StorageGRID® Webscale 11.0

S3 (Servicio Simple de Almacenamiento) Guía de implementación

Octubre 2017 | 215-12408_A0
doccomments@netapp.com

Índice

Compatibilidad con S3 REST API

El sistema StorageGRID Webscale admite el almacenamiento y la recuperación de objetos desde las aplicaciones cliente que interactúan con el sistema StorageGRID Webscale utilizando la Interfaz de Programación de Aplicaciones de Transferencia de Estado Representacional (REST API) del Servicio Simple de Almacenamiento (S3) .

La compatibilidad con la S3 REST API le permite conectar aplicaciones orientadas a servicios desarrolladas para los servicios web S3 con el almacenamiento in-situ de objetos que utiliza el sistema StorageGRID Webscale. Todo ello exige un mínimo de cambios al uso actual de la aplicación cliente de las llamadas S3 REST API.

Cambios a la compatibilidad con S3 REST API

Debe tener en cuenta los cambios realizados en relación con la compatibilidad del sistema StorageGRID Webscale para la S3 REST API.
La siguiente tabla enumera los cambios que se han producido en la compatibilidad del sistema StorageGRID Webscale con la S3 REST API:

Fecha	Versión	Comentarios
Septiembre 2014	10.0	Compatibilidad inicial de la API REST S3 con el sistema StorageGRID Webscale. La versión actualmente compatible con el Simple Storage Service API Reference es la 2006-03-01.
Abril 2015	10.1	Se agregó compatibilidad para la carga múltiple, las peticiones virtuales de estilo alojado y con la autenticación v4.
Diciembre 2015	10.2	Se agregó compatibilidad con las políticas de acceso de grupos y buckets, y para copias múltiples (Cargar Parte - Copia).
Junio 2016	10.3	Se añadió compatibilidad para el Control de versiones.
Abril 2017	10.4	Se añadió compatibilidad para los cambios de escaneado de ILM para el control de versiones, actualizaciones de la página de nombres de dominio de Endpoint, condiciones y variables en políticas, ejemplos de políticas y el permiso PutOverwriteObject.
Octubre 2017	11.0	Se agregó compatibilidad para configurar los servicios de la plataforma (duplicación CloudMirror, notificaciones e integración de búsqueda Elasticsearch) para los buckets. También se agregó compatibilidad con las restricciones de ubicación de etiquetado de objetos para los buckets y con la configuración del control de coherencia disponible.

Versiones compatibles

StorageGRID Webscale es compatible con las siguientes versiones específicas de S3 y HTTP.

Elemento	Versión
Especificación S3	Simple Storage Service API Reference (Referencia API del Servicio Simple de Almacenamiento) 2006-03-01
HTTP	1.1 Si desea obtener más información sobre HTTP, consulte HTTP/1.1 (RFC 2616).

Información relacionada

IETF RFC 2616: Protocolo de Transferencia Hipertexto (HTTP/1.1)

Documentación de los Servicios Web de Amazon (AWS): Referencia a la API del Simple Storage Service (Servicio Simple de Almacenamiento)

Compatibilidad con los servicios de plataforma StorageGRID Webscale

Los servicios de plataforma StorageGRID Webscale le permiten aprovechar los servicios externos tales como un bucket S3 remoto, un endpoint del Servicio de notificación simple (SNS) o un clúster Elasticsearch para ampliar los servicios proporcionados por una malla.

Atención: StorageGRID Webscale 11.0 incluye la versión inicial de los servicios de plataforma. En la actualidad, la duplicación CloudMirror, las notificaciones y la integración de búsquedas solo resultan apropiadas para determinadas situaciones y cargas de trabajo. Tendrá que ponerse en contacto con su representante de NetApp si desea utilizar la versión inicial de estos servicios.

La Guía del Administrador del tenant incluye una descripción completa de los servicios de la plataforma, incluidas las instrucciones sobre cómo crear un endpoint que represente al servicio remoto en su cuenta tenant, algo necesario antes de que se pueda configurar un servicio. Esta guía analiza la implementación de StorageGRID Webscale en las API de configuración.

La siguiente tabla resume los servicios de plataforma disponibles y las API utilizadas para configurarlos.

Servicio de plataforma	Propósito	S3 API utilizada para configurar el servicio
Duplicación CloudMirror	Duplica objetos desde un bucket origen de StorageGRID Webscale al bucket S3 configurado de forma remota.	PUT Duplicación de bucket
Notificaciones	Envía notificaciones sobre eventos ocurridos en un bucket origen de StorageGRID Webscale hacia un endpoint configurado del Servicio Simple de Notificación (SNS).	PUT notificación de bucket
Buscar integración	Envía objetos de metadatos de los objetos almacenados en un bucket StorageGRID Webscale hacia un índice configurado de Elasticserach.	PUT notificación de metadatos de buckets **Nota:** Se trata de una API S3 personalizada de StorageGRID Webscale.

Antes de poder utilizar los servicios de la plataforma, un administrador de la malla debe permitir el

uso de los mismos a una cuenta tenant, tal y como se describe en la Guía del Administrador.

Conceptos relacionados

Operaciones sobre buckets en la página 18

Referencias relacionadas

Solicitud de configuración de notificación de metadatos PUT Bucket en la página 45

Información relacionada

Guía del Administrador de tenant de la aplicación StorageGRID Webscale 11.0
Guía del administrador de StorageGRID Webscale 11.0

Cómo utilizan las aplicaciones cliente las conexiones HTTP

Las aplicaciones cliente utilizan las conexiones HTTP para acceder y comunicarse con el sistema StorageGRID Webscale.

Las aplicaciones cliente se conectan directamente con un Nodo Gateway de la API o con un Nodo de Almacenamiento para almacenar y recuperar objetos. Para equilibrar la carga entre los nodos de almacenamiento, puede conectarse a un nodo Gateway de la API, que gestionará por usted el equilibrio de la carga. De lo contrario, puede conectarse directamente con un Nodo de Almacenamiento.

Nota: IPv6 solo es compatible con conexiones de aplicaciones cliente mediante el Nodo Gateway de la API.

Las aplicaciones cliente pueden emitir peticiones "OPTIONS /" HTTPS al puerto S3 en un nodo de almacenamiento, sin proporcionar credenciales de autenticación S3, para determinar si el servicio LDR está disponible. Puede utilizar esta solicitud para supervisar o para permitir que los equilibradores de carga externos identifiquen los periodos de inactividad de un Nodo de Almacenamiento.

Configurar la conexión a las aplicaciones clientes implica las siguientes tareas:

- Creación de una cuenta S3 para el tenant.
- Identificar direcciones IP para los Nodos de almacenamiento y los nodos de pasarela de la API
- Identificar los números de puerto S3 para los Nodos de Almacenamiento y los Nodos de Pasarela de la API
- Copiar el certificado de la autoridad certificadora del sistema (CA) para aplicaciones cliente que requieren validación del servidor

Información relacionada

Guía del administrador de StorageGRID Webscale 11.0

Cuentas tenant S3 en StorageGRID Webscale

Puede configurar el sistema StorageGRID Webscale para aceptar conexiones desde aplicaciones cliente desarrolladas originalmente para usar servicios web S3 creando una cuenta tenant S3.

Las cuentas tenant S3 se crean y eliminan utilizando la opción del menú **Tenants** de la interfaz de gestión o utilizando la API de gestión. Los grupos y usuarios de S3, incluyendo las claves de acceso S3, las opciones de bucket S3 y las políticas de grupo S3, se gestionan utilizando la Interfaz de Gestión del tenant o utilizando la API del tenant. Cuando configure un cliente S3, debe proporcionar la información de la cuenta S3 que se utiliza en el proceso de autenticación. Si desea obtener más información, consulte la Guía del Administrador y la Guía del Administrador del tenant.

Información relacionada

Guía del administrador de StorageGRID Webscale 11.0
Guía del Administrador del tenant de StorageGRID Webscale 11.0

Cómo especificar nombres de dominio de endpoint para la API de S3

Para permitir las peticiones de estilo de host virtual S3, deberá configurar la lista de los nombres de dominio de endpoint a los que se conectarán los clientes S3.

Antes de comenzar

- Debe iniciar sesión en la Interfaz de gestión utilizando un navegador compatible.
- Para realizar esta tarea necesita disponer de permisos específicos de acceso. Para obtener más detalles, consulte la información sobre el control de acceso al sistema con cuentas y grupos de usuarios de administración.

Acerca de esta tarea

Los nombres de dominio de endpoint de la API se configuran después de crear los nombres de dominio plenamente calificados en el servidor DNS, dependiendo de los nodos de la malla a los que se conectarán los clientes de S3:

- Si los clientes S3 se conectan a uno o más Nodos de la Pasarela de la API, deberá incluir el nombre de dominio de cada Nodo de la Pasarela de la API.
- Si los clientes S3 se conectan a uno o más Nodos de Almacenamiento, debe incluir el nombre de dominio de cada Nodo de Almacenamiento.
- Si los clientes S3 se conectan a través de un equilibrador externo de carga, debe incluir el nombre de dominio del equilibrador de carga.

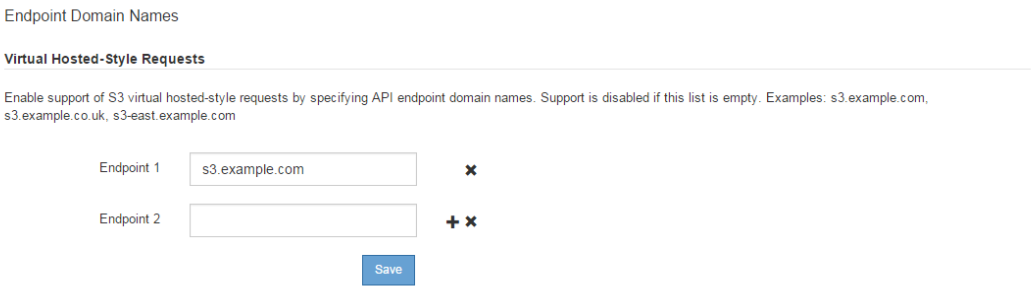
Si esta lista está vacía, se desactivará la compatibilidad con las peticiones virtuales de estilo host de S3.

También deberá configurar un certificado personalizado del servidor para el sistema StorageGRID Webscale con un Nombre Alternativo de Sujeto de tipo comodín (SAN) para cada endpoint y, posiblemente, para cada dirección IP. Estos pasos resultan necesarios para validar el certificado SSL y verificar el nombre de host cuando las aplicaciones cliente API se conectan con el endpoint.

Pasos

1. Seleccione **Configuration > Domain Names** (Configuración > Nombres de dominio).

   Aparece la página Endpoint Domain Names (Nombres de Dominio de endpoint)



2. Utilizando el icono (+) para agregar nuevos campos, escriba la lista de nombres de dominio de endpoint de la API de S3 en los campos **Número de Endpoint** y haga clic en **Save (Guardar).**

   Aviso: No realice ningún cambio en la configuración del nombre de dominio cuando esté en marcha la actualización de una malla.

Información relacionada

Guía del administrador de StorageGRID Webscale 11.0

Identificar direcciones IP para los Nodos de almacenamiento y los nodos de pasarela de la API

Necesita la dirección IP del nodo de la malla para conectar las aplicaciones cliente de la API a StorageGRID Webscale.

Pasos

1. Debe iniciar sesión en la Interfaz de gestión utilizando un navegador compatible.
2. Seleccione **Grid** (Malla).
3. En el árbol **Grid Topology** (Topología de la Malla), localice y expanda el Nodo de Almacenamiento o el Nodo de la Pasarela de la API con la que desea conectarse.

   Aparecerán los servicios asociados con la malla seleccionada.
4. En el Nodo de Almacenamiento o en el Nodo de Pasarela de la Api, seleccione **SSM > Resources** (Recursos) y, posteriormente, desplácese hasta la tabla **Network Addresses** (Direcciones de red).

   Puede establecer conexiones HTTPS desde las aplicaciones cliente de la API a cualquiera de las direcciones IP enumeradas.

Números de puerto para los Nodos de Almacenamiento y los Nodos de Pasarela de la API

Los Nodos de Pasarela y los Nodos de Almacenamiento de la API solo están disponibles para conexiones HTTP desde aplicaciones cliente al sistema StorageGRID Webscale en determinados puertos específicos.

Los siguientes puertos son utilizados por las aplicaciones cliente que se comunican con el sistema StorageGRID Webscale a través de S3:

Nodo de la malla	Número de puerto
Nodo de la Pasarela API (Puerto CLB S3)	8082
Nodo de almacenamiento (Puerto LDR S3)	18082

Cómo probar su configuración S3 REST API

Puede utilizar la interfaz de la línea de mandatos de los Servicios Web de Amazon (AWS CLI) para probar su conexión con el sistema y verificar que puede leer y escribir objetos en el sistema.

Antes de comenzar

- Deberá haber descargado e instalado la AWS CLI desde aws.amazon.com/cli.
- Debe haber creado una cuenta tenant S3 en el sistema StorageGRID Webscale.

Pasos

1. Configure las opciones de los servicios Web de Amazon para usar la cuenta que creó en el sistema StorageGRID Webscale:

   a. Especifique el modo de configuración:

   ```
   aws configure
   ```

   b. Especifique el ID de la clave de acceso AWS para la cuenta que ha creado.

c. Especifique la clave de acceso secreta AWS para la cuenta que ha creado.

d. Especifique la región predeterminada a utilizar, por ejemplo, us-east-1.

e. Especifique el formato de salida predeterminado a utilizar o pulse **Intro** para seleccionar JSON.

2. Cree un bucket.

```
aws s3api --endpoint-url https://10.96.101.17:8082
--no-verify-ssl create-bucket --bucket testbucket
```

Si el bucket se crea con éxito, se obtendrá la ubicación del bucket, como se muestra en el siguiente ejemplo:
`"Location": "/testbucket"`

3. Cargue un objeto.

```
aws s3api --endpoint-url https://10.96.101.17:8082 --no-verify-ssl put-
object --bucket testbucket --key s3.pdf --body C:\s3-test\upload
\s3.pdf
```

Si el objeto se transfiere con éxito, se devuelve un Etag que es un hash de los datos del objeto.

4. Liste los contenidos del bucket para verificar que se transfirió el objeto.

```
aws s3api --endpoint-url https://10.96.101.17:8082 --no-verify-ssl
list-objects --bucket testbucket
```

5. Elimine el objeto.

```
aws s3api --endpoint-url https://10.96.101.17:8082 --no-verify-ssl
delete-object --bucket testbucket --key s3.pdf
```

6. Elimine el bucket

```
aws s3api --endpoint-url https://10.96.101.17:8082 --no-verify-ssl
delete-bucket --bucket testbucket
```

En qué forma StorageGRID Webscale implementa la S3 REST API

Una aplicación cliente puede usar llamadas a la API REST S3 para conectarse a Nodos de Almacenamiento y Nodos de Pasarela API, para crear buckets y para almacenar y recuperar objetos.

Para gestionar estos objetos, el sistema StorageGRID Webscale utiliza reglas de gestión del ciclo de vida de la información (ILM).

Para obtener más información sobre las reglas ILM, consulte la Guía del Administrador.

Solicitudes conflictivas del cliente

Las solicitudes conflictivas de los clientes, como cuando dos clientes escriben la misma clave, se resuelven según la regla "la última gana". La temporización de la evaluación "la última gana" se basa en el momento en que el sistema StorageGRID Webscale completa una solicitud determinada y no cuándo los clientes S3 comienzan una operación.

Garantías y controles de coherencia

De forma predeterminada, StorageGRID Webscale garantiza la coherencia "lectura tras escritura" para los objetos de reciente creación. Cualquier GET que siga a un PUT completado con éxito podrá leer los datos recién escritos. Las sobrescrituras de objetos existentes, actualizaciones de metadatos y eliminaciones son finalmente coherentes. Las sobrescrituras generalmente tardan segundos o minutos en propagarse, pero pueden demorarse hasta 15 días.

StorageGRID Webscale le permitirá controlar el control de coherencia utilizado para cada solicitud API o bucket. Puede modificar el control de coherencia para compensar entre la disponibilidad de los objetos y la coherencia de esos objetos en diferentes sitios y Nodos de Almacenamiento, según lo requiera su aplicación.

Si desea obtener más información sobre cómo configurar los controles de consistencia para buckets, consulte solicitud de coherencia de GET Bucket en la página 36 y solicitud de coherencia de PUT Bucket en la página 37.

Para ajustar el control de coherencia para una operación API individual, los controles de coherencia deben ser compatibles con la operación y debe especificar el control de coherencia en la cabecera de la solicitud. Este ejemplo define el control de coherencia para **strong-site** para una operación GET Objeto.

```
GET /bucket/object HTTP/1.1
Date: Sat, 29 Nov 2015 01:02:17 GMT
Authorization: AWS 9MOYPG9ACWPAJA1S72R5:jUGbYkLdBApjCWBgK4TxvOjfock=
Host: test.com
Consistency-Control: strong-site
```

Puede definir la cabecera `Consistency-Control` con uno de los siguientes valores:

- **all** (todo): Proporciona la mayor garantía de la coherencia "lectura tras la escritura". Todos los nodos reciben los datos inmediatamente o la solicitud fallará.
- **strong-global** (fuerte-global): Garantiza la coherencia "lectura tras la escritura" para todas las solicitudes de los clientes en todas las sedes.
- **strong-site** (fuerte-sede): Garantiza la coherencia "lectura tras la escritura" para todas las solicitudes de los clientes dentro de una sede.
- **default** (predeterminado) ("lectura tras la escritura" para nuevos objetos): Proporciona coherencia "lectura tras la escritura" para nuevos objetos y coherencia final para actualizaciones de objetos. Ofrece alta disponibilidad y garantías de protección de datos. Coincide con las garantías de coherencia de AWS S3.

Nota: Si su aplicación intenta operaciones HEAD sobre claves que no existen, ajuste el nivel de coherencia a **available** (disponible) salvo que requiera garantías de coherencia de AWS S3. De lo contrario, se pueden producir un elevado número de errores del tipo "500 Internal Server error" (error interno del servidor) si uno o más nodos de almacenamiento no se encuentran disponibles.

- **available** (disponible) (coherencia final para operaciones HEAD): Se comporta igual que el nivel de coherencia **default** (predeterminado), pero solo proporciona una coherencia final para las operaciones HEAD. Ofrece una disponibilidad superior para las operaciones HEAD que **default** para el caso de que los Nodos de Almacenamiento no estén disponibles. Se distingue de las garantías de coherencia de AWS S3 solo para las operaciones HEAD.
- **weak** (débil): Proporciona coherencia final y alta disponibilidad, con garantías mínimas de protección de datos, especialmente si falla un Nodo de Almacenamiento o no está disponible. Adecuado solo para cargas de trabajo pesadas en escritura que requieran alta disponibilidad, no requieran coherencia de "lectura tras escritura" y pueden tolerar la posible pérdida de datos si falla un nodo.

El siguiente comportamiento se aplica a las operaciones realizadas sobre el bucket:

- Los controles de coherencia solo afectan a los objetos dentro de un bucket. Establecer el control de coherencia para un bucket no afecta al propio bucket; en su lugar, define el control de coherencia para las operaciones S3 realizadas sobre los objetos contenidos en el propio bucket.

  Nota: En general, podrá usar el valor de control de coherencia **default** (predeterminado). Cuando las solicitudes no funcionen correctamente, si es posible modifique el comportamiento del cliente de la aplicación. O bien, configure el cliente para que especifique el control de coherencia para cada solicitud de API. Defina el control de coherencia a nivel bucket solo como último recurso.

- El comportamiento predeterminado de la operación S3 HEAD es comenzar con una baja coherencia. Si el resultado del primer intento HEAD es Not Found (No encontrado) la malla reintenta la operación con una mayor coherencia en cada reintento, hasta que se utilice **todo** el valor de control de coherencia. Si utiliza la operación HEAD sobre una clave que no existe, los reintentos "burbujearán" a **todos** los controles de coherencia, lo que requiere que todos los nodos se encuentren en línea. En este caso se pueden producir un elevado número de errores del tipo "500 Internal Server error" (error interno del servidor) si uno o más Nodos de almacenamiento no se encuentran disponibles. Para evitar estos errores, cambie el control de coherencia a **available** (disponible). Esto detendrá el "burbujeo" del comportamiento de coherencia para las operaciones HEAD, aunque ya no se garantizará la coherencia de "lectura tras escritura" en los metadatos de objeto recién creados.
- El "burbujeo" del comportamiento de coherencia también se aplica a las operaciones GET. Sin embargo, las operaciones PUT y DELETE no siguen este patrón; tienen éxito o fallan en el control de coherencia especificado.

Nota: Debe usar la misma cabecera de `Control de coherencia` de StorageGRID Webscale para las operaciones PUT Objeto y GET Objeto. Por ejemplo, si utiliza **weak** para escribir un objeto y luego emplea **strong-global** para leer el mismo objeto, no proporcionará una coherencia fuerte en todas las sedes.

Información relacionada

Guía del administrador de StorageGRID Webscale 11.0

En qué forma las reglas de StorageGRID Webscale ILM gestionan los objetos

Puede crear reglas de administración del ciclo de vida de la información (ILM) que gestione datos de objetos ingeridos en el sistema StorageGRID Webscale desde aplicaciones cliente de la API REST S3. Utilice estas reglas de ILM para determinar cómo y dónde se almacenarán los datos de

objeto a lo largo del tiempo.

Las opciones de ILM determinan los siguientes aspectos de un objeto:

Geography (Geografía)
La ubicación de los datos de un objeto dentro del sistema StorageGRID Webscale.

Storage grade (grado de almacenamiento)
El tipo de almacenamiento utilizado para almacenar datos de objetos (disco o medios de archivo).

Loss protection (Protección contra pérdida)
Cómo se hacen las copias: duplicación, codificación de borrado o ambas.

Retention (Retención)
Los cambios a lo largo del tiempo sobre cómo se administran los datos de un objeto, dónde se almacenan y cómo están protegidos contra pérdidas.

Puede configurar reglas de ILM para filtrar y seleccionar objetos. Para los objetos ingeridos usando S3, puede filtrar objetos en base a los siguientes metadatos:

- Cuenta tenant
- Nombre del bucket
- Hora de ingestión
- Clave
- Última hora de acceso

  Nota: De forma predeterminada, las actualizaciones de la última hora de acceso se encuentran deshabilitadas para todos los buckets S3. Si su sistema StorageGRID Webscale incluye una regla ILM que utiliza la opción Last Access Time (última hora de acceso), deberá habilitar las actualizaciones a la última hora de acceso para los buckets S3 especificados en dicha regla. Puede habilitar las actualizaciones de la última hora de acceso utilizando la solicitud de la última hora de acceso de PUT Bucket, la casilla de verificación **S3 > Buckets > Configure Last Access Time** en la Interfaz de Administración del tenant o en la API de Administración del tenant. Al habilitar las actuaciones de última hora de acceso, deberá tener en cuenta que se puede disminuir el rendimiento de StorageGRID Webscale, especialmente en aquellos sistemas que cuenten con pequeños objetos. Consulte la Guía del Administrador del tenant para obtener más información.

- Restricción de ubicación
- Tamaño del objeto
- Metadatos de usuario
- Etiqueta objeto

Para obtener más detalles sobre las reglas ILM, consulte la Guía del Administrador.

Información relacionada

Guía del Administrador del tenant de StorageGRID Webscale 11.0
Guía del administrador de StorageGRID Webscale 11.0

Control de versiones de objetos

Puede utilizar el control de versiones para conservar varias versiones de un objeto, lo que protege contra la eliminación accidental de objetos, y le permite recuperar y restaurar versiones anteriores de un objeto.

El sistema StorageGRID Webscale implementa el control de versiones con soporte para la mayoría de las funciones y con algunas limitaciones.

En StorageGRID Webscale, el control de versiones de objetos se puede combinar con la Gestión del

Ciclo de vida de la Información (ILM), en lugar de utilizar la Administración del Ciclo de vida de Objetos, que no es compatible en este caso. Debe habilitar explícitamente el control de versiones para cada bucket para activar esta funcionalidad para dicho bucket. A cada objeto de su bucket se le asigna una ID de versión generada por el sistema StorageGRID Webscale.

No es compatible con la autenticación mediante múltiples factores (MFA).

Nota: El control de versiones solo se puede habilitar en los buckets creados con StorageGRID Webscale Versión 10.3 o posterior.

ILM y el control de versiones

Las políticas de ILM se aplican a cada versión de un objeto. Un proceso de escaneo de ILM explora continuamente todos los objetos y los vuelve a evaluar contra la política actual de ILM. Cualquier cambio que realice a las políticas ILM se aplicará a todos los objetos ingeridos previamente. Esto incluye a las versiones previamente ingeridas si se ha habilitado el control de versiones. La exploración ILM aplica nuevos cambios de ILM a los objetos previamente ingeridos.

Para los objetos S3 contenidos en los buckets habilitados para el control de versiones, la compatibilidad con el control de versiones permite la creación de reglas ILM que utilizan un tiempo de referencia no actualizado. Cuando ese objeto se actualiza, este enfoque provoca que sus versiones anteriores dejen de estar actualizadas. Si utiliza un filtro de tiempo no actualizado podrá crear políticas que reduzcan el impacto del almacenamiento de versiones anteriores de los objetos.

Limitaciones y operaciones soportados por la S3 REST API

El sistema StorageGRID Webscale implementa la Referencia API del Servicio Simple de Almacenamiento (API Versión 2006-03-01) compatible con la mayoría de las operaciones y con ciertas limitaciones. Debe conocer los detalles de la implementación cuando esté integrando aplicaciones cliente de API REST S3.

El sistema StorageGRID Webscale permite las solicitudes de estilo host virtual y las peticiones de estilo ruta.

Información relacionada

Documentación de los Servicios Web de Amazon (AWS): Referencia a la API del Simple Storage Service (Servicio Simple de Almacenamiento) de Amazon

Respuestas de error

El sistema StorageGRID Webscale permite el empleo de todos los estándares de respuestas de error de S3 REST API que son de aplicación. Además, la implementación de StorageGRID Webscale añade las respuestas personalizadas `XNotImplemented` y`NoSuchEndpoint` .

Códigos de error API de S3 utilizados

Nombre	Estado HTTP
AccessDenied (Acceso denegado)	403 Forbidden (Prohibido)
BadDigest (resumen erróneo)	400 Bad Request (Solicitud errónea)
BucketAlreadyExists (El bucket ya existe)	409 Conflict (conflicto)
BucketNotEmpty (El bucket no está vacío)	409 Conflict (conflicto)
IncompleteBody (Cuerpo incompleto)	400 Bad Request (Solicitud errónea)
InternalError (error interno)	500 Internal Server Error (Error interno del servidor)
InvalidAccessKeyId (Id de la clave de acceso no válida)	403 Forbidden (Prohibido)

InvalidBucketName (Nombre de bucket no válido)	400 Bad Request (Solicitud errónea)
InvalidDigest (Resumen no válido)	400 Bad Request (Solicitud errónea)
InvalidEncryptionAlgorithmError (error en el algoritmo de cifrado, no válido)	400 Bad Request (Solicitud errónea)
InvalidPart (Parte no válida)	400 Bad Request (Solicitud errónea)
InvalidPartOrder (Solicitud de parte no válida)	400 Bad Request (Solicitud errónea)
InvalidRange (Rango no válido)	416 Requested Range Not Satisfiable (Rango solicitado no satisfactorio)
InvalidRequest (Solicitud no válida)	400 Bad Request (Solicitud errónea)
InvalidStorageClass (Clase de almacenamiento	400 Bad Request (Solicitud errónea)
InvalidTag (Etiqueta no válida)	400 Bad Request (Solicitud errónea)
InvalidURI (URI no válida)	400 Bad Request (Solicitud errónea)
KeyTooLong (Clave demasiado larga)	400 Bad Request (Solicitud errónea)
MalformedXML (XML mal formada)	400 Bad Request (Solicitud errónea)

Nombre	**Estado HTTP**
MetadataTooLarge (Metadatos demasiado largo)	400 Bad Request (Solicitud errónea)
MethodNotAllowed (Método no permitido)	405 Method Not Allowed (Método no permitido)
MissingContentLength (longitud de contenido	411 Length Required (se necesita la longitud)
MissingRequestBodyError (error del cuerpo de la solicitud no encontrada)	400 Bad Request (Solicitud errónea)
MissingSecurityHeader (Cabecera de seguridad	400 Bad Request (Solicitud errónea)
NoSuchBucket (no tal bucket)	404 Not Found (No encontrada)
NoSuchKey (No tal clave)	404 Not Found (No encontrada)
NoSuchUpload (No tal carga)	404 Not Found (No encontrada)
NotImplemented (No implementada)	501 Not Implemented (No implementada)
NoSuchBucketPolicy (No tal política de bucket)	404 Not Found (No encontrada)
PreconditionFailed (Condición previa fallida)	412 Precondition Failed (Condición previa fallida)
RequestTimeTooSkewed (Solicitud temporal	403 Forbidden (Prohibido)
ServiceUnavailable (Servicio no disponible)	503 Service Unavailable (Servicio no disponible)
SignatureDoesNotMatch (La firma no coincide)	403 Forbidden (Prohibido)
TooManyBuckets (Demasiados buckets)	400 Bad Request (Solicitud errónea)
UserKeyMustBeSpecified (Se debe especificar	400 Bad Request (Solicitud errónea)

Códigos de error específicos de StorageGRID Webscale

Código de error	Descripción	Código de estado de HTTP
NoSuchEndpoint (No hay tal endpoint)	El endpoint que intenta utilizar para configurar la duplicación del Bucket o las notificaciones de metadatos del Bucket no existe.	404 Not Found (No encontrado)
XNotImplemented (No implementado)	La solicitud que proporcionó implica una funcionalidad que no está implementada.	501 Not Implemented (No implementada)

Gestión de fechas

La implementación de StorageGRID Webscale de la API REST de S3 solo es compatible con los formatos válidos de fecha de HTTP.

El sistema StorageGRID Webscale solo permite el uso de formatos de fecha válidos para HTTP para cualquier cabecera que acepte valores de datos. La porción que indica la hora dentro de la fecha se puede especificar en formato de Greenwich Mean Time (GMT, Hora de Greenwich) o en formato de Hora Universal Coordinada (UTC) sin desfase de zona horaria (se debe especificar +0000). Si incluye la cabecera `x-amz-date` en su solicitud, sobreescribirá cualquier valor especificado en la cabecera de solicitud de Fecha. Cuando emplee la firma AWS Versión 4, la cabecera `x-amz-fecha` debe estar presente en la solicitud firmada ya que no se puede utilizar la cabecera de fecha.

Cabeceras de solicitud común

El sistema StorageGRID Webscale permite el empleo de cabeceras de solicitud común definidas por la Referencia API del Servicio Simple de Almacenamiento, con un par de excepciones.

La siguiente lista enumera las cabeceras de solicitud que no son compatibles:

Cabecera de solicitud	Implementación
Authorization (Autorización)	Compatibilidad total con la firma AWS versión 2 Compatibilidad con la firma AWS Versión 4, con las siguientes excepciones: • El valor SHA256 no se calcula en el cuerpo de la solicitud. El valor enviado por el usuario se acepta sin validación.
x-amz-copy-source-server-side-encryption-customer-key	No implementado.
x-amz-copy-source-server-side-encryption-customer-key-MD5	No implementado.
x-amz-security-token	No implementado. Devuelve `XNotImplemented`.
x-amz-server-side-encryption-customer-algorithm	No implementado.

Cabeceras comunes de respuesta

El sistema StorageGRID Webscale permite el uso de cabeceras de respuesta común con algunas excepciones.

El sistema StorageGRID Webscale permite el uso de todas las cabeceras de respuesta común definidas por la Referencia API del Servicio Simple de Almacenamiento, con las siguientes excepciones:

Cabecera de respuesta	Implementación
x-amz-id-2	No usada
X-amz-expiration	No usada

Autenticación de solicitudes

El sistema StorageGRID Webscale permite el empleo de acceso autenticado y anónimo a objetos que utilicen la API S3.

La API S3 permite el uso de Firma Versión 2 y Firma Versión 4 para la autenticación de solicitudes API S3. Las solicitudes autenticadas se deben firmar utilizando su ID de clave de acceso y la clave de acceso secreta.

El sistema StorageGRID Webscale permite el empleo de dos métodos de autenticación: La cabecera **`Authorization`** de HTTP y el empleo de parámetros de consulta.

Empleo de la cabecera Authorization de HTTP

La cabecera **`Authorization`** de HTTP es utilizada por todas las operaciones API S3 salvo las solicitudes Anonymous cuando así lo permite la política de bucket. La cabecera **`Authorization`** contiene toda la información de firma necesaria para autenticar una solicitud.

Empleo de los parámetros de consulta

Puede usar los parámetros de consulta para agregar información de autenticación a una URL. Esto se conoce como la pre-firma de la URL, que se puede usar para otorgar acceso temporal a recursos específicos. Los usuarios con la URL pre-firmada no necesitan conocer la clave de acceso secreta para acceder al recurso, lo que le permite proporcionar a terceros acceso restringido a un recurso.

Operaciones sobre el servicio

El sistema StorageGRID Webscale permite el empleo de las siguientes operaciones sobre el servicio:

Operación	Implementación
GET Servicio	Implementado con todo el comportamiento de Amazon S3 REST API.
GET Storage Usage	La solicitud GET Empleo de Almacenamiento le indica la cantidad total de almacenamiento que una cuenta está utilizando, y para cada uno de los buckets asociados con la cuenta. Se trata de una operación sobre el servicio al que se añade una ruta de / y un parámetro de consulta personalizado (?x-ntap-sg-usage).
OPCIONES /	Las aplicaciones cliente pueden emitir solicitudes al puerto S3 en un Nodo de Almacenamiento, sin proporcionar credenciales de autenticación S3, para determinar si el servicio LDR está disponible.

Operaciones sobre buckets

El sistema StorageGRID Webscale permite el uso de un máximo de 1000 buckets por cuenta tenant S3. Los nombres de bucket cumplen las restricciones de la región estándar AWS US Standard, pero debe restringirlas aún más siguiendo las convenciones de denominación de DNS para admitir las solicitudes de estilo de host virtual de S3.

Las operaciones de versiones GET Bucket y GET Bucket (Lista de Objetos) son compatibles con los controles de coherencia de StorageGRID Webscale. Para obtener más información sobre cómo utilizar la cabecera `Consistency-Control`, consulte Cómo StorageGRID Webscale implementa el S3 REST API en la página 11.

Puede comprobar si las actualizaciones a la hora del último acceso se encuentran habilitadas o deshabilitadas para buckets individuales. Para obtener más información, consulte Solicitud de la hora del último acceso a GET Bucket en la página 40.

Para otras operaciones sobre buckets consulte "Operaciones personalizadas sobre buckets."

La siguiente tabla describe qué operaciones sobre buckets se encuentran implementadas, y la forma en que se implementan, en el sistema StorageGRID Webscale.

Operación	Implementación
DELETE Bucket	Implementada con todo el comportamiento de Amazon S3 REST API.
DELETE Bucket Policy	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación borrará la política unida al bucket. Si desea obtener más información sobre el lenguaje de política utilizado en el sistema StorageGRID Webscale, consulte Políticas de acceso al grupo y bucket en la página 50.

DELETE Bucket replication	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación borrará la configuración de duplicación unida al bucket.

Operación	Implementación
GET Bucket (Lista de objetos)	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación devuelve algunos o todos (hasta 1000) los objetos contenidos en un bucket. La Storage Class para objetos se lista siempre como STANDARD, incluso cuando los objetos se toman especificando la clase de almacenamiento REDUCED_REDUNDANCY. Cuando un objeto se introduce en StorageGRID Webscale con la clase de almacenamiento REDUCED_REDUNDANCY, significará que los objetos se introducen utilizando una operación de ingesta de compromiso único (single-commit). Este hecho no implica que el objeto se almacene en niveles inferiores de redundancia en el sistema StorageGRID Webscale. **Atención:** Tenga cuidado cuando ingiera objetos utilizando REDUCED_REDUNDANCY para crear solo una copia inicial única de los datos del objeto. Si la copia única se crea en un Nodo de Almacenamiento que falla, e ILM aún no está en funcionamiento, el resultado es una pérdida irrecuperable de los datos. Las solicitudes que atraviesan una gran cantidad de claves requieren un manejo especial. La solicitud puede devolver una respuesta truncada o una respuesta vacía para evitar que el tiempo de espera se agote. Para obtener un conjunto completo de resultados, tendrá que seguir haciendo solicitudes a la vez que actualiza el parámetro **`marker`** tal y como lo suele hacer, con un resultado truncado. Utilice siempre **`NextMarker`** si está presente. En ciertos casos, la implementación StorageGRID Webscale del S3 REST API proporciona un**`NextMarker`**, cuando el Amazon S3 REST API no lo haría, porque es un mejor marcador que la última clave devuelta.
GET Bucket acl	Si se proporcionan las necesarias credenciales de acceso para la cuenta, esta operación devuelve una respuesta positiva y el ID, DisplayName, y el Permiso del propietario del bucket, indicando que el propietario tiene pleno acceso al bucket.
GET Bucket location	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación devolverá la región del bucket. De forma predeterminada, se devuelve us- east-1 salvo que se defina una región utilizando el elemento LocationConstraint en la solicitud PUT Bucket.
Versiones GET Bucket Object	Con acceso READ sobre un bucket, esta operación con el subrecurso `versions` lista los metadatos de todas las versiones de los objetos contenidos en el bucket.
GET Bucket notificación	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación devolverá la configuración de notificación unida al bucket.
GET Bucket Policy	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación devuelve la política unida al bucket. Si desea obtener más información sobre el lenguaje de política utilizado en el sistema StorageGRID Webscale, consulte Políticas de acceso al grupo y bucket en la página 50.

GET Bucket duplicación	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación devolverá la configuración de duplicación unida al bucket.

Operación	Implementación
GET Bucket Control de versiones	Esta implementación utiliza el subrecurso `versioning` para devolver el estado del control de versiones de un bucket. Para recuperar el estado de control de versiones de un bucket, deberá ser el propietario del bucket. El estado del control de versiones devuelto indica si el bucket no ha sido versionado o si el bucket tiene la versión "habilitada" o "suspendida".
HEAD Bucket	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación determina si el bucket existe y si usted dispone del permiso para acceder a él.
PUT Bucket	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación creará un nuevo bucket. Al crear el bucket, se convertirá en su propietario. De forma predeterminada, los buckets se crean en la región us-east-1. Para especificar una región diferente utilice el elemento de solicitud LocationConstraint y especifique el nombre exacto de una región que se haya definido utilizando la Interfaz de gestión de StorageGRID Webscale o de la API de administración. Póngase en contacto con el administrador del sistema si no conoce el nombre de la región que debe usar. **Nota:** Se producirá un error si su solicitud PUT Bucket utiliza una región que no haya sido definida en StorageGRID Webscale.

Operación	Implementación
PUT Bucket notificación	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación configura las notificaciones para el bucket utilizando la configuración de notificación XML incluida en el cuerpo de la solicitud. **Atención:** StorageGRID Webscale 11.0 incluye la versión inicial de los servicios de plataforma. En la actualidad, la duplicación CloudMirror, las notificaciones y la integración de búsquedas solo resultan apropiadas para determinadas situaciones y cargas de trabajo. Tendrá que ponerse en contacto con su representante de NetApp si desea utilizar la versión inicial de estos servicios. Debe tener en cuenta los siguientes detalles de implementación: • StorageGRID Webscale es compatible con los temas del Servicio Simple de Notificación (SNS) como destinos. No es compatible con los endpoints Amazon Lambda o Simple Queue Service (SQS). • Se deben especificar los destinos de las notificaciones como el URN de un endpoint de StorageGRID Webscale. Los endpoints se pueden crear utilizando la Interfaz de Administración del tenant o la API de Administración del tenant. El endpoint debe existir para que la configuración de notificación tenga éxito. Si el endpoint no existe, se devolverá un error `400 Bad Request (Solicitud errónea)` con el código `InvalidArgument` (Argumento no válido). • No se pueden utilizar notificaciones sobre el evento `s3:ReducedRedundancyLostObject .` • Los mensajes de notificación de eventos usan valores estándar para la mayoría de las claves, a excepción de las siguientes: ◦ `eventSource` devuelve `sgws:s3` ◦ `awsRegion`: esta clave no se devuelve ◦ `x-amz-id-2`: esta clave no se devuelve ◦ `arn` devuelve `urn:sgws:s3:::bucket_name` Consulte la Guía del Administrador del tenant si desea obtener más información sobre cómo implementar notificaciones sobre los buckets S3.
PUT Bucket política	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación define la política unida al bucket. Si desea obtener más información sobre el lenguaje de política utilizado en el sistema StorageGRID Webscale, consulte Políticas de acceso al grupo y al bucket en la página 50.

Operación	Implementación
PUT Bucket duplicación	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación configura la duplicación CloudMirror de StorageGRID Webscale para el bucket utilizando la configuración de duplicación XML proporcionada en el cuerpo de la solicitud. **Atención:** StorageGRID Webscale 11.0 incluye la versión inicial de los servicios de plataforma. En la actualidad, la duplicación CloudMirror, las notificaciones y la integración de búsquedas solo resultan apropiadas para determinadas situaciones y cargas de trabajo. Tendrá que ponerse en contacto con su representante de NetApp si desea utilizar la versión inicial de estos servicios. Para la duplicación CloudMirror, debe tener en cuenta los siguientes detalles de implementación: • La duplicación del bucket se puede configurar en buckets con versión y sin versión. • Puede especificar otro bucket de destino en cada regla de la configuración de duplicación XML. Cada bucket se puede duplicar en más de un bucket de destino. • Los buckets de destino se deben especificar como el URN de los endpoints de StorageGRID Webscale tal y como se especifican en la Interfaz de Administración del tenant o en la API del Administrador del tenant. El endpoint debe existir para que la configuración de duplicación tenga éxito. Si el endpoint no existe, se devolverá un mensaje `404 Not Found (No encontrado)` con el código `NoSuchEndpoint` (No existe tal endpoint). • No necesita especificar un `Role` o una `StorageClass` en el código XML de configuración, ya que estos valores no son utilizados por StorageGRID Webscale y se ignorarán en el caso de que se envíen. StorageGRID Webscale no necesita un rol definido para almacenar objetos duplicados en el bucket de destino y utiliza, de forma predeterminada, la clase de almacenamiento `STANDARD` . Consulte la Guía del Administrador del tenant si desea obtener más información sobre el empleo de la duplicación de buckets para implementar la duplicación de CloudMirror de StorageGRID Webscale.

PUT Bucket Control de versiones	Esta implementación utiliza el subrecurso `versioning` para definir el estado del control de versiones de un bucket existente. Para ajustar el estado del control de versiones, deberá ser el propietario del bucket. Podrá ajustar el estado del control de versiones con uno de los valores siguientes: • Habilitado: habilita el control de versiones para los objetos contenidos en el bucket. Todos los objetos que se añaden al bucket reciben un ID de versión único. • Suspendido: deshabilita el control de versiones para los objetos contenidos en el bucket. Todos los objetos que se añaden al bucket reciben un ID de versión de valor `null`..

Información relacionada

Guía del Administrador del tenant de StorageGRID Webscale 11.0

Operaciones personalizadas sobre buckets

El sistema StorageGRID Webscale admite operaciones personalizadas que se agregan a la API REST de S3 y son específicas del sistema.

La siguiente tabla describe las operaciones personalizadas sobre buckets que se encuentran implementadas, y la forma en que se implementan, en el sistema StorageGRID Webscale.

Operación	Implementación
Solicitud de coherencia de GET Bucket	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación proporciona el nivel de coherencia que se aplica a un determinado bucket. Para obtener más información, consulte Solicitud de coherencia de GET Bucket en la página 36.
Solicitud de coherencia de PUT Bucket 37	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación define el nivel de coherencia que se aplica a un determinado bucket. Para obtener más información, consulte Solicitud de coherencia de PUT Bucket en la página 37.
Solicitud de la hora del último acceso a GET Bucket	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación proporciona información sobre si se encuentra habilitada o deshabilitada la actualización de la última hora de acceso para un determinado bucket. Para obtener más información, consulte Solicitud de la hora del último acceso a GET Bucket . en la página 40.
Solicitud de la hora del último acceso a PUT Bucket	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación le permitirá habilitar o deshabilitar las actualizaciones de la última hora de acceso para un determinado bucket. De forma predeterminada, la última hora de acceso se encuentra deshabilitada para todos los buckets creados con la versión 10.3.0, o posterior. Para obtener más información, consulte Solicitud de la hora del último acceso a PUT Bucket . en la página 41.
DELETE Bucket, configuración de notificación de metadatos	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación elimina la configuración XML de notificación de metadatos asociada con un determinado bucket. Para obtener más información, consulte DELETE Bucket solicitud de la configuración de notificación de metadatos en la página 42.
Configuración de la notificación de metadatos GET Bucket	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación proporciona la configuración XML de notificación de metadatos asociada con un determinado bucket. Para obtener más información, consulte GET Bucket solicitud de la configuración de notificación de metadatos en la página 42.

Operación	Implementación
PUT Bucket, configuración de la notificación de metadatos	Si se proporcionan las credenciales de acceso necesarias para la cuenta, esta operación configura el servicio de notificación de metadatos para un bucket utilizando la configuración XML incluida en el cuerpo de la solicitud. Consulte la Guía del Administrador del tenant para obtener más información sobre cómo implementar el servicio de integración de búsquedas. Para obtener más información sobre la API de configuración de notificación de metadatos de PUT, consulte PUT Bucket, solicitud de configuración de la notificación de metadatos en la página 45.

Información relacionada

Guía del Administrador del tenant de StorageGRID Webscale 11.0

Operaciones sobre objetos

No se puede acceder mediante S3 a los objetos de datos ingeridos en el sistema StorageGRID Webscale a través de Swift.

Todas las operaciones sobre objetos, salvo GET Object ACL y OPTIONS /, permiten el empleo de los controles de coherencia de StorageGRID Webscale. Para obtener más información sobre cómo utilizar la cabecera `Consistency-Control`, consulte Cómo StorageGRID Webscale implementa la S3 REST API en la página 11.

Las solicitudes conflictivas de los clientes, como las de dos clientes que escriben la misma clave, se resuelven según la regla "la última gana". La temporización de la evaluación "la última gana" se basa en el momento en que el sistema StorageGRID Webscale completa una solicitud determinada, y no cuándo los clientes S3 comienzan una operación.

Todos los objetos contenidos en un bucket de StorageGRID Webscale pertenecen al propietario del bucket, incluyendo los objetos creados por un usuario anónimo, o por otra cuenta.

El sistema StorageGRID Webscale permite el empleo de las siguientes operaciones sobre los objetos:

Operación	Implementación

DELETE Objeto	No se permite el empleo de la Autenticación Multi-Factor (MFA) y la cabecera de respuesta `x-amz- mfa`. **Control de versiones** Para eliminar una versión específica, el solicitante debe ser el propietario del bucket y utilizar el subrecurso `versionId`. El uso de este subrecurso elimina permanentemente la versión. Si `versionId` se corresponde con un marcador de borrado, se asignará a la cabecera de respuesta `x-amz-delete-marker` el valor `true`. • Si se elimina un objeto sin el subrecurso `versionId` sobre un bucket habilitado con versión, provocará la generación de un marcador de borrado. Se proporcionará el `versionId` para el marcador de borrado utilizando la cabecera de respuesta `x-amz-version-id`, además se proporcionará la cabecera de respuesta `x-amz-delete- marker` con el valor `true`. • Si se elimina un objeto sin el subrecurso `versionId` sobre un bucket sin versión, se provocará un borrado permanente de una versión 'null' ya existente o un marcador de borrado 'null', y la generación de un nuevo marcador de borrado 'null'. A la cabecera de respuesta `x-amz-delete- marker` se le asignará el valor `true`.

Operación	Implementación
DELETE Varios objetos	No se permite el empleo de la Autenticación Multi-Factor (MFA) y la cabecera de respuesta `x-amz- mfa`. Se pueden eliminar varios objetos en el mismo mensaje de solicitud. **Nota:** La solicitud DELETE Varios objetos no se puede utilizar en buckets versionados. A diferencia de lo que sucede con la operación PUT Objeto, la operación DELETE Varios objetos no permite el empleo de la codificación de transferencia troceada ni de los atributos gzip de codificación de contenidos.
DELETE Etiquetado de objeto	Utiliza el subrecurso `tagging` para eliminar todas las etiquetas de un objeto. Implementada con todo el comportamiento de Amazon S3 REST API. **Control de versiones** Si en la solicitud no se especifica el parámetro de consulta `versionId`, la operación elimina todas las etiquetas desde la versión más reciente del objeto en un bucket versionado. Si la versión actual del objeto es un marcador de borrado, se proporcionará un estado "MethodNotAllowed" (Método no permitido) asignando a la cabecera de respuesta `x-amz- delete-marker` el valor `true`.
GET Objeto	Las siguientes cabeceras de solicitud no son compatibles y devuelven `XNotImplemented`: • `x-amz-restore` • `x-amz-website-redirect-location` **Control de versiones** Si no se especifica un subrecurso `versionId`, la operación proporciona la versión más reciente del objeto en un bucket versionado. Si la versión actual del objeto es un marcador de borrado, se proporcionará un estado "Not Found" (No encontrado) asignando a la cabecera de respuesta `x-amz- delete-marker` el valor `true`.
GET Objeto ACL	Si se proporcionan las necesarias credenciales de acceso para la cuenta, esta operación proporciona una respuesta positiva y el ID DisplayName, y el Permiso del propietario del bucket, indicando que el propietario tiene pleno acceso al objeto.
GET Etiquetado de Objeto	Utiliza el subrecurso `tagging` para proporcionar todas las etiquetas de un objeto. Implementada con todo el comportamiento de Amazon S3 REST API. **Control de versiones** Si en la solicitud no se especifica el parámetro de consulta `versionId`, la operación proporciona todas las etiquetas desde la versión más reciente del objeto en un bucket versionado. Si la versión actual del objeto es un marcador de borrado, se proporcionará un estado "MethodNotAllowed" (Método no permitido) asignando a la cabecera de respuesta `x-amz- delete-marker` el valor `true`.

Operación	Implementación
HEAD Objeto	Las siguientes cabeceras de solicitud no son compatibles y devuelven `XNotImplemented`: • `x-amz-restore` • `x-amz-website-redirect-location` **Control de versiones** Si no se especifica un subrecurso `versionId` , la operación proporciona la versión más reciente del objeto en un bucket versionado. Si la versión actual del objeto es un marcador de borrado, se proporcionará un estado “Not Found” (No encontrado) asignando a la cabecera de respuesta `x-amz- delete-marker` el valor `true`.
PUT Objeto	**Resolución de conflictos** Las solicitudes conflictivas de los clientes, como las de dos clientes que escriben la misma clave, se resuelven según la regla "la última gana". La temporización de la evaluación "la última gana" se basa en el momento en que el sistema StorageGRID Webscale completa una solicitud determinada, y no cuándo los clientes S3 comienzan una operación. **Tamaño del objeto** StorageGRID Webscale permite el empleo de objetos de hasta 5 TB de tamaño. **Propiedad de objetos** En StorageGRID Webscale, todos los objetos son propiedad de la cuenta del propietario del bucket, incluidos los objetos creados por una cuenta que no sea del propietario o de un usuario anónimo.
PUT Objeto (continuación)	**Opciones de la clase de almacenamiento** La cabecera de solicitud `x-amz-storage-class` es compatible con los siguientes valores enumerados: • `STANDARD`: (Valor predeterminado) Especifica una operación de ingesta de compromiso doble (dual-commit). • `REDUCED_REDUNDANCY`: Especifica una operación de ingesta de compromiso único (single-commit). **Nota:** La clase de almacenamiento `REDUCED_REDUNDANCY` (Redundancia reducida) es una opción utilizada para limitar el almacenamiento redundante de datos que se podrá duplicar mejor en cualquier otro lugar, tal como sucede con las políticas ILM. Por ello, especificar el valor `REDUCED_REDUNDANCY` no afecta a la política ILM especificada, y no provoca que los datos se almacenen con niveles inferiores de redundancia en el sistema StorageGRID Webscale. **Atención:** Tenga cuidado cuando ingiera objetos utilizando REDUCED_REDUNDANCY para crear una copia inicial única de los datos del objeto. Si la copia única se crea en un Nodo de Almacenamiento que falla, y el ILM aún no está en funcionamiento, el resultado es una pérdida irrecuperable de los datos.

Operación	Implementación
PUT Objeto (continuación)	**Cabeceras de solicitud** Se permite el empleo de las siguientes cabeceras de solicitud: • `x-amz-tagging` • `x-amz-server-side-encryption` • `x-amz-meta-` pares nombre-valor para metadatos definidos por el usuario Para registrar el tiempo de creación del objeto, con el fin de que pueda utilizar la opción User Defined Creation Time (Hora de creación definida por el usuario) para la hora de referencia en una regla ILM, tendrá que almacenar el valor en una cabecera definida por el usuario denominada `x- amz-meta-creation-time`. Por ejemplo: `x-amz-meta- creation-time`=1443399726. El valor de este campo se evalúa en segundos desde el 1 de enero de 1970. Si desea obtener más información, consulte "Hora de referencia" en la Guía del Administrador. Se permite el empleo de las siguientes cabeceras de solicitud solo con los siguientes valores: • `Transfer-Encoding:chunked` • `Content-Encoding:aws-chunked` **Nota:** Enviar otros valores a `Content-Encoding` puede provocar resultados inesperados, o fallos debido a esquemas de codificación no reconocidos o fallos de verificación MD5. Las siguientes cabeceras de solicitud no son compatibles: • `Expires` • `x-amz-acl` Las siguientes cabeceras de solicitud no son compatibles y devuelven `XNotImplemented`: • `x-amz-server-side-encryption-customer-algorithm` • `x-amz-server-side-encryption-customer-key` • `x-amz-server-side-encryption-customer-key-MD5` • `x-amz-website-redirect-location`
PUT Objeto (continuación)	**Control de versiones** Si el control de versiones se encuentra habilitado para un bucket, se generará automáticamente un `versionId` único para la versión del objeto que se está almacenando. Este `versionId` también se proporciona en la respuesta utilizando la cabecera de respuesta `x-amz- version-id`. Si se suspende el control de versiones, la versión del objeto se almacena con un `versionID` de valor null y en el caso de que ya exista una versión null se sobrescribirá. Si desea obtener más información sobre el control de versiones, consulte los apartados "PUT Bucket Control de versiones" y "GET Bucket Control de versiones" en Operaciones sobre buckets en la página 18.

Operación	Implementación
PUT Objeto - Copia	**Resolución de conflictos** Las solicitudes conflictivas de los clientes, como las de dos clientes que escriben la misma clave, se resuelven según la regla "la última gana". La temporización de la evaluación "la última gana" se basa en el momento en que el sistema StorageGRID Webscale completa una solicitud determinada, y no cuándo los clientes S3 comienzan una operación. **Cabeceras de solicitud** Se permite el empleo de las siguientes cabeceras de solicitud: • `x-amz-meta-` pares nombre-valor para metadatos definidos por el usuario Para registrar el tiempo de creación del objeto, con el fin de que pueda utilizar la opción User Defined Creation Time (Hora de creación definida por el usuario) para la hora de referencia en una regla ILM, tendrá que almacenar el valor en una cabecera definida por el usuario denominada `x- amz-meta-creation-time`. Por ejemplo: `x-amz-meta- creation-time`=1443399726. El valor de este campo se evalúa en segundos desde el 1 de enero de 1970. Si desea obtener más información, consulte “Hora de referencia” en la Guía del Administrador. • `x-amz-metadata-directive`: El valor predeterminado es `COPY`, que le permitirá copiar el objeto y los metadatos asociados. Puede especificar `REPLACE` para sobreescribir los metadatos existentes cuando copie el objeto, o para actualizar los metadatos del objeto. • `x-amz-copy-source` • `x-amz-copy-source-if-match` • `x-amz-copy-source-if-none-match` • `x-amz-copy-source-if-unmodified-since` • `x-amz-copy-source-if-modified-since` • `x-amz-server-side-encryption` • `x-amz-storage-class` • `x-amz-tagging-directive`: El valor predeterminado es `COPY`, que le permitirá copiar el objeto y todas las etiquetas. Puede especificar `REPLACE` para sobrescribir las etiquetas existentes cuando copie el objeto, o para actualizar las etiquetas.

Operación	Implementación
PUT Objeto - Copia (continuación)	Las siguientes cabeceras de solicitud no son compatibles y proporcionan `XNotImplemented`: • `x-amz-server-side-encryption-customer-algorithm` • `x-amz-server-side-encryption-customer-key` • `x-amz-server-side-encryption-customer-key-MD5` • `x-amz-website-redirect-location` Si la clave y el bucket origen, especificados en la cabecera `x-amz-copy-source`, son distintos de la clave y del bucket destino, se escribirá una copia del objeto origen en el destino. Si el origen y el destino coinciden, y se especifica la cabecera `x-amz-metadata-directive` como `REPLACE`, se actualizarán los metadatos del objeto con los valores de los metadatos suministrados en la petición. **Nota:** No se puede actualizar el valor `server-side-encryption` del objeto. En su lugar, realice una copia utilizando un nuevo valor server-side-encryption (cifrado del lado del servidor) empleando `x-amz-metadata-directive`: `REPLACE`.
PUT Objeto - Copia (continuación)	**Control de versiones** Si se versiona el bucket origen, podrá usar la cabecera `x-amz-copy-source` para copiar la última versión de un objeto. Para copiar una versión específica de un objeto, deberá especificar explícitamente la versión a copiar utilizando el subrecurso `versionId`. Si se realiza una versión del bucket destino, la versión generada se proporcionará en la cabecera de respuesta `x-amz-version- id`. Si se suspende el control de versiones para el bucket destino, `x-amz-version-id` proporcionará un valor “null”.
PUT Etiquetado de Objeto	Utiliza el subrecurso `tagging` para añadir un grupo de etiquetas a un objeto existente. Implementada con todo el comportamiento de Amazon S3 REST API. **Resolución de conflictos** Las solicitudes conflictivas de los clientes, como las de dos clientes que escriben la misma clave, se resuelven según la regla "la última gana". La temporización de la evaluación "la última gana" se basa en el momento en que el sistema StorageGRID Webscale completa una solicitud determinada, y no cuándo los clientes S3 comienzan una operación. **Control de versiones** Si en la solicitud no se especifica el parámetro de consulta `versionId`, la operación agrega etiquetas a la versión más reciente del objeto en un bucket versionado. Si la versión actual del objeto es un marcador de borrado, se proporcionará un estado “MethodNotAllowed” (Método no permitido) asignando a la cabecera de respuesta `x-amz- delete-marker` el valor `true`.

Operaciones para cargas múltiples

Nota: No debe superar las 1000 cargas simultáneas de varias partes a un solo bucket porque los resultados de las consultas de Cargas de listas múltiples para ese bucket pueden proporcionar

resultados incompletos.

Nota: Cada elemento de una carga múltiple, salvo el último, debe estar comprendido entre 5 MB y 5 GB. El último elemento debe ser inferior a 5 MB. Este cliente debe cumplir estos límites, ya que no viene impuesto por StorageGRID Webscale.

Todas las operaciones de carga múltiple permiten el empleo de los controles de coherencia StorageGRID Webscale. Para obtener más información sobre cómo utilizar la cabecera `Consistency-Control`, consulte Cómo StorageGRID Webscale implementa la S3 REST API en la página 11.

Operación	Implementación
Lista de cargas múltiples	Se permite el empleo de las siguientes cabeceras de solicitud: • `encoding-type` • `max-uploads` • `key-marker` • `prefix` • `upload-id-marker` No se puede emplear el parámetro de solicitud `delimiter`. **Control de versiones** La carga múltiple consta de operaciones independientes para iniciar la carga, listar cargas, subir partes, ensamblar las partes cargadas y completar la transferencia. Una vez realizada la operación Carga múltiple, ese es el momento en que se crean los objetos (y las versiones, si corresponde).

Operación	Implementación

Iniciar Carga Múltiple	La cabecera de solicitud `x-amz-storage-class` es compatible con los siguientes valores enumerados: • `STANDARD` Valor predeterminado. Especifica una operación de ingesta de compromiso doble (dual-commit). • `REDUCED_REDUNDANCY:` Especifica una operación de ingesta de compromiso único (single-commit). **Nota:** La clase de almacenamiento `REDUCED_REDUNDANCY` (Redundancia reducida) es una opción utilizada para limitar el almacenamiento redundante de datos que se podrá duplicar mejor en cualquier otro lugar, por ejemplo, empleando las políticas ILM. Por lo tanto, especificar el valor `REDUCED_REDUNDANCY` no afecta a la política ILM especificada, y no provoca que los datos se almacenen con niveles inferiores de redundancia en el sistema StorageGRID Webscale. **Atención:** Tenga cuidado cuando ingiera objetos utilizando REDUCED_REDUNDANCY para crear una copia inicial única de los datos del objeto. Si la copia única se crea en un Nodo de Almacenamiento que falla, e ILM aún no está en funcionamiento, el resultado es una pérdida irrecuperable de los datos. Se permite el empleo de las siguientes cabeceras de solicitud: • `Content-Type (Tipo de contenido)` • `x-amz-meta-` pares nombre-valor para metadatos definidos por el usuario Para registrar el tiempo de creación del objeto, con el fin de que pueda utilizar la opción User Defined Creation Time (Hora de creación definida por el usuario) para la hora de referencia en una regla ILM, tendrá que almacenar el valor en una cabecera definida por el usuario denominada `x- amz-meta-creation-time`. Por ejemplo: `x-amz-meta- creation-time`=1443399726. El valor de este campo se evalúa en segundos desde el 1 de enero de 1970. Si desea obtener más información, consulte “Hora de referencia” en la Guía del Administrador. La cabecera `x-amz-server-side-encryption` no se puede utilizar directamente para las solicitudes Initiate Multipart Upload (Iniciar Cargas Múltiples). Si necesita cifrado del lado del servidor para una carga múltiple, tendrá que especificar la cabecera `x-amz-server-side-encryption` para cada una de las partes cargadas, pero no puede especificarla como parte de la cabecera Initiate Multipart Upload o fallará la solicitud. Las siguientes cabeceras de solicitud no son compatibles y proporcionan `XNotImplemented`: • `x-amz-website-redirect-location` • `x-amz-server-side-encryption-customer-key` **Control de versiones** La carga múltiple consta de operaciones independientes para iniciar la carga, listar cargas, subir partes, ensamblar las partes cargadas y completar la transferencia. Una vez realizada la operación Carga múltiple completada, ese es el momento en que se crean los objetos (y las versiones, si corresponde).

Operación	Implementación
Cargar Parte	Se permite el empleo de las siguientes cabeceras de solicitud: • `Content-Length (Longitud del contenido)` • `x-amz-server-side-encryption` Si necesita especificar un cifrado del lado del servidor para una carga múltiple, tendrá que especificar la cabecera `x-amz-server-side- encryption` para cada una de las partes individuales a transferir. Las siguientes cabeceras de solicitud no son compatibles y proporcionan `XNotImplemented`: • `x-amz-server-side-encryption-customer-algorithm` • `x-amz-server-side-encryption-customer-key` • `x-amz-server-side-encryption-customer-key-MD5` **Control de versiones** La carga múltiple consta de operaciones independientes para iniciar la carga, listar cargas, subir partes, ensamblar las partes cargadas y completar la transferencia. Una vez realizada la operación Carga múltiple, ese es el momento en que se crean los objetos (y las versiones, si corresponde).
Cargar Parte - Copiar	Implementada con todo el comportamiento de Amazon S3 REST API. Esta solicitud lee y escribe los datos de objeto especificados en `x-amz-copy-source-range` en el sistema StorageGRID Webscale . Se permite el empleo de las siguientes cabeceras de solicitud: • `x-amz-copy-source-if-match` • `x-amz-copy-source-if-none-match` • `x-amz-copy-source-if-unmodified-since` • `x-amz-copy-source-if-modified-since` **Control de versiones** La carga múltiple consta de operaciones independientes para iniciar la carga, listar cargas, subir partes, ensamblar las partes cargadas y completar la transferencia. Una vez realizada la operación Carga múltiple, ese es el momento en que se crean los objetos (y las versiones, si corresponde).

Operación	Implementación
Completar Carga Múltiple	**Resolución de conflictos** Las solicitudes conflictivas de los clientes, como las de dos clientes que escriben la misma clave, se resuelven según la regla "la última gana". La temporización de la evaluación "la última gana" se basa en el momento en que el sistema StorageGRID Webscale completa una solicitud determinada, y no cuándo los clientes S3 comienzan una operación. **Cabeceras de solicitud** La cabecera de solicitud `x-amz-storage-class` es compatible con los siguientes valores enumerados: • `STANDARD` Valor predeterminado. Especifica una operación de ingesta de compromiso doble (dual-commit). • `REDUCED_REDUNDANCY` Especifica una operación de ingesta de compromiso único (single-commit). **Nota:** La clase de almacenamiento `REDUCED_REDUNDANCY` (Redundancia reducida) es una opción utilizada para limitar el almacenamiento redundante de datos que se podrá duplicar mejor en cualquier otro lugar, tal como sucede con las políticas ILM. Por ello, especificar el valor `REDUCED_REDUNDANCY` no afecta a la política ILM especificada, y no provoca que los datos se almacenen con niveles inferiores de redundancia en el sistema StorageGRID Webscale. **Atención:** Tenga cuidado cuando configure las políticas ILM para los datos configurados con `REDUCED_REDUNDANCY` para duplicar una copia simple de datos de objeto antes de satisfacer la política ILM. Si la copia única se crea en un Nodo de Almacenamiento que falla, y la política ILM aún no está en funcionamiento, el resultado es una pérdida irrecuperable de los datos. **Atención:** Si una carga múltiple no se completa en 15 días, la operación se marcará como inactiva y se eliminarán del sistema todos los datos asociados. **Nota:** El valor `ETag` proporcionado no es una suma MD5 de los datos, si no que sigue la implementación Amazon S3 API del valor `ETag` para objetos múltiples.

Operación	Implementación
Completar Carga Múltiple (continuación)	**Control de versiones** Esta operación completa una carga múltiple Si el control de versiones está habilitado para un bucket, se creará la versión del objeto tras la finalización de la carga múltiple. Si el control de versiones se encuentra habilitado para un bucket, se generará automáticamente un `versionId` único para la versión del objeto que se está almacenando. Este `versionId` también se proporciona en la respuesta utilizando la cabecera de respuesta `x-amz- version-id`. Si se suspende el control de versiones, la versión del objeto se almacena con un `versionID` de valor null y en el caso de que ya exista una versión null se sobreescribirá. **Nota:** Si el control de versiones se encuentra habilitado para un bucket, al completar una carga múltiple siempre se crea una nueva versión, incluso si existen cargas múltiples simultáneas que se completan sobre la misma clave de objeto. Cuando el bucket no tiene habilitado el control de versiones, es posible iniciar una carga múltiple y, posteriormente, iniciar y completar otra carga múltiple sobre la misma clave de objeto. En los buckets sin control de versiones, tendrá prioridad la carga múltiple que se completa en último lugar. **Duplicación, notificación o notificación de metadatos fallida.** Si el bucket donde se produce la carga múltiple se configura para un servicio de plataforma, la carga múltiple se realizará correctamente incluso aunque falle la acción de duplicación o de notificación. Si este hecho se produjera, se provocará una alarma en la Interfaz de gestión sobre Eventos Totales (SMTT). El mensaje Last Event (Último Evento) en **Grid** > ***site*** > ***Storage Node*** > **SSM** > **Events** muestra el mensaje " Failed to publish notifications for *`bucket-name object key`* " (Fallo en la publicación de notificaciones para nombre de bucket - clave del objeto) para el último objeto para el cual ha fallado la notificación. Los mensajes de los eventos también se enumeran en `/var/local/log/ bycasterr.log` Un tenant puede activar la duplicación o la notificación fallida actualizando los metadatos o etiquetas del objeto. Puede volver a enviar los valores existentes para evitar realizar cambios no deseados.
Abortar Cargas Múltiples	Implementada con todo el comportamiento de Amazon S3 REST API.
Listar Partes	Implementada con todo el comportamiento de Amazon S3 REST API.

Información relacionada

Guía del administrador de StorageGRID Webscale 11.0

Operaciones rastreadas en los registros de auditoría

Se realizan seguimientos de varias operaciones sobre buckets y operaciones sobre objetos en los registros de auditoría de StorageGRID Webscale

Operaciones sobre buckets rastreadas en los registros de auditoría
DELETE Bucket
DELETE Varios Objetos
GET Bucket (Lista de objetos)

Operaciones sobre buckets rastreadas en los registros de auditoría
GET Bucket Versiones de objeto
HEAD Bucket
PUT Bucket
PUT Bucket Control de versiones

Operaciones sobre objetos rastreadas en los registros de auditoría
Completar Carga Múltiple
DELETE Objeto
GET Objeto
HEAD Objeto
PUT Objeto
PUT Objeto - Copia

Operaciones de la API StorageGRID Webscale S3 REST

Existen operaciones añadidas a la API REST de S3 que son específicas del sistema StorageGRID Webscale.

Solicitud de coherencia de GET Bucket

La Solicitud de coherencia de GET Bucket le permitirá determinar el nivel de coherencia aplicado a un determinado bucket.

Los controles de coherencia predeterminados están configurados para garantizar la lectura-tras-la escritura para los objetos recién creados. Para obtener más información sobre la cabecera `Consistency-Control`, consulte la información sobre Cómo StorageGRID Webscale implementa la S3 REST API.

Debe disponer del permiso S3: GetBucketConsistency para poder completar esta operación.

Solicitud

Cabecera HTTP de la solicitud	Descripción
Authorization (Autorización)	Especifica la firma AWS y la ID de la clave de acceso para la cuenta que se usará para la solicitud.
Date (Fecha)	La fecha y hora de la solicitud.
Host	El nombre del host al que se dirige la solicitud.

Ejemplo de solicitud

```
GET /bucket?x-ntap-sg-consistency HTTP/1.1
Date: Sat, 29 Nov 2015 01:02:17 GMT
Authorization: AWS 9MOYPG9ACWPAJA1S72R5:jUGbYkLdBApjCWBgK4TxvOjfock=
Host: test.com
```

Respuesta

Cabecera HTTP de la	Descripción
Connection (Conexión)	Especifica si la conexión al servidor está abierta o cerrada.
Content-Length (Longitud del contenido)	La longitud del cuerpo de la respuesta.
Content-Type (Tipo de contenido)	El tipo Multipurpose Internet Mail Extensions (MIME-Extensiones multipropósito del correo de internet) del cuerpo de la respuesta.
Date (Fecha)	La fecha y hora de la respuesta.
Server (Servidor)	El servidor que crea la respuesta.
x-amz-request-id	El identificador que identifica de manera única la solicitud. Creada por la S3 API.

En la respuesta XML, `<Consistency>` proporcionará uno de los siguientes valores:

- **all** (todo): Proporciona la mayor garantía de la coherencia "lectura tras la escritura". Todos los nodos reciben los datos inmediatamente o la solicitud fallará.

- **strong-global** (fuerte-global): Garantiza la coherencia "lectura tras la escritura" para todas las solicitudes de los clientes en todas las sedes.
- **strong-site** (fuerte-sede): Garantiza la coherencia "lectura tras la escritura" para todas las solicitudes de los clientes dentro de una sede.
- **default** (predeterminado) ("lectura tras la escritura" para nuevo objeto): Proporciona coherencia "lectura tras la escritura" para nuevos objetos y coherencia final para actualizaciones de objetos. Ofrece alta disponibilidad y garantías de protección de datos. Coincide con las garantías de coherencia de AWS S3.

  Nota: Si su aplicación intenta operaciones HEAD sobre claves que no existen, ajuste el nivel de coherencia a **available** (disponible) salvo que requiera garantías de coherencia de AWS S3. De lo contrario, se pueden producir un elevado número de errores del tipo "500 Internal Server error" (error interno del servidor) si uno o más nodos de almacenamiento no se encuentran disponibles.

- **available** (disponible) (coherencia final para operaciones HEAD): Se comporta igual que el nivel de coherencia **default** (predeterminado), pero solo proporciona una coherencia final para las operaciones HEAD. Ofrece una disponibilidad superior para las operaciones HEAD que **default** para el caso de que los Nodos de Almacenamiento no estén disponibles. Se distingue de las garantías de coherencia de AWS S3 solo para las operaciones HEAD.
- **weak** (débil): Proporciona coherencia final y alta disponibilidad, con garantías mínimas de protección de datos, especialmente si falla un Nodo de Almacenamiento o no está disponible. Adecuado solo para cargas de trabajo pesadas en escritura que requieran alta disponibilidad, no requieran coherencia de "lectura tras escritura" y pueden tolerar la posible pérdida de datos si falla un nodo.

Ejemplo de respuesta

```
HTTP/1.1 200 OK
Date: Sat, 29 Nov 2015 01:02:18 GMT
Connection: CLOSE
Server: StorageGRID/10.3.0
x-amz-request-id: 12345
Content-Length: 127
Content-Type: application/xml
<?xml version="1.0" encoding="UTF-8"?>
<Consistency xmlns="http://s3.storagegrid.com/doc/2015-02-01/">default</
Consistency>
```

Conceptos relacionados

Cómo StorageGRID Webscale implementa la S3 REST API en la página 11

Solicitud de coherencia de PUT Bucket

La solicitud de coherencia de PUT Bucket le permitirá especificar el nivel de coherencia para aplicar a las operaciones realizadas sobre un bucket.

Los controles de coherencia predeterminados están configurados para garantizar la lectura-tras-la escritura para los objetos recién creados. Para obtener más información sobre la cabecera `Consistency-Control` , consulte la información sobre Cómo StorageGRID Webscale implementa la S3 REST API.

Debe disponer del permiso s3:PutBucketConsistency para poder completar esta operación.

Solicitud

Cabecera HTTP de la solicitud	Descripción

Authorization (Autorización)	Especifica la firma AWS y la ID de la clave de acceso para la cuenta que se usará para la solicitud.
Date (Fecha)	La fecha y hora de la solicitud.

Cabecera HTTP de la solicitud	Descripción
Host	El nombre del host al que se dirige la solicitud.

El parámetro `x-ntap-sg-consistency` debe contener uno de los siguientes valores:

- **all** (todo): Proporciona la mayor garantía de la coherencia "lectura tras la escritura". Todos los nodos reciben los datos inmediatamente o la solicitud fallará.
- **strong-global** (fuerte-global): Garantiza la coherencia "lectura tras la escritura" para todas las solicitudes de los clientes en todas las sedes.
- **strong-site** (fuerte-sede): Garantiza la coherencia "lectura tras la escritura" para todas las solicitudes de los clientes dentro de una sede.
- **default** (predeterminado) ("lectura tras la escritura" para nuevo): Proporciona coherencia "lectura tras la escritura" para nuevos objetos y coherencia final para actualizaciones de objetos. Ofrece alta disponibilidad y garantías de protección de datos. Coincide con las garantías de coherencia de AWS S3.

  Nota: Si su aplicación intenta operaciones HEAD sobre claves que no existen, ajuste el nivel de coherencia a **available** (disponible) salvo que requiera garantías de coherencia de AWS S3. De lo contrario, se pueden producir un elevado número de errores del tipo "500 Internal Server error" (error interno del servidor) si uno o más nodos de almacenamiento no se encuentran disponibles.

- **available** (disponible) (coherencia final para operaciones HEAD): Se comporta igual que el nivel de coherencia **default** (predeterminado), pero solo proporciona una coherencia final para las operaciones HEAD. Ofrece una disponibilidad superior para las operaciones HEAD que **default** para el caso de que los Nodos de Almacenamiento no estén disponibles. Se distingue de las garantías de coherencia de AWS S3 solo para las operaciones HEAD.
- **weak** (débil): Proporciona coherencia final y alta disponibilidad, con garantías mínimas de protección de datos, especialmente si falla un Nodo de Almacenamiento o no está disponible. Adecuado solo para cargas de trabajo pesadas en escritura que requieran alta disponibilidad, no requieran coherencia de "lectura tras escritura" y pueden tolerar la posible pérdida de datos si falla un nodo.

Nota: En general, podrá usar el valor de control de coherencia **default** (predeterminado). Cuando las solicitudes no funcionen correctamente, si es posible modifique el comportamiento del cliente de la aplicación. O bien, configure el cliente para que especifique el control de coherencia para cada solicitud de API. Defina el control de coherencia a nivel bucket solo como último recurso.

Ejemplo de solicitud

```
PUT /bucket?x-ntap-sg-consistency=strong-global HTTP/1.1
Date: Sat, 29 Nov 2015 01:02:17 GMT
Authorization: AWS 9MOYPG9ACWPAJA1S72R5:jUGbYkLdBApjCWBgK4TxvOjfock=
Host: test.com
```

Conceptos relacionados

Cómo StorageGRID Webscale implementa la S3 REST API en la página 11

GET, Solicitud de Empleo de Almacenamiento

La solicitud GET Empleo de Almacenamiento le indica la cantidad total de almacenamiento que una cuenta está utilizando, y para cada uno de los buckets asociados con la cuenta.

La cantidad de almacenamiento empleada por una cuenta y sus buckets se puede obtener mediante

una solicitud GET Servicio modificada utilizando el parámetro de consulta `x-ntap-sg-usage` . El emplo del almacenamiento del bucket se supervisa de forma separada mediante solicitudes PUT y DELETE procesadas por el sistema. Puede haber algún retraso antes de que los valores de empleo coincidan con los valores esperados en función del procesamiento de las solicitudes, en particular si el sistema está sometido a una pesada carga de trabajo.

Debe disponer del permiso s3:ListAllMyBuckets para poder completar esta operación.

Solicitud

Cabecera HTTP de la solicitud	Descripción
Authorization (Autorización)	Especifica la firma AWS y la ID de la clave de acceso para la cuenta que se usará para la solicitud.
Date (Fecha)	La fecha y hora de la solicitud.
Host	El nombre del host al que se dirige la solicitud.

Ejemplo de solicitud

```
GET /?x-ntap-sg-usage HTTP/1.1
Date: Sat, 29 Nov 2015 00:49:04 GMTT
Authorization: AWS 9MOYPG9ACWPAJA1S72R5:jUGbYkLdBApjCWBgK4TxvOjfock=
Host: test.com
```

Respuesta

Cabecera HTTP de la	Descripción
Connection (Conexión)	Especifica si la conexión al servidor está abierta o cerrada.
Content-Length (Longitud del contenido)	La longitud del cuerpo de la respuesta.
Content-Type (Tipo de contenido)	El tipo Multipurpose Internet Mail Extensions (MIME-Extensiones multipropósito del correo de internet) del cuerpo de la respuesta.
Date (Fecha)	La fecha y hora de la respuesta.
Server (Servidor)	El servidor que crea la respuesta.
x-amz-request-id	El identificador que identifica de manera única la solicitud. Creada por la S3 API.

Ejemplo de respuesta

```
HTTP/1.1 200 OK
Date: Sat, 29 Nov 2015 00:49:05 GMT
Connection: KEEP-ALIVE
Server: StorageGRID/10.2.0
x-amz-request-id: 727237123
Content-Length: 427
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<UsageResult xmlns="http://s3.storagegrid.com/doc/2015-02-01">
<CalculationTime>2014-11-19T05:30:11.000000Z</CalculationTime>
<ObjectCount>4</ObjectCount>
<DataBytes>12</DataBytes>
<Buckets>
<Bucket>
<Name>bucket1</Name>
<ObjectCount>2</ObjectCount>
<DataBytes>6</DataBytes>
</Bucket>
```

```
<Bucket>
<Name>bucket2</Name>
<ObjectCount>2</ObjectCount>
<DataBytes>6</DataBytes>
</Bucket>
</Buckets>
</UsageResult>
```

Control de versiones

Cada versión almacenada del objeto contribuirá a los valores `ObjectCount` y `DataBytes` en la respuesta. Los marcadores de borrado no se añadirán al `ObjectCount` total.

Solicitud de la hora del último acceso a GET Bucket

La Solicitud de la hora del último acceso a GET Bucket le permitirá determinar si la actualización de la hora del último acceso se encuentra habilitada o deshabilitada para cada bucket.

Debe disponer del permiso s3:GetBucketLastAccessTime para poder completar esta operación.

Solicitud

Cabecera HTTP de la solicitud	Descripción
Authorization (Autorización)	Especifica la firma AWS y la ID de la clave de acceso para la cuenta que se usará para la solicitud.
Date (Fecha)	La fecha y hora de la solicitud.
Host	El nombre del host al que se dirige la solicitud.

Ejemplo de solicitud

```
GET /bucket?x-ntap-sg-lastaccesstime HTTP/1.1
Date: Sat, 29 Nov 2015 01:02:17 GMT
Authorization: AWS 9MOYPG9ACWPAJA1S72R5:jUGbYkLdBApjCWBgK4TxvOjfock=
Host: test.com
```

Respuesta

Cabecera HTTP de la	Descripción
Connection (Conexión)	Especifica si la conexión al servidor está abierta o cerrada.
Content-Length (Longitud del	La longitud del cuerpo de la respuesta.
Content-Type (Tipo de contenido)	El tipo Multipurpose Internet Mail Extensions (MIME-Extensiones multipropósito del correo de internet) del cuerpo de la respuesta.
Date (Fecha)	La fecha y hora de la respuesta.
Server (Servidor)	El servidor que crea la respuesta.
x-amz-request-id	El identificador que identifica de manera única la solicitud. Creada por la S3 API.

Ejemplo de respuesta

El valor de `<LastAccessTime>` puede estar habilitado o deshabilitado.

```
HTTP/1.1 200 OK
Date: Sat, 29 Nov 2015 01:02:18 GMT
Connection: CLOSE
Server: StorageGRID/10.3.0
```

```
x-amz-request-id: 12345
Content-Length: 127
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<LastAccessTime xmlns="http://s3.storagegrid.com/doc/2015-02-01/">enabled
</LastAccessTime>
```

Solicitud de la hora del último acceso a PUT Bucket

La solicitud de la hora del último acceso mediante PUT Bucket le permitirá habilitar o deshabilitar las actualizaciones de la Última hora de acceso para cada bucket. Al deshabilitar las actualizaciones de la última hora de acceso se mejora el rendimiento, y es la configuración predeterminada para todos los buckets creados con la versión 10.3.0 o posterior.

Debe disponer del permiso s3:PutBucketLastAccessTime para un bucket para habilitar o deshabilitar las opciones de configuración de la última hora de acceso para dicho bucket.

> **Nota:** Comenzando por la versión 10.3 de StorageGRID Webscale, las actualizaciones de la última hora de acceso se encuentran deshabilitadas de forma predeterminada para todos los nuevos buckets. Si tiene buckets que se hayan creado con una versión anterior de StorageGRID Webscale y desea utilizar el nuevo comportamiento predeterminado, debe deshabilitar explícitamente las actualizaciones de la última hora de acceso para cada uno de esos buckets anteriores. Puede habilitar o deshabilitar las actualizaciones de la última hora de acceso utilizando la solicitud de la última hora de acceso de PUT Bucket, la casilla de verificación **S3 > Buckets > Change Last Access Setting** en la Interfaz de Administración del tenant o en la API de Administración del tenant.

Si las actualizaciones de la última hora de acceso están deshabilitadas para un bucket, se aplica el siguiente comportamiento a las operaciones sobre el bucket:

- Las solicitudes GET Objeto, GET Objeto ACL, GET Etiquetado de Objeto y HEAD Objeto no actualizan el valor de la última hora de acceso. El objeto no se agrega a las colas para la evaluación de la administración del ciclo de vida de la información (ILM).
- Las solicitudes PUT Copia de objeto y PUT Etiquetado de Objeto que actualizan solo los metadatos, también actualizan la última hora de acceso. El objeto se añade a las colas para la evaluación ILM.
- Si las actualizaciones de la última hora de acceso se encuentran deshabilitadas para el bucket origen, las solicitudes PUT Objeto Copia no actualizan la última hora de acceso para el bucket origen. El objeto copiado no se añade a las colas para la evaluación ILM para el bucket origen. Sin embargo, para el destino, las solicitudes Objeto Copia siempre actualizan la última hora de acceso. La copia del objeto se añade a las colas para la evaluación ILM.
- Las solicitudes Completar Carga Múltiple actualizan la última hora de acceso. El objeto completado se añade a las colas para la evaluación ILM.

Solicitud

Cabecera HTTP de la solicitud	Descripción
Authorization (Autorización)	Especifica la firma AWS y la ID de la clave de acceso para la cuenta que se usará para la solicitud.
Date (Fecha)	La fecha y hora de la solicitud.
Host	El nombre del host al que se dirige la solicitud.

Ejemplos de solicitud

Habilitación de la última hora de acceso para un bucket

```
PUT /bucket?x-ntap-sg-lastaccesstime=enabled HTTP/1.1
Date: Sat, 29 Nov 2015 01:02:17 GMT
```

```
Authorization: AWS 9MOYPG9ACWPAJA1S72R5:jUGbYkLdBApjCWBgK4TxvOjfock=
Host: test.com
```

Deshabilitación de la última hora de acceso para un bucket

```
PUT /bucket?x-ntap-sg-lastaccesstime=disabled HTTP/1.1
Date: Sat, 29 Nov 2015 01:02:17 GMT
Authorization: AWS 9MOYPG9ACWPAJA1S72R5:jUGbYkLdBApjCWBgK4TxvOjfock=
Host: test.com
```

Solicitud de configuración de notificación de metadatos DELETE Bucket

La solicitud de configuración de notificación de metadatos DELETE Bucket le permitirá deshabilitar el servicio de integración de búsqueda para cada bucket, eliminando el código XML de configuración.

Debe disponer del permiso S3: DeleteBucketMetadataNotification para un bucket, para poder eliminar las opciones de configuración de la integración de búsqueda para dicho bucket.

Solicitud

Cabecera HTTP de la solicitud	Descripción
Authorization (Autorización)	Especifica la firma AWS y la ID de la clave de acceso para la cuenta que se usará para la solicitud.
Date (Fecha)	La fecha y hora de la solicitud.
Host	El nombre del host al que se dirige la solicitud.

Ejemplos de solicitud

Deshabilitación del servicio de integración de búsqueda de un bucket

```
DELETE /test1?x-ntap-sg-metadata-notification HTTP/1.1
Host: example.com
Date: Fri, 21 Jul 2017 18:21:34 +0000
Authorization: AWS QYUTN90RX0RXO70QEGU8:y50RN9wUAYL5BnK+eFci4fz0D7U=
```

Solicitud de configuración de notificación de metadatos GET Bucket

La solicitud de configuración de la notificación de metadatos GET Bucket le permite recuperar el código XML de configuración utilizado para configurar la integración de búsqueda para cada bucket.

Debe disponer del permiso s3:GetBucketMetadataNotification para poder completar esta operación.

Solicitud

Cabecera HTTP de la solicitud	Descripción
Authorization (Autorización)	Especifica la firma AWS y la ID de la clave de acceso para la cuenta que se usará para la solicitud.
Date (Fecha)	La fecha y hora de la solicitud.
Host	El nombre del host al que se dirige la solicitud.

Ejemplo de solicitud

Esta solicitud obtiene la configuración de notificación de metadatos para el bucket denominado "bucket".

```
GET /bucket?x-ntap-sg-metadata-notification HTTP/1.1
Host: example.com
Date: Thu, 20 Jul 2017 18:25:38 +0000
Authorization: AWS QYUTN90RX0RXO70QEGU8:/XpYXJFVGp5pXh0se26ZzxxkuNE=
```

Respuesta

Cabecera HTTP de la	Descripción
Connection (Conexión)	Especifica si la conexión al servidor está abierta o cerrada.
Content-Length (Longitud del	La longitud del cuerpo de la respuesta.
Content-Type (Tipo de contenido)	El tipo Multipurpose Internet Mail Extensions (MIME-Extensiones multipropósito del correo de internet) del cuerpo de la respuesta.
Date (Fecha)	La fecha y hora de la respuesta.
Server (Servidor)	El servidor que crea la respuesta.
x-amz-request-id	El identificador que identifica de manera única la solicitud. Creada por la S3 API.

El cuerpo de la respuesta incluye la configuración de la notificación del metadato para el bucket. La configuración de notificación de los metadatos le permite determinar cómo se configura el bucket para la integración de búsqueda. Es decir, le permite determinar qué objetos están indexados y a qué endpoints se enviarán los metadatos de los objetos.

```
<MetadataNotificationConfiguration>
    <Rule>
        <ID>Rule-1</ID>
        <Status>rule-status</Status>
        <Prefix>key-prefix</Prefix>
        <Destination>
           <Urn>arn:aws:es:region:account-ID:domain/mydomain/myindex/
mytype</Urn>
        </Destination>
    </Rule>
    <Rule>
        <ID>Rule-2</ID>
          ...
    </Rule>
     ...
</MetadataNotificationConfiguration>
```

Cada configuración de notificación de metadatos incluye una o más reglas. Cada regla especifica los objetos a los que se aplica y el destino donde el sistema StorageGRID Webscale debe enviar los metadatos del objeto. Se deben especificar los destinos utilizando el URN de un endpoint de StorageGRID Webscale. Consulte la Guía del Administrador del tenant para obtener más información sobre cómo configurar endpoints y sobre el servicio de integración de búsquedas.

Nombre	Descripción	¿Necesario
MetadataNotificationConfiguration	Etiqueta contenedor para las reglas utilizadas para especificar los objetos y el destino de las notificaciones de metadatos. Contiene uno o más elementos de Regla.	Sí

Nombre	Descripción	¿Necesario
Rule (regla)	Etiqueta contenedor para una regla que identifica los objetos cuyos metadatos deben agregarse a un índice especificado. Se rechazarán las reglas que tengan prefijos superpuestos. Incluida en el elemento MetadataNotificationConfiguration.	Sí
ID	Identificador único para la regla. Incluida en el elemento Rule (regla)	No
Status (estado)	El estado puede ser 'Enabled' o 'Disabled' (Habilitado o Deshabilitado). No se realiza ninguna acción para las reglas que estén deshabilitadas. Incluida en el elemento Rule (regla)	Sí
Prefix (Prefijo)	Los objetos que coinciden con el prefijo se ven afectados por la regla y sus metadatos se envían al destino especificado. Para coincidir con todos los objetos, especifique un prefijo vacío. Incluida en el elemento Rule (regla)	Sí
Destination (Destino)	Etiqueta contenedor para el destino de una regla. Incluida en el elemento Rule (regla)	Sí
Urn	URN de destino donde se envían los metadatos del objeto. Debe ser el URN de un endpoint de StorageGRID Webscale con las siguientes propiedades: • "es" debe ser el tercer elemento. • El URN debe finalizar con el índice y el tipo donde se almacenan los metadatos, en la forma "nombre de dominio / miíndice/ mitipo". Los endpoints se configuran utilizando la Interfaz de Administración del tenant o la API de Administración del tenant. Tienen el siguiente formato: • `arn:aws:es:`*`region`*`:`*`account-ID`*`:domain/`*`midominio/miindice/mitipo`* • `urn:misitio:es:::`*`midominio/miíndice/mitipo`* El endpoint debe configurarse antes de enviar el código XML de configuración o la configuración fallará produciendo un error 404. La Urn está incluida en el elemento Destination (Destino)	Sí

Ejemplo de respuesta

El código XML incluido entre las etiquetas `<MetadataNotificationConfiguration> </MetadataNotificationConfiguration>` muestra la forma en que se encuentra configurada la integración con un endpoint de búsqueda para el bucket. En este ejemplo, se envían los metadatos del objeto a un índice Elasticsearch denominado “current” y con un tipo denominado “2017” que se encuentra almacenado en un dominio AWS denominado “records.”

```
HTTP/1.1 200 OK
Date: Thu, 20 Jul 2017 18:24:05 GMT
Connection: KEEP-ALIVE
Server: StorageGRID/11.0.0
x-amz-request-id: 3832973499
Content-Length: 264
Content-Type: application/xml

<MetadataNotificationConfiguration>
    <Rule>
        <ID>Rule-1</ID>
        <Status>Enabled</Status>
        <Prefix>2017</Prefix>
        <Destination>
           <Urn>arn:aws:es:us-east-1:3333333:domain/records/current/2017</Urn>
        </Destination>
    </Rule
       >
</MetadataNotificationConfiguration>
```

Información relacionada

Guía del Administrador del tenant de StorageGRID Webscale 11.0

Solicitud de configuración de notificación de metadatos PUT Bucket

La solicitud de configuración de notificación de metadatos PUT Bucket le permitirá habilitar el servicio de integración de búsqueda para buckets individuales. El código XML de configuración de notificación de metadatos que proporciona en el cuerpo de la solicitud especifica los objetos cuyos metadatos se envían al índice de búsqueda de destino.

> **Atención:** StorageGRID Webscale 11.0 incluye la versión inicial de los servicios de plataforma. En la actualidad, la duplicación CloudMirror, las notificaciones y la integración de búsquedas solo resultan apropiadas para determinadas situaciones y cargas de trabajo. Tendrá que ponerse en contacto con su representante de NetApp si desea utilizar la versión inicial de estos servicios.

Debe disponer del permiso s3:PutBucketMetadataNotification para un bucket para habilitar o deshabilitar la configuración de notificación de metadatos para el bucket.

Solicitud

Cabecera HTTP de la solicitud	Descripción
Authorization (Autorización)	Especifica la firma AWS y la ID de la clave de acceso para la cuenta que se usará para la solicitud.
Date (Fecha)	La fecha y hora de la solicitud.
Host	El nombre del host al que se dirige la solicitud.

La solicitud debe incluir la configuración de la notificación de los metadatos en el cuerpo de la solicitud. Cada configuración de notificación de metadatos incluye una o más reglas. Cada regla especifica los objetos a los que se aplica y el destino donde el sistema StorageGRID Webscale debe enviar los metadatos del objeto.

Los objetos se pueden filtrar utilizando el prefijo del nombre del objeto. Por ejemplo, puede enviar metadatos para objetos con el prefijo "/imagenes" a un destino y objetos con el prefijo "/videos" a otro.

Aquellas configuraciones que cuenten con prefijos que se superpongan no resultarán válidas, y serán

rechazadas cuando se envíen. Por ejemplo, no se permitiría una configuración que incluyera una regla para objetos con el prefijo "prueba" y una segunda regla para objetos con el prefijo "prueba2".

Los destinos se deben especificar utilizando el URN de un endpoint de StorageGRID Webscale. El endpoint debe existir cuando se envía la configuración de la notificación de los metadatos o, en caso contrario, la solicitud fallará recibiendo un mensaje `404 Not Found` y el código de error `NoSuchEndpoint` .

Consulte la Guía del Administrador del tenant para obtener más información sobre cómo configurar los endpoints y sobre el servicio de integración de búsquedas.

```
<MetadataNotificationConfiguration>
    <Rule>
        <ID>Rule-1</ID>
        <Status>rule-status</Status>
        <Prefix>key-prefix</Prefix>
        <Destination>
           <Urn>arn:aws:es:region:account-ID:domain/mydomain/myindex/
mytype</Urn>
        </Destination>
    </Rule>
    <Rule>
        <ID>Rule-2</ID>
         ...
    </Rule>
     ...
</MetadataNotificationConfiguration>
```

La tabla mostrada a continuación describe los elementos contenidos en el código XML de configuración de la notificación de metadatos.

Nombre	Descripción	¿Necesario
MetadataNotificationConfiguration	Etiqueta contenedor para las reglas utilizadas para especificar los objetos y el destino de las notificaciones de metadatos. Contiene uno o más elementos de Regla.	Sí
Rule (regla)	Etiqueta contenedor para una regla que identifica los objetos cuyos metadatos deben agregarse a un índice especificado. Se rechazarán las reglas que tengan prefijos superpuestos. Incluida en el elemento MetadataNotificationConfiguration.	Sí
ID	Identificador único para la regla. Incluida en el elemento Rule (regla)	No
Status (estado)	El estado puede ser 'Enabled' o 'Disabled' (Habilitado o Deshabilitado). No se realiza ninguna acción para las reglas que estén deshabilitadas. Incluida en el elemento Rule (regla)	Sí
Prefix (Prefijo)	Los objetos que coinciden con el prefijo se ven afectados por la regla y sus metadatos se envían al destino especificado. Para coincidir con todos los objetos, especifique un prefijo vacío. Incluida en el elemento Rule (regla)	Sí
Destination (Destino)	Etiqueta contenedor para el destino de una regla. Incluida en el elemento Rule (regla)	Sí

Nombre	Descripción	¿Necesario
Urn	URN de destino donde se envían los metadatos del objeto. Debe ser el URN de un endpoint de StorageGRID Webscale con las siguientes propiedades: • "es" debe ser el tercer elemento. • El URN debe finalizar con el índice y el tipo donde se almacenan los metadatos, en la forma "nombre de dominio / miíndice/ mitipo". Los endpoints se configuran utilizando la Interfaz de Administración del tenant o la API de Administración del tenant. Tienen el siguiente formato: • `arn:aws:es:`*`region`*`:`*`account-ID`*`:domain/`*`midominio/miindice/ mitipo`* • `urn:misitio:es:::`*`midominio/miíndice/mitipo`* El endpoint debe configurarse antes de enviar el código XML de configuración o la configuración fallará produciendo un error 404. La Urn está incluida en el elemento Destination (Destino)	Sí

Ejemplos de solicitud

Habilitación del servicio de integración de búsqueda de un bucket

En este ejemplo, se envían al mismo destino los metadatos de objeto para todos los objetos

```
PUT /test1?x-ntap-sg-metadata-notification HTTP/1.1
Host: example.com
Date: Thu, 20 Jul 2017 18:21:34 +0000
Authorization: AWS QYUTN90RX0RXO70QEGU8:y50RN9wUAYL5BnK+eFci4fz0D7U=

<MetadataNotificationConfiguration>
    <Rule>
        <ID>Rule-1</ID>
        <Status>Enabled</Status>
        <Prefix></Prefix>
        <Destination>
           <Urn>urn:sgws:es:::sgws-notifications/test1/all</Urn>
        </Destination>
    </Rule>
</MetadataNotificationConfiguration>
```

En este ejemplo, los metadatos de los objetos que coinciden con el prefijo `/images` se envían a un destino, mientras que los metadatos correspondientes a los objetos que coinciden con el prefijo `/videos` se envían a un segundo destino.

```
PUT /graphics?x-ntap-sg-metadata-notification HTTP/1.1
Host: example.com
Date: Thu, 20 Jul 2017 18:21:34 +0000
Authorization: AWS QYUTN90RX0RXO70QEGU8:y50RN9wUAYL5BnK+eFci4fz0D7U=

<MetadataNotificationConfiguration>
    <Rule>
        <ID>Images-rule</ID>
```

```
        <Status>Enabled</Status>
        <Prefix>/images</Prefix>
        <Destination>

           <Urn>arn:aws:es:us-east-1:3333333:domain/es-domain/graphics/imagetype</
Urn>
        </Destination>
    </Rule>
    <Rule>
        <ID>Videos-rule</ID>
        <Status>Enabled</Status>
        <Prefix>/videos</Prefix>
        <Destination>
           <Urn>arn:aws:es:us-west-1:22222222:domain/es-domain/graphics/
```

```
videotype</Urn>
        </Destination>
    </Rule>
</MetadataNotificationConfiguration>
```

Información relacionada

Guía del Administrador del tenant de StorageGRID Webscale 11.0

Metadatos de objetos incluidos en notificaciones de metadatos

La tabla siguiente enumera todos los campos que se incluyen en el documento JSON que se envía al endpoint de destino cuando la integración de búsqueda está habilitada.

El nombre del documento incluye el nombre del bucket, el nombre del objeto y la ID de la versión, si está presente.

Tipo	Nombre del	Descripción
Información del bucket y del objeto	bucket	Nombre del bucket
	key (clave)	Nombre de la clave del objeto
	versionID	Versión del objeto, para los objetos contenidos en buckets con versión
Metadatos del sistema	md5	hash del objeto
	size (tamaño)	Tamaño del objeto (en bytes) tal y como se presenta ante un cliente HTTP
Metadatos de usuario	metadata *key:value (Clave:valor)*	Todos los metadatos de usuario para el objeto, como pares clave-valor
Etiquetas	tags *key:value (Clave:valor)*	Todas las etiquetas de objeto definidas para el objeto, como pares clave-valor

JSON generado por el servicio de integración de búsqueda

Cuando se habilita el servicio de integración de búsqueda para un bucket, se genera un documento JSON que se envía al endpoint de destino cada vez que se agregan, actualizan o eliminan metadatos o etiquetas de un objeto.

Este ejemplo muestra un JSON que podría generarse cuando se crea un objeto con la clave SGWS / Tagging.txt en un bucket llamado "test". El bucket "test" no está versionado, por lo que la etiqueta versionID está vacía.

```
{
  "bucket": "test",
  "key": "SGWS/Tagging.txt",
  "versionId": "",
  "accountId": "86928401983529626822",
  "size": 38,
  "md5": "3d6c7634a85436eee06d43415012855",
  "metadata": {
    "age": "25"
  },
  "tags": {
    "color": "yellow"
  }
}
```

Configuración de la seguridad para la REST API

Debe comprender las medidas de seguridad implementadas para la REST API y cómo proteger su

sistema.

Cómo implementa el sistema StorageGRID Webscale la seguridad para la REST API

El sistema StorageGRID Webscale utiliza la seguridad de conexión Transport Layer Security (TLS, Seguridad de la Capa de Transporte), autenticación del servidor, autenticación del cliente y autorización del cliente. Al pensar en los problemas de seguridad, puede resultarle útil comprender la forma en que el sistema StorageGRID Webscale implementa la seguridad, la autenticación y la autorización para la REST API.

El sistema StorageGRID Webscale acepta instrucciones HTTPS enviadas a través de una conexión de red que usa TLS para proporcionar seguridad de conexión, autenticación de aplicaciones y, opcionalmente, cifrado de transporte. Los comandos que no usan TLS son rechazados. Si un objeto se cifra cuando se ingiere, permanecerá cifrado durante su tiempo de vida en el sistema StorageGRID Webscale.

TLS permite el intercambio de certificados como credenciales de entidad y permite una negociación que puede usar algoritmos de hash y de cifrado.

Cuando se instala un sistema StorageGRID Webscale, se genera un certificado por la autoridad de certificación (CA) para el sistema, así como certificados de servidor para cada nodo de almacenamiento. Todos estos certificados son firmados por la CA del sistema. Tiene que configurar las aplicaciones cliente para que confíen en este certificado CA. Cuando una aplicación cliente se conecta a cualquier Nodo de Almacenamiento mediante TLS, la aplicación puede autenticar el Nodo de Almacenamiento verificando que el certificado del servidor presentado por el Nodo de Almacenamiento esté firmado por la CA del sistema de confianza.

Como alternativa, puede optar por suministrar un único certificado de servidor personalizado que se debe usar en todos los Nodos de Almacenamiento en lugar de en los generados. El certificado de servidor personalizado debe estar firmado por una CA seleccionada por el administrador. El proceso de autenticación del servidor por la aplicación cliente es el mismo, pero en este caso con una CA de confianza diferente. Si desea obtener más información, consulte "Configuración de certificados" en la Guía del Administrador.

La siguiente tabla muestra cómo se implementan los temas de seguridad en las API REST de S3 y Swift:

Tema de seguridad	Implementación en la REST API
Seguridad de la conexión	TLS
Autenticación del servidor	Certificado del servidor X.509 firmado por una CA del sistema o certificado personalizado del servidor suministrado por el administrador
Autenticación del cliente	• S3: cuenta S3 (ID de la clave de acceso y clave de acceso secreta) • Swift: Cuenta Swift (nombre de usuario y contraseña) **Nota:** Bajo petición, puede habilitar el servicio de identidad Keystone de OpenStack para su empleo con la Swift REST API. En el caso de que Keystone esté habilitado, podrá usar un testigo adicional para la validación. Para habilitar la compatibilidad con Keystone, póngase en contacto con su representante NetApp.

Tema de seguridad	Implementación en la REST API
Autorización del cliente	• S3: Propiedad del bucket y todas las políticas de control de acceso que sean aplicables • Swift: Acceso al rol de administración de la cuenta

Información relacionada

Guía del administrador de StorageGRID Webscale 11.0

Políticas de acceso al grupo y bucket

El sistema StorageGRID Webscale implementa un subconjunto del lenguaje de la política REST API de S3 que puede usar para controlar el acceso a los buckets y objetos contenidos en esos buckets.

Introducción

StorageGRID Webscale utiliza la sintaxis del lenguaje de políticas de Amazon Web Services (AWS, Servicios Web de Amazon) para permitir a los tenants de S3 la creación de políticas de acceso para sus datos. Las políticas de acceso para la API de S3 están escritas en JSON. Hay dos tipos de políticas de acceso permitidas por StorageGRID Webscale:

- **Políticas de bucket**, que se configuran utilizando las operaciones de la API de S3 de las políticas GET Bucket, PUT Bucket y DELETE Bucket. Las políticas de bucket están unidas a los buckets, por lo que están configuradas para controlar el acceso de los usuarios en la cuenta del propietario del bucket u otras cuentas del bucket y a los objetos que contiene. Una política de bucket es de aplicación a un solo bucket y, posiblemente, a grupos múltiples.
- **Políticas de grupo** que se configuran utilizando la Interfaz de Administración del tenant o la API de Administración del tenant. Las políticas de grupo se unen a un grupo en la cuenta, por lo que están configuradas para permitir que ese grupo acceda a los recursos específicos que pertenecen a esa cuenta. Las políticas de grupo se aplican a un solo grupo y, posiblemente, a varios buckets.

Las políticas de grupo y de bucket de StorageGRID Webscale siguen una gramática específica definida por Amazon. Dentro de cada política hay una matriz de instrucciones de política, y cada declaración contiene los siguientes elementos:

- ID de la declaración (ID) (opcional)
- Efecto
- Principal / No Principal
- Recurso / Sin recurso
- Acción / Sin Acción
- Condición (opcional)

Las declaraciones de la política se construyen utilizando la siguiente estructura para especificar los permisos: Conceder <Efecto> para permitir / denegar
<Principal> para realizar <Acción> sobre <Recurso> cuando se cumpla

<Condición>. Cada elemento de la política se utiliza para una función

específica:

Elemento	Descripción
Sid	El elemento Sid es opcional. El objetivo del Sid es servir solo como una descripción para el usuario. El sistema StorageGRID Webscale lo almacena pero no lo interpreta.

Elemento	Descripción
Efecto	Utilice el elemento efecto (Effect) para establecer si las operaciones especificadas han sido permitidas o denegadas. Debe identificar las operaciones que permite (o niega) en los buckets u objetos utilizando las palabras clave del elemento de acción soportado.
Principal / No Principal	Puede permitir que usuarios, grupos y cuentas accedan a recursos específicos y realicen acciones específicas. Si no se incluye una firma S3 en la solicitud, se permite el acceso anónimo especificando el carácter comodín (*) como el principal. De forma predeterminada, solo la cuenta raíz tiene acceso a recursos que son propiedad de la cuenta. Solo necesita especificar el elemento Principal en una política de bucket. Para las políticas de grupo, el grupo al que se encuentra unida la política es el elemento principal implícito.
Recurso / Sin recurso	El elemento Recurso (Resource) identifica buckets y objetos. Puede permitir o denegar permisos a buckets y objetos utilizando el nombre uniforme de recurso (URN) para identificar el recurso.
Acción / Sin Acción	Los elementos Acción (Action) y Efecto (Effect) son los dos componentes de los permisos. Cuando un grupo solicita a un recurso, se le puede conceder o denegar el acceso a dicho recurso. Se le negará el acceso salvo que específicamente le asigne permisos, pero puede denegar explícitamente para sobrescribir un permiso concedido por otra política.
Condición	El elemento condición (Condition) es opcional. Las condiciones le permiten crear expresiones para determinar cuándo se debe aplicar una política.

En el elemento Acción, puede usar el carácter comodín (*) para especificar todas las operaciones o un subconjunto de las mismas. Por ejemplo, esta Acción concede permisos tales como s3:GetObject, s3:PutObject, y s3:DeleteObject.

```
s3:*Object
```

En el elemento Recurso (Resource) podrá utilizar caracteres comodín (*) y (?). Mientras que el asterisco (*) sustituye a 0 o más caracteres, el signo de interrogación (?) sustituye a un solo carácter.

El elemento Principal no permite el uso de caracteres comodín salvo para definir acceso anónimo, que permite el acceso de todo el mundo. Por ejemplo, podrá asignar el comodín (*) como el valor Principal.

```
"Principal":"*"
```

En el ejemplo siguiente, la declaración utiliza los elementos Efecto, Principal, Acción y Recurso. Este ejemplo muestra una declaración de política completa de bucket que utiliza el Efecto "Allow" para proporcionar al grupo de administradores `federated-group/admin` y al grupo financiero `federated-group/ finance`, permisos para realizar la Acción `s3:ListBucket` en el bucket denominado "mybucket" y la Acción `s3:GetObject` en todos los objetos contenidos en dichobucket.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "SGWS": [
          "urn:sgws:identity::27233906934684427525:federated-group/admin",
          "urn:sgws:identity::27233906934684427525:federated-group/finance"
        ]
      },
      "Action": [
        "s3:ListBucket",
```

```
        "s3:GetObject"
      ],
      "Resource": [
        "urn:sgws:s3:::mybucket",
        "urn:sgws:s3:::mybucket/*"
      ]
    }
  ]
}
```

La política del bucket tiene un tamaño límite de 20.480 bytes y la política de grupo tiene un tamaño límite de 5.120 bytes.

Opciones de control de coherencia para políticas

De manera predeterminada, cualquier actualización que realice a las políticas del grupo serán coherentes. Una vez que una política de grupo se vuelve coherente, los cambios pueden tardar 15 minutos más en aplicarse, debido al almacenamiento en caché de la política.

De manera predeterminada, cualquier actualización que realice a las políticas del bucket finalmente serán coherentes. Sin embargo, según sea necesario, puede cambiar las garantías de coherencia para las actualizaciones de la política del bucket. Por ejemplo, puede que desee cambiar la política de un bucket para que sea efectiva tan pronto como sea posible por motivos de seguridad.

En este caso, podrá definir la cabecera `Consistency-Control (Control de coherencia)` en la solicitud de política PUT Bucket o podrá utilizar la Solicitud de coherencia de PUT Bucket. Cuando cambie el control de coherencia para esta solicitud deberá utilizar el valor **all** (todo), que proporciona la mayor garantía de la coherencia "lectura tras la escritura". Si especifica cualquier otro valor del control de coherencia en una cabecera para la Solicitud de coherencia de PUT Bucket, se rechazará la solicitud. Si especifica cualquier otro valor para una solicitud de política PUT Bucket, se ignorará el valor. Una vez que una política de bucket se vuelve coherente, los cambios pueden tardar 8 segundos más en aplicarse, debido al almacenamiento en caché de la política.

> **Nota:** Si define el nivel de coherencia como **all** (todos) para forzar que una nueva política de bucket entre antes en vigor, asegúrese de volver a asignar el valor original al control de nivel de bucket cuando haya terminado. En caso contrario, todas las futuras solicitudes de bucket utilizarán el valor **all** .

Empleo del URN en declaraciones de política

En declaraciones de política, el URN se utiliza en los elementos Principal y Resource (Recurso) en la forma siguiente.

- Utilice esta sintaxis para especificar el URN del recurso S3:

```
urn:sgws:s3:::nombre_bucketurn:sgws:s3:::no
mbre_bucket/object_key
```

- Utilice esta sintaxis para especificar el URN del recurso de identidad (usuarios y grupos):

```
urn:sgws:identity::account_id:root
urn:sgws:identity::account_id:user/user_name
urn:sgws:identity::account_id:group/group_name
urn:sgws:identity::account_id:federated-user/user_name
urn:sgws:identity::account_id:federated-group/group_name
```

Otras cuestiones:

- Puede utilizar el asterisco (*) como un comodín para que coincida con cero o más caracteres dentro de la clave del objeto.
- Los caracteres internacionales, que se pueden especificar en la clave del objeto, se deben codificar utilizando JSON UTF-8 o utilizando las secuencias de escape JSON \ u. La codificación porcentaje, tal y como se indica en la sintaxis RFC 2141 URN , no es compatible. El cuerpo de la solicitud HTTP para la operación de política PUT Bucket debe estar codificada con charset=UTF-8.

Especificación de los recursos en una política

En las declaraciones de política, puede usar el elemento Resource (Recurso) para especificar el bucket o el objeto para el cual se permiten o niegan los permisos.

- Cada declaración de política requiere un elemento Resource. En una política, los recursos se declaran mediante el elemento "Resource", o como alternativa, "NotResource" para el caso de exclusión.
- Podrá especificar recursos con un URN de recurso de S3. Por ejemplo:

```
"Resource": "urn:sgws:s3:::mybucket/*"
```

- También puede usar variables de política dentro de la clave del objeto. Por ejemplo:

```
"Resource": "urn:sgws:s3:::mybucket/home/${sgws:username}/*"
```

Consulte Especificación de variables en una política en la página 58 para ver una lista de las variables de política que están disponibles.

- El valor del recurso puede especificar un bucket que aún no existe cuando se crea una política de grupo.

Especificación de principales en una política

Use el elemento Principal para identificar al usuario, al grupo o a la cuenta tenant que puede acceder, o no, al recurso mediante la declaración de política.

- Al construir una declaración de política de grupo, no se especifica ningún elemento principal porque se entiende que el grupo es el principal.
- Cada declaración de política debe incluir un elemento Principal (a menos que sea una política de grupo). En una política, los principales se indican mediante el elemento "Principal" o alternativamente "NotPrincipal" para indicar exclusión.
- Las identidades basadas en cuenta se deben especificar utilizando un ID o un URN:

```
"Principal": { "SGWS": "account_id"}
"Principal": { "SGWS": "identity_urn" }
```

- En el siguiente ejemplo, se utiliza el ID de la cuenta tenant 27233906934684427525, que incluye la raíz de la cuenta y todos los usuarios de dicha cuenta:

```
 "Principal": { "SGWS": "27233906934684427525" }
```

- Podrá especificar solo la cuenta raíz:

```
"Principal": { "SGWS": "urn:sgws:identity::
27233906934684427525:root" }
```

- Podrá especificar un usuario determinado ("Bob"):

```
"Principal": { "SGWS": "urn:sgws:identity::
27233906934684427525:federated-user/Bob" }
```

- También puede especificar un grupo específico ("Managers"):

```
"Principal": { "SGWS": "urn:sgws:identity::
27233906934684427525:federated-group/Managers"}
```

- Puede especificar un principal que sea anónimo:

```
"Principal": "*"
```

- En el caso de que se borrara el nombre de usuario `Bob` una vez que éste dejara la empresa y, posteriormente, otro Bob se incorporara en la organización y se le asignara el mismo nombre de usuario `Bob`, podría heredar de forma no intencionada los permisos concedidos al Bob anterior. Para evitar este tipo de ambigüedades, se puede utilizar el UUID del usuario en lugar del nombre de usuario. Por ejemplo:

```
urn:sgws:identity::27233906934684427525:user-uuid/de305d54-75b4-431b-
adb2-eb6b9e546013
```

- El valor principal puede especificar un nombre de grupo / usuario que aún no existe cuando se crea una política de bucket.

Cómo especificar permisos en una política

En una política, el elemento Action (Acción) se usa para permitir / denegar permisos a un recurso. Hay un conjunto de permisos que puede especificar en una política, que se denotan mediante el elemento "Action" o, alternativamente, "NoAction" para el caso de querer indicar exclusión. Cada uno de estos elementos se correlaciona con operaciones específicas de la API de REST de S3.

Tabla 1: Permisos aplicables a los buckets

Permisos	Operaciones de la API REST de S3
s3:CreateBucket	PUT Bucket
s3:DeleteBucket	DELETE Bucket
s3:DeleteBucketMetadataNotification	DELETE Bucket, configuración de notificación de metadatos
s3:DeleteBucketPolicy	DELETE Política de bucket
s3:GetBucketAcl	GET Bucket ACL
s3:GetBucketConsistency	GET Coherencia de Bucket
s3:GetBucketLastAccessTime	GET Hora del último acceso al Bucket
s3:GetBucketLocation	GET Ubicación del bucket
s3:GetBucketMetadataNotification	GET Configuración de la notificación de metadatos del bucket
s3:GetBucketNotification	GET Notificación del Bucket
s3:GetBucketPolicy	GET Política del bucket
s3:GetBucketReplication	GET Duplicación del bucket
s3:GetBucketVersioning	GET Control de versiones del bucket
s3:ListAllMyBuckets	GET Servicio, GET Empleo de almacenamiento
s3:ListBucket	GET Bucket (Lista de objetos), HEAD Bucket
s3:ListBucketMultipartUploads	Lista de cargas múltiples
s3:ListBucketVersions	GET Versiones del bucket
s3:PutBucketConsistency	PUT Coherencia del bucket
s3:PutBucketLastAccessTime	PUT Hora del último acceso al Bucket

Permisos	Operaciones de la API REST de S3
s3:PutBucketMetadataNotification	PUT Configuración de la notificación de metadatos del bucket
s3:PutBucketNotification	PUT Notificación del bucket
s3:PutBucketPolicy	PUT Política del bucket
s3:PutBucketReplication	PUT Duplicación del bucket
s3:PutBucketVersioning	PUT Control de versiones del bucket

Tabla 2: Permisos aplicables a los objetos

Permisos	Operaciones de la API REST de S3
s3:AbortMultipartUpload	Abortar Cargas Múltiples
s3:DeleteObject	DELETE Objeto, DELETE Varios objetos
s3:DeleteObjectTagging	DELETE Etiquetado de objeto
s3:DeleteObjectVersionTagging	DELETE Etiquetado de objeto (una versión específica del objeto)
s3:DeleteObjectVersion	DELETE objeto (una versión específica del objeto)
s3:GetObject	GET Objeto, HEAD Objeto
s3:GetObjectAcl	GET Objeto ACL
s3:GetObjectTagging	GET Etiquetado del Objeto
s3:GetObjectVersionTagging	GET Etiquetado de objeto (una versión específica del objeto)
s3:GetObjectVersion	GET Objeto (una versión específica del objeto)
s3:ListMultipartUploadParts	Listar Partes
s3:PutObject	PUT Objeto, PUT Objeto - Copia, Iniciar Carga Múltiple, Completar Carga Múltiple, cargar partes (Cargar Parte y Cargar Parte - Copia)
s3:PutObjectTagging	PUT Etiquetado de Objeto
s3:PutObjectVersionTagging	PUT Etiquetado de objeto (una versión específica del objeto)
s3:PutOverwriteObject	PUT Objeto, PUT Objeto - Copia, Completar Carga Múltiple

Empleo del permiso PutOverwriteObject

El permiso PutOverwriteObject se aplica a las operaciones que crean o actualizan objetos (por ejemplo, PUT nuevos objetos o PUT Copia para actualizar los metadatos). La configuración de este permiso determina si el cliente puede sobrescribir los datos o metadatos de un objeto. Las opciones posibles incluyen Allow (el cliente puede sobrescribir un objeto) o Deny (el cliente no puede sobrescribir un objeto). El valor predeterminado es Allow. Cuando este permiso no esté presente, el efecto es el mismo que si se configurara Allow (permitir).

Cuando se configura como Deny, estos permisos funcionan de la siguiente manera.

- Si se encuentra un objeto existente en la misma ruta:

  - No se puede sobrescribir ningún dato o metadato del objeto.
  - Se cancelará cualquier operación de ingestión que esté en marcha y se proporciona un error.
  - Si el control de versiones S3 se encuentra activado, el permiso Deny no tiene ningún efecto, ya que al tener el control de versiones habilitado no se permite la modificación de los datos o metadatos.

- Si no se encuentra ningún objeto existente, este permiso no tiene ningún efecto.

Importante: si la política S3 actual permite la sobrescritura y el permiso PutOverwriteObject tiene asignado el valor Deny, el cliente no puede sobrescribir los datos o metadatos de un objeto. Además, si la opción de malla denominada Prevent Client Modify (Impedir la modificación del cliente) tiene asignado el valor Enabled (Habilitado), esta opción sobrescribe la opción del permiso PutOverwriteObject.

Si desea ver un ejemplo del empleo del permiso PutOverwriteObject, consulte Ejemplo: permiso PutOverwriteObject en la página 64.

Cómo especificar permisos en una política

Puede usar condiciones para permitir que las políticas surtan efecto en función de los valores de la solicitud.

Los operadores de condiciones se clasifican de la siguiente manera:

- String (cadena)
- Numeric (numérico)
- Boolean
- IP address (dirección IP)
- Comprobación Null

Condiciones de empleo de los pares clave-valor para la evaluación. Un elemento Condition (Condición) puede contener varias condiciones y cada condición puede contener varios pares clave-valor. El bloque condición utiliza el siguiente formato:

```
Condition: {
     condition_type: {
          condition_key: condition_values
```

En el ejemplo siguiente la condición IpAddress utiliza la clave de condición SourceIp.

```
  "Condition": {
    "IpAddress": {
      "sgws:SourceIp": "54.240.143.0/24"
        ...
},
        ...
```

Tabla 3: operadores de condición compatibles

Operadores de condición	Descripción
StringEquals	Compara una clave con un valor de cadena basándose en la igualdad exacta (distingue entre mayúsculas y minúsculas).
StringNotEquals	Compara una clave con un valor de cadena basándose en la no igualdad exacta (distingue entre mayúsculas y minúsculas).

StringEqualsIgnoreCase	Compara una clave con un valor de cadena basándose en la igualdad exacta (no distingue entre mayúsculas y minúsculas).

Operadores de condición	Descripción
StringNotEqualsIgnoreCase	Compara una clave con un valor de cadena basándose en la no igualdad exacta (no distingue entre mayúsculas y minúsculas).
StringLike	Compara una clave con un valor de cadena y proporciona acceso si hay una coincidencia exacta (distingue entre mayúsculas y minúsculas). Puede incluir los caracteres comodines * y ?.
StringNotLike	Compara una clave con un valor de cadena y proporciona acceso a todas excepto a la cadena especificada (distingue entre mayúsculas y minúsculas). Puede incluir los caracteres comodines * y ?.
NumericEquals	Compara una clave con un valor numérico y proporciona acceso si hay una coincidencia exacta.
NumericNotEquals	Compara una clave con un valor numérico y proporciona acceso a todos salvo al valor especificado.
NumericGreaterThan	Compara una clave con un valor numérico y proporciona acceso si hay una coincidencia "mayor que".
NumericGreaterThanEquals	Compara una clave con un valor numérico y proporciona acceso si hay una coincidencia "mayor que o igual a".
NumericLessThan	Compara una clave con un valor numérico y proporciona acceso si hay una coincidencia "menor que".
NumericLessThanEquals	Compara una clave con un valor numérico y proporciona acceso si hay una coincidencia "menor que o igual a".
Bool	Compara una clave con un valor booleano y proporciona acceso en base a una coincidencia "verdadera o falsa".
IpAddress	Compara una clave con un valor numérico y proporciona acceso si hay una coincidencia con una IP o con un rango de direcciones IP.
NotIpAddress	Compara una clave con un valor numérico y proporciona acceso a todas las direcciones salvo a la IP o al rango de direcciones IP especificadas.
Null	Comprueba si una clave de condición está presente en el contexto de la solicitud actual.

Tabla 4: Claves de condición compatibles

Categoría	Claves de condición aplicable	Descripción
Operadores IP	sgws:SourceIp	Comparará con la dirección IP desde la cual se envió la solicitud. Se puede utilizar para operaciones sobre objetos o buckets.
Recurso / Identidad	sgws:username	Comparará con el nombre de usuario del remitente que envió la solicitud. Se puede utilizar para operaciones sobre objetos o buckets.

Categoría	Claves de condición	Descripción
Permisos S3:ListBucket y S3:ListBucketVersions	s3:delimiter	Comparará con el parámetro delimitador especificado en una solicitud de versiones GET Bucket o GET Bucket Object.
	s3:max-keys	Comparará con el parámetro max-keys especificado en una solicitud de versiones GET Bucket o GET Bucket Object.
	s3:prefix	Comparará con el parámetro prefix (prefijo) especificado en una solicitud de versiones GET Bucket o GET Bucket Object.

Especificación de variables en una política

Puede usar variables en las políticas para completar la información de la política cuando esté disponible. Puede utilizar variables de política en el elemento `Recurso` y en las comparaciones de cadena en el elemento `Condition`.

En este ejemplo, la variable `${sgws:username}` forma parte del elemento Resource:

```
"Resource": "urn:sgws:s3:::bucket-name/home/${sgws:username}/*"
```

En este ejemplo, la variable `${sgws:username}` forma parte del valor de condición en el bloque condition:

```
  "Condition": {
    "StringLike": {
      "s3:prefix": "${sgws:username}/*"
        ...
},
        ...
```

Variable	Descripción
${sgws:SourceIp}	Utiliza la clave SurceIp como variable proporcionada.
${sgws:username}	Utiliza la clave username como variable proporcionada.
${s3:prefix}	Utiliza la clave prefix específica del servicio como variable proporcionada.
${s3:max-keys}	Utiliza la clave max-keys específica del servicio como variable proporcionada.
${*}	Carácter especial Utiliza el carácter como un carácter * literal.
${?}	Carácter especial Utiliza el carácter como un carácter ? literal.
${$}	Carácter especial Utiliza el carácter como un carácter $ literal.

Creación de políticas que requieren una manipulación especial

En ocasiones, una política puede otorgar permisos que resultan peligrosos para la seguridad o peligrosos para operaciones continuas, como bloquear al usuario raíz de la cuenta. La implementación de la API REST S3 de StorageGRID Webscale es menos restrictiva durante la validación de políticas que Amazon, pero es igualmente estricta durante la evaluación de políticas.

Descripción de la política	Tipo de política	Comportamiento de Amazon	Comportamiento de StorageGRID
Negarse a sí mismo cualquier permiso a la cuenta raíz	Bucket	Válido y aplicado, pero la cuenta de usuario raíz conserva el permiso para todas las operaciones de política del bucket de S3	Igual
Se auto deniega cualquier permiso al usuario / grupo	Grupo	Válido y aplicado	Igual
Permite cualquier permiso a un grupo de cuentas externas	Bucket	Principal no válido	Válido, pero los permisos para todas las operaciones de política de bucket de S3 proporcionan un error del tipo 405 Method Not Allowed `(Método no permitido)` cuando una política lo permite
Permite cualquier permiso a un usuario o raíz de cuenta externa	Bucket	Válido, pero los permisos para todas las operaciones de política de bucket de S3 proporcionan un error del tipo 405 Method Not Allowed `(Método no permitido)` cuando una política lo permite	Igual
Permite todos los permisos para todas las acciones	Bucket	Válido, pero los permisos para todas las operaciones de política de bucket de S3 proporcionan un error del tipo 405 Method Not Allowed `(Método no permitido)` para los usuarios y la raíz de la cuenta externa	Igual
Deniega todos los permisos para todas las acciones	Bucket	Válido y aplicado, pero la cuenta de usuario raíz conserva el permiso para todas las operaciones de política del bucket de S3	Igual
El principal es un usuario o grupo no existente	Bucket	Principal no válido	Válido
El recurso es un bucket S3 no existente	Grupo	Válido	Igual
El principal es un grupo local	Bucket	Principal no válido	Válido

La política otorga a una cuenta no propietaria (incluidas las cuentas anónimas) permisos para objetos PUT	Bucket	Válido. Los objetos son propiedad de la cuenta creadora y no es de aplicación la política del bucket. La cuenta del creador debe otorgar permisos de acceso para el objeto utilizando las ACL del objeto.	Válido. Los objetos son propiedad de la cuenta dueña del bucket. Se aplica la política del bucket.

Conceptos relacionados

Protección Write-once-read-many (WORM - Una escritura Muchas lecturas) en la página 65

Información relacionada

Guía del administrador de StorageGRID Webscale 11.0

Ejemplos de política

Utilice los ejemplos mostrados en esta sección para construir las políticas de acceso de StorageGRID Webscale para buckets y grupos.

Ejemplos de política de bucket

Las políticas de bucket especifican los permisos de acceso para el bucket al que está asociada la política. Las políticas del bucket se configuran utilizando la API PutBucketPolicy de S3.

Una política de bucket se puede configurar utilizando la AWS CLI tal y como se muestra en la siguiente instrucción:

```
> aws s3api put-bucket-policy --bucket examplebucket --policy file://policy.json
```

Por ejemplo: permite a todo el mundo acceso de solo lectura a un bucket

En este ejemplo, todos, incluido anónimo, pueden Listar el bucket y realizar operaciones GetObject en todos los objetos contenidos en el bucket. Ninguna otra operación está disponible. Tenga en cuenta que esta política puede no ser particularmente útil ya que nadie, excepto la cuenta raíz, tiene permisos para escribir en el bucket.

```
{
  "Statement": [
    {
      "Sid": "AllowEveryoneReadOnlyAccess",
      "Effect": "Allow",
      "Principal": "*",
      "Action": [ "s3:GetObject", "s3:ListBucket" ],
      "Resource": ["urn:sgws:s3:::examplebucket","urn:sgws:s3:::examplebucket/*"],
    }
  ]
}
```

Ejemplo: Permitir a todos los usuarios de una cuenta acceso completo a un bucket y a todos los usuarios de otra cuenta acceso de solo lectura a un bucket.

En este ejemplo, todos los usuarios de una cuenta especificada tienen acceso completo a un bucket, mientras que todos los usuarios de otra cuenta especificada solo tienen permiso para listar el bucket y realizar operaciones GetObject sobre los objetos contenidos en el bucket que comienza con el prefijo de clave de objeto "shared /'.

Nota: En StorageGRID Webscale, todos los objetos son propiedad de la cuenta del propietario del bucket, incluidos los objetos creados por una cuenta que no sea del propietario (incluyendo las cuentas anónimas). La política del bucket se aplica a estos objetos.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "SGWS": "95390887230002558202"
      },
      "Action": "s3:*",
      "Resource": [
        "urn:sgws:s3:::examplebucket",
        "urn:sgws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Allow",
      "Principal": {
        "SGWS": "31181711887329436680"
      },
      "Action": "s3:GetObject",
      "Resource": "urn:sgws:s3:::examplebucket/shared/*"
    },
    {
```

```
      "Effect": "Allow",
      "Principal": {
        "SGWS": "31181711887329436680"
      },
      "Action": "s3:ListBucket",
      "Resource": "urn:sgws:s3:::examplebucket",
      "Condition": {
        "StringLike": {
          "s3:prefix": "shared/*"
        }
      }
    }
  ]
}
```

Ejemplo: Permitir a todo el mundo acceso de solo lectura a un bucket y pleno acceso a un grupo especificado.

En este ejemplo, todos, incluyendo la cuenta anónima, pueden Listar el bucket y realizar operaciones GetObject en todos los objetos del bucket, mientras que solo los usuarios que pertenecen al grupo `Marketing` en la cuenta especificada tienen acceso total.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "SGWS": "urn:sgws:identity::95390887230002558202:federated-group/Marketing"
      },
      "Action": "s3:*",
      "Resource": [
        "urn:sgws:s3:::examplebucket",
        "urn:sgws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": ["s3:ListBucket","s3:GetObject"],
      "Resource": [
        "urn:sgws:s3:::examplebucket",
        "urn:sgws:s3:::examplebucket/*"
      ]
    }
  ]
}
```

Ejemplo: Permitir a todos acceso de lectura y escritura a un bucket si el cliente está en el rango de IP.

En este ejemplo, todos, incluyendo la cuenta anónima, pueden Listar el bucket y realizar operaciones Object sobre todos los objetos contenidos en el bucket, suponiendo que las solicitudes provienen desde un rango de IP especificadas (54.240.143.0 a 54.240.143.255, salvo 54.240.143.188). Todas las demás operaciones se denegarán y todas las solicitudes que provengan de direcciones IP que no pertenezcan al rango se denegarán.

Nota: La palabra clave ‘Condition’ solo se permite en la Interfaz de Gestión del tenant (Tenant Management Interface) (versión 10.4 de StorageGRID Webscale o superior).

```
{
  "Statement": [
    {
      "Sid": "AllowEveryoneReadWriteAccessIfInSourceIpRange",
      "Effect": "Allow",
      "Principal": "*",
      "Action": [ "s3:*Object", "s3:ListBucket" ],
      "Resource": ["urn:sgws:s3:::examplebucket","urn:sgws:s3:::examplebucket/*"],
      "Condition": {
        "IpAddress": {"sgws:SourceIp": "54.240.143.0/24"},
        "NotIpAddress": {"sgws:SourceIp": "54.240.143.188"}
      }
    }
  ]
}
```

Ejemplo: permite acceso total a un bucket exclusivamente a un usuario federado especificado.

En este ejemplo, el usuario federado llamado Bob tiene garantizado el acceso total al bucket `examplebucket` y a sus objetos. Todos los demás usuarios, incluyendo 'root', tienen expresamente denegadas todas las operaciones. Observe, sin embargo, que a 'root' nunca se le negarán los permisos para Put/Get/DeleteBucketPolicy.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "SGWS": "urn:sgws:identity::95390887230002558202:federated-user/Bob"
      },
      "Action": [
        "s3:*"
      ],
      "Resource": [
        "urn:sgws:s3:::examplebucket",
        "urn:sgws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Deny",
      "NotPrincipal": {
        "SGWS": "urn:sgws:identity::95390887230002558202:federated-user/Bob"
      },
      "Action": [
        "s3:*"
      ],
      "Resource": [
        "urn:sgws:s3:::examplebucket",
        "urn:sgws:s3:::examplebucket/*"
      ]
    }
  ]
}
```

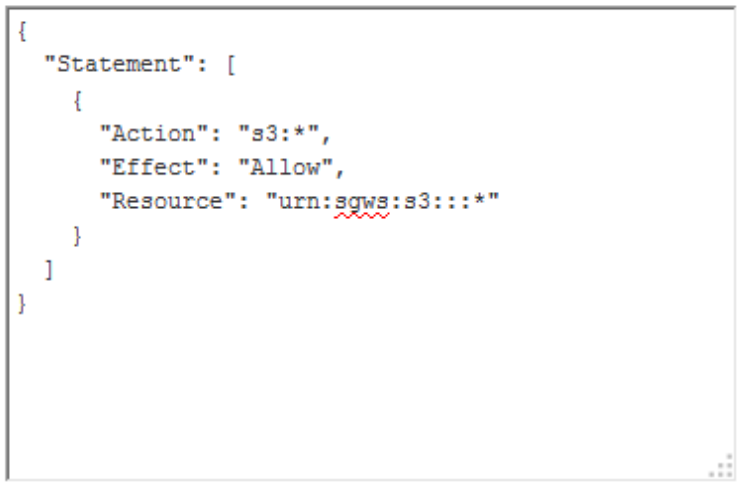
Ejemplos de política de grupo

Las políticas de grupo especifican los permisos de acceso para el grupo al que está asociada la política. No existe elemento `Principal` en la política ya que se encuentra implícita. Las políticas de grupo que se configuran utilizando la Interfaz de Administración del tenant o la API.

Ejemplo: Configuración de la política de grupo utilizando la interfaz de Administración del tenant.

Cuando utilice la Interfaz de Administración del tenant para añadir o editar un grupo, podrá usar el cuadro de diálogo **S3 Policy** para crear y actualizar las políticas de grupo utilizando cadenas JSON válidas:

S3 Policy



```
{
  "Statement": [
    {
      "Action": "s3:*",
      "Effect": "Allow",
      "Resource": "urn:sgws:s3:::*"
    }
  ]
}
```

Ejemplo: Permitir acceso total a un grupo para todos los buckets

En este ejemplo, todos los miembros del grupo tendrán garantizado el acceso total a todos los buckets que sean propiedad de la cuenta tenant salvo que se niegue explícitamente en la política del bucket.

```
{
  "Statement": [
    {
      "Action": "s3:*",
      "Effect": "Allow",
      "Resource": "urn:sgws:s3:::*"
    }
  ]
}
```

Ejemplo: Permitir acceso de solo lectura a un grupo para todos los buckets

En este ejemplo, todos los miembros del grupo tendrán garantizado el acceso de solo lectura a todos los buckets salvo que se niegue explícitamente en la política del bucket. Se permitirá el acceso a los buckets pertenecientes a esta cuenta, a menos que la política del bucket de destino lo niegue explícitamente.

```
{
  "Statement": [
    {
      "Sid": "AllowGroupReadOnlyAccess",
      "Effect": "Allow",
      "Action": [ "s3:ListAllMyBuckets", "s3:ListBucket", "s3:GetObject" ],
      "Resource": "urn:sgws:s3:::*"
    }
  ]
}
```

Ejemplo: permitir acceso total a los miembros del grupo solo al "folder" de su bucket.

En este ejemplo, los miembros del grupo solo pueden listar y acceder a su carpeta específica (prefijo de clave) en el bucket especificado. Tenga en cuenta que deberá tener en cuenta los permisos de acceso desde otras políticas de grupo y la política del bucket cuando determine la privacidad de estas carpetas.

Nota: La palabra clave ‘Condition’ y la variable `sgws:username` solo se pueden utilizar en la Interfaz de Administración del tenant (StorageGRID Webscale versión 10.4 o posterior).

```
{
  "Statement": [
    {
      "Sid": "AllowListBucketOfASpecificUserPrefix",
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": "urn:sgws:s3:::department_bucket",
      "Condition": {
        "StringLike": {
          "s3:prefix": "${sgws:username}/*"
        }
      }
    },
    {
      "Sid": "AllowUserSpecificActionsOnlyInTheSpecificUserPrefix",
      "Effect": "Allow",
      "Action": "s3:*Object",
      "Resource": "urn:sgws:s3:::department_bucket/${sgws:username}/*"
    }
  ]
```

```
}
```

Ejemplo: permiso PutOverwriteObject

En este ejemplo, el efecto `Deny` para PutOverwriteObject y DeleteObject protege los datos y metadatos del objeto de su borrado o modificación.

Si desea obtener más información, consulte Empleo del permiso PutOverwriteObject en la página 55 y la protección Escrito- una vez -leído-muchas (WORM) en la página 65.

```
{
"Sid": "WORMExamplePolicy",
"Effect": "Deny",
"Action": ["s3:PutOverwriteObject", "s3:DeleteObject"],
"Resource": ["urn:sgws:s3:::*"],
}
```

Cómo utilizan los certificados las aplicaciones cliente para la seguridad con las REST APIs

Cuando una aplicación cliente establece una sesión TLS en el sistema StorageGRID Webscale, el sistema envía un certificado de servidor a la aplicación cliente para su verificación con el fin de garantizar que la conexión HTTPS sea segura.

La aplicación del cliente carga el certificado CA de la malla y lo utiliza para verificar que la aplicación cliente se está comunicando con el sistema StorageGRID Webscale esperado. Este proceso protege contra los ataques del tipo hombre-en-el-medio y suplantación.

Algoritmos de hash y cifrado utilizados con las librerías TLS

Las aplicaciones cliente usan el protocolo HTTPS para comunicarse con el sistema StorageGRID Webscale a través de una conexión de red que usa Transport Layer Security (TLS, Seguridad de la Capa de Transporte). El sistema StorageGRID Webscale admite un conjunto limitado de algoritmos de cifrado y hash de las bibliotecas TLS que las aplicaciones cliente pueden usar al establecer una sesión TLS. Cuando configure los procesos de comunicación, es importante que sepa qué algoritmos de seguridad utiliza el sistema.

El sistema StorageGRID Webscale admite los siguientes algoritmos de seguridad de la suite de cifrado:

TLS Versión	Suite de cifrado	Beneficio
v1.0	TLS_RSA_WITH_AES_128_CBC_SHA	Proporciona cifrado seguro y procesamiento eficiente de los objetos.
	TLS_RSA_WITH_AES_256_CBC_SHA	
v1.1	TLS_RSA_WITH_AES_128_CBC_SHA	
	TLS_RSA_WITH_AES_256_CBC_SHA	
v1.2	TLS_RSA_WITH_AES_128_CBC_SHA	
	TLS_RSA_WITH_AES_256_CBC_SHA	
	TLS_RSA_WITH_AES_128_GCM_SHA256	Proporciona un cifrado seguro y un procesamiento más eficiente de los objetos de gran tamaño.
	TLS_RSA_WITH_AES_256_GCM_SHA384	
	TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	Compatible con el secreto hacia delante perfecto.
	TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	
	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	
	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	

La sesión TLS negocia la conexión, utilizando AES128 o AES256 según los requisitos de la aplicación cliente y la necesidad de equilibrar el rendimiento con la seguridad del cifrado.

Atención: SSLv3 ya no es compatible con las conexiones para el CLB o LDR.

Protección Write-once-read-many (WORM - Una escritura Muchas lecturas)

Puede crear buckets del tipo Una escritura Muchas lecturas (WORM) para proteger datos y metadatos. Configure los buckets WORM para permitir la creación de nuevos objetos y evitar sobrescrituras o borrados del contenido existente. Use uno de los enfoques que se describen a continuación.

Para asegurarse de que nunca se va a permitir una sobrescritura, puede:

- Desde la Interfaz de gestión, ajuste la opción global Prevent Client Modify (Evitar Modificación del Cliente) a **Enabled** (Habilitado).
- Aplique las siguientes reglas y políticas S3
  - Añada una operación PutOverwriteObject DENY a la política S3.
  - Añada una operación DeleteObject DENY a la política S3.
  - Añada una operación PUT Object ALLOW a la política S3.

Atención: Asignar el valor DENY a DeleteObject en una política S3 no impide a ILM eliminar objetos cuando exista una regla del tipo "cero copias tras 30 días". Consulte la Guía del Administrador para obtener más información.

Atención: incluso cuando se apliquen todas estas reglas y políticas, no estará protegido frente a escrituras simultáneas (consulte la Situación A). Protegen contra sobrescrituras completadas de forma secuencial (ver Situación B).

Situación A — Escrituras simultáneas (no protege contra ellas)

```
/mybucket/important.doc
PUT#1 ---> OK
PUT#2 -------> OK
```

Situación B — Sobrescrituras completadas de forma secuencial (protege contra ellas)

```
/mybucket/important.doc
PUT#1 -------> PUT#2 ---X (denied)
```

Si desea ver un ejemplo del empleo del permiso PutOverwriteObject, consulte Ejemplo: permiso PutOverwriteObject en la página 64.

Conceptos relacionados

Políticas de acceso al grupo y bucket en la página 50
En qué forma las reglas de StorageGRID Webscale ILM gestionan los objetos en la página 12
Ejemplos de política en la página 60

Supervisión y auditoría de operaciones

Puede supervisar la salud de las conexiones de su aplicación cliente con el sistema StorageGRID Webscale viendo atributos de resumen que enumeran recuentos de transacciones para los servicios LDR en todos los Nodos de Almacenamiento, o puede ver las transacciones para un Nodo de Almacenamiento específico. Además, puede usar mensajes de auditoría para supervisar las operaciones y transacciones del sistema StorageGRID Webscale.

Cómo ver las transacciones para los objetos S3

Puede ver el número de intentos fallidos y exitosos realizados por las aplicaciones cliente para leer, escribir y modificar objetos S3 en el sistema StorageGRID Webscale. Puede ver un resumen de todas las transacciones para todos los servicios LDR, o puede ver las transacciones para un servicio LDR específico. Es posible que desee hacer esto para evaluar el estado del sistema.

Pasos

1. Debe iniciar sesión en la Interfaz de gestión utilizando un navegador compatible.
2. Seleccione **Grid** (Malla).
3. Seleccione ***site*** > **Overview** > **Main**, y luego vea la zona de **Operaciones de la API**.

   El área de Operaciones de la API muestra un resumen de la información de todos los servicios de LDR que admiten aplicaciones cliente de S3.
4. Seleccione ***Storage Node*** > **LDR** > **S3** > **Overview** > **Main** para ver la información asociada con los servicios LDR individuales.

Acceso y revisión de los registros de auditoría

El sistema StorageGRID Webscale transporta de forma segura y fiable los mensajes de auditoría desde cada servicio dentro del sistema StorageGRID Webscale a uno o más repositorios de auditoría. Los mensajes de auditoría específicos de la API proporcionan datos críticos de seguridad, de operaciones y de supervisión del rendimiento que pueden ayudarle a evaluar el estado de su sistema.

Acerca de esta tarea

El sistema StorageGRID Webscale comprime los registros de auditoría al pasar un día y los renombra utilizando el formato `YYYY-MM-DD.txt.gz` (conservando siempre la fecha original).

Pasos

1. Inicie sesión en el servidor utilizando el nombre de usuario y la contraseña tal y como se guarda en el archivo `Passwords.txt`.
2. Acceda al directorio de registro de auditoría a través de una línea de comandos del servidor que aloja el servicio AMS.
3. Vaya al directorio `/var/local/audit/export/`.
4. Vea el contenido del archivo`audit.log`.

Información relacionada

Referencia del mensaje de auditoría StorageGRID Webscale 11.0

Beneficios de las conexiones HTTP activas, inactivas y concurrentes

La forma de configurar las conexiones HTTP puede afectar al rendimiento del sistema StorageGRID Webscale. Las configuraciones difieren dependiendo de si la conexión HTTP está activa o inactiva o si tiene conexiones múltiples concurrentes.

Beneficios de los diferentes tipos de conexiones HTTP

El tipo de duración de la conexión HTTP puede afectar al rendimiento del sistema StorageGRID Webscale.

Puede identificar los beneficios sobre el rendimiento para los siguientes tipos de conexiones HTTP:

- Conexiones HTTP inactivas
- Conexiones HTTP activas
- Conexiones HTTP concurrentes

Beneficios por mantener abiertas las conexiones HTTP inactivas

Puede mantener las conexiones HTTP abiertas incluso cuando las aplicaciones cliente estén inactivas para permitir que las aplicaciones del cliente realicen transacciones a través de la conexión abierta. En base a las mediciones del sistema y a la experiencia de integración, debe mantener una conexión HTTP abierta durante un máximo de 10 minutos. El servicio LDR podría cerrar automáticamente una conexión HTTP que se mantenga abierta e inactiva durante más de 10 minutos.

Abrir y mantener abierta una conexión HTTP inactiva brinda los siguientes beneficios:

- Menor latencia desde el momento en que el sistema StorageGRID Webscale determina que debe realizar una transacción HTTP hasta el momento en que el sistema StorageGRID Webscale puede realizar la transacción

  Una menor latencia es la principal ventaja, especialmente por la cantidad de tiempo requerido para establecer las conexiones TCP / IP y TLS.
- Mayor velocidad de transferencia de datos al cebar el algoritmo de arranque lento TCP / IP con transferencias realizadas anteriormente
- Notificación instantánea de varias clases de condiciones de fallo que interrumpen la conectividad entre la aplicación cliente y el sistema StorageGRID Webscale

Determinar cuánto tiempo hay que mantener abierta una conexión inactiva es una contrapartida entre los beneficios de un arranque lento asociado con la conexión existente y la adaptación ideal de la conexión a los recursos internos del sistema.

Beneficios de las conexiones HTTP activas

Debe limitar la duración de una conexión HTTP activa durante un máximo de 10 minutos, incluso aunque la conexión HTTP realice continuamente transacciones. Determinar la duración máxima que una conexión debe mantenerse abierta es una decisión que debe ponderar los beneficios de la persistencia de la conexión y la asignación ideal de la conexión a los recursos internos del sistema.

Las conexiones HTTP activas de duración limitada proporcionan los siguientes beneficios:

- Permite un equilibrio de carga óptimo en el sistema StorageGRID Webscale

Para optimizar el equilibrio de carga en el sistema StorageGRID Webscale, debe evitar las conexiones TCP / IP de larga - duración. Debe configurar las aplicaciones cliente para rastrear la duración de cada conexión HTTP y cerrar la conexión HTTP después de un tiempo definido para que la conexión HTTP pueda ser reestablecida y reequilibrada.

El sistema StorageGRID Webscale equilibra su carga cuando una aplicación cliente establece una conexión HTTP. Con el tiempo, una conexión HTTP que el sistema StorageGRID Webscale utiliza para un recurso informático puede no ser óptima a medida que cambien los requisitos de equilibrio de carga. El sistema realiza su mejor equilibrio de carga cuando las aplicaciones cliente establecen una conexión HTTP independiente para cada transacción, pero esto anula las ganancias mucho más valiosas asociadas con las conexiones persistentes.

- Permite que los procedimientos de mantenimiento comiencen

  Algunos procedimientos de mantenimiento comienzan solo después de que se completen todas las conexiones HTTP en curso.

- Permite que las aplicaciones cliente dirijan las transacciones HTTP a los servicios LDR que tienen espacio disponible.

Beneficios de las conexiones HTTP concurrentes

Debe mantener abiertas varias conexiones TCP / IP con el sistema StorageGRID Webscale para permitir que las conexiones inactivas realicen las transacciones necesarias. El número de aplicaciones cliente también afecta la forma en que usted maneja conexiones TCP / IP múltiples.

Las conexiones HTTP concurrentes proporcionan los siguientes beneficios:

- Latencia reducida

  Las transacciones pueden comenzar inmediatamente en lugar de esperar a que se completen otras transacciones.

- Mayor rendimiento

  El sistema StorageGRID Webscale puede realizar transacciones paralelas y aumentar el rendimiento total de las transacciones.

Las aplicaciones cliente deben establecer múltiples conexiones HTTP, ya sea cliente por cliente o en base a un conjunto de conexiones. Cuando una aplicación cliente tiene que realizar una transacción, puede seleccionar e inmediatamente usar cualquier conexión establecida que no esté procesando actualmente una transacción.

La topología de cada sistema StorageGRID Webscale tiene un rendimiento pico diferente para las transacciones y conexiones concurrentes antes de que el rendimiento comience a degradarse. El rendimiento pico depende de factores tales como los recursos informáticos, los recursos de red, los recursos de almacenamiento y los enlaces WAN. La cantidad de servidores y servicios y la cantidad de aplicaciones que admite el sistema StorageGRID Webscale también son factores a considerar.

Los sistemas StorageGRID Webscale a menudo admiten múltiples aplicaciones cliente. Debe tener esto en cuenta cuando determine el número máximo de conexiones simultáneas utilizadas por una aplicación cliente. Si la aplicación cliente consiste en varias entidades de software que establecen conexiones con el sistema StorageGRID Webscale, debe agregar todas las conexiones entre las entidades. Es posible que deba ajustar la cantidad máxima de conexiones simultáneas en las siguientes situaciones:

- La topología del sistema StorageGRID Webscale afecta a la cantidad máxima de transacciones y conexiones simultáneas que el sistema puede admitir.

- Las aplicaciones cliente que interactúan con el sistema StorageGRID Webscale en una red con ancho de banda limitado pueden tener que reducir el grado de concurrencia para garantizar que las transacciones individuales se completan en un tiempo razonable.

- Cuando muchas aplicaciones cliente compartan el sistema StorageGRID Webscale, es posible que tenga que reducir el grado de concurrencia para evitar exceder los límites del sistema.

Separación de los grupos de conexiones HTTP para las operaciones de lectura y escritura

Puede usar grupos separados de conexiones HTTP para operaciones de lectura y escritura y controlar qué cantidad de un grupo tiene que utilizar para cada una de ellas. Los grupos independientes de conexiones HTTP le permiten controlar mejor las transacciones y equilibrar las cargas.

Las aplicaciones cliente pueden crear cargas que tengan predominio de tareas de recuperación (lectura) o de almacenamiento (escritura). Con grupos independientes de conexiones HTTP para las transacciones de lectura y escritura, puede ajustar la cantidad de cada grupo que va a dedicar a las transacciones de lectura o escritura.

Información del copyright

Información de marca comercial

Cómo enviar comentarios sobre la documentación y recibir notificaciones de actualización

Puede ayudarnos a mejorar la calidad de nuestra documentación enviándonos sus comentarios. Puede recibir notificaciones automáticas cuando se libere inicialmente la documentación del nivel de producción (GA / FCS) o cuando se realicen cambios importantes en los documentos de nivel de producción existentes.

Si tiene sugerencias para mejorar este documento, envíenos sus comentarios por correo electrónico.

doccomments@netapp.com

Para ayudarnos a dirigir sus comentarios al departamento correcto, incluya en el asunto el nombre del producto, la versión y el sistema operativo.

Si desea que se le notifique automáticamente cuando se edite documentación del nivel de producción o cuando se realicen cambios importantes en los documentos de nivel de producción existentes, siga la cuenta de Twitter @NetAppDoc.

También puede ponerse en contacto con nosotros de las siguientes maneras:

- NetApp, Inc., 495 East Java Drive, Sunnyvale, CA 94089 U.S.
- Teléfono: +1 (408) 822-6000
- Fax: +1 (408) 822-4501
- Teléfono de soporte: +1 (888) 463-8277

Índice

D

E

F

G

H

I

L

M

O

P

Q

R

S

T

V

X