Truk
Pacific/Wake
```

```
Pacific/Wallis
```

<file extension>No puede cambiar el campo de sufijo excepto. Al habilitar la conversión o compresión de formatos de datos, Firehose añadirá una extensión de archivo en función de la configuración. En la siguiente tabla se explica la extensión de archivo predeterminada adjunta por Firehose:

Configuración	Extensión de archivo
Conversión de formato de datos: Parquet	.parquet
Conversión de formato de datos: ORC	.orc
Compresión: Gzip	.gz
Compresión: Zip	.zip
Compresión: rápida	.snappy
Compresión: Hadoop-Snappy	.hsnappy

También puedes especificar la extensión de archivo que prefieras en la consola o la API de Firehose. La extensión del archivo debe empezar con un punto (.) y puede contener los caracteres permitidos: 0-9a-z! -_.*' (). La extensión del archivo no puede superar los 128 caracteres.

> **Note**
>
> Al especificar una extensión de archivo, esta anulará la extensión de archivo predeterminada que Firehose añada cuando esté habilitada la conversión o compresión de formatos de datos.

Configura la rotación del índice para el servicio OpenSearch

Para el destino del OpenSearch servicio, puede especificar una opción de rotación de índices basada en el tiempo entre una de las cinco opciones siguientes:NoRotation,OneHour, OneDayOneWeek, oOneMonth.

Según la opción de rotación que elija, Amazon Data Firehose añade una parte de la marca horaria de llegada UTC al nombre de índice especificado. También rota la marca de tiempo añadida en consecuencia. El siguiente ejemplo muestra el nombre del índice resultante en OpenSearch Service para cada opción de rotación del índice, donde es el nombre del índice especificado myindex y la marca de tiempo de llegada. `2016-02-25T13:00:00Z`

RotationPeriod	IndexName
`NoRotation`	`myindex`
`OneHour`	`myindex-2016-02-25-13`
`OneDay`	`myindex-2016-02-25`
`OneWeek`	`myindex-2016-w08`
`OneMonth`	`myindex-2016-02`

> **Note**
>
> Con la opción `OneWeek`, Data Firehose crea índices automáticamente con el formato <AÑO>-w<NÚMERO DE LA SEMANA> (por ejemplo, `2020-w33`), donde el número de la semana se calcula según la hora UTC y las siguientes convenciones de EE. UU.:
>
> - Una semana comienza el domingo
> - La primera semana del año es la primera semana que contiene un sábado de este año

Comprenda la entrega en todas las AWS cuentas y regiones

Amazon Data Firehose admite la entrega de datos a destinos de punto final HTTP en todas las AWS cuentas. La transmisión Firehose y el punto final HTTP que elijas como destino pueden pertenecer a cuentas diferentes AWS .

Amazon Data Firehose también admite la entrega de datos a destinos de punto final HTTP en AWS todas las regiones. Puede entregar datos desde una transmisión de Firehose en una AWS región a un punto final HTTP en otra AWS región. También puede entregar datos desde una transmisión Firehose a un destino de punto final HTTP fuera de AWS las regiones, por ejemplo, a su propio servidor local, configurando la URL del punto de enlace HTTP en el destino deseado. En estos casos, se agregan cargos de transferencia de datos adicionales a los gastos de entrega. Para obtener más información, consulte la sección Transferencia de datos de la página “Precios de las instancias bajo demanda”.

Registros duplicados

Amazon Data Firehose utiliza la at-least-once semántica para la entrega de datos. En algunas circunstancias, como cuando se agota el tiempo de espera para la entrega de datos, los reintentos de entrega por parte de Amazon Data Firehose podrían introducir duplicados si finalmente se aprueba la solicitud de entrega de datos original. Esto se aplica a todos los tipos de destinos compatibles con Amazon Data Firehose.

Pausa y reanuda una transmisión de Firehose

Después de configurar una transmisión Firehose, los datos disponibles en la fuente de transmisión se envían continuamente al destino. Si se produce alguna situación en la que el destino del flujo no esté disponible temporalmente (por ejemplo, durante las operaciones de mantenimiento planificadas), puede que desee pausar temporalmente la entrega de datos y reanudarla cuando el destino vuelva a estar disponible. En las siguientes secciones se muestra cómo hacerlo:

> **Important**
>
> Si utiliza el enfoque que se describe a continuación para pausar y reanudar una transmisión, después de reanudarla, verá que se envían pocos registros al depósito de errores de Amazon S3, mientras que el resto de la transmisión continúa enviándose al destino. Esta

es una limitación conocida del enfoque y se debe a que se registra como fallido un número reducido de registros que antes no se podían entregar al destino tras varios reintentos.

Entender cómo Firehose gestiona los fallos de entrega

Cuando configuras una transmisión Firehose, para muchos destinos, como OpenSearch Splunk y puntos de conexión HTTP, también configuras un bucket de S3 en el que se pueden hacer copias de seguridad de los datos que no se puedan entregar. Para obtener más información sobre cómo Firehose hace copias de seguridad de los datos en caso de entregas fallidas, consulte Gestión de errores en la entrega de datos. Para obtener más información sobre cómo conceder acceso a los buckets de S3 donde se pueden hacer copias de seguridad de los datos que no se puedan entregar, consulte Otorgar a Firehose Access to an Amazon S3 Destination. Cuando Firehose (a) no entrega los datos al destino de la transmisión y (b) no escribe los datos en el depósito S3 de respaldo en caso de entregas fallidas, detiene la entrega de la transmisión hasta que los datos puedan entregarse al destino o escribirse en la ubicación de S3 de respaldo.

Pausar una transmisión de Firehose

Para pausar la entrega de transmisiones en Firehose, primero elimine los permisos para que Firehose escriba en la ubicación de respaldo de S3 en caso de entregas fallidas. Por ejemplo, si quieres pausar la transmisión de Firehose con un OpenSearch destino, puedes hacerlo actualizando los permisos. Para obtener más información, consulte Otorgar a Firehose acceso a un destino de OpenSearch servicio público.

Elimine el permiso `"Effect": "Allow"` de la acción `s3:PutObject` y agregue de forma explícita una declaración que aplique el permiso `Effect": "Deny"` en la acción `s3:PutObject` para el bucket de S3 que se utiliza para hacer copias de seguridad de las entregas con errores. A continuación, desactiva el destino de la transmisión (por ejemplo, desactiva el OpenSearch dominio de destino) o quita los permisos para que Firehose escriba en el destino. Para actualizar los permisos de otros destinos, consulte la sección correspondiente a su destino en Controlling Access with Amazon Data Firehose. Tras completar estas dos acciones, Firehose dejará de emitir transmisiones y podrás monitorizarlo mediante CloudWatch las métricas de Firehose.

> **Important**
>
> Al pausar la entrega de transmisiones en Firehose, debe asegurarse de que la fuente de la transmisión (por ejemplo, en Kinesis Data Streams o en Managed Service for Kafka) esté

configurada para conservar los datos hasta que se reanude la entrega de la transmisión y los datos se entreguen al destino. Si la fuente es DirectPut, Firehose conservará los datos durante 24 horas. Se pueden producir pérdidas de datos si no se reanuda el flujo y no se entregan los datos antes de que venza el periodo de retención de datos.

Reanudación de una transmisión de Firehose

Para reanudar la entrega, primero revierta el cambio realizado anteriormente al destino de la transmisión activando el destino y asegurándose de que Firehose tenga los permisos para entregar la transmisión al destino. A continuación, revierta los cambios llevados a cabo anteriormente en los permisos aplicados al bucket de S3 para hacer copias de seguridad de las entregas con errores. Es decir, aplique el permiso `"Effect": "Allow"` a la acción `s3:PutObject` y elimine el permiso `"Effect": "Deny"` de la acción `s3:PutObject` para el bucket de S3 que se utiliza para hacer copias de seguridad de las entregas con errores. Por último, monitorea el uso de CloudWatch métricas de Firehose para confirmar que la transmisión se entrega al destino. Para ver y solucionar los errores, utiliza la supervisión de Amazon CloudWatch Logs para Firehose.

Supervisión de Amazon Data Firehose

Puede monitorizar Amazon Data Firehose mediante las siguientes funciones:

Temas

- Prácticas recomendadas con alarmas de CloudWatch
- Supervisión de Amazon Data Firehose mediante métricas CloudWatch
- Acceso a CloudWatch las métricas de Amazon Data Firehose
- Supervisión de Amazon Data Firehose mediante registros CloudWatch
- Acceso a CloudWatch los registros de Amazon Data Firehose
- Supervisión del estado del agente de Kinesis
- Registro de llamadas a la API Firehose de Amazon Data con AWS CloudTrail

Prácticas recomendadas con alarmas de CloudWatch

Añada CloudWatch alarmas cuando las siguientes métricas superen el límite de almacenamiento en búfer (un máximo de 15 minutos):

- `DeliveryToS3.DataFreshness`
- `DeliveryToSplunk.DataFreshness`
- `DeliveryToAmazonOpenSearchService.DataFreshness`
- `DeliveryToAmazonOpenSearchServerless.DataFreshness`
- `DeliveryToHttpEndpoint.DataFreshness`

Además, cree alarmas basadas en las siguientes expresiones matemáticas métricas.

- `IncomingBytes (Sum per 5 Minutes) / 300` se acerca a un porcentaje de `BytesPerSecondLimit`.
- `IncomingRecords (Sum per 5 Minutes) / 300` se acerca a un porcentaje de `RecordsPerSecondLimit`.
- `IncomingPutRequests (Sum per 5 Minutes) / 300` se acerca a un porcentaje de `PutRequestsPerSecondLimit`.

Otra métrica para la que recomendamos una alarma es `ThrottledRecords`.

Para obtener más información sobre cómo solucionar problemas cuando las alarmas están en estado `ALARM`, consulte Resolución de problemas.

Supervisión de Amazon Data Firehose mediante métricas CloudWatch

> **Important**
>
> Asegúrese de activar las alarmas en todas CloudWatch las métricas que pertenezcan a su destino para identificar los errores a tiempo.

Amazon Data Firehose se integra con CloudWatch las métricas de Amazon para que pueda recopilar, ver y analizar CloudWatch las métricas de sus transmisiones de Firehose. Por ejemplo, puede supervisar las `IncomingRecords` métricas `IncomingBytes` y realizar un seguimiento de los datos que los productores de datos ingieren en Amazon Data Firehose.

Amazon Data Firehose recopila y publica CloudWatch métricas cada minuto. Sin embargo, si las ráfagas de datos de entrada se producen solo durante unos segundos, es posible que no se capturen por completo ni sean visibles en las métricas de un minuto. Esto se debe a que CloudWatch las métricas se agregan desde Amazon Data Firehose en intervalos de un minuto.

Las métricas recopiladas para las transmisiones de Firehose son gratuitas. Para obtener información acerca de las métricas de agente de Kinesis, consulte Supervisión del estado del agente de Kinesis.

Temas

- Métricas de particionamiento CloudWatch dinámico
- CloudWatch Métricas de entrega de datos
- Métricas de adquisición de datos
- Métricas a nivel de API CloudWatch
- CloudWatch Métricas de transformación de datos
- CloudWatch Registra las métricas de descompresión
- Métricas de conversión de CloudWatch formato
- Métricas de cifrado del lado del servidor (SSE) CloudWatch

- Dimensiones de Amazon Data Firehose
- Métricas de uso de Amazon Data Firehose

Métricas de particionamiento CloudWatch dinámico

Si la partición dinámica está habilitada, el espacio de nombres AWS/Firehose incluye las siguientes métricas.

Métrica	Descripción
`ActivePartitionsLimit`	El número máximo de particiones activas que procesa una transmisión de Firehose antes de enviar datos al depósito de errores. Unidades: recuento
`PartitionCount`	Número de particiones que se procesan, es decir, el recuento de particiones activas. Este número varía entre 1 y el límite del recuento de particiones de 500 (valor predeterminado). Unidades: recuento
`PartitionCountExceeded`	Esta métrica indica si supera el límite del recuento de particiones. Emite 1 o 0 en función de si se supera el límite o no.
`JQProcessing.Duration`	Devuelve el tiempo que se tardó en ejecutar la expresión JQ en la función de Lambda JQ. Unidades: milisegundos
`PerPartitionThroughput`	Indica el rendimiento que se procesa por partición. Esta métrica le permite supervisar el rendimiento por partición. Unidades: StandardUnit. BytesSecond
`DeliveryToS3.ObjectCount`	Indica la cantidad de objetos que se van a entregar en su bucket de S3.

Métrica	Descripción
	Unidades: recuento

CloudWatch Métricas de entrega de datos

El espacio de nombres `AWS/Firehose` incluye las siguientes métricas de nivel de servicio. Si observa pequeñas caídas en el promedio de `BackupToS3.Success`, `DeliveryToS3.Success`, `DeliveryToSplunk.Success`, `DeliveryToAmazonOpenSearchService.Success` o `DeliveryToRedshift.Success`, eso no indica que se estén perdiendo datos. Amazon Data Firehose vuelve a intentar entregar los errores y no avanza hasta que los registros se entreguen correctamente al destino configurado o al bucket S3 de respaldo.

Temas

- Entrega al servicio OpenSearch
- Entrega a Serverless OpenSearch
- Entrega en Amazon Redshift
- Entrega en Amazon S3
- Entrega a Snowflake
- Entrega a Splunk
- Entrega en puntos de conexión HTTP

Entrega al servicio OpenSearch

Métrica	Descripción
`DeliveryToAmazonOpenSearchService.Bytes`	El número de bytes indexados al OpenSearch Servicio durante el período de tiempo especificado. Unidades: bytes
`DeliveryToAmazonOpenSearchService.DataFreshness`	La antigüedad (desde que se utilizó Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Todos los registros con una

Métrica	Descripción
	antigüedad superior a esta edad se han entregado al OpenSearch Servicio. Unidades: segundos
`DeliveryToAmazonOpenSearchService.Records`	El número de registros indexados al OpenSearch Servicio durante el período de tiempo especificado. Unidades: recuento
`DeliveryToAmazonOpenSearchService.Success`	La suma de los registros indexados correctamente con respecto a la suma de los registros que se intentaron indexar.
`DeliveryToS3.Bytes`	Número de bytes entregados en Amazon S3 durante el periodo de tiempo especificado. Amazon Data Firehose emite esta métrica solo cuando se habilita la copia de seguridad de todos los documentos. Unidades: recuento
`DeliveryToS3.DataFreshness`	La antigüedad (desde que se utilizó Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Los registros anteriores a este valor se han enviado al bucket de S3. Amazon Data Firehose emite esta métrica solo cuando se habilita la copia de seguridad de todos los documentos. Unidades: segundos
`DeliveryToS3.Records`	Número de registros entregados en Amazon S3 durante el periodo de tiempo especificado. Amazon Data Firehose emite esta métrica solo cuando se habilita la copia de seguridad de todos los documentos. Unidades: recuento

Métrica	Descripción
DeliveryToS3.Success	Suma de comandos put de Amazon S3 ejecutados correctamente con respecto a la suma de todos los comandos put de Amazon S3. Amazon Data Firehose siempre emite esta métrica, independientemente de si la copia de seguridad está habilitada solo para los documentos con errores o para todos los documentos.
DeliveryToAmazonOpenSearchService.AuthFailure	Error de autenticación o autorización. Compruebe la política del clúster de OS/ES y los permisos del rol. Un 0 indica que no hay ningún problema y un 1 indica un error de autenticación.
DeliveryToAmazonOpenSearchService.DeliveryRejected	Error de entrega rechazada. Compruebe la política del clúster de OS/ES y los permisos del rol. Un 0 indica que no hay ningún problema y un 1 indica que se ha producido un error en la entrega.

Entrega a Serverless OpenSearch

Métrica	Descripción
DeliveryToAmazonOpenSearchServerless.Bytes	El número de bytes indexados a OpenSearch Serverless durante el período de tiempo especificado. Unidades: bytes
DeliveryToAmazonOpenSearchServerless.DataFreshness	La antigüedad (desde que se utilizó Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Todos los registros con una antigüedad superior a esta edad se han enviado a OpenSearch Serverless. Unidades: segundos

Métrica	Descripción
DeliveryToAmazonOpenSearchServerless.Records	El número de registros indexados en OpenSearch Serverless durante el período de tiempo especificado. Unidades: recuento
DeliveryToAmazonOpenSearchServerless.Success	La suma de los registros indexados correctamente con respecto a la suma de los registros que se intentaron indexar.
DeliveryToS3.Bytes	Número de bytes entregados en Amazon S3 durante el periodo de tiempo especificado. Amazon Data Firehose emite esta métrica solo cuando se habilita la copia de seguridad de todos los documentos. Unidades: recuento
DeliveryToS3.DataFreshness	La antigüedad (desde que se utilizó Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Los registros anteriores a este valor se han enviado al bucket de S3. Amazon Data Firehose emite esta métrica solo cuando se habilita la copia de seguridad de todos los documentos. Unidades: segundos
DeliveryToS3.Records	Número de registros entregados en Amazon S3 durante el periodo de tiempo especificado. Amazon Data Firehose emite esta métrica solo cuando se habilita la copia de seguridad de todos los documentos. Unidades: recuento

Métrica	Descripción
DeliveryToS3.Success	Suma de comandos put de Amazon S3 ejecutados correctamente con respecto a la suma de todos los comandos put de Amazon S3. Amazon Data Firehose siempre emite esta métrica, independientemente de si la copia de seguridad está habilitada solo para los documentos con errores o para todos los documentos.
DeliveryToAmazonOpenSearchServerless.AuthFailure	Error de autenticación o autorización. Compruebe la política del clúster de OS/ES y los permisos del rol. Un 0 indica que no hay ningún problema y un 1 indica que se ha producido un error de autenticación.
DeliveryToAmazonOpenSearchServerless.DeliveryRejected	Error de entrega rechazada. Compruebe la política del clúster de OS/ES y los permisos del rol. Un 0 indica que no hay ningún problema y un 1 indica que se ha producido un error en la entrega.

Entrega en Amazon Redshift

Métrica	Descripción
DeliveryToRedshift.Bytes	Número de bytes copiados en Amazon Redshift durante el periodo de tiempo especificado. Unidades: recuento
DeliveryToRedshift.Records	Número de registros copiados en Amazon Redshift durante el periodo de tiempo especificado. Unidades: recuento
DeliveryToRedshift.Success	Suma de comandos COPY de Amazon Redshift ejecutados correctamente con respecto a la suma de todos los comandos COPY de Amazon Redshift.

Métrica	Descripción
DeliveryToS3.Bytes	Número de bytes entregados en Amazon S3 durante el periodo de tiempo especificado. Unidades: bytes
DeliveryToS3.DataFreshness	La antigüedad (desde que se utilizó Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Los registros anteriores a este valor se han enviado al bucket de S3. Unidades: segundos
DeliveryToS3.Records	Número de registros entregados en Amazon S3 durante el periodo de tiempo especificado. Unidades: recuento
DeliveryToS3.Success	Suma de comandos put de Amazon S3 ejecutados correctamente con respecto a la suma de todos los comandos put de Amazon S3.
BackupToS3.Bytes	Número de bytes entregados en Amazon S3 para copias de seguridad durante el periodo de tiempo especificado. Amazon Data Firehose emite esta métrica cuando se habilita la copia de seguridad en Amazon S3. Unidades: recuento
BackupToS3.DataFreshness	Antigüedad (desde el ingreso a Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Los registros anteriores a este valor se han entregado en el bucket de Amazon S3 con fines de copia de seguridad. Amazon Data Firehose emite esta métrica cuando se habilita la copia de seguridad en Amazon S3. Unidades: segundos

Métrica	Descripción
BackupToS3.Records	Número de registros entregados en Amazon S3 para copias de seguridad durante el periodo de tiempo especificado. Amazon Data Firehose emite esta métrica cuando se habilita la copia de seguridad en Amazon S3. Unidades: recuento
BackupToS3.Success	Suma de comandos put de copia de seguridad de Amazon S3 ejecutados correctamente con respecto a la suma de todos los comandos put de copia de seguridad de Amazon S3. Amazon Data Firehose emite esta métrica cuando se habilita la copia de seguridad en Amazon S3.

Entrega en Amazon S3

Las métricas de la siguiente tabla están relacionadas con la entrega a Amazon S3 cuando es el destino principal de la transmisión Firehose.

Métrica	Descripción
DeliveryToS3.Bytes	Número de bytes entregados en Amazon S3 durante el periodo de tiempo especificado. Unidades: bytes
DeliveryToS3.DataFreshness	La antigüedad (desde que se utilizó Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Los registros anteriores a este valor se han enviado al bucket de S3. Unidades: segundos
DeliveryToS3.Records	Número de registros entregados en Amazon S3 durante el periodo de tiempo especificado.

Métrica	Descripción
	Unidades: recuento
`DeliveryToS3.Success`	Suma de comandos put de Amazon S3 ejecutados correctamente con respecto a la suma de todos los comandos put de Amazon S3.
`BackupToS3.Bytes`	Número de bytes entregados en Amazon S3 para copias de seguridad durante el periodo de tiempo especificado. Amazon Data Firehose emite esta métrica cuando la copia de seguridad está habilitada (lo que solo es posible cuando la transformación de datos también está habilitad a). Unidades: recuento
`BackupToS3.DataFreshness`	Antigüedad (desde el ingreso a Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Los registros anteriores a este valor se han entregado en el bucket de Amazon S3 con fines de copia de seguridad. Amazon Data Firehose emite esta métrica cuando la copia de seguridad está habilitada (lo que solo es posible cuando la transformación de datos también está habilitada). Unidades: segundos
`BackupToS3.Records`	Número de registros entregados en Amazon S3 para copias de seguridad durante el periodo de tiempo especificado. Amazon Data Firehose emite esta métrica cuando la copia de seguridad está habilitada (lo que solo es posible cuando la transformación de datos también está habilitada). Unidades: recuento

Métrica	Descripción
`BackupToS3.Success`	Suma de comandos put de copia de seguridad de Amazon S3 ejecutados correctamente con respecto a la suma de todos los comandos put de copia de seguridad de Amazon S3. Amazon Data Firehose emite esta métrica cuando la copia de seguridad está habilitada (lo que solo es posible cuando la transformación de datos también está habilitada).

Entrega a Snowflake

Métrica	Descripción
`DeliveryToSnowflake.Bytes`	El número de bytes entregados a Snowflake durante el período de tiempo especificado. Unidades: bytes
`DeliveryToSnowflake.DataFreshness`	Antigüedad (desde que entré en Firehose hasta ahora) del registro más antiguo de Firehose. Cualquier registro anterior a esta edad ha sido entregado a Snowflake . Tenga en cuenta que la confirmación de datos en Snowflake puede tardar unos segundos después de que la llamada de inserción de Firehose se haya realizado correctamente. Para ver el tiempo que se tarda en enviar los datos a Snowflake, consulta la métrica. `DeliveryToSnowflake.DataCommitLatency` Unidades: segundos
`DeliveryToSnowflake.DataCommitLatency`	El tiempo que tardan los datos en enviarse a Snowflake después de que Firehose haya insertado los registros correctamente. Unidades: segundos

Métrica	Descripción
DeliveryToSnowflake.Records	El número de registros entregados a Snowflake durante el período de tiempo especificado. Unidades: recuento
DeliveryToSnowflake.Success	La suma de las llamadas de inserción realizadas correctamente a Snowflake respecto a la suma de las llamadas de inserción que se intentaron realizar.
DeliveryToS3.Bytes	Número de bytes entregados en Amazon S3 durante el periodo de tiempo especificado. Esta métrica solo está disponible cuando se produce un error en la entrega a Snowflake y Firehose intenta hacer copias de seguridad de los datos fallidos en S3. Unidades: bytes
DeliveryToS3.Records	Número de registros entregados en Amazon S3 durante el periodo de tiempo especificado. Esta métrica solo está disponible cuando se produce un error en la entrega a Snowflake y Firehose intenta hacer copias de seguridad de los datos fallidos en S3. Unidades: recuento
DeliveryToS3.Success	Suma de comandos put de Amazon S3 ejecutados correctamente con respecto a la suma de todos los comandos put de Amazon S3. Esta métrica solo está disponible cuando se produce un error en la entrega a Snowflake y Firehose intenta hacer copias de seguridad de los datos fallidos en S3.

Métrica	Descripción
BackupToS3.DataFreshness	Antigüedad (desde Firehose hasta la actualidad) del registro más antiguo de Firehose. Todos los registros con una antigüedad superior a esta edad se guardan en el bucket de Amazon S3. Esta métrica está disponible cuando la transmisión Firehose está configurada para hacer copias de seguridad de todos los datos. Unidades: segundos
BackupToS3.Records	Número de registros entregados en Amazon S3 para copias de seguridad durante el periodo de tiempo especificado. Esta métrica está disponible cuando la transmisión Firehose está configurada para hacer copias de seguridad de todos los datos. Unidades: recuento
BackupToS3.Bytes	Número de bytes entregados en Amazon S3 para copias de seguridad durante el periodo de tiempo especific ado. Esta métrica está disponible cuando la transmisi ón Firehose está configurada para hacer copias de seguridad de todos los datos. Unidades: recuento
BackupToS3.Success	La suma de los comandos put de Amazon S3 realizado s correctamente para la copia de seguridad respecto a la suma de todos los comandos de backup put de Amazon S3. Firehose emite esta métrica cuando la transmisi ón Firehose está configurada para hacer copias de seguridad de todos los datos.

Entrega a Splunk

Métrica	Descripción
DeliveryToSplunk.Bytes	El número de bytes enviados a Splunk durante el periodo de tiempo especificado. Unidades: bytes
DeliveryToSplunk.DataAckLatency	El tiempo aproximado que tarda en recibir un acuse de recibo de Splunk después de que Amazon Data Firehose envíe sus datos. La tendencia creciente o decreciente de esta métrica es más útil que el valor aproximado o absoluto. Las tendencias crecientes pueden indicar velocidades de indexación y de reconocimiento más lentas de los indexadores de Splunk. Unidades: segundos
DeliveryToSplunk.DataFreshness	Antigüedad (desde el ingreso a Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Los registros anteriores a este valor se han enviado a Splunk. Unidades: segundos
DeliveryToSplunk.Records	El número de registros enviados a Splunk durante el periodo de tiempo especificado. Unidades: recuento
DeliveryToSplunk.Success	La suma de los registros indexados correctamente con respecto a la suma de los registros que se intentaron indexar.
DeliveryToS3.Success	Suma de comandos put de Amazon S3 ejecutados correctamente con respecto a la suma de todos los comandos put de Amazon S3. Esta métrica se emite

Métrica	Descripción
	cuando la copia de seguridad en Amazon S3 está habilitada.
BackupToS3.Bytes	Número de bytes entregados en Amazon S3 para copias de seguridad durante el periodo de tiempo especificado. Amazon Data Firehose emite esta métrica cuando la transmisión Firehose está configurada para hacer copias de seguridad de todos los documentos. Unidades: recuento
BackupToS3.DataFreshness	Antigüedad (desde el ingreso a Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Los registros anteriores a este valor se han entregado en el bucket de Amazon S3 con fines de copia de seguridad. Amazon Data Firehose emite esta métrica cuando la transmisión Firehose está configurada para hacer copias de seguridad de todos los documentos. Unidades: segundos
BackupToS3.Records	Número de registros entregados en Amazon S3 para copias de seguridad durante el periodo de tiempo especificado. Amazon Data Firehose emite esta métrica cuando la transmisión Firehose está configurada para hacer copias de seguridad de todos los documentos. Unidades: recuento
BackupToS3.Success	Suma de comandos put de copia de seguridad de Amazon S3 ejecutados correctamente con respecto a la suma de todos los comandos put de copia de seguridad de Amazon S3. Amazon Data Firehose emite esta métrica cuando la transmisión Firehose está configurada para hacer copias de seguridad de todos los documentos.

Entrega en puntos de conexión HTTP

Métrica	Descripción
`DeliveryToHttpEndpoint.Bytes`	Número de bytes entregados correctamente en el punto de conexión HTTP. Unidades: bytes
`DeliveryToHttpEndpoint.Records`	Número de registros entregados correctamente en el punto de conexión HTTP. Unidades: recuentos.
`DeliveryToHttpEndpoint.DataFreshness`	Antigüedad del registro más antiguo de Amazon Data Firehose. Unidades: segundos
`DeliveryToHttpEndpoint.Success`	Suma de todas las solicitudes de entrega de datos realizadas correctamente al punto de conexión HTTP Unidades: recuento
`DeliveryToHttpEndpoint.ProcessedBytes`	Número de bytes procesados intentados, incluidos los reintentos.
`DeliveryToHttpEndpoint.ProcessedRecords`	Número de registros intentados, incluidos los reintentos.

Métricas de adquisición de datos

Temas

- Ingesta de datos a través de Kinesis Data Streams
- Adquisición de datos a través de PUT directo
- Ingesta de datos de MSK

Ingesta de datos a través de Kinesis Data Streams

Métrica	Descripción
`DataReadFromKinesisStream.Bytes`	Cuando el origen de datos es un flujo de datos de Kinesis, esta métrica indica el número de bytes leídos de dicho flujo. Este número incluye las repeticiones de lecturas debido a conmutaciones por error. Unidades: bytes
`DataReadFromKinesisStream.Records`	Cuando el origen de datos es un flujo de datos de Kinesis, esta métrica indica el número de registros leídos de dicho flujo. Este número incluye las repeticiones de lecturas debido a conmutaciones por error. Unidades: recuento
`ThrottledDescribeStream`	El número total de veces que se limita la operación `DescribeStream` cuando el origen de datos es una secuencia de datos de Kinesis. Unidades: recuento
`ThrottledGetRecords`	El número total de veces que se limita la operación `GetRecords` cuando el origen de datos es una secuencia de datos de Kinesis. Unidades: recuento
`ThrottledGetShardIterator`	El número total de veces que se limita la operación `GetShardIterator` cuando el origen de datos es una secuencia de datos de Kinesis. Unidades: recuento

Adquisición de datos a través de PUT directo

Métrica	Descripción
`BackupToS3.Bytes`	Número de bytes entregados en Amazon S3 para copias de seguridad durante el periodo de tiempo especificado. Amazon Data Firehose emite esta métrica cuando se habilita la transformación de datos para los destinos de Amazon S3 o Amazon Redshift. Unidades: bytes
`BackupToS3.DataFreshness`	Antigüedad (desde el ingreso a Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Los registros anteriores a este valor se han entregado en el bucket de Amazon S3 con fines de copia de seguridad. Amazon Data Firehose emite esta métrica cuando se habilita la transformación de datos para los destinos de Amazon S3 o Amazon Redshift. Unidades: segundos
`BackupToS3.Records`	Número de registros entregados en Amazon S3 para copias de seguridad durante el periodo de tiempo especificado. Amazon Data Firehose emite esta métrica cuando se habilita la transformación de datos para los destinos de Amazon S3 o Amazon Redshift. Unidades: recuento
`BackupToS3.Success`	Suma de comandos put de copia de seguridad de Amazon S3 ejecutados correctamente con respecto a la suma de todos los comandos put de copia de seguridad de Amazon S3. Amazon Data Firehose emite esta métrica cuando se habilita la transformación de datos para los destinos de Amazon S3 o Amazon Redshift.
`BytesPerSecondLimit`	El número máximo actual de bytes por segundo que una transmisión de Firehose puede ingerir antes de la

Métrica	Descripción
	regulación. Para solicitar un aumento de este límite, vaya a AWS Support Center y elija Crear caso y, a continuac ión, seleccione Aumento del límite de servicio.
`DataReadFromKinesisStream.Bytes`	Cuando el origen de datos es un flujo de datos de Kinesis, esta métrica indica el número de bytes leídos de dicho flujo. Este número incluye las repeticiones de lecturas debido a conmutaciones por error. Unidades: bytes
`DataReadFromKinesisStream.Records`	Cuando el origen de datos es un flujo de datos de Kinesis, esta métrica indica el número de registros leídos de dicho flujo. Este número incluye las repeticiones de lecturas debido a conmutaciones por error. Unidades: recuento
`DeliveryToAmazonOpenSearchService.Bytes`	El número de bytes indexados al OpenSearch Servicio durante el período de tiempo especificado. Unidades: bytes
`DeliveryToAmazonOpenSearchService.DataFreshness`	La antigüedad (desde que se utilizó Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Todos los registros con una antigüedad superior a esta edad se han entregado al OpenSearch Servicio. Unidades: segundos
`DeliveryToAmazonOpenSearchService.Records`	El número de registros indexados al OpenSearch Servicio durante el período de tiempo especificado. Unidades: recuento

Métrica	Descripción
DeliveryToAmazonOpenSearchService.Success	La suma de los registros indexados correctamente con respecto a la suma de los registros que se intentaron indexar.
DeliveryToRedshift.Bytes	Número de bytes copiados en Amazon Redshift durante el periodo de tiempo especificado. Unidades: bytes
DeliveryToRedshift.Records	Número de registros copiados en Amazon Redshift durante el periodo de tiempo especificado. Unidades: recuento
DeliveryToRedshift.Success	Suma de comandos COPY de Amazon Redshift ejecutados correctamente con respecto a la suma de todos los comandos COPY de Amazon Redshift.
DeliveryToS3.Bytes	Número de bytes entregados en Amazon S3 durante el periodo de tiempo especificado. Unidades: bytes
DeliveryToS3.DataFreshness	La antigüedad (desde que se utilizó Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Los registros anteriores a este valor se han enviado al bucket de S3. Unidades: segundos
DeliveryToS3.Records	Número de registros entregados en Amazon S3 durante el periodo de tiempo especificado. Unidades: recuento
DeliveryToS3.Success	Suma de comandos put de Amazon S3 ejecutados correctamente con respecto a la suma de todos los comandos put de Amazon S3.

Métrica	Descripción
DeliveryToSplunk.Bytes	El número de bytes enviados a Splunk durante el periodo de tiempo especificado. Unidades: bytes
DeliveryToSplunk.DataAckLatency	El tiempo aproximado que tarda en recibir un acuse de recibo de Splunk después de que Amazon Data Firehose envíe sus datos. La tendencia creciente o decreciente de esta métrica es más útil que el valor aproximado o absoluto. Las tendencias crecientes pueden indicar velocidades de indexación y de reconocimiento más lentas de los indexadores de Splunk. Unidades: segundos
DeliveryToSplunk.DataFreshness	Antigüedad (desde el ingreso a Amazon Data Firehose hasta la actualidad) del registro más antiguo de Amazon Data Firehose. Los registros anteriores a este valor se han enviado a Splunk. Unidades: segundos
DeliveryToSplunk.Records	El número de registros enviados a Splunk durante el periodo de tiempo especificado. Unidades: recuento
DeliveryToSplunk.Success	La suma de los registros indexados correctamente con respecto a la suma de los registros que se intentaron indexar.

Métrica	Descripción
`IncomingBytes`	El número de bytes ingresados correctamente en la transmisión Firehose durante el período de tiempo especificado. La ingesta de datos se puede limitar cuando supera uno de los límites de flujo de Firehose. Los datos restringidos no se tendrán en cuenta. `IncomingBytes` Unidades: bytes
`IncomingPutRequests`	El número de PutRecordBatch solicitudes PutRecord satisfactorias y durante un período de tiempo específico. Unidades: recuento
`IncomingRecords`	El número de registros ingresados correctamente en la transmisión Firehose durante el período de tiempo especificado. La ingesta de datos se puede limitar cuando supera uno de los límites de flujo de Firehose. Los datos restringidos no se tendrán en cuenta. `IncomingRecords` Unidades: recuento
`KinesisMillisBehindLatest`	Cuando el origen de datos es una secuencia de datos de Kinesis, esta métrica indica el número de milisegundos de retraso que lleva el último registro leído con respecto al registro más reciente de la secuencia de datos de Kinesis. Unidades: milisegundos
`RecordsPerSecondLimit`	El número máximo actual de registros por segundo que una transmisión de Firehose puede ingerir antes de la aceleración. Unidades: recuento

Métrica	Descripción
`ThrottledRecords`	La cantidad de registros que se limitaron porque la ingesta de datos superó uno de los límites de flujo de Firehose. Unidades: recuento

Ingesta de datos de MSK

Métrica	Descripción
`DataReadFromSource .Records`	Número de registros leídos del tema de Kafka de origen. Unidades: recuento
`DataReadFromSource.Bytes`	Número de bytes leídos del tema de Kafka de origen. Unidades: bytes
`SourceThrottled.Delay`	Tiempo que tarda el clúster de Kafka de origen en devolver los registros del tema de Kafka de origen. Unidades: milisegundos
`BytesPerSecondLimit`	Límite actual de rendimiento al que Firehose va a leer desde cada partición del tema de Kafka de origen. Unidades: bytes/segundo
`KafkaOffsetLag`	Diferencia entre el mayor desplazamiento del registro que Firehose ha leído del tema de Kafka de origen y el mayor desplazamiento del registro disponible del tema de Kafka de origen. Unidades: recuento
`FailedValidation.Records`	Número de registros que no han superado la validación de registros.

Métrica	Descripción
	Unidades: recuento
`FailedValidation.Bytes`	Número de bytes que no han superado la validación de registros. Unidades: bytes
`DataReadFromSource.Backpressured`	Indica que una transmisión de Firehose se retrasa en la lectura de los registros de la partición de origen, ya sea porque se ha superado BytesPerSecondLimit cada partición o porque el flujo normal de entrega es lento o se ha detenido Unidades: booleano

Métricas a nivel de API CloudWatch

El espacio de nombres `AWS/Firehose` incluye las siguientes métricas de nivel de API.

Métrica	Descripción
`DescribeDeliveryStream.Latency`	El tiempo que tarda cada operación `DescribeDeliveryStream` , medido durante el periodo de tiempo especificado. Unidades: milisegundos
`DescribeDeliveryStream.Requests`	El número total de solicitudes `DescribeDeliveryStream` . Unidades: recuento
`ListDeliveryStreams.Latency`	El tiempo que tarda cada operación `ListDeliveryStream` , medido durante el periodo de tiempo especificado. Unidades: milisegundos

Métrica	Descripción
`ListDeliveryStreams.Requests`	El número total de solicitudes `ListFirehose` . Unidades: recuento
`PutRecord.Bytes`	El número de bytes que se utilizan en la transmisión Firehose `PutRecord` durante el período de tiempo especificado. Unidades: bytes
`PutRecord.Latency`	El tiempo que tarda cada operación `PutRecord` , medido durante el periodo de tiempo especificado. Unidades: milisegundos
`PutRecord.Requests`	El número total de solicitudes `PutRecord` , que es igual al número total de registros de las operaciones `PutRecord` . Unidades: recuento
`PutRecordBatch.Bytes`	El número de bytes que se utilizan en la transmisión Firehose `PutRecordBatch` durante el período de tiempo especificado. Unidades: bytes
`PutRecordBatch.Latency`	El tiempo que tarda cada operación `PutRecordBatch` , medido durante el periodo de tiempo especificado. Unidades: milisegundos
`PutRecordBatch.Records`	El número total de registros de las operaciones `PutRecordBatch` . Unidades: recuento

Métrica	Descripción
PutRecordBatch.Requests	El número total de solicitudes PutRecordBatch . Unidades: recuento
PutRequestsPerSecondLimit	El número máximo de solicitudes de venta por segundo que puede gestionar una transmisión de Firehose antes de la aceleración. Este número incluye PutRecord todas las solicitudes. PutRecordBatch Unidades: recuento
ThrottledDescribeStream	El número total de veces que se limita la operación DescribeStream cuando el origen de datos es una secuencia de datos de Kinesis. Unidades: recuento
ThrottledGetRecords	El número total de veces que se limita la operación GetRecords cuando el origen de datos es una secuencia de datos de Kinesis. Unidades: recuento
ThrottledGetShardIterator	El número total de veces que se limita la operación GetShardIterator cuando el origen de datos es una secuencia de datos de Kinesis. Unidades: recuento
UpdateDeliveryStream.Latency	El tiempo que tarda cada operación UpdateDeliveryStream , medido durante el periodo de tiempo especificado. Unidades: milisegundos

Métrica	Descripción
`UpdateDeliveryStream.Requests`	El número total de solicitudes `UpdateDeliveryStream` . Unidades: recuento

CloudWatch Métricas de transformación de datos

Si la transformación de datos con Lambda está habilitada, el espacio de nombres `AWS/Firehose` incluye las siguientes métricas.

Métrica	Descripción
`ExecuteProcessing.Duration`	El tiempo que tarda Firehose en invocar cada función de Lambda. Unidades: milisegundos
`ExecuteProcessing.Success`	Suma de las invocaciones de funciones de Lambda correctas con respecto a la suma del total de invocaciones de funciones de Lambda.
`SucceedProcessing.Records`	Número de registros procesados correctamente durante el periodo de tiempo especificado. Unidades: recuento
`SucceedProcessing.Bytes`	Número de bytes procesados correctamente durante el periodo de tiempo especificado. Unidades: bytes

CloudWatch Registra las métricas de descompresión

Si la descompresión está habilitada para la entrega de CloudWatch registros, el espacio de `AWS/Firehose` nombres incluye las siguientes métricas.

Métrica	Descripción
`OutputDecompressedBytes.Success`	Se descomprimieron correctamente los datos en bytes Unidades: bytes
`OutputDecompressedBytes.Failed`	Fallo al descomprimir los datos en bytes Unidades: bytes
`OutputDecompressedRecords.Success`	Número de registros descomprimidos correctamente Unidades: recuento
`OutputDecompressedRecords.Failed`	Número de registros descomprimidos fallidos Unidades: recuento

Métricas de conversión de CloudWatch formato

Si la conversión del formato está habilitada, el espacio de nombres `AWS/Firehose` incluye las siguientes métricas.

Métrica	Descripción
`SucceedConversion.Records`	El número de registros convertidos correctamente. Unidades: recuento
`SucceedConversion.Bytes`	El tamaño de los registros convertidos correctamente. Unidades: bytes
`FailedConversion.Records`	El número de registros que no se han podido convertir. Unidades: recuento
`FailedConversion.Bytes`	El tamaño de los registros que no se han podido convertir.

Métrica	Descripción
	Unidades: bytes

Métricas de cifrado del lado del servidor (SSE) CloudWatch

El espacio de nombres `AWS/Firehose` incluye las siguientes métricas relacionadas con SSE.

Métrica	Descripción
`KMSKeyAccessDenied`	El número de veces que el servicio encuentra una `KMSAccessDeniedException` para la transmisión Firehose. Unidades: recuento
`KMSKeyDisabled`	El número de veces que el servicio encuentra una `KMSDisabledException` para la transmisión Firehose. Unidades: recuento
`KMSKeyInvalidState`	El número de veces que el servicio encuentra una `KMSInvalidStateException` para la transmisión Firehose. Unidades: recuento
`KMSKeyNotFound`	El número de veces que el servicio encuentra una `KMSNotFoundException` para la transmisión Firehose. Unidades: recuento

Dimensiones de Amazon Data Firehose

Para filtrar las métricas por flujo de Firehose, usa la `DeliveryStreamName` dimensión.

Métricas de uso de Amazon Data Firehose

Puede utilizar las métricas CloudWatch de uso para proporcionar visibilidad sobre el uso de los recursos de su cuenta. Usa estas métricas para visualizar tu uso actual del servicio en CloudWatch gráficos y paneles.

Las métricas de uso de la cuota de servicio se encuentran en el espacio de nombres AWS/Usage y se recopilan cada minuto.

Actualmente, el único nombre de métrica de este espacio de nombres que se publica es. CloudWatch `ResourceCount` Esta métrica se publica con las dimensiones `Service`, `Class`, `Type` y `Resource`.

Métrica	Descripción
`ResourceCount`	El número de los recursos especificados que se ejecutan en su cuenta. Los recursos se definen por las dimension es asociadas a la métrica. La estadística más útil para esta métrica es MAXIMUM, que representa el número máximo de recursos utilizados durante el periodo de un minuto.

Las siguientes dimensiones se utilizan para refinar las métricas de uso que publica Amazon Data Firehose.

Dimensión	Descripción
`Service`	El nombre del AWS servicio que contiene el recurso. Para las métricas de uso de Amazon Data Firehose, el valor de esta dimensión es. `Firehose`
`Class`	La clase de recurso a la que se realiza el seguimiento. Las métricas de uso de la API Amazon Data Firehose utilizan esta dimensión con un valor de. `None`

Dimensión	Descripción
`Type`	El tipo de recurso al que se realiza el seguimiento. Actualmente, cuando la dimensión Service es `Firehose`, el único valor válido para Type es `Resource`.
`Resource`	El nombre del AWS recurso. Actualmente, cuando la dimensión Service es `Firehose`, el único valor válido para Resource es `DeliveryStreams` .

Acceso a CloudWatch las métricas de Amazon Data Firehose

Puede supervisar las métricas de Amazon Data Firehose mediante la CloudWatch consola, la línea de comandos o CloudWatch la API. Los siguientes procedimientos le muestran cómo obtener acceso a las métricas a través de los distintos métodos descritos a continuación.

Para acceder a las métricas mediante la consola CloudWatch

1. Abra la CloudWatch consola en https://console.aws.amazon.com/cloudwatch/.
2. Seleccione una región en la barra de navegación.
3. En el panel de navegación, seleccione Métricas.
4. Elija el espacio de nombres de Firehose.
5. Elige Firehose Stream Metrics o Firehose Metrics.
6. Seleccione una métrica para añadirla al gráfico.

Para acceder a las métricas mediante el AWS CLI

Utilice las métricas y get-metric-statisticslos comandos de la lista.

```
aws cloudwatch list-metrics --namespace "AWS/Firehose"
```

```
aws cloudwatch get-metric-statistics --namespace "AWS/Firehose" \
--metric-name DescribeDeliveryStream.Latency --statistics Average --period 3600 \
--start-time 2017-06-01T00:00:00Z --end-time 2017-06-30T00:00:00Z
```

Supervisión de Amazon Data Firehose mediante registros CloudWatch

Amazon Data Firehose se integra con Amazon CloudWatch Logs para que pueda ver los registros de errores específicos cuando se produzca un error en la invocación de Lambda para la transformación o la entrega de datos. Puedes activar el registro de errores de Amazon Data Firehose al crear tu transmisión de Firehose.

Si habilita el registro de errores de Amazon Data Firehose en la consola de Amazon Data Firehose, se crearán un grupo de registros y los flujos de registros correspondientes para la transmisión de Firehose en su nombre. El formato del nombre del grupo de registros es`aws/kinesisfirehose/`*`delivery-stream-name`*, donde *`delivery-stream-name`* es el nombre de la transmisión Firehose correspondiente. `DestinationDelivery`es el flujo de registro que se crea y se utiliza para registrar cualquier error relacionado con la entrega al destino principal. Otro flujo de registro denominado `BackupDelivery` se crea solo si la copia de seguridad de S3 está habilitada para el destino. El flujo de registro `BackupDelivery` se utiliza para registrar cualquier error relacionado con la entrega en la copia de seguridad de S3.

Por ejemplo, si crea una transmisión de Firehose «MyStream» con Amazon Redshift como destino y habilita el registro de errores de Amazon Data Firehose, se crea lo siguiente en su nombre: un grupo de registros denominado y dos secuencias de registros `aws/kinesisfirehose/MyStream` denominadas y. `DestinationDelivery` BackupDelivery En este ejemplo, se utilizará `DestinationDelivery` para registrar cualquier error relacionado con la entrega en el destino de Amazon Redshift y también en el destino intermedio de S3. `BackupDelivery`, en caso de que la copia de seguridad de S3 esté habilitada, se utilizará para registrar cualquier error relacionado con la entrega en el bucket de copias de seguridad de S3.

Puede activar el registro de errores de Amazon Data Firehose a través de la API o AWS CloudFormation mediante la `CloudWatchLoggingOptions` configuración. AWS CLI Para ello, cree previamente un grupo de registro y un flujo de registro. Recomendamos reservar ese grupo de registros y ese flujo de registros exclusivamente para el registro de errores de Amazon Data Firehose. Asegúrese también de que la política de IAM asociada tenga el permiso `"logs:putLogEvents"`. Para obtener más información, consulte Control del acceso con Amazon Data Firehose.

Tenga en cuenta que Amazon Data Firehose no garantiza que todos los registros de errores de entrega se envíen a CloudWatch Logs. En circunstancias en las que la tasa de errores en la entrega

es alta, Amazon Data Firehose toma muestras de los registros de errores de entrega antes de enviarlos a CloudWatch Logs.

Hay un cargo nominal por los registros de errores enviados a CloudWatch Logs. Para obtener más información, consulta los CloudWatch precios de Amazon.

Contenido

- Errores de entrega de datos

Errores de entrega de datos

La siguiente es una lista de mensajes y códigos de error en la entrega de datos para cada destino de Amazon Data Firehose. Cada mensaje de error también describe qué debe hacerse para solucionar el problema.

Errores

- Errores de entrega de datos de Amazon S3
- Errores de entrega de datos de Amazon Redshift
- Errores en la entrega de datos de Snowflake
- Errores de entrega de datos relacionados con Splunk
- ElasticSearch Errores en la entrega de datos
- Errores de entrega de datos de puntos de conexión HTTPS
- Errores en la entrega de datos de Amazon OpenSearch Service
- Errores de invocación de Lambda
- Errores de invocación de Kinesis
- Errores de invocación de Kinesis DirectPut
- AWS Glue Errores de invocación
- DataFormatConversion Errores de invocación

Errores de entrega de datos de Amazon S3

Amazon Data Firehose puede enviar los siguientes errores relacionados con Amazon S3 a Logs. CloudWatch

Código de error	Mensaje de error e información
`S3.KMS.NotFoundException`	«No se encontró la AWS KMS clave proporcionada. Si está utilizando lo que cree que es una AWS KMS clave válida con la función correcta, compruebe si hay algún problema con la cuenta a la que está asociada la AWS KMS clave».
`S3.KMS.RequestLimitExceeded`	"El límite de solicitudes de KMS por segundo se ha superado al intentar cifrar objetos de S3. Aumente el límite de solicitudes por segundo." Para obtener más información, consulte Límites en la Guía para desarrolladores de AWS Key Management Service .
`S3.AccessDenied`	"Acceso denegado. Asegúrese de que la política de confianza para la función de IAM proporcionada permita a Amazon Data Firehose asumir la función y que la política de acceso permita el acceso al bucket de S3».
`S3.AccountProblem`	«Hay un problema con su AWS cuenta que impide que la operación se complete correctamente. Contact AWS Support.”
`S3.AllAccessDisabled`	"El acceso a la cuenta proporcionada se ha deshabilitado. Póngase en contacto con AWS Support».
`S3.InvalidPayer`	"El acceso a la cuenta proporcionada se ha deshabilitado. Póngase en contacto con AWS Support».
`S3.NotSignedUp`	“The account is not signed up for Amazon S3. Inscríbase o utilice otra cuenta."
`S3.NoSuchBucket`	"El bucket especificado no existe. Créelo o utilice otro que exista."
`S3.MethodNotAllowed`	"Este recurso no admite el método especificado. Modify the bucket’s policy to allow the correct Amazon S3 operation permissions.”
`InternalError`	"Se ha producido un error interno al intentar entregar los datos. Se volverá a intentar la entrega; si el error persiste, se informará al respecto AWS para su resolución».

Código de error	Mensaje de error e información
`S3.KMS.KeyDisabled`	“The provided KMS key is disabled. Enable the key or use a different key.”
`S3.KMS.InvalidStateException`	“The provided KMS key is in an invalid state. Please use a different key.”
`KMS.InvalidStateException`	“The provided KMS key is in an invalid state. Please use a different key.”
`KMS.DisabledException`	“The provided KMS key is disabled. Please fix the key or use a different key.”
`S3.SlowDown`	“The rate of put request to the specified bucket was too high. Aumente el tamaño del búfer de flujo Firehose o reduzca las solicitudes de venta de otras aplicaciones».
`S3.SubscriptionRequired`	“Access was denied when calling S3. Ensure that the IAM role and the KMS Key (if provided) passed in has Amazon S3 subscription.”
`S3.InvalidToken`	“The provided token is malformed or otherwise invalid. Please check the credentials provided.”
`S3.KMS.KeyNotConfigured`	“KMS key not configured. Configure su MasterKey ID de KMS o deshabilite el cifrado de su bucket de S3».
`S3.KMS.AsymmetricCMKNotSupported`	“Amazon S3 supports only symmetric CMKs. You cannot use an asymmetric CMK to encrypt your data in Amazon S3. Para obtener el tipo de CMK, utilice la DescribeKey operación KMS».
`S3.IllegalLocationConstraintException`	“Firehose currently uses s3 global endpoint for data delivery to the configured s3 bucket. The region of the configured s3 bucket doesn't support s3 global endpoint. Cree una transmisión de Firehose en la misma región que el depósito s3 o utilice el depósito s3 en la región que admite el punto final global».

Código de error	Mensaje de error e información
`S3.InvalidPrefixConfigurationException`	“The custom s3 prefix used for the timestamp evaluation is invalid. Check your s3 prefix contains valid expressions for the current date and time of the year.”
`DataFormatConversion.MalformedData`	“Illegal character found between tokens.”

Errores de entrega de datos de Amazon Redshift

Amazon Data Firehose puede enviar los siguientes errores relacionados con Amazon Redshift a Logs. CloudWatch

Código de error	Mensaje de error e información
`Redshift.TableNotFound`	"La tabla a la que cargar datos no se ha encontrado. Asegúrese de que la tabla especificada exista." The destination table in Amazon Redshift to which data should be copied from S3 was not found. Tenga en cuenta que Amazon Data Firehose no crea la tabla de Amazon Redshift si no existe.
`Redshift.SyntaxError`	"El comando COPY contiene un error de sintaxis. Reintente el comando."
`Redshift.AuthenticationFailed`	"Error de autenticación del nombre de usuario y la contraseña. Proporcione un nombre de usuario y contraseña válidos."
`Redshift.AccessDenied`	"Acceso denegado. Asegúrese de que la política de confianza del rol de IAM proporcionado permita a Amazon Data Firehose asumir el rol».

Código de error	Mensaje de error e información
`Redshift.S3BucketAccessDenied`	"El comando COPY no ha podido obtener acceso al bucket de S3. Ensure that the access policy for the provided IAM role allows access to the S3 bucket.”
`Redshift.DataLoadFailed`	"Se ha producido un error al cargar los datos en la tabla. Revise la tabla de sistema STL_LOAD_ERRORS para obtener más información."
`Redshift.ColumnNotFound`	"Una columna del comando COPY no existe en la tabla. Especifique un nombre de columna válido."
`Redshift.DatabaseNotFound`	“The database specified in the Amazon Redshift destination configuration or JDBC URL was not found. Especifique un nombre de base de datos válido."
`Redshift.IncorrectCopyOptions`	"Se han proporcionado opciones de COPY redundantes o en conflicto. Algunas opciones no son compatibles en determinadas combinaciones. Consulte la referencia de comandos COPY para obtener más información." Para obtener más información, consulte Comando COPY de Amazon Redshift en la Guía para desarrolladores de bases de datos de Amazon Redshift.
`Redshift.MissingColumn`	"El esquema de la tabla incluye una columna definida como NO NULL sin un valor DEFAULT, pero que no se encuentra en la lista de columnas. Exclude this column, ensure that the loaded data always provides a value for this column, or add a default value to the Amazon Redshift schema for this table.”
`Redshift.ConnectionFailed`	“The connection to the specified Amazon Redshift cluster failed. Asegúrese de que la configuración de seguridad permita las conexiones de Amazon Data Firehose, de que el clúster o la base de datos especificados en la configuración de destino de Amazon Redshift o en la URL de JDBC sean correctos y de que el clúster esté disponible».

Código de error	Mensaje de error e información
`Redshift.ColumnMismatch`	"La cantidad de jsonpath del comando COPY y la cantidad de columnas de la tabla de destino deben coincidir. Reintente el comando."
`Redshift.IncorrectOrMissingRegion`	“Amazon Redshift attempted to use the wrong region endpoint for accessing the S3 bucket. Either specify a correct region value in the COPY command options or ensure that the S3 bucket is in the same region as the Amazon Redshift database.”
`Redshift.IncorrectJsonPathsFile`	"El formato del archivo jsonpath proporcionado no es un formato JSON compatible. Reintente el comando."
`Redshift.MissingS3File`	“One or more S3 files required by Amazon Redshift have been removed from the S3 bucket. Revise las políticas del bucket de S3 para borrar cualquier eliminación automática de archivos de S3."
`Redshift.InsufficientPrivilege`	"El usuario no tiene permisos para cargar datos en la tabla. Check the Amazon Redshift user permissions for the INSERT privilege.”
`Redshift.ReadOnlyCluster`	"La consulta no se puede ejecutar porque el sistema está en modo de cambio de tamaño. Intente ejecutar la consulta de nuevo más tarde."
`Redshift.DiskFull`	"No se han podido cargar los datos ya que el disco está lleno. Increase the capacity of the Amazon Redshift cluster or delete unused data to free disk space.”
`InternalError`	"Se ha producido un error interno al intentar entregar los datos. Se volverá a intentar la entrega; si el error persiste, se informará al respecto para que se resuelva». AWS
`Redshift.ArgumentNotSupported`	“The COPY command contains unsupported options.”

Código de error	Mensaje de error e información
`Redshift.AnalyzeTableAccessDenied`	“Access denied. Copy from S3 to Redshift is failing because analyze table can only be done by table or database owner.”
`Redshift.SchemaNotFound`	«No se encontró el esquema especificado en la configuración DataTable Name de destino de Amazon Redshift. Specify a valid schema name.”
`Redshift.ColumnSpecifiedMoreThanOnce`	“There is a column specified more than once in the column list. Ensure that duplicate columns are removed.”
`Redshift.ColumnNotNullWithoutDefault`	“There is a non-null column without DEFAULT that is not included in the column list. Ensure that such columns are included in the column list.”
`Redshift.IncorrectBucketRegion`	“Redshift attempted to use a bucket in a different region from the cluster. Please specify a bucket within the same region as the cluster.”
`Redshift.S3SlowDown`	“High request rate to S3. Reduce the rate to avoid getting throttled.”
`Redshift.InvalidCopyOptionForJson`	“Please use either auto or a valid S3 path for json copyOption.”
`Redshift.InvalidCopyOptionJSONPathFormat`	“COPY failed with error \"Invalid JSONPath format. Array index is out of range.\” Please rectify the JSONPath expression.”

Código de error	Mensaje de error e información
`Redshift.InvalidCopyOptionRBACAclNotAllowed`	“COPY failed with error \"Cannot use RBAC acl framework while permission propagation is not enabled.\”
`Redshift.DiskSpaceQuotaExceeded`	“Transaction aborted due to disk space quota exceed. Free up disk space or request increased quota for the schema(s).”
`Redshift.ConnectionsLimitExceeded`	“Connection limit exceeded for user.”
`Redshift.SslNotSupported`	“The connection to the specified Amazon Redshift cluster failed because the server does not support SSL. Please check your cluster settings.”
`Redshift.HoseNotFound`	“The hose has been deleted. Please check the status of your hose.”
`Redshift.Delimiter`	“The copyOptions delimiter in the copyCommand is invalid. Ensure that it is a single character.”
`Redshift.QueryCancelled`	“The user has canceled the COPY operation.”
`Redshift.CompressionMismatch`	“Hose is configured with UNCOMPRESSED, but copyOption includes a compression format.”
`Redshift.EncryptionCredentials`	“The ENCRYPTED option requires credentials in the format: 'aws_iam_role=...;master_symmetric_key=...' or 'aws_access_key_id=...;aws_secret_access_key=...[;token=...];master_symmetric_key=...'.”

Código de error	Mensaje de error e información
`Redshift.InvalidCopyOptions`	“Invalid COPY configuration options.”
`Redshift.InvalidMessageFormat`	“Copy command contains an invalid character.”
`Redshift.TransactionIdLimitReached`	“Transaction ID limit reached.”
`Redshift.DestinationRemoved`	“Please verify that the redshift destination exists and is configured correctly in the Firehose configuration.”
`Redshift.OutOfMemory`	“The Redshift cluster is running out of memory. Please ensure the cluster has sufficient capacity.”
`Redshift.CannotForkProcess`	“The Redshift cluster is running out of memory. Please ensure the cluster has sufficient capacity.”
`Redshift.SslFailure`	“The SSL connection closed during the handshake.”
`Redshift.Resize`	“The Redshift cluster is resizing. Firehose will not be able to deliver data while the cluster is resizing.”
`Redshift.ImproperQualifiedName`	“The qualified name is improper (too many dotted names).”
`Redshift.InvalidJsonPathFormat`	“Invalid JSONPath Format.”

Código de error	Mensaje de error e información
`Redshift.TooManyConnectionsException`	“Too many connections to Redshift.”
`Redshift.PSQLException`	«PS QlException observado desde Redshift».
`Redshift.DuplicateSecondsSpecification`	“Duplicate seconds specification in date/time format.”
`Redshift.RelationCouldNotBeOpened`	“Encountered Redshift error, relation could not be opened. Check Redshift logs for the specified DB.”
`Redshift.TooManyClients`	“Encountered too many clients exception from Redshift. Revisit max connections to the database if there are multiple producers writing to it simultaneously.”

Errores en la entrega de datos de Snowflake

Firehose puede enviar los siguientes errores relacionados con Snowflake a Logs. CloudWatch

Código de error	Mensaje de error e información
`Snowflake.InvalidUrl`	«Firehose no puede conectarse a Snowflake. Asegúrese de que la URL de la cuenta esté especificada correctamente en la configuración de destino de Snowflake».
`Snowflake.InvalidUser`	«Firehose no puede conectarse a Snowflake. Asegúrese de que el usuario esté especificado correctamente en la configuración de destino de Snowflake».

Código de error	Mensaje de error e información
`Snowflake.InvalidRole`	«El rol de snowflake especificado no existe o no está autorizado. Asegúrese de que el rol esté asignado al usuario especificado»
`Snowflake.InvalidTable`	«La tabla proporcionada no existe o no está autorizada»
`Snowflake.InvalidSchema`	«El esquema proporcionado no existe o no está autorizado»
`Snowflake.InvalidDatabase`	«La base de datos proporcionada no existe o no está autorizada»
`Snowflake.InvalidPrivateKeyOrPassphrase`	«La clave privada o la frase de contraseña especificada no es válida. Tenga en cuenta que la clave privada proporcionada debe ser una clave privada PEM RSA válida»
`Snowflake.MissingColumns`	«La solicitud de inserción se rechaza porque faltan columnas en la carga útil de entrada. Asegúrese de que los valores estén especificados para todas las columnas que no admiten valores NULL»
`Snowflake.ExtraColumns`	«La solicitud de inserción se rechaza debido a que hay columnas adicionales. No se deben especificar las columnas que no estén presentes en la tabla»
`Snowflake.InvalidInput`	«La entrega ha fallado debido a un formato de entrada no válido. Asegúrese de que la carga útil de entrada proporcionada esté en el formato JSON aceptable»
`Snowflake.IncorrectValue`	«La entrega ha fallado debido a un tipo de datos incorrecto en la carga útil de entrada. Asegúrese de que los valores JSON especificados en la carga útil de entrada se ajusten al tipo de datos declarado en la definición de la tabla de Snowflake.

Errores de entrega de datos relacionados con Splunk

Amazon Data Firehose puede enviar los siguientes errores relacionados con Splunk a Logs. CloudWatch

Código de error	Mensaje de error e información
`Splunk.ProxyWithoutStickySessions`	«Si tiene un proxy (ELB u otro) entre Amazon Data Firehose y el nodo HEC, debe habilitar las sesiones fijas para admitir los ACK HEC».
`Splunk.DisabledToken`	"El token de HEC está deshabilitado. Habilite el token para permitir la entrega de datos a Splunk.".
`Splunk.InvalidToken`	"El token de HEC no es válido. Actualice Amazon Data Firehose con un token HEC válido».
`Splunk.InvalidDataFormat`	"Los datos no están en el formato correcto. Para ver cómo dar a los datos el formato correcto para puntos de enlace de HEC de eventos o sin procesar, consulte Splunk Event Data”.
`Splunk.InvalidIndex`	"El token o la entrada de HEC están configurados con un índice no válido. Compruebe la configuración del índice e inténtelo de nuevo.".
`Splunk.ServerError`	“Data delivery to Splunk failed due to a server error from the HEC node. Amazon Data Firehose volverá a intentar enviar los datos si la duración del reintento en su Amazon Data Firehose es superior a 0. Si todos los reintentos fallan, Amazon Data Firehose realiza una copia de seguridad de los datos en Amazon S3».
`Splunk.DisabledAck`	"El reconocimiento de indexadores está deshabilitado en el token de HEC. Habilite el reconocimiento de indexadores e inténtelo de nuevo. Para obtener más información, consulte Enable indexer acknowledgement.".
`Splunk.AckTimeout`	"No recibió ningún reconocimiento de parte del HEC antes de que el tiempo de espera de reconocimiento del HEC se agotara. Despite the acknowledgement timeout, it's possible the data was indexed successfu

Código de error	Mensaje de error e información
	lly in Splunk. Amazon Data Firehose realiza copias de seguridad en Amazon S3 de los datos cuyo tiempo de espera de confirmación ha expirado».
`Splunk.MaxRetriesFailed`	"Error al entregar datos a Splunk o al recibir confirmación. Compruebe el estado del HEC y vuelva a intentarlo.".
`Splunk.ConnectionTimeout`	"Se ha agotado el tiempo de espera de conexión a Splunk. This might be a transient error and the request will be retried. Amazon Data Firehose realiza copias de seguridad de los datos en Amazon S3 si todos los intentos fallan».
`Splunk.InvalidEndpoint`	"No se ha podido establecer una conexión con el punto de enlace del HEC. Asegúrese de que la URL del punto de conexión HEC es válida y se puede acceder a ella desde Amazon Data Firehose».
`Splunk.ConnectionClosed`	"No se pueden enviar datos a Splunk debido a un error de conexión. Posiblemente sea un error temporal. Aumentar la duración de los reintentos en la configuración de Amazon Data Firehose podría evitar este tipo de errores transitorios».
`Splunk.SSLUnverified`	"No se ha podido establecer una conexión con el punto de enlace del HEC. El host no coincide con el certificado proporcionadas por el homólogo. Asegúrese de que el certificado y el host son válidos."
`Splunk.SSLHandshake`	"No se ha podido establecer una conexión con el punto de enlace del HEC. Asegúrese de que el certificado y el host son válidos."
`Splunk.URLNotFound`	“The requested URL was not found on the Splunk server. Please check the Splunk cluster and make sure it is configured correctly.”
`Splunk.ServerError.ContentTooLarge`	“Data delivery to Splunk failed due to a server error with a statusCode: 413, message: the request your client sent was too large. See splunk docs to configure max_content_length.”

Código de error	Mensaje de error e información
`Splunk.IndexerBusy`	“Data delivery to Splunk failed due to a server error from the HEC node. Make sure HEC endpoint or the Elastic Load Balancer is reachable and is healthy.”
`Splunk.ConnectionRecycled`	“The connection from Firehose to Splunk has been recycled. Delivery will be retried.”
`Splunk.AcknowledgementsDisabled`	“Could not get acknowledgements on POST. Make sure that acknowled gements are enabled on HEC endpoint.”
`Splunk.InvalidHecResponseCharacter`	“Invalid characters found in HEC response, make sure to check to the service and HEC configuration.”

ElasticSearch Errores en la entrega de datos

Amazon Data Firehose puede enviar los siguientes ElasticSearch errores a CloudWatch Logs.

Código de error	Mensaje de error e información
`ES.AccessDenied`	"Acceso denegado. Ensure that the provided IAM role associated with firehose is not deleted.”
`ES.ResourceNotFound`	«El dominio de AWS Elasticsearch especificado no existe».

Errores de entrega de datos de puntos de conexión HTTPS

Amazon Data Firehose puede enviar los siguientes errores relacionados con el punto de enlace HTTP a Logs. CloudWatch Si ninguno de estos errores coincide con el problema que experimenta, el error predeterminado es el siguiente: “An internal error occurred while attempting to deliver data.

Se volverá a intentar la entrega; si el error persiste, se informará al respecto para AWS que se resuelva».

Código de error	Mensaje de error e información
`HttpEndpoint.RequestTimeout`	Se agotó el tiempo de espera de la entrega antes de recibir una respuesta y se volverá a intentar. Si el error persiste, póngase en contacto con el equipo del servicio Firehose de AWS .
`HttpEndpoint.ResponseTooLarge`	“The response received from the endpoint is too large. Contact the owner of the endpoint to resolve this issue.”
`HttpEndpoint.InvalidResponseFromDestination`	“The response received from the specified endpoint is invalid. Contact the owner of the endpoint to resolve the issue.”
`HttpEndpoint.DestinationException`	“The following response was received from the endpoint destination.”
`HttpEndpoint.ConnectionFailed`	“Unable to connect to the destination endpoint. Contact the owner of the endpoint to resolve this issue.”
`HttpEndpoint.ConnectionReset`	“Unable to maintain connection with the endpoint. Contact the owner of the endpoint to resolve this issue.”
`HttpEndpoint.ConnectionReset`	“Trouble maintaining connection with the endpoint. Please reach out to the owner of the endpoint.”
`HttpEndpoint.ResponseReason`	“The response reason phrase received from the endpoint exceed the configured limit of 64 characters.”

Código de error	Mensaje de error e información
`PhraseExceededLimit`	
`HttpEndpoint.InvalidResponseFromDestination`	“The response received from the endpoint is invalid. See Troubleshooting HTTP Endpoints in the Firehose documentation for more information. Reason: .”
`HttpEndpoint.DestinationException`	“Delivery to the endpoint was unsuccessful. See Troubleshooting HTTP Endpoints in the Firehose documentation for more information. Response received with status code .”
`HttpEndpoint.InvalidStatusCode`	“Received an invalid response status code.”
`HttpEndpoint.SSLHandshakeFailure`	“Unable to complete an SSL Handshake with the endpoint. Contact the owner of the endpoint to resolve this issue.”
`HttpEndpoint.SSLHandshakeFailure`	“Unable to complete an SSL Handshake with the endpoint. Contact the owner of the endpoint to resolve this issue.”
`HttpEndpoint.SSLFailure`	“Unable to complete TLS handshake with the endpoint. Contact the owner of the endpoint to resolve this issue.”
`HttpEndpoint.SSLHandshakeCertificatePathFailure`	“Unable to complete an SSL Handshake with the endpoint due to invalid certification path. Contact the owner of the endpoint to resolve this issue.”

Código de error	Mensaje de error e información
`HttpEndpoint.SSLHandshakeCertificatePathValidationFailure`	“Unable to complete an SSL Handshake with the endpoint due to certification path validation failure. Contact the owner of the endpoint to resolve this issue.”
`HttpEndpoint.MakeRequestFailure.IllegalUriException`	«la HttpEndpoint solicitud ha fallado debido a una entrada no válida en el URI. Please make sure all the characters in the input URI are valid.”
`HttpEndpoint.MakeRequestFailure.IllegalCharacterInHeaderValue`	«la HttpEndpoint solicitud falló debido a un error de respuesta ilegal. Illegal character '\n' in header value.”
`HttpEndpoint.IllegalResponseFailure`	«la HttpEndpoint solicitud falló debido a un error de respuesta ilegal. HTTP message must not contain more than one Content-Type header.”
`HttpEndpoint.IllegalMessageStart`	«la HttpEndpoint solicitud falló debido a un error de respuesta ilegal. Illegal HTTP message start. See Troubleshooting HTTP Endpoints in the Firehose documentation for more information.”

Errores en la entrega de datos de Amazon OpenSearch Service

Para el destino del OpenSearch servicio, Amazon Data Firehose envía los errores a los CloudWatch registros a medida que los devuelve el servicio. OpenSearch

Además de los errores que pueden provenir de OpenSearch los clústeres, es posible que se produzcan los dos errores siguientes:

- Se produce un error de autenticación/autorización al intentar entregar datos al clúster de OpenSearch servicio de destino. Esto puede ocurrir debido a problemas con los permisos o de forma intermitente cuando se modifica la configuración del dominio del OpenSearch servicio de destino de Amazon Data Firehose. Compruebe la política del clúster y los permisos del rol.
- No se pudieron enviar los datos al clúster de OpenSearch servicios de destino debido a errores de autenticación o autorización. Esto puede ocurrir debido a problemas con los permisos o de forma intermitente cuando se modifica la configuración del dominio del OpenSearch servicio de destino de Amazon Data Firehose. Compruebe la política del clúster y los permisos del rol.

Código de error	Mensaje de error e información
`OS.AccessDenied`	"Acceso denegado. Asegúrese de que la política de confianza del rol de IAM proporcionado permita a Firehose asumir el rol y que la política de acceso permita el acceso a la API de OpenSearch Amazon Service».
`OS.AccessDenied`	"Acceso denegado. Asegúrese de que la política de confianza del rol de IAM proporcionado permita a Firehose asumir el rol y que la política de acceso permita el acceso a la API de OpenSearch Amazon Service».
`OS.AccessDenied`	"Acceso denegado. Ensure that the provided IAM role associated with firehose is not deleted.”
`OS.AccessDenied`	"Acceso denegado. Ensure that the provided IAM role associated with firehose is not deleted.”
`OS.ResourceNotFound`	«El dominio de Amazon OpenSearch Service especificado no existe».
`OS.ResourceNotFound`	«El dominio de Amazon OpenSearch Service especificado no existe».
`OS.AccessDenied`	"Acceso denegado. Asegúrese de que la política de confianza del rol de IAM proporcionado permita a Firehose asumir el rol y que la política de acceso permita el acceso a la API de OpenSearch Amazon Service».

Código de error	Mensaje de error e información
OS.RequestTimeout	«Se agotó el tiempo de espera de la solicitud OpenSearch al clúster de Amazon Service o a la recogida OpenSearch sin servidor. Ensure that the cluster or collection has sufficient capacity for the current workload.”
OS.ClusterError	«El clúster OpenSearch de Amazon Service ha devuelto un error no especificado».
OS.RequestTimeout	«Se agotó el tiempo de espera de la solicitud OpenSearch al clúster de Amazon Service. Ensure that the cluster has sufficient capacity for the current workload.”
OS.ConnectionFailed	«Problemas para conectarse al clúster de Amazon OpenSearch Service o a la colección OpenSearch Serverless. Ensure that the cluster or collection is healthy and reachable.”
OS.ConnectionReset	«No se puede mantener la conexión con el clúster de Amazon OpenSearch Service o la colección OpenSearch Serverless. Contact the owner of the cluster or collection to resolve this issue.”
OS.ConnectionReset	«Problemas para mantener la conexión con el clúster de Amazon OpenSearch Service o la colección OpenSearch Serverless. Ensure that the cluster or collection is healthy and has sufficient capacity for the current workload.”
OS.ConnectionReset	«Problemas para mantener la conexión con el clúster de Amazon OpenSearch Service o la colección OpenSearch Serverless. Ensure that the cluster or collection is healthy and has sufficient capacity for the current workload.”
OS.AccessDenied	"Acceso denegado. Asegúrese de que la política de acceso del clúster de Amazon OpenSearch Service conceda acceso a la función de IAM configurada».

Código de error	Mensaje de error e información
`OS.ValidationException`	«El OpenSearch clúster devolvió un ES. ServiceException Una de las razones es que el clúster se ha actualizado a OS 2.x o superior, pero la manguera aún tiene el TypeName parámetro configurado. Actualice la configuración de la TypeName manguera configurándola en una cadena vacía o cambie el punto final al clúster, que admite el parámetro Type».
`OS.ValidationException`	“Member must satisfy regular expression pattern: [a-z][a-z0-9\\-]+.”
`OS.JsonParseException`	«El clúster OpenSearch de Amazon Service devolvió un JsonParse Exception. Ensure that the data being put is valid.”
`OS.AmazonOpenSearchServiceParseException`	«El clúster OpenSearch de Amazon Service devolvió un AmazonOpenSearchServiceParseException. Ensure that the data being put is valid.”
`OS.ExplicitIndexInBulkNotAllowed`	«Asegúrese de que rest.action.multi.allow_explicit_index esté establecido en true en el clúster de Amazon Service». OpenSearch
`OS.ClusterError`	«El clúster de Amazon OpenSearch Service o la colección OpenSearch Serverless devolvieron un error no especificado».
`OS.ClusterBlockException`	«El clúster devolvió un. ClusterBlockException It may be overloaded.”
`OS.InvalidARN`	«El ARN OpenSearch de Amazon Service proporcionado no es válido. Compruebe su DeliveryStream configuración».
`OS.MalformedData`	“One or more records are malformed. Please ensure that each record is single valid JSON object and that it does not contain newlines.”
`OS.InternalError`	“An internal error occurred when attempting to deliver data. Se volverá a intentar la entrega; si el error persiste, se informará al respecto AWS para su resolución».

Código de error	Mensaje de error e información
`OS.AliasWithMultipleIndicesNotAllowed`	“Alias has more than one indices associated with it. Ensure that the alias has only one index associated with it.”
`OS.UnsupportedVersion`	«Amazon Data Firehose no admite actualmente Amazon OpenSearch Service 6.0. Póngase en contacto con AWS Support para obtener más información».
`OS.CharConversionException`	“One or more records contained an invalid character.”
`OS.InvalidDomainNameLength`	“The domain name length is not within valid OS limits.”
`OS.VPCDomainNotSupported`	«Actualmente, no se admiten los dominios de Amazon OpenSearch Service en las VPC».
`OS.ConnectionError`	«El servidor http cerró la conexión de forma inesperada. Comprueba el estado del clúster de Amazon OpenSearch Service o de la colección OpenSearch Serverless».
`OS.LargeFieldData`	«El clúster OpenSearch de Amazon Service abortó la solicitud porque contenía un campo de datos superior al permitido».
`OS.BadGateway`	«El clúster de Amazon OpenSearch Service o la colección OpenSearch Serverless abortaron la solicitud con una respuesta: 502 Bad Gateway».
`OS.ServiceException`	«Se ha recibido un error del clúster de Amazon OpenSearch Service o de la colección OpenSearch Serverless. If the cluster or collection is behind a VPC, ensure network configuration allows connectivity.”

Código de error	Mensaje de error e información
`OS.GatewayTimeout`	«Firehose detectó errores de tiempo de espera al conectarse al clúster de Amazon OpenSearch Service o a la colección OpenSearch Serverless».
`OS.MalformedData`	«Amazon Data Firehose no admite los comandos de la API Amazon OpenSearch Service Bulk incluidos en el registro de Firehose».
`OS.ResponseEntryCountMismatch`	“The response from the Bulk API contained more entries than the number of records sent. Ensure that each record contains only one JSON object and that there are no newlines.”

Errores de invocación de Lambda

Amazon Data Firehose puede enviar los siguientes errores de invocación de Lambda a Logs. CloudWatch

Código de error	Mensaje de error e información
`Lambda.AssumeRoleAccessDenied`	"Acceso denegado. Asegúrese de que la política de confianza del rol de IAM proporcionado permita a Amazon Data Firehose asumir el rol».
`Lambda.InvokeAccessDenied`	"Acceso denegado. Ensure that the access policy allows access to the Lambda function.”
`Lambda.JsonProcessingException`	“There was an error parsing returned records from the Lambda function. Asegúrese de que los registros devueltos sigan el modelo de estado exigido por Amazon Data Firehose». Para obtener más información, consulte Modelo de estados y transformación de datos.
`Lambda.InvokeLimitExceeded`	“The Lambda concurrent execution limit is exceeded. Aumente el límite de ejecución simultánea."

Código de error	Mensaje de error e información
	Para obtener más información, consulte Límites de AWS Lambda en la Guía para desarrolladores de AWS Lambda .
`Lambda.DuplicatedRecordId`	"Se han devuelto varios registros con el mismo ID de registro. Ensure that the Lambda function returns unique record IDs for each record." Para obtener más información, consulte Modelo de estados y transformación de datos.
`Lambda.MissingRecordId`	"Uno o varios ID de registro no se devolverán. Ensure that the Lambda function returns all received record IDs." Para obtener más información, consulte Modelo de estados y transformación de datos.
`Lambda.ResourceNotFound`	"The specified Lambda function does not exist. Use otra función que exista."
`Lambda.InvalidSubnetIDException`	"The specified subnet ID in the Lambda function VPC configuration is invalid. Asegúrese de que el ID de subred sea válido."
`Lambda.InvalidSecurityGroupIDException`	"The specified security group ID in the Lambda function VPC configuration is invalid. Asegúrese de que el ID del grupo de seguridad sea válido."
`Lambda.SubnetIPAddressLimitReachedException`	«no AWS Lambda pudo configurar el acceso a la VPC para la función Lambda porque una o más subredes configuradas no tienen direcciones IP disponibles. Aumente el límite de direcciones IP." Para obtener más información, consulte Límites de Amazon VPC: VPC y subredes en la Guía del usuario de Amazon VPC.

Código de error	Mensaje de error e información
`Lambda.ENILimitReachedException`	«no AWS Lambda se pudo crear una interfaz de red elástica (ENI) en la VPC, especificada como parte de la configuración de la función Lambda, porque se alcanzó el límite de las interfaces de red. Aumente el límite de interfaces de red." Para obtener más información, consulte Límites de Amazon VPC: interfaces de red en la Guía del usuario de Amazon VPC.
`Lambda.FunctionTimedOut`	Se agotó el tiempo de espera de la función de Lambda. Aumente la configuración de tiempo de espera en la función de Lambda. Para obtener más información, consulte Configuración del tiempo de espera de la función.
`Lambda.FunctionError`	Puede deberse a uno de los siguientes errores: • La estructura de la salida no es válida. Compruebe su función y asegúrese de que la salida esté en el formato requerido. Además, asegúrese de que los registros procesados contengan un estado de resultado válido (`Dropped`, `Ok` o `ProcessingFailed`). • La función de Lambda se invocó correctamente, pero devolvió un resultado de error. • Lambda no pudo descifrar las variables de entorno porque se denegó el acceso a la clave de KMS. Compruebe la configuración de las claves de KMS de la función, así como la política de claves. Para obtener más información, consulte Troubleshooting Key Access.
`Lambda.FunctionRequestTimedOut`	Amazon Data Firehose encontró que la solicitud no se completó antes del error de configuración del tiempo de espera de la solicitud al invocar Lambda. Revise el código Lambda para comprobar si el código Lambda está destinado a ejecutarse más allá del tiempo de espera configurado. Si es así, considere la posibilidad de ajustar la configuración de Lambda, incluida la memoria y el tiempo de espera. Para obtener más información, consulte Configuración de las opciones de las funciones de Lambda.

Código de error	Mensaje de error e información
`Lambda.TargetServerFailedToRespond`	Amazon Data Firehose ha detectado un error. El servidor de destino no respondió al error al llamar al AWS servicio Lambda.
`Lambda.InvalidZipFileException`	Amazon Data Firehose se encontró InvalidZipFileException al invocar la función Lambda. Compruebe la configuración de la función de Lambda y el archivo zip de código de Lambda.
`Lambda.InternalServerError`	«Amazon Data Firehose se encontró InternalServerError al llamar al servicio Lambda AWS . Amazon Data Firehose volverá a intentar enviar los datos un número fijo de veces. Puede especificar o anular las opciones de reintento con las API `CreateDeliveryStream` o `UpdateDestination` . Si el error persiste, póngase en contacto con el equipo de soporte de AWS Lambda.
`Lambda.ServiceUnavailable`	Amazon Data Firehose se encontró ServiceUnavailableException al llamar al servicio Lambda AWS . Amazon Data Firehose volverá a intentar enviar los datos un número fijo de veces. Puede especificar o anular las opciones de reintento con las API `CreateDeliveryStream` o `UpdateDestination` . Si el error persiste, póngase en contacto con el soporte de AWS Lambda.
`Lambda.InvalidSecurityToken`	No se puede invocar la función de Lambda debido a que el token de seguridad no es válido. No se admite la invocación de Lambda entre particiones.

Código de error	Mensaje de error e información
`Lambda.InvocationFailure`	Puede deberse a uno de los siguientes errores: • Amazon Data Firehose detectó errores al llamar a AWS Lambda. La operación se reintentará y, si el error persiste, se notificará a AWS para solucionarlo. • Amazon Data Firehose encontró un KMS de InvalidStateException Lambda. Lambda no pudo descifrar las variables de entorno porque la clave KMS utilizada no es un estado válido para Descifrar. Compruebe la clave de KMS de la función de Lambda. • Amazon Data Firehose encontró una de AWS LambdaException Lambda. Lambda no pudo inicializar la imagen del contenedor proporcionada. Verifique la imagen. • Amazon Data Firehose detectó errores de tiempo de espera al llamar a Lambda. AWS El tiempo de espera máximo admitido de la función es de 5 minutos. Para obtener más información, consulte Data Transformation Execution Duration.
`Lambda.JsonMappingException`	Se ha producido un error al analizar los registros devueltos de la función de Lambda. Asegúrese de que el campo de datos esté codificado en base64.

Errores de invocación de Kinesis

Amazon Data Firehose puede enviar los siguientes errores de invocación de Kinesis a Logs. CloudWatch

Código de error	Mensaje de error e información
`Kinesis.AccessDenied`	“Access was denied when calling Kinesis. Ensure the access policy on the IAM role used allows access to the appropriate Kinesis APIs.”
`Kinesis.ResourceNotFound`	“Firehose failed to read from the stream. If the Firehose is attached with Kinesis Stream, the stream may not exist, or the shard may have been

Código de error	Mensaje de error e información
	merged or split. Si la Firehose es de DirectPut este tipo, es posible que la Firehose ya no exista».
`Kinesis.Subscriptionrequired`	“Access was denied when calling Kinesis. Asegúrese de que la función de IAM transferida para el acceso a las transmisiones de Kinesis tenga una suscripción a AWS Kinesis».
`Kinesis.Throttling`	“Throttling error encountered when calling Kinesis. Esto puede deberse a que otras aplicaciones llaman a las mismas API que la transmisión de Firehose o a que ha creado demasiadas transmisiones de Firehose con la misma transmisión de Kinesis como fuente».
`Kinesis.Throttling`	“Throttling error encountered when calling Kinesis. Esto puede deberse a que otras aplicaciones llaman a las mismas API que la transmisión de Firehose o a que ha creado demasiadas transmisiones de Firehose con la misma transmisión de Kinesis como fuente».
`Kinesis.AccessDenied`	“Access was denied when calling Kinesis. Ensure the access policy on the IAM role used allows access to the appropriate Kinesis APIs.”
`Kinesis.AccessDenied`	«Se denegó el acceso al intentar llamar a las operaciones de la API en la transmisión de Kinesis subyacente. Asegúrese de que la función de IAM esté propagada y sea válida».
`Kinesis.KMS.AccessDeniedException`	“Firehose does not have access to the KMS Key used to encrypt/decrypt the Kinesis Stream. Please grant the Firehose delivery role access to the key.”
`Kinesis.KMS.KeyDisabled`	“Firehose is unable to read from the source Kinesis Stream because the KMS key used to encrypt/decrypt it is disabled. Enable the key so that reads can proceed.”
`Kinesis.KMS.InvalidStateException`	“Firehose is unable to read from the source Kinesis Stream because the KMS key used to encrypt it is in an invalid state.”

Código de error	Mensaje de error e información
`Kinesis.KMS.NotFoundException`	“Firehose is unable to read from the source Kinesis Stream because the KMS key used to encrypt it was not found.”

Errores de invocación de Kinesis DirectPut

Amazon Data Firehose puede enviar los siguientes errores de DirectPut invocación de Kinesis a Logs. CloudWatch

Código de error	Mensaje de error e información
`Firehose.KMS.AccessDeniedException`	“Firehose does not have access to the KMS Key. Please check the key policy.”
`Firehose.KMS.InvalidStateException`	“Firehose is unable to decrypt the data because the KMS key used to encrypt it is in an invalid state.”
`Firehose.KMS.NotFoundException`	“Firehose is unable to decrypt the data because the KMS key used to encrypt it was not found.”
`Firehose.KMS.KeyDisabled`	“Firehose is unable to decrypt the data because the KMS key used to encrypt the data is disabled. Enable the key so that data delivery can proceed.”

AWS Glue Errores de invocación

Amazon Data Firehose puede enviar los siguientes errores de AWS Glue invocación a Logs. CloudWatch

Código de error	Mensaje de error e información
`DataFormatConversion.InvalidSchema`	“The schema is invalid.”
`DataFormatConversion.EntityNotFound`	“The specified table/database could not be found. Please ensure that the table/database exists and that the values provided in the schema configuration are correct, especially with regards to casing.”
`DataFormatConversion.InvalidInput`	“Could not find a matching schema from glue. Please make sure the specified database with the supplied catalog ID exists.”
`DataFormatConversion.InvalidInput`	“Could not find a matching schema from glue. Please make sure the passed ARN is in the correct format.”
`DataFormatConversion.InvalidInput`	“Could not find a matching schema from glue. Please make sure the catalogId provided is valid.”
`DataFormatConversion.InvalidVersionId`	“Could not find a matching schema from glue. Please make sure the specified version of the table exists.”
`DataFormatConversion.NonExistentColumns`	“Could not find a matching schema from glue. Please make sure the table is configured with a non-null storage descriptor containing the target columns.”

Código de error	Mensaje de error e información
`DataFormatConversion.AccessDenied`	“Access was denied when assuming role. Please ensure that the role specified in the data format conversion configuration has granted the Firehose service permission to assume it.”
`DataFormatConversion.ThrottledByGlue`	“Throttling error encountered when calling Glue. Either increase the request rate limit or reduce the current rate of calling glue through other applications.”
`DataFormatConversion.AccessDenied`	“Access was denied when calling Glue. Please ensure that the role specified in the data format conversion configuration has the necessary permissions.”
`DataFormatConversion.InvalidGlueRole`	“Invalid role. Please ensure that the role specified in the data format conversion configuration exists.”
`DataFormatConversion.InvalidGlueRole`	“The security token included in the request is invalid. Ensure that the provided IAM role associated with firehose is not deleted.”
`DataFormatConversion.GlueNotAvailableInRegion`	«AWS Glue aún no está disponible en la región que has especificado; por favor, especifica una región diferente».
`DataFormatConversion.GlueEncryptionException`	“There was an error retrieving the master key. Ensure that the key exists and has the correct access permissions.”

Código de error	Mensaje de error e información
`DataFormatConversion.SchemaValidationTimeout`	“Timed out while retrieving table from Glue. Si tienes un gran número de versiones de tablas Glue, añade el permiso «glue:GetTableVersion» (recomendado) o elimina las versiones de tablas no utilizadas. Si no tienes un gran número de mesas en Glue, ponte en contacto con AWS Support».
`DataFirehose.InternalError`	“Timed out while retrieving table from Glue. Si tienes un gran número de versiones de tablas Glue, añade el permiso «glue:GetTableVersion» (recomendado) o elimina las versiones de tablas no utilizadas. Si no tienes un gran número de mesas en Glue, ponte en contacto con AWS Support».
`DataFormatConversion.GlueEncryptionException`	“There was an error retrieving the master key. Ensure that the key exists and state is correct.”

DataFormatConversion Errores de invocación

Amazon Data Firehose puede enviar los siguientes errores de DataFormatConversion invocación a Logs. CloudWatch

Código de error	Mensaje de error e información
`DataFormatConversion.InvalidSchema`	“The schema is invalid.”
`DataFormatConversion.ValidationException`	“Column names and types must be non-empty strings.”

Código de error	Mensaje de error e información
`DataFormatConversion.ParseError`	“Encountered malformed JSON.”
`DataFormatConversion.MalformedData`	“Data does not match the schema.”
`DataFormatConversion.MalformedData`	“Length of json key must not be greater than 262144”
`DataFormatConversion.MalformedData`	“The data cannot be decoded as UTF-8.”
`DataFormatConversion.MalformedData`	“Illegal character found between tokens.”
`DataFormatConversion.InvalidTypeFormat`	“The type format is invalid. Check the type syntax.”
`DataFormatConversion.InvalidSchema`	“Invalid Schema. Asegúrese de que no haya caracteres especiales ni espacios en blanco en los nombres de las columnas».

Código de error	Mensaje de error e información
`DataFormatConversion.InvalidRecord`	“Record is not as per schema. One or more map keys were invalid for map<string,string>.”
`DataFormatConversion.MalformedData`	“The input JSON contained a primitive at the top level. The top level must be an object or array.”
`DataFormatConversion.MalformedData`	“The input JSON contained a primitive at the top level. The top level must be an object or array.”
`DataFormatConversion.MalformedData`	“The record was empty or contained only whitespace.”
`DataFormatConversion.MalformedData`	“Encountered invalid characters.”
`DataFormatConversion.MalformedData`	“Encountered invalid or unsupported timestamp format. Please see the Firehose developer guide for supported timestamp formats.”
`DataFormatConversion.MalformedData`	“A scalar type was found in the data but a complex type was specified on the schema.”

Código de error	Mensaje de error e información
`DataFormatConversion.MalformedData`	“Data does not match the schema.”
`DataFormatConversion.MalformedData`	“A scalar type was found in the data but a complex type was specified on the schema.”
`DataFormatConversion.ConversionFailureException`	"ConversionFailureException"
`DataFormatConversion.DataFormatConversionCustomerErrorException`	"DataFormatConversionCustomerErrorException"
`DataFormatConversion.DataFormatConversionCustomerErrorException`	"DataFormatConversionCustomerErrorException"

Código de error	Mensaje de error e información
`DataFormatConversion.MalformedData`	“Data does not match the schema.”
`DataFormatConversion.InvalidSchema`	“The schema is invalid.”
`DataFormatConversion.MalformedData`	“Data does not match the schema. Invalid format for one or more dates.”
`DataFormatConversion.MalformedData`	“Data contains a highly nested JSON structure that is not supported.”
`DataFormatConversion.EntityNotFound`	“The specified table/database could not be found. Please ensure that the table/database exists and that the values provided in the schema configuration are correct, especially with regards to casing.”
`DataFormatConversion.InvalidInput`	“Could not find a matching schema from glue. Please make sure the specified database with the supplied catalog ID exists.”
`DataFormatConversion.InvalidInput`	“Could not find a matching schema from glue. Please make sure the passed ARN is in the correct format.”

Código de error	Mensaje de error e información
`DataFormatConversion.InvalidInput`	“Could not find a matching schema from glue. Please make sure the catalogId provided is valid.”
`DataFormatConversion.InvalidVersionId`	“Could not find a matching schema from glue. Please make sure the specified version of the table exists.”
`DataFormatConversion.NonExistentColumns`	“Could not find a matching schema from glue. Please make sure the table is configured with a non-null storage descriptor containing the target columns.”
`DataFormatConversion.AccessDenied`	“Access was denied when assuming role. Please ensure that the role specified in the data format conversion configuration has granted the Firehose service permission to assume it.”
`DataFormatConversion.ThrottledByGlue`	“Throttling error encountered when calling Glue. Either increase the request rate limit or reduce the current rate of calling glue through other applications.”
`DataFormatConversion.AccessDenied`	“Access was denied when calling Glue. Please ensure that the role specified in the data format conversion configuration has the necessary permissions.”
`DataFormatConversion.InvalidGlueRole`	“Invalid role. Please ensure that the role specified in the data format conversion configuration exists.”

Código de error	Mensaje de error e información
`DataFormatConversion.GlueNotAvailableInRegion`	«AWS Glue aún no está disponible en la región que has especificado; por favor, especifica una región diferente».
`DataFormatConversion.GlueEncryptionException`	“There was an error retrieving the master key. Ensure that the key exists and has the correct access permissions.”
`DataFormatConversion.SchemaValidationTimeout`	“Timed out while retrieving table from Glue. Si tienes un gran número de versiones de tablas Glue, añade el permiso «glue:GetTableVersion» (recomendado) o elimina las versiones de tablas no utilizadas. Si no tienes un gran número de mesas en Glue, ponte en contacto con AWS Support».
`DataFirehose.InternalError`	“Timed out while retrieving table from Glue. Si tienes un gran número de versiones de tablas Glue, añade el permiso «glue:GetTableVersion» (recomendado) o elimina las versiones de tablas no utilizadas. Si no tienes un gran número de mesas en Glue, ponte en contacto con AWS Support».
`DataFormatConversion.MalformedData`	“One or more fields have incorrect format.”

Acceso a CloudWatch los registros de Amazon Data Firehose

Puede ver los registros de errores relacionados con un error en la entrega de datos de Amazon Data Firehose utilizando la consola Amazon Data Firehose o la consola. CloudWatch Los siguientes procedimientos explican cómo obtener acceso a los logs de errores.

Para acceder a los registros de errores mediante la consola Amazon Data Firehose

1. Inicia sesión en la consola Firehose AWS Management Console y ábrela en https://console.aws.amazon.com/firehose
2. En la barra de navegación, selecciona una AWS región.
3. Elige un nombre de transmisión de Firehose para ir a la página de detalles de la transmisión de Firehose.
4. Seleccione Error Log para ver una lista de logs de errores relacionados con los errores de entrega de datos.

Para acceder a los registros de errores mediante la consola CloudWatch

1. Abra la CloudWatch consola en https://console.aws.amazon.com/cloudwatch/.
2. Seleccione una región en la barra de navegación.
3. En el panel de navegación, elija Logs (Registros).
4. Elija un grupo y una secuencia de registros para ver una lista de los registros de errores relacionados con el error de entrega de datos.

Supervisión del estado del agente de Kinesis

Kinesis Agent publica CloudWatch métricas personalizadas con un espacio de nombres de. AWS KinesisAgent Ayuda a evaluar si el agente está en buen estado, envía los datos a Amazon Data Firehose según lo especificado y consume la cantidad adecuada de recursos de CPU y memoria del productor de datos.

Las métricas, como el número de registros y los bytes enviados, son útiles para comprender la velocidad a la que el agente envía los datos a la transmisión Firehose. Cuando estas métricas caen por debajo de los umbrales previstos en determinado porcentaje o pasan a ser cero, esto podría indicar que existen problemas de configuración, errores de red o problemas con el estado del agente. Las métricas como, por ejemplo, el consumo de CPU y memoria de host y los contadores de errores del agente indican el uso de los recursos por parte del productor y proporcionan información útil sobre posibles errores de host o de configuración. Por último, el agente también registra excepciones de servicio para ayudar a investigar los problemas del agente.

Estas métricas relacionadas con el agente se notifican en la región especificada en la opción de configuración del agente `cloudwatch.endpoint`. Para obtener más información, consulte Ajustes de la configuración del agente.

Las métricas de CloudWatch publicadas desde varios agentes de Kinesis se agregan o combinan.

Se aplica un cargo nominal por las métricas emitidas desde el agente de Kinesis, que están habilitadas de forma predeterminada. Para obtener más información, consulta los CloudWatch precios de Amazon.

Monitorear con CloudWatch

Kinesis Agent envía las siguientes métricas a. CloudWatch

Métrica	Descripción
`BytesSent`	El número de bytes enviados a la transmisión Firehose durante el período de tiempo especificado. Unidades: bytes
`RecordSendAttempts`	El número de registros que se ha intentado grabar (como primer intento o como repetición) en una llamada a `PutRecordBatch` durante el periodo de tiempo especificado. Unidades: recuento
`RecordSendErrors`	El número de registros que han devuelto un estado de error en una llamada a `PutRecordBatch` , incluidos los intentos repetidos, durante el periodo de tiempo especificado. Unidades: recuento
`ServiceErrors`	El número de llamadas a `PutRecordBatch` que ocasionaron un error de servicio (distinto de un error de limitación controlada) durante el periodo especificado. Unidades: recuento

Registro de llamadas a la API Firehose de Amazon Data con AWS CloudTrail

Amazon Data Firehose está integrado con AWS CloudTrail un servicio que proporciona un registro de las acciones realizadas por un usuario, un rol o un AWS servicio en Amazon Data Firehose. CloudTrail captura todas las llamadas a la API de Amazon Data Firehose como eventos. Las llamadas capturadas incluyen llamadas desde la consola Amazon Data Firehose y llamadas en código a las operaciones de la API Amazon Data Firehose. Si crea una ruta, puede habilitar la entrega continua de CloudTrail eventos a un bucket de Amazon S3, incluidos los eventos de Amazon Data Firehose. Si no configura una ruta, podrá ver los eventos más recientes en la CloudTrail consola, en el historial de eventos. Con la información recopilada por CloudTrail, puede determinar la solicitud que se realizó a Amazon Data Firehose, la dirección IP desde la que se realizó la solicitud, quién la hizo, cuándo se realizó y detalles adicionales.

Para obtener más información CloudTrail, incluido cómo configurarlo y habilitarlo, consulte la Guía del AWS CloudTrail usuario.

Información sobre Amazon Data Firehose en CloudTrail

CloudTrail está activado en su AWS cuenta al crear la cuenta. Cuando se produce una actividad de eventos admitida en Amazon Data Firehose, esa actividad se registra en un CloudTrail evento junto con otros eventos de AWS servicio en el historial de eventos. Puede ver, buscar y descargar los últimos eventos de la cuenta de AWS . Para obtener más información, consulte Visualización de eventos con el historial de CloudTrail eventos.

Para obtener un registro continuo de los eventos de su AWS cuenta, incluidos los eventos de Amazon Data Firehose, cree una ruta. Un rastro permite CloudTrail entregar archivos de registro a un bucket de Amazon S3. De forma predeterminada, cuando crea una ruta en la consola, la ruta se aplica a todas AWS las regiones. La ruta registra los eventos de todas las regiones de la AWS partición y envía los archivos de registro al bucket de Amazon S3 que especifique. Además, puede configurar otros AWS servicios para analizar más a fondo los datos de eventos recopilados en los CloudTrail registros y actuar en función de ellos. Para más información, consulte los siguientes temas:

- Introducción a la creación de registros de seguimiento
- CloudTrail Integraciones y servicios compatibles
- Configuración de las notificaciones de Amazon SNS para CloudTrail

- Recibir archivos de CloudTrail registro de varias regiones y recibir archivos de CloudTrail registro de varias cuentas

Amazon Data Firehose admite el registro de las siguientes acciones como eventos en archivos de CloudTrail registro:

- CreateDeliveryStream
- DeleteDeliveryStream
- DescribeDeliveryStream
- ListDeliveryStreams
- ListTagsForDeliveryStream
- TagDeliveryStream
- StartDeliveryStreamEncryption
- StopDeliveryStreamEncryption
- UntagDeliveryStream
- UpdateDestination

Cada entrada de registro o evento contiene información sobre quién generó la solicitud. La información de identidad del usuario lo ayuda a determinar lo siguiente:

- Si la solicitud se realizó con credenciales de usuario root o AWS Identity and Access Management (IAM).
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado.
- Si la solicitud la realizó otro AWS servicio.

Para obtener más información, consulte el elemento CloudTrailUserIdentity.

Ejemplo: entradas del archivo de registro de Amazon Data Firehose

Un rastro es una configuración que permite la entrega de eventos como archivos de registro a un bucket de Amazon S3 que usted especifique. CloudTrail Los archivos de registro contienen una o más entradas de registro. Un evento representa una solicitud única de cualquier fuente e incluye información sobre la acción solicitada, la fecha y la hora de la acción, los parámetros de la solicitud,

etc. CloudTrail Los archivos de registro no son un registro ordenado de las llamadas a la API pública, por lo que no aparecen en ningún orden específico.

En el siguiente ejemplo, se muestra una entrada de CloudTrail registro que muestra las `DeleteDeliveryStream` acciones `CreateDeliveryStream` `DescribeDeliveryStream` `ListDeliveryStreamsUpdateDestination`,, y.

```
{
  "Records":[
        {
            "eventVersion":"1.02",
            "userIdentity":{
                "type":"IAMUser",
                "principalId":"AKIAIOSFODNN7EXAMPLE",
                "arn":"arn:aws:iam::111122223333:user/CloudTrail_Test_User",
                "accountId":"111122223333",
                "accessKeyId":"AKIAI44QH8DHBEXAMPLE",
                "userName":"CloudTrail_Test_User"
            },
            "eventTime":"2016-02-24T18:08:22Z",
            "eventSource":"firehose.amazonaws.com",
            "eventName":"CreateDeliveryStream",
            "awsRegion":"us-east-1",
            "sourceIPAddress":"127.0.0.1",
            "userAgent":"aws-internal/3",
            "requestParameters":{
                "deliveryStreamName":"TestRedshiftStream",
                "redshiftDestinationConfiguration":{
                "s3Configuration":{
                    "compressionFormat":"GZIP",
                    "prefix":"prefix",
                    "bucketARN":"arn:aws:s3:::firehose-cloudtrail-test-bucket",
                    "roleARN":"arn:aws:iam::111122223333:role/Firehose",
                    "bufferingHints":{
                        "sizeInMBs":3,
                        "intervalInSeconds":900
                    },
                    "encryptionConfiguration":{
                        "kMSEncryptionConfig":{
                            "aWSKMSKeyARN":"arn:aws:kms:us-east-1:key"
                        }
                    }
                },
```

```
                "clusterJDBCURL":"jdbc:redshift://example.abc123.us-
west-2.redshift.amazonaws.com:5439/dev",
                "copyCommand":{
                    "copyOptions":"copyOptions",
                    "dataTableName":"dataTable"
                },
                "password":"",
                "username":"",
                "roleARN":"arn:aws:iam::111122223333:role/Firehose"
            }
        },
        "responseElements":{
            "deliveryStreamARN":"arn:aws:firehose:us-
east-1:111122223333:deliverystream/TestRedshiftStream"
        },
        "requestID":"958abf6a-db21-11e5-bb88-91ae9617edf5",
        "eventID":"875d2d68-476c-4ad5-bbc6-d02872cfc884",
        "eventType":"AwsApiCall",
        "recipientAccountId":"111122223333"
    },
    {
        "eventVersion":"1.02",
        "userIdentity":{
            "type":"IAMUser",
            "principalId":"AKIAIOSFODNN7EXAMPLE",
            "arn":"arn:aws:iam::111122223333:user/CloudTrail_Test_User",
            "accountId":"111122223333",
            "accessKeyId":"AKIAI44QH8DHBEXAMPLE",
            "userName":"CloudTrail_Test_User"
        },
        "eventTime":"2016-02-24T18:08:54Z",
        "eventSource":"firehose.amazonaws.com",
        "eventName":"DescribeDeliveryStream",
        "awsRegion":"us-east-1",
        "sourceIPAddress":"127.0.0.1",
        "userAgent":"aws-internal/3",
        "requestParameters":{
            "deliveryStreamName":"TestRedshiftStream"
        },
        "responseElements":null,
        "requestID":"aa6ea5ed-db21-11e5-bb88-91ae9617edf5",
        "eventID":"d9b285d8-d690-4d5c-b9fe-d1ad5ab03f14",
        "eventType":"AwsApiCall",
        "recipientAccountId":"111122223333"
```

```
    },
    {
        "eventVersion":"1.02",
        "userIdentity":{
            "type":"IAMUser",
            "principalId":"AKIAIOSFODNN7EXAMPLE",
            "arn":"arn:aws:iam::111122223333:user/CloudTrail_Test_User",
            "accountId":"111122223333",
            "accessKeyId":"AKIAI44QH8DHBEXAMPLE",
            "userName":"CloudTrail_Test_User"
        },
        "eventTime":"2016-02-24T18:10:00Z",
        "eventSource":"firehose.amazonaws.com",
        "eventName":"ListDeliveryStreams",
        "awsRegion":"us-east-1",
        "sourceIPAddress":"127.0.0.1",
        "userAgent":"aws-internal/3",
        "requestParameters":{
            "limit":10
        },
        "responseElements":null,
        "requestID":"d1bf7f86-db21-11e5-bb88-91ae9617edf5",
        "eventID":"67f63c74-4335-48c0-9004-4ba35ce00128",
        "eventType":"AwsApiCall",
        "recipientAccountId":"111122223333"
    },
    {
        "eventVersion":"1.02",
        "userIdentity":{
            "type":"IAMUser",
            "principalId":"AKIAIOSFODNN7EXAMPLE",
            "arn":"arn:aws:iam::111122223333:user/CloudTrail_Test_User",
            "accountId":"111122223333",
            "accessKeyId":"AKIAI44QH8DHBEXAMPLE",
            "userName":"CloudTrail_Test_User"
        },
        "eventTime":"2016-02-24T18:10:09Z",
        "eventSource":"firehose.amazonaws.com",
        "eventName":"UpdateDestination",
        "awsRegion":"us-east-1",
        "sourceIPAddress":"127.0.0.1",
        "userAgent":"aws-internal/3",
        "requestParameters":{
            "destinationId":"destinationId-000000000001",
```

```
            "deliveryStreamName":"TestRedshiftStream",
            "currentDeliveryStreamVersionId":"1",
            "redshiftDestinationUpdate":{
                "roleARN":"arn:aws:iam::111122223333:role/Firehose",
                "clusterJDBCURL":"jdbc:redshift://example.abc123.us-
west-2.redshift.amazonaws.com:5439/dev",
                "password":"",
                "username":"",
                "copyCommand":{
                    "copyOptions":"copyOptions",
                    "dataTableName":"dataTable"
                },
                "s3Update":{
                    "bucketARN":"arn:aws:s3:::firehose-cloudtrail-test-bucket-update",
                    "roleARN":"arn:aws:iam::111122223333:role/Firehose",
                    "compressionFormat":"GZIP",
                    "bufferingHints":{
                        "sizeInMBs":3,
                        "intervalInSeconds":900
                    },
                    "encryptionConfiguration":{
                        "kMSEncryptionConfig":{
                            "aWSKMSKeyARN":"arn:aws:kms:us-east-1:key"
                        }
                    },
                    "prefix":"arn:aws:s3:::firehose-cloudtrail-test-bucket"
                }
            }
        },
        "responseElements":null,
        "requestID":"d549428d-db21-11e5-bb88-91ae9617edf5",
        "eventID":"1cb21e0b-416a-415d-bbf9-769b152a6585",
        "eventType":"AwsApiCall",
        "recipientAccountId":"111122223333"
    },
    {
        "eventVersion":"1.02",
        "userIdentity":{
            "type":"IAMUser",
            "principalId":"AKIAIOSFODNN7EXAMPLE",
            "arn":"arn:aws:iam::111122223333:user/CloudTrail_Test_User",
            "accountId":"111122223333",
            "accessKeyId":"AKIAI44QH8DHBEXAMPLE",
            "userName":"CloudTrail_Test_User"
```

```
        },
        "eventTime":"2016-02-24T18:10:12Z",
        "eventSource":"firehose.amazonaws.com",
        "eventName":"DeleteDeliveryStream",
        "awsRegion":"us-east-1",
        "sourceIPAddress":"127.0.0.1",
        "userAgent":"aws-internal/3",
        "requestParameters":{
            "deliveryStreamName":"TestRedshiftStream"
        },
        "responseElements":null,
        "requestID":"d85968c1-db21-11e5-bb88-91ae9617edf5",
        "eventID":"dd46bb98-b4e9-42ff-a6af-32d57e636ad1",
        "eventType":"AwsApiCall",
        "recipientAccountId":"111122223333"
    }
  ]
}
```

Prefijos personalizados para los objetos de Amazon S3

Los objetos entregados a Amazon S3 siguen el formato de nombre de <evaluated prefix><suffix>. Puede especificar un prefijo personalizado que incluya expresiones que se evalúan en tiempo de ejecución. El prefijo personalizado que especifique anulará el prefijo predeterminado de. `YYYY/MM/dd/HH`

Puede utilizar expresiones de las siguientes formas en el prefijo personalizado: `!{namespace:value}`, donde `namespace` puede ser una de estas opciones, tal como se explica en las secciones siguientes.

- `firehose`
- `timestamp`
- `partitionKeyFromQuery`
- `partitionKeyFromLambda`

Si un prefijo termina por una barra inclinada, aparece como una carpeta en el bucket de Amazon S3. Para obtener más información, consulte el formato de nombre de objeto de Amazon S3 en la Amazon Data FirehoseDeveloper Guide.

El espacio de nombres **`timestamp`**

Los valores válidos para este espacio de nombres son cadenas que son cadenas Java DateTimeFormatter válidas. Por ejemplo, en el año 2018, la expresión `!{timestamp:yyyy}` se evalúa como `2018`.

Al evaluar las marcas de tiempo, Firehose utiliza la marca de tiempo de llegada aproximada del registro más antiguo contenido en el objeto de Amazon S3 que se está grabando.

De forma predeterminada, la marca de tiempo está en UTC. Sin embargo, puedes especificar la zona horaria que prefieras. Por ejemplo, puede configurar la zona horaria en Asia/Tokio en la configuración de parámetros de la AWS Management Console API (CustomTimeZone) si desea utilizar la hora estándar de Japón en lugar de la UTC. Para ver la lista de zonas horarias compatibles, consulte Formato de nombre de objeto de Amazon S3.

Si se utiliza el espacio de nombres `timestamp` más de una vez en la misma expresión de prefijo, cada instancia se evalúa como el mismo instante en el tiempo.

El espacio de nombres **`firehose`**

Hay dos valores que se pueden utilizar con este espacio de nombres: `error-output-type` y `random-string`. En la tabla siguiente, se explica cómo hacerlo.

Los valores del espacio de nombres **`firehose`**

Conversion (Conversión)	Descripción	Ejemplo de entrada	Ejemplo de resultado	Notas
`error-output-type`	Se evalúa como una de las siguientes cadenas, según la configuración de la transmisión de Firehose y el motivo del error: {processing-failed, -failed, splunk-failed AmazonOpenSearchService,}. format-conversion-failed http-endpoint-failed Si lo utiliza más de una vez en la misma expresión, cada instancia se evalúa como la misma cadena de error.	`myPrefix/result=!{firehose:error-output-type}/!{timestamp:yyyy/MM/dd}`	`myPrefix/result=processing-failed/2018/08/03`	El error-output-type valor solo se puede usar en el campo. ErrorOutputPrefix

Conversion (Conversión)	Descripción	Ejemplo de entrada	Ejemplo de resultado	Notas
`random-string`	Se evalúa como una cadena aleatoria de 11 caracteres. Si lo utiliza más de una vez en la misma expresión, cada instancia se evalúa como una cadena aleatoria nueva.	`myPrefix/!{firehose:random-string}/`	`myPrefix/046b6c7f-0b/`	Puede utilizarlo con ambos tipos de prefijos. Puede colocarlo al principio de la cadena de formato para obtener un prefijo aleatorio, que a veces es necesario para alcanzar un rendimiento extremadamente alto con Amazon S3.

Espacios de nombres **`partitionKeyFromLambda`** y **`partitionKeyFromQuery`**

En el caso del particionamiento dinámico, debe usar el siguiente formato de expresión en el prefijo de bucket de S3: `!{namespace:value}`, donde el espacio de nombres puede ser `partitionKeyFromQuery`, `partitionKeyFromLambda` o ambos. Si utiliza el análisis en línea para crear las claves de particionamiento para sus datos de origen, debe especificar un valor de prefijo de bucket de S3 que conste de expresiones especificadas en el siguiente formato: `"partitionKeyFromQuery:keyID"`. Si utiliza una función de AWS Lambda para crear claves de particionamiento para sus datos de origen, debe especificar un valor de prefijo de bucket de S3 que conste de expresiones especificadas en el siguiente formato: `"partitionKeyFromLambda:keyID"`. Para obtener más información, consulte la sección «Elija Amazon S3 como destino» en Creación de una transmisión de Amazon Firehose.

Reglas semánticas

Las siguientes reglas se aplican a las expresiones `Prefix` y `ErrorOutputPrefix`.

- En el espacio de nombres `timestamp`, se evalúa cualquier carácter que no esté entre comillas simples. En otras palabras, cualquier cadena encerrada entre comillas simples en el campo de valor se interpreta literalmente.
- Si especificas un prefijo que no contiene una expresión de espacio de nombres de marca de tiempo, Firehose anexa la expresión al valor del `!{timestamp:yyyy/MM/dd/HH/}` campo. `Prefix`
- La secuencia `!{` solo puede aparecer en expresiones `!{namespace:`*`value`*`}`.
- `ErrorOutputPrefix` únicamente puede ser null si `Prefix` no contiene ninguna expresión. En este caso, `Prefix` evalúa a `<specified-prefix>yyyy/MM/DDD/HH/` y `ErrorOutputPrefix` evalúa a `<specified-prefix><error-output-type>YYYY/MM/DDD/HH/`. `DDD` representa el día del año.
- Si especifica una expresión para `ErrorOutputPrefix`, debe incluir al menos una instancia de `!{firehose:error-output-type}`.
- `Prefix` no puede contener `!{firehose:error-output-type}`.
- Una vez evaluados, ni `Prefix` ni `ErrorOutputPrefix` pueden tener una longitud superior a 512 caracteres.
- Si el destino es Amazon Redshift, `Prefix` no debe contener expresiones y `ErrorOutputPrefix` debe ser null.
- Cuando el destino es Amazon OpenSearch Service o Splunk y no `ErrorOutputPrefix` se especifica ningún, Firehose utiliza `Prefix` el campo para los registros fallidos.
- Cuando el destino es Amazon S3, `Prefix` y `ErrorOutputPrefix` en la configuración de destino de Amazon S3 se utilizan para los registros correctos y los registros con errores, respectivamente. Si utiliza AWS CLI o la API, puede utilizar `ExtendedS3DestinationConfiguration` para especificar una configuración de copia de seguridad de Amazon S3 con sus propios valores de `Prefix` y `ErrorOutputPrefix`.
- Al usar Amazon S3 AWS Management Console y establecer el destino en Amazon S3, Firehose usa `Prefix` y `ErrorOutputPrefix` en la configuración de destino para los registros correctos y los registros fallidos, respectivamente. Si especificas un prefijo pero no un prefijo de error, Firehose establece automáticamente el prefijo de error en. `!{firehose:error-output-type}/`

- Cuando lo usas `ExtendedS3DestinationConfiguration` con la AWS CLI API o AWS CloudFormation, si especificas una`S3BackupConfiguration`, Firehose no proporciona un valor predeterminado. `ErrorOutputPrefix`
- No puedes usar `partitionKeyFromQuery` espacios de nombres `partitionKeyFromLambda` y al crear expresiones. ErrorOutputPrefix

Ejemplos de prefijos

Ejemplos de **`Prefix`** y **`ErrorOutputPrefix`**

Entrada	Prefijo evaluado (a las 10:30 h UTC del 27 de agosto de 2018)

`Prefix`: sin especificar
 	
`ErrorOutputPrefix` : `myFirehoseFailures/!{firehose:error-output-type}/`
	`Prefix`: `2018/08/27/10`
	
	`ErrorOutputPrefix` : `myFirehoseFailures/processing-failed/`

`Prefix`: `!{timestamp:yyyy/MM/dd}` `ErrorOutputPrefix` : sin especificar	Entrada no válida: `ErrorOutputPrefix` no puede ser null si Prefix contiene expresiones

`Prefix`: `myFirehose/DeliveredYear=!{timestamp:yyyy}/anyMonth/rand=!{firehose:random-string}`
 	
`ErrorOutputPrefix` : `myFirehoseFailures/!{firehose:error-output-type}/!{timestamp:yyyy}/anyMonth/!{timestamp:dd}`
	`Prefix`: `myFirehose/DeliveredYear=2018/anyMonth/rand=5abf82daaa5`
	
	`ErrorOutputPrefix` : `myFirehoseFailures/processing-failed/2018/anyMonth/10`

`Prefix`: `myPrefix/year=!{timestamp:yyyy}/month=!{timestamp:MM}/day=!{timestamp:dd}/hour=!{timestamp:HH}/`
 	
`ErrorOutputPrefix` : `myErrorPrefix/year=!{timestamp:yyyy}/month=!`
	`Prefix`: `myPrefix/year=2018/month=07/day=06/hour=23/`
	
	`ErrorOutputPrefix` : `myErrorPrefix/year=2018/month=07/day=06/hour=23/processing-failed`

Entrada	Prefijo evaluado (a las 10:30 h UTC del 27 de agosto de 2018)
`{timestamp:MM}/day=!{timesta mp:dd}/hour=!{timestamp:HH}/! {firehose:error-output-type}`	

`Prefix`: `myFirehosePrefix/`
 	
`ErrorOutputPrefix` : sin especificar
	`Prefix`: `myFirehosePrefix/2 018/08/27/`
	
	`ErrorOutputPrefix` : `myFirehos ePrefix/processing-failed/2 018/08/27/`

Uso de Amazon Data Firehose con AWS PrivateLink

Puntos de enlace de VPC de interfaz ()AWS PrivateLink para Amazon Data Firehose

Puede utilizar un punto de enlace de VPC de interfaz para evitar que el tráfico entre su Amazon VPC y Amazon Data Firehose salga de la red de Amazon. Los puntos finales de la interfaz VPC no requieren una puerta de enlace a Internet, un dispositivo NAT, una conexión VPN o una conexión. AWS Direct Connect Los puntos de enlace de la interfaz VPC funcionan con una AWS tecnología que permite la comunicación privada entre AWS servicios mediante una interfaz de red elástica con direcciones IP privadas en su Amazon VPC. AWS PrivateLink Para obtener más información, consulte Amazon Virtual Private Cloud.

Uso de puntos de enlace de VPC de interfaz ()AWS PrivateLink para Amazon Data Firehose

Para empezar, cree un punto de enlace de VPC de interfaz para que el tráfico de Amazon Data Firehose procedente de sus recursos de Amazon VPC comience a fluir a través del punto de enlace de VPC de interfaz. Al crear un punto de conexión, puede adjuntarle una política de punto final que controle el acceso a Amazon Data Firehose. Para obtener más información sobre el uso de políticas para controlar el acceso desde un punto de enlace de VPC a Amazon Data Firehose, consulte Control del acceso a los servicios con puntos de enlace de VPC.

El siguiente ejemplo muestra cómo puede configurar una AWS Lambda función en una VPC y crear un punto de enlace de VPC para permitir que la función se comunique de forma segura con el servicio Amazon Data Firehose. En este ejemplo, utiliza una política que permite a la función Lambda enumerar las transmisiones de Firehose de la región actual, pero no describir ninguna transmisión de Firehose.

Crear un punto de conexión de VPC

1. Inicie sesión en la consola de Amazon VPC AWS Management Console y ábrala en https://console.aws.amazon.com/vpc/.
2. En el panel de la VPC, elija Endpoints (Puntos de enlace).
3. Seleccione Crear punto de conexión.

4. En la lista de nombres de servicio, elija `com.amazonaws.`*`your_region`*`.kinesis-firehose`.
5. Elija la VPC y una o varias subredes en las que se debe crear el punto de enlace.
6. Elija uno o varios grupos de seguridad para asociar con el punto de enlace.
7. En Policy (Política), elija Custom (Personalizada) y pegue la siguiente política:

```
{
    "Statement": [
        {
            "Sid": "Allow-only-specific-PrivateAPIs",
            "Principal": "*",
            "Action": [
                "firehose:ListDeliveryStreams"
            ],
            "Effect": "Allow",
            "Resource": [
                "*"
            ]
        },
        {
            "Sid": "Allow-only-specific-PrivateAPIs",
            "Principal": "*",
            "Action": [
                "firehose:DescribeDeliveryStream"
            ],
            "Effect": "Deny",
            "Resource": [
                "*"
            ]
        }
    ]
}
```

8. Seleccione Crear punto de conexión.

Creación de un rol de IAM que se usará con la función de Lambda

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de la izquierda, elija Roles y, a continuación, seleccione Crear rol.

3. En Seleccionar tipo de entidad de confianza, deje la selección predeterminada Servicio de AWS .
4. En Choose the service that will use this role (Elegir el servicio que usará este rol), elija Lambda.
5. Elija Next: Permissions (Siguiente: permisos).
6. En la lista de políticas, busque y añada las dos políticas denominadas `AWS LambdaVPCAccessExecutionRole` y `AmazonDataFirehoseReadOnlyAccess`.

> **Important**
>
> A continuación se muestra un ejemplo: Es posible que necesite políticas más estrictas para su entorno de producción.

7. Elija Siguiente: etiquetas. Para este ejercicio, no es necesario que añada ninguna etiqueta. Elija Siguiente: Revisar.
8. Ingrese un nombre para el rol y, a continuación, seleccione Crear rol.

Creación de una función de Lambda en la VPC

1. Abra la AWS Lambda consola en https://console.aws.amazon.com/lambda/.
2. Elija Crear función.
3. Elija Crear desde cero.
4. Introduzca un nombre para la función y, a continuación, establezca Runtime en `Python 3.9` o en una posición superior.
5. En Permissions (Permisos), expanda Choose or create an execution role (Seleccionar o crear un rol de ejecución).
6. En la lista Execution role (Rol de ejecución), elija Use an existing role (Usar un rol existente).
7. En la lista Existing role (Rol existente), elija el rol que creó antes.
8. Elija Crear función.
9. En Function code (Código de la función), pegue el siguiente código.

```
import json
import boto3
import os
from botocore.exceptions import ClientError
```

```
   def lambda_handler(event, context):
       REGION = os.environ['AWS_REGION']
       client = boto3.client(
           'firehose',
           REGION

       )
       print("Calling list_delivery_streams with ListDeliveryStreams allowed
policy.")
       delivery_stream_request = client.list_delivery_streams()
       print("Successfully returned list_delivery_streams request %s." % (
           delivery_stream_request
       ))
       describe_access_denied = False
       try:
           print("Calling describe_delivery_stream with DescribeDeliveryStream
denied policy.")
           delivery_stream_info =
client.describe_delivery_stream(DeliveryStreamName='test-describe-denied')
       except ClientError as e:
           error_code = e.response['Error']['Code']
           print ("Caught %s." % (error_code))
           if error_code == 'AccessDeniedException':
               describe_access_denied = True

       if not describe_access_denied:
           raise
       else:
           print("Access denied test succeeded.")
```

10. En Basic settings (Configuración básica), establezca el tiempo de espera en 1 minuto.
11. En Network (Red), elija la VPC en la que creó el punto de enlace y, a continuación, elija las subredes y el grupo de seguridad con los que asoció el punto de enlace cuando lo creó.
12. Cerca de la parte superior de la página, elija Guardar.
13. Seleccione Probar.
14. Ingrese un nombre de evento y, a continuación, elija Crear.
15. Elija Test (Probar) de nuevo. Esto hace que la función se ejecute. Cuando aparezca el resultado de la ejecución, expanda Details (Detalles) y compare la salida de registro con el código de la función. Los resultados satisfactorios muestran una lista de los arroyos Firehose de la región, así como el siguiente resultado:

```
Calling describe_delivery_stream.

AccessDeniedException

Access denied test succeeded.
```

Disponibilidad

Los puntos de enlace de la VPC de tipo interfaz se admiten actualmente en las regiones siguientes:

- US East (Ohio)
- Este de EE. UU. (Norte de Virginia)
- Oeste de EE. UU. (Norte de California)
- Oeste de EE. UU. (Oregón)
- Asia-Pacífico (Bombay)
- Asia-Pacífico (Seúl)
- Asia-Pacífico (Singapur)
- Asia-Pacífico (Sídney)
- Asia-Pacífico (Tokio)
- Asia-Pacífico (Hong Kong)
- Canadá (centro)
- Oeste de Canadá (Calgary
- China (Pekín)
- China (Ningxia)
- Europa (Fráncfort)
- Europa (Irlanda)
- Europa (Londres)
- Europa (París)
- América del Sur (São Paulo)
- AWS GovCloud (Este de EE. UU.)
- AWS GovCloud (Estados Unidos-Oeste)
- Europa (España)

- Medio Oriente (EAU)
- Asia-Pacífico (Yakarta)
- Asia-Pacífico (Osaka)
- Israel (Tel Aviv)

Cómo etiquetar las transmisiones de Firehose en Amazon Data Firehose

Puede asignar sus propios metadatos a las transmisiones de Firehose que cree en Amazon Data Firehose en forma de etiquetas. Una etiqueta es un par clave-valor definido por el usuario para un flujo. El uso de etiquetas es una forma sencilla pero eficaz de gestionar AWS los recursos y organizar los datos, incluidos los datos de facturación.

Temas

- Conceptos básicos de etiquetas
- Seguimiento de costos utilizando el etiquetado
- Restricciones de las etiquetas
- Etiquetado de transmisiones de Firehose mediante la API Firehose de Amazon Data

Conceptos básicos de etiquetas

Puede utilizar la API Amazon Data Firehose para realizar las siguientes tareas:

- Añade etiquetas a una transmisión de Firehose.
- Haz una lista de las etiquetas de tus transmisiones de Firehose.
- Quita las etiquetas de una transmisión de Firehose.

Puedes usar etiquetas para categorizar tus transmisiones de Firehose. Por ejemplo, puede clasificar las transmisiones Firehose por propósito, propietario o entorno. Dado que define la clave y el valor de cada etiqueta, puede crear un conjunto de categorías personalizadas para satisfacer sus necesidades específicas. Por ejemplo, puedes definir un conjunto de etiquetas que te ayuden a rastrear las transmisiones de Firehose por propietario y aplicación asociada.

A continuación, se muestran varios ejemplos de etiquetas:

- `Project: Project name`
- `Owner: Name`
- `Purpose: Load testing`

- `Application: Application name`
- `Environment: Production`

Si especifica etiquetas en la `CreateDeliveryStream` acción, Amazon Data Firehose realiza una autorización adicional sobre la `firehose:TagDeliveryStream` acción para comprobar si los usuarios tienen permisos para crear etiquetas. Si no concedes este permiso, las solicitudes para crear nuevas transmisiones de Firehose con etiquetas de recursos de IAM fallarán con un resultado `AccessDeniedException` como el siguiente.

```
AccessDeniedException
User: arn:aws:sts::x:assumed-role/x/x is not authorized to perform:
 firehose:TagDeliveryStream on resource: arn:aws:firehose:us-east-1:x:deliverystream/x
 with an explicit deny in an identity-based policy.
```

El siguiente ejemplo muestra una política que permite a los usuarios crear una transmisión Firehose y aplicar etiquetas.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "firehose:CreateDeliveryStream",
            "Resource": "*",
            }
        },
        {
            "Effect": "Allow",
            "Action": "firehose:TagDeliveryStream",
            "Resource": "*",
            }
        }
    ]
}
```

Seguimiento de costos utilizando el etiquetado

Puede usar etiquetas para categorizar y realizar un seguimiento de sus AWS costos. Cuando aplicas etiquetas a tus AWS recursos, incluidas las transmisiones de Firehose, tu informe de asignación de

AWS costos incluye el uso y los costos agregados por etiquetas. Si aplica etiquetas que representen categorías de negocio (por ejemplo, centros de costos, nombres de aplicaciones o propietarios), puede organizar los costos entre diferentes servicios. Para obtener más información, consulte Utilizar etiquetas de asignación de costos para informes de facturación personalizados en la Guía del usuario de AWS Billing .

Restricciones de las etiquetas

Las siguientes restricciones se aplican a las etiquetas de Amazon Data Firehose.

Restricciones básicas

- El número máximo de etiquetas por recurso (flujo) es 50.
- Las claves y los valores de las etiquetas distinguen entre mayúsculas y minúsculas.
- No se pueden cambiar ni editar etiquetas de un flujo eliminado.

Restricciones de clave de etiqueta

- Cada clave de etiqueta debe ser única. Si agrega una etiqueta con una clave que ya está en uso, la nueva etiqueta sobrescribe el par clave-valor existente.
- Una clave de etiqueta no puede comenzar por `aws:` porque este prefijo está reservado para su utilización por AWS. AWS crea etiquetas cuyo nombre comienza por este prefijo por usted, pero usted no puede editarlas ni eliminarlas.
- Las claves de etiqueta deben tener entre 1 y 128 caracteres Unicode de longitud.
- Las claves de etiquetas deben constar de los siguientes caracteres: letras Unicode, números, espacios en blanco y los siguientes caracteres especiales: `_ . / = + - @`.

Restricciones de valor de etiqueta

- Los valores de etiqueta deben tener entre 0 y 255 caracteres Unicode de longitud.
- Los valores de etiqueta pueden estar en blanco. De lo contrario, deben constar de los siguientes caracteres: letras Unicode, números, espacios en blanco y cualquiera de los siguientes caracteres especiales: `_ . / = + - @`.

Etiquetado de transmisiones de Firehose mediante la API Firehose de Amazon Data

Puedes especificar etiquetas al invocar CreateDeliveryStreampara crear una nueva transmisión de Firehose. Para las transmisiones de Firehose existentes, puedes añadir, enumerar y eliminar etiquetas mediante las tres operaciones siguientes:

- TagDeliveryStream
- ListTagsForDeliveryStream
- UntagDeliveryStream

Tutorial: Ingiera los registros de flujo de VPC en Splunk mediante Amazon Data Firehose

Para ver un tutorial, consulte Ingesta de registros de flujo de VPC en Splunk mediante Amazon Data Firehose.

Solución de problemas de Amazon Data Firehose

Si Firehose detecta errores durante la entrega o el procesamiento de los datos, lo vuelve a intentar hasta que caduque el tiempo de reintento configurado. Si la duración del reintento finaliza antes de que los datos se entreguen correctamente, Firehose hace una copia de seguridad de los datos en el depósito de respaldo de S3 configurado. Si el destino es Amazon S3 y se produce un error en la entrega o si se produce un error en la entrega al bucket S3 de respaldo, Firehose vuelve a intentarlo hasta que finalice el período de retención. En el `DirectPut` caso de las transmisiones de Firehose, Firehose conserva los registros durante 24 horas. En el caso de una transmisión de Firehose cuya fuente de datos sea una transmisión de datos de Kinesis, puede cambiar el período de retención tal y como se describe en Cambiar el período de retención de datos.

Si la fuente de datos es una transmisión de datos de Kinesis, Firehose vuelve a intentar las siguientes operaciones de forma indefinida:, y. `DescribeStream` `GetRecords` `GetShardIterator`

Si la transmisión de Firehose lo usa`DirectPut`, comprueba las `IncomingRecords` métricas `IncomingBytes` y para ver si hay tráfico entrante. Si utiliza `PutRecord` o `PutRecordBatch`, asegúrese de que detecta las excepciones y vuelva a intentarlo. Le recomendamos que utilice una política de reintentos con retardo exponencial con fluctuaciones y varios reintentos. Además, si utilizas la `PutRecordBatch` API, asegúrate de que el código compruebe el valor de FailedPutCounten la respuesta incluso cuando la llamada a la API se realice correctamente.

Si la transmisión de Firehose utiliza una transmisión de datos de Kinesis como fuente, compruebe `IncomingRecords` las métricas `IncomingBytes` y de la transmisión de datos de origen. Además, asegúrate de que las `DataReadFromKinesisStream.Records` métricas `DataReadFromKinesisStream.Bytes` y se emitan para la transmisión de Firehose.

Para obtener información sobre el seguimiento de los errores de entrega mediante el uso de CloudWatch, consultethe section called “Monitorización con registros CloudWatch ”.

Problemas comunes

Estos son algunos de los problemas más comunes y cómo puedes resolverlos.

- La transmisión de Firehose no está disponible como destino para CloudWatch registros, CloudWatch eventos o acciones de AWS IoT: algunos AWS servicios solo pueden enviar mensajes

y eventos a una transmisión de Firehose que se encuentre en la misma. Región de AWS Verifica que tu transmisión de Firehose esté ubicada en la misma región que el resto de tus servicios.

- No hay datos en el destino a pesar de tener buenas métricas: si no hay problemas de ingesta de datos y las métricas emitidas para la transmisión de Firehose son buenas, pero no ves los datos en el destino, comprueba la lógica del lector. Asegúrese de que su lector esté analizando correctamente todos los datos.

Solución de problemas de Amazon S3

Compruebe lo siguiente si los datos no se entregan en su bucket de Amazon Simple Storage Service (Amazon S3).

- Comprueba la Firehose `IncomingBytes` y `IncomingRecords` las métricas para asegurarte de que los datos se envíen correctamente a tu transmisión de Firehose. Para obtener más información, consulte Supervisión de Amazon Data Firehose mediante métricas CloudWatch .
- Si la transformación de datos con Lambda está habilitada, compruebe la `ExecuteProcessingSuccess` métrica Firehose para asegurarse de que Firehose ha intentado invocar la función Lambda. Para obtener más información, consulte Supervisión de Amazon Data Firehose mediante métricas CloudWatch .
- Comprueba la `DeliveryToS3.Success` métrica de Firehose para asegurarte de que Firehose ha intentado colocar datos en tu bucket de Amazon S3. Para obtener más información, consulte Supervisión de Amazon Data Firehose mediante métricas CloudWatch .
- Habilite el registro de errores si aún no está habilitado y busque errores de entrega en los logs de errores. Para obtener más información, consulte Supervisión de Amazon Data Firehose mediante registros CloudWatch .
- Si ves un mensaje de error en el registro que dice «Se encontró una Firehose InternalServerError al llamar al servicio Amazon S3». Se volverá a intentar la operación; si el error persiste, póngase en contacto con S3 para obtener una solución. , podría deberse al aumento significativo de las tasas de solicitud en una sola partición de S3. Puede optimizar los patrones de diseño de los prefijos de S3 para mitigar el problema. Para obtener más información, consulte Patrones de diseño de prácticas recomendadas: optimización del rendimiento de Amazon S3. Si esto no resuelve el problema, ponte en contacto con AWS Support para obtener más ayuda.
- Asegúrese de que el bucket de Amazon S3 que se especificó en su transmisión de Firehose siga existiendo.

- Si la transformación de datos con Lambda está habilitada, asegúrese de que la función Lambda especificada en la transmisión de Firehose siga existiendo.
- Asegúrese de que la función de IAM que se especifica en la transmisión de Firehose tenga acceso al bucket de S3 y a la función Lambda (si la transformación de datos está habilitada). Además, asegúrate de que la función de IAM tenga acceso a los grupos de registros y a los flujos de CloudWatch registros para comprobar los registros de errores. Para obtener más información, consulte Conceda a Amazon Data Firehose acceso a un destino de Amazon S3.
- Si usa la transformación de datos, asegúrese de que la función de Lambda nunca devuelva respuestas cuyo tamaño de carga sea superior a 6 MB. Para obtener más información, consulte Amazon Data FirehoseData Transformation.

Solución de problemas de Amazon Redshift

Compruebe lo siguiente si los datos no se entregan en el clúster aprovisionado de Amazon Redshift o el grupo de trabajo de Amazon Redshift sin servidor.

Los datos se entregan en el bucket de S3 antes de cargarse en Amazon Redshift. Si los datos no se han entregado en el bucket de S3, consulte Solución de problemas de Amazon S3.

- Compruebe la `DeliveryToRedshift.Success` métrica de Firehose para asegurarse de que Firehose ha intentado copiar datos del bucket de S3 al clúster aprovisionado de Amazon Redshift o al grupo de trabajo Amazon Redshift Serverless. Para obtener más información, consulte Supervisión de Amazon Data Firehose mediante métricas CloudWatch .
- Habilite el registro de errores si aún no está habilitado y busque errores de entrega en los logs de errores. Para obtener más información, consulte Supervisión de Amazon Data Firehose mediante registros CloudWatch .
- Consulte la `STL_CONNECTION_LOG` tabla de Amazon Redshift para ver si Firehose puede establecer conexiones satisfactorias. En esta tabla, debería poder ver las conexiones y su estado por nombre de usuario. Para obtener más información, consulte `STL_CONNECTION_LOG` en la Guía para desarrolladores de bases de datos de Amazon Redshift.
- Si al consultar la tabla anterior observa que se están estableciendo conexiones, consulte la tabla `STL_LOAD_ERRORS` de Amazon Redshift para dar con el motivo del error de COPY. Para obtener más información, consulte `STL_LOAD_ERRORS` en la Guía para desarrolladores de bases de datos de Amazon Redshift.
- Asegúrese de que la configuración de Amazon Redshift de su transmisión de Firehose sea precisa y válida.

- Asegúrese de que la función de IAM que se especifica en la transmisión de Firehose pueda acceder al bucket de S3 desde el que Amazon Redshift copia los datos y también a la función Lambda para la transformación de datos (si la transformación de datos está habilitada). Además, asegúrese de que la función de IAM tenga acceso al grupo de registros y a las secuencias de CloudWatch registros para comprobar los registros de errores. Para obtener más información, consulte Conceda a Amazon Data Firehose acceso a un destino de Amazon Redshift .
- Si el clúster aprovisionado de Amazon Redshift o el grupo de trabajo Amazon Redshift Serverless se encuentra en una nube privada virtual (VPC), asegúrese de que el clúster permita el acceso desde las direcciones IP de Firehose. Para obtener más información, consulte Conceda a Amazon Data Firehose acceso a un destino de Amazon Redshift .
- Asegúrese de que el clúster aprovisionado de Amazon Redshift o el grupo de trabajo de Amazon Redshift sin servidor esté disponible públicamente.
- Si usa la transformación de datos, asegúrese de que la función de Lambda nunca devuelva respuestas cuyo tamaño de carga sea superior a 6 MB. Para obtener más información, consulte Amazon Data FirehoseData Transformation.

Solución de problemas de Amazon OpenSearch Service

Comprueba lo siguiente si los datos no se envían a tu dominio OpenSearch de servicio.

Se pueden crear copias de seguridad de los datos en su bucket de Amazon S3 simultáneamente. Si los datos no se han entregado al bucket de S3, consulte Solución de problemas de Amazon S3.

- Comprueba la Firehose `IncomingBytes` y `IncomingRecords` las métricas para asegurarte de que los datos se envíen correctamente a tu transmisión de Firehose. Para obtener más información, consulte Supervisión de Amazon Data Firehose mediante métricas CloudWatch .
- Si la transformación de datos con Lambda está habilitada, compruebe la `ExecuteProcessingSuccess` métrica Firehose para asegurarse de que Firehose ha intentado invocar la función Lambda. Para obtener más información, consulte Supervisión de Amazon Data Firehose mediante métricas CloudWatch .
- Compruebe la `DeliveryToAmazonOpenSearchService.Success` métrica Firehose para asegurarse de que Firehose ha intentado indexar los datos en el clúster de servicio. OpenSearch Para obtener más información, consulte Supervisión de Amazon Data Firehose mediante métricas CloudWatch .

- Habilite el registro de errores si aún no está habilitado y busque errores de entrega en los logs de errores. Para obtener más información, consulte Supervisión de Amazon Data Firehose mediante registros CloudWatch .
- Asegúrese de que la configuración del OpenSearch servicio de su transmisión de Firehose sea precisa y válida.
- Si la transformación de datos con Lambda está habilitada, asegúrese de que la función Lambda especificada en la transmisión de Firehose siga existiendo. Además, asegúrese de que la función de IAM tenga acceso a los grupos de registros y a los flujos de CloudWatch registros para comprobar los registros de errores. Para obtener más información, consulte Concesión FirehoseAccess a un destino de OpenSearch servicio público.
- Asegúrese de que la función de IAM que se especifica en la transmisión de Firehose pueda acceder al OpenSearch clúster de servicios, al depósito de backup de S3 y a la función Lambda (si la transformación de datos está habilitada). Además, asegúrate de que la función de IAM tenga acceso a los grupos de registros y a los flujos de CloudWatch registros para comprobar los registros de errores. Para obtener más información, consulte Concesión FirehoseAccess a un destino de OpenSearch servicio público.
- Si usa la transformación de datos, asegúrese de que la función de Lambda nunca devuelva respuestas cuyo tamaño de carga sea superior a 6 MB. Para obtener más información, consulte Amazon Data FirehoseData Transformation.
- Amazon Data Firehose no admite actualmente la entrega de registros CloudWatch al OpenSearch destino de Amazon Service porque Amazon CloudWatch combina varios eventos de registro en un registro de Firehose y OpenSearch Amazon Service no puede aceptar varios eventos de registro en un registro. Como alternativa, puedes considerar usar un filtro de suscripción para Amazon OpenSearch Service in CloudWatch Logs.

Solución de problemas de Splunk

Realice las siguientes comprobaciones si los datos no se entregan a su punto de enlace de Splunk.

- Si tu plataforma Splunk está en una VPC, asegúrate de que Firehose pueda acceder a ella. Para obtener más información, consulte Acceso a Splunk en VPC.
- Si utilizas un balanceador de AWS cargas, asegúrate de que sea un Classic Load Balancer o un Application Load Balancer. Además, habilite las sesiones fijas basadas en la duración con la caducidad de las cookies deshabilitada para Classic Load Balancer y la caducidad establecida en el máximo (7 días) para Application Load Balancer. Para obtener información sobre cómo hacerlo,

consulte Duración de la sesión basada en la duración para Classic Load Balancer o Application Load Balancer.

- Revise los requisitos de la plataforma Splunk. El complemento Splunk para Firehose requiere la versión 6.6.X o posterior de la plataforma Splunk. Para obtener más información, consulte Splunk Add-on for Amazon Kinesis Firehose.
- Si tienes un proxy (Elastic Load Balancing u otro) entre Firehose y el nodo HTTP Event Collector (HEC), habilita las sesiones fijas para admitir los reconocimientos de HEC (ACK).
- Asegúrese de utilizar un token de HEC válido.
- Asegúrese de que el token de HEC esté habilitado. Consulte Enable and disable Event Collector tokens.
- Compruebe que los datos que está enviando a Splunk tienen el formato correcto. Para obtener más información, consulte Format events for HTTP Event Collector.
- Asegúrese de que el token de HEC y el evento de entrada estén configurados con un índice válido.
- Cuando una carga en Splunk falla debido a un error del servidor desde el nodo del HEC, la solicitud vuelve a enviarse automáticamente. Si fallan todos los reintentos, se crea una copia de seguridad de los datos en Amazon S3. Compruebe si los datos aparecen en Amazon S3; esa es una indicación de este tipo de error.
- Asegúrese de que ha activado la confirmación de indexadores en el token de HEC. Para obtener más información, consulte Enable indexer acknowledgement.
- Aumente el valor de `HECAcknowledgmentTimeoutInSeconds` la configuración de destino de Splunk de su transmisión Firehose.
- Aumente el valor de `DurationInSeconds` under `RetryOptions` en la configuración de destino de Splunk de su transmisión de Firehose.
- Compruebe el estado del HEC.
- Si usa la transformación de datos, asegúrese de que la función de Lambda nunca devuelva respuestas cuyo tamaño de carga sea superior a 6 MB. Para obtener más información, consulte Amazon Data FirehoseData Transformation.
- Asegúrese de que el parámetro Splunk denominado `ackIdleCleanup` se establece en `true`. Su valor predeterminado es false. Para establecer este parámetro en `true`, haga lo siguiente:
  - Para una implementación de Splunk Cloud administrada, envíe un caso utilizando el portal de soporte de Splunk. En este caso, pida al soporte de Splunk que habilite el recopilador de eventos HTTP, establezca `ackIdleCleanup` en `true` en `inputs.conf` y cree o modifique un balanceador de carga para usarlo con este complemento.

  - Para efectuar una implementación distribuida de Splunk Enterprise, establezca el parámetro `ackIdleCleanup` en true en el archivo `inputs.conf`. Para los usuarios de *nix, este archivo se encuentra en `$SPLUNK_HOME/etc/apps/splunk_httpinput/local/`. Para los usuarios de Windows, se encuentra en `%SPLUNK_HOME%\etc\apps\splunk_httpinput\local\`.
  - Para efectuar una implementación de una sola instancia de Splunk Enterprise, establezca el parámetro `ackIdleCleanup` en `true` en el archivo `inputs.conf`. Para los usuarios de *nix, este archivo se encuentra en `$SPLUNK_HOME/etc/apps/splunk_httpinput/local/`. Para los usuarios de Windows, se encuentra en `%SPLUNK_HOME%\etc\apps\splunk_httpinput\local\`.
- Asegúrese de que la función de IAM que se especifica en la transmisión de Firehose pueda acceder al depósito de respaldo de S3 y a la función Lambda para la transformación de datos (si la transformación de datos está habilitada). Además, asegúrate de que la función de IAM tenga acceso al grupo de registros y a CloudWatch los flujos de registros para comprobar los registros de errores. Para obtener más información, consulte Concesión FirehoseAccess a un destino de Splunk.
- Consulte Troubleshoot the Splunk Add-on for Amazon Kinesis Firehose.

Solución de problemas de Snowflake

En esta sección se describen los pasos de solución de problemas habituales al utilizar Snowflake como destino

La creación de la transmisión de Firehose falla

Si se produce un error al crear una transmisión de Firehose para una transmisión que entrega datos a un clúster de Snowflake PrivateLink habilitado, esto indica que Firehose no puede acceder al VPCE-ID. Esto puede deberse a uno de los siguientes motivos:

- VPCE-ID incorrecto. Confirme que no haya errores tipográficos.
- Firehose no admite las URL de Snowflake sin región en la vista previa. Proporcione la URL mediante el localizador de cuentas de Snowflake. Consulte la documentación de Snowflake para obtener más información.
- Confirma que el arroyo Firehose se haya creado en la misma AWS región que la región de los copos de nieve.
- Si el problema persiste, ponte en contacto con el servicio de asistencia. AWS

Fallos en la entrega

Compruebe lo siguiente si los datos no se envían a su tabla de Snowflake. Los datos fallidos en la entrega de Snowflake se enviarán al depósito de errores de S3 junto con un código de error y un mensaje de error correspondientes a la carga útil. Los siguientes son algunos de los escenarios de error más comunes. Para ver la lista completa de códigos de error, consulteErrores en la entrega de datos de Snowflake.

- Código de error: Snowflake. DefaultRoleMissing: Indica que el rol de snowflake no está configurado al crear la transmisión de Firehose. Si el rol de Snowflake no está configurado, asegúrese de establecer un rol predeterminado para el usuario de Snowflake especificado.
- Código de error: Snowflake. ExtraColumns: Indica que se ha rechazado la inserción en Snowflake debido a que hay columnas adicionales en la carga útil de entrada. No se deben especificar las columnas que no estén presentes en la tabla. Tenga en cuenta que los nombres de las columnas de Snowflake distinguen mayúsculas de minúsculas. Si la entrega no se realiza correctamente debido a este error a pesar de que la columna esté presente en la tabla, asegúrese de que las mayúsculas y minúsculas del nombre de la columna en la carga útil de entrada coincidan con el nombre de la columna indicado en la definición de la tabla.
- Código de error: Snowflake. MissingColumns: Indica que la inserción en Snowflake se ha rechazado porque faltan columnas en la carga útil de entrada. Asegúrese de que los valores estén especificados para todas las columnas que no admiten valores NULL.
- Código de error: Snowflake. InvalidInput: Esto puede ocurrir si Firehose no ha podido analizar la carga útil de entrada proporcionada en un formato JSON válido. Asegúrate de que la carga útil json esté bien formada y que no tenga comillas dobles, comillas, caracteres de escape adicionales, etc. Actualmente, Firehose solo admite un elemento JSON como carga útil de registro, no se admiten matrices JSON.
- Código de error: Snowflake. InvalidValue: Indica que la entrega ha fallado debido a un tipo de datos incorrecto en la carga útil de entrada. Asegúrese de que los valores de JSON especificados en la carga útil de entrada se ajusten al tipo de datos declarado en la definición de la tabla de Snowflake.
- Código de error: Snowflake. InvalidTableType: Indica que el tipo de tabla configurado en la transmisión Firehose no es compatible. Consulte las limitaciones (en Limitaciones) de la transmisión automática automática para conocer las tablas, columnas y tipos de datos compatibles.

> **Note**
>
> Por cualquier motivo, si la definición de la tabla o los permisos del rol se cambian en tu destino de Snowflake después de crear la transmisión de Firehose, Firehose puede tardar varios minutos en detectar esos cambios. Si ves errores de entrega debido a esto, intenta eliminar y volver a crear la transmisión de Firehose.

Solución de problemas de accesibilidad a los puntos finales de Firehose

Si se agota el tiempo de espera de la API Firehose, realiza los siguientes pasos para probar la accesibilidad de los puntos finales:

- Compruebe si las solicitudes de API se realizan desde un host de una VPC. Todo el tráfico de una VPC requiere la configuración de un punto final de VPC Firehose. Para obtener más información, consulte Uso de Firehose con. AWS PrivateLink
- Si el tráfico proviene de una red pública o una VPC con el punto final de la VPC Firehose configurado en una subred determinada, ejecute los siguientes comandos desde el host para comprobar la conectividad de la red. El punto final de Firehose se encuentra en los puntos finales y cuotas de Firehose.
  - Utilice herramientas como traceroute o tcping para comprobar si la configuración de la red es correcta. Si eso no funciona, comprueba la configuración de la red:

    Por ejemplo:

    ```
    traceroute firehose.us-east-2.amazonaws.com
    ```

    o

    ```
    tcping firehose.us-east-2.amazonaws.com 443
    ```

  - Si parece que la configuración de red es correcta y se produce un error en el siguiente comando, compruebe si Amazon CA (Autoridad de certificación) está en la cadena de confianza.

    Por ejemplo:

```
curl firehose.us-east-2.amazonaws.com
```

Si los comandos anteriores se ejecutan correctamente, vuelva a probar la API para comprobar si la API devuelve una respuesta.

Solución de problemas de puntos de conexión HTTP

En esta sección se describen los pasos de solución de problemas habituales cuando Amazon Data Firehose entrega datos a destinos genéricos de puntos de enlace HTTP y a destinos de socios, como Datadog, Dynatrace, LogicMonitor MongoDB, New Relic, Splunk o Sumo Logic. A efectos de esta sección, todos los destinos aplicables se denominan puntos de conexión HTTP. Asegúrese de que la función de IAM que se especifica en la transmisión de Firehose pueda acceder al depósito de respaldo de S3 y a la función Lambda para la transformación de datos (si la transformación de datos está habilitada). Además, asegúrate de que la función de IAM tenga acceso a los grupos de registros y a los flujos de CloudWatch registros para comprobar los registros de errores. Para obtener más información, consulte Conceder a Firehose acceso a un destino de punto final HTTP.

> **Note**
>
> La información de esta sección no se aplica a los siguientes destinos: Splunk, OpenSearch Service, S3 y Redshift.

CloudWatch Registros

Se recomienda encarecidamente activar CloudWatch Logging for Firehose. Los registros solo se publican cuando hay errores en la entrega en su destino.

Excepciones de destino

ErrorCode: HttpEndpoint.DestinationException

```
{
    "deliveryStreamARN": "arn:aws:firehose:us-east-1:123456789012:deliverystream/
ronald-test",
    "destination": "custom.firehose.endpoint.com...",
```

```
    "deliveryStreamVersionId": 1,
    "message": "The following response was received from the endpoint destination.
 413: {\"requestId\": \"43b8e724-dbac-4510-adb7-ef211c6044b9\", \"timestamp\":
 1598556019164, \"errorMessage\": \"Payload too large\"}",
    "errorCode": "HttpEndpoint.DestinationException",
    "processor": "arn:aws:lambda:us-east-1:379522611494:function:httpLambdaProcessing"
}
```

Las excepciones de destino indican que Firehose puede establecer una conexión con su punto de conexión y realizar una solicitud HTTP, pero no ha recibido un código de respuesta 200. Las respuestas 2xx que no sean 200 también generarán una excepción de destino. Amazon Data Firehose registra en Logs el código de respuesta y una carga útil de respuesta truncada recibida desde el punto de enlace configurado. CloudWatch Como Amazon Data Firehose registra el código de respuesta y la carga útil sin modificarlos ni interpretarlos, corresponde al punto final indicar el motivo exacto por el que rechazó la solicitud de entrega HTTP de Amazon Data Firehose. A continuación se indican las recomendaciones de solución de problemas más comunes para estas excepciones:

- 400: Indica que estás enviando una solicitud incorrecta debido a una mala configuración de tu Amazon Data Firehose. Asegúrese de tener la URL, los atributos comunes, la codificación del contenido, la clave de acceso y las sugerencias de almacenamiento en búfer correctos para su destino. Consulte la documentación específica del destino sobre la configuración requerida.
- 401: Indica que la clave de acceso que configuraste para tu transmisión de Firehose es incorrecta o falta.
- 403: Indica que la clave de acceso que configuraste para tu transmisión de Firehose no tiene permisos para entregar datos al punto final configurado.
- 413: Indica que la carga útil de solicitud que Amazon Data Firehose envía al punto final es demasiado grande para que el punto final la gestione. Intente reducir la sugerencia de almacenamiento en búfer al tamaño recomendado para su destino.
- 429: Indica que Amazon Data Firehose envía solicitudes a un ritmo superior al que puede gestionar el destino. Ajuste la sugerencia de almacenamiento en búfer mediante el aumento del tiempo de almacenamiento en búfer o del tamaño del búfer (pero dentro del límite de su destino).
- 5xx: indica que hay un problema con el destino. El servicio Amazon Data Firehose sigue funcionando correctamente.

> **Important**
>
> Importante: Si bien estas son las recomendaciones habituales para la solución de problemas, es posible que los puntos de conexión específicos tengan diferentes motivos para proporcionar los códigos de respuesta, por lo que primero se deben seguir las recomendaciones específicas de los puntos de conexión.

Respuesta no válida

ErrorCode: HttpEndpoint.InvalidResponseFromDestination

```
{
    "deliveryStreamARN": "arn:aws:firehose:us-east-1:123456789012:deliverystream/
ronald-test",
    "destination": "custom.firehose.endpoint.com...",
    "deliveryStreamVersionId": 1,
    "message": "The response received from the specified endpoint is invalid.
 Contact the owner of the endpoint to resolve the issue. Response for request
 2de9e8e9-7296-47b0-bea6-9f17b133d847 is not recognized as valid JSON or has unexpected
 fields. Raw response received: 200 {\"requestId\": null}",
    "errorCode": "HttpEndpoint.InvalidResponseFromDestination",
    "processor": "arn:aws:lambda:us-east-1:379522611494:function:httpLambdaProcessing"
}
```

Las excepciones de respuesta no válidas indican que Amazon Data Firehose recibió una respuesta no válida del punto final de destino. La respuesta debe ajustarse a las especificaciones de la respuesta o Amazon Data Firehose considerará que el intento de entrega ha sido un error y volverá a entregar los mismos datos hasta que se supere el tiempo de reintento configurado. Amazon Data Firehose trata las respuestas que no siguen las especificaciones de respuesta como errores, incluso si la respuesta tiene un estado 200. Si está desarrollando un punto final compatible con Amazon Data Firehose, siga las especificaciones de respuesta para garantizar que los datos se entreguen correctamente.

A continuación, se muestran algunos de los tipos comunes de respuestas no válidas y cómo solucionarlos:

- JSON no válido o campos inesperados: indica que la respuesta no se puede deserializar correctamente como JSON o que tiene campos inesperados. Asegúrese de que la respuesta no tenga el contenido codificado.
- Falta RequestId: indica que la respuesta no contiene un RequestID.
- RequestId no coincide: indica que el RequestID de la respuesta no coincide con el RequestID saliente.
- Falta la marca de tiempo: indica que la respuesta no contiene ningún campo de marca de tiempo. El campo de marca de tiempo debe ser un número y no una cadena.
- Falta el encabezado Content-Type: indica que la respuesta no contiene un encabezado “content-type: application/json”. No se acepta ningún otro content-type.

> **Important**
>
> Importante: Amazon Data Firehose solo puede entregar datos a puntos de conexión que cumplan con las especificaciones de solicitud y respuesta de Firehose. Si está configurando su destino para un servicio de terceros, asegúrese de utilizar el punto de conexión correcto compatible con Amazon Data Firehose, que probablemente sea diferente del punto de conexión de ingesta público. Por ejemplo, el punto final Amazon Data Firehose de Datadog es https://aws-kinesis-http-intake.logs.datadoghq.com/, mientras que su punto final público es https://api.datadoghq.com/.

Otros errores habituales

A continuación se indican los códigos de error y las definiciones adicionales.

- Código HttpEndpoint de error:. RequestTimeout- Indica que el punto final tardó más de 3 minutos en responder. Si es el propietario del destino, reduzca el tiempo de respuesta del punto de conexión de destino. Si no es el propietario del destino, póngase en contacto con el propietario y pregúntele si se puede hacer algo para reducir el tiempo de respuesta (como reducir la sugerencia de almacenamiento en búfer para que se procesen menos datos por solicitud).
- Código de error: HttpEndpoint. ResponseTooLarge- Indica que la respuesta es demasiado grande. La respuesta debe ser inferior a 1 MiB, incluidos los encabezados.
- Código de error: HttpEndpoint. ConnectionFailed- Indica que no se ha podido establecer una conexión con el punto final configurado. Esto puede deberse a un error tipográfico en la URL

configurada, a que Amazon Data Firehose no pueda acceder al punto de conexión o a que el punto de enlace tarde demasiado en responder a la solicitud de conexión.

- Código de error:. HttpEndpoint ConnectionReset- Indica que se ha establecido una conexión, pero el punto final la ha restablecido o cerrado prematuramente.
- Código de error: HttpEndpoint .SSL HandshakeFailure: indica que no se pudo completar correctamente un protocolo de enlace SSL con el punto final configurado.

Solución de problemas de MSK como origen

En esta sección se describen los pasos comunes de solución de problemas al utilizar MSK como origen.

> **Note**
>
> Para solucionar problemas de procesamiento, transformación o entrega de S3, consulte las secciones anteriores.

Error de creación de conductos

Compruebe lo siguiente si el conducto con MSK como origen no se crea correctamente:

- Compruebe que el clúster de MSK de origen se encuentre en estado activo.
- Si utiliza la conectividad privada, asegúrese de que el enlace privado en el clúster esté activado.

  Si utiliza la conectividad pública, asegúrese de que el acceso público en el clúster esté activado.
- Si utiliza la conectividad privada, asegúrese de agregar una política basada en recursos que permita a Firehose crear un enlace privado. Consulte también: MSK cross account permissions
- Asegúrese de que el rol de la configuración de origen tenga permiso para ingerir datos del tema del clúster.
- Asegúrese de que los grupos de seguridad de la VPC permitan el tráfico de entrada en los puertos que utilizan los servidores de arranque del clúster.

Conducto suspendido

Compruebe lo siguiente si el conducto se encuentra en estado SUSPENDIDO:

- Compruebe que el clúster de MSK de origen se encuentre en estado activo.
- Compruebe que el tema de origen existe. En caso de que el tema se haya eliminado y vuelto a crear, tendrás que eliminar y volver a crear también la transmisión de Firehose.

Conducto contrapresurizado

El valor de DataReadFromSource .Backpressured será 1 si se supera BytesPerSecondLimit cada partición o si el flujo normal de entrega es lento o se detiene.

- Si está acertando BytesPerSecondLimit , compruebe la métrica de DataReadFromSource .Bytes y solicite un aumento del límite.
- Compruebe los CloudWatch registros, las métricas de destino, las métricas de transformación de datos y las métricas de conversión de formato para identificar los cuellos de botella.

Actualización incorrecta de los datos

La actualización de los datos parece incorrecta.

- Firehose calcula la actualización de los datos en función de la marca de tiempo del registro consumido. Para garantizar que esta marca de tiempo se registre correctamente cuando el registro del productor se conserva en los registros del agente de Kafka, defina la configuración del tipo de marca de tiempo del tema de Kafka para que sea `message.timestamp.type=LogAppendTime`.

Problemas de conexión al clúster de MSK

El siguiente procedimiento explica cómo se puede validar la conectividad con los clústeres de MSK. Para obtener más información sobre la configuración del cliente Amazon MSK, consulte Introducción a Amazon MSK en la Guía para desarrolladores de Amazon Managed Streaming for Apache Kafka.

Para validar la conectividad con los clústeres de MSK

1. Cree una instancia Amazon EC2 basada en Unix (preferiblemente AL2). Si solo tiene habilitada la conectividad de VPC en su clúster, asegúrese de que su instancia EC2 se ejecute en la misma VPC. Utilice SSH en la instancia una vez que esté disponible. Para obtener más información, consulte este tutorial en la Guía del usuario de Amazon EC2.

2. Instale Java mediante el administrador de paquetes Yum ejecutando el siguiente comando. Para obtener más información, consulte las instrucciones de instalación en la Guía del usuario de Amazon Corretto 8.

```
sudo yum install java-1.8.0
```

3. Instale el AWS cliente ejecutando el siguiente comando.

```
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install
```

4. Descargue la versión 2.6* del cliente Apache Kafka ejecutando el siguiente comando.

```
wget https://archive.apache.org/dist/kafka/2.6.2/kafka_2.12-2.6.2.tgz
tar -xzf kafka_2.12-2.6.2.tgz
```

5. Vaya al directorio `kafka_2.12-2.6.2/libs` y ejecute el siguiente comando para descargar el archivo JAR de IAM de Amazon MSK.

```
wget https://github.com/aws/aws-msk-iam-auth/releases/download/v1.1.3/aws-msk-iam-
auth-1.1.3-all.jar
```

6. Cree el `client.properties` archivo en la carpeta bin de Kafka.

7. `awsRoleArn`Sustitúyala por la función ARN que utilizaste en tu Firehose `SourceConfiguration` y verifica la ubicación del certificado. Permita que el usuario AWS cliente asuma el rol. `awsRoleArn` AWS el usuario cliente intentará asumir el rol que especificó aquí.

```
[ec2-user@ip-xx-xx-xx-xx bin]$ cat client.properties
security.protocol=SASL_SSL
sasl.mechanism=AWS_MSK_IAM
sasl.jaas.config=software.amazon.msk.auth.iam.IAMLoginModule required
 awsRoleArn="<role arn>" awsStsRegion="<region name>";
sasl.client.callback.handler.class=software.amazon.msk.auth.iam.IAMClientCallbackHandler
awsDebugCreds=true
ssl.truststore.location=/usr/lib/jvm/java-1.8.0-
openjdk-1.8.0.342.b07-1.amzn2.0.1.x86_64/jre/lib/security/cacerts
ssl.truststore.password=changeit
```

8. Ejecute el siguiente comando de Kafka para enumerar los temas. Si su conexión es pública, utilice los servidores Bootstrap de punto final públicos. Si su conexión es privada, utilice los servidores Bootstrap de punto final privados.

```
bin/kafka-topics.sh --list --bootstrap-server <bootstrap servers> --command-config
 bin/client.properties
```

Si la solicitud se realiza correctamente, debería ver un resultado similar al siguiente ejemplo.

```
[ec2-user@ip-xx-xx-xx-xx kafka_2.12-2.6.2]$ bin/kafka-topics.sh --list --bootstrap-
server <bootstrap servers> --command-config bin/client.properties

[xxxx-xx-xx 05:49:50,877] WARN The configuration 'awsDebugCreds' was supplied but
 isn't a known config. (org.apache.kafka.clients.admin.AdminClientConfig)
[xxxx-xx-xx 05:49:50,878] WARN The configuration 'ssl.truststore.location' was
 supplied but isn't a known config.
 (org.apache.kafka.clients.admin.AdminClientConfig)
[xxxx-xx-xx 05:49:50,878] WARN The configuration 'sasl.jaas.config' was supplied
 but isn't a known config. (org.apache.kafka.clients.admin.AdminClientConfig)
[xxxx-xx-xx 05:49:50,878] WARN The configuration
 'sasl.client.callback.handler.class' was supplied but isn't a known config.
 (org.apache.kafka.clients.admin.AdminClientConfig)
[xxxx-xx-xx 05:49:50,878] WARN The configuration 'ssl.truststore.password' was
 supplied but isn't a known config.
 (org.apache.kafka.clients.admin.AdminClientConfig)
[xxxx-xx-xx 05:50:21,629] WARN [AdminClient clientId=adminclient-1] Connection to
 node...
__amazon_msk_canary
__consumer_offsets
```

9. Si tiene problemas al ejecutar el script anterior, compruebe que los servidores de arranque que proporcionó estén accesibles en el puerto especificado. Para ello, puede descargar y utilizar telnet o una utilidad similar, como se muestra en el siguiente comando.

```
sudo yum install telnet
telnet <bootstrap servers><port>
```

Si la solicitud se realiza correctamente, obtendrá el siguiente resultado. Esto significa que puedes conectarte a tu clúster de MSK dentro de tu VPC local y que los servidores bootstrap funcionan correctamente en el puerto especificado.

```
Connected to ..
```

10. Si la solicitud no se realiza correctamente, compruebe las reglas de entrada del grupo de seguridad de VPC. Por ejemplo, puedes usar las siguientes propiedades en la regla de entrada.

    ```
    Type: All traffic
    Port: Port used by the bootstrap server (e.g. 14001)
    Source: 0.0.0.0/0
    ```

    Vuelva a intentar la conexión telnet como se muestra en el paso anterior. Si sigues sin poder conectarte o la conexión de la Firehose sigue fallando, ponte en contacto con el AWS servicio de asistencia.

La métrica de actualización de los datos aumenta o no se emite

La actualización de los datos es una medida de la actualidad de los datos dentro de la transmisión Firehose. Es la edad del registro de datos más antiguo del flujo Firehose, medida desde el momento en que Firehose ingirió los datos hasta la actualidad. Firehose proporciona métricas que puede usar para monitorear la actualización de los datos. Para identificar la métrica de antigüedad de los datos para un destino determinado, consulte the section called “Monitorización con métricas CloudWatch ”.

Si habilita la copia de seguridad para todos los eventos o todos los documentos, monitorice dos métricas de antigüedad de los datos distintas: una para el destino principal y otra para la copia de seguridad.

Si no se emite la métrica de actualización de los datos, significa que no hay ninguna entrega activa para la transmisión de Firehose. Esto sucede cuando la entrega de datos está completamente bloqueada o cuando no hay datos entrantes.

Si la métrica de antigüedad de los datos aumenta constantemente, esto significa que la entrega de los datos está retrasada. Esto puede suceder por una de las siguientes razones.

- El destino no es capaz de soportar la velocidad de entrega. Si Firehose detecta errores transitorios debido al tráfico intenso, es posible que la entrega se retrase. Esto puede ocurrir en destinos distintos de Amazon S3 (puede ocurrir en OpenSearch Service, Amazon Redshift o Splunk). Asegúrese de que su destino tiene suficiente capacidad para tratar el tráfico entrante.

- El destino es lento. La entrega de datos podría retrasarse si Firehose encuentra una latencia alta. Monitorice la métrica de latencia del destino.
- La función de Lambda es lenta. Esto podría provocar una velocidad de entrega de datos inferior a la tasa de ingesta de datos de la transmisión Firehose. Si es posible, mejore la eficiencia de la función de Lambda. Por ejemplo, si la función realiza operaciones de E/S de red, use varios subprocesos o utilice operaciones de E/S asincrónicas para aumentar el paralelismo. Además, considere la posibilidad de aumentar el tamaño de memoria de la función de Lambda para que la asignación de CPU pueda aumentar en consecuencia. Esto podría producir invocaciones de Lambda más rápidas. Para obtener información sobre la configuración de las funciones de Lambda, consulte Configuración de funciones de Lambda AWS.
- Hay errores durante la entrega de datos. Para obtener información sobre cómo supervisar los errores mediante Amazon CloudWatch Logs, consultethe section called “Monitorización con registros CloudWatch ”.
- Si la fuente de datos de la transmisión de Firehose es una transmisión de datos de Kinesis, es posible que se esté produciendo una limitación. Compruebe las métricas `ThrottledGetRecords`, `ThrottledGetShardIterator` y `ThrottledDescribeStream`. Si hay varios consumidores asociados a la secuencia de datos de Kinesis, tenga en cuenta lo siguiente:
  - Si las métricas `ThrottledGetShardIterator` y `ThrottledGetRecords` son elevadas, le recomendamos que aumente el número de particiones aprovisionadas para la secuencia de datos.
  - Si `ThrottledDescribeStream` es alta, le recomendamos que añada el `kinesis:listshards` permiso al rol configurado en. KinesisStreamSourceConfiguration
- Sugerencias de almacenamiento en búfer bajo para el destino. Esto podría aumentar el número de viajes de ida y vuelta que Firehose debe realizar hasta el destino, lo que podría provocar retrasos en la entrega. Considere la posibilidad de aumentar el valor de las sugerencias de almacenamiento en búfer. Para obtener más información, consulte BufferingHints.
- Una duración de reintentos alta podría hacer que la entrega se retrasara cuando los errores son frecuentes. Considere la posibilidad de reducir la duración de los reintentos. Además, monitorice los errores e intente reducirlos. Para obtener información sobre cómo supervisar los errores mediante Amazon CloudWatch Logs, consultethe section called “Monitorización con registros CloudWatch ”.
- Si el destino es Splunk y la métrica `DeliveryToSplunk.DataFreshness` es elevada pero `DeliveryToSplunk.Success` tiene un valor correcto, es posible que el clúster de Splunk esté ocupado. Libere el clúster de Splunk si es posible. Como alternativa, ponte en contacto con AWS

Support y solicita un aumento del número de canales que Firehose utiliza para comunicarse con el clúster de Splunk.

La conversión del formato de registro a Apache Parquet falla

Esto ocurre si toma datos de DynamoDB que incluyen ese tipo, `Set` los transmite a través de Lambda a una transmisión de Firehose y utiliza AWS Glue Data Catalog an para convertir el formato de registro a Apache Parquet.

Cuando el AWS Glue rastreador indexa los tipos de datos del conjunto de DynamoDB (`StringSet`, y`BinarySet`)`NumberSet`, los almacena en el catálogo de datos como`SET<STRING>`, y, respectivamente. `SET<BIGINT>` `SET<BINARY>` Sin embargo, para que Firehose convierta los registros de datos al formato Apache Parquet, necesita los tipos de datos de Apache Hive. Dado que los tipos de conjunto no son tipos de datos de Apache Hive válidos, la conversión produce un error. Para que la conversión funcione, actualice el catálogo de datos con los tipos de datos de Apache Hive. Puede hacerlo cambiando `set` a `array` en el catálogo de datos.

Para cambiar uno o más tipos de datos de **`set`** a **`array`** en un AWS Glue catálogo de datos

1. Inicie sesión AWS Management Console y abra la AWS Glue consola en https://console.aws.amazon.com/glue/.
2. En el panel izquierdo, en el encabezado Catálogo de datos , elija Tablas.
3. En la lista de tablas, elija el nombre de la tabla en la que desea modificar uno o varios tipos de datos. Esto le lleva a la página de detalles de la tabla.
4. Elija el botón Editar esquema situado en la esquina superior derecha de la página de detalles.
5. En la columna Tipo de datos , elija el primer tipo de datos `set` .
6. En la lista desplegable Tipo de columna , cambie el tipo de `set` a `array`.
7. En el ArraySchemacampo`array<string>`, introduzca o `array<int>array<binary>`, según el tipo de datos apropiado para su escenario.
8. Elija Actualizar.
9. Repita los pasos anteriores para convertir otros tipos `set` a tipos `array` .
10. Elija Guardar.

Cuota de Amazon Data Firehose

Amazon Data Firehose tiene la siguiente cuota.

- Con Amazon MSK como fuente de la transmisión Firehose, cada transmisión Firehose tiene una cuota predeterminada de 10 MB/s de rendimiento de lectura por partición y un tamaño de registro máximo de 10 MB. Puede utilizar el aumento de la cuota del servicio para solicitar un aumento de la cuota predeterminada de 10 MB/s de rendimiento de lectura por partición.
- Con Amazon MSK como fuente de la transmisión Firehose, hay un tamaño de registro máximo de 6 Mb si AWS Lambda está habilitada y un tamaño de registro máximo de 10 Mb si Lambda está deshabilitada. AWS Lambda limita su registro entrante a 6 MB y Amazon Data Firehose reenvía los registros de más de 6 MB a un bucket de error de S3. Si Lambda está deshabilitada, Firehose limita su registro entrante a 10 MB. Si Amazon Data Firehose recibe un tamaño de registro de Amazon MSK superior a 10 MB, Amazon Data Firehose envía este registro al depósito de errores de S3 y emite las métricas de Cloudwatch a su cuenta. Para obtener más información sobre los límites de AWS Lambda, consulte: https://docs.aws.amazon.com/lambda/latest/dg/gettingstarted-limits.html.
- Cuando se habilita la partición dinámica en una transmisión de Firehose, hay una cuota predeterminada de 500 particiones activas que se pueden crear para esa transmisión de Firehose. El recuento de particiones activas es el número total de particiones activas en el búfer de entrega. Por ejemplo, si la consulta de particionamiento dinámico crea 3 particiones por segundo y tiene una configuración de sugerencias de búfer que activa la entrega cada 60 segundos, tendrá un promedio de 180 particiones activas. Una vez que los datos se entregan en una partición, dicha partición deja de estar activa. Puede utilizar el formulario Amazon Data Firehose Limits para solicitar un aumento de esta cuota hasta 5000 particiones activas por transmisión de Firehose determinada. Si necesitas más particiones, puedes crear más transmisiones de Firehose y distribuir las particiones activas entre ellas.
- Cuando se habilita la partición dinámica en una transmisión Firehose, se admite un rendimiento máximo de 1 GB por segundo para cada partición activa.
- Cada cuenta tendrá la siguiente cuota de retransmisiones de Firehose por región:
  - EE.UU. Este (Norte de Virginia), EE.UU. Este (Ohio), EE.UU. Oeste (Oregón), Europa (Irlanda), Asia-Pacífico (Tokio): 5,000 Firehose Streams
  - Europa (Fráncfort), Europa (Londres), Asia Pacífico (Singapur), Asia Pacífico (Sídney), Asia Pacífico (Seúl), Asia Pacífico (Bombay), AWS GovCloud (EE. UU.), Canadá (oeste), Canadá (centro): 2000 arroyos Firehose

- Europa (París), Europa (Milán), Europa (Estocolmo), Asia Pacífico (Hong Kong), Asia Pacífico (Osaka), Sudamérica (São Paulo), China (Ningxia), China (Pekín), Oriente Medio (Bahréin), (EEUU-Este), África AWS GovCloud (Ciudad del Cabo): 500 arroyos Firehose
- Europa (Zúrich), Europa (España), Asia Pacífico (Hyderabad), Asia Pacífico (Yakarta), Asia Pacífico (Melbourne), Oriente Medio (Emiratos Árabes Unidos), Israel (Tel Aviv), Canadá oeste (Calgary), Canadá (centro): 100 arroyos Firehose
- Si superas este número, una llamada a este número supone una excepción CreateDeliveryStream. `LimitExceededException` Para aumentar esta cuota, puede utilizar Service Quotas si está disponible en su región. Para obtener más información acerca de cómo usar Service Quotas, consulte Requesting a Quota Increase. Si las cuotas de servicio no están disponibles en tu región, puedes utilizar el formulario Amazon Data Firehose Limits para solicitar un aumento.

- Cuando Direct PUT se configura como fuente de datos, cada transmisión de Firehose proporciona la siguiente cuota combinada de solicitudes PutRecordy PutRecordBatchcuotas:
  - Para Este de EE. UU. (Norte de Virginia), Oeste de EE. UU. (Oregón) y Europa (Irlanda): 500 000 registros por segundo, 2000 solicitudes por segundo y 5 MiB por segundo.
  - Para EE. UU. Este (Ohio), EE. UU. Oeste AWS GovCloud (Norte de California), AWS GovCloud (EE. UU. Este), Asia Pacífico (Hong Kong), Asia Pacífico (Bombay), Asia Pacífico (Seúl), Asia Pacífico (Singapur), China (Beijing), China (Ningxia), Asia Pacífico (Sídney), Asia Pacífico (Tokio), Asia Pacífico (Tokio), Canadá (Central), Canadá Oeste (Calgary), Europa (Frankfurt), Europa (Londres), Europa (París), Europa (Estocolmo), Oriente Medio (Bahréin), Sudamérica (São Paulo), África (Ciudad del Cabo) y Europa (Milán): 100 000 registros por segundo, 1000 solicitudes por segundo y 1 MiB/segundo.

  Para solicitar un aumento de la cuota, utiliza el formulario Amazon Data Firehose Limits. Las tres cuotas escalan proporcionalmente. Por ejemplo, si aumenta la cuota de rendimiento en Este de EE. UU. (Norte de Virginia), Oeste de EE. UU. (Oregón) o Europa (Irlanda) a 10 MiB por segundo, las otras dos cuotas aumentan a 4000 solicitudes por segundo y 1 000 000 registros por segundo.

  > **Important**
  >
  > Si la cuota incrementada es muy superior al tráfico en ejecución, la entrega a los destinos se produce en lotes pequeños. Esto resulta poco eficiente y puede provocar costos superiores en los servicios de destino. Asegúrese de incrementar la cuota actual al nivel

necesario para satisfacer solo el tráfico en ejecución, e increméntela más si el tráfico aumenta.

> **Important**
>
> Tenga en cuenta que los registros de datos más pequeños pueden generar costos más altos. Los precios de ingesta de Firehose se basan en la cantidad de registros de datos que envías al servicio, multiplicada por el tamaño de cada registro redondeado al alza a los 5 KB (5120 bytes) más cercanos. Por lo tanto, para el mismo volumen de datos de entrada (bytes), si hay un número mayor de registros de entrada, el costo incurrido será mayor. Por ejemplo, si el volumen total de datos de entrada es de 5 MiB, enviar 5 MiB de datos de más de 5000 registros cuesta más que enviar la misma cantidad de datos con 1000 registros. Para obtener más información, consulte Amazon Data Firehose en la AWS calculadora.

> **Note**
>
> Cuando Kinesis Data Streams se configura como fuente de datos, esta cuota no se aplica y Amazon Data Firehose se amplía o reduce sin límite.

- Cada Firehose Stream almacena los registros de datos durante un máximo de 24 horas en caso de que el destino de entrega no esté disponible y la fuente sí lo esté. DirectPut Si el origen es Kinesis Data Streams (KDS) y el destino no está disponible, los datos se conservarán en función de la configuración de KDS.
- El tamaño máximo de un registro enviado a Amazon Data Firehose, antes de la codificación en base64, es de 1000 KiB.
- La PutRecordBatchoperación puede tomar hasta 500 registros por llamada o 4 MiB por llamada, lo que sea menor. Esta cuota no se puede cambiar.
- Las siguientes operaciones pueden proporcionar hasta cinco invocaciones por segundo (se trata de un límite máximo): `CreateDeliveryStream`, `DeleteDeliveryStream`, `DescribeDeliveryStream`, `ListDeliveryStreams`, `UpdateDestination`, `TagDeliveryStream`, `UntagDeliveryStream`, `ListTagsForDeliveryStream`, `StartDeliveryStreamEncryption`, `StopDeliveryStreamEncryption`.
- Las sugerencias de intervalo del búfer oscilan entre 60 y 900 segundos.

- Para la entrega de Amazon Data Firehose a Amazon Redshift, solo se admiten los clústeres de Amazon Redshift de acceso público.
- El intervalo de duración de los reintentos es de 0 a 7200 segundos para Amazon Redshift y Service Delivery. OpenSearch
- Firehose es compatible con las versiones 1.5, 2.3, 5.1, 5.3, 5.5 y 5.6 de Elasticsearch, así como con todas las versiones 6.* y 7.* y con Amazon OpenSearch Service 2.x hasta 2.11.
- Cuando el destino es Amazon S3, Amazon Redshift o OpenSearch Service, Amazon Data Firehose permite hasta 5 invocaciones Lambda pendientes por fragmento. En Splunk, la cuota es de 10 invocaciones de Lambda pendientes por partición.
- Puedes usar una CMK del tipo CUSTOMER_MANAGED_CMK para cifrar hasta 500 transmisiones de Firehose.

Apéndice: especificaciones de solicitudes y respuestas de entrega de puntos de conexión HTTP

Para que Amazon Data Firehose pueda entregar datos correctamente a puntos de enlace HTTP personalizados, estos puntos de enlace deben aceptar solicitudes y enviar respuestas utilizando determinados formatos de solicitud y respuesta de Amazon Data Firehose. En esta sección se describen las especificaciones de formato de las solicitudes HTTP que el servicio Amazon Data Firehose envía a puntos de enlace HTTP personalizados, así como las especificaciones de formato de las respuestas HTTP que espera el servicio Amazon Data Firehose. Los puntos de enlace HTTP tienen 3 minutos para responder a una solicitud antes de que Amazon Data Firehose agote el tiempo de espera de esa solicitud. Amazon Data Firehose trata las respuestas que no siguen el formato adecuado como errores de entrega.

Temas

- Formato de las solicitudes
- Formato de respuesta
- Ejemplos

Formato de las solicitudes

Parámetros de ruta y URL

Los configura directamente como parte de un único campo de URL. Amazon Data Firehose los envía tal y como están configurados sin modificarlos. Solo se admiten los destinos HTTPS. Las restricciones de URL se aplican durante la configuración del flujo de entrega.

> **Note**
>
> Actualmente, solo se admite el puerto 443 para la entrega de datos de puntos de conexión HTTP.

Encabezados HTTP: X-Amz-Firehose-Protocol-Version

Este encabezado se usa para indicar la versión de los formatos de solicitudes o respuestas. Actualmente, la única versión es la 1.0.

Encabezados HTTP: X-Amz-Firehose-Request-Id

El valor de este encabezado es un GUID opaco que se puede utilizar con fines de depuración y eliminación de duplicados. Si es posible, las implementaciones de puntos de conexión deben registrar el valor de este encabezado, tanto para las solicitudes que son correctas como para las que no lo son. El ID de solicitud se mantiene igual durante varios intentos de la misma solicitud.

Encabezados HTTP: Content-Type

El valor del encabezado Content-Type es siempre `application/json`.

Encabezados HTTP: Content-Encoding

Se puede configurar una transmisión de Firehose para que utilice GZIP para comprimir el cuerpo al enviar solicitudes. Cuando esta compresión está habilitada, el valor del encabezado Content-Encoding se establece en gzip, según la práctica habitual. Si la compresión no está habilitada, el encabezado Content-Encoding no aparece.

Encabezados HTTP: Content-Length

Se usa de forma estándar.

Encabezados HTTP: X-Amz-Firehose-Source-Arn:

El ARN de la secuencia Firehose representada en formato de cadena ASCII. El ARN codifica la región, el ID de la AWS cuenta y el nombre de la transmisión. Por ejemplo, `arn:aws:firehose:us-east-1:123456789:deliverystream/testStream`.

Encabezados HTTP: X-Amz-Firehose-Access-Key

Este encabezado contiene una clave de API u otras credenciales. Puede crear o actualizar la clave de API (también conocida como token de autorización) al crear o actualizar el flujo de entrega. Amazon Data Firehose restringe el tamaño de la clave de acceso a 4096 bytes. Amazon Data Firehose no intenta interpretar esta clave de ninguna manera. La clave configurada se copia palabra por palabra en el valor de este encabezado.

El contenido puede ser arbitrario y, potencialmente, representar un token JWT o una ACCESS_KEY. Si un punto de conexión requiere credenciales de varios campos (por ejemplo, nombre de usuario y contraseña), los valores de todos los campos deben almacenarse juntos en una única clave de acceso en un formato que el punto de conexión comprenda (JSON o CSV). Este campo puede codificarse en base64 si el contenido original es binario. Amazon Data Firehose no modifica ni codifica el valor configurado y utiliza el contenido tal cual.

Encabezados HTTP: X-Amz-Firehose-Common-Attributes

Este encabezado contiene los atributos comunes (metadatos) que pertenecen a toda la solicitud o a todos los registros de la solicitud. Tú los configuras directamente al crear una transmisión de Firehose. El valor de este atributo está codificado como un objeto JSON con el siguiente esquema:

```
"$schema": http://json-schema.org/draft-07/schema#

properties:
  commonAttributes:
    type: object
    minProperties: 0
    maxProperties: 50
    patternProperties:
      "^.{1,256}$":
        type: string
        minLength: 0
        maxLength: 1024
```

A continuación se muestra un ejemplo:

```
"commonAttributes": {
    "deployment -context": "pre-prod-gamma",
    "device-types": ""
  }
```

Cuerpo: tamaño máximo

Configura el tamaño máximo del cuerpo, que puede ser de hasta 64 MiB antes de la compresión.

Cuerpo: esquema

El cuerpo contiene un único documento JSON con el siguiente esquema JSON (escrito en YAML):

```
"$schema": http://json-schema.org/draft-07/schema#
```

```
title: FirehoseCustomHttpsEndpointRequest
description: >
  The request body that the Firehose service sends to
  custom HTTPS endpoints.
type: object
properties:
  requestId:
    description: >
      Same as the value in the X-Amz-Firehose-Request-Id header,
      duplicated here for convenience.
    type: string
  timestamp:
    description: >
      The timestamp (milliseconds since epoch) at which the Firehose
      server generated this request.
    type: integer
  records:
    description: >
      The actual records of the Firehose stream, carrying
      the customer data.
    type: array
    minItems: 1
    maxItems: 10000
    items:
      type: object
      properties:
        data:
          description: >
            The data of this record, in Base64. Note that empty
            records are permitted in Firehose. The maximum allowed
            size of the data, before Base64 encoding, is 1024000
            bytes; the maximum length of this field is therefore
            1365336 chars.
          type: string
          minLength: 0
          maxLength: 1365336

required:
  - requestId
  - records
```

A continuación se muestra un ejemplo:

```
{
  "requestId": "ed4acda5-034f-9f42-bba1-f29aea6d7d8f",
  "timestamp": 1578090901599
  "records": [
    {
      "data": "aGVsbG8="
    },
    {
      "data": "aGVsbG8gd29ybGQ="
    }
  ]
}
```

Formato de respuesta

Comportamiento predeterminado en caso de error

Si una respuesta no cumple con los requisitos que se indican a continuación, el servidor Firehose la tratará como si tuviera un código de estado 500 sin cuerpo.

Código de estado

El código de estado HTTP DEBE estar en el rango 2XX, 4XX o 5XX.

El servidor Amazon Data Firehose NO sigue las redirecciones (códigos de estado 3XX). Solo el código de respuesta 200 se considera una entrega correcta de los registros a HTTP/EP. El código de respuesta 413 (tamaño excedido) se considera un error permanente y, si está configurado, el lote de registros no se envía al bucket de errores. Todos los demás códigos de respuesta se consideran errores recuperables y están sujetos a un algoritmo de reintentos de retroceso que se explica más adelante.

Encabezados: tipo de contenido

El único tipo de contenido aceptable es application/json.

Encabezados HTTP: Content-Encoding

NO SE DEBE utilizar Content-Encoding. El cuerpo DEBE estar descomprimido.

Encabezados HTTP: Content-Length

El encabezado Content-Length DEBE estar presente si la respuesta tiene un cuerpo.

Cuerpo: tamaño máximo

El cuerpo de la respuesta debe tener un tamaño de 1 MiB o menos.

```
"$schema": http://json-schema.org/draft-07/schema#

title: FirehoseCustomHttpsEndpointResponse

description: >
  The response body that the Firehose service sends to
  custom HTTPS endpoints.
type: object
properties:
  requestId:
    description: >
      Must match the requestId in the request.
    type: string

  timestamp:
    description: >
      The timestamp (milliseconds since epoch) at which the
      server processed this request.
    type: integer

  errorMessage:
    description: >
      For failed requests, a message explaining the failure.
      If a request fails after exhausting all retries, the last
      Instance of the error message is copied to error output
      S3 bucket if configured.
    type: string
    minLength: 0
    maxLength: 8192
required:
  - requestId
  - timestamp
```

A continuación se muestra un ejemplo:

```
Failure Case (HTTP Response Code 4xx or 5xx)
{
  "requestId": "ed4acda5-034f-9f42-bba1-f29aea6d7d8f",
  "timestamp": "1578090903599",
  "errorMessage": "Unable to deliver records due to unknown error."
}
Success case (HTTP Response Code 200)
{
  "requestId": "ed4acda5-034f-9f42-bba1-f29aea6d7d8f",
  "timestamp": 1578090903599
}
```

Gestión de las respuestas de errores

En todos los casos de error, el servidor Amazon Data Firehose vuelve a intentar entregar el mismo lote de registros mediante un algoritmo de retroceso exponencial. Los reintentos se retrasan utilizando un tiempo de espera inicial (1 segundo) con un factor de fluctuación del (15%) y cada reintento posterior se retrasa utilizando la fórmula (initial-backoff-time * (multiplier (2) ^ retry_count)) con una fluctuación adicional. El tiempo de retroceso está limitado a un intervalo máximo de 2 minutos. Por ejemplo, en el enésimo reintento, el tiempo de espera es = MAX (120, 2^n) * random (0,85, 1,15).

Los parámetros especificados en la ecuación anterior están sujetos a cambios. Consulte la documentación de AWS Firehose para conocer el tiempo de retroceso inicial exacto, el tiempo máximo de retroceso y los porcentajes de multiplicador y fluctuación utilizados en el algoritmo de retroceso exponencial.

En cada intento posterior, la clave de acceso o el destino al que se envían los registros pueden cambiar en función de la configuración actualizada de la transmisión Firehose. El servicio Amazon Data Firehose utiliza el mismo identificador de solicitud en todos los reintentos de la mejor manera posible. El servidor de puntos de conexión HTTP puede utilizar esta última característica con fines de eliminación de duplicados. Si la solicitud sigue sin entregarse después del tiempo máximo permitido (según la configuración de la transmisión de Firehose), el lote de registros se puede entregar opcionalmente a un depósito de errores según la configuración de la transmisión.

Ejemplos

Ejemplo de una solicitud originada en CWLog:

```
{
  "requestId": "ed4acda5-034f-9f42-bba1-f29aea6d7d8f",
  "timestamp": 1578090901599,
  "records": [
   {
    "data": {
      "messageType": "DATA_MESSAGE",
      "owner": "123456789012",
      "logGroup": "log_group_name",
      "logStream": "log_stream_name",
      "subscriptionFilters": [
        "subscription_filter_name"
      ],
      "logEvents": [
        {
          "id": "01234567890123456789012345678901234567890123456789012345",
          "timestamp": 1510109208016,
          "message": "log message 1"
        },
        {
          "id": "01234567890123456789012345678901234567890123456789012345",
          "timestamp": 1510109208017,
          "message": "log message 2"
        }
      ]
    }
   }
  ]
}
```

Historial de documentos

En la siguiente tabla se describen los cambios importantes en la documentación de Amazon Data Firehose.

Cambio	Descripción	Fecha de modificación
Snowflake como destino en nuevas regiones	Snowflake ya está disponible como destino en Asia Pacífico (Singapur), Asia Pacífico (Seúl) y Asia Pacífico (Sídney). Consulte, the section called “Configure los ajustes de destino para Snowflake”.	19 de junio de 2024
Amazon Data Firehose se integra con AWS Secrets Manager	Ahora puede acceder a sus datos secretos y automatizar la rotación de credenciales de forma segura con Secrets Manager. Consulte, the section called “Autenticate con AWS Secrets Manager”.	06 de junio de 2024
Se agregó soporte para la ingesta de registros para Dynatrace	Ahora puede enviar registros y eventos a Dynatrace para su posterior análisis. Consulte, the section called “Configure los ajustes de destino para Dynatrace”.	18 de abril de 2024
Versión de disponibilidad general (GA) para Snowflake como destino	Snowflake ahora está disponible de forma general como destino. Consulte the section called “Configure los ajustes de destino para Snowflake”.	17 de abril de 2024
Amazon Kinesis Data Firehose ahora se conoce como Amazon Data Firehose	Amazon Kinesis Data Firehose ha cambiado su nombre a Amazon Data Firehose. Consulte ¿Qué es Amazon Data Firehose?	9 de febrero de 2024
Se agregó Snowflake como	Puedes crear un arroyo Firehose con Snowflake como destino. Consulte the section called “Configure los ajustes de destino para Snowflake”.	19 de enero de 2024

Cambio	Descripción	Fecha de modificación
destino (versión preliminar pública)		
Se agregó la descompresión automática de los registros CloudWatch	Puedes habilitar la descompresión en transmisiones nuevas o existentes para enviar datos de CloudWatch Logs descomprimidos a los destinos de Firehose. Consulte the section called “Escribir usando registros CloudWatch ”.	15 de diciembre de 2023
Se agregó Splunk Observability Cloud como destino	Puedes crear una transmisión Firehose con Splunk Observability Cloud como destino. Consulte the section called “Configure los ajustes de destino para Splunk Observability Cloud”.	3 de octubre de 2023
Se agregó Amazon Managed Streaming para Apache Kafka como origen de datos	Ahora puede configurar Amazon MSK para enviar información a una transmisión de Firehose. Consulte the section called “Escritura mediante Amazon MSK”.	26 de septiembre de 2023
Se agregó compatibilidad con el tipo DocumentID para el destino del servicio OpenSearch	Si el destino de la transmisión de Firehose es OpenSearch Service, el tipo DocumentID indica el método para configurar el ID del documento. Los métodos admitidos son el ID del documento generado por Firehose y el ID del documento generado por el OpenSearch servicio. Consulte the section called “Configurar los ajustes de destino”.	10 de mayo de 2023
Se agregó compatibilidad con el particionamiento dinámico	Se ha añadido compatibilidad con la partición dinámica continua de los datos de streaming en Amazon Data Firehose. Consulte Particionamiento dinámico.	31 de agosto de 2021

Cambio	Descripción	Fecha de modificación
Se ha añadido un tema sobre los prefijos personalizados	Se agregó un tema sobre las expresiones que puede utilizar para crear un prefijo personalizado para los datos que se entregan en Amazon S3. Consulte Prefijos personalizados de Amazon S3.	20 de diciembre de 2018
Se ha añadido un nuevo tutorial sobre Amazon Data Firehose	Se agregó un tutorial que muestra cómo enviar registros de flujo de Amazon VPC a Splunk a través de Amazon Data Firehose. Consulte Tutorial: Ingiera los registros de flujo de VPC en Splunk mediante Amazon Data Firehose.	30 de octubre de 2018
Se agregaron cuatro nuevas regiones de Amazon Data Firehose	Se han añadido París, Mumbai, São Paulo y Londres. Para obtener más información, consulte Cuota de Amazon Data Firehose.	27 de junio de 2018
Se agregaron dos nuevas regiones de Amazon Data Firehose	Se han añadido Seúl y Montreal. Para obtener más información, consulte Cuota de Amazon Data Firehose.	13 de junio de 2018
Nueva característica de Kinesis Streams como origen	Se agregó Kinesis Streams como posible fuente de registros para una transmisión de Firehose. Para obtener más información, consulte Configurar el origen y el destino.	18 de agosto de 2017
Actualización de la documentación de la consola	Se actualizó el asistente de creación de transmisiones de Firehose. Para obtener más información, consulte Crea una transmisión de Firehose.	19 de julio de 2017
Nueva transformación de datos	Puede configurar Amazon Data Firehose para transformar los datos antes de entregarlos. Para obtener más información, consulte Transformación de datos en Amazon Data Firehose.	19 de diciembre de 2016

Cambio	Descripción	Fecha de modificación
Nuevo reintento de COPY de Amazon Redshift	Puede configurar Amazon Data Firehose para que vuelva a intentar ejecutar un comando COPY en su clúster de Amazon Redshift si se produce un error. Para obtener más información, consulte Crea una transmisión de Firehose, Conozca la entrega de datos de Amazon Data Firehose y Cuota de Amazon Data Firehose.	18 de mayo de 2016
Amazon Service, nuevo destino para Amazon Data Firehose OpenSearch	Puedes crear una transmisión de Firehose con Amazon OpenSearch Service como destino. Para obtener más información, consulte Crea una transmisión de Firehose, Conozca la entrega de datos de Amazon Data Firehose y Conceda a Amazon Data Firehose acceso a un destino de servicio público OpenSearch .	19 de abril de 2016
Nuevas CloudWatch métricas y funciones de solución de problemas mejoradas	Actualización de Supervisión de Amazon Data Firehose y Solución de problemas de Amazon Data Firehose.	19 de abril de 2016
Nuevo agente de Kinesis mejorado	Actualizado Escribir en Amazon Data Firehose mediante Kinesis Agent.	11 de abril de 2016
Nuevos agentes de Kinesis	Escribir en Amazon Data Firehose mediante Kinesis Agent añadido.	2 de octubre de 2015
Versión inicial	Publicación inicial de la guía para desarrolladores de Amazon Data Firehose.	4 de octubre de 2015

Glosario de AWS

Para ver la terminología más reciente de AWS, consulte el Glosario de AWS en la Referencia de Glosario de AWS.