# Use S3

StorageGRID

NetApp
October 03, 2025

# Tabla de contenidos

# Use S3

## Utilice S3: Descripción general

StorageGRID admite la API de simple Storage Service (S3), que se implementa como un conjunto de servicios web de transferencia de estado de representación (REST). La compatibilidad con la API REST de S3 permite conectar aplicaciones orientadas a los servicios desarrolladas para los servicios web S3 con un almacenamiento de objetos en las instalaciones que usa el sistema StorageGRID. Esto requiere cambios mínimos en el uso actual de llamadas API DE REST de S3 por parte de una aplicación cliente.

### Cambios en la compatibilidad con la API DE REST de S3

Debe estar al tanto de los cambios en la compatibilidad del sistema StorageGRID con la API DE REST de S3.

| Liberar | Comentarios |
|---|---|
| 11.6 | • Se ha agregado soporte para utilizar `partNumber` Parámetro de solicitud en GET Object y HEAD Object peticiones.<br>• Se añadió compatibilidad con un modo de retención predeterminado y un período de retención predeterminado en el nivel de bloque para S3 Object Lock.<br>• Se ha añadido compatibilidad con `s3:object-lock-remaining-retention-days` clave de condición de política para configurar el rango de períodos de retención permitidos para los objetos.<br>• El tamaño máximo *recommended* para una única operación PUT Object es ahora de 5 GIB (5,368,709,120 bytes). Si tiene objetos que sean mayores de 5 GIB, utilice la carga de varias partes en su lugar.<br>ⓘ En StorageGRID 11.6, el tamaño máximo *admitido* para una única operación PUT Object sigue siendo de 5 TIB (5,497,558,138,880 bytes). Sin embargo, la alerta * S3 PUT Object size demasiado grande* se activará si intenta cargar un objeto que supere los 5 GIB. |
| 11.5 | • Se ha agregado compatibilidad para gestionar el cifrado de bloques.<br>• Se añadió compatibilidad con el bloqueo de objetos S3 y las solicitudes de cumplimiento heredadas obsoletas.<br>• Se ha agregado soporte para el uso DE DELETE Multiple Objects en cubos con versiones.<br>• La `Content-MD5` el encabezado de la solicitud ahora es correctamente compatible. |

| Liberar | Comentarios |
| --- | --- |
| 11.4 | • Se añadió compatibilidad con el etiquetado DE bloques DE DELETE, GET Bucket y PUT Bucket. No se admiten etiquetas de asignación de costes.<br>• En el caso de bloques creados en StorageGRID 11.4, ya no es necesario restringir los nombres de claves de objetos para cumplir con las prácticas recomendadas de rendimiento.<br>• Se ha agregado compatibilidad con las notificaciones de bloques en la `s3:ObjectRestore:Post` tipo de evento.<br>• Ahora se aplican los límites de tamaño de AWS para piezas multiparte. Cada parte de una carga de varias partes debe tener entre 5 MIB y 5 GIB. La última parte puede ser menor que 5 MIB.<br>• Se ha agregado compatibilidad con TLS 1.3 y se ha actualizado la lista de conjuntos de cifrado TLS compatibles.<br>• El servicio CLB está obsoleto. |
| 11.3 | • Se ha añadido compatibilidad con el cifrado en el servidor de los datos de objetos con las claves proporcionadas por el cliente (SSE-C).<br>• Se ha añadido compatibilidad para operaciones DE ELIMINACIÓN, GET y PUT Bucket Lifecycle (solo acción de caducidad) y para el `x-amz-expiration` encabezado de respuesta.<br>• Se han actualizado PUT Object, PUT Object - Copy y Multipart Upload para describir el impacto de las reglas de ILM que utilizan la colocación síncrona en el procesamiento.<br>• Lista actualizada de conjuntos de cifrado TLS admitidos. Ya no se admiten los cifrados TLS 1.1. |
| 11.2 | Compatibilidad añadida para la restauración DE objetos POSTERIOR para uso con pools de almacenamiento en cloud. Se añadió compatibilidad con el uso de la sintaxis AWS para ARN, claves de condición de política y variables de política en políticas de grupos y bloques. Se seguirán soportando las políticas de grupo y bloque existentes que utilicen la sintaxis StorageGRID.<br><br>**Nota:** los usos de ARN/URN en otra configuración JSON/XML, incluidos los utilizados en las características personalizadas de StorageGRID, no han cambiado. |
| 11.1 | Se ha agregado soporte para uso compartido de recursos de origen cruzado (CORS), conexiones de clientes HTTP para S3 a nodos de grid y configuración de cumplimiento en bloques. |
| 11.0 | Se añadió compatibilidad para configurar servicios de plataforma (replicación de CloudMirror, notificaciones e integración de búsqueda de Elasticsearch) para los bloques. También se añadió compatibilidad con las restricciones de ubicación del etiquetado de objetos para bloques y la configuración de control de coherencia disponible. |

| Liberar | Comentarios |
| --- | --- |
| 10.4 | Se ha agregado compatibilidad con los cambios de análisis de ILM en las versiones, las actualizaciones de página de nombres de dominio de extremo, las condiciones y variables en las directivas, los ejemplos de directivas y el permiso PutOverwriteObject. |
| 10.3 | Se ha añadido compatibilidad con las versiones. |
| 10.2 | Se ha añadido compatibilidad con las políticas de acceso a grupos y bloques y para la copia de varias partes (cargar artículo - copia). |
| 10.1 | Se añadió compatibilidad con la carga de varias partes, las solicitudes de estilo hospedado virtual y la autenticación v4. |
| 10.0 | Soporte inicial de la API DE REST de S3 por parte del sistema StorageGRID.la versión actualmente admitida de *simple Storage Service API Reference* es 2006-03-01. |

## Versiones compatibles

StorageGRID admite las siguientes versiones específicas de S3 y HTTP.

| Elemento | Versión |
| --- | --- |
| Especificación de S3 | *Simple Storage Service referencia de API* 2006-03-01 |
| HTTP | 1.1<br><br>Para obtener más información acerca de HTTP, vea HTTP/1.1 (RFC 7230-35).<br><br>**Nota**: StorageGRID no admite canalización HTTP/1.1. |

**Información relacionada**

"RFC de IETF 2616: Protocolo de transferencia de hipertexto (HTTP/1.1)"

"Documentación de Amazon Web Services (AWS): Referencia de API de Amazon simple Storage Service"

## Soporte para servicios de plataforma StorageGRID

Los servicios de plataforma StorageGRID permiten que las cuentas de inquilinos StorageGRID aprovechen servicios externos, como un bloque de S3 remoto, un extremo de servicio de notificación simple (SNS) o un clúster de Elasticsearch para ampliar los servicios que ofrece un grid.

La tabla siguiente resume los servicios de plataforma disponibles y las API S3 que se utilizan para configurarlos.

| Servicio de plataforma | Específico | API de S3 que se utiliza para configurar el servicio |
|---|---|---|
| Replicación de CloudMirror | Replica objetos de un bloque StorageGRID de origen en el bloque S3 remoto configurado. | PUT Bucket replication |
| Notificaciones | Envía notificaciones acerca de eventos en un bloque de StorageGRID de origen a un extremo de servicio simple de notificación (SNS) configurado. | NOTIFICACIÓN DE PUT Bucket |
| Integración de búsqueda | Envía metadatos de objetos para los objetos almacenados en un bloque de StorageGRID a un índice de Elasticsearch configurado. | PUT bucket metadata notification<br><br>**Nota:** se trata de una API StorageGRID S3 personalizada. |

Un administrador de grid debe habilitar el uso de los servicios de plataforma para una cuenta de inquilino antes de poder utilizarlos. A continuación, un administrador de arrendatarios debe crear un extremo que represente el servicio remoto en la cuenta de arrendatario. Este paso es necesario para poder configurar un servicio.

### Recomendaciones para el uso de servicios de plataformas

Antes de utilizar los servicios de plataforma, debe tener en cuenta las siguientes recomendaciones:

- NetApp recomienda que no permita más de 100 inquilinos activos con solicitudes S3 que requieran la replicación, las notificaciones y la integración de búsqueda de CloudMirror. Tener más de 100 inquilinos activos puede dar como resultado un rendimiento del cliente S3 más lento.
- Si un bloque de S3 en el sistema StorageGRID tiene habilitadas las versiones y la replicación de CloudMirror, NetApp recomienda que el extremo de destino también tenga habilitada el control de versiones de bloque de S3. Esto permite que la replicación de CloudMirror genere versiones de objetos similares en el extremo.
- La replicación de CloudMirror no es compatible si el bloque de origen tiene la función S3 Object Lock habilitada.
- La replicación de CloudMirror generará un error ACCESSDENIED si el bloque de destino tiene activada la conformidad heredada.

**Información relacionada**

Usar cuenta de inquilino

Administre StorageGRID

Operaciones en bloques

PUT bucket metadata notification Configuration

# Configure las conexiones y las cuentas de inquilino

Para configurar StorageGRID para aceptar conexiones desde aplicaciones cliente, es necesario crear una o más cuentas de inquilino y configurar las conexiones.

## Cree y configure cuentas de inquilino de S3

Se requiere una cuenta de inquilino de S3 para que los clientes de la API de S3 puedan almacenar y recuperar objetos en StorageGRID. Cada cuenta de inquilino tiene su propio ID de cuenta, grupos y usuarios, y contenedores y objetos.

Las cuentas de inquilino S3 las crea un administrador de grid de StorageGRID mediante Grid Manager o la API de gestión de grid. Al crear una cuenta de inquilino de S3, el administrador de grid especifica la siguiente información:

- Nombre para mostrar del arrendatario (el ID de cuenta del arrendatario se asigna automáticamente y no se puede modificar).
- Si la cuenta de inquilino tiene permiso para utilizar los servicios de plataforma. Si se permite el uso de servicios de plataforma, la cuadrícula debe configurarse para que admita su uso.
- Opcionalmente, una cuota de almacenamiento para la cuenta de inquilino: El número máximo de gigabytes, terabytes o petabytes disponibles para los objetos del inquilino. La cuota de almacenamiento de un inquilino representa una cantidad lógica (tamaño de objeto), no una cantidad física (tamaño en disco).
- Si está habilitada la federación de identidades para el sistema StorageGRID, el grupo federado tiene permiso de acceso raíz para configurar la cuenta de inquilino.
- Si el sistema StorageGRID no utiliza el inicio de sesión único (SSO), tanto si la cuenta de inquilino usará su propio origen de identidad como si comparte el origen de identidad de la cuadrícula, así como la contraseña inicial del usuario raíz local del inquilino.

Una vez creada una cuenta de inquilino de S3, los usuarios de inquilinos pueden acceder al administrador de inquilinos para realizar tareas como las siguientes:

- Configure la federación de identidades (a menos que el origen de identidades se comparta con la cuadrícula) y cree grupos y usuarios locales
- Gestión de claves de acceso de S3
- Cree y gestione bloques de S3, incluidos los bloques con el bloqueo de objetos S3 habilitado
- Utilizar servicios de plataforma (si están habilitados)
- Supervise el uso del almacenamiento

> Los usuarios inquilinos S3 pueden crear y gestionar bloques de S3 con el administrador de inquilinos, pero deben tener claves de acceso S3 y usar la API REST de S3 para procesar y gestionar objetos.

**Información relacionada**

Administre StorageGRID

Usar cuenta de inquilino

## Cómo se pueden configurar las conexiones de clientes

Un administrador de grid toma opciones de configuración que afectan a la forma en que los clientes S3 se conectan a StorageGRID para almacenar y recuperar datos. La información específica que necesita para realizar una conexión depende de la configuración elegida.

Las aplicaciones cliente pueden almacenar o recuperar objetos conectándose a cualquiera de los siguientes elementos:

- El servicio Load Balancer en los nodos de administrador o de puerta de enlace, o bien, de forma opcional, la dirección IP virtual de un grupo de alta disponibilidad (ha) de nodos de administración o nodos de puerta de enlace
- El servicio CLB en los nodos de puerta de enlace o, opcionalmente, la dirección IP virtual de un grupo de nodos de puerta de enlace de alta disponibilidad

> El servicio CLB está obsoleto. Los clientes configurados antes de la versión StorageGRID 11.3 pueden seguir utilizando el servicio CLB en los nodos de puerta de enlace. El resto de aplicaciones cliente que dependen de StorageGRID para proporcionar equilibrio de carga se deben conectar mediante el servicio Load Balancer.

- Nodos de almacenamiento, con o sin un equilibrador de carga externo

Al configurar StorageGRID, un administrador de grid puede utilizar Grid Manager o la API de gestión de grid para realizar los siguientes pasos, todos ellos opcionales:

1. Configure los extremos para el servicio Load Balancer.

   Debe configurar los extremos para usar el servicio Load Balancer. El servicio Load Balancer en los nodos de administrador o de puerta de enlace distribuye conexiones de red entrantes desde aplicaciones cliente hasta los nodos de almacenamiento. Al crear un extremo de equilibrio de carga, el administrador de StorageGRID especifica un número de puerto, tanto si el extremo acepta conexiones HTTP o HTTPS, como el tipo de cliente (S3 o Swift) que utilizará el extremo y el certificado que se utilizará para las conexiones HTTPS (si procede).

2. Configure redes de cliente no fiables.

   Si un administrador de StorageGRID configura la red cliente de un nodo para que no sea de confianza, el nodo sólo acepta conexiones entrantes en la red cliente en puertos que se configuran explícitamente como extremos equilibradores de carga.

3. Configuración de grupos de alta disponibilidad.

   Si un administrador crea un grupo de alta disponibilidad, las interfaces de red de varios nodos de administrador o nodos de puerta de enlace se colocan en una configuración de backup activo. Las conexiones de clientes se realizan mediante la dirección IP virtual del grupo de alta disponibilidad.

Para obtener más información acerca de cada opción, consulte las instrucciones para administrar StorageGRID.

**Información relacionada**

Administre StorageGRID

## Resumen: Direcciones IP y puertos para conexiones cliente

Las aplicaciones cliente se conectan a StorageGRID mediante la dirección IP de un nodo de grid y el número de puerto de un servicio en ese nodo. Si se configuran los grupos de alta disponibilidad, las aplicaciones cliente se pueden conectar mediante la dirección IP virtual del grupo de alta disponibilidad.

#### Información necesaria para realizar conexiones de cliente

La tabla resume las distintas maneras en que los clientes pueden conectarse a StorageGRID y las direcciones IP y los puertos que se utilizan para cada tipo de conexión. Póngase en contacto con el administrador de StorageGRID para obtener más información o consulte las instrucciones para administrar StorageGRID para obtener una descripción de cómo encontrar esta información en el administrador de grid.

| Dónde se realiza la conexión | Servicio al que se conecta el cliente | Dirección IP | Puerto |
|---|---|---|---|
| Grupo de ALTA DISPONIBILIDAD | Equilibrador de carga | La dirección IP virtual de un grupo de alta disponibilidad | • Puerto de punto final del equilibrador de carga |
| Grupo de ALTA DISPONIBILIDAD | CLB<br><br>**Nota:** el servicio CLB está en desuso. | La dirección IP virtual de un grupo de alta disponibilidad | Puertos S3 predeterminados:<br><br>• HTTPS: 8082<br>• HTTP: 8084 |
| Nodo de administración | Equilibrador de carga | La dirección IP del nodo de administrador | • Puerto de punto final del equilibrador de carga |
| Nodo de puerta de enlace | Equilibrador de carga | La dirección IP del nodo de puerta de enlace | • Puerto de punto final del equilibrador de carga |
| Nodo de puerta de enlace | CLB<br><br>**Nota:** el servicio CLB está en desuso. | La dirección IP del nodo de puerta de enlace<br><br>**Nota:** de forma predeterminada, los puertos HTTP para CLB y LDR no están habilitados. | Puertos S3 predeterminados:<br><br>• HTTPS: 8082<br>• HTTP: 8084 |
| Nodo de almacenamiento | LDR | La dirección IP del nodo de almacenamiento | Puertos S3 predeterminados:<br><br>• HTTPS: 18082<br>• HTTP: 18084 |

#### Ejemplo

Para conectar un cliente S3 al extremo de equilibrio de carga de un grupo ha de nodos de puerta de enlace,

utilice una URL estructurada como se muestra a continuación:

- `https://VIP-of-HA-group:_LB-endpoint-port_`

Por ejemplo, si la dirección IP virtual del grupo de alta disponibilidad es 192.0.2.5 y el número de puerto de un extremo de equilibrio de carga de S3 es 10443, un cliente de S3 puede usar la siguiente URL para conectarse a StorageGRID:

- `https://192.0.2.5:10443`

Es posible configurar un nombre DNS para la dirección IP que utilizan los clientes para conectarse a StorageGRID. Póngase en contacto con el administrador de red local.

**Información relacionada**

Administre StorageGRID

### Decidir usar conexiones HTTPS o HTTP

Cuando se realizan conexiones de cliente mediante un extremo de equilibrio de carga, es necesario realizar conexiones mediante el protocolo (HTTP o HTTPS) especificado para ese extremo. Para utilizar HTTP para las conexiones de clientes a nodos de almacenamiento o al servicio CLB en nodos de puerta de enlace, debe habilitar su uso.

De forma predeterminada, cuando las aplicaciones cliente se conectan a nodos de almacenamiento o al servicio CLB en nodos de puerta de enlace, deben utilizar HTTPS cifrado para todas las conexiones. Opcionalmente, puede habilitar conexiones HTTP menos seguras seleccionando la opción de cuadrícula **Activar conexión HTTP** en el Administrador de grid. Por ejemplo, una aplicación cliente puede utilizar HTTP al probar la conexión a un nodo de almacenamiento en un entorno no de producción.

> Tenga cuidado al habilitar HTTP para una cuadrícula de producción, ya que las solicitudes se enviarán sin cifrar.

> El servicio CLB está obsoleto.

Si se selecciona la opción **Activar conexión HTTP**, los clientes deben utilizar puertos diferentes para HTTP que los que utilizan para HTTPS. Consulte las instrucciones para administrar StorageGRID.

**Información relacionada**

Administre StorageGRID

Ventajas de las conexiones HTTP activas, inactivas y simultáneas

## Nombres de dominio extremo para solicitudes de S3

Para poder utilizar los nombres de dominio S3 para las solicitudes de cliente, un administrador de StorageGRID debe configurar el sistema para aceptar conexiones que usen nombres de dominio S3 en solicitudes de estilo de ruta de acceso S3 y de estilo virtual alojado S3.

**Acerca de esta tarea**

Para permitir utilizar solicitudes de estilo alojadas virtuales de S3, un administrador de grid debe realizar las siguientes tareas:

- Use Grid Manager para añadir los nombres de dominio de extremo S3 al sistema StorageGRID.

- Asegúrese de que el certificado que utiliza el cliente para las conexiones HTTPS a StorageGRID esté firmado para todos los nombres de dominio que el cliente necesita.

  Por ejemplo, si el extremo es `s3.company.com`, El administrador de grid debe asegurarse de que el certificado utilizado para las conexiones HTTPS incluye `s3.company.com` Nombre alternativo (SAN) del asunto comodín del extremo y del extremo: `*.s3.company.com`.

- Configure el servidor DNS utilizado por el cliente para incluir registros DNS que coincidan con los nombres de dominio de extremo, incluidos los registros comodín necesarios.

Si el cliente se conecta mediante el servicio Load Balancer, el certificado que el administrador de grid configura es el certificado para el extremo de equilibrio de carga que utiliza el cliente.

> Cada extremo de equilibrador de carga tiene su propio certificado y cada extremo se puede configurar para reconocer diferentes nombres de dominio de extremo.

Si el cliente se conecta a nodos de almacenamiento o al servicio CLB en nodos de puerta de enlace, el certificado que el administrador de grid configura es el único certificado de servidor personalizado utilizado para la cuadrícula.

> El servicio CLB está obsoleto.

Consulte las instrucciones para administrar StorageGRID si desea obtener más información.

Una vez completados estos pasos, puede utilizar solicitudes virtuales de estilo hospedado (por ejemplo, `bucket.s3.company.com`).

**Información relacionada**

Administre StorageGRID

Configure la seguridad de la API DE REST

## Probar configuración de la API DE REST de S3

Puede utilizar la interfaz de línea de comandos (CLI de AWS) de Amazon Web Services para probar la conexión al sistema y verificar que puede leer y escribir objetos en el sistema.

**Lo que necesitará**

- Ha descargado e instalado la CLI de AWS desde "aws.amazon.com/cli".
- Creó una cuenta de inquilino de S3 en el sistema StorageGRID.

**Pasos**

1. Configure los ajustes de Amazon Web Services para que utilicen la cuenta que creó en el sistema StorageGRID:
   a. Entrar al modo de configuración: `aws configure`
   b. Introduzca el ID de clave de acceso de AWS para la cuenta que creó.
   c. Introduzca la clave de acceso secreto de AWS para la cuenta que ha creado.
   d. Introduzca la región predeterminada que desea utilizar, por ejemplo, US-East-1.
   e. Introduzca el formato de salida predeterminado que se va a utilizar o pulse **Intro** para seleccionar JSON.

2. Crear un bucket.

```
aws s3api --endpoint-url https://10.96.101.17:10443
--no-verify-ssl create-bucket --bucket testbucket
```

Si el bloque se crea correctamente, se devuelve la ubicación del bloque, como se puede ver en el ejemplo siguiente:

```
"Location": "/testbucket"
```

1. Cargue un objeto.

```
aws s3api --endpoint-url https://10.96.101.17:10443 --no-verify-ssl
put-object --bucket testbucket --key s3.pdf --body C:\s3-
test\upload\s3.pdf
```

Si el objeto se carga correctamente, se devuelve un ETag que es un hash de los datos del objeto.

2. Enumere el contenido del cucharón para verificar que el objeto se ha cargado.

```
aws s3api --endpoint-url https://10.96.101.17:10443 --no-verify-ssl
list-objects --bucket testbucket
```

3. Elimine el objeto.

```
aws s3api --endpoint-url https://10.96.101.17:10443 --no-verify-ssl
delete-object --bucket testbucket --key s3.pdf
```

4. Eliminar el bloque.

```
aws s3api --endpoint-url https://10.96.101.17:10443 --no-verify-ssl
delete-bucket --bucket testbucket
```

# Cómo StorageGRID implementa la API DE REST de S3

Una aplicación cliente puede utilizar llamadas API DE REST de S3 para conectarse a StorageGRID y crear, eliminar y modificar bloques, así como almacenar y recuperar objetos.

## Solicitudes de clientes en conflicto

Las solicitudes de clientes en conflicto, como dos clientes que escriben en la misma clave, se resuelven en función de las "últimas victorias".

El plazo para la evaluación de "logros más recientes" se basa en cuándo el sistema StorageGRID completa una solicitud determinada, y no en cuándo los clientes de S3 comienzan una operación.

## Controles de consistencia

Los controles de consistencia proporcionan un equilibrio entre la disponibilidad de los objetos y la coherencia de dichos objetos en diferentes nodos de almacenamiento y sitios, según lo requiera la aplicación.

De forma predeterminada, StorageGRID garantiza la coherencia de lectura tras escritura de los objetos recién creados. Cualquier OBTENER después de un PUESTO completado correctamente podrá leer los datos recién escritos. Las sobrescrituras de objetos existentes, actualizaciones de metadatos y eliminaciones son coherentes en la actualidad. Por lo general, las sobrescrituras tardan segundos o minutos en propagarse, pero pueden tardar hasta 15 días.

Si desea realizar operaciones de objetos en un nivel de coherencia diferente, puede especificar un control de coherencia para cada bloque o para cada operación de API.

### Controles de consistencia

El control de consistencia afecta a cómo los metadatos que utiliza StorageGRID para realizar un seguimiento de los objetos se distribuyen entre los nodos y, por lo tanto, la disponibilidad de los objetos para las solicitudes del cliente.

Puede establecer el control de coherencia de un bloque o una operación API en uno de los siguientes valores:

- **Todos**: Todos los nodos reciben los datos inmediatamente, o la solicitud fallará.
- **Strong-global**: Garantiza la coherencia de lectura tras escritura para todas las solicitudes de clientes en todos los sitios.
- **Strong-site**: Garantiza la coherencia de lectura después de escritura para todas las solicitudes de cliente dentro de un sitio.
- **Read-after-new-write**: (Predeterminado) proporciona consistencia de lectura después de escritura para nuevos objetos y consistencia eventual para actualizaciones de objetos. Ofrece garantías de alta disponibilidad y protección de datos. Recomendado para la mayoría de los casos.
- **Disponible**: Proporciona consistencia eventual tanto para nuevos objetos como para actualizaciones de objetos. Para los depósitos S3, utilice sólo según sea necesario (por ejemplo, para un depósito que contiene valores de log que rara vez se leen, o para operaciones HEAD u GET en claves que no existen). No se admite para bloques de FabricPool S3.

### Utilizar controles de coherencia «reescritura tras escritura» y «disponibles»

Cuando una OPERACIÓN HEAD u GET utiliza el control de consistencia «relativamente una vez que una nueva escritura», StorageGRID realiza la búsqueda en varios pasos, como se indica a continuación:

- Primero busca el objeto con una baja consistencia.
- Si esa búsqueda falla, repite la búsqueda en el siguiente nivel de coherencia hasta que alcanza un nivel

de coherencia equivalente al comportamiento de la búsqueda global.

Si una operación HEAD u GET utiliza el control de consistencia “READ-after-new-write”, pero el objeto no existe, la búsqueda de objetos siempre alcanzará un nivel de consistencia equivalente al comportamiento de strong-global. Debido a que este nivel de coherencia requiere que haya varias copias de los metadatos del objeto disponibles en cada sitio, puede recibir un número elevado de errores del servidor interno 500 si uno o más nodos de almacenamiento del mismo sitio no están disponibles.

A menos que necesite garantías de coherencia similares a las de Amazon S3, podrá evitar estos errores en LAS operaciones DE CABEZA y OBTENER al establecer el control de coherencia en "'Available'". Cuando un CABEZAL o UNA operación DE OBTENER utiliza el control de consistencia "'disponible'", StorageGRID sólo proporciona consistencia eventual. No vuelve a intentar una operación fallida en los niveles de coherencia crecientes, por lo que no es necesario que haya disponibles varias copias de los metadatos de objeto.

#### Especifique el control de coherencia para la operación de API

Para configurar el control de coherencia para una operación de API individual, deben ser compatibles los controles de coherencia para la operación y debe especificar el control de coherencia en el encabezado de la solicitud. En este ejemplo se establece el control de coherencia en «punto de referencia» para una operación GET Object.

```
GET /bucket/object HTTP/1.1
Date: date
Authorization: authorization name
Host: host
Consistency-Control: strong-site
```

Debe usar el mismo control de coherencia para las operaciones PUT Object y GET Object.

#### Especifique el control de coherencia para el bloque

Para establecer el control de consistencia para el bloque, puede utilizar StorageGRID la solicitud de consistencia PUT Bucket y LA solicitud DE consistencia GET Bucket. También puede usar el Administrador de inquilinos o la API de gestión de inquilinos.

Cuando configure los controles de coherencia para un cucharón, tenga en cuenta lo siguiente:

- La configuración del control de coherencia para un bloque determina el control de coherencia que se utiliza para las operaciones de S3 realizadas en los objetos del bloque o en la configuración de bloques. No afecta a las operaciones del propio cucharón.
- El control de coherencia de una operación API individual anula el control de coherencia del bloque.
- En general, los cucharones deben utilizar el control de coherencia predeterminado, «entre una y otra escritura». Si las solicitudes no funcionan correctamente, cambie el comportamiento del cliente de la aplicación, si es posible. O bien, configure el cliente para especificar el control de consistencia de cada solicitud API. Establecer el control de consistencia a nivel de cucharón únicamente como último recurso.

#### Cómo interactúan los controles de consistencia y las reglas de ILM para afectar a la protección de datos

Tanto la elección del control de coherencia como la regla de ILM afectan a la forma en que se protegen los objetos. Estos ajustes pueden interactuar.

Por ejemplo, el control de consistencia usado cuando se almacena un objeto afecta a la colocación inicial de los metadatos de objetos, mientras que el comportamiento de procesamiento seleccionado para la regla de ILM afecta a la colocación inicial de las copias de objetos. Dado que StorageGRID requiere acceso tanto a los metadatos de un objeto como a sus datos para cumplir con las solicitudes de los clientes, seleccionar los niveles de protección correspondientes para el nivel de coherencia y el comportamiento de ingesta puede proporcionar una mejor protección de datos inicial y respuestas más predecibles del sistema.

Para las reglas de ILM hay disponibles los siguientes comportamientos de consumo:

- **Estricto**: Todas las copias especificadas en la regla ILM deben hacerse antes de que el éxito se devuelva al cliente.
- **Balanceado**: StorageGRID intenta hacer todas las copias especificadas en la regla ILM en la ingesta; si esto no es posible, se hacen copias provisionales y se devuelve éxito al cliente. Las copias especificadas en la regla ILM se realizan cuando es posible.
- **Commit doble**: StorageGRID realiza inmediatamente copias provisionales del objeto y devuelve éxito al cliente. Las copias especificadas en la regla ILM se realizan cuando es posible.

Antes de seleccionar el comportamiento de ingesta para una regla de ILM, lea la descripción completa de estas opciones de configuración en Gestión de objetos con ILM.

**Ejemplo de cómo puede interactuar el control de consistencia y la regla de ILM**

Suponga que tiene una cuadrícula de dos sitios con la siguiente regla de ILM y la siguiente configuración de nivel de coherencia:

- **Norma ILM**: Cree dos copias de objetos, una en el sitio local y otra en un sitio remoto. Se ha seleccionado el comportamiento de procesamiento estricto.
- **Nivel de coherencia**: "Strong-global" (los metadatos de objetos se distribuyen inmediatamente a todos los sitios).

Cuando un cliente almacena un objeto en el grid, StorageGRID realiza copias de objetos y distribuye los metadatos en ambos sitios antes de devolver el éxito al cliente.

El objeto está completamente protegido contra la pérdida en el momento del mensaje de procesamiento correcto. Por ejemplo, si el sitio local se pierde poco después del procesamiento, seguirán existiendo copias de los datos del objeto y los metadatos del objeto en el sitio remoto. El objeto se puede recuperar completamente.

Si en su lugar usa la misma regla de ILM y el nivel de consistencia de «otrong-site», es posible que el cliente reciba un mensaje de éxito después de replicar los datos del objeto en el sitio remoto, pero antes de que los metadatos del objeto se distribuyan allí. En este caso, el nivel de protección de los metadatos de objetos no coincide con el nivel de protección de los datos de objetos. Si el sitio local se pierde poco después del procesamiento, se pierden los metadatos del objeto. No se puede recuperar el objeto.

La interrelación entre los niveles de coherencia y las reglas del ILM puede ser compleja. Póngase en contacto con NetApp si necesita ayuda.

**Información relacionada**

OBTENGA la solicitud de consistencia de bloque

PONER solicitud de consistencia de bloque

## Cómo gestionan las reglas de ILM de StorageGRID los objetos

El administrador de grid crea reglas de gestión del ciclo de vida de la información (ILM) para gestionar los datos de los objetos que se ingieren en el sistema StorageGRID desde aplicaciones cliente de la API REST S3. A continuación, estas reglas se añaden a la política de ILM para determinar cómo y dónde se almacenan los datos de objetos con el tiempo.

La configuración de ILM determina los siguientes aspectos de un objeto:

- **Geografía**

  La ubicación de los datos de un objeto, ya sea en el sistema StorageGRID (pool de almacenamiento) o en un pool de almacenamiento en el cloud.

- **Grado de almacenamiento**

  El tipo de almacenamiento utilizado para almacenar datos de objetos, como la tecnología flash o el disco giratorio.

- **Protección contra pérdidas**

  Cuántas copias se hacen y los tipos de copias que se crean: Replicación, codificación de borrado o ambos.

- **Retención**

  Los cambios se producen a lo largo del tiempo en el modo en que se gestionan los datos de un objeto, dónde se almacenan y cómo se protegen de pérdidas.

- **Protección durante la ingesta**

  El método utilizado para proteger los datos de objetos durante el procesamiento: Colocación síncrona (utilizando las opciones equilibradas o estrictas para el comportamiento de ingesta) o creación de copias provisionales (mediante la opción Dual Commit).

Las reglas de ILM pueden filtrar y seleccionar objetos. Para los objetos ingeridos mediante S3, las reglas de ILM pueden filtrar objetos en función de los siguientes metadatos:

- Cuenta de inquilino
- Nombre del bloque
- Tiempo de ingesta
- Clave
- Hora del último acceso

De forma predeterminada, las actualizaciones del último tiempo de acceso se deshabilitan para todos los bloques S3. Si el sistema StorageGRID incluye una regla de ILM que usa la opción Last Access Time, debe habilitar las actualizaciones a la hora del último acceso para los bloques S3 especificados en esa regla. Puede habilitar las actualizaciones de la última hora de acceso mediante LA solicitud DE LA última hora de acceso DE PUT Bucket, la casilla de verificación **S3** > **Cuchos** > **Configurar la última hora de acceso** en el Administrador de inquilinos o mediante la API de administración de inquilinos. Al habilitar las actualizaciones del último tiempo de acceso, tenga en cuenta que el rendimiento de StorageGRID puede reducirse, especialmente en sistemas con objetos pequeños.

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

Para obtener más información sobre ILM, consulte las instrucciones para gestionar objetos con la gestión del ciclo de vida de la información.

**Información relacionada**

Usar cuenta de inquilino

Gestión de objetos con ILM

PUT Bucket última solicitud de tiempo de acceso

## 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 implementa versiones con compatibilidad para la mayoría de las funciones y con algunas limitaciones. StorageGRID admite hasta 1,000 versiones de cada objeto.

El control de versiones de objetos puede combinarse con la gestión del ciclo de vida de la información (ILM) de StorageGRID o con la configuración del ciclo de vida de bloques de S3. Debe habilitar explícitamente el control de versiones para cada segmento a fin de activar esta funcionalidad para el bloque. A cada objeto de su bloque se le asigna un ID de versión, que genera el sistema StorageGRID.

No se admite el uso de la autenticación multifactor (MFA).

El control de versiones solo se puede habilitar en bloques creados con StorageGRID versión 10.3 o posterior.

### ILM y versiones

Las políticas de ILM se aplican a cada versión de un objeto. Un proceso de análisis de ILM analiza continuamente todos los objetos y los vuelve a evaluar en relación con la política actual de ILM. Todos los cambios realizados en las políticas de ILM se aplican a todos los objetos procesados anteriormente. Esto incluye versiones que se han ingerido previamente si la versión está activada. El análisis de ILM aplica nuevos cambios de ILM a los objetos procesados previamente.

En el caso de objetos S3 en bloques habilitados para versionado, la compatibilidad con versionado le permite crear reglas de ILM que usen hora no corriente como tiempo de referencia. Cuando se actualiza un objeto, sus versiones anteriores se vuelven no actuales. El uso de un filtro de tiempo no actual permite crear políticas que reduzcan el impacto en el almacenamiento de las versiones anteriores de objetos.

> Cuando se carga una nueva versión de un objeto mediante una operación de carga de varias partes, la hora no actual de la versión original del objeto se refleja cuando se creó la carga de varias partes para la nueva versión, no cuando se completó la carga de varias partes. En casos limitados, la hora no actual de la versión original puede ser horas o días antes de la hora de la versión actual.

Consulte las instrucciones para gestionar objetos con gestión del ciclo de vida de la información para ver un ejemplo de política de ILM para objetos con versiones de S3.

**Información relacionada**

Gestión de objetos con ILM

## Recomendaciones para implementar la API REST de S3

Debe seguir estas recomendaciones al implementar la API DE REST de S3 para usar con StorageGRID.

### Recomendaciones para las cabezas a los objetos no existentes

Si su aplicación comprueba de forma rutinaria si existe un objeto en una ruta en la que no espera que el objeto exista realmente, debe utilizar el control de consistencia "'disponible'". Por ejemplo, debe utilizar el control de coherencia "'disponible'" si su aplicación dirige una ubicación antes DE PONERLA en práctica.

De lo contrario, si la operación HEAD no encuentra el objeto, es posible que reciba un número elevado de 500 errores internos de Server si uno o más nodos de almacenamiento no están disponibles.

Puede establecer el control de consistencia "'Available'" para cada bloque mediante LA solicitud DE consistencia PUT Bucket, o bien puede especificar el control de consistencia en el encabezado de solicitud para una operación de API individual.

### Recomendaciones para las claves de objeto

En el caso de los bloques creados en StorageGRID 11.4 o posterior, ya no es necesario restringir los nombres de claves de objetos para cumplir con las prácticas recomendadas de rendimiento. Por ejemplo, ahora puede utilizar valores aleatorios para los primeros cuatro caracteres de nombres de claves de objeto.

Para los bloques que se crearon en las versiones anteriores a StorageGRID 11.4, siga estas recomendaciones para los nombres de claves de objetos:

- No debe utilizar valores aleatorios como los primeros cuatro caracteres de claves de objeto. Esto contrasta con la anterior recomendación de AWS para prefijos clave. En su lugar, debe utilizar prefijos no aleatorios y no únicos, como `image`.
- Si sigue la recomendación anterior de AWS de utilizar caracteres aleatorios y únicos en prefijos clave, debe anteponer las claves de objeto con un nombre de directorio. Es decir, utilice este formato:

```
mybucket/mydir/f8e3-image3132.jpg
```

En lugar de este formato:

```
mybucket/f8e3-image3132.jpg
```

**Recomendaciones para «lecturas de rango»**

Si se selecciona la opción **Compress Stored Objects** (**CONFIGURATION** > **System** > **Grid options**), las aplicaciones cliente S3 deberían evitar realizar operaciones GET Object que especifiquen un intervalo de bytes. Estas operaciones de «lectura de rango» son ineficientes, ya que StorageGRID debe descomprimir de forma efectiva los objetos para acceder a los bytes solicitados. LAS operaciones GET Object que solicitan un rango pequeño de bytes de un objeto muy grande son especialmente ineficientes; por ejemplo, es muy ineficiente leer un rango de 10 MB de un objeto comprimido de 50 GB.

Si se leen rangos de objetos comprimidos, las solicitudes del cliente pueden tener un tiempo de espera.

> Si necesita comprimir objetos y su aplicación cliente debe utilizar lecturas de rango, aumente el tiempo de espera de lectura de la aplicación.

**Información relacionada**

- Controles de consistencia
- PONER solicitud de consistencia de bloque
- Administre StorageGRID

# Operaciones y limitaciones compatibles con la API REST de S3

El sistema StorageGRID implementa la API de servicio de almacenamiento simple (API 2006-03-01) con compatibilidad para la mayoría de las operaciones y con algunas limitaciones. Debe comprender los detalles de la implementación al integrar las aplicaciones cliente de la API DE REST de S3.

El sistema StorageGRID admite tanto solicitudes virtuales de tipo hospedado como solicitudes de tipo path.

## Gestión de fechas

La implementación de StorageGRID de la API REST de S3 solo admite formatos de fecha HTTP válidos.

El sistema StorageGRID sólo admite formatos de fecha HTTP válidos para cualquier encabezado que acepte valores de fecha. La parte horaria de la fecha puede especificarse en formato de hora media de Greenwich (GMT) o en formato de hora universal coordinada (UTC) sin desplazamiento de zona horaria (se debe especificar +0000). Si incluye el `x-amz-date` Encabezado de la solicitud, anula cualquier valor especificado en el encabezado de solicitud de fecha. Al utilizar la versión 4 de la firma de AWS, el `x-amz-date` el encabezado debe estar presente en la solicitud firmada porque no se admite el encabezado de fecha.

## Encabezados de solicitud comunes

El sistema StorageGRID admite encabezados de solicitud comunes definidos por la "Documentación de Amazon Web Services (AWS): Referencia de API de Amazon simple Storage Service", con una excepción.

| Solicite el encabezado | Implementación |
| --- | --- |
| Autorización | Compatibilidad completa con la firma AWS Versión 2<br><br>Compatibilidad con la versión 4 de la firma de AWS, con las siguientes excepciones:<br><br>• El valor SHA256 no se calcula para el cuerpo de la solicitud. El valor enviado por el usuario se acepta sin validación, como si fuera el valor `UNSIGNED-PAYLOAD` se había proporcionado para el `x-amz-content-sha256` encabezado. |
| x-amz-token de seguridad | No implementada. Retornos `XNotImplemented`. |

## Encabezados de respuesta comunes

El sistema StorageGRID admite todos los encabezados de respuesta comunes definidos por *simple Storage Service API Reference*, con una excepción.

| Encabezado de respuesta | Implementación |
| --- | --- |
| x-amz-id-2 | No se utiliza |

## Autenticar solicitudes

El sistema StorageGRID admite el acceso autenticado y anónimo a objetos mediante la API de S3.

La API S3 admite la versión 2 de Signature y la versión 4 de Signature para autenticar solicitudes de API S3.

Las solicitudes autenticadas deben firmarse mediante su ID de clave de acceso y su clave de acceso secreta.

El sistema StorageGRID admite dos métodos de autenticación: HTTP `Authorization` encabezado y uso de parámetros de consulta.

### Utilice el encabezado autorización HTTP

HTTP `Authorization` Todas las operaciones de la API de S3 utilizan el encabezado excepto las solicitudes anónimas, donde lo permite la directiva de bloques. La `Authorization` encabezado contiene toda la información de firma necesaria para autenticar una solicitud.

### Utilice los parámetros de consulta

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

## Operaciones en el servicio

El sistema StorageGRID admite las siguientes operaciones en el servicio.

| Funcionamiento | Implementación |
|---|---|
| OBTENER servicio | Se implementa con todo el comportamiento de la API DE REST de Amazon S3. |
| Obtenga el uso del almacenamiento | La solicitud GET Storage Usage le indica la cantidad total de almacenamiento que está usando una cuenta y por cada bloque asociado con la cuenta. Se trata de una operación en el servicio con una ruta de / y un parámetro de consulta personalizado (`?x-ntap-sg-usage`) agregado. |
| OPCIONES / | Las aplicaciones cliente pueden emitir `OPTIONS /` Se solicita al puerto S3 en un nodo de almacenamiento, sin proporcionar credenciales de autenticación S3, para determinar si el nodo de almacenamiento está disponible. Puede usar esta solicitud para supervisar o para permitir que los equilibradores de carga externos identifiquen cuando un nodo de almacenamiento esté inactivo. |

**Información relacionada**

OBTENGA la solicitud de uso del almacenamiento

## Operaciones en bloques

El sistema StorageGRID admite un máximo de 1,000 bloques para cada cuenta de inquilino de S3.

Las restricciones de nombres de bloque siguen las restricciones de región del estándar estadounidense de AWS, pero debe restringirlas a convenciones de nomenclatura de DNS para admitir solicitudes de estilo hospedado virtual de S3.

"Documentación de Amazon Web Services (AWS): Restricciones y limitaciones de buckets"

Configure los nombres de dominio de extremo API de S3

Las operaciones GET Bucket (List Objects) Y GET Bucket admiten los controles de coherencia de StorageGRID.

Puede comprobar si las actualizaciones a la hora del último acceso están habilitadas o deshabilitadas para grupos individuales.

En la siguiente tabla se describe cómo StorageGRID implementa operaciones de bloque de API DE REST de S3. Para realizar alguna de estas operaciones, se deben proporcionar las credenciales de acceso necesarias para la cuenta.

| Funcionamiento | Implementación |
| --- | --- |
| ELIMINAR bloque | Se implementa con todo el comportamiento de la API DE REST de Amazon S3. |
| ELIMINAR los cors de cucharón | Esta operación elimina la configuración de CORS para el cucharón. |
| DELETE Bucket Encryption | Esta operación elimina el cifrado predeterminado del bloque. Los objetos cifrados existentes permanecen cifrados, pero los nuevos objetos agregados al bloque no están cifrados. |
| ELIMINAR ciclo de vida de bloque | Esta operación elimina la configuración del ciclo de vida del bloque. |
| ELIMINE la política de bloques | Esta operación elimina la política asociada al bloque. |
| DELETE Bucket replicación | Esta operación elimina la configuración de replicación conectada al bloque. |
| DELETE Bucket tagging | Esta operación utiliza `tagging` subrecurso para quitar todas las etiquetas de un bloque. |
| GET Bucket (List Objects), versión 1 y versión 2 | Esta operación devuelve algunos o todos (hasta 1,000) de los objetos de un bloque. La clase de almacenamiento para los objetos puede tener cualquiera de dos valores, incluso si el objeto se ingirió con la `REDUCED_REDUNDANCY` opción de clase de almacenamiento:<br>• `STANDARD`, Que indica que el objeto se almacena en una agrupación de almacenamiento que consta de nodos de almacenamiento.<br>• `GLACIER`, Que indica que el objeto se ha movido al bloque externo especificado por el grupo de almacenamiento en la nube.<br>Si el bloque contiene un gran número de claves eliminadas que tienen el mismo prefijo, la respuesta podría incluir algunas `CommonPrefixes` que no contienen claves. |
| GET Bucket acl | Esta operación devuelve una respuesta positiva y el ID, DisplayName y permiso del propietario del bloque, lo que indica que el propietario tiene acceso completo al bloque. |
| OBTENGA los cors del cucharón | Esta operación devuelve el `cors` configuración del bloque. |
| OBTENGA el cifrado de bloque | Esta operación devuelve la configuración de cifrado predeterminada del bloque. |

| Funcionamiento | Implementación |
| --- | --- |
| OBTENGA el ciclo de vida de la cuchara | Esta operación devuelve la configuración del ciclo de vida del bloque. |
| OBTENER ubicación de bloque | Esta operación devuelve la región que se estableció mediante el `LocationConstraint` Elemento de la solicitud PUT Bucket. Si la región del cucharón es `us-east-1`, se devuelve una cadena vacía para la región. |
| OBTENGA la notificación DE BUCKET | Esta operación devuelve la configuración de notificación asociada al bloque. |
| OBTENGA las versiones DE objeto Bucket | Con el acceso DE LECTURA en un bloque, esta operación con el `versions` subrecurso enumera los metadatos de todas las versiones de objetos del bloque. |
| OBTENGA la política de bloques | Esta operación devuelve la política asociada al bloque. |
| OBTENGA la replicación de Bucket | Esta operación devuelve la configuración de replicación asociada al bloque. |
| GET Bucket tagging | Esta operación utiliza `tagging` subrecurso para devolver todas las etiquetas de un bloque. |
| OBTENGA el control de versiones de Bucket | Esta implementación usa la `versioning` subrecurso para devolver el estado de control de versiones de un bloque.<br>• *Blank*: Nunca se ha activado el control de versiones (el cucharón es "'unversioned'")<br>• Activado: El control de versiones está activado<br>• Suspendido: El control de versiones se ha habilitado anteriormente y se ha suspendido |
| OBTENER configuración de bloqueo de objeto | Esta operación devuelve el modo de retención predeterminado de bloque y el período de retención predeterminado, si está configurado.<br>Consulte OBTENER configuración de bloqueo de objeto para obtener información detallada. |
| Cubo DE CABEZA | Esta operación determina si existe un bloque y tiene permiso para acceder a él.<br>Esta operación devuelve:<br>• `x-ntap-sg-bucket-id`: El UUID del bloque en formato UUID.<br>• `x-ntap-sg-trace-id`: El ID de traza único de la solicitud asociada. |

| Funcionamiento | Implementación |
| --- | --- |
| COLOQUE el cucharón | Esta operación crea un nuevo bloque. Al crear la cuchara, se convierte en el propietario de la cuchara.<br><br>• Los nombres de los bloques deben cumplir con las siguientes reglas:<br>◦ Debe ser único en cada sistema StorageGRID (no solo dentro de la cuenta de inquilino).<br>◦ Debe ser compatible con DNS.<br>◦ Debe incluir al menos 3 y no más de 63 caracteres.<br>◦ Puede ser una serie de una o más etiquetas, con etiquetas adyacentes separadas por un punto. Cada etiqueta debe comenzar y terminar con una letra minúscula o un número y solo puede utilizar letras minúsculas, números y guiones.<br>◦ No debe ser una dirección IP con formato de texto.<br>◦ No debe utilizar periodos en solicitudes de estilo alojadas virtuales. Los períodos provocarán problemas en la verificación del certificado comodín del servidor.<br>• De forma predeterminada, los bloques se crean en la `us-east-1` región; sin embargo, puede utilizar la `LocationConstraint` elemento de solicitud en el cuerpo de solicitud para especificar una región diferente. Cuando utilice la `LocationConstraint` Elemento, debe especificar el nombre exacto de una región que se ha definido mediante el Administrador de grid o la API de gestión de grid. Póngase en contacto con el administrador del sistema si no conoce el nombre de región que debe utilizar.<br><br>**Nota**: Se producirá un error si la solicitud PUT Bucket utiliza una región que no se ha definido en StorageGRID.<br><br>• Puede incluir el `x-amz-bucket-object-lock-enabled` Solicite el encabezado para crear un bucket con el bloqueo de objetos S3 habilitado. Consulte Utilice el bloqueo de objetos de S3.<br><br>Debe habilitar S3 Object Lock cuando crea el bloque. No se puede añadir o deshabilitar el bloqueo de objetos de S3 después de crear un bloque. S3 Object Lock requiere el control de versiones de bloques, que se habilita automáticamente al crear el bloque. |
| COLOQUE los cors del cucharón | Esta operación establece la configuración de CORS para un cucharón para que éste pueda atender solicitudes de origen cruzado. El uso compartido de recursos de origen cruzado (CORS) es un mecanismo de seguridad que permite a las aplicaciones web de cliente de un dominio acceder a los recursos de un dominio diferente. Por ejemplo, supongamos que se utiliza un bloque de S3 llamado `images` para almacenar gráficos. Mediante el ajuste de la configuración de CORS para `images` bloque, puede permitir que las imágenes de ese bloque se muestren en el sitio web `http://www.example.com`. |

| Funcionamiento | Implementación |
| --- | --- |
| PUT Bucket Encryption | Esta operación establece el estado de cifrado predeterminado de un bloque existente. Cuando se habilita el cifrado a nivel de bloque, se cifran todos los objetos nuevos que se añadan al bloque.StorageGRID admite el cifrado en el lado del servidor con claves gestionadas por StorageGRID. Al especificar la regla de configuración de cifrado del servidor, defina la `SSEAlgorithm` parámetro a. `AES256`, y no utilice `KMSMasterKeyID` parámetro.<br><br>La configuración de cifrado predeterminada de bloque se omite si la solicitud de carga de objeto ya especifica cifrado (es decir, si la solicitud incluye la `x-amz-server-side-encryption-*` encabezado de solicitud). |
| CICLO de vida DE la cuchara | Esta operación crea una nueva configuración del ciclo de vida para el bloque o reemplaza una configuración de ciclo de vida existente. StorageGRID admite hasta 1,000 reglas de ciclo de vida en una configuración del ciclo de vida. Cada regla puede incluir los siguientes elementos XML:<br><br>• Caducidad (días, fecha)<br>• NoncurrentVersionExpiración (NoncurrentDays)<br>• Filtro (prefijo, etiqueta)<br>• Estado<br>• ID<br><br>StorageGRID no admite estas acciones:<br><br>• AbortEncompleteMultipartUpload<br>• ExpiredObjectDeleteMarker<br>• Transición<br><br>Para comprender cómo la acción de caducidad en el ciclo de vida de un bloque interactúa con las instrucciones de colocación de ILM, consulte ""Cómo funciona ILM durante la vida de un objeto" en las instrucciones para gestionar objetos con gestión del ciclo de vida de la información.<br><br>**Nota**: La configuración del ciclo de vida de la cuchara se puede utilizar con cucharones que tengan habilitado el bloqueo de objetos S3, pero la configuración del ciclo de vida de la cuchara no es compatible con cucharones legados compatibles. |

| Funcionamiento | Implementación |
| --- | --- |
| NOTIFICACIÓN DE PUT Bucket | Esta operación configura notificaciones para el bloque mediante el XML de configuración de notificación incluido en el cuerpo de la solicitud. Debe tener en cuenta los siguientes detalles de implementación:<br>• StorageGRID admite temas como destinos el Servicio de notificación simple (SNS). No se admiten extremos de simple Queue Service (SQS) o Amazon Lambda.<br>• El destino de las notificaciones debe especificarse como URN de un extremo de StorageGRID. Se pueden crear extremos con el administrador de inquilinos o la API de gestión de inquilinos.<br>El extremo debe existir para que la configuración de la notificación se realice correctamente. Si el extremo no existe, un `400 Bad Request` se devuelve un error con el código `InvalidArgument`.<br>• No es posible configurar una notificación para los siguientes tipos de eventos. Estos tipos de evento **no** son compatibles.<br>◦ `s3:ReducedRedundancyLostObject`<br>◦ `s3:ObjectRestore:Completed`<br>• Las notificaciones de eventos enviadas desde StorageGRID utilizan el formato JSON estándar excepto que no incluyen algunas claves y utilizan valores específicos para otros, como se muestra en el siguiente listado:<br>• **EventSource**<br>`sgws:s3`<br>• * AwsRegion*<br>no incluido<br>• **x-amz-id-2**<br>no incluido<br>• **arn**<br>`urn:sgws:s3:::bucket_name` |
| POLÍTICA DE PUT Bucket | Esta operación establece la política asociada al bloque. |

| Funcionamiento | Implementación |
| --- | --- |
| PUT Bucket replication | Esta operación configura la replicación de CloudMirror de StorageGRID para el bloque con el XML de configuración de replicación que se proporciona en el cuerpo de la solicitud. Para la replicación de CloudMirror, debe tener en cuenta los siguientes detalles de la implementación:<br><br>• StorageGRID solo admite V1 de la configuración de replicación. Esto significa que StorageGRID no admite el uso de `Filter` Elemento para reglas y sigue las convenciones V1 para eliminar versiones de objetos. Para obtener más detalles, consulte "Documentación de Amazon S3 en la configuración de la replicación".<br>• La replicación de bloques se puede configurar en bloques con versiones o sin versiones.<br>• Puede especificar un segmento de destino diferente en cada regla del XML de configuración de replicación. Un bloque de origen puede replicar en más de un bloque de destino.<br>• Los bloques de destino se deben especificar como URN de extremos StorageGRID tal y como se especifica en el administrador de inquilinos o la API de gestión de inquilinos.<br><br>El extremo debe existir para que la configuración de replicación se complete correctamente. Si el extremo no existe, la solicitud falla como un `400 Bad Request`. El mensaje de error indica: `Unable to save the replication policy. The specified endpoint URN does not exist: URN`.<br><br>• No es necesario especificar un `Role` En el XML de configuración. StorageGRID no utiliza este valor y se ignorará si se envía.<br>• Si omite la clase de almacenamiento del XML de configuración, StorageGRID utiliza `STANDARD` clase de almacenamiento de forma predeterminada.<br>• Si elimina un objeto del bloque de origen o elimina el propio bloque de origen, el comportamiento de replicación entre regiones es el siguiente:<br>◦ Si elimina el objeto o bloque antes de que se haya replicado, el objeto o bloque no se replicará y no se le notificará.<br>◦ Si elimina el objeto o bloque después de haber sido replicado, StorageGRID sigue el comportamiento estándar de eliminación de Amazon S3 para V1 de replicación entre regiones. |

| Funcionamiento | Implementación |
| --- | --- |
| PUT Bucket etiquetaje | Esta operación utiliza `tagging` subrecurso para agregar o actualizar un conjunto de etiquetas para un bloque. Al añadir etiquetas de bloque, tenga en cuenta las siguientes limitaciones:<br>• Tanto StorageGRID como Amazon S3 admiten hasta 50 etiquetas por cada bloque.<br>• Las etiquetas asociadas con un bloque deben tener claves de etiqueta únicas. Una clave de etiqueta puede tener hasta 128 caracteres Unicode de longitud.<br>• Los valores de etiqueta pueden tener una longitud máxima de 256 caracteres Unicode.<br>• La clave y los valores distinguen entre mayúsculas y minúsculas. |
| PONER creación de versiones de bloques | Esta implementación usa la `versioning` subrecurso para establecer el estado de control de versiones de un bloque existente. Puede establecer el estado de control de versiones con uno de los siguientes valores:<br>• Enabled: Activa el control de versiones de los objetos del bloque. Todos los objetos que se agregan al bloque reciben un ID de versión único.<br>• Suspendido: Desactiva el control de versiones de los objetos del bloque. Todos los objetos agregados al bloque reciben el ID de versión `null`. |
| PONER configuración de bloqueo de objeto | Esta operación configura o elimina el modo de retención predeterminado de bloque y el período de retención predeterminado.<br>Si se modifica el período de retención predeterminado, la fecha de retención hasta la de las versiones de objeto existentes seguirá siendo la misma y no se volverá a calcular utilizando el nuevo período de retención predeterminado.<br>Consulte PONER configuración de bloqueo de objeto para obtener información detallada. |

**Información relacionada**

Controles de consistencia

GET Bucket última solicitud de tiempo de acceso

Políticas de acceso a bloques y grupos

Se realizó un seguimiento de las operaciones de S3 en los registros de auditoría

Gestión de objetos con ILM

Usar cuenta de inquilino

#### Cree una configuración del ciclo de vida de S3

Puede crear una configuración del ciclo de vida de S3 para controlar cuándo se eliminan

objetos específicos del sistema StorageGRID.

El ejemplo sencillo de esta sección muestra cómo puede controlar una configuración del ciclo de vida de S3 cuando se eliminan ciertos objetos (caducados) de bloques S3 específicos. El ejemplo de esta sección es solo con fines ilustrativos. Para obtener información completa sobre la creación de configuraciones del ciclo de vida de S3, consulte "Guía para desarrolladores de Amazon simple Storage Service: Gestión del ciclo de vida de los objetos". Tenga en cuenta que StorageGRID solo admite acciones de caducidad, no admite acciones de transición.

#### Qué es la configuración del ciclo de vida

Una configuración de ciclo de vida es un conjunto de reglas que se aplican a los objetos en bloques de S3 específicos. Cada regla especifica qué objetos se ven afectados y cuándo caducarán dichos objetos (en una fecha específica o después de un número determinado de días).

StorageGRID admite hasta 1,000 reglas de ciclo de vida en una configuración del ciclo de vida. Cada regla puede incluir los siguientes elementos XML:

- Caducidad: Elimine un objeto cuando se alcance una fecha especificada o cuando se alcance un número especificado de días, empezando desde el momento en que se ingirió el objeto.
- NoncurrentVersionExpiration: Elimine un objeto cuando se alcance un número especificado de días, empezando desde el momento en que el objeto se volvió no actual.
- Filtro (prefijo, etiqueta)
- Estado
- ID

Si aplica una configuración del ciclo de vida a un bloque, la configuración del ciclo de vida del bloque siempre anula la configuración de ILM de StorageGRID. StorageGRID utiliza la configuración de caducidad del bloque, no de ILM, para determinar si se deben eliminar o conservar objetos específicos.

Como resultado, es posible que se elimine un objeto de la cuadrícula aunque las instrucciones de colocación de una regla de ILM aún se apliquen al objeto. O bien, es posible que un objeto se conserve en la cuadrícula incluso después de que hayan transcurrido las instrucciones de colocación de ILM para el objeto. Para obtener más información, consulte Cómo funciona ILM durante la vida de un objeto.

> La configuración del ciclo de vida de bloques se puede usar con bloques que tienen habilitado el bloqueo de objetos S3, pero la configuración del ciclo de vida de bloques no se admite para bloques compatibles con versiones anteriores.

StorageGRID admite el uso de las siguientes operaciones de bloques para gestionar las configuraciones del ciclo de vida:

- ELIMINAR ciclo de vida de bloque
- OBTENGA el ciclo de vida de la cuchara
- CICLO de vida DE la cuchara

#### Cree la configuración del ciclo de vida

Como primer paso en la creación de una configuración de ciclo de vida, se crea un archivo JSON que incluye una o varias reglas. Por ejemplo, este archivo JSON incluye tres reglas, de la siguiente manera:

1. La regla 1 sólo se aplica a los objetos que coinciden con el prefijo `category1/` y que tienen un `key2` valor

de `tag2`. La `Expiration` Parámetro especifica que los objetos que coinciden con el filtro caducarán a medianoche el 22 de agosto de 2020.

2. La regla 2 sólo se aplica a los objetos que coinciden con el prefijo `category2/`. La `Expiration` el parámetro especifica que los objetos que coinciden con el filtro caducarán 100 días después de que se ingieran.

> Las reglas que especifican un número de días son relativas al momento en que se ingirió el objeto. Si la fecha actual supera la fecha de ingesta más el número de días, es posible que algunos objetos se eliminen del bloque en cuanto se aplique la configuración del ciclo de vida.

3. La regla 3 sólo se aplica a los objetos que coinciden con el prefijo `category3/`. La `Expiration` parámetro especifica que cualquier versión no actual de objetos coincidentes caducará 50 días después de que se conviertan en no actualizados.

```
{
    "Rules": [
        {
            "ID": "rule1",
            "Filter": {
                "And": {
                    "Prefix": "category1/",
                    "Tags": [
                        {
                            "Key": "key2",
                            "Value": "tag2"
                        }
                    ]
                }
            },
            "Expiration": {
                "Date": "2020-08-22T00:00:00Z"
            },
            "Status": "Enabled"
        },
        {
            "ID": "rule2",
            "Filter": {
                "Prefix": "category2/"
            },
            "Expiration": {
                "Days": 100
            },
            "Status": "Enabled"
        },
        {
            "ID": "rule3",
            "Filter": {
                "Prefix": "category3/"
            },
            "NoncurrentVersionExpiration": {
                "NoncurrentDays": 50
            },
            "Status": "Enabled"
        }
    ]
}
```

#### Aplicar la configuración del ciclo de vida al bloque

Después de crear el archivo de configuración del ciclo de vida, se aplica a un bloque emitiendo una solicitud PUT Bucket Lifecycle.

Esta solicitud aplica la configuración del ciclo de vida del archivo de ejemplo a los objetos de un bloque denominado `testbucket`.

```
aws s3api --endpoint-url <StorageGRID endpoint> put-bucket-lifecycle-
configuration
--bucket testbucket --lifecycle-configuration file://bktjson.json
```

Para validar que se ha aplicado correctamente una configuración del ciclo de vida al bloque, emita una solicitud GET Bucket Lifecycle. Por ejemplo:

```
aws s3api --endpoint-url <StorageGRID endpoint> get-bucket-lifecycle-
configuration
 --bucket testbucket
```

Una respuesta correcta muestra la configuración del ciclo de vida que acaba de aplicar.

#### Validar que la caducidad del ciclo de vida del bloque se aplica al objeto

Puede determinar si una regla de caducidad en la configuración del ciclo de vida se aplica a un objeto específico al emitir una solicitud PUT Object, HEAD Object o GET Object. Si se aplica una regla, la respuesta incluye una `Expiration` parámetro que indica cuándo caduca el objeto y qué regla de caducidad se ha coincidido.

> Dado que el ciclo de vida de los bloques anula la gestión del ciclo de vida de `expiry-date` se muestra la fecha real en la que se eliminará el objeto. Para obtener más información, consulte Cómo se determina la retención de objetos.

Por ejemplo, esta solicitud PUT Object fue emitida el 22 de junio de 2020 y coloca un objeto en el `testbucket` cucharón.

```
aws s3api --endpoint-url <StorageGRID endpoint> put-object
--bucket testbucket --key obj2test2 --body bktjson.json
```

La respuesta correcta indica que el objeto caducará en 100 días (01 de octubre de 2020) y que coincide con la regla 2 de la configuración del ciclo de vida.

```
{
      *"Expiration": "expiry-date=\"Thu, 01 Oct 2020 09:07:49 GMT\", rule-
id=\"rule2\"",
      "ETag": "\"9762f8a803bc34f5340579d4446076f7\""
}
```

Por ejemplo, esta solicitud DE OBJETO HEAD se utilizó para obtener metadatos para el mismo objeto en el bloque testbucket.

```
aws s3api --endpoint-url <StorageGRID endpoint> head-object
--bucket testbucket --key obj2test2
```

La respuesta correcta incluye los metadatos del objeto e indica que el objeto caducará en 100 días y que coincide con la regla 2.

```
{
      "AcceptRanges": "bytes",
      *"Expiration": "expiry-date=\"Thu, 01 Oct 2020 09:07:48 GMT\", rule-
id=\"rule2\"",
      "LastModified": "2020-06-23T09:07:48+00:00",
      "ContentLength": 921,
      "ETag": "\"9762f8a803bc34f5340579d4446076f7\""
      "ContentType": "binary/octet-stream",
      "Metadata": {}
}
```

#### Use la retención de bloque predeterminada de Object Lock de S3

Si un bloque tiene habilitado el bloqueo de objetos S3, puede especificar un modo de retención predeterminado y el período de retención predeterminado que se aplicará a cada objeto que se agregue al bloque.

- El bloqueo de objetos de S3 se puede habilitar o deshabilitar para un bloque durante la creación de bloques.
- Si se habilita el bloqueo de objetos S3 para un bloque, puede configurar la retención predeterminada para el bloque.
- La configuración de retención predeterminada especifica:
  - Modo de retención predeterminado: StorageGRID sólo admite el modo de “CUMPLIMIENTO”.
  - Período de retención predeterminado en días o años.

##### OBTENER configuración de bloqueo de objeto

La solicitud GET Object Lock Configuration (OBTENER configuración de bloqueo de objeto) permite determinar si el bloqueo de objeto está habilitado para un bloque y, si está habilitado, ver si hay un modo de

retención predeterminado y un período de retención configurado para el segmento.

Cuando se ingieren nuevas versiones de objetos en el bloque, se aplica el modo de retención predeterminado si `x-amz-object-lock-mode` no se ha especificado. El período de retención predeterminado se utiliza para calcular el valor de retener hasta la fecha if `x-amz-object-lock-retain-until-date` no se ha especificado.

Para completar esta operación, debe tener el permiso s3:GetBucketObjectLockConfiguration o ser la raíz de la cuenta.

**Ejemplo de solicitud**

```
GET /bucket?object-lock HTTP/1.1
Host: host
Accept-Encoding: identity
User-Agent: aws-cli/1.18.106 Python/3.8.2 Linux/4.4.0-18362-Microsoft
botocore/1.17.29
x-amz-date: date
x-amz-content-sha256: authorization string
Authorization: authorization string
```

**Ejemplo de respuesta**

```
HTTP/1.1 200 OK
x-amz-id-2:
iVmcB7OXXJRkRH1FiVq1151/T24gRfpwpuZrEG11Bb9ImOMAAe98oxSpXlknabA0LTvBYJpSIX
k=
x-amz-request-id: B34E94CACB2CEF6D
Date: Fri, 04 Sep 2020 22:47:09 GMT
Transfer-Encoding: chunked
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>
<ObjectLockConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
    <ObjectLockEnabled>Enabled</ObjectLockEnabled>
    <Rule>
        <DefaultRetention>
            <Mode>COMPLIANCE</Mode>
            <Years>6</Years>
        </DefaultRetention>
    </Rule>
</ObjectLockConfiguration>
```

#### PONER configuración de bloqueo de objeto

La solicitud DE CONFIGURACIÓN DE PUT Object Lock permite modificar el modo de retención

predeterminado y el período de retención predeterminado para un bloque que tiene el bloqueo de objetos activado. También es posible eliminar los ajustes de retención predeterminados previamente configurados.

Cuando se ingieren nuevas versiones de objetos en el bloque, se aplica el modo de retención predeterminado si `x-amz-object-lock-mode` no se ha especificado. El período de retención predeterminado se utiliza para calcular el valor de retener hasta la fecha if `x-amz-object-lock-retain-until-date` no se ha especificado.

Si el período de retención predeterminado se modifica tras recibir una versión de objeto, la fecha de retención hasta la de la versión del objeto sigue siendo la misma y no se vuelve a calcular con el nuevo período de retención predeterminado.

Debe tener el permiso s3:PutBucketObjectLockConfiguration o ser la raíz de la cuenta para completar esta operación.

La `Content-MD5` La cabecera de la solicitud se debe especificar en la solicitud PUT.

**Ejemplo de solicitud**

```
PUT /bucket?object-lock HTTP/1.1
Accept-Encoding: identity
Content-Length: 308
Host: host
Content-MD5: request header
User-Agent: s3sign/1.0.0 requests/2.24.0 python/3.8.2
X-Amz-Date: date
X-Amz-Content-SHA256: authorization string
Authorization: authorization string

<ObjectLockConfiguration>
    <ObjectLockEnabled>Enabled</ObjectLockEnabled>
    <Rule>
        <DefaultRetention>
            <Mode>COMPLIANCE</Mode>
            <Years>6</Years>
        </DefaultRetention>
    </Rule>
</ObjectLockConfiguration>
```

**Operaciones personalizadas en bloques**

## El sistema StorageGRID admite operaciones de bloques personalizadas que se añaden a la API DE REST de S3 y son específicas del sistema.

En la siguiente tabla, se enumeran las operaciones de bloque personalizadas que admite StorageGRID.

| Funcionamiento | Descripción | Si quiere más información |
|---|---|---|
| OBTENGA coherencia de bloques | Devuelve el nivel de coherencia que se aplica a un bloque determinado. | OBTENGA la solicitud de consistencia de bloque |
| PONGA la consistencia del cucharón | Establece el nivel de consistencia aplicado a un bloque determinado. | PONER solicitud de consistencia de bloque |
| HORA de último acceso al bloque DE GET | Devuelve si las actualizaciones del último tiempo de acceso están habilitadas o deshabilitadas para un bloque en particular. | GET Bucket última solicitud de tiempo de acceso |
| PUT Bucket última hora de acceso | Permite habilitar o deshabilitar las actualizaciones del último tiempo de acceso para un bloque en particular. | PUT Bucket última solicitud de tiempo de acceso |
| DELETE bucket metadata notification Configuration | Elimina el XML de configuración de notificación de metadatos asociado a un bloque en particular. | DELETE bucket metadata notification Configuration |
| OBTENGA la configuración de notificación de metadatos del bloque de datos | Devuelve el XML de configuración de notificación de metadatos asociado a un bloque determinado. | OBTENGA la solicitud de configuración de notificación de metadatos del bloque |
| PUT bucket metadata notification Configuration | Configura el servicio de notificación de metadatos para un bloque. | PUT bucket metadata notification Configuration |
| COLOQUE el cucharón con la configuración de cumplimiento | Obsoleto y no compatible: Ya no puede crear nuevos bloques con el cumplimiento de normativas habilitado. | Obsoleto: COLOQUE el cucharón con la configuración de cumplimiento |
| Obtenga el cumplimiento de normativas de Bucket | Obsoleto pero compatible: Devuelve la configuración de cumplimiento vigente para un bloque compatible existente. | Obsoleto: GET Bucket Compliance Request |
| CUMPLIR con la normativa de los bloques | Obsoleto pero compatible: Permite modificar la configuración de cumplimiento de un bloque compatible heredado. | Obsoleto: PUT Bucket Compliance Request |

**Información relacionada**

Se realizó un seguimiento de las operaciones de S3 en los registros de auditoría

## Operaciones en objetos

En esta sección se describe cómo el sistema StorageGRID implementa operaciones de la API DE REST de S3 para objetos.

Las siguientes condiciones se aplican a todas las operaciones de objeto:

- StorageGRID controles de consistencia son compatibles con todas las operaciones de los objetos, con la excepción de lo siguiente:
  - OBTENER ACL de objeto
  - `OPTIONS /`
  - PONER objeto legal
  - PUT Object retention
  - SELECCIONE Contenido de objeto
- Las solicitudes de clientes en conflicto, como dos clientes que escriben en la misma clave, se resuelven en función de las "últimas victorias". El plazo para la evaluación de "logros más recientes" se basa en cuándo el sistema StorageGRID completa una solicitud determinada, y no en cuándo los clientes S3 comienzan una operación.
- Todos los objetos de un bloque StorageGRID son propiedad del propietario del bloque, incluidos los objetos creados por un usuario anónimo o por otra cuenta.
- No se puede acceder a los objetos de datos procesados en el sistema StorageGRID a través de Swift a través de S3.

En la siguiente tabla se describe cómo StorageGRID implementa operaciones de objetos API DE REST de S3.

| Funcionamiento | Implementación |
| --- | --- |
| ELIMINAR objeto | Autenticación multifactor (MFA) y el encabezado de respuesta `x-amz-mfa` no son compatibles.<br><br>Al procesar una solicitud DE ELIMINACIÓN de objeto, StorageGRID intenta eliminar inmediatamente todas las copias del objeto de todas las ubicaciones almacenadas. Si se realiza correctamente, StorageGRID devuelve una respuesta al cliente inmediatamente. Si no se pueden eliminar todas las copias en un plazo de 30 segundos (por ejemplo, porque una ubicación no está disponible temporalmente), StorageGRID pone en cola las copias para su eliminación y luego indica que el cliente se ha realizado correctamente.<br><br>**Versioning**<br><br>Para eliminar una versión específica, el solicitante debe ser el propietario del bloque y utilizar el `versionId` subrecurso. El uso de este subrecurso elimina permanentemente la versión. Si la `versionId` corresponde a un marcador de borrado, el encabezado de respuesta `x-amz-delete-marker` se devuelve establecido en `true`.<br><br>• Si se elimina un objeto sin el `versionId` subrecurso en un bloque habilitado para la versión, da como resultado la generación de un marcador de borrado. La `versionId` para el marcador de borrado se devuelve mediante `x-amz-version-id` encabezado de respuesta, y el `x-amz-delete-marker` el encabezado de la respuesta se devuelve establecido en `true`.<br>• Si se elimina un objeto sin el `versionId` subrecurso en un bloque suspendido de la versión, se produce la eliminación permanente de una versión "nula" ya existente o un marcador de borrado "nula" y la generación de un nuevo marcador de borrado "nulo". La `x-amz-delete-marker` el encabezado de la respuesta se devuelve establecido en `true`.<br><br>**Nota**: En algunos casos, pueden existir varios marcadores de borrado para un objeto. |
| ELIMINAR varios objetos | Autenticación multifactor (MFA) y el encabezado de respuesta `x-amz-mfa` no son compatibles.<br><br>Se pueden eliminar varios objetos en el mismo mensaje de solicitud. |

| Funcionamiento | Implementación |
| --- | --- |
| ELIMINAR etiquetado de objetos | Utiliza la `tagging` subrecurso para quitar todas las etiquetas de un objeto. Se implementa con todo el comportamiento de la API DE REST de Amazon S3.<br><br>**Versioning**<br><br>Si la `versionId` el parámetro de consulta no se especifica en la solicitud, la operación elimina todas las etiquetas de la versión más reciente del objeto en un bloque con versiones. Si la versión actual del objeto es un marcador de borrado, se devuelve el estado "MetodNotAllowed" con el `x-amz-delete-marker` encabezado de respuesta establecido en `true`. |
| OBTENER objeto | OBTENER objeto |
| OBTENER ACL de objeto | Si se proporcionan las credenciales de acceso necesarias para la cuenta, la operación devuelve una respuesta positiva y el ID, DisplayName y permiso del propietario del objeto, lo que indica que el propietario tiene acceso completo al objeto. |
| OBTENER retención legal de objetos | Utilice el bloqueo de objetos de S3 |
| OBTENGA retención de objetos | Utilice el bloqueo de objetos de S3 |
| GET Object tagging | Utiliza la `tagging` subrecurso para devolver todas las etiquetas de un objeto. Se implementa con todo el comportamiento de la API DE REST de Amazon S3<br><br>**Versioning**<br><br>Si la `versionId` el parámetro de consulta no se especifica en la solicitud, la operación devuelve todas las etiquetas de la versión más reciente del objeto en un bloque con versiones. Si la versión actual del objeto es un marcador de borrado, se devuelve el estado "MetodNotAllowed" con el `x-amz-delete-marker` encabezado de respuesta establecido en `true`. |
| OBJETO HEAD | OBJETO HEAD |
| Restauración DE objetos posterior | Restauración DE objetos posterior |
| OBJETO PUT | OBJETO PUT |
| PONER objeto: Copiar | PONER objeto: Copiar |
| PONER objeto legal | Utilice el bloqueo de objetos de S3 |

| Funcionamiento | Implementación |
| --- | --- |
| PUT Object retention | Utilice el bloqueo de objetos de S3 |
| PUT Object tagging | Utiliza la `tagging` subrecurso para agregar un conjunto de etiquetas a un objeto existente. Se implementa con todo el comportamiento de la API DE REST de Amazon S3<br><br>**Límites de etiquetas de objeto**<br><br>Puede agregar etiquetas a nuevos objetos cuando los cargue o puede agregarlos a objetos existentes. Tanto StorageGRID como Amazon S3 admiten hasta 10 etiquetas por cada objeto. Las etiquetas asociadas a un objeto deben tener claves de etiqueta únicas. Una clave de etiqueta puede tener hasta 128 caracteres Unicode de longitud y los valores de etiqueta pueden tener hasta 256 caracteres Unicode de longitud. La clave y los valores distinguen entre mayúsculas y minúsculas.<br><br>**Actualizaciones de etiquetas y comportamiento de procesamiento**<br><br>Cuando se utiliza PUT Object tagging para actualizar las etiquetas de un objeto, StorageGRID no vuelve a procesar el objeto. Esto significa que no se utiliza la opción de comportamiento de ingesta especificada en la regla de ILM que coincide. Cualquier cambio en la ubicación del objeto que se active por la actualización se realice cuando los procesos de ILM normales se reevalúan el ILM en segundo plano.<br><br>Esto significa que si la regla ILM utiliza la opción estricta para el comportamiento de procesamiento, no se lleva a cabo ninguna acción si no se pueden realizar las ubicaciones de objetos necesarias (por ejemplo, porque una ubicación recientemente requerida no está disponible). El objeto actualizado conserva su ubicación actual hasta que sea posible la colocación requerida.<br><br>**Resolución de conflictos**<br><br>Las solicitudes de clientes en conflicto, como dos clientes que escriben en la misma clave, se resuelven en función de las "últimas victorias". El plazo para la evaluación de "logros más recientes" se basa en cuándo el sistema StorageGRID completa una solicitud determinada, y no en cuándo los clientes S3 comienzan una operación.<br><br>**Versioning**<br><br>Si la `versionId` el parámetro de consulta no se especifica en la solicitud, la operación agrega etiquetas a la versión más reciente del objeto en un bloque con versiones. Si la versión actual del objeto es un marcador de borrado, se devuelve el estado "MetodNotAllowed" con el `x-amz-delete-marker` encabezado de respuesta establecido en `true`. |

**Información relacionada**

Se realizó un seguimiento de las operaciones de S3 en los registros de auditoría

### Utilice el bloqueo de objetos de S3

Si la opción de configuración global de bloqueo de objetos S3 se encuentra habilitada para el sistema StorageGRID, puede crear bloques con la función S3 Object Lock habilitada y, a continuación, especificar los periodos de retención predeterminados para cada bloque o la configuración específica de retención hasta la fecha y la conservación legal para cada versión de objeto que añada a ese bloque.

El bloqueo de objetos S3 permite especificar configuraciones a nivel de objeto para evitar que los objetos se eliminen o se sobrescriban por un tiempo fijo o por tiempo indefinido.

La función de bloqueo de objetos StorageGRID S3 ofrece un único modo de retención equivalente al modo de cumplimiento de normativas Amazon S3. De forma predeterminada, cualquier usuario no puede sobrescribir ni eliminar una versión de objeto protegido. La función de bloqueo de objetos StorageGRID S3 no es compatible con un modo de gobierno y no permite a los usuarios con permisos especiales omitir la configuración de retención o eliminar objetos protegidos.

#### Habilite S3 Object Lock para bloque

Si la opción de configuración global de bloqueo de objetos S3 se encuentra habilitada para el sistema StorageGRID, también puede habilitar el bloqueo de objetos S3 al crear cada bloque. Es posible usar cualquiera de estos métodos:

- Cree el bloque con el Administrador de arrendatarios.

  Usar cuenta de inquilino

- Cree el segmento mediante una solicitud PUT Bucket con el `x-amz-bucket-object-lock-enabled` solicite el encabezado.

  Operaciones en bloques

No se puede añadir o deshabilitar el bloqueo de objetos de S3 después de crear el bloque. S3 Object Lock requiere el control de versiones de bloques, que se habilita automáticamente al crear el bloque.

Un bloque con S3 Object Lock habilitado puede contener una combinación de objetos con y sin la configuración de S3 Object Lock. StorageGRID admite períodos de retención predeterminados para los objetos en bloques de bloqueo de objetos de S3 y admite la operación DE bloque de configuración PUT Object Lock. La `s3:object-lock-remaining-retention-days` la clave de condición de política establece los períodos de retención mínimos y máximos permitidos para los objetos.

#### Determinar si se habilitó el bloqueo de objetos S3 para el bloque

Para determinar si la opción S3 Object Lock está habilitada, use OBTENER configuración de bloqueo de objeto solicitud.

#### Crear objeto con la configuración de Object Lock de S3

Para especificar la configuración de S3 Object Lock (bloqueo de objetos S3) al agregar una versión de objeto a un bloque que tenga habilitado el bloqueo de objetos S3, emita un objeto PUT, PUT Object - Copy o inicie una solicitud de carga de varias partes. Utilice los siguientes encabezados de solicitud.

Debe habilitar S3 Object Lock cuando se crea un bloque. No se puede añadir o deshabilitar el bloqueo de objetos de S3 después de crear un bloque.

- `x-amz-object-lock-mode`, Que debe ser DE CUMPLIMIENTO (distingue entre mayúsculas y minúsculas).

  Si especifica `x-amz-object-lock-mode`, también debe especificar `x-amz-object-lock-retain-until-date`.

- `x-amz-object-lock-retain-until-date`
  - El valor retener hasta la fecha debe tener el formato `2020-08-10T21:46:00Z`. Se permiten segundos fraccionarios, pero sólo se conservan 3 dígitos decimales (precisión de milisegundos). No se admiten otros formatos ISO 8601.
  - La fecha de retención debe ser futura.
- `x-amz-object-lock-legal-hold`

  Si la conservación legal está ACTIVADA (distingue entre mayúsculas y minúsculas), el objeto se colocará bajo una retención legal. Si se HA DESACTIVADO la retención legal, no se ha colocado ningún tipo de retención legal. Cualquier otro valor produce un error 400 Bad Request (InvalidArgument).

Si utiliza alguno de estos encabezados de solicitud, tenga en cuenta estas restricciones:

- La `Content-MD5` la cabecera de la solicitud es necesaria si la hay `x-amz-object-lock-*` El encabezado de la solicitud está presente en LA solicitud PUT Object. `Content-MD5` No es necesario PARA PONER objeto: Copiar o iniciar carga de varias partes.
- Si el bloque no tiene habilitado el bloqueo de objetos S3 y un `x-amz-object-lock-*` El encabezado de la solicitud está presente, se devuelve un error de solicitud incorrecta 400 (InvalidRequest).
- La solicitud PUT Object admite el uso de `x-amz-storage-class: REDUCED_REDUNDANCY` Para igualar el comportamiento de AWS. Sin embargo, cuando un objeto se procesa en un bucket con el bloqueo de objetos S3 habilitado, StorageGRID siempre ejecuta un procesamiento de compromiso doble.
- Una respuesta posterior A LA versión GET o HEAD Object incluirá los encabezados `x-amz-object-lock-mode`, `x-amz-object-lock-retain-until-date`, y. `x-amz-object-lock-legal-hold`, si está configurado y si el remitente de la solicitud tiene el correcto `s3:Get*` permisos.
- Una solicitud de ELIMINACIÓN de versión de objeto o ELIMINACIÓN de objetos no se realizará correctamente si se encuentra antes de la fecha de retención o si la retención legal está activada.

#### Actualice la configuración de bloqueo de objetos S3

Si necesita actualizar la configuración de retención legal o retención para una versión de objeto existente, puede realizar las siguientes operaciones de subrecursos de objeto:

- `PUT Object legal-hold`

  Si el nuevo valor de retención legal está ACTIVADO, el objeto se colocará bajo una retención legal. Si el valor de la retención legal está DESACTIVADO, se levanta la retención legal.

- `PUT Object retention`
  - El valor del modo debe ser DE CUMPLIMIENTO (distingue entre mayúsculas y minúsculas).

- El valor retener hasta la fecha debe tener el formato `2020-08-10T21:46:00Z`. Se permiten segundos fraccionarios, pero sólo se conservan 3 dígitos decimales (precisión de milisegundos). No se admiten otros formatos ISO 8601.
- Si una versión de objeto tiene una fecha de retención existente, sólo puede aumentarla. El nuevo valor debe ser el futuro.

**Información relacionada**

Gestión de objetos con ILM

Usar cuenta de inquilino

OBJETO PUT

PONER objeto: Copiar

Inicie la carga de varias partes

Control de versiones de objetos

"Guía del usuario de Amazon simple Storage Service: Uso del bloqueo de objetos de S3"

### Utilice S3 Select

StorageGRID admite las siguientes cláusulas, los tipos de datos y los operadores de AWS S3 Select SelectObjectContent.

> ⓘ No se admiten los elementos que no aparezcan.

Para obtener sintaxis, consulte SelectObjectContent. Para obtener más información acerca de S3 Select, consulte "Documentación de AWS para S3 Select".

Solo las cuentas de inquilino con S3 Select habilitado pueden emitir consultas de SelectObjectContent. Consulte Consideraciones y requisitos para usar S3 Select.

#### Cláusulas

- SELECCIONAR lista
- CLÁUSULA FROM
- Cláusula WHERE
- Cláusula LIMIT

#### Tipos de datos

- bool
- entero
- cadena
- flotante
- decimal, numérico
- fecha/hora

#### Operadores

### Operadores lógicos

- Y..
- NO
- O.

### Operadores de comparación

- <
- >
- ⇐
- >=
- =
- =
- <>
- !=
- ENTRE
- PULG

### Operadores de comparación de patrones

- COMO
- _
- %

### Operadores unitarios

- ES NULL
- NO ES NULL

### Operadores de matemáticas

- +
- -
- *
- /
- %

StorageGRID sigue la prioridad del operador de AWS S3 Select.

#### Funciones de agregados

- MEDIA()
- RECUENTO (*)

- MÁX.()
- MIN()
- SUMA()

#### Funciones condicionales

- CASO
- COALCE
- NULLIF

#### Funciones de conversión

- CAST (para tipo de datos compatible)

#### Funciones de fecha

- FECHA_AÑADIR
- DIF_FECHA
- EXTRAER
- TO_STRING
- TO_TIMESTAMP
- UTCNOW

#### Funciones de cadena

- CHAR_LENGTH, CHARACTER_LENGTH
- INFERIOR
- SUBCADENA
- RECORTE
- SUPERIOR

### Usar cifrado del servidor

El cifrado del lado del servidor le permite proteger los datos de objetos en reposo. StorageGRID cifra los datos mientras escribe el objeto y descifra los datos cuando accede al objeto.

Si desea utilizar el cifrado en el servidor, puede elegir una de las dos opciones mutuamente excluyentes, basándose en cómo se administran las claves de cifrado:

- **SSE (cifrado del lado del servidor con claves administradas por StorageGRID)**: Cuando se emite una solicitud de S3 para almacenar un objeto, StorageGRID cifra el objeto con una clave única. Cuando emite una solicitud S3 para recuperar el objeto, StorageGRID utiliza la clave almacenada para descifrar el objeto.
- **SSE-C (cifrado del lado del servidor con claves proporcionadas por el cliente)**: Cuando se emite una solicitud S3 para almacenar un objeto, se proporciona su propia clave de cifrado. Cuando recupera un objeto, proporciona la misma clave de cifrado que parte de la solicitud. Si las dos claves de cifrado coinciden, el objeto se descifra y se devuelven los datos del objeto.

Mientras que StorageGRID gestiona todas las operaciones de cifrado y descifrado de objetos, debe gestionar las claves de cifrado que proporcione.

> Las claves de cifrado que proporcione no se almacenan nunca. Si pierde una clave de cifrado, perderá el objeto correspondiente.

> Si un objeto está cifrado con SSE o SSE-C, se ignorará cualquier configuración de cifrado a nivel de bloque o de cuadrícula.

#### Utilice SSE

Para cifrar un objeto con una clave única administrada por StorageGRID, se utiliza el siguiente encabezado de solicitud:

```
x-amz-server-side-encryption
```

El encabezado de solicitud SSE es compatible con las siguientes operaciones de objeto:

- OBJETO PUT
- PONER objeto: Copiar
- Inicie la carga de varias partes

#### Utilice SSE-C

Para cifrar un objeto con una clave única que administra, se utilizan tres encabezados de solicitud:

| Solicite el encabezado | Descripción |
|---|---|
| `x-amz-server-side-encryption-customer-algorithm` | Especifique el algoritmo de cifrado. El valor de encabezado debe ser `AES256`. |
| `x-amz-server-side-encryption-customer-key` | Especifique la clave de cifrado que se utilizará para cifrar o descifrar el objeto. El valor de la clave debe estar codificado en base64 de 256 bits. |
| `x-amz-server-side-encryption-customer-key-MD5` | Especifique el resumen MD5 de la clave de cifrado según RFC 1321, que se utiliza para garantizar que la clave de cifrado se haya transmitido sin errores. El valor del resumen MD5 debe estar codificado en base64 de 128 bits. |

Las siguientes operaciones de objeto admiten los encabezados de solicitud de SSE-C:

- OBTENER objeto
- OBJETO HEAD
- OBJETO PUT
- PONER objeto: Copiar
- Inicie la carga de varias partes

- Cargar artículo
- Cargar pieza: Copiar

#### Consideraciones para utilizar el cifrado del servidor con claves proporcionadas por el cliente (SSE-C)

Antes de utilizar SSE-C, tenga en cuenta las siguientes consideraciones:

- Debe usar https.

> StorageGRID rechaza todas las solicitudes realizadas sobre http cuando se utilice SSE-C. Por cuestiones de seguridad, debe tener en cuenta cualquier clave que envíe accidentalmente mediante http para que se vea comprometida. Deseche la llave y gírela según corresponda.

- La ETag en la respuesta no es la MD5 de los datos del objeto.
- Debe gestionar la asignación de claves de cifrado a objetos. StorageGRID no almacena claves de cifrado. Usted es responsable del seguimiento de la clave de cifrado que usted proporciona para cada objeto.
- Si su bloque está habilitado para versionado, cada versión de objeto debe tener su propia clave de cifrado. Usted es responsable del seguimiento de la clave de cifrado utilizada para cada versión del objeto.
- Dado que gestiona las claves de cifrado en el cliente, también debe administrar cualquier protección adicional, como la rotación de claves, en el cliente.

> Las claves de cifrado que proporcione no se almacenan nunca. Si pierde una clave de cifrado, perderá el objeto correspondiente.

- Si la replicación de CloudMirror está configurada para el bloque, no podrá procesar objetos SSE-C. La operación de ingesta fallará.

#### Información relacionada

OBTENER objeto

OBJETO HEAD

OBJETO PUT

PONER objeto: Copiar

Inicie la carga de varias partes

Cargar artículo

Cargar pieza: Copiar

"Guía para desarrolladores de Amazon S3: Protección de datos mediante cifrado en el lado del servidor con claves de cifrado proporcionadas por el cliente (SSE-C)"

### OBTENER objeto

Puede usar la solicitud GET Object de S3 para recuperar un objeto de un bloque de S3.

#### OBJETOS GET y objetos de varias partes

Puede utilizar el `partNumber` parámetro de solicitud para recuperar una parte específica de un objeto de varias partes o segmentado. La `x-amz-mp-parts-count` el elemento de respuesta indica cuántas partes tiene el objeto.

Puede ajustar `partNumber` a 1 para objetos segmentados/multipartes y no segmentados/no multipartes; sin embargo, la `x-amz-mp-parts-count` el elemento de respuesta sólo se devuelve para objetos segmentados o multipartes.

#### Solicitar encabezados para el cifrado del servidor con claves de cifrado proporcionadas por el cliente (SSE-C)

Utilice los tres encabezados si el objeto está cifrado con una clave única que ha proporcionado.

- `x-amz-server-side-encryption-customer-algorithm`: Especificar `AES256`.
- `x-amz-server-side-encryption-customer-key`: Especifique la clave de cifrado del objeto.
- `x-amz-server-side-encryption-customer-key-MD5`: Especifique el resumen MD5 de la clave de cifrado del objeto.

> Las claves de cifrado que proporcione no se almacenan nunca. Si pierde una clave de cifrado, perderá el objeto correspondiente. Antes de utilizar las claves proporcionadas por el cliente para proteger los datos de objetos, revise las consideraciones que se incluyen en el apartado «"usar cifrado en el servidor".

#### Caracteres UTF-8 en los metadatos de usuario

StorageGRID no analiza ni interpreta caracteres UTF-8 escapados en los metadatos definidos por el usuario. LAS solicitudes GET de un objeto con caracteres UTF-8 escapados en metadatos definidos por el usuario no devuelven el `x-amz-missing-meta` encabezado si el nombre o valor de clave incluye caracteres no imprimibles.

#### Encabezado de solicitud no compatible

El siguiente encabezado de solicitud no es compatible y devuelve `XNotImplemented`:

- `x-amz-website-redirect-location`

#### Creación de versiones

Si es un `versionId` no se especifica el subrecurso, la operación busca la versión más reciente del objeto en un bloque con versiones. Si la versión actual del objeto es un marcador de borrado, se devuelve el estado "'no encontrado'" con la `x-amz-delete-marker` encabezado de respuesta establecido en `true`.

#### Comportamiento de OBTENER objeto para objetos de pool de almacenamiento en cloud

Si un objeto se ha almacenado en un Cloud Storage Pool (consulte las instrucciones para gestionar objetos con gestión del ciclo de vida de la información), el comportamiento de una solicitud GET Object depende del estado del objeto. Consulte «'HEAD Object'» para obtener más información.

> Si un objeto está almacenado en un Cloud Storage Pool y existen también una o varias copias del objeto en el grid, GET Object Requests intentará recuperar datos del grid, antes de recuperarlos del Cloud Storage Pool.

| Estado del objeto | Comportamiento DE GET Object |
| --- | --- |
| Objeto ingerido en StorageGRID pero aún no evaluado por ILM, u objeto almacenado en un pool de almacenamiento tradicional o utilizando código de borrado | `200 OK`<br><br>Se recupera una copia del objeto. |
| Objeto en el pool de almacenamiento en cloud pero todavía no ha realizado la transición a un estado no recuperable | `200 OK`<br><br>Se recupera una copia del objeto. |
| Objeto que ha pasado a un estado no recuperable | `403 Forbidden`, `InvalidObjectState`<br><br>Utilice una solicitud DE restauración POSTERIOR a objetos para restaurar el objeto en un estado recuperable. |
| Objeto en proceso de restauración a partir de un estado no recuperable | `403 Forbidden`, `InvalidObjectState`<br><br>Espere a que se complete la solicitud DE restauración DE objeto POSTERIOR. |
| Objeto completamente restaurado en el pool de almacenamiento en cloud | `200 OK`<br><br>Se recupera una copia del objeto. |

#### Objetos de varias partes o segmentados en un pool de almacenamiento en nube

Si cargó un objeto con varias partes o StorageGRID dividió un objeto grande en segmentos, StorageGRID determina si el objeto está disponible en el pool de almacenamiento en cloud al muestrear un subconjunto de las partes o segmentos del objeto. En algunos casos, es posible que UNA solicitud GET Object devuelva incorrectamente `200 OK` cuando algunas partes del objeto ya se han trasladado a un estado no recuperable o cuando algunas partes del objeto aún no se han restaurado.

En estos casos:

- La solicitud GET Object puede devolver algunos datos pero detenerse a mitad de camino a través de la transferencia.
- Una petición GET Object posterior podría devolver `403 Forbidden`.

**Información relacionada**

Usar cifrado del servidor

Gestión de objetos con ILM

Restauración DE objetos posterior

Se realizó un seguimiento de las operaciones de S3 en los registros de auditoría

## OBJETO HEAD

Puede usar la solicitud del ENCABEZADO Object de S3 para recuperar metadatos de un objeto sin devolver el objeto propiamente dicho. Si el objeto se almacena en un pool de almacenamiento en el cloud, puede usar HEAD Object para determinar el estado de transición del objeto.

### OBJETO DE CABECERA y objetos de varias partes

Puede utilizar el `partNumber` parámetro de solicitud para recuperar metadatos de una parte específica de un objeto de varias partes o segmentado. La `x-amz-mp-parts-count` el elemento de respuesta indica cuántas partes tiene el objeto.

Puede ajustar `partNumber` a 1 para objetos segmentados/multipartes y no segmentados/no multipartes; sin embargo, la `x-amz-mp-parts-count` el elemento de respuesta sólo se devuelve para objetos segmentados o multipartes.

### Solicitar encabezados para el cifrado del servidor con claves de cifrado proporcionadas por el cliente (SSE-C)

Utilice los tres encabezados si el objeto está cifrado con una clave única que ha proporcionado.

- `x-amz-server-side-encryption-customer-algorithm`: Especificar `AES256`.
- `x-amz-server-side-encryption-customer-key`: Especifique la clave de cifrado del objeto.
- `x-amz-server-side-encryption-customer-key-MD5`: Especifique el resumen MD5 de la clave de cifrado del objeto.

> Las claves de cifrado que proporcione no se almacenan nunca. Si pierde una clave de cifrado, perderá el objeto correspondiente. Antes de utilizar las claves proporcionadas por el cliente para proteger los datos de objetos, revise las consideraciones que se incluyen en el apartado «"usar cifrado en el servidor".

### Caracteres UTF-8 en los metadatos de usuario

StorageGRID no analiza ni interpreta caracteres UTF-8 escapados en los metadatos definidos por el usuario. Las solicitudes DE CABECERA de un objeto con caracteres UTF-8 escapados en metadatos definidos por el usuario no devuelven el `x-amz-missing-meta` encabezado si el nombre o valor de clave incluye caracteres no imprimibles.

### Encabezado de solicitud no compatible

El siguiente encabezado de solicitud no es compatible y devuelve `XNotImplemented`:

- `x-amz-website-redirect-location`

### Encabezados de respuesta para objetos de Cloud Storage Pool

Si el objeto se almacena en un grupo de almacenamiento en la nube (consulte las instrucciones para gestionar objetos con gestión del ciclo de vida de la información), se devuelven los siguientes encabezados de respuesta:

- `x-amz-storage-class: GLACIER`

- `x-amz-restore`

Los encabezados de respuesta proporcionan información sobre el estado de un objeto a medida que se mueve a un pool de almacenamiento en cloud, y que, opcionalmente, se realiza la transición a un estado no recuperable y se restaura.

| Estado del objeto | Respuesta al OBJETO PRINCIPAL |
|---|---|
| Objeto ingerido en StorageGRID pero aún no evaluado por ILM, u objeto almacenado en un pool de almacenamiento tradicional o utilizando código de borrado | `200 OK` (No se devuelve ningún encabezado de respuesta especial). |
| Objeto en el pool de almacenamiento en cloud pero todavía no ha realizado la transición a un estado no recuperable | `200 OK`<br><br>`x-amz-storage-class: GLACIER`<br><br>`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"`<br><br>Hasta que el objeto se realice la transición a un estado no recuperable, el valor de `expiry-date` se configura a una hora distante en el futuro. El sistema StorageGRID no controla la hora exacta de la transición. |
| El objeto ha pasado a estar en estado no recuperable, pero también existe al menos una copia en la cuadrícula | `200 OK`<br><br>`x-amz-storage-class: GLACIER`<br><br>`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"`<br><br>Valor para `expiry-date` se configura a una hora distante en el futuro.<br><br>**Nota**: Si la copia de la cuadrícula no está disponible (por ejemplo, un nodo de almacenamiento está inactivo), debe emitir una solicitud DE restauración DE objetos POST para restaurar la copia desde el grupo de almacenamiento en la nube antes de poder recuperar el objeto correctamente. |
| El objeto ha pasado a un estado que no se puede recuperar y no existe ninguna copia en la cuadrícula | `200 OK`<br><br>`x-amz-storage-class: GLACIER` |

| Estado del objeto | Respuesta al OBJETO PRINCIPAL |
| --- | --- |
| Objeto en proceso de restauración a partir de un estado no recuperable | `200 OK`<br><br>`x-amz-storage-class: GLACIER`<br><br>`x-amz-restore: ongoing-request="true"` |
| Objeto completamente restaurado en el pool de almacenamiento en cloud | `200 OK`<br><br>`x-amz-storage-class: GLACIER`<br><br>`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2018 00:00:00 GMT"`<br><br>La `expiry-date` Indica si el objeto del Cloud Storage Pool regresará a un estado no recuperable. |

#### Objetos de varias partes o segmentos en el pool de almacenamiento en cloud

Si cargó un objeto con varias partes o StorageGRID dividió un objeto grande en segmentos, StorageGRID determina si el objeto está disponible en el pool de almacenamiento en cloud al muestrear un subconjunto de las partes o segmentos del objeto. En algunos casos, es posible que una solicitud HEAD Object devuelva incorrectamente `x-amz-restore: ongoing-request="false"` cuando algunas partes del objeto ya se han trasladado a un estado no recuperable o cuando algunas partes del objeto aún no se han restaurado.

#### Creación de versiones

Si es un `versionId` no se especifica el subrecurso, la operación busca la versión más reciente del objeto en un bloque con versiones. Si la versión actual del objeto es un marcador de borrado, se devuelve el estado "'no encontrado'" con la `x-amz-delete-marker` encabezado de respuesta establecido en `true`.

#### Información relacionada

Usar cifrado del servidor

Gestión de objetos con ILM

Restauración DE objetos posterior

Se realizó un seguimiento de las operaciones de S3 en los registros de auditoría

### Restauración DE objetos posterior

Puede usar la solicitud DE restauración DE objetos POST de S3 PARA restaurar un objeto almacenado en un pool de almacenamiento en cloud.

#### Tipo de solicitud admitido

StorageGRID solo admite solicitudes POSTERIORES a la restauración de objetos para restaurar un objeto. No admite la `SELECT` tipo de restauración. Seleccione solicitudes de devolución `XNotImplemented`.

#### Creación de versiones

Opcionalmente, especifique `versionId` para restaurar una versión específica de un objeto en un bloque con versiones. Si no especifica `versionId`, se restaura la versión más reciente del objeto

#### Comportamiento de la restauración POSTERIOR de objetos en objetos de Pool de almacenamiento en cloud

Si un objeto se ha almacenado en un Cloud Storage Pool (consulte las instrucciones para gestionar objetos con gestión del ciclo de vida de la información), una solicitud POSTERIOR de restauración de objetos tiene el siguiente comportamiento, en función del estado del objeto. Consulte «'HEAD Object'» para obtener más información.

> Si un objeto se almacena en un Cloud Storage Pool y existen también una o varias copias del objeto en la cuadrícula, no es necesario restaurar el objeto mediante la emisión de una solicitud DE restauración DE objetos POSTERIOR. En su lugar, la copia local se puede recuperar directamente, utilizando UNA solicitud GET Object.

| Estado del objeto | Comportamiento DE la restauración POSTERIOR de objetos |
|---|---|
| El objeto se ingiere en StorageGRID pero aún no se ha evaluado por ILM, o el objeto no está en un pool de almacenamiento cloud | `403 Forbidden`, `InvalidObjectState` |
| Objeto en el pool de almacenamiento en cloud pero todavía no ha realizado la transición a un estado no recuperable | `200 OK` No se han realizado cambios.<br><br>**Nota**: Antes de que un objeto haya pasado a un estado no recuperable, no puede cambiar su `expiry-date`. |
| Objeto que ha pasado a un estado no recuperable | `202 Accepted` Restaura una copia recuperable del objeto en el Pool de almacenamiento en la nube durante la cantidad de días especificada en el cuerpo de la solicitud. Al final de este período, el objeto se devuelve a un estado no recuperable.<br><br>Opcionalmente, utilice la `Tier` solicitar elemento para determinar cuánto tiempo tardará el trabajo de restauración en finalizar (`Expedited`, `Standard`, o. `Bulk`). Si no especifica `Tier`, la `Standard` se utiliza el nivel.<br><br>**Atención**: Si se ha realizado la transición de un objeto a S3 Glacier Deep Archive o el Cloud Storage Pool utiliza Azure Blob Storage, no puede restaurarlo con el `Expedited` nivel. Se devuelve el siguiente error `403 Forbidden`, `InvalidTier`: `Retrieval option is not supported by this storage class`. |

| Estado del objeto | Comportamiento DE la restauración POSTERIOR de objetos |
| --- | --- |
| Objeto en proceso de restauración a partir de un estado no recuperable | `409 Conflict`, `RestoreAlreadyInProgress` |
| Objeto completamente restaurado en el pool de almacenamiento en cloud | `200 OK`<br><br>**Nota:** Si un objeto ha sido restaurado a un estado recuperable, usted puede cambiar su `expiry-date` Volviendo a emitir la solicitud DE restauración DE objeto POSTERIOR con un nuevo valor para `Days`. La fecha de restauración se actualiza en relación con la hora de la solicitud. |

**Información relacionada**

Gestión de objetos con ILM

OBJETO HEAD

Se realizó un seguimiento de las operaciones de S3 en los registros de auditoría

## OBJETO PUT

Puede usar la solicitud PUT Object de S3 para añadir un objeto a un bloque.

### Resolver conflictos

Las solicitudes de clientes en conflicto, como dos clientes que escriben en la misma clave, se resuelven en función de las "últimas victorias". El plazo para la evaluación de "logros más recientes" se basa en cuándo el sistema StorageGRID completa una solicitud determinada, y no en cuándo los clientes de S3 comienzan una operación.

### Tamaño del objeto

El tamaño máximo *recomendado* para una única operación PUT Object es de 5 GIB (5,368,709,120 bytes). Si tiene objetos que sean mayores de 5 GIB, utilice la carga de varias partes en su lugar.

> En StorageGRID 11.6, el tamaño máximo *admitido* para una única operación PUT Object es de 5 TIB (5,497,558,138,880 bytes). Sin embargo, la alerta * S3 PUT Object size demasiado grande* se activará si intenta cargar un objeto que supere los 5 GIB.

### Tamaño de los metadatos del usuario

Amazon S3 limita el tamaño de los metadatos definidos por el usuario dentro de cada encabezado de solicitud PUT a 2 KB. StorageGRID limita los metadatos de usuario a 24 KiB. El tamaño de los metadatos definidos por el usuario se mide tomando la suma del número de bytes de la codificación UTF-8 de cada clave y valor.

### Caracteres UTF-8 en los metadatos de usuario

Si una solicitud incluye (no escapadas) valores UTF-8 en el nombre de clave o el valor de los metadatos definidos por el usuario, el comportamiento de StorageGRID no está definido.

StorageGRID no analiza ni interpreta los caracteres UTF-8 escapados incluidos en el nombre de clave o el valor de los metadatos definidos por el usuario. Los caracteres UTF-8 que se han escapado se tratan como caracteres ASCII:

- LAS solicitudes PUT, PUT Object-Copy, GET y HEAD se realizan correctamente si los metadatos definidos por el usuario incluyen caracteres UTF-8 que se han escapado.
- StorageGRID no devuelve el `x-amz-missing-meta` encabezado si el valor interpretado del nombre o valor de clave incluye caracteres no imprimibles.

#### Límites de etiqueta de objeto

Puede agregar etiquetas a nuevos objetos cuando los cargue o puede agregarlos a objetos existentes. Tanto StorageGRID como Amazon S3 admiten hasta 10 etiquetas por cada objeto. Las etiquetas asociadas a un objeto deben tener claves de etiqueta únicas. Una clave de etiqueta puede tener hasta 128 caracteres Unicode de longitud y los valores de etiqueta pueden tener hasta 256 caracteres Unicode de longitud. La clave y los valores distinguen entre mayúsculas y minúsculas.

#### Propiedad del objeto

En StorageGRID, todos los objetos son propiedad de la cuenta de propietario del bloque, incluidos los objetos creados por una cuenta que no sea propietaria o un usuario anónimo.

#### Encabezados de solicitud admitidos

Se admiten los siguientes encabezados de solicitud:

- `Cache-Control`
- `Content-Disposition`
- `Content-Encoding`

  Al especificar `aws-chunked` para `Content-Encoding`StorageGRID no verifica los siguientes elementos:

  - StorageGRID no verifica el `chunk-signature` contra los datos del fragmento.
  - StorageGRID no verifica el valor indicado para `x-amz-decoded-content-length` contra el objeto.
- `Content-Language`
- `Content-Length`
- `Content-MD5`
- `Content-Type`
- `Expires`
- `Transfer-Encoding`

  La codificación de transferencia con chunked es compatible si `aws-chunked` también se utiliza la firma de carga útil.

- `x-amz-meta-`, seguido de un par nombre-valor que contiene metadatos definidos por el usuario.

  Cuando especifique la pareja nombre-valor para los metadatos definidos por el usuario, utilice este formato general:

```
x-amz-meta-name: value
```

Si desea utilizar la opción **tiempo de creación definido por el usuario** como tiempo de referencia para una regla de ILM, debe utilizar `creation-time` como nombre de los metadatos que registran cuando se creó el objeto. Por ejemplo:

```
x-amz-meta-creation-time: 1443399726
```

Valor para `creation-time` Se evalúa como segundos desde el 1 de enero de 1970.

> Una regla de ILM no puede utilizar un **tiempo de creación definido por el usuario** para el tiempo de referencia y las opciones equilibradas o estrictas para el comportamiento de procesamiento. Se devuelve un error cuando se crea la regla de ILM.

- `x-amz-tagging`
- Encabezados de solicitud de bloqueo de objetos de S3
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    Si se realiza una solicitud sin estos encabezados, la configuración de retención predeterminada del bloque se utiliza para calcular la versión del objeto mantener hasta la fecha.

  Utilice el bloqueo de objetos de S3
- Encabezados de solicitud SSE:
  - `x-amz-server-side-encryption`
  - `x-amz-server-side-encryption-customer-key-MD5`
  - `x-amz-server-side-encryption-customer-key`
  - `x-amz-server-side-encryption-customer-algorithm`

    Consulte Solicitar encabezados para el cifrado del servidor

#### Encabezados de solicitud no compatibles

No se admiten los siguientes encabezados de solicitud:

- La `x-amz-acl` no se admite el encabezado de la solicitud.
- La `x-amz-website-redirect-location` el encabezado de la solicitud no es compatible y devuelve `XNotImplemented`.

#### Opciones para clase de almacenamiento

La `x-amz-storage-class` se admite el encabezado de la solicitud. El valor enviado para `x-amz-`

`storage-class` Afecta la forma en que StorageGRID protege los datos de objetos durante el procesamiento y no cuántas copias persistentes del objeto se almacenan en el sistema StorageGRID (determinado por ILM).

Si la regla de ILM que coincide con un objeto ingerido utiliza la opción estricta para el comportamiento de la ingesta, la `x-amz-storage-class` el encabezado no tiene efecto.

Se pueden utilizar los siguientes valores para `x-amz-storage-class`:

- `STANDARD` (Predeterminado)
  - **Commit** doble: Si la regla ILM especifica la opción COMMIT doble para el comportamiento de procesamiento, tan pronto como un objeto se ingiere una segunda copia de ese objeto se crea y se distribuye a un nodo de almacenamiento diferente (COMMIT doble). Cuando se evalúa el ILM, StorageGRID determina si estas copias provisionales iniciales satisfacen las instrucciones de colocación en la regla. Si no lo hacen, es posible que sea necesario realizar nuevas copias de objetos en ubicaciones diferentes y que sea necesario eliminar las copias provisionales iniciales.
  - **Balanceado**: Si la regla ILM especifica la opción equilibrada y StorageGRID no puede realizar inmediatamente todas las copias especificadas en la regla, StorageGRID realiza dos copias provisionales en nodos de almacenamiento diferentes.

    Si StorageGRID puede crear inmediatamente todas las copias de objeto especificadas en la regla de ILM (ubicación síncrona), la `x-amz-storage-class` el encabezado no tiene efecto.

- `REDUCED_REDUNDANCY`
  - **Commit doble**: Si la regla ILM especifica la opción COMMIT doble para el comportamiento de la ingesta, StorageGRID crea una única copia provisional mientras se ingiere el objeto (COMMIT único).
  - **Balanceado**: Si la regla ILM especifica la opción equilibrada, StorageGRID realiza una única copia provisional sólo si el sistema no puede hacer inmediatamente todas las copias especificadas en la regla. Si StorageGRID puede realizar una colocación síncrona, este encabezado no tiene ningún efecto. La `REDUCED_REDUNDANCY` Se recomienda utilizar la opción cuando la regla de ILM que coincide con el objeto crea una única copia replicada. En este caso, utilizar `REDUCED_REDUNDANCY` elimina la creación y eliminación innecesarias de una copia de objetos adicional en cada operación de procesamiento.

  Con el `REDUCED_REDUNDANCY` la opción no se recomienda en otras circunstancias. `REDUCED_REDUNDANCY` aumenta el riesgo de pérdida de datos de objetos durante el procesamiento. Por ejemplo, puede perder datos si la única copia se almacena inicialmente en un nodo de almacenamiento que falla antes de que se pueda realizar la evaluación de ILM.

**Atención**: Tener sólo una copia replicada durante cualquier período de tiempo pone los datos en riesgo de pérdida permanente. Si sólo existe una copia replicada de un objeto, éste se pierde si falla un nodo de almacenamiento o tiene un error importante. También perderá temporalmente el acceso al objeto durante procedimientos de mantenimiento, como las actualizaciones.

Especificando `REDUCED_REDUNDANCY` sólo afecta al número de copias que se crean cuando un objeto se ingiere por primera vez. No afecta al número de copias del objeto que se realizan cuando el objeto se evalúa mediante la política de ILM activa y no provoca que los datos se almacenen en niveles inferiores de redundancia en el sistema StorageGRID.

**Nota**: Si está ingiriendo un objeto en un cubo con el bloqueo de objetos S3 activado, el `REDUCED_REDUNDANCY` opción ignorada. Si está ingiriendo un objeto en un bloque compatible heredado, el `REDUCED_REDUNDANCY` opción devuelve un error. StorageGRID siempre realizará una ingesta con doble confirmación para garantizar que se cumplan los requisitos de cumplimiento.

#### Solicitar encabezados para el cifrado del servidor

Puede utilizar los siguientes encabezados de solicitud para cifrar un objeto con cifrado del servidor. Las opciones SSE y SSE-C son mutuamente excluyentes.

- **SSE**: Utilice el siguiente encabezado si desea cifrar el objeto con una clave única gestionada por StorageGRID.
  - `x-amz-server-side-encryption`
- **SSE-C**: Utilice los tres encabezados si desea cifrar el objeto con una clave única que proporciona y administra.
  - `x-amz-server-side-encryption-customer-algorithm`: Especificar `AES256`.
  - `x-amz-server-side-encryption-customer-key`: Especifique la clave de cifrado para el nuevo objeto.
  - `x-amz-server-side-encryption-customer-key-MD5`: Especifique el resumen MD5 de la clave de cifrado del nuevo objeto.

**Atención:** las claves de cifrado que usted proporciona nunca se almacenan. Si pierde una clave de cifrado, perderá el objeto correspondiente. Antes de utilizar las claves proporcionadas por el cliente para proteger los datos de objetos, revise las consideraciones que se incluyen en el apartado «"usar cifrado en el servidor".

**Nota:** Si un objeto está cifrado con SSE o SSE-C, se ignorará cualquier configuración de cifrado a nivel de bloque o de cuadrícula.

#### Creación de versiones

Si el control de versiones está habilitado para un bloque, un valor único `versionId` se genera automáticamente para la versión del objeto almacenado. Este `versionId` también se devuelve en la respuesta mediante el `x-amz-version-id` encabezado de respuesta.

Si se suspende el control de versiones, la versión del objeto se almacena con un valor nulo `versionId` y si ya existe una versión nula, se sobrescribirá.

#### Información relacionada

Gestión de objetos con ILM

Operaciones en bloques

Se realizó un seguimiento de las operaciones de S3 en los registros de auditoría

Usar cifrado del servidor

Cómo se pueden configurar las conexiones de clientes

### PONER objeto: Copiar

Puede usar la solicitud PUT Object - Copy de S3 para crear una copia de un objeto que ya está almacenado en S3. UNA operación PONER objeto - copia es la misma que realizar UNA GET y LUEGO UN PUT.

#### Resolver conflictos

Las solicitudes de clientes en conflicto, como dos clientes que escriben en la misma clave, se resuelven en función de las "últimas victorias". El plazo para la evaluación de "logros más recientes" se basa en cuándo el sistema StorageGRID completa una solicitud determinada, y no en cuándo los clientes de S3 comienzan una operación.

#### Tamaño del objeto

El tamaño máximo *recomendado* para una única operación PUT Object es de 5 GIB (5,368,709,120 bytes). Si tiene objetos que sean mayores de 5 GIB, utilice la carga de varias partes en su lugar.

> En StorageGRID 11.6, el tamaño máximo *admitido* para una única operación PUT Object es de 5 TIB (5,497,558,138,880 bytes). Sin embargo, la alerta * S3 PUT Object size demasiado grande* se activará si intenta cargar un objeto que supere los 5 GIB.

#### Caracteres UTF-8 en los metadatos de usuario

Si una solicitud incluye (no escapadas) valores UTF-8 en el nombre de clave o el valor de los metadatos definidos por el usuario, el comportamiento de StorageGRID no está definido.

StorageGRID no analiza ni interpreta los caracteres UTF-8 escapados incluidos en el nombre de clave o el valor de los metadatos definidos por el usuario. Los caracteres UTF-8 que se han escapado se tratan como caracteres ASCII:

- Las solicitudes se realizan correctamente si los metadatos definidos por el usuario incluyen caracteres UTF-8 que se han escapado.
- StorageGRID no devuelve el `x-amz-missing-meta` encabezado si el valor interpretado del nombre o valor de clave incluye caracteres no imprimibles.

#### Encabezados de solicitud admitidos

Se admiten los siguientes encabezados de solicitud:

- `Content-Type`
- `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-meta-`, seguido de un par nombre-valor que contiene metadatos definidos por el usuario
- `x-amz-metadata-directive`: El valor predeterminado es `COPY`, que permite copiar el objeto y los metadatos asociados.

  Puede especificar `REPLACE` para sobrescribir los metadatos existentes al copiar el objeto o actualizar los metadatos del objeto.

- `x-amz-storage-class`
- `x-amz-tagging-directive`: El valor predeterminado es `COPY`, que le permite copiar el objeto y todas

las etiquetas.

Puede especificar `REPLACE` para sobrescribir las etiquetas existentes al copiar el objeto o actualizar las etiquetas.

- Encabezados de solicitud de bloqueo de objetos S3:
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    Si se realiza una solicitud sin estos encabezados, la configuración de retención predeterminada del bloque se utiliza para calcular la versión del objeto mantener hasta la fecha.

  Utilice el bloqueo de objetos de S3

- Encabezados de solicitud SSE:
  - `x-amz-copy-source-server-side-encryption-customer-algorithm`
  - `x-amz-copy-source-server-side-encryption-customer-key`
  - `x-amz-copy-source-server-side-encryption-customer-key-MD5`
  - `x-amz-server-side-encryption`
  - `x-amz-server-side-encryption-customer-key-MD5`
  - `x-amz-server-side-encryption-customer-key`
  - `x-amz-server-side-encryption-customer-algorithm`

    Consulte Solicitar encabezados para el cifrado del servidor

#### Encabezados de solicitud no compatibles

No se admiten los siguientes encabezados de solicitud:

- `Cache-Control`
- `Content-Disposition`
- `Content-Encoding`
- `Content-Language`
- `Expires`
- `x-amz-website-redirect-location`

#### Opciones para clase de almacenamiento

La `x-amz-storage-class` Se admite el encabezado de la solicitud y afecta al número de copias de objetos que crea StorageGRID si la regla de ILM coincidente especifica un comportamiento de ingesta de COMMIT doble o de equilibrado.

- `STANDARD`

(Predeterminado) especifica una operación de procesamiento de confirmación doble cuando la regla ILM utiliza la opción Commit doble o cuando la opción equilibrada vuelve a crear copias provisionales.

- `REDUCED_REDUNDANCY`

  Especifica una operación de procesamiento de confirmación única cuando la regla de ILM utiliza la opción Commit doble o cuando la opción equilibrada vuelve a crear copias provisionales.

  > Si va a procesar un objeto en un bloque con el bloqueo de objetos S3 habilitado, el `REDUCED_REDUNDANCY` opción ignorada. Si está ingiriendo un objeto en un bloque compatible heredado, el `REDUCED_REDUNDANCY` opción devuelve un error. StorageGRID siempre realizará una ingesta con doble confirmación para garantizar que se cumplan los requisitos de cumplimiento.

#### Uso de x-amz-copy-source en PUT Object - Copy

Si el bloque de origen y la clave, especificados en la `x-amz-copy-source` header, son diferentes del bloque y la clave de destino, se escribe una copia de los datos del objeto de origen en el destino.

Si el origen y el destino coinciden, y la `x-amz-metadata-directive` el encabezado se especifica como `REPLACE`, los metadatos del objeto se actualizan con los valores de metadatos proporcionados en la solicitud. En este caso, StorageGRID no vuelve a procesar el objeto. Esto tiene dos consecuencias importantes:

- No se puede utilizar PONER objeto - Copiar para cifrar un objeto existente en su lugar ni para cambiar el cifrado de un objeto existente en su lugar. Si proporciona el `x-amz-server-side-encryption` cabecera o la `x-amz-server-side-encryption-customer-algorithm` Encabezamiento, StorageGRID rechaza la solicitud y devuelve `XNotImplemented`.
- No se utiliza la opción de comportamiento de procesamiento especificado en la regla de ILM que coincida. Cualquier cambio en la ubicación del objeto que se active por la actualización se realice cuando los procesos de ILM normales se reevalúan el ILM en segundo plano.

  Esto significa que si la regla ILM utiliza la opción estricta para el comportamiento de procesamiento, no se lleva a cabo ninguna acción si no se pueden realizar las ubicaciones de objetos necesarias (por ejemplo, porque una ubicación recientemente requerida no está disponible). El objeto actualizado conserva su ubicación actual hasta que sea posible la colocación requerida.

#### Solicitar encabezados para el cifrado del servidor

Si utiliza cifrado del servidor, los encabezados de solicitud que proporcione dependerán de si el objeto de origen está cifrado y de si planea cifrar el objeto de destino.

- Si el objeto de origen se cifra utilizando una clave proporcionada por el cliente (SSE-C), debe incluir los tres encabezados siguientes en LA solicitud PUT Object - Copy, para que el objeto se pueda descifrar y copiar a continuación:
  - `x-amz-copy-source-server-side-encryption-customer-algorithm` Especifique `AES256`.
  - `x-amz-copy-source-server-side-encryption-customer-key` Especifique la clave de cifrado que proporcionó cuando creó el objeto de origen.
  - `x-amz-copy-source-server-side-encryption-customer-key-MD5`: Especifique el resumen MD5 que proporcionó cuando creó el objeto de origen.
- Si desea cifrar el objeto de destino (la copia) con una clave única que proporciona y administra, incluya los

tres encabezados siguientes:

- `x-amz-server-side-encryption-customer-algorithm`: Especificar `AES256`.
- `x-amz-server-side-encryption-customer-key`: Especifique una nueva clave de cifrado para el objeto de destino.
- `x-amz-server-side-encryption-customer-key-MD5`: Especifique el resumen MD5 de la nueva clave de cifrado.

**Atención:** las claves de cifrado que usted proporciona nunca se almacenan. Si pierde una clave de cifrado, perderá el objeto correspondiente. Antes de utilizar las claves proporcionadas por el cliente para proteger los datos de objetos, revise las consideraciones que se incluyen en el apartado «"usar cifrado en el servidor".

- Si desea cifrar el objeto de destino (la copia) con una clave única administrada por StorageGRID (SSE), incluya este encabezado en LA solicitud DE PUT Object - Copy:
  - `x-amz-server-side-encryption`

**Nota:** la `server-side-encryption` el valor del objeto no se puede actualizar. En su lugar, haga una copia con un nuevo `server-side-encryption` valor con `x-amz-metadata-directive`: `REPLACE`.

#### Creación de versiones

Si se crea una versión del contenedor de origen, puede utilizar `x-amz-copy-source` encabezado para copiar la versión más reciente de un objeto. Para copiar una versión específica de un objeto, debe especificar explícitamente la versión que desea copiar mediante `versionId` subrecurso. Si se crea una versión del bloque de destino, la versión generada se devuelve en el `x-amz-version-id` encabezado de respuesta. Si se suspende el control de versiones para el bloque de destino, entonces `x-amz-version-id` devuelve un valor «'null'».

#### Información relacionada

Gestión de objetos con ILM

Usar cifrado del servidor

Se realizó un seguimiento de las operaciones de S3 en los registros de auditoría

OBJETO PUT

### SelectObjectContent

Puede utilizar la solicitud S3 SelectObjectContent para filtrar el contenido de un objeto S3 en función de una simple instrucción SQL.

Para obtener más información, consulte "Documentación de AWS para SelectObjectContent".

#### Lo que necesitará

- La cuenta de inquilino tiene el permiso de S3 Select.
- Ya tienes `s3:GetObject` permiso para el objeto al que desea consultar.
- El objeto que desea consultar tiene el formato CSV o es un archivo comprimido GZIP o BZIP2 que contiene un archivo con formato CSV.
- La expresión SQL tiene una longitud máxima de 256 KB.

- Cualquier registro de la entrada o de los resultados tiene una longitud máxima de 1 MIB.

**Ejemplo de sintaxis de solicitud**

```
POST /{Key+}?select&select-type=2 HTTP/1.1
Host: Bucket.s3.abc-company.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<SelectObjectContentRequest xmlns="http://s3.amazonaws.com/doc/2006-03-
01/">
   <Expression>string</Expression>
   <ExpressionType>string</ExpressionType>
   <RequestProgress>
      <Enabled>boolean</Enabled>
   </RequestProgress>
   <InputSerialization>
      <CompressionType>GZIP</CompressionType>
      <CSV>
         <AllowQuotedRecordDelimiter>boolean</AllowQuotedRecordDelimiter>
         <Comments>#</Comments>
         <FieldDelimiter>\t</FieldDelimiter>
         <FileHeaderInfo>USE</FileHeaderInfo>
         <QuoteCharacter>'</QuoteCharacter>
         <QuoteEscapeCharacter>\\</QuoteEscapeCharacter>
         <RecordDelimiter>\n</RecordDelimiter>
      </CSV>
   </InputSerialization>
   <OutputSerialization>
      <CSV>
         <FieldDelimiter>string</FieldDelimiter>
         <QuoteCharacter>string</QuoteCharacter>
         <QuoteEscapeCharacter>string</QuoteEscapeCharacter>
         <QuoteFields>string</QuoteFields>
         <RecordDelimiter>string</RecordDelimiter>
      </CSV>
   </OutputSerialization>
   <ScanRange>
      <End>long</End>
      <Start>long</Start>
   </ScanRange>
</SelectObjectContentRequest>
```

**Ejemplo de consulta SQL**

Esta consulta obtiene el nombre del estado, 2010 poblaciones, 2015 poblaciones estimadas y el porcentaje de cambio con respecto a los datos del censo estadounidense. Los registros del archivo que no son estados se

omiten.

```
SELECT STNAME, CENSUS2010POP, POPESTIMATE2015, CAST((POPESTIMATE2015 -
CENSUS2010POP) AS DECIMAL) / CENSUS2010POP * 100.0 FROM S3Object WHERE
NAME = STNAME
```

Las primeras líneas del archivo a consultar, `SUB-EST2020_ALL.csv`, mire como esto:

```
SUMLEV,STATE,COUNTY,PLACE,COUSUB,CONCIT,PRIMGEO_FLAG,FUNCSTAT,NAME,STNAME,
CENSUS2010POP,
ESTIMATESBASE2010,POPESTIMATE2010,POPESTIMATE2011,POPESTIMATE2012,POPESTIM
ATE2013,POPESTIMATE2014,
POPESTIMATE2015,POPESTIMATE2016,POPESTIMATE2017,POPESTIMATE2018,POPESTIMAT
E2019,POPESTIMATE042020,
POPESTIMATE2020
040,01,000,00000,00000,00000,0,A,Alabama,Alabama,4779736,4780118,4785514,4
799642,4816632,4831586,
4843737,4854803,4866824,4877989,4891628,4907965,4920706,4921532
162,01,000,00124,00000,00000,0,A,Abbeville
city,Alabama,2688,2705,2699,2694,2645,2629,2610,2602,
2587,2578,2565,2555,2555,2553
162,01,000,00460,00000,00000,0,A,Adamsville
city,Alabama,4522,4487,4481,4474,4453,4430,4399,4371,
4335,4304,4285,4254,4224,4211
162,01,000,00484,00000,00000,0,A,Addison
town,Alabama,758,754,751,750,745,744,742,734,734,728,
725,723,719,717
```

**Ejemplo de uso de AWS-CLI**

```
aws s3api select-object-content --endpoint-url https://10.224.7.44:10443
--no-verify-ssl  --bucket 619c0755-9e38-42e0-a614-05064f74126d --key SUB-
EST2020_ALL.csv --expression-type SQL --input-serialization '{"CSV":
{"FileHeaderInfo": "USE", "Comments": "#", "QuoteEscapeCharacter": "\"",
"RecordDelimiter": "\n", "FieldDelimiter": ",", "QuoteCharacter": "\"",
"AllowQuotedRecordDelimiter": false}, "CompressionType": "NONE"}' --output
-serialization '{"CSV": {"QuoteFields": "ASNEEDED",
"QuoteEscapeCharacter": "#", "RecordDelimiter": "\n", "FieldDelimiter":
",", "QuoteCharacter": "\""}}' --expression "SELECT STNAME, CENSUS2010POP,
POPESTIMATE2015, CAST((POPESTIMATE2015 - CENSUS2010POP) AS DECIMAL) /
CENSUS2010POP * 100.0 FROM S3Object WHERE NAME = STNAME" changes.csv
```

Las primeras líneas del archivo de salida, `changes.csv`, mire como esto:

```
Alabama,4779736,4854803,1.5705260708959658022953568983726297854
Alaska,710231,738430,3.9703983633493891424057806544631253775
Arizona,6392017,6832810,6.8959922978928247531256565807005832431
Arkansas,2915918,2979732,2.1884703204959810255295244928012378949
California,37253956,38904296,4.4299724839960620557988526104449148971
Colorado,5029196,5454328,8.4532796097030221132761578590295546246
```

## Operaciones para cargas de varias partes

En esta sección se describe cómo StorageGRID admite las operaciones para cargas de varias partes.

Las siguientes condiciones y notas se aplican a todas las operaciones de carga de varias partes:

- No debe exceder 1,000 cargas simultáneas de varias partes en un solo bloque, ya que los resultados de List Multipart cargan consultas para ese bloque pueden devolver resultados incompletos.
- StorageGRID aplica los límites de tamaño de AWS para piezas multiparte. Los clientes de S3 deben seguir estas directrices:
  - Cada parte de una carga de varias partes debe estar entre 5 MIB (5,242,880 bytes) y 5 GIB (5,368,709,120 bytes).
  - La última parte puede ser más pequeña que 5 MIB (5,242,880 bytes).
  - En general, los tamaños de las piezas deben ser lo más grandes posible. Por ejemplo, utilice tamaños de parte de 5 GIB para un objeto de 100 GIB. Dado que cada parte se considera un objeto único, el uso de tamaños de pieza grandes reduce la sobrecarga de metadatos de StorageGRID.
  - En el caso de objetos de menor tamaño de 5 GIB, considere usar la carga sin varias partes.
- ILM se evalúa para cada parte de un objeto de varias partes tal como se procesa y para el objeto como un todo cuando se completa la carga de varias partes, si la regla de ILM utiliza el comportamiento estricto o equilibrado del procesamiento. Debe saber cómo afecta esto a la ubicación de objetos y piezas:
  - Si ILM cambia mientras se carga varias partes de S3, es posible que cuando la carga de varias partes completa algunas partes del objeto no cumplan los requisitos actuales de ILM. Cualquier pieza que no se haya colocado correctamente se coloca en la cola de reevaluación de ILM y se mueve posteriormente a la ubicación correcta.
  - Al evaluar ILM para una pieza, StorageGRID filtra el tamaño de la pieza, no el tamaño del objeto. Esto significa que las partes de un objeto se pueden almacenar en ubicaciones que no cumplen los requisitos de ILM para el objeto como un todo. Por ejemplo, si una regla especifica que todos los objetos de 10 GB o más se almacenan en DC1 mientras que todos los objetos más pequeños se almacenan en DC2, al ingerir cada parte de 1 GB de una carga multiparte de 10 partes se almacena en DC2. Cuando se evalúa ILM para el objeto como un todo, todas las partes del objeto se mueven a DC1.
- Todas las operaciones de carga de varias partes admiten controles de coherencia de StorageGRID.
- Según sea necesario, puede utilizar el cifrado del servidor con cargas en varias partes. Para usar SSE (cifrado en el servidor con claves gestionadas por StorageGRID), incluye el `x-amz-server-side-encryption` Solicite el encabezado sólo en la solicitud Iniciar carga de varias partes. Para utilizar SSE-C (cifrado del servidor con claves proporcionadas por el cliente), debe especificar los mismos tres encabezados de solicitud de clave de cifrado en la solicitud de carga de varias partes iniciada y en cada solicitud de artículo de carga posterior.

| Funcionamiento | Implementación |
| --- | --- |
| Enumerar cargas de varias partes | Consulte Enumerar cargas de varias partes |
| Inicie la carga de varias partes | Consulte Inicie la carga de varias partes |
| Cargar artículo | Consulte Cargar artículo |
| Cargar pieza: Copiar | Consulte Cargar pieza: Copiar |
| Completar carga de varias partes | Consulte Completar carga de varias partes |
| Cancelar carga de varias partes | Se implementa con todo el comportamiento de la API DE REST de Amazon S3 |
| Enumerar piezas | Se implementa con todo el comportamiento de la API DE REST de Amazon S3 |

**Información relacionada**

- Controles de consistencia
- Usar cifrado del servidor

### Enumerar cargas de varias partes

La operación List Multipart carga enumera las cargas de varias partes en curso para un bloque.

Se admiten los siguientes parámetros de solicitud:

- `encoding-type`
- `max-uploads`
- `key-marker`
- `prefix`
- `upload-id-marker`

La `delimiter` el parámetro request no es compatible.

#### Creación de versiones

La carga de varias partes consiste en operaciones independientes para iniciar la carga, enumerar cargas, cargar piezas, ensamblar las piezas cargadas y completar la carga. Cuando se realiza la operación de carga de varias partes completa, ese es el punto en el que se crean objetos (y se crean versiones si procede).

### Inicie la carga de varias partes

La operación Iniciar carga de varias partes inicia una carga de varias partes para un objeto y devuelve un ID de carga.

La `x-amz-storage-class` se admite el encabezado de la solicitud. El valor enviado para `x-amz-storage-class` Afecta la forma en que StorageGRID protege los datos de objetos durante el procesamiento y no cuántas copias persistentes del objeto se almacenan en el sistema StorageGRID (determinado por ILM).

Si la regla de ILM que coincide con un objeto ingerido utiliza la opción estricta para el comportamiento de la ingesta, la `x-amz-storage-class` el encabezado no tiene efecto.

Se pueden utilizar los siguientes valores para `x-amz-storage-class`:

- `STANDARD` (Predeterminado)
  - **Commit** doble: Si la regla ILM especifica la opción COMMIT doble para el comportamiento de procesamiento, tan pronto como un objeto se ingiere una segunda copia de ese objeto se crea y se distribuye a un nodo de almacenamiento diferente (COMMIT doble). Cuando se evalúa el ILM, StorageGRID determina si estas copias provisionales iniciales satisfacen las instrucciones de colocación en la regla. Si no lo hacen, es posible que sea necesario realizar nuevas copias de objetos en ubicaciones diferentes y que sea necesario eliminar las copias provisionales iniciales.
  - **Balanceado**: Si la regla ILM especifica la opción equilibrada y StorageGRID no puede realizar inmediatamente todas las copias especificadas en la regla, StorageGRID realiza dos copias provisionales en nodos de almacenamiento diferentes.

    Si StorageGRID puede crear inmediatamente todas las copias de objeto especificadas en la regla de ILM (ubicación síncrona), la `x-amz-storage-class` el encabezado no tiene efecto.

- `REDUCED_REDUNDANCY`
  - **Commit doble**: Si la regla ILM especifica la opción COMMIT doble para el comportamiento de la ingesta, StorageGRID crea una única copia provisional mientras se ingiere el objeto (COMMIT único).
  - **Balanceado**: Si la regla ILM especifica la opción equilibrada, StorageGRID realiza una única copia provisional sólo si el sistema no puede hacer inmediatamente todas las copias especificadas en la regla. Si StorageGRID puede realizar una colocación síncrona, este encabezado no tiene ningún efecto. La `REDUCED_REDUNDANCY` Se recomienda utilizar la opción cuando la regla de ILM que coincide con el objeto crea una única copia replicada. En este caso, utilizar `REDUCED_REDUNDANCY` elimina la creación y eliminación innecesarias de una copia de objetos adicional en cada operación de procesamiento.

  Con el `REDUCED_REDUNDANCY` la opción no se recomienda en otras circunstancias. `REDUCED_REDUNDANCY` aumenta el riesgo de pérdida de datos de objetos durante el procesamiento. Por ejemplo, puede perder datos si la única copia se almacena inicialmente en un nodo de almacenamiento que falla antes de que se pueda realizar la evaluación de ILM.

**Atención**: Tener sólo una copia replicada durante cualquier período de tiempo pone los datos en riesgo de pérdida permanente. Si sólo existe una copia replicada de un objeto, éste se pierde si falla un nodo de almacenamiento o tiene un error importante. También perderá temporalmente el acceso al objeto durante procedimientos de mantenimiento, como las actualizaciones.

Especificando `REDUCED_REDUNDANCY` sólo afecta al número de copias que se crean cuando un objeto se ingiere por primera vez. No afecta al número de copias del objeto que se realizan cuando el objeto se evalúa mediante la política de ILM activa y no provoca que los datos se almacenen en niveles inferiores de redundancia en el sistema StorageGRID.

**Nota**: Si está ingiriendo un objeto en un cubo con el bloqueo de objetos S3 activado, el `REDUCED_REDUNDANCY` opción ignorada. Si está ingiriendo un objeto en un bloque compatible heredado, el

`REDUCED_REDUNDANCY` opción devuelve un error. StorageGRID siempre realizará una ingesta con doble confirmación para garantizar que se cumplan los requisitos de cumplimiento.

Se admiten los siguientes encabezados de solicitud:

- `Content-Type`
- `x-amz-meta-`, seguido de un par nombre-valor que contiene metadatos definidos por el usuario

  Cuando especifique la pareja nombre-valor para los metadatos definidos por el usuario, utilice este formato general:

  ```
  x-amz-meta-_name_: `value`
  ```

  Si desea utilizar la opción **tiempo de creación definido por el usuario** como tiempo de referencia para una regla de ILM, debe utilizar `creation-time` como nombre de los metadatos que registran cuando se creó el objeto. Por ejemplo:

  ```
  x-amz-meta-creation-time: 1443399726
  ```

  Valor para `creation-time` Se evalúa como segundos desde el 1 de enero de 1970.

  > Adición `creation-time` Como metadatos definidos por el usuario no se permite si va a agregar un objeto a un bloque que tiene la conformidad heredada habilitada. Se devolverá un error.

- Encabezados de solicitud de bloqueo de objetos S3:
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    Si se realiza una solicitud sin estos encabezados, la configuración de retención predeterminada del bloque se utiliza para calcular la versión del objeto mantener hasta la fecha.

  Uso del bloqueo de objetos de S3

- Encabezados de solicitud SSE:
  - `x-amz-server-side-encryption`
  - `x-amz-server-side-encryption-customer-key-MD5`
  - `x-amz-server-side-encryption-customer-key`
  - `x-amz-server-side-encryption-customer-algorithm`

    Solicitar encabezados para el cifrado del servidor

Para obtener información acerca de cómo StorageGRID maneja los caracteres UTF-8, consulte la documentación de PUT Object.

#### Solicitar encabezados para el cifrado del servidor

Puede utilizar los siguientes encabezados de solicitud para cifrar un objeto de varias partes con cifrado del servidor. Las opciones SSE y SSE-C son mutuamente excluyentes.

- **SSE**: Utilice el siguiente encabezado en la solicitud Iniciar carga de varias partes si desea cifrar el objeto con una clave única gestionada por StorageGRID. No especifique este encabezado en ninguna de las solicitudes de artículo de carga.
  - `x-amz-server-side-encryption`
- **SSE-C**: Utilice los tres encabezados de la solicitud de carga de varias partes iniciada (y en cada solicitud de artículo de carga posterior) si desea cifrar el objeto con una clave única que proporciona y gestiona.
  - `x-amz-server-side-encryption-customer-algorithm`: Especificar `AES256`.
  - `x-amz-server-side-encryption-customer-key`: Especifique la clave de cifrado para el nuevo objeto.
  - `x-amz-server-side-encryption-customer-key-MD5`: Especifique el resumen MD5 de la clave de cifrado del nuevo objeto.

**Atención:** las claves de cifrado que usted proporciona nunca se almacenan. Si pierde una clave de cifrado, perderá el objeto correspondiente. Antes de utilizar las claves proporcionadas por el cliente para proteger los datos de objetos, revise las consideraciones que se incluyen en el apartado «"usar cifrado en el servidor".

#### Encabezados de solicitud no compatibles

El siguiente encabezado de solicitud no es compatible y devuelve `XNotImplemented`

- `x-amz-website-redirect-location`

#### Creación de versiones

La carga de varias partes consiste en operaciones independientes para iniciar la carga, enumerar cargas, cargar piezas, ensamblar las piezas cargadas y completar la carga. Los objetos se crean (y se crean versiones si corresponde) cuando se realiza la operación de carga de varias partes completa.

**Información relacionada**

Gestión de objetos con ILM

Usar cifrado del servidor

OBJETO PUT

### Cargar artículo

La operación cargar pieza carga una pieza en una carga de varias partes para un objeto.

#### Encabezados de solicitud admitidos

Se admiten los siguientes encabezados de solicitud:

- `Content-Length`
- `Content-MD5`

#### Solicitar encabezados para el cifrado del servidor

Si ha especificado el cifrado SSE-C para la solicitud de carga de varias partes iniciada, también debe incluir los siguientes encabezados de solicitud en cada solicitud de artículo de carga:

- `x-amz-server-side-encryption-customer-algorithm`: Especificar `AES256`.
- `x-amz-server-side-encryption-customer-key`: Especifique la misma clave de cifrado que proporcionó en la solicitud Iniciar carga de varias partes.
- `x-amz-server-side-encryption-customer-key-MD5`: Especifique el mismo resumen MD5 que ha proporcionado en la solicitud Iniciar carga de varias partes.

> Las claves de cifrado que proporcione no se almacenan nunca. Si pierde una clave de cifrado, perderá el objeto correspondiente. Antes de utilizar las claves proporcionadas por el cliente para proteger los datos de objetos, revise las consideraciones que se incluyen en el apartado «"usar cifrado en el servidor".

#### Creación de versiones

La carga de varias partes consiste en operaciones independientes para iniciar la carga, enumerar cargas, cargar piezas, ensamblar las piezas cargadas y completar la carga. Los objetos se crean (y se crean versiones si corresponde) cuando se realiza la operación de carga de varias partes completa.

#### Información relacionada

Usar cifrado del servidor

### Cargar pieza: Copiar

## La operación cargar pieza - Copiar carga una parte de un objeto copiando datos de un objeto existente como origen de datos.

La operación cargar pieza - copia se implementa con todo el comportamiento de la API DE REST de Amazon S3.

Esta solicitud lee y escribe los datos del objeto especificados en `x-amz-copy-source-range` En el sistema StorageGRID.

Se admiten los siguientes encabezados 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`

#### Solicitar encabezados para el cifrado del servidor

Si ha especificado el cifrado SSE-C para la solicitud de carga de varias partes iniciada, también debe incluir los siguientes encabezados de solicitud en cada parte de carga - solicitud de copia:

- `x-amz-server-side-encryption-customer-algorithm`: Especificar `AES256`.
- `x-amz-server-side-encryption-customer-key`: Especifique la misma clave de cifrado que proporcionó en la solicitud Iniciar carga de varias partes.
- `x-amz-server-side-encryption-customer-key-MD5`: Especifique el mismo resumen MD5 que ha proporcionado en la solicitud Iniciar carga de varias partes.

Si el objeto de origen se cifra utilizando una clave proporcionada por el cliente (SSE-C), debe incluir los tres encabezados siguientes en la solicitud cargar pieza - Copiar, para que el objeto se pueda descifrar y copiar a continuación:

- `x-amz-copy-source-server-side-encryption-customer-algorithm`: Especificar `AES256`.
- `x-amz-copy-source-server-side-encryption-customer-key`: Especifique la clave de cifrado que proporcionó cuando creó el objeto de origen.
- `x-amz-copy-source-server-side-encryption-customer-key-MD5`: Especifique el resumen MD5 que proporcionó cuando creó el objeto de origen.

> Las claves de cifrado que proporcione no se almacenan nunca. Si pierde una clave de cifrado, perderá el objeto correspondiente. Antes de utilizar las claves proporcionadas por el cliente para proteger los datos de objetos, revise las consideraciones que se incluyen en el apartado «"usar cifrado en el servidor".

#### Creación de versiones

La carga de varias partes consiste en operaciones independientes para iniciar la carga, enumerar cargas, cargar piezas, ensamblar las piezas cargadas y completar la carga. Los objetos se crean (y se crean versiones si corresponde) cuando se realiza la operación de carga de varias partes completa.

### Completar carga de varias partes

La operación de carga de varias partes completa completa finaliza una carga de varias partes de un objeto mediante el montaje de las piezas previamente cargadas.

#### Resolver conflictos

Las solicitudes de clientes en conflicto, como dos clientes que escriben en la misma clave, se resuelven en función de las "últimas victorias". El plazo para la evaluación de "logros más recientes" se basa en cuándo el sistema StorageGRID completa una solicitud determinada, y no en cuándo los clientes de S3 comienzan una operación.

#### Solicitar encabezados

La `x-amz-storage-class` Se admite el encabezado de la solicitud y afecta al número de copias de objetos que crea StorageGRID si la regla de ILM coincidente especifica un comportamiento de ingesta de COMMIT doble o de equilibrado.

- `STANDARD`

  (Predeterminado) especifica una operación de procesamiento de confirmación doble cuando la regla ILM utiliza la opción Commit doble o cuando la opción equilibrada vuelve a crear copias provisionales.

- `REDUCED_REDUNDANCY`

Especifica una operación de procesamiento de confirmación única cuando la regla de ILM utiliza la opción Commit doble o cuando la opción equilibrada vuelve a crear copias provisionales.

> Si va a procesar un objeto en un bloque con el bloqueo de objetos S3 habilitado, el `REDUCED_REDUNDANCY` opción ignorada. Si está ingiriendo un objeto en un bloque compatible heredado, el `REDUCED_REDUNDANCY` opción devuelve un error. StorageGRID siempre realizará una ingesta con doble confirmación para garantizar que se cumplan los requisitos de cumplimiento.

> Si no se completa una carga de varias partes en un plazo de 15 días, la operación se Marca como inactiva y todos los datos asociados se eliminan del sistema.

> La `ETag` El valor devuelto no es una suma MD5 de los datos, sino que sigue a la implementación de API de Amazon S3 de `ETag` valor para objetos de varias piezas.

#### Creación de versiones

Esta operación completa una carga de varias partes. Si el control de versiones está habilitado para un bloque, la versión del objeto se crea al finalizar la carga de varias partes.

Si el control de versiones está habilitado para un bloque, un valor único `versionId` se genera automáticamente para la versión del objeto almacenado. Este `versionId` también se devuelve en la respuesta mediante el `x-amz-version-id` encabezado de respuesta.

Si se suspende el control de versiones, la versión del objeto se almacena con un valor nulo `versionId` y si ya existe una versión nula, se sobrescribirá.

> Cuando se habilita el control de versiones para un bloque, al completar una carga de varias partes siempre se crea una versión nueva, incluso si hay cargas simultáneas de varias partes completadas en la misma clave de objeto. Cuando el control de versiones no está habilitado para un bloque, es posible iniciar una carga de varias partes y, a continuación, hacer que se inicie y finalice otra carga de varias partes primero en la misma clave de objeto. En cubos sin versiones, la carga de varias partes que finaliza por última vez tiene prioridad.

#### Error en la replicación, notificación o notificación de metadatos

Si el bloque donde se produce la carga de varias partes está configurado para un servicio de plataforma, la carga de varias partes se realiza correctamente incluso si la acción de replicación o notificación asociada falla.

Si esto ocurre, se genera una alarma en el administrador de grid en eventos totales (SMTT). El mensaje Last Event muestra "'error al publicar notificaciones para la clave de objeto de nombre de bloque'" del último objeto cuya notificación ha fallado. (Para ver este mensaje, seleccione **NODES** > ***Storage Node*** > **Events**. Ver último evento en la parte superior de la tabla). Los mensajes de eventos también se muestran en la `/var/local/log/bycast-err.log`.

Un inquilino puede activar la replicación o notificación con errores actualizando los metadatos o las etiquetas del objeto. Un arrendatario puede volver a enviar los valores existentes para evitar realizar cambios no deseados.

**Información relacionada**

Gestión de objetos con ILM

## Respuestas de error

El sistema StorageGRID es compatible con todas las respuestas de error estándar de la API DE REST de S3 que se aplican. Además, la implementación de StorageGRID añade varias respuestas personalizadas.

**códigos de error API de S3 admitidos**

| Nombre | Estado de HTTP |
|---|---|
| ACCESSDENIED | 403 Prohibido |
| BadDigest | 400 solicitud incorrecta |
| BucketAlreadyExists | 409 conflicto |
| BucketNotEmpty | 409 conflicto |
| IncompleteBody | 400 solicitud incorrecta |
| Internalerror | 500 error de servidor interno |
| InvalidAccessKeyId | 403 Prohibido |
| InvalidArgument | 400 solicitud incorrecta |
| InvalidBucketName | 400 solicitud incorrecta |
| InvalidBucketState | 409 conflicto |
| InvalidDigest | 400 solicitud incorrecta |
| InvalidEncryptionAlgorithmError | 400 solicitud incorrecta |
| InvalidPart | 400 solicitud incorrecta |
| InvalidPartOrder | 400 solicitud incorrecta |
| InvalidRange | 416 rango solicitado no utilizable |
| InvalidRequest | 400 solicitud incorrecta |
| InvalidStorageClass | 400 solicitud incorrecta |
| InvalidTag | 400 solicitud incorrecta |

| Nombre | Estado de HTTP |
| --- | --- |
| InvalidURI | 400 solicitud incorrecta |
| KeyTooLong | 400 solicitud incorrecta |
| MalformedXML | 400 solicitud incorrecta |
| MetadataTooLarge | 400 solicitud incorrecta |
| MethodNotAllowed | 405 método no permitido |
| MissingContentLength | 411 longitud requerida |
| MissingRequestBodyError | 400 solicitud incorrecta |
| MissingSecurityHeader | 400 solicitud incorrecta |
| NoSuchBucket | 404 no encontrado |
| NoSuchKey | 404 no encontrado |
| NoSuchUpload | 404 no encontrado |
| NotImplimed | 501 no implementada |
| NoSuchBucketPolicy | 404 no encontrado |
| ObjectLockConfigurationNotFoundError | 404 no encontrado |
| Error de precondiciónError | Error de condición 412 |
| RequestTimeTooSowed | 403 Prohibido |
| ServiceUnavailable | 503 Servicio no disponible |
| SignatureDoesNotMatch | 403 Prohibido |
| Cucharones TooMany | 400 solicitud incorrecta |
| UserKeyMustBeSpecified | 400 solicitud incorrecta |

**códigos de error personalizados de StorageGRID**

| Nombre | Descripción | Estado de HTTP |
| --- | --- | --- |
| XBucketLifecycleNotAllowed | No se permite la configuración del ciclo de vida de los bloques en un bloque compatible heredado | 400 solicitud incorrecta |
| XBucketPolicyParseException | Error al analizar la política JSON de bloques recibidos. | 400 solicitud incorrecta |
| XCondit. Cumplimiento | Operación denegada debido a la configuración de cumplimiento anterior. | 403 Prohibido |
| XDSLA ReducedRedundancyForbidden | No se permite una redundancia reducida en el bloque compatible con la tecnología heredada | 400 solicitud incorrecta |
| XMaxBucketPolicyLengthExceeded | Su política supera la longitud máxima permitida de la política de bloques. | 400 solicitud incorrecta |
| XMissingInternalRequestHeader | Falta un encabezado de una solicitud interna. | 400 solicitud incorrecta |
| Cumplimiento de XNoSuchBucketCompliance | El bloque especificado no tiene la conformidad heredada activada. | 404 no encontrado |
| XNotAcceptable | La solicitud contiene uno o más encabezados de aceptación que no se han podido satisfacer. | 406 no aceptable |
| XNotImplemed | La solicitud que ha proporcionado implica una funcionalidad que no se ha implementado. | 501 no implementada |

# Operaciones de la API de REST de StorageGRID S3

Existen operaciones añadidas en la API DE REST de S3 específicas del sistema StorageGRID.

- OBTENGA la solicitud de consistencia de bloque

  La solicitud DE consistencia DE GET Bucket permite determinar el nivel de consistencia que se aplica a un bloque determinado.

- PONER solicitud de consistencia de bloque

  La solicitud PUT Bucket Consistency permite especificar el nivel de coherencia que se va a aplicar a las operaciones realizadas en un bloque.

- GET Bucket última solicitud de tiempo de acceso

  La solicitud DE tiempo DE acceso del último bloque DE GET Bucket permite determinar si las actualizaciones de la última hora de acceso están habilitadas o deshabilitadas para bloques individuales.

- PUT Bucket última solicitud de tiempo de acceso

  La solicitud DE la última hora de acceso al bloque DE PUT permite habilitar o deshabilitar las actualizaciones del último tiempo de acceso para bloques individuales. Al deshabilitar las actualizaciones de la última hora de acceso, se mejora el rendimiento, y es la configuración predeterminada para todos los bloques creados con la versión 10.3.0 o posterior.

- DELETE bucket metadata notification Configuration

  La solicitud de configuración DE notificación DE metadatos DELETE Bucket permite deshabilitar el servicio de integración de búsqueda para bloques individuales al eliminar el XML de configuración.

- OBTENGA la solicitud de configuración de notificación de metadatos del bloque

  La solicitud de configuración DE notificación DE metadatos GET Bucket permite recuperar el XML de configuración que se utiliza para configurar la integración de búsqueda de bloques individuales.

- PUT bucket metadata notification Configuration

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

- OBTENGA la solicitud de uso del almacenamiento

  La solicitud GET Storage Usage le indica la cantidad total de almacenamiento que está usando una cuenta y por cada bloque asociado con la cuenta.

- Solicitudes de bloques obsoletas para cumplimiento de normativas heredadas

  Es posible que deba utilizar la API DE REST de StorageGRID S3 para gestionar los bloques creados con la función de cumplimiento heredada.

## OBTENGA la solicitud de consistencia de bloque

La solicitud DE consistencia DE GET Bucket permite determinar el nivel de consistencia que se aplica a un bloque determinado.

Los controles de consistencia predeterminados se establecen para garantizar la lectura tras escritura de los objetos recién creados.

Tiene el permiso s3:GetBucketConsistency, o bien sea raíz de la cuenta, para completar esta operación.

**Ejemplo de solicitud**

```
GET /bucket?x-ntap-sg-consistency HTTP/1.1
Date: <em>date</em>
Authorization: <em>authorization string</em>
Host: <em>host</em>
```

**Respuesta**

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

| Control de consistencia | Descripción |
|---|---|
| todo | Todos los nodos reciben los datos inmediatamente o se produce un error en la solicitud. |
| fuerte en todo el mundo | Garantiza la coherencia de lectura tras escritura para todas las solicitudes del cliente en todos los sitios. |
| sitio seguro | Garantiza la coherencia de lectura tras escritura para todas las solicitudes del cliente dentro de un sitio. |
| lectura-después-nueva-escritura | (Predeterminado) proporciona coherencia de lectura tras escritura para los nuevos objetos y coherencia final para las actualizaciones de objetos. Ofrece garantías de alta disponibilidad y protección de datos. La mayoría concuerda estrechamente con las garantías de coherencia de Amazon S3.<br><br>**Nota:** Si su aplicación utiliza PETICIONES HEAD en objetos que no existen, puede recibir un número elevado de 500 errores de Internal Server si uno o más nodos de almacenamiento no están disponibles. Para evitar estos errores, establezca el control de coherencia en "'Available'" a menos que necesite garantías de coherencia similares a las de Amazon S3. |
| Disponible (coherencia eventual para operaciones DE CABEZAL) | Se comporta del mismo modo que el nivel de consistencia "read-after-new-write", pero sólo proporciona consistencia eventual para las operaciones DE CABEZA. Ofrece una mayor disponibilidad para las OPERACIONES DE CABEZAL que una «escritura tras otra» si los nodos de almacenamiento no están disponibles. Se diferencia de las garantías de coherencia de Amazon S3 solo para operaciones HEAD. |

**Ejemplo de respuesta**

```
HTTP/1.1 200 OK
Date: Fri, 18 Sep 2020 01:02:18 GMT
Connection: CLOSE
Server: StorageGRID/11.5.0
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/">read-after-
new-write</Consistency>
```

**Información relacionada**

Controles de consistencia

## PONER solicitud de consistencia de bloque

La solicitud PUT Bucket Consistency permite especificar el nivel de coherencia que se va a aplicar a las operaciones realizadas en un bloque.

Los controles de consistencia predeterminados se establecen para garantizar la lectura tras escritura de los objetos recién creados.

Tiene el permiso s3:PutBucketConsistency, o bien sea raíz de la cuenta, para completar esta operación.

### Solicitud

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

| Control de consistencia | Descripción |
|---|---|
| todo | Todos los nodos reciben los datos inmediatamente o se produce un error en la solicitud. |
| fuerte en todo el mundo | Garantiza la coherencia de lectura tras escritura para todas las solicitudes del cliente en todos los sitios. |
| sitio seguro | Garantiza la coherencia de lectura tras escritura para todas las solicitudes del cliente dentro de un sitio. |

| Control de consistencia | Descripción |
| --- | --- |
| lectura-después-nueva-escritura | (Predeterminado) proporciona coherencia de lectura tras escritura para los nuevos objetos y coherencia final para las actualizaciones de objetos. Ofrece garantías de alta disponibilidad y protección de datos. La mayoría concuerda estrechamente con las garantías de coherencia de Amazon S3.<br><br>**Nota:** Si su aplicación utiliza PETICIONES HEAD en objetos que no existen, puede recibir un número elevado de 500 errores de Internal Server si uno o más nodos de almacenamiento no están disponibles. Para evitar estos errores, establezca el control de coherencia en "'Available'" a menos que necesite garantías de coherencia similares a las de Amazon S3. |
| Disponible (coherencia eventual para operaciones DE CABEZAL) | Se comporta del mismo modo que el nivel de consistencia "read-after-new-write", pero sólo proporciona consistencia eventual para las operaciones DE CABEZA. Ofrece una mayor disponibilidad para las OPERACIONES DE CABEZAL que una «escritura tras otra» si los nodos de almacenamiento no están disponibles. Se diferencia de las garantías de coherencia de Amazon S3 solo para operaciones HEAD. |

**Nota:** en general, se debe utilizar el valor de control de la coherencia "read-after-new-write". Si las solicitudes no funcionan correctamente, cambie el comportamiento del cliente de la aplicación, si es posible. O bien, configure el cliente para especificar el control de consistencia de cada solicitud API. Establecer el control de consistencia a nivel de cucharón únicamente como último recurso.

#### Ejemplo de solicitud

```
PUT /bucket?x-ntap-sg-consistency=strong-global HTTP/1.1
Date: <em>date</em>
Authorization: <em>authorization string</em>
Host: <em>host</em>
```

#### Información relacionada

Controles de consistencia

## GET Bucket última solicitud de tiempo de acceso

La solicitud DE tiempo DE acceso del último bloque DE GET Bucket permite determinar si las actualizaciones de la última hora de acceso están habilitadas o deshabilitadas para bloques individuales.

Tiene el permiso s3:GetBucketLastAccessTime, o sea la raíz de la cuenta, para completar esta operación.

**Ejemplo de solicitud**

```
GET /bucket?x-ntap-sg-lastaccesstime HTTP/1.1
Date: <em>date</em>
Authorization: <em>authorization string</em>
Host: <em>host</em>
```

**Ejemplo de respuesta**

Este ejemplo muestra que las actualizaciones de la última hora de acceso están habilitadas para el bloque.

```
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>
```

## PUT Bucket última solicitud de tiempo de acceso

La solicitud DE la última hora de acceso al bloque DE PUT permite habilitar o deshabilitar las actualizaciones del último tiempo de acceso para bloques individuales. Al deshabilitar las actualizaciones de la última hora de acceso, se mejora el rendimiento, y es la configuración predeterminada para todos los bloques creados con la versión 10.3.0 o posterior.

Tiene el permiso s3:PutBucketLastAccessTime para que un bloque, o sea la raíz de la cuenta, complete esta operación.

> A partir de la versión 10.3 de StorageGRID, las actualizaciones de la última hora de acceso se deshabilitan de forma predeterminada para todos los bloques nuevos. Si tiene bloques que se crearon con una versión anterior de StorageGRID y desea coincidir con el nuevo comportamiento predeterminado, debe deshabilitar explícitamente las actualizaciones de la última hora de acceso para cada uno de esos bloques anteriores. Puede habilitar o deshabilitar las actualizaciones para la última hora de acceso mediante LA solicitud DE LA última hora de ACCESO DE PUT Bucket, la casilla de verificación **S3** > **Cuchos** > **Cambiar la última configuración de acceso** en el Administrador de inquilinos o la API de administración de inquilinos.

Si se desactivan las actualizaciones de la última hora de acceso para un bloque, se aplicará el siguiente comportamiento a las operaciones del bloque:

- LAS solicitudes GET Object, GET Object ACL, GET Object Etiquetado y HEAD Object no actualizan la última hora de acceso. El objeto no se agrega a las colas para la evaluación de la gestión del ciclo de vida de la información (ILM).
- PUT Object (PONER objeto): Copie y COLOQUE las solicitudes de etiquetado de objetos que sólo actualizan los metadatos. También actualice la hora del último acceso. El objeto se agrega a las colas para la evaluación de ILM.
- Si las actualizaciones a la hora del último acceso están deshabilitadas para el bloque de origen, PUT Object - Copy Requests no actualizan la hora del último acceso para el bloque de origen. El objeto que se copió no se agrega a colas para la evaluación de ILM para el bloque de origen. Sin embargo, PARA el destino, PONER objeto - Copiar solicitudes siempre actualizar la última hora de acceso. La copia del objeto se agrega a las colas para la evaluación de ILM.
- Completar solicitudes de carga de varias partes actualizar la última hora de acceso. El objeto completado se agrega a las colas para la evaluación de ILM.

#### Solicitar ejemplos

En este ejemplo se habilita la hora de último acceso para un bloque.

```
PUT /bucket?x-ntap-sg-lastaccesstime=enabled HTTP/1.1
Date: <em>date</em>
Authorization: <em>authorization string</em>
Host: <em>host</em>
```

En este ejemplo se deshabilita la hora de último acceso para un bloque.

```
PUT /bucket?x-ntap-sg-lastaccesstime=disabled HTTP/1.1
Date: <em>date</em>
Authorization: <em>authorization string</em>
Host: <em>host</em>
```

#### Información relacionada

Usar cuenta de inquilino

## DELETE bucket metadata notification Configuration

La solicitud de configuración DE notificación DE metadatos DELETE Bucket permite deshabilitar el servicio de integración de búsqueda para bloques individuales al eliminar el XML de configuración.

Tiene el permiso s3:DeleteBucketMetadataNotification para un bucket, o sea root de la cuenta, para completar esta operación.

#### Ejemplo de solicitud

Este ejemplo muestra cómo deshabilitar el servicio de integración de búsqueda para un bloque.

```
DELETE /test1?x-ntap-sg-metadata-notification HTTP/1.1
Date: <em>date</em>
Authorization: <em>authorization string</em>
Host: <em>host</em>
```

## OBTENGA la solicitud de configuración de notificación de metadatos del bloque

La solicitud de configuración DE notificación DE metadatos GET Bucket permite recuperar el XML de configuración que se utiliza para configurar la integración de búsqueda de bloques individuales.

Tiene el permiso s3:GetBucketMetadataNotification o ser raíz de la cuenta, para completar esta operación.

### Ejemplo de solicitud

Esta solicitud recupera la configuración de notificación de metadatos del bloque denominado `bucket`.

```
GET /bucket?x-ntap-sg-metadata-notification HTTP/1.1
Date: <em>date</em>
Authorization: <em>authorization string</em>
Host: <em>host</em>
```

### Respuesta

El cuerpo de la respuesta incluye la configuración de notificación de metadatos para el bloque. La configuración de notificaciones de metadatos permite determinar cómo se configura el bloque para la integración de búsquedas. Es decir, permite determinar a qué objetos se indexan y a qué extremos se envían los metadatos de sus 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 varias reglas. Cada regla especifica los objetos a los que se aplica y el destino al que StorageGRID debe enviar metadatos de objetos. Los destinos se deben especificar con el URN de un extremo de StorageGRID.

| Nombre | Descripción | Obligatorio |
|---|---|---|
| MetadataNotificationConfiguration | Etiqueta de contenedor para las reglas que se usan para especificar los objetos y el destino de las notificaciones de metadatos.<br><br>Contiene uno o más elementos Regla. | Sí |
| Regla | Código de contenedor de una regla que identifica los objetos cuyos metadatos deben agregarse a un índice especificado.<br><br>Se rechazan las reglas con prefijos superpuestos.<br><br>Incluido en el elemento MetadataNotificationConfiguration. | Sí |
| ID | Identificador único de la regla.<br><br>Incluido en el elemento Regla. | No |

| Nombre | Descripción | Obligatorio |
| --- | --- | --- |
| Estado | El estado puede ser "activado" o "desactivado". No se toman medidas para las reglas que están desactivadas.<br><br>Incluido en el elemento Regla. | Sí |
| Prefijo | Los objetos que coinciden con el prefijo se ven afectados por la regla y sus metadatos se envían al destino especificado.<br><br>Para hacer coincidir todos los objetos, especifique un prefijo vacío.<br><br>Incluido en el elemento Regla. | Sí |
| Destino | Etiqueta de contenedor para el destino de una regla.<br><br>Incluido en el elemento Regla. | Sí |

| Nombre | Descripción | Obligatorio |
| --- | --- | --- |
| No | URN del destino donde se envían los metadatos del objeto. Debe ser URN de un extremo de StorageGRID con las siguientes propiedades:<br>• `es` debe ser el tercer elemento.<br>• El URN debe terminar con el índice y el tipo donde se almacenan los metadatos, en el formulario `domain-name/myindex/mytype`.<br>Los extremos se configuran con el administrador de inquilinos o la API de gestión de inquilinos. Tienen el siguiente formato:<br>• `arn:aws:es:_region:account-ID_:domain/mydomain/myindex/mytype`<br>• `urn:mysite:es:::mydomain/myindex/mytype`<br>El punto final debe estar configurado antes de enviar el XML de configuración o la configuración fallará con un error 404.<br>El valor de urn se incluye en el elemento Destination. | Sí |

**Ejemplo de respuesta**

El XML incluido entre `<MetadataNotificationConfiguration></MetadataNotificationConfiguration>` tags muestra cómo se configura la integración con un extremo de integración de búsqueda para el bloque. En este ejemplo, los metadatos del objeto se envían a un índice de Elasticsearch llamado `current` y escriba named `2017` Que se aloja en un dominio de AWS llamado `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**

Usar cuenta de inquilino

## PUT bucket metadata notification Configuration

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

Tiene el permiso s3:PutBucketMetadataNotification para un bloque o ser raíz de la cuenta, para completar esta operación.

### Solicitud

La solicitud debe incluir la configuración de notificación de metadatos en el cuerpo de la solicitud. Cada configuración de notificación de metadatos incluye una o varias reglas. Cada regla especifica los objetos a los que se aplica y el destino al que StorageGRID debe enviar metadatos de objetos.

Los objetos se pueden filtrar según el prefijo del nombre del objeto. Por ejemplo, puede enviar metadatos de los objetos con el prefijo `/images` en un destino y objetos con el prefijo `/videos` a otro.

Las configuraciones que tienen prefijos superpuestos no son válidas y se rechazan cuando se envían. Por ejemplo, una configuración que incluía una regla para objetos con el prefijo `test` y una segunda regla para los objetos con el prefijo `test2` no se permitirá.

Los destinos se deben especificar con el URN de un extremo de StorageGRID. El extremo debe existir cuando

se envía la configuración de notificación de metadatos o la solicitud falla como un `400 Bad Request`. El mensaje de error indica: `Unable to save the metadata notification (search) policy. The specified endpoint URN does not exist: URN`.

```
<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>
```

En la tabla se describen los elementos del XML de configuración de notificaciones de metadatos.

| Nombre | Descripción | Obligatorio |
|---|---|---|
| MetadataNotificationConfiguration | Etiqueta de contenedor para las reglas que se usan para especificar los objetos y el destino de las notificaciones de metadatos.<br><br>Contiene uno o más elementos Regla. | Sí |
| Regla | Código de contenedor de una regla que identifica los objetos cuyos metadatos deben agregarse a un índice especificado.<br><br>Se rechazan las reglas con prefijos superpuestos.<br><br>Incluido en el elemento MetadataNotificationConfiguration. | Sí |
| ID | Identificador único de la regla.<br><br>Incluido en el elemento Regla. | No |

| Nombre | Descripción | Obligatorio |
| --- | --- | --- |
| Estado | El estado puede ser "activado" o "desactivado". No se toman medidas para las reglas que están desactivadas.<br><br>Incluido en el elemento Regla. | Sí |
| Prefijo | Los objetos que coinciden con el prefijo se ven afectados por la regla y sus metadatos se envían al destino especificado.<br><br>Para hacer coincidir todos los objetos, especifique un prefijo vacío.<br><br>Incluido en el elemento Regla. | Sí |
| Destino | Etiqueta de contenedor para el destino de una regla.<br><br>Incluido en el elemento Regla. | Sí |

| Nombre | Descripción | Obligatorio |
| --- | --- | --- |
| No | URN del destino donde se envían los metadatos del objeto. Debe ser URN de un extremo de StorageGRID con las siguientes propiedades:<br>• `es` debe ser el tercer elemento.<br>• El URN debe terminar con el índice y el tipo donde se almacenan los metadatos, en el formulario `domain-name/myindex/mytype`.<br>Los extremos se configuran con el administrador de inquilinos o la API de gestión de inquilinos. Tienen el siguiente formato:<br>• `arn:aws:es:region:account-ID:domain/mydomain/myindex/mytype`<br>• `urn:mysite:es:::mydomain/myindex/mytype`<br>El punto final debe estar configurado antes de enviar el XML de configuración o la configuración fallará con un error 404.<br>El valor de urn se incluye en el elemento Destination. | Sí |

**Solicitar ejemplos**

Este ejemplo muestra habilitar la integración de búsqueda de un bloque. En este ejemplo, los metadatos de objeto de todos los objetos se envían al mismo destino.

```
PUT /test1?x-ntap-sg-metadata-notification HTTP/1.1
Date: <em>date</em>
Authorization: <em>authorization string</em>
Host: <em>host</em>

<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, metadatos de objeto para objetos que coinciden con el prefijo `/images` se envía a un destino, mientras que los metadatos de objetos de los objetos que coinciden con el prefijo `/videos` se envía a un segundo destino.

```
PUT /graphics?x-ntap-sg-metadata-notification HTTP/1.1
Date: <em>date</em>
Authorization: <em>authorization string</em>
Host: <em>host</em>

<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>
```

**JSON generado por el servicio de integración de búsqueda**

Al habilitar el servicio de integración de búsqueda para un bloque, se genera un documento JSON y se envía al extremo de destino cada vez que se agregan, actualizan o eliminan metadatos o etiquetas del objeto.

Este ejemplo muestra un ejemplo de JSON que se podría generar cuando un objeto con la clave `SGWS/Tagging.txt` se crea en un bloque llamado `test`. La `test` el bloque no tiene versiones, por lo que el `versionId` la etiqueta está vacía.

```
{
  "bucket": "test",
  "key": "SGWS/Tagging.txt",
  "versionId": "",
  "accountId": "86928401983529626822",
  "size": 38,
  "md5": "3d6c7634a85436eee06d43415012855",
  "region":"us-east-1"
  "metadata": {
    "age": "25"
  },
  "tags": {
    "color": "yellow"
  }
}
```

**Metadatos de objetos incluidos en las notificaciones de metadatos**

En la tabla se enumeran todos los campos que se incluyen en el documento JSON que se envían al extremo de destino cuando la integración de búsqueda está habilitada.

El nombre del documento incluye el nombre del bloque, el nombre del objeto y el ID de versión, si existe.

| Tipo | Nombre del elemento | Descripción |
|---|---|---|
| Información sobre bloques y objetos | cucharón | Nombre del bloque |
| Información sobre bloques y objetos | clave | Nombre de clave de objeto |
| Información sobre bloques y objetos | ID de versión | Versión de objeto, para objetos en bloques con versiones |
| Información sobre bloques y objetos | región | Región de bloque, por ejemplo `us-east-1` |
| Metadatos del sistema | tamaño | Tamaño del objeto (en bytes) visible para un cliente HTTP |
| Metadatos del sistema | md5 | Hash de objeto |
| Metadatos del usuario | metadatos *`key:value`* | Todos los metadatos de usuario del objeto, como pares clave-valor |

| Tipo | Nombre del elemento | Descripción |
| --- | --- | --- |
| Etiquetas | etiquetas *key:value* | Todas las etiquetas de objeto definidas para el objeto, como pares clave-valor |

**Nota:** para etiquetas y metadatos de usuario, StorageGRID pasa fechas y números a Elasticsearch como cadenas o como notificaciones de eventos S3. Para configurar Elasticsearch para interpretar estas cadenas como fechas o números, siga las instrucciones de Elasticsearch para la asignación dinámica de campos y para asignar formatos de fecha. Debe habilitar las asignaciones de campos dinámicos en el índice antes de configurar el servicio de integración de búsqueda. Una vez indizado un documento, no se pueden editar los tipos de campo del documento en el índice.

**Información relacionada**

Usar cuenta de inquilino

## OBTENGA la solicitud de uso del almacenamiento

La solicitud GET Storage Usage le indica la cantidad total de almacenamiento que está usando una cuenta y por cada bloque asociado con la cuenta.

La cantidad de almacenamiento utilizada por una cuenta y sus depósitos puede obtenerse mediante una solicitud DE SERVICIO GET modificada con el `x-ntap-sg-usage` parámetro de consulta. Se realiza un seguimiento del uso del almacenamiento en bloques de forma independiente de las solicitudes DE PUT y DELETE procesadas por el sistema. Es posible que haya algún retraso antes de que los valores de uso coincidan con los valores esperados en función del procesamiento de las solicitudes, especialmente si el sistema está sometido a cargas pesadas.

De forma predeterminada, StorageGRID intenta recuperar la información de uso con una coherencia global fuerte. Si no se puede lograr una coherencia global sólida, StorageGRID intenta recuperar la información de uso con una coherencia de sitio sólida.

Tiene el permiso s3:ListAllMyBuckets o ser la raíz de la cuenta para completar esta operación.

### Ejemplo de solicitud

```
GET /?x-ntap-sg-usage HTTP/1.1
Date: <em>date</em>
Authorization: <em>authorization string</em>
Host: <em>host</em>
```

### Ejemplo de respuesta

Este ejemplo muestra una cuenta que tiene cuatro objetos y 12 bytes de datos en dos bloques. Cada bloque contiene dos objetos y seis bytes de datos.

```
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>
```

#### Creación de versiones

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

#### Información relacionada

Controles de consistencia

## Solicitudes de bloque obsoletas para cumplimiento de normativas heredadas

Es posible que deba utilizar la API DE REST de StorageGRID S3 para gestionar los bloques creados con la función de cumplimiento heredada.

#### Función de cumplimiento de normativas obsoleta

La función de cumplimiento de StorageGRID que estaba disponible en versiones anteriores de StorageGRID quedó obsoleta y se reemplazó por el bloqueo de objetos de S3.

Si anteriormente habilitó la configuración de cumplimiento global, la opción de bloqueo de objetos S3 global se habilita en StorageGRID 11.6. Ya no se pueden crear nuevos bloques con la función de cumplimiento

habilitada; sin embargo, según sea necesario, se puede utilizar la API DE REST de StorageGRID S3 para gestionar bloques existentes que cumplen las normativas.

- Utilice el bloqueo de objetos de S3
- Gestión de objetos con ILM
- "Base de conocimientos de NetApp: Cómo gestionar bloques heredados que cumplen con la normativa StorageGRID 11.5"

Solicitudes de cumplimiento de normativas obsoletas:

- Obsoleto: PONGA modificaciones de solicitud de cucharón para el cumplimiento

  El elemento XML de SGCompliance está obsoleto. Anteriormente, podría incluir este elemento personalizado de StorageGRID en el cuerpo de solicitud XML opcional de SOLICITUDES PUT Bucket para crear un bloque compatible.

- Obsoleto: GET Bucket Compliance Request

  La solicitud DE cumplimiento GET Bucket queda obsoleta. Sin embargo, puede seguir utilizando esta solicitud para determinar la configuración de cumplimiento actual para un bloque compatible heredado existente.

- Obsoleto: PUT Bucket Compliance Request

  La solicitud DE cumplimiento PUT Bucket queda obsoleta. Sin embargo, puede seguir utilizando esta solicitud para modificar la configuración de cumplimiento de un bloque compatible heredado existente. Por ejemplo, puede colocar un bloque existente en la retención legal o aumentar su período de retención.

#### Obsoleto: PONGA modificaciones de solicitud de cucharón para el cumplimiento

El elemento XML de SGCompliance está obsoleto. Anteriormente, podría incluir este elemento personalizado de StorageGRID en el cuerpo de solicitud XML opcional de SOLICITUDES PUT Bucket para crear un bloque compatible.

> La función de cumplimiento de StorageGRID que estaba disponible en versiones anteriores de StorageGRID quedó obsoleta y se reemplazó por el bloqueo de objetos de S3.

Utilice el bloqueo de objetos de S3

Gestión de objetos con ILM

"Base de conocimientos de NetApp: Cómo gestionar bloques heredados que cumplen con la normativa StorageGRID 11.5"

Ya no se pueden crear bloques nuevos con el cumplimiento de normativas habilitado. Se devuelve el siguiente mensaje de error si intenta utilizar las modificaciones DE la solicitud PUT Bucket para cumplir con las normativas a fin de crear un nuevo bloque compatible:

```
The Compliance feature is deprecated.
Contact your StorageGRID administrator if you need to create new Compliant
buckets.
```

**Información relacionada**

Gestión de objetos con ILM

Usar cuenta de inquilino

**Obsoleto: GET Bucket Compliance Request**

La solicitud DE cumplimiento GET Bucket queda obsoleta. Sin embargo, puede seguir utilizando esta solicitud para determinar la configuración de cumplimiento actual para un bloque compatible heredado existente.

> La función de cumplimiento de StorageGRID que estaba disponible en versiones anteriores de StorageGRID quedó obsoleta y se reemplazó por el bloqueo de objetos de S3.

Utilice el bloqueo de objetos de S3

Gestión de objetos con ILM

"Base de conocimientos de NetApp: Cómo gestionar bloques heredados que cumplen con la normativa StorageGRID 11.5"

Tiene el permiso s3:GetBucketCompliance o ser la raíz de la cuenta para completar esta operación.

**Ejemplo de solicitud**

Esta solicitud de ejemplo le permite determinar la configuración de cumplimiento para el bloque denominado `mybucket`.

```
GET /mybucket/?x-ntap-sg-compliance HTTP/1.1
Date: <em>date</em>
Authorization: <em>authorization string</em>
Host: <em>host</em>
```

**Ejemplo de respuesta**

En la respuesta XML, `<SGCompliance>` enumera la configuración de cumplimiento vigente para el bloque. Esta respuesta de ejemplo muestra la configuración de cumplimiento de un bloque en el que se conservará cada objeto durante un año (525,600 minutos), a partir del momento en que el objeto se ingiere en la cuadrícula. Actualmente, no existe ningún derecho legal en este segmento. Cada objeto se eliminará automáticamente después de un año.

```
HTTP/1.1 200 OK
Date: <em>date</em>
Connection: <em>connection</em>
Server: StorageGRID/11.1.0
x-amz-request-id: <em>request ID</em>
Content-Length: <em>length</em>
Content-Type: application/xml

<SGCompliance>
  <RetentionPeriodMinutes>525600</RetentionPeriodMinutes>
  <LegalHold>false</LegalHold>
  <AutoDelete>true</AutoDelete>
</SGCompliance>
```

| Nombre | Descripción |
|---|---|
| RetentionPeriodonMinutes | La duración del período de retención para los objetos que se añadió a este bloque, en minutos. El período de retención se inicia cuando el objeto se ingiere en la cuadrícula. |
| LegalHold | • Cierto: Este segmento está actualmente bajo un control legal. Los objetos de este segmento no se pueden eliminar hasta que se levante la retención legal, incluso si ha caducado su período de retención.<br>• Falso: Este segmento no está actualmente bajo un derecho. Los objetos de este bloque se pueden eliminar cuando expire su período de retención. |
| Eliminación automática | • True: Los objetos de este bloque se eliminarán automáticamente cuando expire su período de retención, a menos que el bloque se encuentre bajo una retención legal.<br>• False: Los objetos de este bloque no se eliminarán automáticamente cuando finalice el período de retención. Debe eliminar estos objetos manualmente si necesita eliminarlos. |

#### Respuestas de error

Si el bloque no se creó para ser compatible, el código de estado HTTP para la respuesta es `404 Not Found`, Con un código de error S3 de `XNoSuchBucketCompliance`.

#### Información relacionada

Gestión de objetos con ILM

Usar cuenta de inquilino

### Obsoleto: PUT Bucket Compliance Request

La solicitud DE cumplimiento PUT Bucket queda obsoleta. Sin embargo, puede seguir utilizando esta solicitud para modificar la configuración de cumplimiento de un bloque compatible heredado existente. Por ejemplo, puede colocar un bloque existente en la retención legal o aumentar su período de retención.

> La función de cumplimiento de StorageGRID que estaba disponible en versiones anteriores de StorageGRID quedó obsoleta y se reemplazó por el bloqueo de objetos de S3.

Utilice el bloqueo de objetos de S3

Gestión de objetos con ILM

"Base de conocimientos de NetApp: Cómo gestionar bloques heredados que cumplen con la normativa StorageGRID 11.5"

Tiene el permiso s3:PutBucketCompliance o ser la raíz de la cuenta para completar esta operación.

Debe especificar un valor para cada campo de la configuración de cumplimiento al emitir una solicitud DE cumplimiento PUT Bucket.

#### Ejemplo de solicitud

Esta solicitud de ejemplo modifica la configuración de cumplimiento del bloque denominado `mybucket`. En este ejemplo, los objetos de `mybucket` ahora se conservará durante dos años (1,051,200 minutos) en lugar de un año, a partir del momento en que el objeto se ingiere en la cuadrícula. No existe ningún derecho legal en este segmento. Cada objeto se eliminará automáticamente después de dos años.

```
PUT /mybucket/?x-ntap-sg-compliance HTTP/1.1
Date: <em>date</em>
Authorization: <em>authorization name</em>
Host: <em>host</em>
Content-Length: 152

<SGCompliance>
  <RetentionPeriodMinutes>1051200</RetentionPeriodMinutes>
  <LegalHold>false</LegalHold>
  <AutoDelete>true</AutoDelete>
</SGCompliance>
```

| Nombre | Descripción |
| --- | --- |
| RetentionPeriodonMinutes | La duración del período de retención para los objetos que se añadió a este bloque, en minutos. El período de retención se inicia cuando el objeto se ingiere en la cuadrícula.<br><br>**Atención:** al especificar un nuevo valor para RetentionPeriodonMinutes, debe especificar un valor igual o mayor que el período de retención actual del cucharón. Una vez establecido el período de retención del segmento, no podrá disminuir dicho valor; sólo podrá aumentarlo. |
| LegalHold | • Cierto: Este segmento está actualmente bajo un control legal. Los objetos de este segmento no se pueden eliminar hasta que se levante la retención legal, incluso si ha caducado su período de retención.<br>• Falso: Este segmento no está actualmente bajo un derecho. Los objetos de este bloque se pueden eliminar cuando expire su período de retención. |
| Eliminación automática | • True: Los objetos de este bloque se eliminarán automáticamente cuando expire su período de retención, a menos que el bloque se encuentre bajo una retención legal.<br>• False: Los objetos de este bloque no se eliminarán automáticamente cuando finalice el período de retención. Debe eliminar estos objetos manualmente si necesita eliminarlos. |

#### Nivel de coherencia para la configuración de cumplimiento de normativas

Cuando se actualiza la configuración de cumplimiento de normativas para un bloque de S3 con una solicitud DE cumplimiento PUT Bucket, StorageGRID intenta actualizar los metadatos del bloque en el grid. De forma predeterminada, StorageGRID utiliza el nivel de consistencia **strong-global** para garantizar que todos los sitios de centros de datos y todos los nodos de almacenamiento que contienen metadatos de bloques tengan coherencia de lectura tras escritura para la configuración de cumplimiento modificada.

Si StorageGRID no puede alcanzar el nivel de consistencia **strong-global** debido a que un sitio de centro de datos o varios nodos de almacenamiento de un sitio no están disponibles, el código de estado HTTP de la respuesta es `503 Service Unavailable.`

Si recibe esta respuesta, debe ponerse en contacto con el administrador de grid para garantizar que los servicios de almacenamiento requeridos estén disponibles en Lo antes posible.. Si el administrador de grid no puede hacer que haya suficientes nodos de almacenamiento en cada sitio disponibles, el soporte técnico puede pedirle que vuelva a intentar la solicitud fallida forzando el nivel de consistencia de **sitio seguro**.

Nunca fuerce el nivel de consistencia de **sitio fuerte** para EL cumplimiento DE LA cuchara DE PUT a menos que usted haya sido dirigido a hacerlo por el soporte técnico y a menos que usted entienda las consecuencias potenciales de usar este nivel.

Cuando el nivel de consistencia se reduce a **sitio seguro**, StorageGRID garantiza que la configuración de cumplimiento actualizada tendrá coherencia de lectura tras escritura sólo para las solicitudes de cliente dentro de un sitio. Esto significa que el sistema StorageGRID podría tener temporalmente varias configuraciones incoherentes para este bloque hasta que todos los sitios y nodos de almacenamiento estén disponibles. La configuración incoherente puede dar como resultado un comportamiento inesperado y no deseado. Por ejemplo, si coloca un bloque bajo una retención legal y fuerza un nivel de coherencia más bajo, la configuración de cumplimiento anterior del bloque (es decir, la retención legal) puede seguir vigente en algunos centros de datos. Como resultado, los objetos que cree que están en retención legal se pueden eliminar cuando caduque su período de retención, ya sea por el usuario o por AutoDelete, si está activado.

Para forzar el uso del nivel de consistencia de **sitio fuerte**, vuelva a emitir la solicitud DE cumplimiento DE PUT Bucket e incluya el `Consistency-Control` Encabezado de solicitud HTTP, de la siguiente manera:

```
PUT /mybucket/?x-ntap-sg-compliance HTTP/1.1
Consistency-Control: strong-site
```

**Respuestas de error**

- Si el bloque no se creó para ser compatible, el código de estado HTTP para la respuesta es `404 Not Found`.
- Si `RetentionPeriodMinutes` En la solicitud es inferior al período de retención actual del bloque, el código de estado HTTP es `400 Bad Request`.

**Información relacionada**

Obsoleto: PONGA modificaciones de solicitud de cucharón para el cumplimiento

Usar cuenta de inquilino

Gestión de objetos con ILM

# Políticas de acceso a bloques y grupos

StorageGRID utiliza el lenguaje de políticas de Amazon Web Services (AWS) para permitir que los inquilinos S3 controlen el acceso a bloques y objetos dentro de esos bloques. El sistema StorageGRID implementa un subconjunto del lenguaje de políticas de la API DE REST de S3. Las políticas de acceso para la API de S3 se escriben en JSON.

## Información general sobre las políticas de acceso

Existen dos tipos de políticas de acceso compatibles con StorageGRID.

- **Políticas de bloque**, que se configuran mediante las operaciones API Get Bucket, PUT Bucket y DELETE Bucket Policy S3. Las políticas de bloque se asocian a bloques, por lo que se configuran para controlar el acceso de los usuarios de la cuenta de propietario del bloque u otras cuentas al bloque y a los objetos en

él. La política de bloques se aplica únicamente a un bloque y, posiblemente, a varios grupos.

- **Políticas de grupo**, que se configuran mediante el Administrador de inquilinos o la API de administración de inquilinos. Las directivas de grupo se asocian a un grupo de la cuenta, por lo que se configuran para permitir que dicho grupo tenga acceso a recursos específicos propiedad de dicha cuenta. La política de grupo se aplica únicamente a un grupo y, posiblemente, a varios bloques.

Las políticas de bloque y grupo de StorageGRID siguen una gramática específica definida por Amazon. Dentro de cada política hay una serie de declaraciones de política y cada sentencia contiene los siguientes elementos:

- ID de sentencia (Sid) (opcional)
- Efecto
- Principal/NotPrincipal
- Recurso/NotResource
- Acción/NotAction
- Condición (opcional)

Las sentencias de directiva se crean utilizando esta estructura para especificar permisos: Conceda <Effect> para permitir/denegar que <Principal> ejecute <Action> en <Resource> cuando se aplique <Condition>.

Cada elemento de directiva se utiliza para una función específica:

| Elemento | Descripción |
|---|---|
| SID | El elemento Sid es opcional. El Sid sólo se ha diseñado como una descripción para el usuario. El sistema StorageGRID lo almacena pero no lo interpreta. |
| Efecto | Utilice el elemento Effect para establecer si se permiten o deniegan las operaciones especificadas. Debe identificar las operaciones que permite (o deniega) en cubos u objetos utilizando las palabras clave del elemento Acción admitido. |
| Principal/NotPrincipal | Puede permitir a los usuarios, grupos y cuentas acceder a recursos específicos y realizar acciones específicas. Si no se incluye ninguna firma S3 en la solicitud, se permite el acceso anónimo especificando el carácter comodín (*) como principal. De forma predeterminada, sólo la raíz de la cuenta tiene acceso a los recursos que pertenecen a la cuenta.<br><br>Sólo es necesario especificar el elemento Principal en una política de bloque. Para las directivas de grupo, el grupo al que se asocia la directiva es el elemento Principal implícito. |
| Recurso/NotResource | El elemento Resource identifica los bloques y los objetos. Puede permitir o denegar permisos para cubos y objetos utilizando el nombre de recurso de Amazon (ARN) para identificar el recurso. |

| Elemento | Descripción |
| --- | --- |
| Acción/NotAction | Los elementos Acción y efecto son los dos componentes de los permisos. Cuando un grupo solicita un recurso, se le concede o se le deniega el acceso al recurso. Se deniega el acceso a menos que asigne permisos de forma específica, pero puede utilizar Denegar explícito para anular un permiso otorgado por otra directiva. |
| Condición | El elemento Condition es opcional. Las condiciones permiten crear expresiones para determinar cuándo se debe aplicar una directiva. |

En el elemento Action , puede utilizar el carácter comodín (*) para especificar todas las operaciones o un subconjunto de operaciones. Por ejemplo, esta acción coincide con permisos como s3:GetObject, s3:PutObject y s3:DeleteObject.

```
s3:*Object
```

En el elemento Resource , puede utilizar los caracteres comodín (*) y (?). Aunque el asterisco (*) coincide con 0 o más caracteres, el signo de interrogación (?) coincide con cualquier carácter.

En el elemento Principal, los caracteres comodín no se admiten excepto para establecer el acceso anónimo, que concede permiso a todos. Por ejemplo, el comodín (*) se establece como el valor Principal.

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

En el ejemplo siguiente, la instrucción utiliza los elementos Effect, Principal, Acción y recurso. En este ejemplo se muestra una sentencia de directiva de bloque completa que utiliza el efecto "permitir" para dar a los principales, el grupo admin `federated-group/admin` y el grupo financiero `federated-group/finance`, Permisos para realizar la acción `s3:ListBucket` en el bloque llamado `mybucket` Y la Acción `s3:GetObject` en todos los objetos dentro de ese cucharón.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::27233906934684427525:federated-group/admin",
          "arn:aws:iam::27233906934684427525:federated-group/finance"
        ]
      },
      "Action": [
        "s3:ListBucket",
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:iam:s3:::mybucket",
        "arn:aws:iam:s3:::mybucket/*"
      ]
    }
  ]
}
```

La política de bloque tiene un límite de tamaño de 20,480 bytes y la política de grupo tiene un límite de tamaño de 5,120 bytes.

**Información relacionada**

Usar cuenta de inquilino

## Configuración de control de coherencia para políticas

De forma predeterminada, cualquier actualización que realice a las directivas de grupo será consistente. Una vez que la política de grupo sea coherente, los cambios pueden tardar 15 minutos más en aplicarse, debido al almacenamiento en caché de políticas. De forma predeterminada, las actualizaciones que realice en las políticas de bloques también serán coherentes.

Según sea necesario, puede cambiar las garantías de coherencia para las actualizaciones de la política de bloques. Por ejemplo, es posible que desee que un cambio en una política de bloque se convierta en una Lo antes posible. efectiva por motivos de seguridad.

En este caso, puede ajustar la `Consistency-Control` En la solicitud DE política PUT Bucket, o puede utilizar la solicitud DE consistencia PUT Bucket. Al cambiar el control de coherencia para esta solicitud, debe utilizar el valor **all**, que ofrece la mayor garantía de coherencia de lectura tras escritura. Si especifica cualquier otro valor de control de consistencia en un encabezado para LA solicitud DE consistencia PUT Bucket, la solicitud será rechazada. Si especifica cualquier otro valor para una solicitud DE política PUT Bucket, el valor se ignorará. Una vez que una política de bloques se vuelve coherente, los cambios pueden tardar 8 segundos más en aplicarse, debido al almacenamiento en caché de la política.

Si establece el nivel de consistencia en **all** para forzar la aplicación de una nueva política de cucharón antes, asegúrese de volver a establecer el control de nivel de cucharón en su valor original cuando haya terminado. De lo contrario, todas las solicitudes de segmentos futuras utilizarán la configuración **all**.

## Utilice ARN en las declaraciones de política

En las declaraciones de política, el ARN se utiliza en los elementos Principal y Recursos.

- Utilice esta sintaxis para especificar el recurso ARN de S3:

```
arn:aws:s3:::bucket-name
arn:aws:s3:::bucket-name/object_key
```

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

```
arn:aws:iam::account_id:root
arn:aws:iam::account_id:user/user_name
arn:aws:iam::account_id:group/group_name
arn:aws:iam::account_id:federated-user/user_name
arn:aws:iam::account_id:federated-group/group_name
```

Otras consideraciones:

- Puede utilizar el asterisco (*) como comodín para que coincida con cero o más caracteres dentro de la clave de objeto.
- Los caracteres internacionales, que se pueden especificar en la clave de objeto, deben codificarse mediante JSON UTF-8 o mediante secuencias de escape JSON \u. No se admite el porcentaje de codificación.

  "Sintaxis de URN RFC 2141"

  El cuerpo de solicitud HTTP para la operación DE política PUT Bucket debe codificarse con charset=UTF-8.

## Especifique recursos en una política

En las sentencias de directiva, puede utilizar el elemento Resource para especificar el bloque o el objeto para el que se permiten o deniegan los permisos.

- Cada instrucción de directiva requiere un elemento Resource. En una política, el elemento denota los recursos `Resource`o bien, `NotResource` para la exclusión.
- Se especifican recursos con un ARN de recurso S3. Por ejemplo:

```
"Resource": "arn:aws:s3:::mybucket/*"
```

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

```
"Resource": "arn:aws:s3:::mybucket/home/${aws:username}/*"
```

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

**Información relacionada**

Especifique las variables en una política

## Especifique los principales en una directiva

Utilice el elemento Principal para identificar al usuario, grupo o cuenta de arrendatario que la sentencia de directiva permite o deniega el acceso al recurso.

- Cada sentencia de política de una política de bloque debe incluir un elemento Principal. Las declaraciones de política en una política de grupo no necesitan el elemento Principal porque se entiende que el grupo es el principal.
- En una política, los directores son denotados por el elemento «'Principal,'» o «'NotPrincipal» para la exclusión.
- Las identidades basadas en cuentas se deben especificar mediante un ID o un ARN:

```
"Principal": { "AWS": "account_id"}
"Principal": { "AWS": "identity_arn" }
```

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

```
 "Principal": { "AWS": "27233906934684427525" }
```

- Puede especificar sólo la raíz de la cuenta:

```
"Principal": { "AWS": "arn:aws:iam::27233906934684427525:root" }
```

- Puede especificar un usuario federado específico ("Alex"):

```
"Principal": { "AWS": "arn:aws:iam::27233906934684427525:federated-
user/Alex" }
```

- Puede especificar un grupo federado específico ("managers"):

```
"Principal": { "AWS": "arn:aws:iam::27233906934684427525:federated-
group/Managers"  }
```

- Puede especificar un principal anónimo:

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

- Para evitar ambigüedades, puede utilizar el UUID de usuario en lugar del nombre de usuario:

```
arn:aws:iam::27233906934684427525:user-uuid/de305d54-75b4-431b-adb2-
eb6b9e546013
```

  Por ejemplo, supongamos que Alex abandona la organización y el nombre de usuario `Alex` se ha eliminado. Si un nuevo Alex se une a la organización y se le asigna la misma `Alex` nombre de usuario, es posible que el nuevo usuario herede sin querer los permisos concedidos al usuario original.

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

## Especificar permisos en una directiva

En una directiva, el elemento Acción se utiliza para permitir/denegar permisos a un recurso. Hay un conjunto de permisos que puede especificar en una directiva, que se indican mediante el elemento "Acción" o, alternativamente, "NotAction" para la exclusión. Cada uno de estos elementos se asigna a operaciones de API de REST de S3 específicas.

En las tablas se enumeran los permisos que se aplican a los bloques y los permisos que se aplican a los objetos.

> Amazon S3 utiliza ahora el permiso s3:PutReplicationConfiguration para LAS acciones de replicación PUT y DELETE Bucket. StorageGRID utiliza permisos independientes para cada acción, que coinciden con la especificación original de Amazon S3.

> SE realiza UNA ELIMINACIÓN cuando se utiliza UNA PUESTA para sobrescribir un valor existente.

**Permisos que se aplican a los bloques**

| Permisos | OPERACIONES DE LA API DE REST DE S3 | Personalizado para StorageGRID |
|---|---|---|
| s3:CreateBucket | COLOQUE el cucharón | |
| s3:DeleteBucket | ELIMINAR bloque | |

| Permisos | OPERACIONES DE LA API DE REST DE S3 | Personalizado para StorageGRID |
| --- | --- | --- |
| s3:DeleteBucketMetadataNotification | DELETE bucket metadata notification Configuration | Sí |
| s3:DeleteBucketPolicy | ELIMINE la política de bloques | |
| s3:DeleteReplicationConfiguration | DELETE Bucket replicación | Sí, separe los permisos PARA PUT y DELETE* |
| s3:GetBucketAcl | GET Bucket ACL | |
| s3:GetBucketCompliance | CUMPLIMIENTO de LA normativa GET Bucket (obsoleto) | Sí |
| s3:GetBucketConsistency | OBTENGA coherencia de bloques | Sí |
| s3: GetBucketCORS | OBTENGA los cors del cucharón | |
| s3:GetEncryptionConfiguration | OBTENGA el cifrado de bloque | |
| s3:GetBucketLastAccessTime | HORA de último acceso al bloque DE GET | Sí |
| s3:GetBucketLocation | OBTENER ubicación de bloque | |
| s3:GetBucketMetadataNotification | OBTENGA la configuración de notificación de metadatos del bloque de datos | Sí |
| s3:GetBucketNotification | OBTENGA la notificación DE BUCKET | |
| s3:GetBucketObjectLockConfiguration | OBTENER configuración de bloqueo de objeto | |
| s3:GetBucketPolicy | OBTENGA la política de bloques | |
| s3:GetBucketTagging | GET Bucket tagging | |
| s3:GetBucketVersioning | OBTENGA el control de versiones de Bucket | |
| s3:GetLifecycleConfiguration | OBTENGA el ciclo de vida de la cuchara | |

| Permisos | OPERACIONES DE LA API DE REST DE S3 | Personalizado para StorageGRID |
| --- | --- | --- |
| s3:GetReplicationConfiguration | OBTENGA la replicación de Bucket | |
| s3:ListAllMyBuckets | • OBTENER servicio<br>• Obtenga el uso del almacenamiento | Sí, PARA OBTENER el uso del almacenamiento |
| s3:ListBucket | • GET Bucket (objetos de lista)<br>• Cubo DE CABEZA<br>• Restauración DE objetos posterior | |
| s3:ListBucketMultipartUploads | • Enumerar cargas de varias partes<br>• Restauración DE objetos posterior | |
| s3:ListBucketVersions | OBTENGA las versiones DE Bucket | |
| s3:PutBucketCompliance | CUMPLIMIENTO de PUT Bucket (obsoleto) | Sí |
| s3:PutBucketConsistency | PONGA la consistencia del cucharón | Sí |
| s3: PutBucketCORS | • ELIMINAR los segmentos de cucharón†<br>• COLOQUE los cors del cucharón | |
| s3:PutEncryptionConfiguration | • DELETE Bucket Encryption<br>• PUT Bucket Encryption | |
| s3:PutBucketLastAccessTime | PUT Bucket última hora de acceso | Sí |
| s3:PutBucketMetadataNotification | PUT bucket metadata notification Configuration | Sí |
| s3:PutBucketNotification | NOTIFICACIÓN DE PUT Bucket | |

| Permisos | OPERACIONES DE LA API DE REST DE S3 | Personalizado para StorageGRID |
|---|---|---|
| s3:PutBucketObjectLockConfiguration | • COLOQUE el cucharón con el `x-amz-bucket-object-lock-enabled: true` Encabezado de solicitud (también requiere el permiso s3:CreateBucket)<br>• PONER configuración de bloqueo de objeto | |
| s3:PutBucketPolicy | POLÍTICA DE PUT Bucket | |
| s3:PutBucketEtiquetado | • ELIMINAR etiquetado de bloque†<br>• PUT Bucket etiquetaje | |
| s3:PutBucketVersioning | PONER creación de versiones de bloques | |
| s3:PutLipeycleConfiguration | • ELIMINAR ciclo de vida del cucharón†<br>• CICLO de vida DE la cuchara | |
| s3:PutReplicationConfiguration | PUT Bucket replication | Sí, separe los permisos PARA PUT y DELETE* |

**Permisos que se aplican a objetos**

| Permisos | OPERACIONES DE LA API DE REST DE S3 | Personalizado para StorageGRID |
|---|---|---|
| s3:AbortMultipartUpload | • Cancelar carga de varias partes<br>• Restauración DE objetos posterior | |
| s3:DeleteObject | • ELIMINAR objeto<br>• ELIMINAR varios objetos<br>• Restauración DE objetos posterior | |
| s3:DeleteObjectTagging | ELIMINAR etiquetado de objetos | |
| s3:DeleteObjectVersionTagging | ELIMINAR etiquetado de objetos (una versión específica del objeto) | |

| Permisos | OPERACIONES DE LA API DE REST DE S3 | Personalizado para StorageGRID |
|---|---|---|
| s3:DeleteObjectVersion | ELIMINAR objeto (una versión específica del objeto) | |
| s3:GetObject | • OBTENER objeto<br>• OBJETO HEAD<br>• Restauración DE objetos posterior<br>• SELECCIONE Contenido de objeto | |
| s3:GetObjectAcl | OBTENER ACL de objeto | |
| s3:GetObjectLegalHold | OBTENER retención legal de objetos | |
| s3:GetObjectRetention | OBTENGA retención de objetos | |
| s3:GetObjectTagging | OBTENER etiquetado de objetos | |
| s3:GetObjectVersionTagging | OBTENER etiquetado de objetos (una versión específica del objeto) | |
| s3:GetObjectVersion | GET Object (una versión específica del objeto) | |
| s3:ListMultipartUploadParts | Elementos de lista, restauración POSTERIOR al objeto | |
| s3:PutObject | • OBJETO PUT<br>• PONER objeto: Copiar<br>• Restauración DE objetos posterior<br>• Inicie la carga de varias partes<br>• Completar carga de varias partes<br>• Cargar artículo<br>• Cargar pieza: Copiar | |
| s3:PutObjectLegalHold | PONER objeto legal | |
| s3:PutObjectRetention | PUT Object retention | |
| s3:PutObjectEtiquetado | PONER etiquetado de objetos | |

| Permisos | OPERACIONES DE LA API DE REST DE S3 | Personalizado para StorageGRID |
|---|---|---|
| s3:PutObjectVersionEtiquetado | PONER etiquetado de objetos (una versión específica del objeto) | |
| s3:PutOverwriteObject | • OBJETO PUT<br>• PONER objeto: Copiar<br>• PUT Object tagging<br>• ELIMINAR etiquetado de objetos<br>• Completar carga de varias partes | Sí |
| s3:RestoreObject | Restauración DE objetos posterior | |

## Utilice el permiso PutOverwriteObject

el permiso s3:PutOverwriteObject es un permiso StorageGRID personalizado que se aplica a operaciones que crean o actualizan objetos. La configuración de este permiso determina si el cliente puede sobrescribir los datos de un objeto, metadatos definidos por el usuario o el etiquetado de objetos S3.

Entre los posibles ajustes para este permiso se incluyen:

- **Permitir**: El cliente puede sobrescribir un objeto. Esta es la configuración predeterminada.
- **Denegar**: El cliente no puede sobrescribir un objeto. Cuando se establece en Denegar, el permiso PutOverwriteObject funciona de la siguiente manera:
  - Si se encuentra un objeto existente en la misma ruta:
    - No se pueden sobrescribir los datos, los metadatos definidos por el usuario ni el etiquetado de objetos de S3 del objeto.
    - Se cancela cualquier operación de ingesta en curso y se devuelve un error.
    - Si se habilita el control de versiones de S3, la configuración Denegar evita QUE LAS operaciones PUT Object tagging o DELETE Object tagging modifiquen el conjunto de etiquetas para un objeto y sus versiones no actuales.
  - Si no se encuentra un objeto existente, este permiso no tiene efecto.
- Cuando este permiso no está presente, el efecto es el mismo que si se estableció permitir.

> Si la política actual de S3 permite sobrescribir y el permiso PutOverwriteObject se establece en Deny, el cliente no puede sobrescribir los datos de un objeto, metadatos definidos por el usuario ni el etiquetado de objetos. Además, si la casilla de verificación **evitar modificación de cliente** está activada (**CONFIGURACIÓN** > **sistema** > **Opciones de cuadrícula**), esa configuración anula la configuración del permiso PutOverwriteObject.

**Información relacionada**

Ejemplos de políticas de grupo S3

## Especificar condiciones en una política

Las condiciones definen cuándo estará en vigor una política. Las condiciones consisten en operadores y pares clave-valor.

Condiciones Utilice pares clave-valor para la evaluación. Un elemento Condition puede contener varias condiciones y cada condición puede contener varios pares clave-valor. El bloque Condition 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": {
      "aws:SourceIp": "54.240.143.0/24"
        ...
},
        ...
```

### Operadores de condición admitidos

Los operadores de condición se categorizan de la siguiente manera:

- Cadena
- Numérico
- Booleano
- Dirección IP
- Comprobación nula

| Operadores de condición | Descripción |
|---|---|
| StringEquals | Compara una clave con un valor de cadena basado en la coincidencia exacta (distingue entre mayúsculas y minúsculas). |
| StringNotEquals | Compara una clave con un valor de cadena basado en la coincidencia negada (distingue entre mayúsculas y minúsculas). |
| StringEqualizsIgnoreCase | Compara una clave con un valor de cadena basado en la coincidencia exacta (omite Case). |

| Operadores de condición | Descripción |
| --- | --- |
| StringNotEqualizsIgnoreCase | Compara una clave con un valor de cadena basado en la coincidencia negada (omite Case). |
| StringLike | Compara una clave con un valor de cadena basado en la coincidencia exacta (distingue entre mayúsculas y minúsculas). Puede incluir * y ? caracteres comodín. |
| StringNotLike | Compara una clave con un valor de cadena basado en la coincidencia negada (distingue entre mayúsculas y minúsculas). Puede incluir * y ? caracteres comodín. |
| Valores numéricos | Compara una clave con un valor numérico basado en la coincidencia exacta. |
| NumericNotEquals | Compara una clave con un valor numérico basado en la coincidencia negada. |
| NumericGreatertan | Compara una clave con un valor numérico basado en la coincidencia "'mayor que'". |
| NumericGreaterThanEquals | Compara una clave con un valor numérico basado en la coincidencia "'mayor que o igual'". |
| NumericLessThan | Compara una clave con un valor numérico basado en la coincidencia "'less than'". |
| NumericLesThanEquals | Compara una clave con un valor numérico basado en la coincidencia "'menor que o igual'". |
| Bool | Compara una clave con un valor booleano basado en la coincidencia "'true o false'". |
| IPAddress | Compara una clave con una dirección IP o un rango de direcciones IP. |
| NotIpAddress | Compara una clave con una dirección IP o un intervalo de direcciones IP basándose en la coincidencia negada. |
| Nulo | Comprueba si hay una clave de condición en el contexto actual de la solicitud. |

**Teclas de condición compatibles**

| Categoría | Teclas de condición aplicables | Descripción |
|---|---|---|
| Operadores IP | aws:SourceIp | Comparará con la dirección IP desde la que se envió la solicitud. Se puede utilizar para operaciones de bloques u objetos.<br><br>**Nota:** Si la solicitud S3 se envió a través del servicio Load Balancer en nodos Admin y nodos de Gpuertas de enlace, se comparará con la dirección IP anterior al servicio Load Balancer.<br><br>**Nota**: Si se utiliza un equilibrador de carga no transparente de terceros, se comparará con la dirección IP de ese equilibrador de carga. Cualquiera `X-Forwarded-For` se ignorará el encabezado ya que no se puede comprobar su validez. |
| Recurso/identidad | aws:nombre de usuario | Comparará con el nombre de usuario del remitente desde el que se envió la solicitud. Se puede utilizar para operaciones de bloques u objetos. |
| s3:ListBucket y.<br><br>s3:ListBucketVersions permisos | s3:delimitador | Comparará con el parámetro delimitador especificado en una solicitud GET Bucket o GET Bucket Object Versions. |
| s3:ListBucket y.<br><br>s3:ListBucketVersions permisos | s3:max-keys | Comparará con el parámetro max-keys especificado en una solicitud GET Bucket o GET Bucket Object Versions. |
| s3:ListBucket y.<br><br>s3:ListBucketVersions permisos | s3:prefijo | Se comparará con el parámetro prefix especificado en una solicitud GET Bucket o GET Bucket Object Versions. |

| Categoría | Teclas de condición aplicables | Descripción |
| --- | --- | --- |
| s3:PutObject | s3:retención-días restante del bloqueo de objetos | Compara con la fecha de retención hasta especificada en `x-amz-object-lock-retain-until-date` cabecera de solicitud o calculada desde el período de retención predeterminado de bloque para asegurarse de que estos valores están dentro del intervalo permitido para las siguientes solicitudes:<br>• OBJETO PUT<br>• PONER objeto: Copiar<br>• Inicie la carga de varias partes |
| s3:PutObjectRetention | s3:retención-días restante del bloqueo de objetos | Compara con la fecha de retención especificada en la solicitud DE RETENCIÓN DE objeto PUT para garantizar que se encuentra dentro del intervalo permitido. |

## Especifique las variables en una política

Las variables de las directivas se pueden utilizar para rellenar la información de directivas cuando esté disponible. Se pueden usar variables de política en la `Resource` comparaciones entre elementos y cadenas en la `Condition` elemento.

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

```
"Resource": "arn:aws:s3:::bucket-name/home/${aws:username}/*"
```

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

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

| Variable | Descripción |
| --- | --- |
| `${aws:SourceIp}` | Utiliza la clave SourceIp como la variable proporcionada. |

| Variable | Descripción |
| --- | --- |
| `${aws:username}` | Utiliza la clave de nombre de usuario como la variable proporcionada. |
| `${s3:prefix}` | Utiliza la clave de prefijo específica del servicio como variable proporcionada. |
| `${s3:max-keys}` | Utiliza la clave de max-keys específica del servicio como la variable proporcionada. |
| `${*}` | Carácter especial. Utiliza el carácter como carácter literal *. |
| `${?}` | Carácter especial. Utiliza el carácter como literal ? carácter. |
| `${$}` | Carácter especial. Utiliza el carácter como carácter literal $. |

## Crear directivas que requieran un manejo especial

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

| Descripción de la política | Tipo de política | Comportamiento de Amazon | Comportamiento de StorageGRID |
| --- | --- | --- | --- |
| Denegar a sí mismo cualquier permiso a la cuenta raíz | Cucharón | Válido y reforzado, pero la cuenta de usuario raíz conserva el permiso para todas las operaciones de política de bloques de S3 | Igual |
| Denegar a sí mismo cualquier permiso al usuario o grupo | Grupo | Válido y reforzado | Igual |
| Permitir cualquier permiso para un grupo de cuentas externo | Cucharón | Principal no válido | Válidos, pero los permisos para todas las operaciones de política de bloques de S3 devuelven un método 405 no permitido cuando lo permite una política |

| Descripción de la política | Tipo de política | Comportamiento de Amazon | Comportamiento de StorageGRID |
|---|---|---|---|
| Permitir cualquier permiso para una raíz de cuenta externa o para un usuario | Cucharón | Válidos, pero los permisos para todas las operaciones de política de bloques de S3 devuelven un método 405 no permitido cuando lo permite una política | Igual |
| Permitir que todos tengan permisos para todas las acciones | Cucharón | Válido, pero los permisos para todas las operaciones de política de bloques de S3 devuelven un error de método 405 no permitido para la raíz de cuenta externa y los usuarios | Igual |
| Denegar a todos los permisos a todas las acciones | Cucharón | Válido y reforzado, pero la cuenta de usuario raíz conserva el permiso para todas las operaciones de política de bloques de S3 | Igual |
| Principal es un usuario o grupo inexistente | Cucharón | Principal no válido | Válido |
| El recurso es un bloque de S3 que no existe | Grupo | Válido | Igual |
| El director es un grupo local | Cucharón | Principal no válido | Válido |
| La directiva otorga a una cuenta que no es propietaria (incluidas las cuentas anónimas) permisos para COLOCAR objetos | Cucharón | Válido. Los objetos son propiedad de la cuenta creadora y la política de bucket no se aplica. La cuenta de creador debe otorgar permisos de acceso al objeto mediante ACL de objeto. | Válido. Los objetos son propiedad de la cuenta de propietario del bloque. Se aplica la política de bloques. |

## Protección WORM (escritura única lectura múltiple)

Se pueden crear bloques DE escritura única y lectura múltiple (WORM) para proteger los datos, los metadatos de objetos definidos por el usuario y el etiquetado de objetos de S3. Puede configurar los bloques WORM para permitir la creación de objetos nuevos y evitar sobrescrituras o eliminaciones del contenido existente. Utilice uno de los enfoques aquí descritos.

Para asegurarse de que las sobrescrituras se deniegan siempre, puede:

- En Grid Manager, vaya a **CONFIGURACIÓN** > **sistema** > **Opciones de cuadrícula** y seleccione la casilla de verificación **evitar modificación de cliente**.
- Aplique las siguientes reglas y políticas de S3:
  - Agregue una operación PUTOVERWRITEOBJECT DENY a la directiva S3.
  - Agregue una operación DeleteObject DENY a la directiva S3.
  - Añada una operación PUT Object ALLOW a la política de S3.

> Al establecer DeleteObject en DENEGAR en una directiva S3, no se impide que ILM elimine objetos cuando existe una regla como "'copias cero después de 30 días'".

> Incluso cuando se aplican todas estas reglas y políticas, no se protegen contra las escrituras simultáneas (véase la situación A). Protegen contra sobrescrituras completadas secuenciales (consulte la situación B).

**Situación A**: Escrituras simultáneas (no protegidas contra)

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

**Situación B**: Sobrescrituras completadas secuenciales (protegidas contra)

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

**Información relacionada**

Gestión de objetos con ILM

Crear directivas que requieran un manejo especial

Cómo gestionan las reglas de ILM de StorageGRID los objetos

Ejemplos de políticas de grupo S3

## Ejemplos de políticas de S3

Utilice los ejemplos de esta sección para crear políticas de acceso de StorageGRID para bloques y grupos.

### Ejemplos de políticas de bloques de S3

Las políticas de bloque especifican los permisos de acceso para el bloque al que está asociada la directiva. Las políticas de bloque se configuran mediante la API de S3 PutBucketPolicy.

Se puede configurar una política de bloques mediante la CLI de AWS según el siguiente comando:

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

**Ejemplo: Permitir que todos tengan acceso de solo lectura a un bloque**

En este ejemplo, todos, incluido el anónimo, pueden enumerar objetos en el bloque y realizar operaciones Get Object en todos los objetos del bloque. Se denegarán todas las demás operaciones. Tenga en cuenta que esta directiva podría no ser particularmente útil ya que nadie, excepto la raíz de la cuenta, tiene permisos para escribir en el bloque.

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

**Ejemplo: Permitir que todos en una cuenta tengan acceso total y que todas las personas de otra cuenta tengan acceso de solo lectura a un bloque**

En este ejemplo, se permite a todos los integrantes de una cuenta especificada el acceso completo a un bloque, mientras que a todos los miembros de otra cuenta especificada sólo se les permite enumerar el bloque y realizar operaciones GetObject en los objetos del bloque empezando por el `shared/` prefijo de clave de objeto.

> En StorageGRID, los objetos creados por una cuenta que no es propietaria (incluidas las cuentas anónimas) son propiedad de la cuenta de propietario del bloque. La política de bloque se aplica a estos objetos.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "95390887230002558202"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "31181711887329436680"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::examplebucket/shared/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "31181711887329436680"
      },
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::examplebucket",
      "Condition": {
        "StringLike": {
          "s3:prefix": "shared/*"
        }
      }
    }
  ]
}
```

**Ejemplo: Permitir que todo el mundo tenga acceso de solo lectura a un bloque y acceso completo por un grupo especificado**

En este ejemplo, todos los usuarios, incluido el anónimo, pueden enumerar el bloque y realizar operaciones GET Object en todos los objetos del bloque, mientras que sólo los usuarios que pertenecen al grupo `Marketing` en la cuenta especificada se permite el acceso completo.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-
group/Marketing"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": ["s3:ListBucket","s3:GetObject"],
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    }
  ]
}
```

**Ejemplo: Permitir que todo el mundo tenga acceso de lectura y escritura a un bloque si un cliente se encuentra en el rango de IP**

En este ejemplo, todos, incluido el anónimo, pueden enumerar el bloque y realizar cualquier operación Object en todos los objetos del bloque, siempre que las solicitudes provengan de un intervalo IP especificado (54.240.143.0 a 54.240.143.255, excepto 54.240.143.188). Se denegarán todas las demás operaciones y se denegarán todas las solicitudes que estén fuera del rango de IP.

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

**Ejemplo: Permitir el acceso completo a un bloque exclusivamente por un usuario federado especificado**

En este ejemplo, el usuario federado Alex tiene permiso de acceso completo al `examplebucket` cucharón y sus objetos. A todos los demás usuarios, incluido "root", se les deniega explícitamente todas las operaciones. Tenga en cuenta, sin embargo, que "root" nunca se le deniegan los permisos para poner/obtener/DeleteBucketPolicy.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-user/Alex"
      },
      "Action": [
        "s3:*"
      ],
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Deny",
      "NotPrincipal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-user/Alex"
      },
      "Action": [
        "s3:*"
      ],
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    }
  ]
}
```

**Ejemplo: Permiso PutOverwriteObject**

En este ejemplo, la `Deny` Effect para PutOverwriteObject y DeleteObject garantiza que nadie puede sobrescribir ni eliminar los datos del objeto, los metadatos definidos por el usuario y el etiquetado de objetos S3.

```
{
  "Statement": [
    {
      "Effect": "Deny",
      "Principal": "*",
      "Action": [
        "s3:PutOverwriteObject",
        "s3:DeleteObject",
        "s3:DeleteObjectVersion"
      ],
      "Resource": "arn:aws:s3:::wormbucket/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-
group/SomeGroup"

},
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::wormbucket"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-
group/SomeGroup"

},
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::wormbucket/*"
    }
  ]
}
```

**Información relacionada**

Operaciones en bloques

### Ejemplos de políticas de grupo S3

Las directivas de grupo especifican los permisos de acceso para el grupo al que está asociada la directiva. No existe `Principal` elemento de la política, ya que está implícito. Las políticas de grupo se configuran con el administrador de inquilinos o la API.

#### Ejemplo: Establecer la directiva de grupo mediante el Administrador de inquilinos

Cuando utilice el Administrador de inquilinos para agregar o editar un grupo, puede seleccionar cómo desea crear la política de grupo que define qué miembros de permisos de acceso S3 de este grupo tendrán, de la siguiente manera:

- **Sin acceso S3**: Opción predeterminada. Los usuarios de este grupo no tienen acceso a los recursos de S3, a menos que se conceda el acceso mediante una política de bloques. Si selecciona esta opción, de forma predeterminada, solo el usuario raíz tendrá acceso a recursos de S3.
- **Acceso de sólo lectura**: Los usuarios de este grupo tienen acceso de sólo lectura a los recursos S3. Por ejemplo, los usuarios de este grupo pueden enumerar objetos y leer datos de objetos, metadatos y etiquetas. Cuando selecciona esta opción, la cadena JSON para una política de grupo de solo lectura aparece en el cuadro de texto. No puede editar esta cadena.
- **Acceso completo**: Los usuarios de este grupo tienen acceso completo a los recursos S3, incluidos los bloques. Cuando selecciona esta opción, la cadena JSON para una política de grupo de acceso completo aparece en el cuadro de texto. No puede editar esta cadena.
- **Personalizado**: A los usuarios del grupo se les conceden los permisos que especifique en el cuadro de texto.

  En este ejemplo, sólo se permite a los miembros del grupo que enumeren y tengan acceso a su carpeta específica (prefijo de clave) en el bloque especificado.


#### Ejemplo: Permite el acceso total de grupos a todos los bloques

En este ejemplo, a todos los miembros del grupo se les permite el acceso completo a todos los segmentos que pertenecen a la cuenta de inquilino, a menos que la política de bloque lo deniegue explícitamente.

```
{
  "Statement": [
    {
      "Action": "s3:*",
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::*"
    }
  ]
}
```

**Ejemplo: Permitir el acceso de solo lectura de grupo a todos los bloques**

En este ejemplo, todos los miembros del grupo tienen acceso de solo lectura a recursos S3, a menos que la política de bloque lo deniegue explícitamente. Por ejemplo, los usuarios de este grupo pueden enumerar objetos y leer datos de objetos, metadatos y etiquetas.

```
{
  "Statement": [
    {
      "Sid": "AllowGroupReadOnlyAccess",
      "Effect": "Allow",
      "Action": [
        "s3:ListAllMyBuckets",
        "s3:ListBucket",
        "s3:ListBucketVersions",
        "s3:GetObject",
        "s3:GetObjectTagging",
        "s3:GetObjectVersion",
        "s3:GetObjectVersionTagging"
      ],
      "Resource": "arn:aws:s3:::*"
    }
  ]
}
```

**Ejemplo: Permitir a los miembros del grupo el pleno acceso sólo a su «carpeta» en un cubo**

En este ejemplo, sólo se permite a los miembros del grupo que enumeren y tengan acceso a su carpeta específica (prefijo de clave) en el bloque especificado. Tenga en cuenta que los permisos de acceso de otras políticas de grupo y la directiva de bloque deben tenerse en cuenta al determinar la privacidad de estas carpetas.

```
{
  "Statement": [
    {
      "Sid": "AllowListBucketOfASpecificUserPrefix",
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::department-bucket",
      "Condition": {
        "StringLike": {
          "s3:prefix": "${aws:username}/*"
        }
      }
    },
    {
      "Sid": "AllowUserSpecificActionsOnlyInTheSpecificUserPrefix",
      "Effect": "Allow",
      "Action": "s3:*Object",
      "Resource": "arn:aws:s3:::department-bucket/${aws:username}/*"
    }
  ]
}
```

**Información relacionada**

Usar cuenta de inquilino

# Configure la seguridad de la API de REST

Debe revisar las medidas de seguridad implementadas para la API REST y entender cómo proteger el sistema.

## Cómo proporciona StorageGRID seguridad para la API de REST

Debe entender cómo el sistema StorageGRID implementa la seguridad, la autenticación y la autorización para la API DE REST.

StorageGRID usa las siguientes medidas de seguridad.

- Las comunicaciones de cliente con el servicio Load Balancer utilizan HTTPS si HTTPS está configurado para el extremo de equilibrio de carga.

  Al configurar un extremo de equilibrio de carga, HTTP se puede habilitar opcionalmente. Por ejemplo, puede usar HTTP para pruebas u otros fines que no sean de producción. Consulte las instrucciones para administrar StorageGRID si desea obtener más información.

- De forma predeterminada, StorageGRID utiliza HTTPS para las comunicaciones del cliente con los nodos de almacenamiento y el servicio CLB en los nodos de puerta de enlace.

  Opcionalmente, HTTP se puede habilitar para estas conexiones. Por ejemplo, puede usar HTTP para

pruebas u otros fines que no sean de producción. Consulte las instrucciones para administrar StorageGRID si desea obtener más información.

> El servicio CLB está obsoleto.

- Las comunicaciones entre StorageGRID y el cliente se cifran mediante TLS.
- Las comunicaciones entre el servicio Load Balancer y los nodos de almacenamiento de la cuadrícula están cifradas si el extremo de equilibrio de carga está configurado para aceptar conexiones HTTP o HTTPS.
- Los clientes deben proporcionar encabezados de autenticación HTTP a StorageGRID para realizar operaciones de API DE REST.

**Certificados de seguridad y aplicaciones cliente**

Los clientes pueden conectarse al servicio Load Balancer en los nodos de Gateway o de administrador, directamente a los nodos de almacenamiento o al servicio CLB en los nodos de Gateway.

En todos los casos, las aplicaciones cliente pueden realizar conexiones TLS mediante un certificado de servidor personalizado cargado por el administrador de grid o un certificado generado por el sistema StorageGRID:

- Cuando las aplicaciones cliente se conectan al servicio Load Balancer, lo hacen utilizando el certificado que se configuró para el extremo de equilibrio de carga específico utilizado para realizar la conexión. Cada extremo tiene su propio certificado, que es un certificado de servidor personalizado cargado por el administrador de grid o un certificado que el administrador de grid generó en StorageGRID al configurar el extremo.
- Cuando las aplicaciones cliente se conectan directamente a un nodo de almacenamiento o al servicio CLB en los nodos de puerta de enlace, utilizan los certificados de servidor generados por el sistema que se generaron para los nodos de almacenamiento cuando se instaló el sistema StorageGRID (firmados por la autoridad de certificación del sistema), o un único certificado de servidor personalizado que un administrador de grid suministra para la cuadrícula.

Los clientes deben configurarse para que confíen en la entidad emisora de certificados que firmó el certificado que utilicen para establecer conexiones TLS.

Consulte las instrucciones para administrar StorageGRID para obtener información sobre la configuración de extremos de equilibrador de carga y para obtener instrucciones sobre cómo agregar un único certificado de servidor personalizado para conexiones TLS directamente a nodos de almacenamiento o al servicio CLB en nodos de puerta de enlace.

**Resumen**

En la siguiente tabla, se muestra cómo se implementan los problemas de seguridad en las API DE REST de S3 y Swift:

| Problema de seguridad | Implementación de la API DE REST |
|---|---|
| Seguridad de la conexión | TLS |
| Autenticación del servidor | Certificado de servidor X.509 firmado por CA del sistema o certificado de servidor personalizado suministrado por el administrador |

| Problema de seguridad | Implementación de la API DE REST |
| --- | --- |
| Autentificación de clientes | • S3: Cuenta de S3 (ID de clave de acceso y clave de acceso secreta)<br>• Swift: Cuenta de Swift (nombre de usuario y contraseña) |
| Autorización de cliente | • S3: Propiedad de bloque y todas las políticas de control de acceso aplicables<br>• Swift: Acceso a roles de administrador |

**Información relacionada**

Administre StorageGRID

## Algoritmos de cifrado y hash compatibles para bibliotecas TLS

El sistema StorageGRID admite un conjunto limitado de conjuntos de cifrado que las aplicaciones cliente pueden utilizar al establecer una sesión de seguridad de la capa de transporte (TLS).

### Versiones compatibles de TLS

StorageGRID admite TLS 1.2 y TLS 1.3.

SSLv3 y TLS 1.1 (o versiones anteriores) ya no son compatibles.

### Paquetes de cifrado compatibles

| Versión TLS | Nombre IANA de conjunto cifrado |
| --- | --- |
| 1.2 | TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 |
| 1.2 | TLS_ECDHE_RSA_WITH_CHA20_POLY1305_SHA256 |
| 1.2 | TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 |
| 1.3 | TLS_AES_256_GCM_SHA384 |
| 1.3 | TLS_CHA20_POLY1305_SHA256 |
| 1.3 | TLS_AES_128_GCM_SHA256 |

### Suites de cifrado obsoletas

Los siguientes conjuntos de cifrado están desaprobados. La compatibilidad con estos cifrados se eliminará en una versión futura.

| Nombre de IANA |
| --- |
| TLS_RSA_WITH_AES_128_GCM_SHA256 |
| TLS_RSA_WITH_AES_256_GCM_SHA384 |

**Información relacionada**

Cómo se pueden configurar las conexiones de clientes

# Supervisar y auditar operaciones

Puede supervisar las cargas de trabajo y las eficiencias de las operaciones del cliente al ver las tendencias de las transacciones de todo el grid o de nodos específicos. Puede utilizar mensajes de auditoría para supervisar las operaciones y transacciones del cliente.

## Supervise las tasas de procesamiento y recuperación de objetos

Es posible supervisar las tasas de procesamiento y recuperación de objetos, así como las métricas para el número de objetos, consultas y verificación. Puede ver el número de intentos fallidos y correctos por las aplicaciones cliente para leer, escribir y modificar objetos en el sistema StorageGRID.

**Pasos**

1. Inicie sesión en Grid Manager mediante una navegador web compatible.
2. En la consola, busque la sección Operaciones de protocolo.

   En esta sección se resume el número de operaciones de cliente que realiza su sistema StorageGRID. La media de las tasas de protocolo se hace durante los últimos dos minutos.

3. Seleccione **NODES**.
4. En la página de inicio de nodos (nivel de implementación), haga clic en la ficha **Load Balancer**.

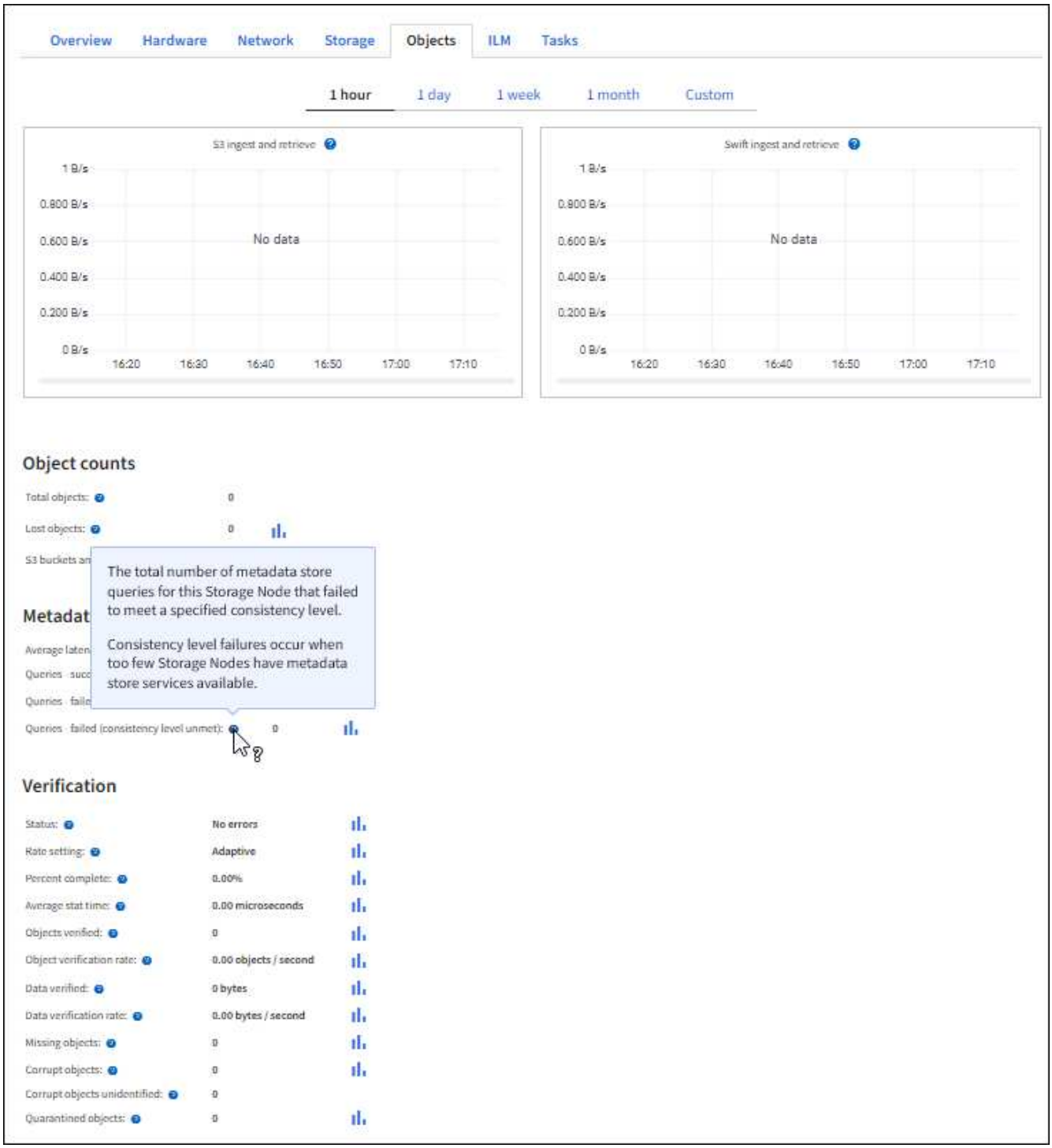
   Los gráficos muestran tendencias para todo el tráfico de cliente dirigido a los extremos de equilibrador de carga dentro de la cuadrícula. Es posible seleccionar un intervalo de tiempo en horas, días, semanas, meses o años. también puede aplicar un intervalo personalizado.

5. En la página de inicio de nodos (nivel de implementación), haga clic en la ficha **objetos**.

   El gráfico muestra las tasas de procesamiento y recuperación de todo el sistema StorageGRID en bytes por segundo y bytes totales. Es posible seleccionar un intervalo de tiempo en horas, días, semanas, meses o años. también puede aplicar un intervalo personalizado.

6. Para ver información sobre un nodo de almacenamiento en particular, seleccione el nodo en la lista de la izquierda y haga clic en la ficha **objetos**.

   El gráfico muestra las tasas de procesamiento y recuperación de objetos de este nodo de almacenamiento. La pestaña también incluye métricas para el recuento de objetos, consultas y verificación. Puede hacer clic en las etiquetas para ver las definiciones de estas métricas.




7. Si desea aún más detalles:
   a. Seleccione **SUPPORT** > **Tools** > **Topología de cuadrícula**.
   b. Seleccione ***site*** > **Descripción general** > **Principal**.

      La sección API Operations muestra información resumida de la cuadrícula completa.

   c. Seleccione ***Storage Node*** > **LDR** > ***Client Application*** > **Overview** > **Main**

      La sección Operaciones muestra información de resumen del nodo de almacenamiento seleccionado.

## Acceder y revisar registros de auditoría

Los servicios de StorageGRID generan los mensajes de auditoría y se almacenan en archivos de registro de texto. Los mensajes de auditoría específicos de API de los registros de auditoría ofrecen datos críticos de seguridad, operación y supervisión del rendimiento que pueden ayudar a evaluar el estado del sistema.

**Lo que necesitará**

- Tiene permisos de acceso específicos.
- Usted tiene la `Passwords.txt` archivo.
- Conoce la dirección IP de un nodo de administrador.

**Acerca de esta tarea**

Se denomina el archivo de registro de auditoría activo `audit.log`, Y se almacena en los nodos Admin.

Una vez al día, se guarda el archivo audit.log activo, y otro nuevo `audit.log` se ha iniciado el archivo. El nombre del archivo guardado indica cuándo se guardó, en el formato *`yyyy-mm-dd.txt`*.

Después de un día, el archivo guardado se comprime y cambia su nombre, en el formato *`yyyy-mm-dd.txt.gz`*, que conserva la fecha original.

En este ejemplo se muestra el activo `audit.log` archivo, el archivo del día anterior (`2018-04-15.txt`), y el archivo comprimido del día anterior (`2018-04-14.txt.gz`).

```
audit.log
2018-04-15.txt
2018-04-14.txt.gz
```

**Pasos**

1. Inicie sesión en un nodo de administrador:
   a. Introduzca el siguiente comando: `ssh admin@primary_Admin_Node_IP`
   b. Introduzca la contraseña que aparece en `Passwords.txt` archivo.
2. Vaya al directorio que contiene los archivos del registro de auditoría:

   ```
   cd /var/local/audit/export
   ```

3. Ver el archivo de registro de auditoría actual o guardado, según sea necesario.

**Se realizó un seguimiento de las operaciones de S3 en los registros de auditoría**

Se realiza un seguimiento de varias operaciones de bloques y de objetos en los registros de auditoría de StorageGRID.

**Se realizó un seguimiento de las operaciones de bloque en los registros de auditoría**

- ELIMINAR bloque
- DELETE Bucket tagging
- ELIMINAR varios objetos
- GET Bucket (objetos de lista)
- OBTENGA las versiones DE objeto Bucket
- GET Bucket tagging
- Cubo DE CABEZA
- COLOQUE el cucharón
- CUMPLIR con la normativa de los bloques
- PUT Bucket etiquetaje
- PONER creación de versiones de bloques

**Se realizó un seguimiento de las operaciones de objetos en los registros de auditoría**

- Completar carga de varias partes
- Cargar pieza (cuando la regla ILM usa los comportamientos de consumo estrictos o equilibrados)
- Cargar pieza: Copia (cuando la regla ILM usa los comportamientos de ingesta estrictos o equilibrados)
- ELIMINAR objeto
- OBTENER objeto
- OBJETO HEAD
- Restauración DE objetos posterior
- OBJETO PUT
- PONER objeto: Copiar

**Información relacionada**

Operaciones en bloques

Operaciones en objetos

# Ventajas de las conexiones HTTP activas, inactivas y simultáneas

La forma en que se configuran las conexiones HTTP puede afectar el rendimiento del sistema StorageGRID. Las configuraciones varían en función de si la conexión HTTP está activa o inactiva o si tiene varias conexiones simultáneas.

Puede identificar las ventajas en el rendimiento de los siguientes tipos de conexiones HTTP:

- Conexiones HTTP inactivas
- Conexiones HTTP activas
- Conexiones HTTP simultáneas

## Ventajas de mantener abiertas las conexiones HTTP inactivas

Debe mantener las conexiones HTTP abiertas incluso cuando las aplicaciones cliente están inactivas para permitir que las aplicaciones cliente realicen transacciones posteriores a través de la conexión abierta. Basándose en las mediciones del sistema y en la experiencia de integración, debe mantener abierta una conexión HTTP inactiva durante un máximo de 10 minutos. StorageGRID puede cerrar automáticamente una conexión HTTP que se mantenga abierta y inactiva durante más de 10 minutos.

Las conexiones HTTP abiertas y inactivas proporcionan las siguientes ventajas:

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

  La latencia reducida es la ventaja principal, especialmente por la cantidad de tiempo necesario para establecer las conexiones TCP/IP y TLS.

- Aumento de la velocidad de transferencia de datos mediante la preparación del algoritmo de inicio lento TCP/IP con transferencias realizadas previamente
- Notificación instantánea de varias clases de condiciones de fallo que interrumpen la conectividad entre la aplicación cliente y el sistema StorageGRID

Determinar durante cuánto tiempo mantener abierta una conexión inactiva-es un intercambio entre las ventajas del inicio lento que se asocia a la conexión existente y la asignación ideal de la conexión a los recursos internos del sistema.

## Ventajas de las conexiones HTTP activas

Para conexiones directamente a nodos de almacenamiento o al servicio CLB (obsoleto) en nodos de puerta de enlace, debe limitar la duración de una conexión HTTP activa a un máximo de 10 minutos, incluso si la conexión HTTP realiza transacciones continuamente.

La determinación de la duración máxima de la apertura de una conexión es un intercambio-entre los beneficios de la persistencia de la conexión y la asignación ideal de la conexión a los recursos internos del sistema.

Para conexiones de clientes a nodos de almacenamiento o al servicio CLB, limitar las conexiones HTTP activas proporciona las siguientes ventajas:

- Permite un balanceo de carga óptimo en el sistema StorageGRID.

  Cuando utilice el servicio CLB, debe evitar conexiones TCP/IP de larga duración-para optimizar el equilibrio de carga en todo el sistema StorageGRID. Debe configurar las aplicaciones cliente para realizar un seguimiento de la duración de cada conexión HTTP y cerrar la conexión HTTP después de una hora establecida para que la conexión HTTP se pueda restablecer y reequilibrar.

  El servicio CLB equilibra la carga a través del sistema StorageGRID en el momento en que una aplicación cliente establece una conexión HTTP. Con el tiempo, es posible que una conexión HTTP ya no sea óptima a medida que cambian 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 niega las ganancias mucho más valiosas asociadas con conexiones persistentes.

El servicio CLB está obsoleto.

- Permite a las aplicaciones cliente dirigir transacciones HTTP a servicios LDR que tengan espacio disponible.
- Permite iniciar los procedimientos de mantenimiento.

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

En el caso de las conexiones cliente al servicio Load Balancer, limitar la duración de las conexiones abiertas puede ser útil para permitir que algunos procedimientos de mantenimiento se inicien con rapidez. Si la duración de las conexiones cliente no es limitada, las conexiones activas pueden tardar varios minutos en terminarse automáticamente.

## Ventajas de las conexiones HTTP simultáneas

Debe mantener abiertas varias conexiones TCP/IP al sistema StorageGRID para permitir el paralelismo, lo que aumenta el rendimiento. El número óptimo de conexiones paralelas depende de diversos factores.

Las conexiones HTTP simultáneas proporcionan las siguientes ventajas:

- Latencia reducida

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

- Aumento de la productividad

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

Las aplicaciones cliente deben establecer varias conexiones HTTP. Cuando una aplicación cliente tiene que realizar una transacción, puede seleccionar y utilizar inmediatamente cualquier conexión establecida que no esté procesando actualmente una transacción.

Antes de que el rendimiento empiece a degradarse, cada topología de los sistemas StorageGRID tiene un rendimiento máximo diferente para transacciones y conexiones simultáneas. El rendimiento máximo depende de factores como los recursos informáticos, los recursos de red, los recursos de almacenamiento y los enlaces WAN. También son factores que influyen en el número de servidores y servicios y el número de aplicaciones que admite el sistema StorageGRID.

A menudo, los sistemas StorageGRID admiten varias aplicaciones cliente. Debe tener esto en cuenta al determinar el número máximo de conexiones simultáneas que utiliza una aplicación cliente. Si la aplicación cliente consta de varias entidades de software que cada una establece conexiones al sistema StorageGRID, debe agregar todas las conexiones a través de las entidades. Es posible que tenga que ajustar el número máximo de conexiones simultáneas en las siguientes situaciones:

- La topología del sistema StorageGRID afecta al número máximo de transacciones y conexiones simultáneas que puede admitir el sistema.
- Las aplicaciones cliente que interactúan con el sistema StorageGRID a través de una red con ancho de banda limitado pueden tener que reducir el grado de concurrencia para garantizar que las transacciones

individuales se completen en un tiempo razonable.

- Cuando muchas aplicaciones cliente comparten el sistema StorageGRID, puede que tenga que reducir el nivel de concurrencia para evitar superar los límites del sistema.

## Separación de grupos de conexiones HTTP para operaciones de lectura y escritura

Puede utilizar pools independientes de conexiones HTTP para operaciones de lectura y escritura y controlar la cantidad de un pool que debe utilizar para cada uno. Los grupos separados de conexiones HTTP le permiten controlar mejor las transacciones y equilibrar las cargas.

Las aplicaciones cliente pueden crear cargas que sean dominantes de la recuperación (lectura) o del almacén (escritura). Con grupos separados de conexiones HTTP para transacciones de lectura y escritura, puede ajustar la cantidad de cada pool que se va a dedicar a transacciones de lectura o escritura.

## Información de copyright



## Información de la marca comercial