aws

Guía del usuario de

AWS Elemental MediaStore

AWS Elemental MediaStore: Guía del usuario de

Table of Contents

¿Qué es MediaStore? 1
Conceptos y terminología 1
Servicios relacionados 3
Acceder MediaStore 3
Precios 4
Regiones y puntos de enlace 4
Configuración de AWS Elemental MediaStore 6
Registro para obtener una Cuenta de AWS 6
Crear un usuario administrativo 7
Introducción 8
Paso 1: Acceda a AWS Elemental MediaStore 8
Paso 2: Crear un contenedor 8
Paso 3: Cargar un objeto 9
Paso 4: Obtener acceso a un objeto 10
Contenedores 11
Reglas para los nombres de contenedor 11
Creación de un contenedor 11
Visualización de detalles del contenedor 13
Visualización de una lista de contenedores 14
Eliminación de un contenedor 15
Políticas 16
Políticas de contenedor 16
Visualización de una política de contenedor 17
Edición de una política de contenedor 18
Ejemplos de políticas de contenedor 19
Políticas CORS 26
Escenarios de casos de uso 26
Agregar una política de CORS 27
Visualización de una política de CORS 28
Edición de una política de CORS 29
Eliminación de una política de CORS 30
Solución de problemas 31
Ejemplos de políticas de CORS 32
Políticas de ciclo de vida de objetos 33

Componentes de una política de ciclo de vida de objetos 34
Agregar una política de ciclo de vida de objetos 41
Visualización de una política de ciclo de vida de objetos 43
Edición de una política de ciclo de vida de objetos 44
Eliminación de una política de ciclo de vida de objetos 45
Ejemplo de políticas de ciclo de vida de objetos 45
Políticas de métricas 50
Agregar una política de métricas 51
Visualización de una política de métricas 51
Edición de una política de métricas 51
Políticas de métricas de ejemplo 52
Carpetas 56
Reglas para los nombres de carpeta 57
Creación de una carpeta 57
Eliminación de una carpeta 57
Objetos 59
Carga de un objeto 59
Visualización de una lista 61
Visualización de detalles del objeto 64
Descarga de un objeto 65
Eliminación de objetos 66
Eliminación de un solo objeto 66
Vaciar un contenedor 67
Seguridad 69
Protección de datos 70
Cifrado de datos 71
Identity and Access Management 71
Público 72
Autenticación con identidades 72
Administración de acceso mediante políticas 76
Cómo funciona AWS Elemental MediaStore con IAM 79
Ejemplos de políticas basadas en identidades 87
Solución de problemas 90
Registro y monitoreo 92
Alarmas de Amazon CloudWatch 92
AWS CloudTrailRegistros de 92

AWS Trusted Advisor 93
Validación de conformidad 93
Resiliencia 94
Seguridad de infraestructuras 95
Prevención de la sustitución confusa entre servicios 95
Monitoreo y etiquetado 98
Registrar llamadas a la API con CloudTrail 99
MediaStoreInformación en CloudTrail 99
Ejemplo: entradas de archivos de registro 101
Monitorización con CloudWatch 102
CloudWatch Registros 103
CloudWatch Eventos 113
Métricas de CloudWatch 117
Etiquetado 121
Recursos compatibles en AWS Elemental MediaStore 122
Convenciones de nomenclatura y uso de las etiquetas 122
Administración de etiquetas 123
Uso de CDN 124
Cómo permitir que CloudFront obtenga acceso a un contenedor 124
Uso del control de acceso a origen (OAC) 125
Uso de secretos compartidos 125
Interacción de MediaStore con cachés HTTP 128
Solicitudes condicionales 128
Cuotas 130
Información relacionada 133
Historial de documentos 134
Glosario de AWS 139

¿Qué es AWS Elemental MediaStore?

AWS Elemental MediaStore es un servicio de creación y almacenamiento de vídeo que ofrece el alto rendimiento y la coherencia inmediata necesarios para la creación en directo. Con él MediaStore, puede administrar los activos de vídeo como objetos en contenedores para crear flujos de trabajo multimedia fiables y basados en la nube.

Para utilizar el servicio, se cargan objetos desde un origen, como, por ejemplo, un codificador o una fuente de datos, a un contenedor que se crea en MediaStore.

MediaStore es una excelente opción para almacenar archivos de vídeo fragmentados cuando se necesita una gran coherencia, lecturas y escrituras de baja latencia y la capacidad de gestionar grandes volúmenes de solicitudes simultáneas. Si no distribuyes vídeos en directo, considera utilizar Amazon Simple Storage Service (Amazon S3) en su lugar.

Temas

- MediaStore Conceptos y terminología de AWS Elemental
- Servicios relacionados
- Acceso a AWS Elemental MediaStore
- Precios de AWS Elemental MediaStore
- Regiones y puntos de enlace de AWS Elemental MediaStore

MediaStore Conceptos y terminología de AWS Elemental

ARN

Un Nombre de recurso de Amazon.

Cuerpo

Los datos que se van a cargar en un objeto.

Rango (de bytes)

Un subconjunto de datos de un objeto a los que se tiene acceso. Para obtener más información, consulte intervalo de la especificación HTTP.

Contenedor

Un espacio de nombres que contiene objetos. Un contenedor tiene un punto de enlace que se puede utilizar para escribir y recuperar objetos y asociar políticas de acceso.

Punto de conexión

Un punto de entrada al MediaStore servicio, indicado como URL raíz de HTTPS.

ETag

Una etiqueta de entidad, que es un hash de los datos del objeto.

Carpeta

Una división de un contenedor. Una carpeta puede contener objetos y otras carpetas.

Elemento

Un término que se usa para hacer referencia a los objetos y las carpetas.

Objeto

Un activo, similar a un objeto de Amazon S3. Los objetos son las entidades fundamentales que se almacenan en MediaStore. El servicio acepta todos los tipos de archivos.

Servicio de distribución

MediaStore se considera un servicio de origen porque es el punto de distribución para la entrega de contenido multimedia.

Ruta

Un identificador único para un objeto o una carpeta, que indica su ubicación en el contenedor.

Parte

Un subconjunto de datos (fragmento) de un objeto.

Política

Una política de IAM.

Recurso

Una entidad en AWS con la que puede trabajar. A cada recurso de AWS se le asigna un nombre de recurso de Amazon (ARN) que actúa como un identificador único. En MediaStore, este es el recurso y su formato ARN:

- Contenedor: `aws:mediastore:region:account-id:container/:containerName`

Servicios relacionados

- Amazon CloudFront es un servicio de red global de entrega de contenido (CDN) que entrega datos y vídeos de forma segura a sus espectadores. Utilice CloudFront para enviar contenido con el mejor desempeño posible. Para obtener más información, consulta la Guía para CloudFront desarrolladores de Amazon.
- AWS CloudFormation es un servicio que le ayuda a modelar y configurar los recursos de AWS. Usted crea una plantilla que describe todos los AWS recursos que desea (como los MediaStore contenedores) y AWS CloudFormation se encarga de aprovisionar y configurar esos recursos por usted. No es necesario crear y configurar individualmente los recursos de AWS ni averiguar qué depende de qué. AWS CloudFormation se encarga de todo eso. Para obtener más información, consulte la Guía del usuario de AWS CloudFormation.
- AWS CloudTrailes un servicio que le permite supervisar las llamadas realizadas a la CloudTrail API de su cuenta, incluidas las llamadas realizadas desde la consola de administración de AWS y otros servicios. AWS CLI Para obtener más información, consulte la Guía del usuario de AWS CloudTrail.
- Amazon CloudWatch es un servicio de supervisión de los recursos de la AWS nube y las aplicaciones en las que se ejecutaAWS. Usa CloudWatch Events para realizar un seguimiento de los cambios en el estado de los contenedores y objetos que contienen MediaStore. Para obtener más información, consulta la CloudWatch documentación de Amazon.
- AWS Identity and Access Management (IAM) es un servicio web que ayuda a controlar de forma segura el acceso de los usuarios a los recursos de AWS. Utilice IAM para controlar quién puede usar los recursos de AWS (autenticación), así como cuáles de ellos pueden usar y cómo pueden hacerlo (autorización). Para obtener más información, consulte Configuración de AWS Elemental MediaStore.
- Amazon Simple Storage Service (Amazon S3) es un almacenamiento de objetos creado para almacenar y recuperar cualquier cantidad de datos desde cualquier lugar. Para obtener más información, consulte la documentación de Amazon S3.

Acceso a AWS Elemental MediaStore

Puede acceder MediaStore mediante cualquiera de los siguientes métodos:

- Consola de administración de AWS: los procedimientos de esta guía explican cómo utilizar la consola de administración de AWS para realizar tareas para MediaStore. Para acceder MediaStore mediante la consola:

```
https://<region>.console.aws.amazon.com/mediastore/home
```

- AWS Command Line Interface— Para obtener más información, consulte la Guía AWS Command Line Interface del usuario. Para acceder MediaStore mediante el punto final CLI:

```
aws mediastore
```

- MediaStore API: si utilizas un lenguaje de programación para el que no hay un SDK disponible, consulta la referencia de la AWS Elemental MediaStore API para obtener información sobre las acciones de la API y sobre cómo realizar solicitudes a la API. Para acceder MediaStore mediante el punto final de la API REST:

```
https://mediastore.<region>.amazonaws.com
```

- SDK de AWS: si utiliza un lenguaje de programación para el que AWS proporciona un SDK, puede usar un SDK para obtener acceso a MediaStore. Los SDK simplifican la autenticación, se integran fácilmente con su entorno de desarrollo y proporcionan acceso sencillo a los comandos de MediaStore . Para obtener más información, consulte Herramientas para Amazon Web Services.
- Herramientas de AWS para Windows PowerShell: para obtener más información, consulte la Guía del AWS Tools for Windows PowerShell usuario.

Precios de AWS Elemental MediaStore

Al igual que con otros AWS productos, no hay contratos ni compromisos mínimos de uso MediaStore. Se cobra una tarifa por GB adquirido cuando se incorpora contenido al servicio y una tarifa mensual por GB por el contenido que se almacena en el servicio. Para obtener más información, consulte los MediaStore precios de AWS Elemental.

Regiones y puntos de enlace de AWS Elemental MediaStore

Para reducir la latencia de los datos en sus aplicaciones, MediaStore ofrece un punto de enlace regional para realizar su solicitud:

```
https://mediastore.<region>.amazonaws.com
```

Para ver la lista completa de las regiones MediaStore de AWS disponibles, consulte los MediaStore puntos de enlace y las cuotas de AWS Elemental en la Referencia general de AWS.

Configuración de AWS Elemental MediaStore

Esta sección le guía a través de los pasos necesarios para configurar el acceso de los usuarios a AWS Elemental MediaStore. Para obtener contexto e información adicional sobre la gestión de identidades y accesos a MediaStore, consulte Identidad y gestión de acceso para AWS Elemental MediaStore.

Antes de empezar a utilizar AWS Elemental MediaStore, realice los siguientes pasos.

Temas

- Registro para obtener una Cuenta de AWS
- Crear un usuario administrativo

Registro para obtener una Cuenta de AWS

Si no dispone de una Cuenta de AWS, siga estos pasos para crear una.

Cómo registrarse en una Cuenta de AWS

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.

   Al registrarse en una Cuenta de AWS, se crea un Usuario raíz de la cuenta de AWS. 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 administrativo y utilice únicamente el usuario raíz para realizar tareas que requieran acceso de usuario raíz.

AWS le enviará un correo electrónico de confirmación luego de completar el proceso de registro. Puede ver la actividad de la cuenta y administrar la cuenta en cualquier momento entrando en https://aws.amazon.com/ y seleccionando Mi cuenta.

Crear un usuario administrativo

Después de registrarse para obtener una Cuenta de AWS, asegure su Usuario raíz de la cuenta de AWS, habilite su AWS IAM Identity Center y cree un usuario administrativo para no utilizar el usuario raíz en las tareas cotidianas.

Protección de su Usuario raíz de la cuenta de AWS

1. Inicie sesión en la AWS Management Console como propietario de cuenta, elija Usuario raíz e ingrese el email de su Cuenta de AWS. En la siguiente página, escriba su contraseña.

   Para obtener ayuda para iniciar sesión con el usuario raíz, consulte Iniciar sesión como usuario raíz en la Guía del usuario de AWS Sign-In.
2. Active la autenticación multifactor (MFA) para el usuario raíz.

   Para obtener instrucciones, consulte Habilitar un dispositivo MFA virtual para el usuario raíz de la Cuenta de AWS (consola) en la Guía del usuario de IAM.

Creación de un usuario administrativo

1. Activar IAM Identity Center

   Para ver las instrucciones, consulte Habilitación de AWS IAM Identity Center en la Guía del usuario de AWS IAM Identity Center.
2. En el IAM Identity Center, conceda acceso administrativo a un usuario administrativo.

   Para ver un tutorial sobre cómo utilizar Directorio de IAM Identity Center como origen de identidad, consulte Configuración del acceso de usuarios con el Directorio de IAM Identity Center predeterminado en la Guía del usuario de AWS IAM Identity Center.

Cómo iniciar sesión como usuario administrativo

- Para iniciar sesión con el usuario del IAM Identity Center, utilice la URL de inicio de sesión que se envió a la dirección de correo electrónico cuando creó el usuario del IAM Identity Center.

  Para obtener ayuda para iniciar sesión con un usuario del IAM Identity Center, consulte Iniciar sesión en el portal de acceso de AWS en la Guía del Usuario de AWS Sign-In.

Introducción a AWS Elemental MediaStore

Este tutorial de introducción muestra cómo usar AWS Elemental MediaStore para crear un contenedor y cargar un objeto.

Temas

- Paso 1: Acceda a AWS Elemental MediaStore
- Paso 2: Crear un contenedor
- Paso 3: Cargar un objeto
- Paso 4: Obtener acceso a un objeto

Paso 1: Acceda a AWS Elemental MediaStore

Una vez que haya configurado su cuenta de AWS y creado los usuarios y roles, inicie sesión en la consola de AWS Elemental MediaStore.

Para acceder a AWS Elemental MediaStore

- Inicie sesión en la MediaStore consolaAWS Management Console y ábrala en https://console.aws.amazon.com/mediastore/.

> **Note**
>
> Puede iniciar sesión con cualquiera de las credenciales de IAM que ha creado para esta cuenta. Para obtener información sobre la creación de credenciales de IAM, consulte Configuración de AWS Elemental MediaStore.

Paso 2: Crear un contenedor

Los contenedores de AWS Elemental se utilizan MediaStore para almacenar las carpetas y los objetos. Puede utilizar los contenedores para agrupar objetos relacionados del mismo modo en que usa un directorio para agrupar archivos en un sistema de archivos. No se le cobrará por crear contenedores; solo se le cobrará cuando cargue un objeto en un contenedor.

Para crear un contenedor

1. En la página Containers (Contenedores), elija Create container (Crear contenedor).
2. En Container name (Nombre del contenedor), escriba un nombre para el contenedor. Para obtener más información, consulte Reglas para los nombres de contenedor.
3. Elige Crear contenedor. AWS Elemental MediaStore añade el nuevo contenedor a una lista de contenedores. Inicialmente, el estado del contenedor es Creating (Creándose) y luego cambia a Active (Activo).

Paso 3: Cargar un objeto

Puede cargar objetos (de hasta 25 MB cada uno) en un contenedor o en una carpeta dentro de un contenedor. Para cargar un objeto en una carpeta, debe especificar la ruta a la carpeta. Si la carpeta ya existe, AWS Elemental MediaStore almacena el objeto en la carpeta. Si la carpeta no existe, el servicio la crea y, a continuación, almacena el objeto en ella.

> **Note**
>
> Los nombres de archivo de los objetos solo pueden contener letras, números, puntos (.), guiones bajos (_), tildes (~) y guiones (-).

Para cargar un objeto

1. En la página Containers (Contenedores), elija el nombre del contenedor que acaba de crear. Aparecerá la página de detalles del contenedor.
2. Elija Upload object (Cargar objeto).
3. En Target path (Ruta de destino), escriba una ruta para las carpetas. Por ejemplo, `premium/canada`. Si alguna de las carpetas de la ruta aún no existe, AWS Elemental la MediaStore crea automáticamente.
4. En Object (Objeto), elija Browse (Examinar).
5. Vaya a la carpeta correspondiente y, a continuación, elija el objeto que desea cargar.
6. Elija Open (Abrir) y, a continuación, Upload (Cargar).

Paso 4: Obtener acceso a un objeto

Es posible descargar los objetos en un punto de enlace especificado.

1. En la página Containers (Contenedores), elija el nombre del contenedor que tiene el objeto que desea descargar.
2. Si el objeto que desea descargar se encuentra en una subcarpeta, continúe eligiendo los nombres de las carpetas hasta que vea el objeto.
3. Elija el nombre del objeto.
4. En la página de detalles del objeto, elija Download (Descargar).

Contenedores en AWS ElementalMediaStore

Los contenedores de MediaStore se utilizan para almacenar las carpetas y los objetos. Los objetos relacionados se pueden agrupar en contenedores del mismo modo en que se usa un directorio para agrupar archivos en un sistema de archivos. No se le cobrará por crear contenedores; solo se le cobrará cuando cargue un objeto en un contenedor. Para obtener más información acerca de los cargos de, consulteAWS ElementalMediaStorePrecios.

Temas

- Reglas para los nombres de contenedor
- Creación de un contenedor
- Visualización de los detalles de un contenedor
- Visualización de una lista de contenedores
- Eliminación de un contenedor

Reglas para los nombres de contenedor

Al elegir un nombre para el contenedor, recuerde lo siguiente:

- El nombre debe ser único dentro del cuenta actual para la región de AWS actual.
- El nombre puede contener letras en mayúsculas y minúsculas, números y guiones bajos (_).
- El nombre debe tener entre 1 y 255 caracteres.
- Los nombres distinguen mayúsculas de minúsculas. Por ejemplo, puede tener un contenedor denominado `myContainer` y una carpeta con el nombre `mycontainer`, ya que esos nombres son únicos.
- No se puede cambiar el nombre de un contenedor una vez creado.

Creación de un contenedor

Puede crear hasta 100 contenedores por cuenta de AWS. Puede crear tantas carpetas como desee, siempre que no estén anidadas más de 10 niveles dentro de un contenedor. Asimismo, puede cargar tantos objetos como desee en cada contenedor.

> **Tip**
>
> También puede crear un contenedor automáticamente utilizando una plantilla de AWS CloudFormation. La plantilla de AWS CloudFormation administra los datos para cinco acciones de la API: creación de un contenedor, establecimiento del registro de acceso, actualización de la política de contenedor predeterminada, adición de una política de uso compartido de recursos entre orígenes (CORS) y adición de una política de ciclo de vida de objetos. Para obtener más información, consulte la Guía del usuario de AWS CloudFormation.

Para crear un contenedor (consola)

1. Abra el iconoMediaStoreConsola dehttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija Create container (Crear contenedor).
3. En Container name (Nombre del contenedor), escriba un nombre para el contenedor. Para obtener más información, consulte Reglas para los nombres de contenedor.
4. ElegirCreación de contenedor. AWS ElementalMediaStoreañade el contenedor nuevo a una lista de contenedores. Inicialmente, el estado del contenedor es Creating (Creándose) y luego cambia a Active (Activo).

Para crear un contenedor (AWS CLI)

- En la AWS CLI, utilice el comando `create-container`:

```
aws mediastore create-container --container-name ExampleContainer --region us-
west-2
```

En el siguiente ejemplo, se muestra el valor de retorno:

```
{
    "Container": {
        "AccessLoggingEnabled": false,
        "CreationTime": 1563557265.0,
        "Name": "ExampleContainer",
        "Status": "CREATING",
        "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer"
```

```
    }
}
```

Visualización de los detalles de un contenedor

Los detalles de un contenedor incluyen la política de contenedor, el punto de enlace, el ARN y la hora de creación.

Para ver los detalles de un contenedor (consola)

1. Abra el iconoMediaStoreConsola dehttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor.

   Aparecerá la página de detalles del contenedor. Esta página se divide en dos secciones:

   - La sección Objects (Objetos), que enumera los objetos y las carpetas del contenedor.
   - La sección Container policy (Política del contenedor), que muestra la política basada en recursos que está asociada con este contenedor. Para obtener información sobre las políticas de recursos, consulte Políticas de contenedor.

Para ver los detalles de un contenedor (AWS CLI)

- En la AWS CLI, utilice el comando `describe-container`:

  ```
  aws mediastore describe-container --container-name ExampleContainer --region us-
  west-2
  ```

  En el siguiente ejemplo, se muestra el valor de retorno:

  ```
  {
      "Container": {
          "CreationTime": 1563558086.0,
          "AccessLoggingEnabled": false,
          "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
  ExampleContainer",
          "Status": "ACTIVE",
          "Name": "ExampleContainer",
          "Endpoint": "https://aaabbbcccdddee.data.mediastore.us-
  west-2.amazonaws.com"
  ```

```
    }
}
```

Visualización de una lista de contenedores

Puede ver una lista de todos los contenedores que están asociados a su cuenta.

Para ver una lista de contenedores (consola)

- Abra el iconoMediaStoreConsola dehttps://console.aws.amazon.com/mediastore/.

  Aparece la página Containers (Contenedores), que muestra todos los contenedores que están asociados a la cuenta.

Para ver una lista de contenedores (AWS CLI)

- En la AWS CLI, utilice el comando `list-containers`.

  ```
  aws mediastore list-containers --region us-west-2
  ```

  En el siguiente ejemplo, se muestra el valor de retorno:

  ```
  {
      "Containers": [
          {
              "CreationTime": 1505317931.0,
              "Endpoint": "https://aaabbbcccdddee.data.mediastore.us-
  west-2.amazonaws.com",
              "Status": "ACTIVE",
              "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
  ExampleLiveDemo",
              "AccessLoggingEnabled": false,
              "Name": "ExampleLiveDemo"
          },
          {
              "CreationTime": 1506528818.0,
              "Endpoint": "https://fffggghhhiiijj.data.mediastore.us-
  west-2.amazonaws.com",
              "Status": "ACTIVE",
  ```

```
            "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer",
            "AccessLoggingEnabled": false,
            "Name": "ExampleContainer"
        }
    ]
}
```

Eliminación de un contenedor

Un contenedor únicamente se puede eliminar si no tiene objetos.

Para eliminar un contenedor (consola)

1. Abra el iconoMediaStoreConsola dehttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija la opción situada a la izquierda del nombre del contenedor.
3. Elija Eliminar (Delete).

Para eliminar un contenedor (AWS CLI)

- En la AWS CLI, utilice el comando `delete-container`:

  ```
  aws mediastore delete-container --container-name=ExampleLiveDemo --region us-west-2
  ```

  Este comando no tiene ningún valor de retorno.

Políticas en AWS ElementalMediaStore

Puede aplicar una o varias de estas políticas a AWS Elemental.MediaStoreContenedor:

- Política de contenedor- Establece los derechos de acceso a todas las carpetas y objetos dentro del contenedor. MediaStoreestablece una política predeterminada que permite a los usuarios realizar todoMediaStoreoperaciones en el contenedor. Esta política especifica que todas las operaciones deben realizarse a través de HTTPS. Después de crear un contenedor, puede editar la política de contenedor.
- Política de uso compartido de recursos entre orígenes (CORS)- Permite a aplicaciones web clientes de un dominio interactuar con los recursos de un dominio distinto. MediaStoreno establece una política CORS predeterminada.
- Política de métricas- PermiteMediaStorepara enviar métricas a AmazonCloudWatch. MediaStoreno establece una política de métricas predeterminada.
- Política de ciclo de vida de objetos- Controla cuánto tiempo permanecen los objetos en unMediaStoreContenedor. MediaStoreno establece una política de ciclo de vida de objetos predeterminada.

Políticas de contenedores en AWS ElementalMediaStore

Cada contenedor tiene una política basada en recursos que rige los derechos de acceso a todas las carpetas y objetos de dicho contenedor. La política predeterminada, que se asocia automáticamente a todos los contenedores nuevos, permite el acceso a todos los AWS ElementalMediaStoreoperaciones en el contenedor. Especifica que este acceso requiere HTTPS para las operaciones. Después de crear un contenedor, puede editar la política que se asocia a dicho contenedor.

También puede utilizar una política de ciclo de vida de objetos que rija la fecha de vencimiento de objetos en un contenedor. Después de que los objetos alcancen la máxima antigüedad especificada, el servicio elimina los objetos del contenedor.

Temas

- Visualización de una política de contenedor
- Edición de una política de contenedor
- Ejemplos de políticas de contenedor

Visualización de una política de contenedor

Puede utilizar la consola o la AWS CLI para ver la política basada en recursos de un contenedor.

Para ver una política de contenedor (consola)

1. Abra el iconoMediaStoreConsola de enhttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor.

   Aparecerá la página de detalles del contenedor. La política se muestra en la sección Container policy (Política de contenedor).

Para ver una política de contenedor (AWS CLI)

- En la AWS CLI, utilice el comando `get-container-policy`:

  ```
  aws mediastore get-container-policy --container-name ExampleLiveDemo --region us-west-2
  ```

  En el siguiente ejemplo, se muestra el valor de retorno:

  ```
  {
    "Policy": {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Sid": "PublicReadOverHttps",
          "Effect": "Allow",
          "Principal": {
            "AWS": "arn:aws:iam::111122223333:root",
          },
          "Action": [
            "mediastore:GetObject",
            "mediastore:DescribeObject",
          ],
          "Resource": "arn:aws:mediastore:us-west-2:111122223333:container/ExampleLiveDemo/*",
          "Condition": {
            "Bool": {
              "aws:SecureTransport": "true"
            }
  ```

```
        }
      }
    ]
  }
}
```

Edición de una política de contenedor

Puede editar los permisos de la política de contenedor predeterminada, o puede crear una política nueva para sustituirla. Se necesita hasta cinco minutos para que la nueva política surta efecto.

Para editar una política de contenedor (consola)

1. Abra el iconoMediaStoreConsola de enhttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor.
3. Elija Edit policy (Editar política). Para ver ejemplos que ilustran cómo establecer diferentes permisos, consulte the section called “Ejemplos de políticas de contenedor”.
4. Realice los cambios apropiados y elija Save (Guardar).

Para editar una política de contenedor (AWS CLI)

1. Cree un archivo que defina la política del contenedor:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadOverHttps",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal": "*",
      "Resource": "arn:aws:mediastore:us-
west-2:111122223333:container/ExampleLiveDemo/*",
      "Condition": {
        "Bool": {
            "aws:SecureTransport": "true"
        }
      }
    }
  ]
```

```
}
```

2. En la AWS CLI, utilice el comando `put-container-policy`:

```
aws mediastore put-container-policy --container-name ExampleLiveDemo --
policy file://ExampleContainerPolicy.json --region us-west-2
```

Este comando no tiene ningún valor de retorno.

Ejemplos de políticas de contenedor

En los siguientes ejemplos se muestran políticas de contenedor creadas para distintos grupos de usuarios.

Temas

- Ejemplo de política de contenedor: política Valor predeterminado
- Ejemplo de política de contenedor: política Acceso de lectura público a través de HTTPS
- Ejemplo de política de contenedor: política Acceso de lectura público a través de HTTP o HTTPS
- Ejemplo de política de contenedor: política Acceso de lectura entre cuentas con HTTP habilitado
- Ejemplo de política de contenedor: política Acceso de lectura entre cuentas a través de HTTPS
- Ejemplo de política de contenedor: política Acceso de lectura entre cuentas a un rol
- Ejemplo de política de contenedor: política Acceso completo entre cuentas a un rol
- Ejemplo de política de contenedor: política Acceso restringido a direcciones IP específicas

Ejemplo de política de contenedor: política Valor predeterminado

Al crear un contenedor, AWS ElementalMediaStorele asocia automáticamente la siguiente política basada en recursos:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "MediaStoreFullAccess",
      "Action": [ "mediastore:*" ],
      "Principal":{
          "AWS" : "arn:aws:iam::<aws_account_number>:root"},
```

```
      "Effect": "Allow",
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
      "Condition": {
        "Bool": { "aws:SecureTransport": "true" }
      }
    }
  ]
}
```

La política está integrada en el servicio, por lo que no es necesario crearla. Sin embargo, sí puedeeditar la políticaen el contenedor si los permisos de la directiva predeterminada no se alinean con los permisos que desea utilizar para el contenedor.

La política predeterminada que se asigna a todos los contenedores nuevos permite el acceso a todas las operaciones de MediaStore en el contenedor. Especifica que este acceso requiere HTTPS para las operaciones.

Ejemplo de política de contenedor: política Acceso de lectura público a través de HTTPS

Este ejemplo de política permite a los usuarios recuperar un objeto mediante una solicitud HTTPS. Permite acceso de lectura a cualquier persona a través de una conexión SSL/TLS segura: usuarios autenticados y usuarios anónimos (los usuarios que no han iniciado sesión). La instrucción se denomina `PublicReadOverHttps`. Permite el acceso a las operaciones `GetObject` y `DescribeObject` en cualquier objeto (tal como especifica el carácter * al final de la ruta de recurso). Especifica que este acceso requiere HTTPS para las operaciones:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadOverHttps",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal": "*",
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
      "Condition": {
        "Bool": {
            "aws:SecureTransport": "true"
```

```
        }
      }
    }
  ]
}
```

Ejemplo de política de contenedor: política Acceso de lectura público a través de HTTP o HTTPS

Este ejemplo de política permite el acceso a las operaciones `GetObject` y `DescribeObject` en cualquier objeto (tal como especifica el carácter * al final de la ruta de recurso). Permite acceso de lectura a todo el mundo, incluidos todos los usuarios autenticados y los usuarios anónimos (los usuarios que no han iniciado sesión):

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadOverHttpOrHttps",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal": "*",
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
      "Condition": {
        "Bool": { "aws:SecureTransport": ["true", "false"] }
      }
    }
  ]
}
```

Ejemplo de política de contenedor: política Acceso de lectura entre cuentas con HTTP habilitado

Esta política permite a los usuarios recuperar un objeto mediante una solicitud HTTP. Concede este acceso a los usuarios autenticados con acceso entre cuentas. No es necesario que el objeto esté alojado en un servidor con un certificado SSL/TLS:

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

```
    "Sid" : "CrossAccountReadOverHttpOrHttps",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "arn:aws:iam::<other acct number>:root"
    },
    "Action" : [ "mediastore:GetObject", "mediastore:DescribeObject" ],
    "Resource" : "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
    "Condition" : {
      "Bool" : {
        "aws:SecureTransport" : [ "true", "false" ]
      }
    }
  } ]
}
```

Ejemplo de política de contenedor: política Acceso de lectura entre cuentas a través de HTTPS

Este ejemplo de política permite el acceso a la`GetObject`y`DescribeObject`en cualquier objeto (tal como especifica el carácter * al final de la ruta de recurso) que pertenEZCA al usuario raíz del especificado<other acct number>. Especifica que este acceso requiere HTTPS para las operaciones:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CrossAccountReadOverHttps",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal":{
        "AWS": "arn:aws:iam::<other acct number>:root"},
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
      "Condition": {
        "Bool": {
            "aws:SecureTransport": "true"
        }
      }
    }
  ]
}
```

Ejemplo de política de contenedor: política Acceso de lectura entre cuentas a un rol

La política de ejemplo permite el acceso a las operaciones `GetObject` y `DescribeObject` en cualquier objeto (tal como especifica el carácter * al final de la ruta de recurso) que pertenezca al <número de la cuenta propietaria>. Concede este acceso a los usuarios del <otro número de cuenta> si esa cuenta ha asumido la función que se ha especificado en <nombre de función>:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CrossAccountRoleRead",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal":{
        "AWS": "arn:aws:iam::<other acct number>:role/<role name>"},
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
    }
  ]
}
```

Ejemplo de política de contenedor: política Acceso completo entre cuentas a un rol

Este ejemplo de política permite acceso entre cuentas para actualizar cualquier objeto de la cuenta, siempre y cuando el usuario haya iniciado sesión sobre HTTP. También permite el acceso entre cuentas para eliminar, descargar y describir objetos a través de HTTP o HTTPS en una cuenta que ha asumido el rol especificado:

- La primera instrucción es `CrossAccountRolePostOverHttps`. Permite el acceso a la operación `PutObject` en cualquier objeto y permite este acceso a los usuarios de la cuenta especificada si esta ha asumido la función especificada en <nombre de función>. Especifica que este acceso requiere HTTPS para la operación (esta condición se debe incluir siempre al proporcionar acceso a `PutObject`).

  En otras palabras, cualquier entidad principal que tenga acceso entre cuentas puede obtener acceso a `PutObject`, pero solo por medio de HTTPS.
- La segunda instrucción es `CrossAccountFullAccessExceptPost`. Permite el acceso a todas las operaciones, excepto a `PutObject`, en cualquier objeto. Concede este acceso a los usuarios

de la cuenta especificada si esa cuenta ha asumido la función que se ha especificado en <nombre de función>. Este acceso no requiere HTTPS para las operaciones.

En otras palabras, cualquier cuenta que tenga acceso entre cuentas puede obtener acceso a `DeleteObject`, `GetObject` y así sucesivamente (pero no a `PutObject`), y puede hacerlo mediante HTTP o HTTPS.

Si no excluye `PutObject` de la segunda instrucción, la instrucción no será válida (ya que si incluye `PutObject`, debe establecer explícitamente HTTPS como condición).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CrossAccountRolePostOverHttps",
      "Effect": "Allow",
      "Action": "mediastore:PutObject",
      "Principal":{
        "AWS": "arn:aws:iam::<other acct number>:role/<role name>"},
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
      "Condition": {
        "Bool": {
            "aws:SecureTransport": "true"
        }
      }
    },
    {
      "Sid": "CrossAccountFullAccessExceptPost",
      "Effect": "Allow",
      "NotAction": "mediastore:PutObject",
      "Principal":{
        "AWS": "arn:aws:iam::<other acct number>:role/<role name>"},
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*"
    }
  ]
}
```

Ejemplo de política de contenedor: política Acceso restringido a direcciones IP específicas

Este ejemplo de política permite el acceso a todos los AWS ElementalMediaStoreoperaciones en los objetos del contenedor especificado. Sin embargo, la solicitud debe proceder del rango de direcciones IP especificado en la condición.

La condición de esta instrucción identifica el rango 198.51.100.* de direcciones IP permitidas en formato de Protocolo de Internet versión 4 (IPv4), con una excepción: 198.51.100.188.

El bloque `Condition` utiliza las condiciones `IpAddress` y `NotIpAddress`, y la clave de condición `aws:SourceIp`, que es una clave de condición general de AWS. Los valores de IPv4 `aws:sourceIp` utilizan la notación CIDR estándar. Para obtener más información, consulteOperadores de condición de dirección IPen la guía del usuario de IAM.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AccessBySpecificIPAddress",
            "Effect": "Allow",
            "Action": [
                "mediastore:GetObject",
                "mediastore:DescribeObject"
            ],
            "Principal": "*",
            "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/
<container name>/*",
            "Condition": {
                "IpAddress": {
                    "aws:SourceIp": [
                        "198.51.100.0/24"
                    ]
                },
                "NotIpAddress": {
                    "aws:SourceIp": "198.51.100.188/32"
                }
            }
        }
    ]
}
```

Políticas del uso compartido de recursos entre orígenes (CORS) en AWS ElementalMediaStore

El uso compartido de recursos entre orígenes (CORS) define una manera para que las aplicaciones web de los clientes cargadas en un dominio interactúen con los recursos de un dominio diferente. Con soporte CORS en AWS ElementalMediaStore, es posible desarrollar aplicaciones web del lado del cliente completas conMediaStorey permitir selectivamente el acceso de origen cruzado a suMediaStorede AWS.

> **Note**
>
> Si utiliza AmazonCloudFrontpara distribuir contenido de un contenedor que tiene una política CORS, asegúrese deconfigurar la distribución para AWS ElementalMediaStore(incluido el paso para editar el comportamiento de la caché para configurar CORS).

En esta sección, se proporciona información general acerca del CORS. En los subtemas, se describe cómo se puede habilitar el CORS con AWS Elemental.MediaStoreconsola o mediante programación mediante laMediaStoreAPI REST y SDK de AWS.

Temas

- Escenarios de casos de uso de CORS
- Agregar una política de CORS a un contenedor
- Visualización de una política de CORS
- Edición de una política de CORS
- Eliminación de una política de CORS
- Solución de problemas de CORS
- Ejemplos de políticas de CORS

Escenarios de casos de uso de CORS

A continuación, se muestran ejemplos de casos para el uso del CORS:

- Escenario 1: Suponga que distribuye streaming de vídeo en directo en un AWS ElementalMediaStoreContenedor denominadoLiveVideo. Los usuarios cargan el

punto de enlace del manifiesto de vídeo `http://livevideo.mediastore.ap-southeast-2.amazonaws.com` desde un origen específico, como `www.example.com`. Si desea utilizar unJavaScriptreproductor de vídeo para acceder a los vídeos procedentes de este contenedor a través de no autenticados`GET`y`PUT`solicitudes. Normalmente, un navegador se bloquearíaJavaScriptde permitir esas solicitudes, pero es posible establecer una política del CORS en el contenedor para que se habiliten de manera explícita estas solicitudes de`www.example.com`.

- Escenario 2: Suponga que desea alojar la misma transmisión en directo que en el Escenario 1 desde suMediaStorecontenedor, pero desea permitir solicitudes de cualquier origen. Puede configurar una política del CORS que especifique el asterisco (*) para los orígenes permitidos, de modo que las solicitudes procedentes de cualquier origen puedan obtener acceso al vídeo.

Agregar una política de CORS a un contenedor

En esta sección, se explica cómo añadir una configuración de uso compartido de recursos entre orígenes (CORS) a un AWS ElementalMediaStoreContenedor. CORS permite que las aplicaciones web clientes cargadas en un dominio puedan interactuar con los recursos de otro dominio.

Para configurar un contenedor para permitir las solicitudes entre orígenes, debe añadir una política del CORS al contenedor. Una política del CORS define las reglas que identifican los orígenes desde los que permitirá el acceso al contenedor, las operaciones (métodos HTTP) permitidas para cada origen y otro tipo de información específica de cada operación.

Cuando se añade una política del CORS al contenedor, las políticas del contenedor (que rigen los derechos de acceso al contenedor) seguirán aplicándose.

Para añadir política del CORS (Consola)

1. Abra el iconoMediaStoreConsola de enhttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor para el que desea crear la política del CORS.

   Aparecerá la página de detalles del contenedor.
3. En la sección Container CORS policy (Política del CORS del contenedor), elija Create CORS policy (Crear política del CORS).
4. Inserte la política en formato JSON y, a continuación, elija Save (Guardar).

Para añadir política del CORS (AWS CLI)

1. Cree un archivo que defina la política del CORS:

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "GET",
      "HEAD"
    ],
    "AllowedOrigins": [
      "*"
    ],
    "MaxAgeSeconds": 3000
  }
]
```

2. En la AWS CLI, utilice el comando `put-cors-policy`.

```
aws mediastore put-cors-policy --container-name ExampleContainer --cors-policy
 file://corsPolicy.json --region us-west-2
```

   Este comando no tiene ningún valor de retorno.

Visualización de una política de CORS

El uso compartido de recursos entre orígenes (CORS) define una manera para que las aplicaciones web de los clientes cargadas en un dominio interactúen con los recursos de un dominio diferente.

Para ver una política del CORS (consola)

1. Abra el iconoMediaStoreConsola de enhttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor cuya política del CORS desea ver.

   Aparece la página de detalles del contenedor, con la política del CORS en la sección Container CORS policy (Política del CORS del contenedor),

Para ver una política del CORS (AWS CLI)

- En la AWS CLI, utilice el comando `get-cors-policy`:

```
aws mediastore get-cors-policy --container-name ExampleContainer --region us-west-2
```

En el siguiente ejemplo, se muestra el valor de retorno:

```
{
    "CorsPolicy": [
        {
            "AllowedMethods": [
                "GET",
                "HEAD"
            ],
            "MaxAgeSeconds": 3000,
            "AllowedOrigins": [
                "*"
            ],
            "AllowedHeaders": [
                "*"
            ]
        }
    ]
}
```

Edición de una política de CORS

El uso compartido de recursos entre orígenes (CORS) define una manera para que las aplicaciones web de los clientes cargadas en un dominio interactúen con los recursos de un dominio diferente.

Para editar una política del CORS (consola)

1. Abra el iconoMediaStoreConsola de enhttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor cuya política del CORS desea editar.

   Aparecerá la página de detalles del contenedor.
3. En la sección Container CORS policy (Política del CORS del contenedor), elija Edit CORS policy (Editar política del CORS).

4. Realice los cambios en la política y, a continuación, elija Save (Guardar).

Para editar una política del CORS (AWS CLI)

1. Cree un archivo que defina la política del CORS actualizada:

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "GET",
      "HEAD"
    ],
    "AllowedOrigins": [
      "https://www.example.com"
    ],
    "MaxAgeSeconds": 3000
  }
]
```

2. En la AWS CLI, utilice el comando `put-cors-policy`.

```
aws mediastore put-cors-policy --container-name ExampleContainer --cors-policy
 file://corsPolicy2.json --region us-west-2
```

   Este comando no tiene ningún valor de retorno.

Eliminación de una política de CORS

El uso compartido de recursos entre orígenes (CORS) define una manera para que las aplicaciones web de los clientes cargadas en un dominio interactúen con los recursos de un dominio diferente. La eliminación de la política del CORS de un contenedor elimina los permisos para las solicitudes entre orígenes.

Para eliminar una política del CORS (consola)

1. Abra el iconoMediaStoreConsola de enhttps://console.aws.amazon.com/mediastore/.

2. En la página Containers (Contenedores), elija el nombre del contenedor cuya política del CORS desea eliminar.

   Aparecerá la página de detalles del contenedor.

3. En la sección Container CORS policy (Política del CORS del contenedor), elija Delete CORS policy (Eliminar política del CORS).
4. Elija Continue (Continuar) para confirmar y, a continuación, elija Save (Guardar).

Para eliminar una política del CORS (AWS CLI)

- En la AWS CLI, utilice el comando `delete-cors-policy`:

```
aws mediastore delete-cors-policy --container-name ExampleContainer --region us-west-2
```

  Este comando no tiene ningún valor de retorno.

Solución de problemas de CORS

Si detecta un comportamiento inesperado al obtener acceso a un contenedor que tiene una política del CORS, siga estos pasos para solucionar el problema.

1. Compruebe que la política del CORS está asociada al contenedor.

   Para obtener instrucciones, consulte the section called “Visualización de una política de CORS”.

2. Capture la solicitud y la respuesta completas con la herramienta que desee (como, por ejemplo, la consola para desarrolladores del navegador). Compruebe que la política del CORS asociada al contenedor incluye al menos una regla del CORS que coincida con los datos de la solicitud, tal y como se indica a continuación:

   a. Compruebe que la solicitud tiene un encabezado `Origin`.

      Si no se encuentra el encabezado, AWS ElementalMediaStoreno considera la solicitud como una solicitud entre orígenes y no devuelve los encabezados de respuesta CORS en la respuesta.

   b. Compruebe que el encabezado `Origin` de la solicitud coincide al menos con uno de los elementos `AllowedOrigins` de la regla `CORSRule` específica.

Los valores de esquema, host y puerto del encabezado de solicitud `Origin` deben coincidir con el elemento `AllowedOrigins` de la regla `CORSRule`. Por ejemplo, si establece que la regla `CORSRule` permita el origen `http://www.example.com`, los orígenes `https://www.example.com` y `http://www.example.com:80` de la solicitud no coinciden con el origen permitido en la configuración.

c. Compruebe que el método de la solicitud (o el método especificado en `Access-Control-Request-Method` en el caso de una solicitud de comprobación preliminar) sea uno de los elementos `AllowedMethods` de la misma regla `CORSRule`.

d. En el caso de una solicitud de comprobación preliminar, si la solicitud incluye un encabezado `Access-Control-Request-Headers`, verifique que la regla `CORSRule` incluya las entradas `AllowedHeaders` para cada valor en el encabezado `Access-Control-Request-Headers`.

Ejemplos de políticas de CORS

En los siguientes ejemplos, se muestran políticas del uso compartido de recursos entre orígenes (CORS).

Temas

- Ejemplos de políticas del CORS: Acceso de lectura para cualquier dominio
- Ejemplos de políticas del CORS: Acceso de lectura para un dominio específico

Ejemplos de políticas del CORS: Acceso de lectura para cualquier dominio

La siguiente política permite a una página web de cualquier dominio recuperar contenido desde un AWS ElementalMediaStoreContenedor. La solicitud incluye todos los encabezados HTTP del dominio de origen y el servicio responde únicamente a las solicitudes HTTP GET y HTTP HEAD procedentes del dominio de origen. Los resultados se almacenan en caché durante 3 000 segundos antes de que se entregue un conjunto de resultados nuevo.

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
```

```
      "GET",
      "HEAD"
    ],
    "AllowedOrigins": [
      "*"
    ],
    "MaxAgeSeconds": 3000
  }
]
```

Ejemplos de políticas del CORS: Acceso de lectura para un dominio específico

La siguiente política permite una página web de`https://www.example.com`para recuperar contenido de su AWS ElementalMediaStoreContenedor. La solicitud incluye todos los encabezados HTTP de `https://www.example.com` y el servicio responde únicamente a las solicitudes HTTP GET y HTTP HEAD procedentes de `https://www.example.com`. Los resultados se almacenan en caché durante 3 000 segundos antes de que se entregue un conjunto de resultados nuevo.

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "GET",
      "HEAD"
    ],
    "AllowedOrigins": [
      "https://www.example.com"
    ],
    "MaxAgeSeconds": 3000
  }
]
```

Políticas de ciclo de vida de objetos en AWS ElementalMediaStore

Para cada contenedor, puede crear una política de ciclo de vida de objetos que rija el tiempo que los objetos deben almacenarse en el contenedor. Cuando los objetos alcanzan la antigüedad máxima especificada, AWS ElementalMediaStoreelimina los objetos. Puede eliminar objetos cuando ya no sean necesarios para ahorrar los costes de almacenamiento.

También puede especificar que MediaStore debe mover los objetos a la clase de almacenamiento de acceso infrecuente (IA) después de que alcancen una cierta edad. Los objetos que se almacenan en la clase de almacenamiento IA tienen tasas de almacenamiento y recuperación diferentes a los objetos almacenados en la clase de almacenamiento estándar. Para obtener más información, consulte Precios de MediaStore.

Una política de ciclo de vida de objetos contiene reglas que determinan la vida útil de objetos por subcarpeta. (No puede asignar una política de ciclo de vida de objetos a objetos individuales). Puede asociar una única política del ciclo de vida de objetos a un contenedor, pero puede añadir hasta 10 reglas a cada política de ciclo de vida de objetos. Para obtener más información, consulte Componentes de una política de ciclo de vida de objetos.

Temas

- Componentes de una política de ciclo de vida de objetos
- Agregar una política de ciclo de vida de objetos a un contenedor
- Visualización de una política de ciclo de vida de objetos
- Edición de una política de ciclo de vida de objetos
- Eliminación de una política de ciclo de vida de objetos
- Ejemplo de políticas de ciclo de vida de objetos

Componentes de una política de ciclo de vida de objetos

Las políticas de ciclo de vida de objetos rigen el tiempo que los objetos permanecen en un AWS ElementalMediaStoreContenedor. Cada política de ciclo de vida de objetos se compone de una o varias reglas, que determinan la vida útil de objetos. Una regla puede aplicarse a una carpeta, a varias carpetas o al contenedor completo.

Puede asociar una política de ciclo de vida de objetos a un contenedor, y cada uno de los objetos de la política de ciclo de vida puede contener hasta 10 reglas. No se puede asignar una política de ciclo de vida de objetos a un objeto individual.

Reglas en una política de ciclo de vida de objetos

Puede crear tres tipos de reglas:

- Datos transitorios
- Eliminar objeto

- Transición de ciclo de vida

Datos transitorios

Una regla de datos transitorios establece que los objetos venzan en cuestión de segundos. Este tipo de regla solo se aplica a los objetos que se agregan al contenedor una vez que la política entra en vigor. MediaStore tarda hasta 20 minutos en aplicar la nueva política al contenedor.

Un ejemplo de una regla para datos transitorios tiene este aspecto:

```
        {
            "definition": {
                "path": [ {"wildcard": "Football/index*.m3u8"} ],
                "seconds_since_create": [
                    {"numeric": [">", 120]}
                ]
            },
            "action": "EXPIRE"
        },
```

Las reglas de datos transitorios tienen tres partes:

- `path`: siempre se establece en `wildcard`. Puede utilizar esta parte para definir qué objetos desea eliminar. Puede utilizar uno o varios comodines, representados por un asterisco (*). Cada comodín representa cualquier combinación de cero o más caracteres. Por ejemplo, `"path": [ {"wildcard": "Football/index*.m3u8"} ],` se aplica a todos los archivos de la carpeta `Football` que coinciden con el patrón `index*.m3u8` (como index.m3u8, index1.m3us8 e index123456.m3u8). Puede incluir hasta 10 rutas en una sola regla.
- `seconds_since_create`: siempre se establece en `numeric`. Puede especificar un valor de 1 a 300 segundos. También puede establecer el operador en mayor que (**>**) o mayor o igual que (**>=**).
- `action`: siempre se establece en `EXPIRE`.

Para las reglas de datos transitorios (los objetos vencen en cuestión de segundos), no hay ningún retraso entre el momento en que un objeto vence y la eliminación del mismo.

> **Note**
>
> Los objetos sujetos a una regla de datos transitorios no se incluyen en una respuesta `list-items`. Además, los objetos que caducan debido a una regla de datos transitoria no emiten unCloudWatchevento cuando caducan.

Eliminar objeto

Una regla de eliminación de objetos establece que los objetos venzan en cuestión de días. Este tipo de regla se aplica a todos los objetos del contenedor, incluso si se añadieron al contenedor antes de que se creara la política. MediaStore tarda hasta 20 minutos en aplicar la nueva política, pero los objetos pueden tardar hasta 24 horas en eliminarse del contenedor.

Un ejemplo de dos reglas para eliminar objetos tiene este aspecto:

```
        {
            "definition": {
                "path": [ { "prefix": "FolderName/" }  ],
                "days_since_create": [
                    {"numeric": [">" , 5]}
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [ { "wildcard": "Football/*.ts" }  ],
                "days_since_create": [
                    {"numeric": [">" , 5]}
                ]
            },
            "action": "EXPIRE"
        }
```

Las reglas de eliminación de objetos tienen tres partes:

- `path`: se establece en `prefix` o `wildcard`. No se puede utilizar `prefix` y `wildcard` en la misma regla. Si desea utilizar ambos, debe crear una regla para `prefix` y una regla distinta para `wildcard`, como se muestra en el ejemplo anterior.

- `prefix`: establezca la ruta de acceso en `prefix` si desea eliminar todos los objetos dentro de una determinada carpeta. Si el parámetro está vacío (`"path": [ { "prefix": "" } ],`), el destino son todos los objetos que se almacenan en cualquier lugar del contenedor actual. Puede incluir hasta 10 rutas `prefix` en una sola regla.
- `wildcard`: establezca la ruta de acceso en `wildcard` si desea eliminar objetos específicos basados en el nombre de archivo y/o tipo de archivo. Puede utilizar uno o varios comodines, representados por un asterisco (*). Cada comodín representa cualquier combinación de cero o más caracteres. Por ejemplo, `"path": [ {"wildcard": "Football/*.ts"} ],` se aplica a todos los archivos de la carpeta `Football` que coincidan con el patrón `*.ts` (como nombreArchivo.ts, nombreArchivo1.ts y nombreArchivo123456.ts). Puede incluir hasta 10 rutas `wildcard` en una sola regla.

- `days_since_create`: siempre se establece en `numeric`. Puede especificar un valor de 1 a 36.500 días. También puede establecer el operador en mayor que (**>**) o mayor o igual que (**>=**).
- `action`: siempre se establece en `EXPIRE`.

Para las reglas de eliminación de objetos (los objetos vencen en cuestión de días), es posible que haya un pequeño retardo desde que vence un objeto hasta que se elimina. Sin embargo, los cambios en la facturación se producen tan pronto como caduca el objeto. Por ejemplo, si una regla de ciclo de vida especifica 10 `days_since_create`, no se factura el objeto en la cuenta después de que el objeto tenga 10 días de antigüedad, incluso si el objeto aún no se ha eliminado.

Transición de ciclo de vida

Una regla de transición de ciclo de vida establece que los objetos se moverán a la clase de almacenamiento de acceso infrecuente (IA) después de que alcancen una cierta antigüedad, medida en días. Los objetos que se almacenan en la clase de almacenamiento IA tienen tasas de almacenamiento y recuperación diferentes a los objetos almacenados en la clase de almacenamiento estándar. Para obtener más información, consulte Precios de MediaStore.

Una vez que un objeto se ha movido a la clase de almacenamiento IA, no puede volver a moverlo a la clase de almacenamiento estándar.

La regla de transición del ciclo de vida se aplica a todos los objetos del contenedor, incluso si se añadieron al contenedor antes de que se creara la política. MediaStore tarda hasta 20 minutos en aplicar la nueva política, pero los objetos pueden tardar hasta 24 horas en eliminarse del contenedor.

Un ejemplo de una regla de transición de ciclo de vida es así:

```
        {
            "definition": {
                "path": [
                    {"prefix": "AwardsShow/"}
                ],
                "days_since_create": [
                    {"numeric": [">=" , 30]}
                ]
            },
            "action": "ARCHIVE"
        }
```

Las reglas de transición de ciclo de vida tienen tres partes:

- `path`: se establece en `prefix` o `wildcard`. No se puede utilizar `prefix` y `wildcard` en la misma regla. Si desea utilizar ambos, debe crear una regla para `prefix` y otra regla independiente para `wildcard`.
  - `prefix`: establecer la ruta de acceso a `prefix` si desea la transición de todos los objetos dentro de una carpeta particular a la clase de almacenamiento IA. Si el parámetro está vacío (`"path": [ { "prefix": "" } ],`), el destino son todos los objetos que se guardan en cualquier lugar del contenedor actual. Puede incluir hasta 10 rutas `prefix` en una sola regla.
  - `wildcard`: se establece la ruta de acceso a `wildcard` si desea la transición de objetos específicos a la clase de almacenamiento IA basado en el nombre de archivo y/o tipo de archivo. Puede utilizar uno o varios comodines, representados por un asterisco (*). Cada comodín representa cualquier combinación de cero o más caracteres. Por ejemplo, `"path": [ {"wildcard": "Football/*.ts"} ],` se aplica a todos los archivos de la carpeta `Football` que coincidan con el patrón `*.ts` (como nombreArchivo.ts, nombreArchivo1.ts y nombreArchivo123456.ts). Puede incluir hasta 10 rutas `wildcard` en una sola regla.
- `days_since_create`: siempre se establece en `"numeric": [">=" , 30]`.
- `action`: siempre se establece en `ARCHIVE`.

Ejemplo

Supongamos que un contenedor denominado `LiveEvents` tenga cuatro subcarpetas: `Football`, `Baseball`, `Basketball` y `AwardsShow`. La política de ciclo de vida de objetos asignada a la carpeta `LiveEvents` puede tener un aspecto similar al siguiente:

```
{
    "rules": [
         {
            "definition": {
                "path": [
                    {"prefix": "Football/"},
                    {"prefix": "Baseball/"}
                ],
                "days_since_create": [
                    {"numeric": [">" , 28]}
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [ { "prefix": "AwardsShow/" }  ],
                "days_since_create": [
                    {"numeric": [">=" , 15]}
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [ { "prefix": "" }  ],
                "days_since_create": [
                    {"numeric": [">" , 40]}
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [ { "wildcard": "Football/*.ts" }  ],
                "days_since_create": [
                    {"numeric": [">" , 20]}
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
```

```
                "path": [
                    {"wildcard": "Football/index*.m3u8"}
                ],
                "seconds_since_create": [
                    {"numeric": [">" , 15]}
                ]
            },
            "action": "EXPIRE"
        },
        {
                "definition": {
                    "path": [
                        {"prefix": "Program/"}
                    ],
                    "days_since_create": [
                        {"numeric": [">=" , 30]}
                    ]
                },
                "action": "ARCHIVE"
            }
    ]
}
```

La política anterior especifica los elementos siguientes:

- La primera regla indica a AWS ElementalMediaStorepara eliminar objetos almacenados en el`LiveEvents/Football`folder`LiveEvents/Baseball`una vez que tengan más de 28 días de antigüedad.
- La segunda regla indica al servicio que elimine los objetos almacenados en la carpeta `LiveEvents/AwardsShow` cuando tengan una antigüedad de 15 días o más.
- La tercera regla indica al servicio eliminar objetos almacenados en cualquier lugar del contenedor `LiveEvents` cuando tengan una antigüedad de 40 días. Esta regla se aplica a los objetos almacenados directamente en el contenedor `LiveEvents`, así como a los objetos almacenados en cualquiera de las cuatro subcarpetas del contenedor.
- La cuarta regla indica al servicio que elimine los objetos de la carpeta `Football` que coincidan con el patrón `*.ts` cuando tengan más de 20 días.
- La quinta regla indica al servicio que elimine objetos en el`Football`folder que coincide con el patrón`index*.m3u8`después de tener más de 15 segundos. MediaStoreelimina estos archivos 16 segundos después de colocarlos en el contenedor.

- La sexta regla indica al servicio que mueva los objetos de la carpeta `Program` a la clase de almacenamiento IA después de que tengan 30 días de antigüedad.

Para obtener más ejemplos de políticas de ciclo de vida de objetos, consulte Ejemplo de políticas de ciclo de vida de objetos.

Agregar una política de ciclo de vida de objetos a un contenedor

Una política de ciclo de vida de objetos le permite especificar el tiempo que se almacenan sus objetos en un contenedor. Puede establecer una fecha de vencimiento y después de la fecha de vencimiento: AWS ElementalMediaStoreelimina los objetos. El servicio tarda hasta 20 minutos en aplicar la nueva política al contenedor.

Para obtener información acerca de cómo crear una política de ciclo de vida, consulte Componentes de una política de ciclo de vida de objetos.

> **Note**
>
> Para las reglas de eliminación de objetos (los objetos vencen en cuestión de días), es posible que haya un pequeño retardo desde que vence un objeto hasta que se elimina. Sin embargo, los cambios en la facturación se producen tan pronto como caduca el objeto. Por ejemplo, si una regla de ciclo de vida especifica 10 `days_since_create`, no se factura el objeto en la cuenta después de que el objeto tenga 10 días de antigüedad, incluso si el objeto aún no se ha eliminado.

Para añadir una política de ciclo de vida de objetos (consola)

1. Abra el iconoMediaStoreConsola de:https://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor para el que desea crear la política de ciclo de vida de objetos.

   Aparecerá la página de detalles del contenedor.
3. En la sección Object lifecycle policy (Política de ciclo de vida de objeto) elija Create object lifecycle policy (Crear política de ciclo de vida de objetos).
4. Inserte la política en formato JSON y, a continuación, elija Save (Guardar).

Para añadir una política de ciclo de vida de objetos (AWS CLI)

1. Cree un archivo que defina la política de ciclo de vida de objetos:

```
{
    "rules": [
         {
            "definition": {
                "path": [
                    {"prefix": "Football/"},
                    {"prefix": "Baseball/"}
                ],
                "days_since_create": [
                    {"numeric": [">" , 28]}
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [
                    {"wildcard": "AwardsShow/index*.m3u8"}
                ],
                "seconds_since_create": [
                    {"numeric": [">" , 8]}
                ]
            },
            "action": "EXPIRE"
        }
    ]
}
```

2. En la AWS CLI, utilice el comando `put-lifecycle-policy`:

```
aws mediastore put-lifecycle-policy --container-name LiveEvents --lifecycle-
policy file://LiveEventsLifecyclePolicy.json --region us-west-2
```

Este comando no tiene ningún valor de retorno. El servicio asocia la política especificada al contenedor.

Visualización de una política de ciclo de vida de objetos

Una política de ciclo de vida de objetos especifica cuánto tiempo deben almacenarse los objetos en un contenedor.

Para ver una política de ciclo de vida de objetos (consola)

1. Abra el iconoMediaStoreConsola de:https://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor cuya política de ciclo de vida de objetos desea ver.

   Aparecerá la página de detalles del contenedor, con la política de ciclo de vida de objetos en la sección Object lifecycle policy (Política de ciclo de vida de objetos).

Para ver una política de ciclo de vida de objetos (AWS CLI)

- En la AWS CLI, utilice el comando `get-lifecycle-policy`:

```
aws mediastore get-lifecycle-policy --container-name LiveEvents --region us-west-2
```

En el siguiente ejemplo, se muestra el valor de retorno:

```
{
    "LifecyclePolicy": "{
        "rules": [
            {
                "definition": {
                    "path": [
                        {"prefix": "Football/"},
                        {"prefix": "Baseball/"}
                    ],
                    "days_since_create": [
                        {"numeric": [">" , 28]}
                    ]
                },
             "action": "EXPIRE"
            }
        ]
    }"
}
```

Edición de una política de ciclo de vida de objetos

No se puede editar una política de ciclo de vida de objetos existente. Sin embargo, puede cambiar una política existente cargando una política de sustitución. El servicio tarda hasta 20 minutos en aplicar la política actualizada al contenedor.

Para editar una política de ciclo de vida de objetos (consola)

1. Abra el iconoMediaStoreConsola de:https://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor cuya política de ciclo de vida de objetos desea editar.

   Aparecerá la página de detalles del contenedor.
3. En la sección Object lifecycle policy (Política de ciclo de vida de objetos) elija Edit object lifecycle policy (Editar política de ciclo de vida de objetos).
4. Realice los cambios en la política y, a continuación, elija Save (Guardar).

Para editar una política de ciclo de vida de objetos (AWS CLI)

1. Cree un archivo que defina la política de ciclo de vida de objetos actualizada:

```
{
    "rules": [
         {
            "definition": {
                "path": [
                    {"prefix": "Football/"},
                    {"prefix": "Baseball/"}
                    {"prefix": "Basketball/"}
                ],
                "days_since_create": [
                    {"numeric": [">" , 28]}
                ]
            },
            "action": "EXPIRE"
        }
    ]
}
```

2. En la AWS CLI, utilice el comando `put-lifecycle-policy`:

```
aws mediastore put-lifecycle-policy --container-name LiveEvents --lifecycle-
policy file://LiveEvents2LifecyclePolicy --region us-west-2
```

Este comando no tiene ningún valor de retorno. El servicio asocia la política especificada al contenedor, sustituyendo la política anterior.

Eliminación de una política de ciclo de vida de objetos

Cuando elimina una política de ciclo de vida de un objeto, el servicio tarda hasta 20 minutos en aplicar el cambio al contenedor.

Para eliminar una política de ciclo de vida de objetos (consola)

1. Abra el iconoMediaStoreConsola de:https://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor cuya política de ciclo de vida de objetos desea eliminar.

   Aparecerá la página de detalles del contenedor.
3. En la sección Object lifecycle policy (Política de ciclo de vida de objetos) elija Delete lifecycle policy (Eliminar política de ciclo de vida de objetos).
4. Elija Continue (Continuar) para confirmar y, a continuación, elija Save (Guardar).

Para eliminar una política de ciclo de vida de objetos (AWS CLI)

- En la AWS CLI, utilice el comando `delete-lifecycle-policy`:

```
aws mediastore delete-lifecycle-policy --container-name LiveEvents --region us-
west-2
```

Este comando no tiene ningún valor de retorno.

Ejemplo de políticas de ciclo de vida de objetos

En los ejemplos siguientes se muestran las políticas de ciclo de vida de objetos.

Temas

- Ejemplo de política de ciclo de vida de objeto: Caducidad en cuestión de segundos
- Ejemplo de política de ciclo de vida de objeto: Caducidad en días
- Ejemplo de política de ciclo de vida de objeto: Transición a clase de almacenamiento de acceso infrecuente
- Ejemplo de política de ciclo de vida de objeto: Múltiples reglas
- Ejemplo de política de ciclo de vida de objeto: Vaciar contenedor

Ejemplo de política de ciclo de vida de objeto: Caducidad en cuestión de segundos

La siguiente política especifica que MediaStore elimina objetos que coinciden con todos los criterios siguientes:

- El objeto se agrega al contenedor una vez que la política entra en vigor.
- El objeto se almacena en la carpeta `Football`.
- El objeto tiene una extensión de archivo de `m3u8`.
- El objeto ha estado en el contenedor durante más de 20 segundos.

```
{
    "rules": [
        {
            "definition": {
                "path": [
                    {"wildcard": "Football/*.m3u8"}
                ],
                "seconds_since_create": [
                    {"numeric": [ ">", 20 ]}
                ]
            },
            "action": "EXPIRE"
        }
    ]
}
```

Ejemplo de política de ciclo de vida de objeto: Caducidad en días

La siguiente política especifica que MediaStore elimina objetos que coinciden con todos los criterios siguientes:

- El objeto se almacena en la carpeta `Program`
- El objeto tiene una extensión de archivo de `ts`
- El objeto ha estado en el contenedor durante más de 5 días

```
{
    "rules": [
        {
            "definition": {
                "path": [
                    {"wildcard": "Program/*.ts"}
                ],
                "days_since_create": [
                    {"numeric": [ ">", 5 ]}
                ]
            },
            "action": "EXPIRE"
        }
    ]
}
```

Ejemplo de política de ciclo de vida de objeto: Transición a clase de almacenamiento de acceso infrecuente

La siguiente política especifica que MediaStore mueve objetos a la clase de almacenamiento de acceso infrecuente (IA) cuando tienen 30 días de antigüedad. Los objetos que se almacenan en la clase de almacenamiento IA tienen tasas de almacenamiento y recuperación diferentes a los objetos almacenados en la clase de almacenamiento estándar.

El campo `days_since_create` debe establecerse en `"numeric": [">=" ,30]`.

```
{
    "rules": [
        {
            "definition": {
                "path": [
                    {"prefix": "Football/"},
                    {"prefix": "Baseball/"}
                ],
                "days_since_create": [
                    {"numeric": [">=" , 30]}
```

```
                ]
            },
            "action": "ARCHIVE"
        }
    ]
}
```

Ejemplo de política de ciclo de vida de objeto: Múltiples reglas

La siguiente política especifica que MediaStore hace lo siguiente:

- Mover los objetos almacenados en la carpeta `AwardsShow` a la clase de almacenamiento de acceso infrecuente (IA) después de 30 días
- Eliminar objetos que tienen una extensión de archivo de `m3u8` y se almacenan en la carpeta `Football` después de 20 segundos
- Eliminar objetos almacenados en la carpeta `April` después de 10 días
- Eliminar objetos que tienen una extensión de archivo de `ts` y se almacenan en la carpeta `Program` después de 5 días

```
{
    "rules": [
        {
            "definition": {
                "path": [
                    {"prefix": "AwardsShow/"}
                ],
                "days_since_create": [
                    {"numeric": [ ">=" , 30 ]}
                ]
            },
            "action": "ARCHIVE"
        },
        {
            "definition": {
                "path": [
                    {"wildcard": "Football/*.m3u8"}
                ],
                "seconds_since_create": [
                    {"numeric": [ ">", 20 ]}
                ]
```

```
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [
                    {"prefix": "April"}
                ],
                "days_since_create": [
                    {"numeric": [ ">", 10 ]}
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [
                    {"wildcard": "Program/*.ts"}
                ],
                "days_since_create": [
                    {"numeric": [ ">", 5 ]}
                ]
            },
            "action": "EXPIRE"
        }
    ]
}
```

Ejemplo de política de ciclo de vida de objeto: Vaciar contenedor

La siguiente política de ciclo de vida de objeto especifica que MediaStore elimina todos los objetos del contenedor, incluidas las carpetas y subcarpetas, 1 día después de que se agregan al contenedor. Si el contenedor contiene objetos antes de aplicar esta política,MediaStoreelimina los objetos un día después de que la política entre en vigor. El servicio tarda hasta 20 minutos en aplicar la nueva política al contenedor.

```
{
    "rules": [
        {
            "definition": {
                "path": [
                    {"wildcard": "*"}
                ],
```

```
                "days_since_create": [
                    {"numeric": [ ">=", 1 ]}
                ]
            },
            "action": "EXPIRE"
        }
    ]
}
```

Políticas de métricas en AWS Elemental MediaStore

Para cada contenedor, puede añadir una política de métricas para permitir que AWS Elemental MediaStore envíe métricas a Amazon CloudWatch. Se necesitan hasta 20 minutos para que la nueva política surta efecto. Para obtener una descripción de cada MediaStore métrica, consulteMediaStore métricas.

Una política de métricas contiene lo siguiente:

- Un valor para habilitar o deshabilitar las métricas en el nivel de contenedor.
- Entre cero y cinco reglas que habilitan métricas en el nivel de objeto. Si la política contiene reglas, cada regla debe incluir lo siguiente:
  - Un grupo de objetos que define los objetos que se van a incluir en el grupo. La definición puede ser una ruta de acceso o un nombre de archivo, pero no puede tener más de 900 caracteres. Los caracteres válidos son: a-z, A-Z, 0-9, _ (guión bajo), = (igual), : (dos puntos), . (punto), - (guión), ~ (tilde), / (barra diagonal) y * (asterisco). Se admiten caracteres comodín (*).
  - Un nombre de grupo de objetos que permite hacer referencia al grupo de objetos. El nombre no puede tener más de 30 caracteres. Los caracteres válidos son a-z, A-Z, 0-9 y _ (guion bajo).

Si un objeto coincide con varias reglas, CloudWatch muestra un punto de datos para cada regla coincidente. Por ejemplo, si un objeto coincide con dos reglas denominadas`rule1` y`rule2`, CloudWatch muestra dos puntos de datos para estas reglas. El primero tiene una dimensión de `ObjectGroupName=rule1` y el segundo tiene una dimensión de `ObjectGroupName=rule2`.

Temas

- Agregar una política de métricas
- Visualización de una política de métricas
- Edición de una política de métricas

- Políticas de métricas de ejemplo

Agregar una política de métricas

Una política de métricas contiene reglas que dictan qué métricas MediaStore envía AWS Elemental a Amazon CloudWatch. Para obtener ejemplos de políticas de métricas, consulte Políticas de métricas de ejemplo.

Para agregar una política de métricas (consola)

1. Abra la MediaStore consola en https://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor para el que desea agregar una política de métricas.

   Aparecerá la página de detalles del contenedor.
3. En la sección Metric policy (Política de métricas), elija Create metric policy (Crear política de métricas).
4. Inserte la política en formato JSON y, a continuación, elija Save (Guardar).

Visualización de una política de métricas

Puede utilizar la consola o la AWS CLI para ver la política de métricas de un contenedor.

Para ver una política de métrica (consola)

1. Abra la MediaStore consola en https://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor.

   Aparecerá la página de detalles del contenedor. La política se muestra en la sección Metric policy (Política de métricas).

Edición de una política de métricas

Una política de métricas contiene reglas que dictan qué métricas MediaStore envía AWS Elemental a Amazon CloudWatch. Cuando edita una política de métricas existente, la nueva política tarda hasta 20 minutos en entrar en vigor. Para obtener ejemplos de políticas de métricas, consulte Políticas de métricas de ejemplo.

Para editar una política de métricas (consola)

1. Abra la MediaStore consola en https://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor.
3. En la sección Metric policy (Política de métricas) elija Edit metric policy (Editar política de métricas).
4. Realice los cambios apropiados y elija Save (Guardar).

Políticas de métricas de ejemplo

En los siguientes ejemplos se muestran políticas de métricas creadas para distintos casos de uso.

Temas

- Ejemplo de política de métricas: métricas de nivel de contenedor
- Ejemplo de política de métricas: métricas de nivel de ruta
- Ejemplo de política de métricas: métricas de nivel de contenedor y de ruta
- Ejemplo de política de métricas: métricas de nivel de ruta con caracteres comodín
- Ejemplo de política de métricas: métricas de nivel de ruta con reglas solapadas

Ejemplo de política de métricas: métricas de nivel de contenedor

Este ejemplo de política indica que AWS Elemental MediaStore debe enviar métricas a Amazon a CloudWatch nivel de contenedor. Por ejemplo, esto incluye la métrica `RequestCount`, que cuenta el número de solicitudes `Put` realizadas al contenedor. También puede establecer este valor en `DISABLED`.

Como esta política no contiene reglas, MediaStore no envía métricas a nivel de ruta. Por ejemplo, no puede ver cuántas solicitudes `Put` se han realizado en una carpeta concreta dentro de este contenedor.

```
{
   "ContainerLevelMetrics": "ENABLED"
}
```

Ejemplo de política de métricas: métricas de nivel de ruta

Este ejemplo de política indica que AWS Elemental no MediaStore debe enviar métricas a Amazon a CloudWatch nivel de contenedor. Además, MediaStore debe enviar métricas para los objetos de dos carpetas específicas: `baseball/saturday` y `football/saturday`. Las métricas de las solicitudes de MediaStore son las siguientes:

- Las solicitudes a la`baseball/saturday` carpeta tienen una CloudWatch dimensión de`ObjectGroupName=baseballGroup`.
- Las solicitudes a la carpeta `football/saturday` tienen una dimensión `ObjectGroupName=footballGroup`.

```
{
   "ContainerLevelMetrics": "DISABLED",
   "MetricPolicyRules": [
     {
       "ObjectGroup": "baseball/saturday",
       "ObjectGroupName": "baseballGroup"
     },
     {
       "ObjectGroup": "football/saturday",
       "ObjectGroupName": "footballGroup"
     }
   ]
}
```

Ejemplo de política de métricas: métricas de nivel de contenedor y de ruta

Este ejemplo de política indica que AWS Elemental MediaStore debe enviar métricas a Amazon a CloudWatch nivel de contenedor. Además, MediaStore debe enviar las métricas de los objetos en dos carpetas específicas:`baseball/saturday` y`football/saturday`. Las métricas de las solicitudes de MediaStore son las siguientes:

- Las solicitudes a la`baseball/saturday` carpeta tienen una CloudWatch dimensión de`ObjectGroupName=baseballGroup`.
- Las solicitudes a la`football/saturday` carpeta tienen una CloudWatch dimensión`ObjectGroupName=footballGroup`.

```
{
   "ContainerLevelMetrics": "ENABLED",
   "MetricPolicyRules": [
     {
       "ObjectGroup": "baseball/saturday",
       "ObjectGroupName": "baseballGroup"
     },
     {
       "ObjectGroup": "football/saturday",
       "ObjectGroupName": "footballGroup"
     }
   ]
}
```

Ejemplo de política de métricas: métricas de nivel de ruta con caracteres comodín

Este ejemplo de política indica que AWS Elemental MediaStore debe enviar métricas a Amazon a CloudWatch nivel de contenedor. Además, también MediaStore debe enviar las métricas de los objetos en función de su nombre de archivo. Un carácter comodín indica que los objetos se pueden almacenar en cualquier lugar del contenedor y que pueden tener cualquier nombre de archivo, siempre que termine con una extensión `.m3u8`.

```
{
   "ContainerLevelMetrics": "ENABLED",
   "MetricPolicyRules": [
     {
       "ObjectGroup": "*.m3u8",
       "ObjectGroupName": "index"
     }
   ]
 }
```

Ejemplo de política de métricas: métricas de nivel de ruta con reglas solapadas

Este ejemplo de política indica que AWS Elemental MediaStore debe enviar métricas a Amazon a CloudWatch nivel de contenedor. Además, MediaStore debe enviar las métricas de dos carpetas:`sports/football/saturday` y`sports/football`.

Las métricas de MediaStore las solicitudes a la`sports/football/saturday` carpeta tienen una CloudWatch dimensión de`ObjectGroupName=footballGroup1`. Como los objetos almacenados en la carpeta `sports/football` coinciden con ambas reglas, CloudWatch muestra dos puntos

de datos para estos objetos: uno con una dimensión `ObjectGroupName=footballGroup1` y el segundo con una dimensión `ObjectGroupName=footballGroup2`.

```
{
   "ContainerLevelMetrics": "ENABLED",
   "MetricPolicyRules": [
     {
       "ObjectGroup": "sports/football/saturday",
       "ObjectGroupName": "footballGroup1"
     },
     {
       "ObjectGroup": "sports/football",
       "ObjectGroupName": "footballGroup2"
     }
   ]
 }
```

Carpetas en AWS ElementalMediaStore

Las carpetas son divisiones de un contenedor. Utilícelas para subdividir el contenedor de la misma forma que crea subcarpetas para dividir una carpeta en un sistema de archivos. Se pueden crear hasta 10 niveles de carpetas (sin incluir el propio contenedor).

Las carpetas son opcionales; si lo desea, puede cargar los objetos directamente en un contenedor en lugar de en una carpeta. Sin embargo, las carpetas son una forma sencilla de organizar los objetos.

Para cargar un objeto en una carpeta, debe especificar la ruta a la carpeta. Si la carpeta ya existe, AWS ElementalMediaStorealmacena el objeto en la carpeta. Si la carpeta no existe, el servicio la crea y, a continuación, almacena el objeto en ella.

Supongamos que tiene un contenedor denominado`movies`, y subes un archivo llamado`mlaw.ts`con el camino`premium/canada`. AWS ElementalMediaStorealmacena el objeto en la subcarpeta canada debajo de la carpeta premium. Si no existe ninguna de las carpetas, el servicio creará las subcarpetas `premium` y `canada`; a continuación, almacenará el objeto en la subcarpeta `canada`. Si solo especifica el contenedor `movies` (sin ninguna ruta), el servicio almacena el objeto directamente en el contenedor.

AWS ElementalMediaStoreelimina automáticamente una carpeta cuando se elimina el último objeto de dicha carpeta. El servicio también elimina las carpetas vacías que están por encima de esa carpeta. Por ejemplo, supongamos que tiene una carpeta denominada premium que no contiene ningún archivo, pero que contiene la subcarpeta `canada`. La subcarpeta `canada` contiene un archivo denominado `mlaw.ts`. Si elimina el archivo `mlaw.ts`, el servicio elimina las carpetas `premium` y `canada`. Esta eliminación automática se aplica únicamente a las carpetas. El servicio no elimina contenedores vacíos.

Temas

- Reglas para los nombres de carpeta
- Creación de una carpeta
- Eliminación de una carpeta

Reglas para los nombres de carpeta

Al elegir un nombre para la carpeta, recuerde lo siguiente:

- El nombre solo debe contener los siguientes caracteres: letras mayúsculas (A-Z), minúsculas (a-z), números (0-9), puntos (.), guiones (-), tildes (~), guiones (_), signo de igual (=) y dos puntos (:).
- El nombre debe tener al menos un carácter de longitud. Nombres de carpetas vacíos (tales como`folder1//folder3/`) no se permiten.
- Los nombres distinguen mayúsculas de minúsculas. Por ejemplo, puede tener una carpeta denominada `myFolder` y una carpeta con el nombre `myfolder` en el mismo contenedor o carpeta, ya que esos nombres son únicos.
- El nombre debe ser único solo dentro del contenedor o la carpeta principal. Por ejemplo, puede crear una carpeta con el nombre `myfolder` en dos contenedores diferentes: `movies/myfolder` y `sports/myfolder`.
- El nombre puede tener el mismo nombre que su contenedor principal.
- No se puede cambiar el nombre de la carpeta una vez creada.

Creación de una carpeta

Puede crear carpetas al cargar los objetos. Para cargar un objeto en una carpeta, debe especificar la ruta a la carpeta. Si la carpeta ya existe, AWS ElementalMediaStorealmacena el objeto en la carpeta. Si la carpeta no existe, el servicio la crea y, a continuación, almacena el objeto en ella.

Para obtener más información, consulte the section called “Carga de un objeto”.

Eliminación de una carpeta

Solo se pueden eliminar las carpeta que están vacías; no es posible eliminar carpetas que contengan objetos.

AWS ElementalMediaStoreelimina automáticamente una carpeta cuando se elimina el último objeto de dicha carpeta. El servicio también elimina las carpetas vacías que están por encima de esa carpeta. Por ejemplo, supongamos que tiene una carpeta denominada `premium` que no contiene ningún archivo, pero que contiene la subcarpeta `canada`. La subcarpeta `canada` contiene un archivo denominado `mlaw.ts`. Si elimina el archivo `mlaw.ts`, el servicio elimina las carpetas `premium` y

`canada`. Esta eliminación automática se aplica únicamente a las carpetas. El servicio no elimina contenedores vacíos.

Para obtener más información, consulte Eliminación de un objeto.

Objetos de AWS ElementalMediaStore

AWS ElementalMediaStorelos recursos de reciben el nombreobjects. Puede cargar un objeto en un contenedor o en una carpeta dentro del contenedor.

En MediaStore, puede cargar, descargar y eliminar objetos:

- Cargar: añadir un objeto a un contenedor o a una carpeta. Esto no es lo mismo que crear el objeto. Debe crear los objetos de forma local antes de cargarlos en MediaStore.
- Descargar: copiar un objeto desde MediaStore a otra ubicación. Esto no elimina el objeto de MediaStore.
- Eliminar: eliminar un objeto de MediaStore definitivamente. Puede eliminar objetos individualmente, o puede añadir una política del ciclo de vida de objeto para eliminar automáticamente objetos dentro de un contenedor después de una duración especificada.

MediaStore acepta todos los tipos de archivos.

Temas

- Carga de un objeto
- Visualización de una lista de objetos
- Visualización de los detalles de un objeto
- Descarga de un objeto
- Eliminación de objetos

Carga de un objeto

Puede cargar objetos en un contenedor o en una carpeta dentro de un contenedor. Para cargar un objeto en una carpeta, debe especificar la ruta a la carpeta. Si la carpeta ya existe, AWS ElementalMediaStorealmacena el objeto en la carpeta. Si la carpeta no existe, el servicio la crea y, a continuación, almacena el objeto en ella. Para obtener más información sobre las carpetas consulte Carpetas en AWS ElementalMediaStore.

Puede utilizar la consola o la MediaStore de AWS CLI para cargar objetos.

MediaStore admite la transferencia fragmentada de objetos, lo que reduce la latencia haciendo que un objeto esté disponible para su descarga mientras se esté cargando. Para utilizar esta capacidad,

establezca la disponibilidad de carga del objeto en `streaming`. Puede establecer el valor de este encabezado cuando cargue el objeto mediante la API. Si no especifica este encabezado en su solicitud, MediaStore asigna el valor predeterminado de `standard` para la disponibilidad de carga del objeto.

Los tamaños de objetos no pueden superar los 25 MB para disponibilidad de carga estándar y los 10 MB para disponibilidad de carga de streaming.

> **Note**
>
> Los nombres de archivo de los objetos solo pueden contener letras, números, puntos (.), guiones bajos (_), tildes (~), guiones (-), signos de igual (=) y dos puntos (:).

Para cargar un objeto (consola)

1. Abra el iconoMediaStoreConsola de enhttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor. Aparecerá el panel de detalles del contenedor.
3. Elija Upload object (Cargar objeto).
4. En Target path (Ruta de destino), escriba una ruta para las carpetas. Por ejemplo, `premium/canada`. Si no existe alguna de las carpetas de la ruta que especifique, el servicio la crea automáticamente.
5. En la sección Object (Objeto), elija Browse (Examinar).
6. Vaya a la carpeta correspondiente y, a continuación, elija el objeto que desea cargar.
7. Elija Open (Abrir) y, a continuación, Upload (Cargar).

> **Note**
>
> Si ya existe un archivo con el mismo nombre en la carpeta seleccionada, el servicio sustituye el archivo original por el archivo cargado.

Para cargar un objeto (AWS CLI)

- En la AWS CLI, utilice el comando `put-object`. También puede incluir cualquiera de los siguientes parámetros: `content-type`, `cache-control` (para permitir que el autor de la

llamada controle el comportamiento de la caché del objeto) y `path` (para colocar el objeto en una carpeta dentro del contenedor).

> **Note**
>
> Después de cargar el objeto, no puede editar `content-type`, `cache-control` ni `path`.

```
aws mediastore-data put-object --endpoint https://
aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --body README.md --path /
folder_name/README.md --cache-control "max-age=6, public" --content-type binary/
octet-stream --region us-west-2
```

En el siguiente ejemplo, se muestra el valor de retorno:

```
{
    "ContentSHA256":
 "74b5fdb517f423ed750ef214c44adfe2be36e37d861eafe9c842cbe1bf387a9d",
    "StorageClass": "TEMPORAL",
    "ETag": "af3e4731af032167a106015d1f2fe934e68b32ed1aa297a9e325f5c64979277b"
}
```

Visualización de una lista de objetos

Puede utilizar AWS ElementalMediaStoreConsola de para ver los elementos (objetos y carpetas) almacenados en el nivel superior de un contenedor o una carpeta. Los elementos almacenados en una subcarpeta del contenedor o la carpeta actual no se mostrarán. Puede utilizar la AWS CLI para ver una lista de los objetos y las carpetas que contiene un contenedor, independientemente del número de carpetas o subcarpetas que contenga el contenedor.

Para ver una lista de los objetos de un contenedor específico (consola)

1. Abra el iconoMediaStoreConsola de enhttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor en el que se encuentra la carpeta que desea ver.
3. Elija el nombre de la carpeta en la lista.

Aparecerá una página de detalles, en la que se muestran todas las carpetas y los objetos que contiene la carpeta.

Para ver una lista de los objetos de una carpeta específica (consola)

1. Abra el iconoMediaStoreConsola de enhttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor en el que se encuentra la carpeta que desea ver.

   Aparecerá una página de detalles, en la que se muestran todas las carpetas y los objetos que contiene el contenedor.

Para ver una lista de los objetos y las carpetas de un contenedor específico (AWS CLI)

- En la AWS CLI, utilice el comando `list-items`:

```
aws mediastore-data list-items --endpoint https://
aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --region us-west-2
```

En el siguiente ejemplo, se muestra el valor de retorno:

```
{
    "Items": [
        {
            "ContentType": "image/jpeg",
            "LastModified": 1563571859.379,
            "Name": "filename.jpg",
            "Type": "OBJECT",
            "ETag":
 "543ab21abcd1a234ab123456a1a2b12345ab12abc12a1234abc1a2bc12345a12",
            "ContentLength": 3784
        },
        {
            "Type": "FOLDER",
            "Name": "ExampleLiveDemo"
        }
    ]
}
```

> **Note**
>
> Los objetos sujetos a una regla `seconds_since_create` no se incluyen en una respuesta `list-items`.

Para ver una lista de los objetos y las carpetas de una carpeta específica (AWS CLI)

- En la AWS CLI, utilice el comando `list-items` y especifique el nombre de carpeta al final de la solicitud.

```
aws mediastore-data list-items --endpoint https://
aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --path /folder_name --
region us-west-2
```

En el siguiente ejemplo, se muestra el valor de retorno:

```
{
    "Items": [
        {
            "Type": "FOLDER",
            "Name": "folder_1"
        },
        {
            "LastModified": 1563571940.861,
            "ContentLength": 2307346,
            "Name": "file1234.jpg",
            "ETag":
 "111a1a22222a1a1a222abc333a444444b55ab1111ab2222222222ab333333a2b",
            "ContentType": "image/jpeg",
            "Type": "OBJECT"
        }
    ]
}
```

> **Note**
>
> Los objetos sujetos a una regla `seconds_since_create` no se incluyen en una respuesta `list-items`.

Visualización de los detalles de un objeto

Después de cargar un objeto, AWS ElementalMediaStorealmacena información detallada sobre él como, por ejemplo, la fecha de modificación, el tamaño del contenido, la ETag (etiqueta de entidad) y el tipo de contenido. Para obtener información sobre cómo se utilizan los metadatos de un objeto, consulte Interacción de MediaStore con cachés HTTP.

Para ver los detalles de un objeto (consola)

1. Abra el iconoMediaStoreConsola de enhttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor en el que se encuentra el objeto que desea ver.
3. Si el objeto que desea ver se encuentra en una carpeta, continúe eligiendo los nombres de las carpetas hasta que vea el objeto.
4. Elija el nombre del objeto.

   Aparecerá una página de detalles, en la que se muestra la información sobre el objeto.

Para ver los detalles de un objeto (AWS CLI)

- En la AWS CLI, utilice el comando `describe-object`:

```
aws mediastore-data describe-object --endpoint https://
aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --path /folder_name/
file1234.jpg --region us-west-2
```

  En el siguiente ejemplo, se muestra el valor de retorno:

```
{
    "ContentType": "image/jpeg",
    "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",
```

```
    "ContentLength": "2307346",
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3"
}
```

Descarga de un objeto

Puede utilizar la consola para descargar un objeto. Puede utilizar la AWS CLI para descargar un objeto o solo parte de él.

Para descargar un objeto (consola)

1. Abra el iconoMediaStoreConsola de enhttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor en el que se encuentra el objeto que desea descargar.
3. Si el objeto que desea descargar se encuentra en una carpeta, continúe eligiendo los nombres de las carpetas hasta que vea el objeto.
4. Elija el nombre del objeto.
5. En la página de detalles de Object (Objeto), elija Download (Descargar).

Para descargar un objeto (AWS CLI)

- En la AWS CLI, utilice el comando `get-object`:

```
aws mediastore-data get-object --endpoint https://
aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --path=/folder_name/
README.md README.md --region us-west-2
```

En el siguiente ejemplo, se muestra el valor de retorno:

```
{
    "ContentLength": "2307346",
    "ContentType": "image/jpeg",
    "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3",
    "StatusCode": 200
}
```

Para descargar parte de un objeto (AWS CLI)

- En la AWS CLI, utilice comando `get-object` y especifique un intervalo.

```
aws mediastore-data get-object --endpoint https://
aaabbbcccddddee.data.mediastore.us-west-2.amazonaws.com --path /folder_name/
README.md --range="bytes=0-100" README2.md --region us-west-2
```

En el siguiente ejemplo, se muestra el valor de retorno:

```
{
    "StatusCode": 206,
    "ContentRange": "bytes 0-100/2307346",
    "ContentLength": "101",
    "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",
    "ContentType": "image/jpeg",
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3"
}
```

Eliminación de objetos

AWS ElementalMediaStoreofrece diferentes opciones para eliminar objetos de contenedores:

- Eliminar un objeto individual. No se aplican cargos.
- Vaciar un contenedor para eliminar todos los objetos dentro de un contenedor a la vez. Como este proceso utiliza llamadas a la API, se aplican los cargos normales de la API.
- Agregar una política de ciclo de vida de objetos para eliminar objetos cuando tengan una cierta antigüedad. No se aplican cargos.

Eliminación de un objeto

Puede eliminar objetos de manera individual mediante la consola o la AWS CLI. Como alternativa, puede agregar una política de ciclo de vida de objetos para eliminar automáticamente objetos cuando tengan una determinada antigüedad en un contenedor, o bien puede vaciar un contenedor para eliminar todos los objetos dentro de ese contenedor.

> **Note**
>
> Cuando se elimina el único objeto de una carpeta, AWS ElementalMediaStoreelimina automáticamente la carpeta y cualquier carpeta vacía situada por encima de dicha carpeta. Por ejemplo, supongamos que tiene una carpeta denominada `premium` que no contiene ningún archivo, pero que contiene la subcarpeta `canada`. La subcarpeta `canada` contiene un archivo denominado `mlaw.ts`. Si elimina el archivo `mlaw.ts`, el servicio elimina las carpetas `premium` y `canada`.

Para eliminar un objeto (consola)

1. Abra el iconoMediaStoreConsola de enhttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor que tiene el objeto que desea eliminar.
3. Si el objeto que desea eliminar se encuentra en una carpeta, continúe eligiendo los nombres de las carpetas hasta que vea el objeto.
4. Elija la opción a la izquierda del nombre del objeto.
5. Elija Eliminar (Delete).

Para eliminar un objeto (AWS CLI)

- En la AWS CLI, utilice el comando `delete-object`.

  Ejemplo:

  ```
  aws mediastore-data --region us-west-2 delete-object --endpoint=https://
  aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --path=/folder_name/
  README.md
  ```

  Este comando no tiene ningún valor de retorno.

Vaciar un contenedor

Puede vaciar un contenedor para eliminar todos los objetos almacenados en el contenedor. Como alternativa, puede agregar una política de ciclo de vida de objetos para eliminar automáticamente los

objetos una vez que tengan una determinada antigüedad en un contenedor, o bien puede eliminar los objetos individualmente.

Para vaciar un contenedor (consola)

1. Abra el iconoMediaStoreConsola de enhttps://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores) elija la opción del contenedor que desea vaciar.
3. Elija Empty container (Vaciar contenedor). Aparece un mensaje de confirmación.
4. Confirme que desea vaciar el contenedor introduciendo el nombre del contenedor en el campo de texto y, a continuación, elijaVacío.

Seguridad en AWS Elemental MediaStore

En AWS, la seguridad en la nube es la máxima prioridad. Como cliente de AWS, se beneficiará de una arquitectura de red y de centros de datos diseñados para satisfacer los requisitos de seguridad de las organizaciones más exigentes.

La seguridad es una responsabilidad compartida entre 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 servicios de AWS en la Nube de AWS. AWS también le proporciona servicios que puede utilizar de forma segura. Los auditores externos prueban y verifican periódicamente la eficacia de nuestra seguridad como parte de los Programas de conformidad de AWS . Para obtener información sobre los programas de conformidad que se aplican a AWS Elemental MediaStore, consulte Servicios de AWS en el ámbito del programa de conformidad .
- Seguridad en la nube: su responsabilidad viene determinada por el servicio de AWS que utilice. También es responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y la normativa aplicables.

Esta documentación le ayuda a comprender cómo aplicar el modelo de responsabilidad compartida cuando se utiliza MediaStore. En los siguientes temas, se le mostrará cómo configurar MediaStore para satisfacer sus objetivos de seguridad y conformidad. También puede aprender a utilizar otros servicios de AWS que lo ayuden a monitorear y proteger los recursos de MediaStore.

Temas

- Protección de datos en AWS Elemental MediaStore
- Identidad y gestión de acceso para AWS Elemental MediaStore
- Registro y monitorización en AWS Elemental MediaStore
- Validación de conformidad para AWS Elemental MediaStore
- Resiliencia en AWS Elemental MediaStore
- Seguridad de la infraestructura en AWS Elemental MediaStore
- Prevención de la sustitución confusa entre servicios

Protección de datos en AWS Elemental MediaStore

El modelo de responsabilidad compartida de AWS se aplica a la protección de datos en AWS Elemental MediaStore. Como se describe en este modelo, AWS es responsable de proteger la infraestructura global que ejecuta la totalidad de Nube de AWS. Usted es responsable de mantener el control sobre el contenido alojado en esta infraestructura. Usted también es responsable de las tareas de administración y configuración de seguridad para los Servicios de AWS que utiliza. Para obtener más información sobre la privacidad de los datos, consulte las Preguntas frecuentes sobre la privacidad de datos. Para obtener información sobre la protección de datos en Europa, consulte la publicación de blog Modelo de responsabilidad compartida y GDPR de AWS en el Blog de seguridad de AWS.

Con fines de protección de datos, recomendamos proteger las credenciales de la cuenta de Cuenta de AWS y configurar cuentas de usuario individuales con AWS IAM Identity Center o AWS Identity and Access Management (IAM). De esta manera, solo se conceden a cada usuario los permisos necesarios para cumplir con sus obligaciones laborales. También recomendamos proteger sus datos de la siguiente manera:

- Utilice autenticación multifactor (MFA) en cada cuenta.
- Utilice SSL/TLS para comunicarse con los recursos de AWS. reSe recomienda el uso de TLS 1.2 y recomendamos TLS 1.3.
- Configure la API y el registro de actividad del usuario con AWS CloudTrail.
- Utilice las soluciones de cifrado de AWS, junto con todos los controles de seguridad predeterminados dentro de los Servicios de AWS.
- Utilice servicios de seguridad gestionados avanzados, como Amazon Macie, que lo ayuden a detectar y proteger los datos confidenciales almacenados en Amazon S3.
- Si necesita módulos criptográficos validados FIPS 140-2 al acceder a AWS a través de una interfaz de la línea de comandos o una API, utilice un punto de conexión de FIPS. Para obtener más información sobre los puntos de conexión de FIPS disponibles, consulte Estándar de procesamiento de la información federal (FIPS) 140-2.

Se recomienda encarecidamente no introducir nunca información confidencial o sensible, como, por ejemplo, direcciones de correo electrónico de clientes, en etiquetas o campos de formato libre, tales como el campo Nombre. Incluye las situaciones en las que debe trabajar con la MediaStore u otros Servicios de AWS a través de la consola, la API, la AWS CLI o los SDK de AWS. Cualquier dato que ingrese en etiquetas o campos de formato libre utilizados para nombres se puede emplear para los

registros de facturación o diagnóstico. Si proporciona una URL a un servidor externo, recomendamos encarecidamente que no incluya información de credenciales en la URL a fin de validar la solicitud para ese servidor.

Cifrado de datos

MediaStore cifra los contenedores y los objetos en reposo usando el algoritmo AES-256, estándar en el sector. Recomendamos usar MediaStore para proteger sus datos de las siguientes formas:

- Cree una política de contenedor para controlar los derechos de acceso a todas las carpetas y objetos de dicho contenedor. Para obtener más información, consulte the section called “Políticas de contenedor”.
- Cree una política de compartición de recursos entre orígenes (CORS) para permitir el acceso entre orígenes de forma selectiva a sus recursos MediaStore. En CORS, puede permitir a aplicaciones web clientes cargadas en un dominio interactuar con los recursos de un dominio distinto. Para obtener más información, consulte the section called “Políticas CORS”.

Identidad y gestión de acceso para AWS Elemental MediaStore

AWS Identity and Access Management (IAM) es un Servicio de AWS que ayuda a los administradores a controlar de forma segura el acceso a los recursos de AWS. Los administradores de IAM controlan quién puede estar autenticado (ha iniciado sesión) y autorizado (tiene permisos) para utilizar recursos de MediaStore. IAM es un servicio de AWS que se puede utilizar sin cargo adicional.

Temas

- Público
- Autenticación con identidades
- Administración de acceso mediante políticas
- Cómo funciona AWS Elemental MediaStore con IAM
- Ejemplos de políticas basadas en identidades de AWS Elemental MediaStore
- Solución de problemas de identidad y acceso a AWS Elemental MediaStore

Público

La forma en que utilice AWS Identity and Access Management (IAM) difiere en función del trabajo que realice en MediaStore.

Usuario de servicio: si utiliza el servicio de MediaStore para realizar su trabajo, su administrador le proporciona las credenciales y los permisos que necesita. A medida que utilice más características de MediaStore para realizar su trabajo, es posible que necesite permisos adicionales. Entender cómo se administra el acceso puede ayudarlo a solicitar los permisos correctos al administrador. Si no puede acceder a una característica en MediaStore, consulte Solución de problemas de identidad y acceso a AWS Elemental MediaStore.

Administrador de servicio: si está a cargo de los recursos de MediaStore en su empresa, es probable que tenga acceso completo a MediaStore. Su trabajo consiste en determinar a qué características y recursos de MediaStore deben acceder los usuarios del servicio. Luego, debe enviar solicitudes a su administrador de IAM para cambiar los permisos de los usuarios de su servicio. Revise la información de esta página para conocer los conceptos básicos de IAM. Para obtener más información sobre cómo su empresa puede utilizar IAM con MediaStore, consulte Cómo funciona AWS Elemental MediaStore con IAM.

Administrador de IAM: si es un administrador de IAM, es posible que quiera conocer más detalles sobre cómo escribir políticas para administrar el acceso a MediaStore. Para consultar ejemplos de políticas basadas en la identidad de MediaStore que puede utilizar en IAM, consulte Ejemplos de políticas basadas en identidades de AWS Elemental MediaStore.

Autenticación con identidades

La autenticación es la manera de iniciar sesión en AWS mediante credenciales de identidad. Debe estar autenticado (haber iniciado sesión en AWS) como el Usuario raíz de la cuenta de AWS, como un usuario de IAM o asumiendo un rol de IAM.

Puede iniciar sesión en AWS como una identidad federada mediante las credenciales proporcionadas a través de una fuente de identidad de AWS IAM Identity Center. Los usuarios (del Centro de identidades de IAM), la autenticación de inicio de sesión único de su empresa y sus credenciales de Google o Facebook son ejemplos de identidades federadas. Al iniciar sesión como una identidad federada, su administrador habrá configurado previamente la federación de identidades mediante roles de IAM. Cuando accede a AWS mediante la federación, está asumiendo un rol de forma indirecta.

Según el tipo de usuario que sea, puede iniciar sesión en la AWS Management Console o en el portal de acceso a AWS. Para obtener más información sobre el inicio de sesión en AWS, consulte Cómo iniciar sesión en su Cuenta de AWS en la Guía del usuario de AWS Sign-In.

Si accede a AWS mediante programación, AWS proporciona un kit de desarrollo de software (SDK) y una interfaz de la línea de comandos (CLI) para firmar criptográficamente las solicitudes mediante el uso de las credenciales. Si no usa las herramientas de AWS, debe firmar usted mismo las solicitudes. Para obtener más información sobre la firma de solicitudes, consulte Firma de solicitudes API de AWS en la Guía del usuario de IAM.

Independientemente del método de autenticación que utilice, es posible que deba proporcionar información de seguridad adicional. Por ejemplo, AWS le recomienda el uso de la autenticación multifactor (MFA) para aumentar la seguridad de su cuenta. Para obtener más información, consulte Autenticación multifactor en la Guía del usuario de AWS Single Sign-On y Uso de la autenticación multifactor (MFA) en AWS en la Guía del usuario de IAM.

Usuario raíz de cuenta de AWS

Cuando se crea una cuenta de AWS, se comienza con una identidad de inicio de sesión que tiene acceso completo a todos los servicios y recursos de AWS de la cuenta. Esta identidad recibe el nombre de usuario raíz de la Cuenta de AWS y se accede a ella iniciando sesión con la dirección de correo electrónico y la contraseña que utilizó para crear la cuenta. Recomendamos encarecidamente que no utilice el usuario raíz para sus tareas diarias. Proteja las credenciales del usuario raíz y utilícelas solo para las tareas que solo el usuario raíz pueda realizar. Para ver la lista completa de las tareas que requieren que inicie sesión como usuario raíz, consulte Tareas que requieren credenciales de usuario raíz en la Guía del usuario de IAM.

Identidad federada

Como práctica recomendada, solicite que los usuarios humanos, incluidos los que requieren acceso de administrador, utilicen la federación con un proveedor de identidades para acceder a los servicios de AWS utilizando credenciales temporales.

Una identidad federada es un usuario del directorio de usuarios de su empresa, un proveedor de identidad web, el AWS Directory Service, el directorio del Identity Center, o cualquier usuario que acceda a Servicios de AWS utilizando credenciales proporcionadas a través de una fuente de identidad. Cuando identidades federadas acceden a las Cuentas de AWS, asumen roles y los roles proporcionan credenciales temporales.

Para una administración de acceso centralizada, le recomendamos que utilice AWS Single Sign-On. Puede crear usuarios y grupos en el IAM Identity Center o puede conectarse y sincronizar con un conjunto de usuarios y grupos de su propia fuente de identidad para usarlos en todas sus aplicaciones y Cuentas de AWS. Para obtener más información, consulte ¿Qué es el Centro de identidades de IAM? en la Guía del usuario de AWS IAM Identity Center.

Usuarios y grupos de IAM

Un usuario de IAM es una identidad en su Cuenta de AWS que dispone de permisos específicos para una sola persona o aplicación. Siempre que sea posible, recomendamos emplear credenciales temporales, en lugar de crear usuarios de IAM que tengan credenciales de larga duración como contraseñas y claves de acceso. No obstante, si tiene casos de uso específicos que requieran credenciales de larga duración con usuarios de IAM, recomendamos rotar las claves de acceso. Para más información, consulte Rotar las claves de acceso periódicamente para casos de uso que requieran credenciales de larga duración en la Guía del Usuario de IAM.

Un grupo de IAM es una identidad que especifica un conjunto de usuarios de IAM. No puede iniciar sesión como grupo. Puede usar los grupos para especificar permisos para varios usuarios a la vez. Los grupos facilitan la administración de los permisos de grandes conjuntos de usuarios. Por ejemplo, podría tener un grupo cuyo nombre fuese IAMAdmins y conceder permisos a dicho grupo para administrar los recursos de IAM.

Los usuarios son diferentes de los roles. Un usuario se asocia exclusivamente a una persona o aplicación, pero la intención es que cualquier usuario pueda asumir un rol que necesite. Los usuarios tienen credenciales permanentes a largo plazo y los roles proporcionan credenciales temporales. Para más información, consulte Cuándo crear un usuario de IAM (en lugar de un rol) en la Guía del Usuario de IAM.

Roles de IAM

Un rol de IAM es una identidad de tu cuenta de AWS que dispone de permisos específicos. Es similar a un usuario de IAM, pero no está asociado a una determinada persona. Puede asumir temporalmente un rol de IAM en la AWS Management Consolecambiando de roles. Puede asumir un rol llamando a una operación de AWS CLI o de la API de AWS, o utilizando una URL personalizada. Para más información sobre los métodos para el uso de roles, consulte Uso de roles de IAM en la Guía del Usuario de IAM.

Los roles de IAM con credenciales temporales son útiles en las siguientes situaciones:

- Acceso de usuario federado: para asignar permisos a una identidad federada, puede crear un rol y definir sus permisos. Cuando se autentica una identidad federada, se asocia la identidad al rol y se le conceden los permisos define el rol. Para obtener información acerca de roles para federación, consulte Creación de un rol para un proveedor de identidades de terceros en la Guía del Usuario de IAM. Si utiliza el IAM Identity Center, debe configurar un conjunto de permisos. El IAM Identity Center correlaciona el conjunto de permisos con un rol en IAM para controlar a qué pueden acceder las identidades después de autenticarse. Para obtener información acerca de los conjuntos de permisos, consulte Conjuntos de permisos en la Guía del usuario de AWS IAM Identity Center.
- Permisos de usuario de IAM temporales: un usuario de IAM puede asumir un rol de IAM para recibir temporalmente permisos distintos que le permitan realizar una tarea concreta.
- Acceso entre cuentas: puede utilizar un rol de IAM para permitir que alguien (una entidad principal de confianza) de otra cuenta acceda a los recursos de la cuenta. Los roles son la forma principal de conceder acceso entre cuentas. No obstante, con algunos Servicios de AWS se puede asociar una política directamente a un recurso (en lugar de utilizar un rol como representante). Para obtener información sobre la diferencia entre los roles y las políticas basadas en recursos para el acceso entre cuentas, consulte Cómo los roles de IAM difieren de las políticas basadas en recursos en la Guía del usuario de IAM.
- Acceso entre servicios: algunos servicios de AWS utilizan características de otros Servicios de AWS. Por ejemplo, cuando realiza una llamada en un servicio, es común que ese servicio ejecute aplicaciones en Amazon EC2 o almacene objetos en Amazon S3. Es posible que un servicio haga esto usando los permisos de la entidad principal, usando un rol de servicio o usando un rol vinculado a servicios.
  - Reenviar sesiones de acceso (FAS): cuando utiliza un rol o un usuario de IAM para llevar a cabo acciones en AWS, se le considera una entidad principal. Cuando utiliza algunos servicios, es posible que realice una acción que desencadene otra acción en un servicio diferente. FAS utiliza los permisos de la entidad principal para llamar a un Servicio de AWS, combinados con el Servicio de AWS solicitante para realizar solicitudes a servicios posteriores. Las solicitudes de FAS solo se realizan cuando un servicio recibe una solicitud que requiere interacciones con otros Servicios de AWS o recursos para completarse. En este caso, debe tener permisos para realizar ambas acciones. Para obtener información sobre las políticas a la hora de realizar solicitudes de FAS, consulte Reenviar sesiones de acceso.
  - Rol de servicio: un rol de servicio es un rol de IAM que adopta un servicio para realizar acciones en su nombre. Un administrador de IAM puede crear, modificar y eliminar un rol de servicio

desde IAM. Para obtener más información, consulte Creación de un rol para delegar permisos a un Servicio de AWS en la Guía del usuario de IAM.

- Rol vinculado a servicios: un rol vinculado a servicios es un tipo de rol de servicio que está vinculado a un Servicio de AWS. El servicio puede asumir el rol para realizar una acción en su nombre. Los roles vinculados a servicios aparecen en su Cuenta de AWS y son propiedad del servicio. Un administrador de IAM puede ver, pero no editar, los permisos de los roles vinculados a servicios.

- Aplicaciones que se ejecutan en Amazon EC2: puede utilizar un rol de IAM que le permita administrar credenciales temporales para las aplicaciones que se ejecutan en una instancia de EC2 y realizan solicitudes a la AWS CLI o a la API de AWS. Es preferible hacerlo de este modo a almacenar claves de acceso en la instancia EC2. Para asignar un rol de AWS a una instancia de EC2 y ponerla a disposición de todas las aplicaciones, cree un perfil de instancia asociado a la instancia. Un perfil de instancia contiene el rol y permite a los programas que se ejecutan en la instancia EC2 obtener credenciales temporales. Para obtener más información, consulte Uso de un rol de IAM para conceder permisos a aplicaciones que se ejecutan en instancias de Amazon EC2 en la Guía del usuario de IAM.

Para obtener información sobre el uso de los roles de IAM, consulte Cuándo crear un rol de IAM (en lugar de un usuario) en la Guía del Usuario de IAM.

Administración de acceso mediante políticas

Para controlar el acceso en AWS, se crean políticas y se adjuntan a identidades o recursos de AWS. Una política es un objeto de AWS que, cuando se asocia a una identidad o un recurso, define sus permisos. AWS evalúa estas políticas cuando una entidad principal (sesión de rol, usuario o usuario raíz) realiza una solicitud. Los permisos en las políticas determinan si la solicitud se permite o se deniega. La mayoría de las políticas se almacenan en AWS como documentos JSON. Para obtener más información sobre la estructura y el contenido de los documentos de política JSON, consulte Información general de las políticas JSON en la Guía del Usuario de IAM.

Los administradores pueden utilizar las políticas JSON de AWS para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

De forma predeterminada, los usuarios y los roles no tienen permisos. Para conceder permiso a los usuarios para realizar acciones en los recursos que necesiten, un administrador de IAM puede crear políticas de IAM. A continuación, el administrador puede añadir las políticas de IAM a roles y los usuarios pueden asumirlos.

Las políticas de IAM definen permisos para una acción independientemente del método que se utilice para realizar la operación. Por ejemplo, suponga que dispone de una política que permite la acción `iam:GetRole`. Un usuario con dicha política puede obtener información del usuario de la consola, AWS CLI o la API de AWS.

Políticas basadas en identidades

Las políticas basadas en identidad son documentos de políticas de permisos JSON que puede adjuntar a una identidad, como un usuario, un grupo de usuarios o un rol de IAM. Estas políticas controlan qué acciones pueden realizar los usuarios y los roles, en qué recursos y en qué condiciones. Para obtener más información sobre cómo crear una política en función de identidad, consulte Creación de políticas de IAM en la Guía del usuario de IAM.

Las políticas basadas en identidad pueden clasificarse además como políticas insertadas o políticas administradas. Las políticas insertadas se integran directamente en un único usuario, grupo o rol. Las políticas administradas son políticas independientes que puede asociar a varios usuarios, grupos y roles de su Cuenta de AWS. Las políticas administradas incluyen las políticas administradas de AWS y las políticas administradas por el cliente. Para obtener más información sobre cómo elegir una política administrada o una política insertada, consulte Elegir entre políticas administradas y políticas insertadas en la Guía del usuario de IAM.

Políticas basadas en recursos

Las políticas basadas en recursos son documentos de política JSON que se asocian a un recurso. Ejemplos de políticas basadas en recursos son las políticas de confianza de roles de IAM y las políticas de bucket de Amazon S3. En los servicios que admiten políticas basadas en recursos, los administradores de servicios pueden utilizarlos para controlar el acceso a un recurso específico. Para el recurso al que se asocia la política, la política define qué acciones puede realizar una entidad principal especificada en ese recurso y en qué condiciones. Debe especificar una entidad principal en una política en función de recursos. Las entidades principales pueden incluir cuentas, usuarios, roles, usuarios federados o servicios de AWS.

Las políticas basadas en recursos son políticas insertadas que se encuentran en ese servicio. No se puede utilizar políticas de IAM administradas por AWS en una política basada en recursos.

Listas de control de acceso (ACL)

Las listas de control de acceso (ACL) controlan qué entidades principales (miembros de cuentas, usuarios o roles) tienen permisos para acceder a un recurso. Las ACL son similares a las políticas basadas en recursos, aunque no utilizan el formato de documento de política JSON.

Amazon S3, AWS WAF y Amazon VPC son ejemplos de servicios que admiten las ACL. Para obtener más información sobre las ACL, consulte Información general de Lista de control de acceso (ACL) en la Guía para Desarrolladores de Amazon Simple Storage Service.

Otros tipos de políticas

AWS admite otros tipos de políticas adicionales menos frecuentes. Estos tipos de políticas pueden establecer el máximo de permisos que los tipos de políticas más frecuentes le conceden.

- Límites de permisos: un límite de permisos es una característica avanzada que le permite establecer los permisos máximos que una política en función de identidad puede conceder a una entidad de IAM (usuario o rol de IAM). Puede establecer un límite de permisos para una entidad. Los permisos resultantes son la intersección de las políticas basadas en la identidad de la entidad y los límites de permisos. Las políticas basadas en recursos que especifiquen el usuario o rol en el campo `Principal` no estarán restringidas por el límite de permisos. Una denegación explícita en cualquiera de estas políticas anulará el permiso. Para obtener más información sobre los límites de los permisos, consulte Límites de permisos para las entidades de IAM en la Guía del Usuario de IAM.
- Políticas de control de servicio (SCP): las SCP son políticas de JSON que especifican los permisos máximos de una organización o una unidad organizativa en AWS Organizations. AWS Organizations es un servicio que le permite agrupar y administrar de manera centralizada varias cuentas de AWS que posea su empresa. Si habilita todas las características en una empresa, entonces podrá aplicar políticas de control de servicio (SCP) a una o todas sus cuentas. Una SCP limita los permisos para las entidades de las cuentas de miembros, incluido cada rootlong. Para más información sobre organizaciones y las SCP, consulte Funcionamiento de las SCP en la Guía del Usuario de AWS Organizations.
- Políticas de sesión: las políticas de sesión son políticas avanzadas que se pasan como parámetro cuando se crea una sesión temporal mediante programación para un rol o un usuario federado. Los permisos de la sesión resultantes son la intersección de las políticas basadas en identidad del rol y las políticas de la sesión. Los permisos también pueden proceder de una política en función de recursos. Una denegación explícita en cualquiera de estas políticas anulará el permiso. Para más información, consulte Políticas de sesión en la Guía del Usuario de IAM.

Varios tipos de políticas

Cuando se aplican varios tipos de políticas a una solicitud, los permisos resultantes son más complicados de entender. Para obtener información sobre cómo AWS decide si permite o no una

solicitud cuando hay varios tipos de políticas implicados, consulte Lógica de evaluación de políticas en la Guía del usuario de IAM.

Cómo funciona AWS Elemental MediaStore con IAM

Antes de utilizar IAM para administrar el acceso a MediaStore, conozca qué características de IAM se pueden utilizar con MediaStore.

Características de IAM que puede utilizar con AWS Elemental MediaStore

Características de IAM	Soporte de MediaStore
Políticas basadas en identidad	Sí
Políticas basadas en recursos	Sí
Acciones de políticas	Sí
Recursos de políticas	Sí
Claves de condición de política (específicas del servicio)	Sí
ACL	No
ABAC (etiquetas en políticas)	Parcial
Credenciales temporales	Sí
Permisos de entidades principales	Sí
Roles de servicio	Sí
Roles vinculados al servicio	No

Para obtener información general sobre cómo funcionan MediaStore y otros servicios de AWS con la mayoría de las características de IAM, consulte Servicios de AWS que funcionan con IAM en la Guía del usuario de IAM.

Políticas basadas en identidades para MediaStore

Compatibilidad con las políticas basadas en identidad	Sí

Las políticas basadas en identidad son documentos de políticas de permisos JSON que puede asociar a una identidad, como un usuario de IAM, un grupo de usuarios o un rol. Estas políticas controlan qué acciones pueden realizar los usuarios y los roles, en qué recursos y en qué condiciones. Para obtener más información sobre cómo crear una política basada en identidad, consulte Creación de políticas de IAM en la Guía del usuario de IAM.

Con las políticas basadas en identidad de IAM, puede especificar las acciones y los recursos permitidos o denegados, así como las condiciones en las que se permiten o deniegan las acciones. No es posible especificar la entidad principal en una política basada en identidad porque se aplica al usuario o rol al que está adjunto. Para obtener más información sobre los elementos que puede utilizar en una política JSON, consulte Referencia de los elementos de las políticas JSON de IAM en la Guía del usuario de IAM.

Ejemplos de políticas basadas en identidades de MediaStore

Para ver ejemplos de políticas basadas en identidad de MediaStore, consulte Ejemplos de políticas basadas en identidades de AWS Elemental MediaStore.

Políticas basadas en recursos de MediaStore

Compatibilidad con las políticas basadas en recursos	Sí

Las políticas basadas en recursos son documentos de políticas JSON que se asocian a un recurso. Ejemplos de políticas basadas en recursos son las políticas de confianza de roles de IAM y las políticas de bucket de Amazon S3. En los servicios que admiten políticas basadas en recursos, los administradores de servicios pueden utilizarlos para controlar el acceso a un recurso específico. Para el recurso al que se asocia la política, la política define qué acciones puede realizar una entidad principal especificada en ese recurso y en qué condiciones. Debe especificar una entidad principal en una política en función de recursos. Las entidades principales pueden incluir cuentas, usuarios, roles, usuarios federados o servicios de AWS.

Para habilitar el acceso entre cuentas, puede especificar toda una cuenta o entidades de IAM de otra cuenta como la entidad principal de una política en función de recursos. Añadir a una política en función de recursos una entidad principal entre cuentas es solo una parte del establecimiento de una relación de confianza. Cuando la entidad principal y el recurso se encuentran en Cuentas de AWS diferentes, un administrador de IAM de la cuenta de confianza también debe conceder a la entidad principal (usuario o rol) permiso para acceder al recurso. Para conceder el permiso, adjunte la entidad a una política basada en identidad. Sin embargo, si la política en función de recursos concede el acceso a una entidad principal de la misma cuenta, no es necesaria una política en función de identidad adicional. Para más información, consulte Cómo los roles de IAM difieren de las políticas basadas en recursos en la Guía del Usuario de IAM.

> **Note**
>
> MediaStore también admite políticas de contenedor que definen qué entidades principales (cuentas, usuarios, roles y usuarios federados) pueden realizar acciones en el contenedor. Para obtener más información, consulte Políticas de contenedor.

Acciones de políticas para MediaStore

Admite acciones de política	Sí

Los administradores pueden utilizar las políticas JSON de AWS para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Action` de una política JSON describe las acciones que puede utilizar para permitir o denegar el acceso en una política. Las acciones de la política generalmente tienen el mismo nombre que la operación de API de AWS asociada. Hay algunas excepciones, como acciones de solo permiso que no tienen una operación de API coincidente. También hay algunas operaciones que requieren varias acciones en una política. Estas acciones adicionales se denominan acciones dependientes.

Incluya acciones en una política para conceder permisos y así llevar a cabo la operación asociada.

Para ver una lista de las acciones de MediaStore, consulte Acciones definidas por AWS Elemental MediaStore en la Referencia de autorizaciones de servicio.

Las acciones de políticas de MediaStore utilizan el siguiente prefijo antes de la acción:

```
mediastore
```

Para especificar varias acciones en una única instrucción, sepárelas con comas.

```
"Action": [
      "mediastore:action1",
      "mediastore:action2"
         ]
```

Para ver ejemplos de políticas basadas en identidad de MediaStore, consulte Ejemplos de políticas basadas en identidades de AWS Elemental MediaStore.

Recursos de políticas para MediaStore

Admite recursos de políticas	Sí

Los administradores pueden utilizar las políticas JSON de AWS para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Resource` de la política JSON especifica el objeto u objetos a los que se aplica la acción. Las instrucciones deben contener un elemento `Resource` o `NotResource`. Como práctica recomendada, especifique un recurso utilizando el Nombre de recurso de Amazon (ARN). Puede hacerlo para acciones que admitan un tipo de recurso específico, conocido como permisos de nivel de recurso.

Para las acciones que no admiten permisos de nivel de recurso, como las operaciones de descripción, utilice un carácter comodín (*) para indicar que la instrucción se aplica a todos los recursos.

```
"Resource": "*"
```

Para ver una lista de los tipos de recursos de MediaStore y sus ARN, consulte Recursos definidos por AWS Elemental MediaStore en la Referencia de autorizaciones de servicio. Para obtener

información sobre las acciones con las que puede especificar el ARN de cada recurso, consulte Acciones definidas por AWS Elemental MediaStore.

El recurso del contenedor de MediaStore tiene el siguiente ARN:

```
arn:${Partition}:mediastore:${Region}:${Account}:container/${containerName}
```

Para obtener más información acerca del formato de los ARN, consulte Nombres de recursos de Amazon (ARN) y espacios de nombres de servicios de AWS.

Por ejemplo, para especificar el contenedor `AwardsShow` en su instrucción, utilice el siguiente ARN:

```
"Resource": "arn:aws:mediastore:us-east-1:111122223333:container/AwardsShow"
```

Claves de condición de políticas para MediaStore

Admite claves de condición de políticas específicas del servicio	Sí

Los administradores pueden utilizar las políticas JSON de AWS para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Condition` (o bloque de `Condition`) permite especificar condiciones en las que entra en vigor una instrucción. El elemento `Condition` es opcional. Puede crear expresiones condicionales que utilicen operadores de condición, tales como igual o menor que, para que la condición de la política coincida con los valores de la solicitud.

Si especifica varios elementos de `Condition` en una instrucción o varias claves en un único elemento de `Condition`, AWS las evalúa mediante una operación lógica `AND`. Si especifica varios valores para una única clave de condición, AWS evalúa la condición con una operación `OR` lógica. Se deben cumplir todas las condiciones antes de que se concedan los permisos de la instrucción.

También puede utilizar variables de marcador de posición al especificar condiciones. Por ejemplo, puede conceder un permiso de usuario de IAM para acceder a un recurso solo si está etiquetado con su nombre de usuario de IAM. Para más información, consulte Elementos de la política de IAM: variables y etiquetas en la Guía del usuario de IAM.

AWS admite claves de condición globales y claves de condición específicas del servicio. Para ver todas las claves de condición globales de AWS, consulte Claves de contexto de condición globales de AWS en la Guía del Usuario de IAM.

Para ver una lista de las claves de condición de MediaStore, consulte Claves de condición para AWS Elemental MediaStore en la Referencia de autorizaciones de servicio. Para obtener más información sobre las acciones y los recursos con los que puede utilizar una clave de condición, consulte Acciones definidas por AWS Elemental MediaStore.

Para ver ejemplos de políticas basadas en identidad de MediaStore, consulte Ejemplos de políticas basadas en identidades de AWS Elemental MediaStore.

ACL en MediaStore

Admite las ACL	No

Las listas de control de acceso (ACL) controlan qué entidades principales (miembros de cuentas, usuarios o roles) tienen permisos para acceder a un recurso. Las ACL son similares a las políticas basadas en recursos, aunque no utilizan el formato de documento de políticas JSON.

ABAC con CloudWatch

Admite ABAC (etiquetas en las políticas)	Parcial

El control de acceso basado en atributos (ABAC) es una estrategia de autorización que define permisos en función de atributos. En AWS, estos atributos se denominan etiquetas. Puede adjuntar etiquetas a entidades de IAM (usuarios o roles) y a muchos recursos de AWS. El etiquetado de entidades y recursos es el primer paso de ABAC. A continuación, designa las políticas de ABAC para permitir operaciones cuando la etiqueta de la entidad principal coincida con la etiqueta del recurso al que se intenta acceder.

ABAC es útil en entornos que crecen con rapidez y ayuda en situaciones en las que la administración de las políticas resulta engorrosa.

Para controlar el acceso en función de etiquetas, debe proporcionar información de las etiquetas en el elemento de condición de una política utilizando las claves de condición `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`* o `aws:TagKeys`.

Si un servicio admite las tres claves de condición para cada tipo de recurso, el valor es Sí para el servicio. Si un servicio admite las tres claves de condición solo para algunos tipos de recursos, el valor es Parcial.

Para obtener más información sobre ABAC, consulte ¿Qué es ABAC? en la Guía del Usuario de IAM. Para ver un tutorial con los pasos para configurar ABAC, consulte Uso del control de acceso basado en atributos (ABAC) en la Guía del Usuario de IAM.

Uso de credenciales temporales con MediaStore

Admite el uso de credenciales temporales	Sí

Algunos Servicios de AWS no funcionan cuando inicia sesión con credenciales temporales. Para obtener información adicional, incluida la información sobre qué Servicios de AWS funcionan con credenciales temporales, consulte Servicios de AWS que funcionan con IAM en la Guía del usuario de IAM.

Utilice credenciales temporales si inicia sesión en la AWS Management Console con cualquier método excepto un nombre de usuario y una contraseña. Por ejemplo, cuando accede a AWS utilizando el enlace de inicio de sesión único (SSO) de la empresa, ese proceso crea automáticamente credenciales temporales. También crea automáticamente credenciales temporales cuando inicia sesión en la consola como usuario y luego cambia de rol. Para obtener más información sobre el cambio de roles, consulte Cambio a un rol (consola) en la Guía del usuario de IAM.

Puede crear credenciales temporales de forma manual mediante la AWS CLI o la API de AWS. A continuación, puede usar esas credenciales temporales para acceder a AWS. AWS recomienda generar credenciales temporales de forma dinámica en lugar de usar claves de acceso a largo plazo. Para más información, consulte Credenciales de seguridad temporales en IAM.

Permisos de entidades principales entre servicios de MediaStore

Admite Forward access sessions (FAS)	Sí

Cuando utiliza un usuario o un rol de IAM para llevar a cabo acciones en AWS, se lo considera una entidad principal. Cuando utiliza algunos servicios, es posible que realice una acción que

desencadene otra acción en un servicio diferente. FAS utiliza los permisos de la entidad principal para llamar a un Servicio de AWS, combinados con el Servicio de AWS solicitante para realizar solicitudes a servicios posteriores. Las solicitudes de FAS solo se realizan cuando un servicio recibe una solicitud que requiere interacciones con otros Servicios de AWS o recursos para completarse. En este caso, debe tener permisos para realizar ambas acciones. Para obtener información sobre las políticas a la hora de realizar solicitudes de FAS, consulte Reenviar sesiones de acceso.

Roles de servicio para MediaStore

Compatible con roles de servicio	Sí

Un rol del servicio es un rol de IAM que asume un servicio para realizar acciones en su nombre. Un administrador de IAM puede crear, modificar y eliminar un rol de servicio desde IAM. Para obtener más información, consulte Creación de un rol para delegar permisos a un Servicio de AWS en la Guía del usuario de IAM.

> **Warning**
>
> Cambiar los permisos de un rol de servicio podría interrumpir la funcionalidad de MediaStore. Edite los roles de servicio solo cuando MediaStore proporcione orientación para hacerlo.

Roles vinculados a servicios para MediaStore

Admite roles vinculados a servicios	No

Un rol vinculado a servicios es un tipo de rol de servicio que está vinculado a un Servicio de AWS. El servicio puede asumir el rol para realizar una acción en su nombre. Los roles vinculados a servicios aparecen en su Cuenta de AWS y son propiedad del servicio. Un administrador de IAM puede ver, pero no editar, los permisos de los roles vinculados a servicios.

Para más información sobre cómo crear o administrar roles vinculados a servicios, consulte Servicios de AWS que funcionan con IAM. Busque un servicio en la tabla que incluya `Yes` en la columna Rol vinculado a un servicio. Seleccione el vínculo Sí para ver la documentación acerca del rol vinculado a servicios para ese servicio.

Ejemplos de políticas basadas en identidades de AWS Elemental MediaStore

De forma predeterminada, los usuarios y roles no tienen permiso para crear, ver ni modificar recursos de MediaStore. Tampoco pueden realizar tareas mediante la AWS Management Console, la AWS Command Line Interface (AWS CLI) o la API de AWS. Para conceder permiso a los usuarios para realizar acciones en los recursos que necesiten, un administrador de IAM puede crear políticas de IAM. A continuación, el administrador puede añadir las políticas de IAM a roles, y los usuarios pueden asumirlos.

Para obtener información sobre cómo crear una política basada en identidad de IAM mediante el uso de estos documentos de políticas JSON de ejemplo, consulte Creación de políticas de IAM en la Guía del usuario de IAM.

A fin de obtener más información sobre las acciones y los tipos de recursos definidos por MediaStore, incluido el formato de los ARN para cada tipo de recurso, consulte Acciones, recursos y claves de condición para AWS Elemental MediaStore en la Referencia de autorizaciones de servicio.

Temas

- Prácticas recomendadas sobre las políticas
- Uso de la consola MediaStore
- Cómo permitir a los usuarios consultar sus propios permisos

Prácticas recomendadas sobre las políticas

Las políticas basadas en identidades determinan si alguien puede crear, acceder o eliminar los recursos de MediaStore de la cuenta. Estas acciones pueden generar costes adicionales para su Cuenta de AWS. Siga estas directrices y recomendaciones al crear o editar políticas basadas en identidades:

- Comience con las políticas administradas de AWS y continúe con los permisos de privilegio mínimo: a fin de comenzar a conceder permisos a los usuarios y las cargas de trabajo, utilice las políticas administradas de AWS, que conceden permisos para muchos casos de uso comunes. Están disponibles en su Cuenta de AWS. Se recomienda definir políticas administradas por el cliente de AWS específicas para los casos de uso a fin de reducir aún más los permisos. Con el fin de obtener más información, consulte las políticas administradas por AWS o las políticas administradas por AWS para funciones de trabajo en la Guía del usuario de IAM.

- Aplique permisos de privilegio mínimo: cuando establezca permisos con políticas de IAM, conceda solo los permisos necesarios para realizar una tarea. Para ello, debe definir las acciones que se pueden llevar a cabo en determinados recursos en condiciones específicas, también conocidos como permisos de privilegios mínimos. Con el fin de obtener más información sobre el uso de IAM para aplicar permisos, consulte Políticas y permisos en IAM en la Guía de usuario de IAM.
- Utilice condiciones en las políticas de IAM para restringir aún más el acceso: puede agregar una condición a sus políticas para limitar el acceso a las acciones y los recursos. Por ejemplo, puede escribir una condición de política para especificar que todas las solicitudes deben enviarse utilizando SSL. También puede usar condiciones para conceder acceso a acciones de servicios si se emplean a través de un Servicio de AWS determinado, como por ejemplo AWS CloudFormation. Para obtener más información, consulte Elementos de la política de JSON de IAM: Condición en la Guía del usuario de IAM.
- Utilice el analizador de acceso de IAM para validar las políticas de IAM con el fin de garantizar la seguridad y funcionalidad de los permisos: el analizador de acceso de IAM valida políticas nuevas y existentes para que respeten el lenguaje (JSON) de las políticas de IAM y las prácticas recomendadas de IAM. El analizador de acceso de IAM proporciona más de 100 verificaciones de políticas y recomendaciones procesables para ayudar a crear políticas seguras y funcionales. Para más información, consulte la Política de validación del analizador de acceso de IAM en la Guía de usuario de IAM.
- Solicite la autenticación multifactor (MFA): si se encuentra en una situación en la que necesita usuarios raíz o de IAM en su Cuenta de AWS, active la MFA para mayor seguridad. Para solicitar la MFA cuando se invocan las operaciones de la API, agregue las condiciones de la MFA a sus políticas. Para obtener más información, consulte Configuración de acceso a una API protegida por MFA en la Guía del usuario de IAM.

Para obtener más información sobre las prácticas recomendadas de IAM, consulte las Prácticas recomendadas de seguridad en IAM en la Guía del usuario de IAM.

Uso de la consola MediaStore

Para acceder a la consola de AWS Elemental MediaStore, debe tener un conjunto mínimo de permisos. Estos permisos deben permitirle registrar y consultar los detalles acerca de los recursos de MediaStore en su Cuenta de AWS. Si crea una política basada en identidades que sea más restrictiva que el mínimo de permisos necesarios, la consola no funcionará del modo esperado para las entidades (usuarios o roles) que tengan esa política.

No es necesario que conceda permisos mínimos para la consola a los usuarios que solo realizan llamadas a la AWS CLI o a la API de AWS. En su lugar, permite acceso únicamente a las acciones que coincidan con la operación de API que intentan realizar.

Para asegurarse de que los usuarios y los roles puedan seguir utilizando la consola de MediaStore, adjunte también la política gestionada *ConsoleAccess* o *ReadOnly* AWS de MediaStore a las entidades. Para obtener más información, consulte Adición de permisos a un usuario en la Guía del usuario de IAM.

Cómo permitir a los usuarios consultar sus propios permisos

En este ejemplo, se muestra cómo podría crear una política que permita a los usuarios de IAM ver las políticas administradas e insertadas que se asocian a la identidad de sus usuarios. Esta política incluye permisos para realizar esta acción en la consola o mediante programación con la AWS CLI o la API de AWS.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
```

```
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Solución de problemas de identidad y acceso a AWS Elemental MediaStore

Utilice la siguiente información para diagnosticar y solucionar los problemas comunes que puedan surgir cuando trabaje con MediaStore e IAM.

Temas

- No estoy autorizado a realizar una acción en MediaStore
- No tengo autorización para realizar la operación iam:PassRole
- Quiero permitir a personas externas a mi Cuenta de AWS el acceso a mis recursos de MediaStore

No estoy autorizado a realizar una acción en MediaStore

Si recibe un error que indica que no tiene autorización para realizar una acción, las políticas se deben actualizar para permitirle realizar la acción.

En el siguiente ejemplo, el error se produce cuando el usuario de IAM `mateojackson` intenta utilizar la consola para consultar los detalles acerca de un recurso ficticio *`my-example-widget`*, pero no tiene los permisos ficticios `mediastore:`*`GetWidget`*.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 mediastore:GetWidget on resource: my-example-widget
```

En este caso, la política del usuario `mateojackson` debe actualizarse para permitir el acceso al recurso *`my-example-widget`* mediante la acción `mediastore:`*`GetWidget`*.

Si necesita ayuda, póngase en contacto con su administrador de AWS. El administrador es la persona que le proporcionó las credenciales de inicio de sesión.

No tengo autorización para realizar la operación iam:PassRole

Si recibe un error que indica que no tiene autorización para realizar la acción `iam:PassRole`, se deben actualizar las políticas a fin de permitirle pasar un rol a MediaStore.

Algunos Servicios de AWS le permiten transferir un rol existente a dicho servicio en lugar de crear un nuevo rol de servicio o uno vinculado a servicios. Para ello, debe tener permisos para transferir el rol al servicio.

En el siguiente ejemplo, el error se produce cuando un usuario de IAM denominado `marymajor` intenta utilizar la consola para realizar una acción en MediaStore. Sin embargo, la acción requiere que el servicio cuente con permisos que concede un rol de servicio. Mary no tiene permisos para transferir el rol al servicio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

En este caso, las políticas de Mary se deben actualizar para permitirle realizar la acción `iam:PassRole`.

Si necesita ayuda, póngase en contacto con su administrador de AWS. El administrador es la persona que le proporcionó las credenciales de inicio de sesión.

Quiero permitir a personas externas a mi Cuenta de AWS el acceso a mis recursos de MediaStore

Puede crear un rol que los usuarios de otras cuentas o las personas externas a la organización puedan utilizar para acceder a sus recursos. Puede especificar una persona de confianza para que asuma el rol. En el caso de los servicios que admitan las políticas basadas en recursos o las listas de control de acceso (ACL), puede utilizar dichas políticas para conceder a las personas acceso a sus recursos.

Para más información, consulte lo siguiente:

- Para obtener información acerca de si MediaStore admite estas características, consulte Cómo funciona AWS Elemental MediaStore con IAM.
- Para obtener información acerca de cómo proporcionar acceso a los recursos de las Cuentas de AWS de su propiedad, consulte Cómo proporcionar acceso a un usuario de IAM a otra Cuenta de AWS de la que es propietario en la Guía del usuario de IAM.

- Para obtener información sobre cómo proporcionar acceso a los recursos a Cuentas de AWS de terceros, consulte Proporcionar acceso a Cuentas de AWS que son propiedad de terceros en la Guía del usuario de IAM.
- Para obtener información sobre cómo proporcionar acceso mediante federación de identidades, consulte Proporcionar acceso a usuarios autenticados externamente (federación de identidades) en la Guía del usuario de IAM.
- Para obtener información sobre la diferencia entre los roles y las políticas basadas en recursos para el acceso entre cuentas, consulte Cómo los roles de IAM difieren de las políticas basadas en recursos en la Guía del Usuario de IAM.

Registro y monitorización en AWS Elemental MediaStore

En esta sección, se proporciona información general acerca de las opciones para registrar y monitorizar en AWS Elemental MediaStore por motivos de seguridad. Para obtener más información acerca del registro y la monitorización en MediaStore, consulte Monitorización y etiquetado en AWS Elemental MediaStore.

La supervisión es un aspecto importante del mantenimiento de la fiabilidad, la disponibilidad y el rendimiento de AWS Elemental MediaStore y sus soluciones de AWS. Debe recopilar datos de monitorización de todas las partes de su solución de AWS para que pueda más fácilmente depurar un error multipunto si se produce. AWS proporciona varias herramientas para monitorizar sus recursos de MediaStore y responder a posibles incidentes:

Alarmas de Amazon CloudWatch

Las alarmas de Amazon CloudWatch le permiten ver una sola métrica durante el período de tiempo que especifique. Si la métrica supera un umbral determinado, se envía una notificación a un tema de Amazon SNS o a una política de AWS Auto Scaling. Las alarmas de CloudWatch no invocan acciones simplemente porque estén en un estado particular. En su lugar, el estado debe haber cambiado y debe mantenerse durante el número de periodos especificado. Para obtener más información, consulte Monitorización con CloudWatch.

AWS CloudTrailRegistros de

CloudTrail proporciona un registro de las acciones que realiza un usuario, un rol o un servicio de AWS en AWS Elemental MediaStore. Mediante la información recopilada por CloudTrail, puede determinar la solicitud que se realizó a MediaStore, la dirección IP desde la que se realizó, quién

la realizó y cuándo, etc. Para obtener más información, consulte Registrar llamadas a la API con CloudTrail.

AWS Trusted Advisor

Trusted Advisor aprovecha las prácticas recomendadas aprendidas al atender a cientos de miles de clientes de AWS. Trusted Advisor inspecciona su entorno de AWS y realiza recomendaciones cuando surge la oportunidad de ahorrar dinero, mejorar el rendimiento y la disponibilidad del sistema o ayudar a cerrar deficiencias de seguridad. Todos los clientes de AWS disponen de acceso a cinco comprobaciones de Trusted Advisor. Los clientes con un plan de soporte Business o Enterprise pueden ver todas las comprobaciones de Trusted Advisor.

Para obtener más información, consulte AWS Trusted Advisor.

Validación de conformidad para AWS Elemental MediaStore

Para saber si un Servicio de AWS está incluido en el ámbito de programas de conformidad específicos, consulte Servicios de AWS en el ámbito del programa de conformidad y elija el programa de conformidad que le interese. Para obtener información general, consulte Programas de conformidad de AWS.

Puede descargar los informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulte Descarga de informes en AWS Artifact.

Su responsabilidad de conformidad al utilizar servicios de AWS se determina en función de la sensibilidad de los datos, los objetivos de cumplimiento de su empresa y la legislación y los reglamentos correspondientes. AWS proporciona los siguientes recursos para ayudar con la conformidad:

- Guías de inicio rápido de seguridad y conformidad: estas guías de implementación tratan consideraciones sobre arquitectura y ofrecen pasos para implementar los entornos de referencia centrados en la seguridad y la conformidad en AWS.
- Arquitectura para la seguridad y el cumplimiento de la HIPAA en Amazon Web Services: en este documento técnico, se describe cómo las empresas pueden utilizar AWS para crear aplicaciones aptas para HIPAA.

> **Note**
>
> No todos los Servicios de AWS son aptos para HIPAA. Para obtener más información, consulte la Referencia de servicios aptos para HIPAA.

- Recursos de conformidad de AWS: este conjunto de manuales y guías podría aplicarse a su sector y ubicación.
- Guías de cumplimiento para clientes de AWS: comprenda el modelo de responsabilidad compartida desde el punto de vista del cumplimiento. Las guías resumen las mejores prácticas para garantizar la seguridad de los Servicios de AWS y orientan los controles de seguridad en varios marcos (incluidos el Instituto Nacional de Estándares y Tecnología (NIST, por sus siglas en inglés), el Consejo de Estándares de Seguridad de la Industria de Tarjetas de Pago (PCI, por sus siglas en inglés) y la Organización Internacional de Normalización (ISO, por sus siglas en inglés)).
- Evaluación de recursos con reglas en la Guía para desarrolladores de AWS Config: el servicio AWS Config evalúa en qué medida las configuraciones de sus recursos cumplen las prácticas internas, las directrices del sector y las normativas.
- AWS Security Hub: este servicio de AWS proporciona una visión completa de su estado de seguridad en AWS. Security Hub utiliza controles de seguridad para evaluar sus recursos de AWS y comprobar su cumplimiento con los estándares y las prácticas recomendadas del sector de la seguridad. Para obtener una lista de los servicios y controles compatibles, consulte la Referencia de controles de Security Hub.
- AWS Audit Manager: este servicio de Servicio de AWS le ayuda a auditar continuamente el uso de AWS con el fin de simplificar la forma en que administra el riesgo y la conformidad con las normativas y los estándares del sector.

Resiliencia en AWS Elemental MediaStore

La infraestructura global de AWS se divide en Regiones de AWS y zonas de disponibilidad. Las Regiones de AWS proporcionan varias zonas de disponibilidad físicamente independientes y aisladas que se encuentran conectadas mediante redes con un alto nivel de rendimiento y redundancia, además de baja latencia. Con las zonas de disponibilidad, puede diseñar y utilizar aplicaciones y bases de datos que conmutan automáticamente entre zonas sin interrupción. Las zonas de disponibilidad tienen una mayor disponibilidad, tolerancia a errores y escalabilidad que las infraestructuras tradicionales de uno o varios centros de datos.

Para obtener más información sobre las Regiones de AWS y las zonas de disponibilidad, consulte la Infraestructura global de AWS.

Además de la infraestructura global de AWS, MediaStore ofrece varias características que le ayudan con sus necesidades de resiliencia y copia de seguridad de los datos.

Seguridad de la infraestructura en AWS Elemental MediaStore

Como se trata de un servicio administrado, AWS Elemental MediaStore está protegido por la seguridad de red global de AWS. Para obtener información sobre los servicios de seguridad de AWS y cómo AWS protege la infraestructura, consulte Seguridad en la nube de AWS. Para diseñar su entorno de AWS conforme a las prácticas recomendadas de seguridad de la infraestructura, consulte Protección de la infraestructura en Pilar de seguridad del Marco de AWS Well-Architected.

Puede utilizar llamadas a la API publicadas en AWS para acceder a la MediaStore 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 principal de IAM. También puede utilizar AWS Security Token Service (AWS STS) para generar credenciales de seguridad temporales para firmar solicitudes.

Prevención de la sustitución confusa entre servicios

El problema de la sustitución confusa es un problema de seguridad en el que una entidad que no tiene permiso para realizar una acción puede obligar a una entidad con más privilegios a realizar la acción. En AWS, la suplantación entre servicios puede dar lugar al problema de la sustitución confusa. La suplantación entre servicios puede producirse cuando un servicio (el servicio que lleva a cabo las llamadas) llama a otro servicio (el servicio al que se llama). El servicio que lleva a cabo las llamadas se puede manipular para utilizar sus permisos a fin de actuar en función de los recursos de otro cliente de una manera en la que no debe tener permiso para acceder. Para evitarlo, AWS proporciona herramientas que lo ayudan a proteger sus datos para todos los servicios con entidades principales de servicio a las que se les ha dado acceso a los recursos de su cuenta.

Se recomienda utilizar las claves de contexto de condición global `aws:SourceArn` o `aws:SourceAccount` en las políticas de recursos para limitar los permisos que AWS Elemental MediaStore concede a otro servicio para el recurso. Utilice `aws:SourceArn` si desea que solo se asocie un recurso al acceso entre servicios. Utilice `aws:SourceAccount` si quiere permitir que cualquier recurso de esa cuenta se asocie al uso entre servicios.

La forma más eficaz de protegerse contra el problema de la sustitución confusa es utilizar la clave de contexto de condición global de `aws:SourceArn` con el ARN completo del recurso. Si no conoce el ARN completo del recurso o si está especificando varios recursos, utilice la clave de condición de contexto global `aws:SourceArn` con caracteres comodines (*) para las partes desconocidas del ARN. Por ejemplo, `arn:aws:`*`servicename`*`:*:`*`123456789012`*`:*`.

Si el valor de `aws:SourceArn` no contiene el ID de cuenta, como un ARN de bucket de Amazon S3, debe utilizar ambas claves de contexto de condición global para limitar los permisos.

El valor de `aws:SourceArn` debe ser la configuración para la que MediaStore publica los registros de CloudWatch en su región y cuenta.

El siguiente ejemplo muestra cómo se pueden utilizar las claves contextuales de condición global `aws:SourceArn` y `aws:SourceAccount` en MediaStore para evitar el problema del adjunto confundido.

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
    "Principal": {
      "Service": "servicename.amazonaws.com"
    },
    "Action": "servicename:ActionName",
    "Resource": [
      "arn:aws:servicename:::ResourceName/*"
    ],
    "Condition": {
      "ArnLike": {
        "aws:SourceArn": "arn:aws:servicename:*:123456789012:*"
      },
      "StringEquals": {
        "aws:SourceAccount": "123456789012"
      }
```

```
      }
    }
}
```

Monitorización y etiquetado en AWS Elemental MediaStore

El monitoreo es una parte importante a la hora de mantener la fiabilidad, la disponibilidad y el rendimiento de AWS Elemental MediaStore yAWS las demás soluciones de. AWSofrece las siguientes herramientas de monitoreo para ver MediaStore, informar cuando algo no funciona y realizar acciones automáticas cuando corresponda:

- AWS CloudTrail captura llamadas a la API y eventos relacionados efectuados por su cuenta de AWS o en su nombre, y entrega los archivos de registro al bucket de Amazon S3 que se haya especificado. También pueden identificar qué usuarios y cuentas llamaron a AWS, la dirección IP de origen de las llamadas y el momento en que se hicieron. Para obtener más información, consulte la Guía del usuario de AWS CloudTrail.
- Amazon CloudWatch monitoreaAWS los recursos de y las aplicaciones que ejecutaAWS en tiempo real. Puede recopilar métricas y realizar un seguimiento de las métricas, crear paneles personalizados y definir alarmas que le advierten o que toman medidas cuando una métrica determinada alcanza el umbral que se especifique. Por ejemplo, puede hacer que realice CloudWatch un seguimiento del uso de la CPU u otras métricas de las instancias de Amazon EC2 y lanzar nuevas instancias automáticamente cuando sea necesario. Para obtener más información, consulte la Guía del CloudWatch usuario de Amazon.
- Amazon CloudWatch Events proporciona un flujo de eventos de sistema que describen cambios en losAWS recursos de. NormalmenteAWS los servicios de enviar notificaciones de CloudWatch eventos a Events en cuestión de segundos, pero a veces pueden tardar un minuto o más. CloudWatch Events habilita la informática basada en eventos automatizada, para que pueda escribir reglas que vigilan determinados eventos y desencadenan acciones automatizadas en otrosAWS servicios de cuando estos eventos se producen. Para obtener más información, consulte la Guía del usuario de Amazon CloudWatch Events.
- Amazon CloudWatch Logs le permite monitorear, almacenar y tener acceso a los archivos de registro desde instancias de Amazon EC2 u otras fuentes. CloudTrail CloudWatch Los registros pueden monitorear información en los archivos de registro y enviarle una notificación cuando se llega a determinados umbrales. También se pueden archivar los datos de los registros en un almacenamiento de larga duración. Para obtener más información, consulte la Guía del usuario de Amazon CloudWatch Logs.

También puede asignar metadatos a los MediaStore contenedores de en forma de etiquetas. Cada etiqueta consta de una clave y un valor definidos. Las etiquetas pueden facilitar la administración, la

búsqueda y el filtrado de recursos. Puede utilizar etiquetas para organizar sus recursos de AWS en la consola de administración de AWS, crear informes de uso y facturación en todos sus recursos de AWS y filtrar los recursos durante las actividades de automatización de la infraestructura.

Temas

- Registro de llamadas a la MediaStore API de AWS Elemental conAWS CloudTrail
- Supervisión de AWS Elemental MediaStore con Amazon CloudWatch
- Etiquetado de los recursos de AWS Elemental MediaStore

Registro de llamadas a la MediaStore API de AWS Elemental conAWS CloudTrail

AWS Elemental MediaStore se integra conAWS CloudTrail, un servicio que proporciona un registro de las acciones llevadas a cabo por un usuario, un rol o unAWS servicio de en MediaStore. CloudTrail captura un subconjunto de llamadas a la API de MediaStore como eventos, incluidas las llamadas procedentes de la MediaStore consola de y las llamadas de código a la MediaStore API de. Si crea un registro de seguimiento, puede habilitar la entrega continua de CloudTrail eventos a un bucket de Amazon S3, incluidos los eventos para MediaStore. Si no configura un registro de seguimiento, puede ver los eventos más recientes de la CloudTrail consola de en el Event history (Historial de eventos). Mediante la información que recopila CloudTrail, se puede determinar la solicitud que se envió a MediaStore, la dirección IP desde la que se realizó la solicitud, quién la realizó, cuándo la realizó y detalles.

Para obtener más información CloudTrail, incluso cómo configurarlo y habilitarlo, consulte la Guía delAWS CloudTrail usuario de.

Temas

- MediaStoreInformación de AWS Elemental en CloudTrail
- Ejemplo: entradas de archivos de MediaStore registro de AWS Elemental

MediaStoreInformación de AWS Elemental en CloudTrail

CloudTrail se habilita en suAWS cuenta de al crearla. Cuando se produce una actividad de eventos compatible en AWS Elemental MediaStore, la actividad se registra en un CloudTrail evento junto con otros eventos deAWS servicios de en Event history (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 de.

Para mantener un registro continuo de eventos en la cuenta de AWS, incluidos los eventos de MediaStore, cree un registro de seguimiento. Un registro de seguimiento CloudTrail permite entregar archivos de registro a un bucket de Amazon S3. De manera predeterminada, cuando se crea un registro de seguimiento en la consola, el registro de seguimiento se aplica a todas las regiones de AWS. El registro de seguimiento registra los eventos de todas las regiones de la partición de AWS y envía los archivos de registro al bucket de Amazon S3 especificado. También es posible configurar otrosAWS servicios de para analizar en profundidad y actuar en función de los datos de eventos recopilados en los CloudTrail registros de. Para obtener más información, consulte los siguientes temas:

- Introducción a la creación de registros de seguimiento
- CloudTrail Integraciones y servicios compatibles
- Configuración de notificaciones de Amazon SNS para CloudTrail
- Recepción de archivos de CloudTrail registro de varias regiones y Recepción de archivos de CloudTrail registro de varias cuentas

AWS Elemental MediaStore admite el registro de las siguientes operaciones como eventos en archivos de CloudTrail registro:

- `CreateContainer`
- `DeleteContainer`
- `DeleteContainerPolicy`
- `DeleteCorsPolicy`
- `DescribeContainer`
- `GetContainerPolicy`
- `GetCorsPolicy`
- `ListContainers`
- `PutContainerPolicy`
- `PutCorsPolicy`

Cada entrada de registro o evento contiene información sobre quién generó la solicitud. La información de identidad del usuario le ayuda a determinar lo siguiente:

- Si la solicitud se realizó con las credenciales del usuario raíz o del usuario de
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado
- Si la solicitud la realizó otro servicio de AWS.

Para obtener más información, consulte el elemento userIdentity de CloudTrail.

Ejemplo: entradas de archivos de MediaStore registro de AWS Elemental

Un registro de seguimiento es una configuración que permite entregar eventos como archivos de registro al bucket de Amazon S3 que especifique. CloudTrail los archivos de registro pueden contener una o varias entradas de registro. Un evento representa una solicitud específica realizada desde un origen y contiene 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 rastro de la pila ordenado de las llamadas a las API públicas, por lo que no aparecen en ningún orden específico.

En el ejemplo siguiente, se muestra una entrada de CloudTrail registro de que ilustra la`CreateContainer` operación:


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "ABCDEFGHIJKL123456789",
        "arn": "arn:aws:iam::111122223333:user/testUser",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "testUser",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-07-09T12:55:42Z"
            }
        },
        "invokedBy": "signin.amazonaws.com"
    },
    "eventTime": "2018-07-09T12:56:54Z",
```

```
    "eventSource": "mediastore.amazonaws.com",
    "eventName": "CreateContainer",
    "awsRegion": "ap-northeast-1",
    "sourceIPAddress": "54.239.119.16",
    "userAgent": "signin.amazonaws.com",
    "requestParameters": {
        "containerName": "TestContainer"
    },
    "responseElements": {
        "container": {
            "status": "CREATING",
            "creationTime": "Jul 9, 2018 12:56:54 PM",
            "name": " TestContainer ",
            "aRN": "arn:aws:mediastore:ap-northeast-1:111122223333:container/
TestContainer"
        }
    },
    "requestID":
 "MNCTGH4HRQJ27GRMBVDPIVHEP4LO2BN6MUVHBCPSHOAWNSOKSXCO24B2UEOBBND5DONRXTMFK3TOJ4G7AHWMESI",
    "eventID": "7085b140-fb2c-409b-a329-f567912d704c",
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```


Supervisión de AWS Elemental MediaStore con Amazon CloudWatch

Puede monitorear el MediaStore uso de AWS Elemental CloudWatch, que recopila datos sin procesar y los procesa en métricas legibles. CloudWatch mantiene las estadísticas durante 15 meses, lo que le permite tener acceso a información histórica y disponer de una mejor perspectiva sobre el rendimiento de su aplicación web o servicio. También puede establecer alarmas que vigilen determinados umbrales y enviar notificaciones o realizar acciones cuando se cumplan dichos umbrales. Para obtener más información, consulte la Guía del CloudWatch usuario de Amazon.

AWSofrece las siguientes herramientas de monitoreo para ver MediaStore, informar cuando algo no funciona y realizar acciones automáticas cuando corresponda:

- Amazon CloudWatch Logs le permite monitorear, almacenar y tener acceso a los archivos de registro desdeAWS servicios como AWS Elemental MediaStore. Puede utilizar CloudWatch Registros para monitorear aplicaciones y sistemas mediante datos de registro. Por ejemplo,

CloudWatch Logs puede realizar el seguimiento de los errores presentes en los registros de las aplicaciones y enviarle una notificación cuando la tasa de errores supere el umbral que se especifique. CloudWatch Logs utiliza los datos de registro de para el monitoreo, por lo que no es necesario cambiar el código. Por ejemplo, puede supervisar los registros de las aplicaciones para detectar términos literales específicos (comoValidationException "») o contar el número de`PutObject` solicitudes que se realizaron durante un período de tiempo determinado. Cuando se encuentra el término que está buscando, CloudWatch Logs notifica los datos a una CloudWatch métrica que especifique. Los datos de registro están cifrados mientras están en tránsito y cuando están en reposo.

- Amazon CloudWatch Events ofrece eventos del sistema que describen los cambios enAWS los recursos, como MediaStore los objetos. NormalmenteAWS los servicios de enviar notificaciones de CloudWatch eventos a Events en cuestión de segundos, pero a veces pueden tardar un minuto o más. Puede configurar reglas para que coincidan los eventos (como una`DeleteObject` solicitud) y dirigirlos a una o más secuencias o funciones de destino. CloudWatch Events toma conocimiento de los cambios operativos a medida que se producen. Además, CloudWatch Events responde a estos cambios operativos y toma medidas correctoras según sea necesario, enviando mensajes para responder al entorno, activando funciones, realizando cambios y captando información de estado.

CloudWatch Registros

El registro de acceso proporciona registros detallados para las solicitudes que se realizan a objetos en un contenedor. Los registros de acceso son útiles para muchas aplicaciones, como, por ejemplo, auditorías de acceso y seguridad. También pueden ayudarle a conocer mejor su base de clientes y entender su MediaStore factura de. CloudWatch Los registros se clasifican de la siguiente manera:

- Un flujo de registro es una secuencia de eventos de registro que comparten la misma fuente.
- Un grupo de registros es un grupo de flujos de registro que comparten la misma configuración de retención, monitoreo y control de acceso. Al habilitar el registro de acceso en un contenedor, MediaStore crea un grupo de registro con un nombre como`/aws/mediastore/MyContainerName`. Puede definir grupos de registros y especificar los flujos que deben incluirse en cada uno. No hay cuotas en el número de flujos de registro que pueden pertenecer a un grupo de registros.

De forma predeterminada, los registros se conservan de forma indefinida y no caducan nunca. Puede ajustar la política de retención para cada grupo de registros, manteniendo la retención indefinida o seleccionar un periodo de retención entre un día y 10 años.

Configuración de permisos para Amazon CloudWatch

UtiliceAWS Identity and Access Management (IAM) para crear un rol que dé MediaStore acceso a AWS Elemental a Amazon CloudWatch. Debe realizar estos pasos para que se publiquen CloudWatch los registros de su cuenta. CloudWatch publica automáticamente las métricas de tu cuenta.

Para permitir el MediaStore acceso a CloudWatch

1. Abra la consola de IAM en https://console.aws.amazon.com/iam/.
2. En el panel de navegación de la consola de IAM, elija Políticas, seguido de Crear política.
3. Elija la pestaña JSON y pegue la siguiente política:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogGroups",
                "logs:CreateLogGroup"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:DescribeLogStreams",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:*:*:log-group:/aws/mediastore/*"
        }
    ]
}
```

Esta política permite MediaStore crear grupos de registro y flujos de registro para cualquier contenedor de cualquier región de suAWS cuenta.

4. Elija Review policy (Revisar política).
5. En la página Review policy (Revisar política), para Name (Nombre), escriba **`MediaStoreAccessLogsPolicy`** y después elija Create policy (Crear política).
6. En el panel de navegación de la consola de IAM, seleccione Roles y, a continuación, seleccione Create role (Crear rol).
7. Elija el tipo de rol Another AWS account (Otra cuenta de AWS).
8. En Account ID (ID de cuenta), escriba el ID de su cuenta de AWS.
9. Elija Next: Permissions (Siguiente: permisos).
10. En el cuadro de búsqueda, escriba **`MediaStoreAccessLogsPolicy`**.
11. Seleccione la casilla de verificación situada junto a su nueva política y, a continuación, seleccione Next: Tags (Siguiente: Etiquetas).
12. Elija Next: Review (Siguiente: Revisar) para obtener una vista previa de su nuevo usuario.
13. En Role name (Nombre de rol), escriba **`MediaStoreAccessLogs`** y luego elija Create role (Crear rol).
14. En el mensaje de confirmación, seleccione el nombre del rol que acaba de crear (**`MediaStoreAccessLogs`**).
15. En la página Summary (Resumen) del rol, elija la pestaña Trust relationships (Relaciones de confianza).
16. Elija Edit trust relationship (Editar relación de confianza).
17. En el documento de la política, cambie la entidad principal por el servicio MediaStore. Debería tener un aspecto similar al siguiente:

```
"Principal": {
   "Service": "mediastore.amazonaws.com"
},
```

La política completa debe ser similar a la siguiente:

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

```
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "mediastore.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {}
    }
  ]
}
```

18. Elija Update Trust Policy (Actualizar política de confianza).

Habilitar el registro de acceso para un contenedor

De forma predeterminada, AWS Elemental MediaStore no recopila registros de acceso. Al habilitar el registro de acceso en un contenedor, MediaStore proporciona registros de acceso para los objetos almacenados en dicho contenedor a Amazon CloudWatch. Los registros de acceso proporcionan registros detallados para solicitudes realizadas a cualquier objeto almacenado en el contenedor. Esta información puede incluir el tipo de solicitud, los recursos especificados en la solicitud y la hora y la fecha en que se procesó la solicitud.

> **Important**
>
> No se aplica ningún cargo adicional por habilitar el registro de acceso en un contenedor de MediaStore. Sin embargo, los archivos que el servicio le envía acumularán los cargos habituales de almacenamiento. (Puede eliminar los archivos de registro en cualquier momento). AWS no evalúa los cargos de transferencia de datos por la entrega de archivos de registro, pero sí incluye el cargo de la tasa normal de transferencia de datos para acceder a los archivos de registro.

Para habilitar el registro de acceso (AWS CLI)

- En la AWS CLI, utilice el comando `start-access-logging`:

  ```
  aws mediastore start-access-logging --container-name LiveEvents --region us-west-2
  ```

  Este comando no tiene ningún valor de retorno.

Deshabilitar el registro de acceso para un contenedor

Al deshabilitar el registro de acceso en un contenedor, AWS Elemental MediaStore deja de enviar registros de acceso a Amazon CloudWatch. Estos registros de acceso no se guardan y no son recuperables.

Para deshabilitar el registro de acceso (AWS CLI)

- En la AWS CLI, utilice el comando `stop-access-logging`:

  ```
  aws mediastore stop-access-logging --container-name LiveEvents --region us-west-2
  ```

  Este comando no tiene ningún valor de retorno.

Solución de problemas de registro de acceso en AWS Elemental MediaStore

Si los registros de MediaStore acceso a AWS Elemental no aparecen en Amazon CloudWatch, consulte la siguiente tabla para conocer las posibles causas y las soluciones.

> **Note**
>
> Asegúrese de habilitar AWS CloudTrail Logs para ayudarle con el proceso de resolución de problemas.

Síntoma	El Problema Might Be...	Pruebe esto...
No ves ningún CloudTrail evento, aunque los CloudTrail registros estén habilitados.	El rol de IAM no existe o tiene el nombre, permisos o política de confianza incorrectos.	Cree un rol con el nombre, permisos y política de confianza correctos. Consulte the section called “Configuración de permisos para CloudWatch”.
Envió una solicitud de API `DescribeContainer` , pero la respuesta muestra que el parámetro `AccessLog`	El rol de IAM no existe o tiene el nombre, permisos o	Cree un rol con el nombre, permisos y política de confianza correctos. Consulte

Síntoma	El Problema Might Be...	Pruebe esto...
`gingEnabled` tiene un valor de `False`. Además, no puede ver ningún evento de CloudTrail para que el rol `MediaStoreAccessLogs` realice una llamada a `DescribeLogGroup` , `CreateLogGroup` , `DescribeLogStream` o `CreateLogStream` satisfactoria.	política de confianza incorrectos.	the section called “Configuración de permisos para CloudWatch”.
	El registro de acceso no está habilitado en el contenedor.	Habilite los registros de acceso para el contenedor. Consulte the section called “Habilitar el registro de acceso”.

En la CloudTrail consola, aparece un evento con un error de acceso denegado relacionado con el`MediaStoreAccessLogs` rol. El CloudTrail evento puede incluir líneas como las siguientes:
`"eventSource": "logs.amazonaws.com",`
`"errorCode": "AccessDenied",`
`"errorMessage": "User: arn:aws:sts::111122223333:assumed-role/MediaStoreAccessLogs/MediaStoreAccessLogsSession is not authorized to perform: logs:DescribeLogGroups on resource: arn:aws:logs:us-west-2:111122223333:log-group::log-stream:",`
	El rol de IAM no tiene los permisos correctos para AWS Elemental MediaStore.	Actualice el rol de IAM para que tenga los permisos y la política de confianza correctos . Consulte the section called “Configuración de permisos para CloudWatch”.

Síntoma	El Problema Might Be...	Pruebe esto...
No puede ver ningún registro de un contenedor completo o de contenedores.	Es posible que su cuenta haya superado la CloudWatch cuota de grupos de registro por cuenta y región. Consulte las cuotas de los grupos de registro en la Guía del usuario de Amazon CloudWatch Logs.	En la CloudWatch consola, determine si su cuenta ha alcanzado la CloudWatch cuota de grupos de registro. Si es necesario, solicite un aumento de cuota.
Ve algunos registros iniciados CloudWatch, pero no todos los registros que espera ver.	Es posible que tu cuenta haya superado la CloudWatch cuota de transacciones por segundo por cuenta y región. Consulte las cuotas `PutLogEvents` en la Guía del usuario de Amazon CloudWatch Logs.	Solicita un aumento de cuota para CloudWatch las transacciones por segundo, cuenta y región.

Formato de registro de acceso

Los archivos de registro de acceso constan de una secuencia de entradas de registro con formato JSON y cada entrada de registro representa una solicitud. El orden de los campos en el registro puede variar. A continuación se muestra un ejemplo de un archivo de registro que se compone de dos registros:

```
{
  "Path": "/FootballMatch/West",
```

```
  "Requester": "arn:aws:iam::111122223333:user/maria-garcia",
  "AWSAccountId": "111122223333",
  "RequestID":
 "aaaAAA111bbbBBB222cccCCC333dddDDD444eeeEEE555fffFFF666gggGGG777hhhHHH888iiiIII999jjjJJJ",
  "ContainerName": "LiveEvents",
  "TotalTime": 147,
  "BytesReceived": 1572864,
  "BytesSent": 184,
  "ReceivedTime": "2018-12-13T12:22:06.245Z",
  "Operation": "PutObject",
  "ErrorCode": null,
  "Source": "192.0.2.3",
  "HTTPStatus": 200,
  "TurnAroundTime': 7,
  "ExpiresAt": "2018-12-13T12:22:36Z"
}
 {
  "Path": "/FootballMatch/West",
  "Requester": "arn:aws:iam::111122223333:user/maria-garcia",
  "AWSAccountId": "111122223333",
  "RequestID":
 "dddDDD444eeeEEE555fffFFF666gggGGG777hhhHHH888iiiIII999jjjJJJ000cccCCC333bbbBBB222aaaAAA",
  "ContainerName": "LiveEvents",
  "TotalTime": 3,
  "BytesReceived": 641354,
  "BytesSent": 163,
  "ReceivedTime": "2018-12-13T12:22:51.779Z",
  "Operation": "PutObject",
  "ErrorCode": "ValidationException",
  "Source": "198.51.100.15",
  "HTTPStatus": 400,
  "TurnAroundTime": 1,
  "ExpiresAt": null
}
```


En la siguiente lista se describen los campos de entrada de registro:

AWSAccountId

El ID de la cuenta de AWS utilizada para realizar la solicitud.

BytesReceived

Número de bytes en el cuerpo de la solicitud que recibe el servidor de MediaStore.

BytesSent

El número de bytes en el cuerpo de la respuesta que envía el servidor de MediaStore. Este valor a menudo es el mismo que el valor del encabezado `Content-Length` incluido con las respuestas del servidor.

ContainerName

El nombre del contenedor que recibió la solicitud.

ErrorCode

El código de MediaStore error (por ejemplo`InternalServerError`). Si no se produce ningún error, aparece el carácter `-`. Podría aparecer un código de error incluso si el código de estado es 200 (que indica una conexión cerrada o un error después de que el servidor iniciara la transmisión de la respuesta).

ExpiresAt

Fecha y hora de caducidad del objeto. Este valor se basa en la edad de caducidad establecida por unatransient data rule política del ciclo de vida que se aplica al contenedor. El valor es una fecha y hora `ISO-8601` y se basa en el reloj del sistema del host que atiende la solicitud. Si la política de ciclo de vida no tiene una regla de datos transitorios que se aplique al objeto, o si no se aplica ninguna política de ciclo de vida al contenedor, el valor de este campo es`null`. Este campo solo se aplica a las siguientes operaciones:`PutObject`,`GetObject`,`DescribeObject`, y`DeleteObject`.

HTTPStatus

El código de estado HTTP numérico de la respuesta.

Operación

La operación realizada, como, por ejemplo, `PutObject` o `ListItems`.

Ruta

La ruta dentro del contenedor en el que se almacena el objeto. Si la operación no toma un parámetro de ruta, aparece el carácter `-`.

ReceivedTime

La hora del día en la que se recibe la solicitud. El valor es una fecha y hora `ISO-8601` y se basa en el reloj del sistema del host que atiende la solicitud.

Solicitante

El nombre de recurso de Amazon (ARN) de usuario de la cuenta utilizada para realizar la solicitud. Para las solicitudes sin autenticar, este valor es `anonymous`. Si la solicitud falla antes de que se complete la autenticación, es posible que este campo no figure en el registro. En ese tipo de solicitudes, `ErrorCode` podría identificar el problema de autorización.

RequestID

Una cadena generada por AWS Elemental MediaStore para identificar de forma inequívoca cada solicitud.

Fuente

La dirección de Internet aparente del solicitante o del principal del servicio de AWS que realiza la llamada. Si los servidores proxy y firewalls intermedios ocultan la dirección de la máquina que realiza la solicitud, el valor se establece en null.

TotalTime

La cantidad de milisegundos (ms) que la solicitud estuvo en tránsito desde la perspectiva del servidor. Este valor se mide comenzando por el momento en que se recibe su solicitud en el servicio y terminando en el momento en que se envía el último byte de la respuesta. Este valor se mide desde la perspectiva del servidor, ya que las medidas realizadas desde la perspectiva del cliente se ven afectadas por la latencia de la red.

TurnAroundTime

La cantidad de milisegundos que se MediaStore tarda en procesar su solicitud. Este valor se mide desde el momento en que se recibió el último byte de su solicitud hasta el momento en que se envió el primer byte de la respuesta.

El orden de los campos en el registro puede variar.

Los cambios del estado del registro se aplican con el tiempo

Los cambios del estado de registros de un contenedor se demoran un tiempo en implementarse efectivamente en el envío de archivos de registro. Por ejemplo, si habilita los registros para un contenedor A, algunas solicitudes que se realizan a la hora siguiente pueden registrarse, mientras que otras no. Si deshabilita el registro para el contenedor B, es posible que se sigan distribuyendo algunos registros durante la siguiente hora, mientras que otros no. En todos los casos, finalmente se aplica la nueva configuración sin que tenga que adoptar medidas adicionales.

Envío de archivos de registro de servidor según el mejor esfuerzo

Las entradas de registro de acceso se envían según el "mejor esfuerzo", es decir, en la medida que sea posible. En la mayoría de las solicitudes de registros para un contenedor debidamente configurado se envían archivos de registro. La mayoría de las entradas de registro se envían en el plazo de unas horas después de su registro, pero se pueden entregar con mayor frecuencia.

No se garantiza que los registros de acceso estén completos ni que lleguen de manera puntual. La entrada de registro de una solicitud determinada puede enviarse mucho después de que la solicitud se haya procesado realmente, y es probable no se envíe en absoluto. El objetivo de los registros de acceso es darle una idea de la naturaleza del tráfico al que se enfrenta su contenedor. Es poco usual perder entradas de registro de acceso, pero los registros de acceso no pretenden ser un recuento completo de todas las solicitudes.

Dada la naturaleza de mejor esfuerzo de la característica de los registros de acceso, los informes de uso disponibles en el portal de AWS (Informes de facturación y administración de costos en la AWS Management Console) podrían incluir una o varias solicitudes de acceso que no aparecen en un registro de acceso enviado.

Consideraciones sobre programación para el formato de registro de acceso

De vez en cuando, podemos ampliar el formato del registro de acceso añadiendo nuevos campos. Se debe escribir el código que analiza los registros de acceso para administrar los campos adicionales que no comprende.

CloudWatch Eventos

Amazon CloudWatch Events le permite automatizarAWS los servicios de y responder automáticamente a eventos del sistema, como problemas de disponibilidad de aplicaciones o cambios de recursos. Puede crear reglas sencillas para indicar qué eventos le resultan de interés, así como qué acciones automatizadas se van a realizar cuando un evento cumple una de las reglas.

> **Important**
>
> NormalmenteAWS los servicios de enviar notificaciones de CloudWatch eventos a Events en cuestión de segundos, pero a veces pueden tardar un minuto o más.

Cuando un archivo se carga en un contenedor o se retira de un contenedor, se activan dos eventos consecutivos en el CloudWatch servicio:

1. the section called “Evento de cambio de estado de objeto”
2. the section called “Evento de cambio de estado de contenedor”

Para obtener información sobre cómo suscribirse a estos eventos, consulte Amazon CloudWatch.

Entre las acciones que se pueden activar automáticamente se incluyen las siguientes:

- Invocar una función de AWS Lambda
- Invocar Ejecutar comando de Amazon EC2
- Desviar el evento a Amazon Kinesis Data Streams
- Activar una máquina de estado de AWS Step Functions
- Notificar un tema o unaAWS SMS cola de Amazon SNS

Algunos ejemplos del uso de CloudWatch eventos con AWS Elemental MediaStore incluyen los siguientes:

- Activación de una función Lambda siempre que se cree un contenedor
- Notificar un tema de Amazon SNS cuando se elimina un objeto

Para obtener más información, consulte la Guía del usuario de Amazon CloudWatch Events.

Temas

- Evento de cambio de estado de MediaStore objetos de AWS Elemental
- Evento de cambio de estado del MediaStore contenedor de AWS Elemental

Evento de cambio de estado de MediaStore objetos de AWS Elemental

Este evento se publica cuando cambia el estado de un objeto (cuando el objeto se carga o se elimina).

> **Note**
>
> Los objetos que caducan debido a una regla de datos transitorios no emiten ningún CloudWatch evento cuando caducan.

Para obtener información sobre cómo suscribirse a este evento, consulte Amazon CloudWatch.

Objeto cargado

```
{
  "version": "1",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "MediaStore Object State Change",
  "source": "aws.mediastore",
  "account": "111122223333",
  "time": "2017-02-22T18:43:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:mediastore:us-east-1:111122223333:MondayMornings/Episode1/
Introduction.avi"
  ],
  "detail": {
    "ContainerName": "Movies",
    "Operation": "UPDATE",
    "Path":"TVShow/Episode1/Pilot.avi",
    "ObjectSize":123456,
    "URL": "https://a832p1qeaznlp9.files.mediastore-us-west-2.com/Movies/
MondayMornings/Episode1/Introduction.avi"
  }
}
```

Objeto eliminado

```
{
  "version": "1",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "MediaStore Object State Change",
  "source": "aws.mediastore",
  "account": "111122223333",
  "time": "2017-02-22T18:43:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:mediastore:us-east-1:111122223333:Movies/MondayMornings/Episode1/
Introduction.avi"
  ],
  "detail": {
    "ContainerName": "Movies",
    "Operation": "REMOVE",
```

```
    "Path":"Movies/MondayMornings/Episode1/Introduction.avi",
    "URL": "https://a832p1qeaznlp9.files.mediastore-us-west-2.com/Movies/
MondayMornings/Episode1/Introduction.avi"
  }
}
```

Evento de cambio de estado del MediaStore contenedor de AWS Elemental

Este evento se publica cuando cambia el estado de un contenedor (cuando el contenedor se añade o se elimina). Para obtener información sobre cómo suscribirse a este evento, consulte Amazon CloudWatch.

Contenedor creado

```
{
  "version": "1",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "MediaStore Container State Change",
  "source": "aws.mediastore",
  "account": "111122223333",
  "time": "2017-02-22T18:43:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:mediastore:us-east-1:111122223333:container/Movies"
  ],
  "detail": {
    "ContainerName": "Movies",
    "Operation": "CREATE"
    "Endpoint": "https://a832p1qeaznlp9.mediastore-us-west-2.amazonaws.com"
  }
}
```

Contenedor eliminado

```
{
  "version": "1",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "MediaStore Container State Change",
  "source": "aws.mediastore",
  "account": "111122223333",
  "time": "2017-02-22T18:43:48Z",
  "region": "us-east-1",
```

```
  "resources": [
    "arn:aws:mediastore:us-east-1:111122223333:container/Movies"
  ],
  "detail": {
    "ContainerName": "Movies",
    "Operation": "REMOVE"
  }
}
```

Supervisión de AWS Elemental MediaStore con CloudWatch métricas de Amazon

Puede monitorear el MediaStore uso de AWS Elemental CloudWatch, que recopila datos sin procesar y los procesa en métricas legibles. CloudWatchmantiene las estadísticas de durante 15 meses, lo que le permite tener acceso a información histórica y disponer de una mejor perspectiva sobre el rendimiento de su aplicación web o servicio. También puede establecer alarmas que vigilen determinados umbrales y enviar notificaciones o realizar acciones cuando se cumplan dichos umbrales. Para obtener más información, consulte la Guía del CloudWatch usuario de Amazon.

En el caso de AWS Elemental MediaStore, es posible que desee ver`BytesDownloaded` y enviarse un correo electrónico cuando esa métrica alcance un umbral determinado.

Para consultar las métricas desde la CloudWatch consola de

Las métricas se agrupan en primer lugar por el espacio de nombres de servicio y, a continuación, por las diversas combinaciones de dimensiones dentro de cada espacio de nombres.

1. Inicie sesión en la CloudWatch consolaAWS Management Console y ábrala en https://console.aws.amazon.com/cloudwatch/.
2. En el panel de navegación, seleccione Metrics (Métricas).
3. En Todas las métricas, elija el espacio deMediaStore nombres AWS/.
4. Elija la dimensión de la métrica para ver las métricas. Por ejemplo, elija `Request metrics by container` para ver las métricas de los diferentes tipos de solicitudes que se han enviado al contenedor.

Para ver métricas mediante la AWS CLI

- En el símbolo del sistema, ejecute el siguiente comando:

```
aws cloudwatch list-metrics --namespace "AWS/MediaStore"
```

MediaStore Métricas de AWS Elemental

La siguiente tabla muestra las métricas a las que se MediaStore envía AWS Elemental CloudWatch.

> **Note**
>
> Para ver las métricas, debes añadir una política de métricas al contenedor que permita MediaStore enviar métricas a Amazon CloudWatch.

Métrica	Descripción
`RequestCount`	Número total de solicitudes HTTP realizadas a un contenedor de MediaStore, separadas por el tipo de operación (`Put`, `Get`, `Delete`, `Describe`, `List`). Unidades: recuento Dimensiones válidas: • Nombre de contenedor • Nombre del grupo de objetos • Tipo de solicitud Estadísticas válidas: Sum
`4xxErrorCount`	La cantidad de solicitudes HTTP realizadas a MediaStore esa dirección provocó un error 4xx. Unidades: recuento Dimensiones válidas: • Nombre de contenedor • Nombre del grupo de objetos • Tipo de solicitud

Métrica	Descripción
	Estadísticas válidas: Sum
`5xxErrorCount`	La cantidad de solicitudes HTTP realizadas a MediaStore esa dirección provocó un error 5xx. Unidades: recuento Dimensiones válidas: • Nombre de contenedor • Nombre del grupo de objetos • Tipo de solicitud Estadísticas válidas: Sum
`BytesUploaded`	El número de bytes cargados para las solicitudes realizadas a un contenedor de MediaStore en las que la solicitud incluye un cuerpo. Unidades: bytes Dimensiones válidas: • Nombre de contenedor • Nombre del grupo de objetos Estadísticas válidas: Average (bytes por solicitud), Sum (bytes por periodo), Sample Count, Min, Max (igual que p100) y cualquier percentil entre p0,0 y p99,0.

Métrica	Descripción
`BytesDownloaded`	Número de bytes descargados para las solicitudes realizadas a un contenedor de MediaStore en las que la respuesta contiene un cuerpo. Unidades: bytes Dimensiones válidas: • Nombre de contenedor • Nombre del grupo de objetos Estadísticas válidas: Average (bytes por solicitud), Sum (bytes por periodo), Sample Count, Min, Max (igual que p100) y cualquier percentil entre p0,0 y p99,0.
`TotalTime`	La cantidad de milisegundos que la solicitud estuvo en tránsito desde la perspectiva del servidor. Este valor se mide desde el momento en que se MediaStore recibe la solicitud hasta el momento en que envía el último byte de la respuesta. Este valor se mide desde la perspectiva del servidor, ya que las medidas realizadas desde la perspectiva del cliente se ven afectadas por la latencia de la red. Unidades: milisegundos Dimensiones válidas: • Nombre de contenedor • Nombre del grupo de objetos • Tipo de solicitud Estadísticas válidas: Average, Min (igual que P0,0), Max (igual que p100), cualquier percentil entre p0,0 y p100

Métrica	Descripción
`TurnaroundTime`	La cantidad de milisegundos que se MediaStore tarda en procesar su solicitud. Este valor se mide desde el momento en que se MediaStore recibe el último byte de la solicitud hasta el momento en que se envía el primer byte de la respuesta. Unidades: milisegundos Dimensiones válidas: • Nombre de contenedor • Nombre del grupo de objetos • Tipo de solicitud Estadísticas válidas: Average, Min (igual que P0,0), Max (igual que p100), cualquier percentil entre p0,0 y p100
`ThrottleCount`	Se limitó el número de solicitudes HTTP realizadas a MediaStore esa dirección. Unidades: recuento Dimensiones válidas: • Nombre de contenedor • Nombre del grupo de objetos • Tipo de solicitud Estadísticas válidas: Sum

Etiquetado de los recursos de AWS Elemental MediaStore

Una etiqueta es un atributo personalizado que usted o AWS asignan a un recurso de AWS. Cada etiqueta tiene dos partes:

- Una clave de etiqueta (por ejemplo, `CostCenter`, `Environment` o `Project`). Las claves de etiqueta distinguen entre mayúsculas y minúsculas.
- Un campo opcional denominado valor de etiqueta (por ejemplo, `111122223333` o `Production`). Omitir el valor de etiqueta es lo mismo que utilizar una cadena vacía. Al igual que las claves de etiqueta, los valores de etiqueta distinguen entre mayúsculas y minúsculas.

Las etiquetas le ayudan a hacer lo siguiente:

- Identificar y organizar sus recursos de AWS. Muchos servicios de AWS admiten el etiquetado, por lo que puede asignar la misma etiqueta a los recursos de diferentes servicios para indicar que los recursos están relacionados. Por ejemplo, puede asignar la misma etiqueta a un MediaStore *`contenedor`* de AWS Elemental que a una AWS Elemental MediaLive entrada.
- Realizar un seguimiento de los costos de AWS. Estas etiquetas se activan en el panel de AWS Billing and Cost Management. AWS utiliza las etiquetas para clasificar los costos y enviarle un informe mensual de asignación de costos. Para obtener más información, consulte Uso de etiquetas de asignación de costos en la Guía del usuario de AWS Billing.

En las siguientes secciones se proporciona más información sobre las etiquetas de AWS Elemental MediaStore.

Recursos compatibles en AWS Elemental MediaStore

Los siguientes recursos de AWS Elemental MediaStore admiten el etiquetado:

- *`contenedor`*

Para obtener información acerca de cómo añadir y administrar etiquetas, consulte Administración de etiquetas.

AWS Elemental MediaStore no admite la función de control de acceso basada en etiquetas de AWS Identity and Access Management (IAM).

Convenciones de nomenclatura y uso de las etiquetas

Las siguientes convenciones básicas de nomenclatura y uso se aplican al uso de etiquetas con MediaStore los recursos de AWS Elemental:

- Cada recurso puede tener un máximo de 50 etiquetas.

- Para cada recurso, cada clave de etiqueta debe ser única y solo puede tener un valor.
- La longitud máxima de la clave de etiqueta es de 128 caracteres Unicode en UTF-8.
- La longitud máxima del valor de etiqueta es de 256 caracteres Unicode en UTF-8.
- Los caracteres permitidos son letras, números y espacios representables en UTF-8, además de los siguientes caracteres: . : + = @ _ / - (guion). Los recursos de Amazon EC2 admiten cualquier carácter.
- Las claves y los valores de las etiquetas distinguen entre mayúsculas y minúsculas. Como práctica recomendada, decida una estrategia de uso de mayúsculas y minúsculas en las etiquetas e implemente esa estrategia sistemáticamente en todos los tipos de recursos. Por ejemplo, decida si se va a utilizar `Costcenter`, `costcenter` o `CostCenter` y utilice la misma convención para todas las etiquetas. Procure no utilizar etiquetas similares con un tratamiento de mayúsculas y minúsculas incoherente.
- El prefijo `aws:` está prohibido para las etiquetas; está reservado para su uso por parte de AWS. Las claves y valores de etiquetas que tienen este prefijo no se pueden editar. Las etiquetas que tengan este prefijo no cuentan para la cuota de etiquetas por recurso.

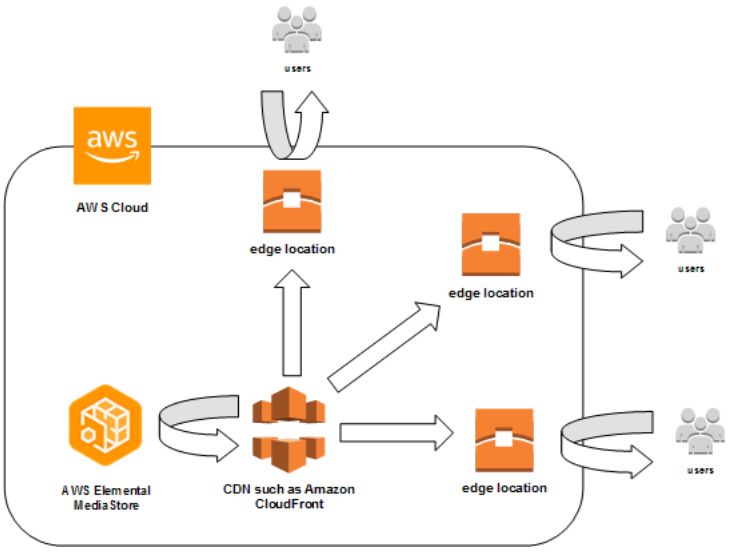
Administración de etiquetas

Las etiquetas se componen de las propiedades `Value` y `Key` de un recurso. Puede usar la API AWS CLI o la MediaStore API para añadir, editar o eliminar los valores de estas propiedades. Para obtener información sobre cómo trabajar con etiquetas, consulte las siguientes secciones de la Referencia de la MediaStore API de AWS Elemental:

- CreateContainer
- ListTagsForResource
- Recursos
- TagResource
- UntagResource

Uso de redes de entrega de contenido (CDN)

Puede utilizar una red de entrega de contenido (CDN) como Amazon CloudFront para ofrecer el contenido que almacena en AWS Elemental MediaStore. Una CDN es un conjunto de servidores distribuidos globalmente que almacena en caché contenido, como, por ejemplo, vídeos. Cuando un usuario solicita contenido, la CDN redirige la solicitud a la ubicación de borde que ofrezca la menor latencia. Si el contenido ya se encuentra en la caché en dicha ubicación de borde, la CDN lo entrega inmediatamente. Si su contenido no se encuentra actualmente en esa ubicación periférica, la CDN lo recupera de su origen (como su MediaStore contenedor) y lo distribuye al usuario.



Temas

- Permitir que Amazon acceda CloudFront a su MediaStore contenedor de AWS Elemental
- Interacción MediaStore de AWS Elemental con cachés HTTP

Permitir que Amazon acceda CloudFront a su MediaStore contenedor de AWS Elemental

Puede utilizar Amazon CloudFront para publicar el contenido que almacena en un contenedor en AWS Elemental MediaStore. Puede hacerlo de una de las siguientes formas:

- Uso del control de acceso a origen (OAC)- (Recomendado) Utilice esta opción siRegión de AWS admite la función OAC de CloudFront.

- Uso de secretos compartidos- Utilice esta opción siRegión de AWS no admite la función OAC de CloudFront.

Uso del control de acceso a origen (OAC)

Puede utilizar la función Origin Access Control (OAC) de Amazon CloudFront para proteger AWS Elemental MediaStore Origins con una seguridad mejorada. Puede habilitar la versión 4 deAWS Signature (Sigv4) en CloudFront las solicitudes de MediaStore orígenes y establecer cuándo y si CloudFront debe firmar las solicitudes. Puede acceder a la función OAC CloudFront a través de la consola, las API, el SDK o la CLI, y su uso no conlleva cargos adicionales.

Para obtener más información sobre el uso de la función OAC con MediaStore, consulte Restringir el acceso a un MediaStore origen en la Guía para CloudFront desarrolladores de Amazon.

Uso de secretos compartidos

Si noRegión de AWS admite la función OAC de Amazon CloudFront, puede adjuntar una política a su MediaStore contenedor de AWS Elemental que conceda acceso de lectura o superior a CloudFront.

> **Note**
>
> Le recomendamos que utilice la función OAC siRegión de AWS la admite. Los siguientes procedimientos requieren que configure MediaStore y CloudFront con secretos compartidos para restringir el acceso a MediaStore los contenedores. Para seguir las mejores prácticas de seguridad, esta configuración manual requiere una rotación periódica de los secretos. Con OAC en MediaStore los orígenes, puede indicar CloudFront que firme las solicitudes mediante SigV4 y reenviarlas MediaStore para que coincidan con las firmas, lo que elimina la necesidad de usar y rotar los secretos. Esto garantiza que las solicitudes se verifiquen automáticamente antes de que se entregue el contenido multimedia, lo que hace que la entrega del contenido multimedia sea más CloudFront sencilla MediaStore y segura.

Para permitir CloudFront el acceso a tu contenedor (consola)

1. Abra la MediaStore consola en https://console.aws.amazon.com/mediastore/.
2. En la página Containers (Contenedores), elija el nombre del contenedor.

   Aparecerá la página de detalles del contenedor.

3. En la sección Política de contenedores, adjunta una política que otorgue acceso de lectura o superior a Amazon CloudFront.

   Example

   La siguiente política de ejemplo, que es similar a la política de ejemplo para el acceso público de lectura a través de HTTPS, cumple con estos requisitos porque permite`GetObject``DescribeObject` recibir órdenes de cualquier persona que envíe solicitudes a tu dominio a través de HTTPS. Además, la siguiente política de ejemplo protege mejor el flujo de trabajo porque solo permite el CloudFront acceso a MediaStore los objetos cuando la solicitud se realiza a través de una conexión HTTPS y contiene el encabezado Referer correcto.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CloudFrontRead",
      "Effect": "Allow",
      "Principal": "*",
      "Action": [
        "mediastore:GetObject",
        "mediastore:DescribeObject"
      ],
      "Resource": "arn:aws:mediastore:<region>:<owner acct
 number>:container/<container name>/*",
      "Condition": {
        "StringEquals": {
          "aws:Referer": "<secretValue>"
        },
        "Bool": {
          "aws:SecureTransport": "true"
        }
      }
    }
 ]}
```

4. En la sección Container CORS policy (Política del CORS de contenedor), asigne un política que permita el nivel de acceso apropiado.

> **Note**
>
> Solo es necesaria una política del CORS si se desea proporcionar acceso a un reproductor basado en navegador.

5. Anote los detalles siguientes:

   - El punto de enlace de datos que se ha asignado a su contenedor de . Puede encontrar esta información en la sección Info (Información) de la página Containers (Contenedores). En CloudFront, el extremo de datos se denomina nombre de dominio de origen.
   - La estructura de carpetas del contenedor donde se almacenan los objetos. En CloudFront, esto se denomina ruta de origen. Tenga en cuenta que este valor es opcional. Para obtener más información sobre las rutas de origen, consulte la Guía para CloudFront desarrolladores de Amazon.

6. En CloudFront, cree una distribución que esté configurada para ofrecer contenido de AWS Elemental MediaStore. Necesitará la información que ha recopilado en el paso anterior.

Después de adjuntar la política a sus MediaStore contenedores, debe configurarlo CloudFront para usar solo conexiones HTTPS para las solicitudes de origen y también agregar un encabezado personalizado con el valor secreto correcto.

Para configurar CloudFront el acceso a su contenedor a través de una conexión HTTPS con un valor secreto para el encabezado Referer (consola)

1. Abra Abra Abra CloudFront la de.
2. En la página Orígenes, elige tu MediaStore origen.
3. Elija Edit (Editar).
4. Elija HTTPS solo para el protocolo.
5. En la sección Agregar encabezado personalizado, selecciona Agregar encabezado.
6. Para el nombre, elija Referer. Para el valor, utilice la misma <secretValue>cadena que utilizó en su política de contenedores.
7. Elige Guardar y deja que se implementen los cambios.

Interacción MediaStore de AWS Elemental con cachés HTTP

AWS Elemental MediaStore almacena los objetos para que las redes de entrega de contenido (CDN) como Amazon puedan almacenarlos en caché de manera correcta y eficiente CloudFront. Cuando un usuario final o una CDN recupera un objeto MediaStore, el servicio devuelve encabezados HTTP que afectan al comportamiento de almacenamiento en caché del objeto. (Los estándares para el comportamiento de almacenamiento en caché HTTP 1.1 se encuentran en la sección 13 de RFC2616). Estos encabezados son:

- **`ETag`** (no personalizable): el encabezado de la etiqueta de entidad es un identificador único para la respuesta que envía MediaStore . Los CDN y los navegadores web que cumplen con los estándares utilizan esta etiqueta como clave para almacenar en caché el objeto. MediaStore genera automáticamente un`ETag` para cada objeto cuando se carga. Puede ver los detalles de un objeto para determinar su valor de ETag.
- **`Last-Modified`**(no personalizable): el valor de este encabezado indica la fecha y la hora en que se modificó el objeto. MediaStore genera automáticamente este valor cuando se carga el objeto.
- **`Cache-Control`** (personalizable): el valor de este encabezado controla cuánto tiempo se debe guardar en caché un objeto antes de que la CDN compruebe si se ha modificado. Puede establecer este encabezado en cualquier valor al cargar un objeto a un MediaStore contenedor mediante la CLI o la API. El conjunto completo de valores válidos se describe en la documentación HTTP/1.1. Si no establece este valor al cargar un objeto, MediaStore no devolverá este encabezado cuando se recupere el objeto.

  Un caso de uso común para el encabezado Cache-Control es especificar una duración para almacenar en caché el objeto. Por ejemplo, supongamos que tiene un archivo de manifiesto de vídeo que un codificador sobrescribe con frecuencia. Puede establecer el `max-age` en 10 para indicar que el objeto debe almacenarlo en caché durante solo 10 segundos. O supongamos que tiene un segmento de vídeo almacenado que nunca se sobrescribirá. Puede establecer el `max-age` para este objeto en 31536000 para almacenarlo en caché durante aproximadamente 1 año.

Solicitudes condicionales

Solicitudes condicionales a MediaStore

MediaStore responde de forma idéntica a las solicitudes condicionales (utilizando encabezados de solicitud como`If-Modified-Since` y`If-None-Match`, como se describe en el RFC7232) y a las

solicitudes incondicionales. Esto significa que cuando MediaStore recibe una`GetObject` solicitud válida, el servicio siempre devuelve el objeto, incluso si el cliente ya lo tiene.

Solicitudes condicionales a CDN

Las CDN que ofrecen contenido en nombre de MediaStore pueden procesar las solicitudes condicionales devolviéndolas`304 Not Modified`, tal como se describe en la sección 4.1 del RFC7232. Esto indica que no es necesario transferir el contenido completo del objeto, porque el solicitante ya tiene un objeto que coincide con la solicitud condicional.

Los CDN (y otras cachés que son compatibles con HTTP/1.1) basan estas decisiones en los encabezados `ETag` y `Cache-Control` que los servidores de origen reenvían. Para controlar la frecuencia con la que las CDN consultan a los servidores de MediaStore origen para obtener actualizaciones de los objetos recuperados repetidamente, defina los`Cache-Control` encabezados de esos objetos cuando los cargue MediaStore.

Cuotas en AWS Elemental MediaStore

La consola Service Quotas proporciona información sobre las MediaStore cuotas de AWS Elemental. Además de ver las cuotas predeterminadas, puede utilizar la consola de cuotas de servicio para solicitar aumentos de cuota para cuotas ajustables.

En la siguiente tabla se describen las cuotas, anteriormente denominadas límites, en AWS Elemental MediaStore. Las cuotas establecen el número máximo de recursos u operaciones de servicio que puede haber en una cuenta de AWS.

> **Note**
>
> Para asignar cuotas a los contenedores individuales de su cuenta, póngase en contacto con el Support de AWS o con su administrador de cuentas. Esta opción puede ayudarte a dividir los límites a nivel de cuenta entre tus contenedores para evitar que un contenedor consuma toda tu cuota.

Recurso u operación	Cuota predeterminada	Comentarios
Contenedores	100	Número máximo de contenedores que puede crear en esta cuenta.
Niveles de carpeta	10	Número máximo de niveles de carpeta que puede crear en un contenedor. Puede crear tantas carpetas como desee, siempre que no estén anidadas más de 10 niveles dentro de un contenedor.
Carpetas	Sin límite	Puede crear tantas carpetas como desee, siempre que no estén anidadas más de 10 niveles dentro de un contenedor.
Tamaño de objeto	25 MB	El tamaño de archivo máximo de un solo objeto.

Recurso u operación	Cuota predeterminada	Comentarios
Objetos	Sin límite	Puede cargar tantos objetos como desee en una carpeta o contenedor de su cuenta.
Tasa de solicitudes de API DeleteObject	100	Número máximo de solicitudes de operación que puede realizar por segundo. Las solicitudes adicionales se limitan. Puede solicitar un aumento de cuota.
Tasa de solicitudes de API DescribeObject	1 000	Número máximo de solicitudes de operación que puede realizar por segundo. Las solicitudes adicionales se limitan. Puede solicitar un aumento de cuota.
Tasa de solicitudes de GetObjectAPI para la disponibilidad de carga estándar	1 000	Número máximo de solicitudes de operación que puede realizar por segundo. Las solicitudes adicionales se limitan. Puede solicitar un aumento de cuota.
Tasa de solicitudes de GetObjectAPI para la disponibilidad de carga en streaming	25	Número máximo de solicitudes de operación que puede realizar por segundo. Las solicitudes adicionales se limitan. Puede solicitar un aumento de cuota.
Tasa de solicitudes de API ListItems	5	Número máximo de solicitudes de operación que puede realizar por segundo. Las solicitudes adicionales se limitan. Puede solicitar un aumento de cuota.

Recurso u operación	Cuota predeterminada	Comentarios
Tasa de solicitudes de PutObject API para la codificación de transferencia fragmentada (también conocida como disponibilidad de carga en streaming)	10	Número máximo de solicitudes de operación que puede realizar por segundo. Las solicitudes adicionales se limitan. Puede solicitar un aumento de cuota. En la solicitud, especifique el TPS solicitado y el tamaño de objeto medio.
Tasa de solicitudes de PutObject API para la disponibilidad de carga estándar	100	Número máximo de solicitudes de operación que puede realizar por segundo. Las solicitudes adicionales se limitan. Puede solicitar un aumento de cuota. En la solicitud, especifique el TPS solicitado y el tamaño de objeto medio.
Reglas de una política de métricas	10	Número máximo de reglas que puede incluir en una política de métrica.
Reglas en una política de ciclo de vida de objetos	10	El número máximo de reglas que puede incluir en una política de ciclo de vida de objetos.

Información MediaStore relacionada con AWS Elemental

En la tabla siguiente, se enumeran los recursos relacionados que le resultarán muy útiles cuando trabaje con AWS Elemental MediaStore.

- Clases y talleres: enlaces a cursos basados en roles y especializados y también a laboratorios autoguiados para ayudarlo aAWS mejorar sus capacidades sobre y obtener experiencia práctica.
- AWSCentro para desarrolladores de: explore los tutoriales, descargue herramientas y obtenga información sobre los eventosAWS para desarrolladores.
- AWSDeveloper Tools (Herramientas para desarrolladores de): enlaces a herramientas para desarrolladores, SDK, conjuntos de herramientas de IDE y herramientas de línea de comandos para desarrollar y administrarAWS aplicaciones de.
- Centro de recursos de introducción: aprenda a configurar su ()Cuenta de AWS, únase a laAWS comunidad y lance su primera aplicación.
- Tutoriales prácticos: comience con step-by-step tutoriales antes de lanzar su primera aplicación enAWS.
- AWSDocumentos técnicos de: enlaces a una lista completa deAWS documentos técnicos de que tratan una gran variedad de temas técnicos, como arquitecturas, seguridad y economía de la nube, escritos por arquitectos deAWS soluciones de o expertos técnicos.
- AWS SupportCentro de : punto para crear y administrar los casos de AWS Support. También incluye enlaces a otros recursos útiles como foros, preguntas técnicas frecuentes, estado de los servicios y AWS Trusted Advisor.
- AWS Support: página web principal para obtener información sobreAWS Support one-on-one, un canal de soporte de respuesta rápida que lo ayudará a crear y ejecutar aplicaciones en la nube.
- Contacte con nosotros – Un punto central de contacto para las consultas relacionadas con la facturación AWS, cuentas, eventos, abuso y demás problemas.
- AWSTérminos del sitio de : información detallada sobre nuestros derechos de autor y marca comercial, su cuenta, licencia y acceso al sitio, entre otros temas.

Historial de revisión de la guía del usuario

En la siguiente tabla, se describe la documentación de la documentación de esta versión de AWS Elemental MediaStore. Para obtener notificaciones sobre las actualizaciones de esta documentación, puede suscribirse a una fuente RSS.

Cambio	Descripción	Fecha
Mejora del control de acceso a origen (OAC)	Se trata de información acerca de cómo utilizar OAC con AWS Elemental MediaStore.	17 de abril de 2023
Actualizaciones de cuotas	Se corrigió el valor de la cuota y la descripción de`Rules in a Metric Policy`.	25 de octubre de 2022
ExpiresAt campo	Los registros de acceso ahora incluyen un`ExpiresAt` campo que indica la fecha y la hora de caducidad del objeto en función de las reglas de datos transitorios de la política de ciclo de vida del contenedor.	16 de julio de 2020
Reglas de transición del ciclo de vida	Ahora puede agregar una regla de transición del ciclo de vida a la política de ciclo de vida de objeto que establezca que los objetos se moverán a la clase de almacenamiento de acceso infrecuente (IA) después de que alcancen una cierta antigüedad.	20 de abril de 2020

Contenedor vacío	Ahora puede eliminar todos los objetos dentro de un contenedor a la vez.	7 de abril de 2020
Support para CloudWatch métricas de Amazon	Puede establecer una política de métricas para determinar a qué métricas se MediaStore envían CloudWatch.	30 de marzo de 2020
Comodín en las reglas de eliminación de objetos	En la política de ciclo de vida de un objeto, ahora puede utilizar un comodín en una regla de eliminación de objetos. Esto le permite especificar los archivos basados en su nombre de archivo o extensión que desea que el servicio elimine después de un determinado número de días.	20 de diciembre de 2019
Políticas del ciclo de vida de los objetos	Ahora puede añadir una regla a la política de ciclo de vida de los objetos que indique un vencimiento por edad en segundos.	13 de septiembre de 2019

Compatibilidad con AWS CloudFormation	A partir de ahora, puede utilizar una plantilla de AWS CloudFormation para crear un contenedor de forma automática. La plantilla de AWS CloudFormation administra los datos para cinco acciones de la API: creación de un contenedor, establecimiento del registro de acceso, actualización de la política de contenedor predeterminada, adición de una política de uso compartido de recursos entre orígenes (CORS) y adición de una política de ciclo de vida de objetos.	17 de mayo de 2019
Cuotas de disponibilidad de subida de contenido en streaming	Para los objetos con disponibilidad de carga de streaming (transferencia fragmentada de objetos), la operación `PutObject` no puede ser superior a 10 TPS y la operación `GetObject` no puede ser superior a 25 TPS.	8 de abril de 2019
Transferencia fragmentada de objetos	Se ha añadido soporte para transferencia fragmentada de objetos. Esta capacidad le permite especificar que un objeto está disponible para su descarga antes de que el objeto se cargue por completo.	5 de abril de 2019

Registro de acceso	MediaStore Ahora, AWS Elemental admite el registro de acceso, que brinda registros detallados para las solicitudes realizadas a los objetos de un contenedor.	25 de febrero de 2019
Políticas del ciclo de vida de los objetos	Se ha agregado soporte para políticas de ciclo de vida de objetos, que rigen la fecha de vencimiento de objetos dentro del contenedor actual.	12 de diciembre de 2018
Mayor cuota de tamaño de objeto	La cuota de tamaño de un objeto ahora es de 25 MB.	10 de octubre de 2018
Mayor cuota de tamaño de objeto	La cuota de tamaño de un objeto ahora es de 20 MB.	6 de septiembre de 2018
Integración de AWS CloudTrail	El contenido CloudTrail de la integración se ha actualizado para alinearlo con los cambios recientes del CloudTrail servicio.	12 de julio de 2018
Colaboración con CDN	Se agregó información sobre cómo usar AWS Elemental MediaStore con una red de entrega de contenido (CDN) como Amazon CloudFront.	14 de abril de 2018

Configuraciones CORS	MediaStore Ahora, AWS Elemental admite Compartir recursos entre orígenes (CORS), lo que permite que las aplicaciones web de los clientes cargadas en un dominio interactúen con los recursos de un dominio diferente.	7 de febrero de 2018
Nuevo servicio y guía	Esta es la versión inicial del servicio de creación y almacenamiento de vídeos, AWS Elemental MediaStore, y de la Guía del MediaStore usuario de AWS Elemental.	27 de noviembre de 2017

Note

- Los ServiciosAWS multimedia no están diseñados ni destinados a utilizarse con aplicaciones ni en situaciones que requieran un rendimiento a prueba de fallos, como operaciones de seguridad personal, sistemas de navegación o comunicación, control de tráfico aéreo o máquinas de soporte vital, en las que la falta de disponibilidad, la interrupción o el fallo de los servicios podrían provocar la muerte, lesiones personales, daños materiales o ambientales.

Glosario de AWS

Para ver la terminología más reciente de AWS, consulte el Glosario de AWS en la Referencia de Glosario de AWS.