Utilize a API REST S3

StorageGRID software

NetApp
July 23, 2026

Índice

Utilize a API REST S3

Versões e atualizações compatíveis da API REST S3 do StorageGRID

StorageGRID oferece suporte à API Simple Storage Service (S3), que é implementada como um conjunto de serviços web API REST.

O suporte à API REST S3 permite que você conecte aplicativos orientados a serviços, desenvolvidos para serviços web do S3, com storage de objetos local que utiliza o sistema StorageGRID. São necessárias alterações mínimas no uso atual das chamadas à API REST S3 pelo aplicativo cliente.

Versões suportadas

StorageGRID suporta as seguintes versões específicas de STS, S3 e HTTP:

Item	Versão
Especificação da API STS AssumeRole	"Documentação da Amazon Web Services (AWS): Referência da API Amazon Assumerole" Para obter mais informações sobre AssumeRole, consulte "Configurar AssumeRole".
Especificação da API S3	"Documentação da AWS: Referência da API do Amazon Simple Storage Service"
HTTP	1,1 Para obter mais informações sobre HTTP, consulte HTTP/1.1 (RFCs 7230-35). "IETF RFC 2616: Protocolo de Transferência de Hipertexto (HTTP/1.1)" **Nota**: StorageGRID não suporta HTTP/1.1 pipelining.

Atualizações no suporte à API REST do S3

Release	Comentários
12,0	• Adicionada a funcionalidade de compartilhamento de recursos entre origens (CORS) para uma interface de gerenciamento, permitindo que outro domínio acesse dados no StorageGRID usando APIs de gerenciamento. "Saiba mais". • Adicionado suporte para STS AssumeRole e política de sessão. Consulte "um exemplo de política de sessão". Você pode configurar AssumeRole em grupos de locatários.

Release	Comentários
11,9	• Adicionamos suporte para valores de checksum SHA-256 pré-calculados para as seguintes solicitações e cabeçalhos compatíveis. Você pode usar este recurso para verificar a integridade dos objetos enviados: ◦ CompleteMultipartUpload: `x-amz-checksum-sha256` ◦ CreateMultipartUpload: `x-amz-checksum-algorithm` ◦ GetObject: `x-amz-checksum-mode` ◦ HeadObject: `x-amz-checksum-mode` ◦ ListParts ◦ PutObject: `x-amz-checksum-sha256` ◦ UploadPart: `x-amz-checksum-sha256` • Adicionada a capacidade para o administrador da grid controlar as configurações de retenção e Compliance em nível de tenant. Essas configurações afetam as configurações de S3 Object Lock. ◦ Modo de retenção padrão do bucket e modo de retenção do objeto: governança ou conformidade, se permitido pelo administrador do grid. ◦ Período de retenção padrão do bucket e data de retenção do objeto: devem ser menores ou iguais ao permitido pelo período de retenção máximo definido pelo administrador do grid. • Suporte aprimorado para `aws-chunked`codificação de conteúdo e streaming `x-amz-content-sha256` valores. Limitações: ◦ Se presente, `chunk-signature`é opcional e não é validado ◦ Se presente, `x-amz-trailer` o conteúdo é ignorado
11,8	Atualizamos os nomes das operações do S3 para corresponder aos nomes usados no "Documentação da Amazon Web Services (AWS): Referência da API do Amazon Simple Storage Service".
11,7	• Adicionado "Referência rápida: solicitações de API S3 compatíveis". • Adicionado suporte para o uso do modo GOVERNANCE com S3 Object Lock. • Adicionado suporte ao cabeçalho de resposta específico do StorageGRID `x-ntap-sg-cgr-replication-status` para solicitações GET Object e HEAD Object. Este cabeçalho fornece o status de replicação de um objeto para replicação entre grids. • SelectObjectContent requests agora suportam objetos Parquet.

Release	Comentários
11,6	• Adicionado suporte para usar o `partNumber`parâmetro request em solicitações GET Object e HEAD Object. • Adicionado suporte para um modo de retenção padrão e um período de retenção padrão no nível do bucket para o S3 Object Lock. • Adicionado suporte para a chave de condição de política `s3:object-lock-remaining-retention-days` para definir o intervalo de períodos de retenção permitidos para seus objetos. • Alteramos o tamanho máximo *recomendado* para uma única operação PUT Object para 5 GiB (5.368.709.120 bytes). Se você tiver objetos maiores que 5 GiB, use upload multipart.
11,5	• Adicionado suporte para gerenciamento de criptografia de bucket. • Adicionado suporte para S3 Object Lock e descontinuadas as solicitações de Compliance legadas. • Adicionado suporte para usar DELETE Multiple Objects em buckets versionados. • O `Content-MD5`cabeçalho da solicitação agora é suportado corretamente.
11,4	• Adicionado suporte para DELETE Bucket tagging, GET Bucket tagging e PUT Bucket tagging. Tags de alocação de custos não são suportadas. • Para buckets criados no StorageGRID 11.4, restringir os nomes das chaves de objeto para atender às melhores práticas de desempenho não é mais necessário. • Adicionado suporte para notificações de bucket no tipo de evento `s3:ObjectRestore:Post`. • Os limites de tamanho da AWS para partes multipartes agora são aplicados. Cada parte em um upload multipart deve ter entre 5 MiB e 5 GiB. A última parte pode ser menor que 5 MiB. • Adicionado suporte para TLS 1.3
11,3	• Adicionado suporte para criptografia do lado do servidor dos dados de objetos com chaves fornecidas pelo cliente (SSE-C). • Adicionado suporte para as operações de ciclo de vida do bucket DELETE, GET e PUT (somente ação de expiração) e para o cabeçalho de resposta `x-amz-expiration`. • Atualizamos os artigos PUT Object, PUT Object - Copy e Multipart Upload para descrever o impacto das regras de gerenciamento do ciclo de vida das informações (ILM) que utilizam posicionamento síncrono na ingestão. • Os algoritmos de criptografia TLS 1.1 não são mais suportados.

Release	Comentários
11,2	Adicionado suporte para restauração de objeto via POST para uso com Cloud Storage Pools. Adicionado suporte para o uso da sintaxe da AWS para ARN, chaves de condição de política e variáveis de política em políticas de grupo e bucket. As políticas de grupo e bucket existentes que usam a sintaxe do StorageGRID continuarão sendo suportadas. **Nota:** Os usos de ARN/URN em outros arquivos de configuração JSON/XML, incluindo aqueles usados em recursos personalizados do StorageGRID, não foram alterados.
11,1	Adicionado suporte para compartilhamento de recursos entre origens (CORS), HTTP para conexões de clientes S3 com nós de grid e configurações de conformidade em buckets.
11,0	Adicionamos suporte para configurar serviços de plataforma (CloudMirror replicação, notificações e integração de pesquisa com o Elasticsearch) para buckets. Também adicionamos suporte para restrições de localização de marcação de objetos para buckets e para a consistência de disponibilidade.
10,4	Adicionado suporte para verificação ILM de alterações no controle de versão, atualizações da página de nomes de domínio de endpoints, condições e variáveis em políticas, exemplos de políticas e a permissão PutOverwriteObject.
10,3	Adicionado suporte para versionamento.
10,2	Adicionado suporte para políticas de acesso a grupos e buckets, e para cópia multipart (Upload Part - Copy).
10,1	Adicionado suporte para upload multipart, solicitações no estilo de hospedagem virtual e autenticação v4.
10,0	Suporte inicial da API REST S3 pelo sistema StorageGRID. A versão atualmente suportada da *Simple Storage Service API Reference* é 2006-03-01.

Solicitações de API S3 suportadas no StorageGRID

Esta página resume como StorageGRID oferece suporte às APIs do Amazon Simple Storage Service (S3).

Esta página inclui apenas as operações do S3 que são suportadas pelo StorageGRID.

Para visualizar a documentação da AWS para cada operação, selecione o link no cabeçalho.

Parâmetros de consulta URI comuns e cabeçalhos de solicitação

Salvo indicação em contrário, os seguintes parâmetros de consulta URI comuns são suportados:

- `versionId` (conforme necessário para operações com objetos)

Salvo indicação em contrário, os seguintes cabeçalhos de solicitação comuns são suportados:

- `Authorization`
- `Connection`
- `Content-Length`
- `Content-MD5`
- `Content-Type`
- `Date`
- `Expect`
- `Host`
- `x-amz-date`

Informações relacionadas

- "Detalhes da implementação da API REST do S3"
- "Referência da API do Amazon Simple Storage Service: cabeçalhos de solicitação comuns"

"AbortMultipartUpload"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todos os parâmetros e cabeçalhos comuns para esta solicitação, além deste parâmetro de consulta URI adicional:

- `uploadId`

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações para uploads multipartes"

"CompleteMultipartUpload"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todos os parâmetros e cabeçalhos comuns para esta solicitação, além deste parâmetro de consulta URI adicional:

- `uploadId`
- `x-amz-checksum-sha256`

Tags XML do corpo da requisição

StorageGRID suporta estas tags XML no corpo da requisição:

- `ChecksumSHA256`
- `CompleteMultipartUpload`
- `ETag`

- `Part`
- `PartNumber`

Documentação do StorageGRID

"CompleteMultipartUpload"

"CopyObject"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todos parâmetros e cabeçalhos comuns para esta solicitação, além destes cabeçalhos adicionais:

- `x-amz-copy-source`
- `x-amz-copy-source-if-match`
- `x-amz-copy-source-if-modified-since`
- `x-amz-copy-source-if-none-match`
- `x-amz-copy-source-if-unmodified-since`
- `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-metadata-directive`
- `x-amz-object-lock-legal-hold`
- `x-amz-object-lock-mode`
- `x-amz-object-lock-retain-until-date`
- `x-amz-server-side-encryption`
- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`
- `x-amz-storage-class`
- `x-amz-tagging`
- `x-amz-tagging-directive`
- `x-amz-meta-<metadata-name>`

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"CopyObject"

"CreateBucket"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todos parâmetros e cabeçalhos comuns para esta solicitação, além destes cabeçalhos adicionais:

- `x-amz-bucket-object-lock-enabled`

Corpo da solicitação

StorageGRID suporta todos os parâmetros do corpo da requisição definidos pela Amazon S3 REST API no momento da implementação.

Documentação do StorageGRID

"Operações em buckets"

"CreateMultipartUpload"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todos parâmetros e cabeçalhos comuns para esta solicitação, além destes cabeçalhos adicionais:

- `Cache-Control`
- `Content-Disposition`
- `Content-Encoding`
- `Content-Language`
- `Expires`
- `x-amz-checksum-algorithm`
- `x-amz-server-side-encryption`
- `x-amz-storage-class`
- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`
- `x-amz-tagging`
- `x-amz-object-lock-mode`
- `x-amz-object-lock-retain-until-date`
- `x-amz-object-lock-legal-hold`
- `x-amz-meta-<metadata-name>`

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"CreateMultipartUpload"

"DeleteBucket"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Documentação do StorageGRID

"Operações em buckets"

"DeleteBucketCors"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"DeleteBucketEncryption"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"DeleteBucketLifecycle"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

- "Operações em buckets"
- "Criar configuração de ciclo de vida do S3"

"DeleteBucketPolicy"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"DeleteBucketReplication"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"DeleteBucketTagging"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"DeleteObject"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todos os parâmetros e cabeçalhos comuns para esta solicitação, além deste cabeçalho de solicitação adicional:

- `x-amz-bypass-governance-retention`

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em objetos"

"DeleteObjects"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todos os parâmetros e cabeçalhos comuns para esta solicitação, além deste cabeçalho de solicitação adicional:

- `x-amz-bypass-governance-retention`

Corpo da solicitação

StorageGRID suporta todos os parâmetros do corpo da requisição definidos pela Amazon S3 REST API no momento da implementação.

Documentação do StorageGRID

"Operações em objetos"

"DeleteObjectTagging"

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em objetos"

"GetBucketAcl"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"GetBucketCors"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"GetBucketEncryption"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"GetBucketLifecycleConfiguration"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

- "Operações em buckets"
- "Criar configuração de ciclo de vida do S3"

"GetBucketLocation"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"GetBucketNotificationConfiguration"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"GetBucketPolicy"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"GetBucketReplication"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"GetBucketTagging"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"GetBucketVersioning"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"GetObject"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todos os parâmetros e cabeçalhos comuns para esta solicitação, além destes parâmetros de consulta URI adicionais:

- `x-amz-checksum-mode`
- `partNumber`
- `response-cache-control`
- `response-content-disposition`
- `response-content-encoding`
- `response-content-language`
- `response-content-type`
- `response-expires`

E estes cabeçalhos de solicitação adicionais:

- `Range`
- `x-amz-server-side-encryption-customer-algorithm`

- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`
- `If-Match`
- `If-Modified-Since`
- `If-None-Match`
- `If-Unmodified-Since`

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"GetObject"

"GetObjectAcl"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em objetos"

"GetObjectLegalHold"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Use a API REST S3 para configurar o S3 Object Lock"

"GetObjectLockConfiguration"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Use a API REST S3 para configurar o S3 Object Lock"

"GetObjectRetention"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Use a API REST S3 para configurar o S3 Object Lock"

"GetObjectTagging"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em objetos"

"HeadBucket"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"HeadObject"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todos parâmetros e cabeçalhos comuns para esta solicitação, além destes cabeçalhos adicionais:

- `x-amz-checksum-mode`
- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`
- `If-Match`
- `If-Modified-Since`
- `If-None-Match`
- `If-Unmodified-Since`

- `Range`

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"HeadObject"

"ListBuckets"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Nenhum

Documentação do StorageGRID

Operações no serviço › ListBuckets

"ListMultipartUploads"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID suporta todos parâmetros e cabeçalhos comuns para esta solicitação, além destes parâmetros adicionais:

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

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"ListMultipartUploads"

"ListObjects"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID suporta todos parâmetros e cabeçalhos comuns para esta solicitação, além destes parâmetros adicionais:

- `delimiter`
- `encoding-type`
- `marker`
- `max-keys`

- `prefix`

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"ListObjectsV2"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID suporta todos parâmetros e cabeçalhos comuns para esta solicitação, além destes parâmetros adicionais:

- `continuation-token`
- `delimiter`
- `encoding-type`
- `fetch-owner`
- `max-keys`
- `prefix`
- `start-after`

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"ListObjectVersions"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID suporta todos parâmetros e cabeçalhos comuns para esta solicitação, além destes parâmetros adicionais:

- `delimiter`
- `encoding-type`
- `key-marker`
- `max-keys`
- `prefix`
- `version-id-marker`

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"Operações em buckets"

"ListParts"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID suporta todos parâmetros e cabeçalhos comuns para esta solicitação, além destes parâmetros adicionais:

- `max-parts`
- `part-number-marker`
- `uploadId`

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"ListMultipartUploads"

"PutBucketCors"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

StorageGRID suporta todos os parâmetros do corpo da requisição definidos pela Amazon S3 REST API no momento da implementação.

Documentação do StorageGRID

"Operações em buckets"

"PutBucketEncryption"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Tags XML do corpo da requisição

StorageGRID suporta estas tags XML no corpo da requisição:

- `ApplyServerSideEncryptionByDefault`
- `Rule`
- `ServerSideEncryptionConfiguration`
- `SSEAlgorithm`

Documentação do StorageGRID

"Operações em buckets"

"PutBucketLifecycleConfiguration"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Tags XML do corpo da requisição

StorageGRID suporta estas tags XML no corpo da requisição:

- `And`
- `Days`
- `Expiration`
- `ExpiredObjectDeleteMarker`
- `Filter`
- `ID`
- `Key`
- `LifecycleConfiguration`
- `NewerNoncurrentVersions`
- `NoncurrentDays`
- `NoncurrentVersionExpiration`
- `Prefix`
- `Rule`
- `Status`
- `Tag`
- `Value`

Documentação do StorageGRID

- "Operações em buckets"
- "Criar configuração de ciclo de vida do S3"

"PutBucketNotificationConfiguration"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Tags XML do corpo da requisição

StorageGRID suporta estas tags XML no corpo da requisição:

- `Event`
- `Filter`
- `FilterRule`
- `Id`

- `Name`
- `NotificationConfiguration`
- `Prefix`
- `S3Key`
- `Suffix`
- `Topic`
- `TopicConfiguration`
- `Value`

Documentação do StorageGRID

"Operações em buckets"

"PutBucketPolicy"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Para obter detalhes sobre os campos de corpo JSON suportados, consulte "Use políticas de acesso a buckets e grupos".

"PutBucketReplication"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Tags XML do corpo da requisição

- `Bucket`
- `Destination`
- `Prefix`
- `ReplicationConfiguration`
- `Rule`
- `Status`
- `StorageClass`

Documentação do StorageGRID

"Operações em buckets"

"PutBucketTagging"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

StorageGRID suporta todos os parâmetros do corpo da requisição definidos pela Amazon S3 REST API no momento da implementação.

Documentação do StorageGRID

"Operações em buckets"

"PutBucketVersioning"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Parâmetros do corpo da requisição

StorageGRID suporta estes parâmetros no corpo da requisição:

- `VersioningConfiguration`
- `Status`

Documentação do StorageGRID

"Operações em buckets"

"PutObject"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todos parâmetros e cabeçalhos comuns para esta solicitação, além destes cabeçalhos adicionais:

- `Cache-Control`
- `Content-Disposition`
- `Content-Encoding`
- `Content-Language`
- `Expires`
- `x-amz-checksum-sha256`
- `x-amz-server-side-encryption`
- `x-amz-storage-class`
- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`
- `x-amz-tagging`
- `x-amz-object-lock-mode`
- `x-amz-object-lock-retain-until-date`
- `x-amz-object-lock-legal-hold`
- `x-amz-meta-<metadata-name>`

Corpo da solicitação

- Dados binários do objeto

Documentação do StorageGRID

"PutObject"

"PutObjectLegalHold"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

StorageGRID suporta todos os parâmetros do corpo da requisição definidos pela Amazon S3 REST API no momento da implementação.

Documentação do StorageGRID

"Use a API REST S3 para configurar o S3 Object Lock"

"PutObjectLockConfiguration"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

StorageGRID suporta todos os parâmetros do corpo da requisição definidos pela Amazon S3 REST API no momento da implementação.

Documentação do StorageGRID

"Use a API REST S3 para configurar o S3 Object Lock"

"PutObjectRetention"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todos os parâmetros e cabeçalhos comuns para esta solicitação, além deste cabeçalho adicional:

- `x-amz-bypass-governance-retention`

Corpo da solicitação

StorageGRID suporta todos os parâmetros do corpo da requisição definidos pela Amazon S3 REST API no momento da implementação.

Documentação do StorageGRID

"Use a API REST S3 para configurar o S3 Object Lock"

"PutObjectTagging"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

StorageGRID suporta todos os parâmetros do corpo da requisição definidos pela Amazon S3 REST API no momento da implementação.

Documentação do StorageGRID

"Operações em objetos"

"RestoreObject"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Para obter detalhes sobre os campos de corpo suportados, consulte "RestoreObject".

"SelectObjectContent"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todas parâmetros e cabeçalhos comuns para esta solicitação.

Corpo da solicitação

Para obter detalhes sobre os campos de corpo suportados, consulte o seguinte:

- "Use o S3 Select"
- "SelectObjectContent"

"UploadPart"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todos os parâmetros e cabeçalhos comuns para esta solicitação, além destes parâmetros de consulta URI adicionais:

- `partNumber`
- `uploadId`

E estes cabeçalhos de solicitação adicionais:

- `x-amz-checksum-sha256`
- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`

Corpo da solicitação

- Dados binários da parte

Documentação do StorageGRID

"UploadPart"

"UploadPartCopy"

Parâmetros de consulta URI e cabeçalhos de solicitação

StorageGRID oferece suporte a todos os parâmetros e cabeçalhos comuns para esta solicitação, além destes parâmetros de consulta URI adicionais:

- `partNumber`
- `uploadId`

E estes cabeçalhos de solicitação adicionais:

- `x-amz-copy-source`
- `x-amz-copy-source-if-match`
- `x-amz-copy-source-if-modified-since`
- `x-amz-copy-source-if-none-match`
- `x-amz-copy-source-if-unmodified-since`
- `x-amz-copy-source-range`
- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`
- `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`

Corpo da solicitação

Nenhum

Documentação do StorageGRID

"UploadPartCopy"

Teste a configuração da API REST S3 do StorageGRID

Você pode usar a Amazon Web Services Command Line Interface (AWS CLI) para testar sua conexão com o sistema e verificar se consegue ler e gravar objetos.

Antes de começar

- Você baixou e instalou a AWS CLI a partir de "aws.amazon.com/cli".
- Opcionalmente, você tem "criou um endpoint de balanceador de carga". Caso contrário, você sabe o endereço IP do Storage Node ao qual deseja se conectar e o número da porta a ser usado. Consulte "Endereços IP e portas para conexões de clientes".
- Você tem "Criou uma conta de locatário S3".
- Você fez login no tenant e "criou uma chave de acesso".

Para obter detalhes sobre essas etapas, consulte "Configurar conexões de cliente".

Passos

1. Configure as definições da AWS CLI para utilizar a conta que você criou no sistema StorageGRID:
   a. Entre no modo de configuração: `aws configure`
   b. Insira o ID da chave de acesso da conta que você criou.
   c. Digite a chave de acesso secreta da conta que você criou.
   d. Insira a região padrão a ser usada. Por exemplo, `us-east-1`.
   e. Digite o formato de saída padrão a ser usado ou pressione **Enter** para selecionar JSON.
2. Crie um bucket.

   Este exemplo pressupõe que você configurou um ponto de extremidade do balanceador de carga para usar o endereço IP 10.96.101.17 e a porta 10443.

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

   Se o bucket for criado com sucesso, a localização do bucket é retornada, como pode ser visto no exemplo a seguir:

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

3. Faça o upload de um 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
   ```

   Se o objeto for carregado com sucesso, um Etag é retornado, que é um hash dos dados do objeto.

4. Liste o conteúdo do bucket para verificar se o objeto foi carregado.

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

5. Exclua o objeto.

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

6. Exclua o bucket.

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

Como StorageGRID implementa a API REST S3

Como a StorageGRID API REST resolve solicitações conflitantes do cliente

Solicitações conflitantes de clientes, como dois clientes escrevendo na mesma chave, são resolvidas com base no princípio "o mais recente vence".

O momento da avaliação "latest-wins" é baseado em quando o sistema StorageGRID conclui uma determinada solicitação, e não em quando os clientes S3 iniciam uma operação.

Como StorageGRID API REST S3 equilibra disponibilidade e consistência

A consistência proporciona um equilíbrio entre a disponibilidade dos objetos e a consistência desses objetos em diferentes Storage Nodes e sites. Você pode alterar a consistência conforme necessário para sua aplicação.

Por padrão, StorageGRID garante a consistência de leitura após gravação para objetos recém-criados. Qualquer GET subsequente a um PUT concluído com sucesso poderá ler os dados recém-gravados. Sobrescritas de objetos existentes, atualizações de metadados e exclusões são eventualmente consistentes.

Se você deseja executar operações de objetos com um nível de consistência diferente, você pode:

- Especifique uma consistência para cada bucket.
- Especifique uma consistência para cada operação de API.
- Altere a consistência padrão de toda a grade executando uma das seguintes tarefas:
  - No Grid Manager, acesse **Configuração** > **Sistema** > **Configurações de armazenamento** > **Consistência padrão do bucket**.
  - .

> Uma alteração na consistência de toda a grade aplica-se apenas aos buckets criados após a alteração da configuração. Para determinar os detalhes de uma alteração, consulte o log de auditoria localizado em `/var/local/log` (pesquise por **consistencyLevel**).

Valores de consistência

A consistência afeta a forma como os metadados que StorageGRID usa para rastrear objetos são distribuídos entre os nós. A consistência afeta a disponibilidade de objetos para solicitações de clientes.

Você pode definir a consistência de um bucket ou de uma operação de API para um dos seguintes valores:

- **Todos**: Todos os nós recebem metadados de objeto imediatamente ou a solicitação falhará.
- **Strong-global**: Garante a consistência de leitura após gravação para todas as solicitações do cliente em

todos os sites. Quando a semântica de Quorum está configurada, os seguintes comportamentos se aplicam:

  - Permite tolerância a falha do local para solicitações de clientes quando as grades têm três ou mais sites. Grades com dois sites não terão tolerância a falha do local.
  - As seguintes operações do S3 não serão bem-sucedidas com um site inativo:
    - DeleteBucketEncryption
    - PutBucketBranch
    - PutBucketEncryption
    - PutBucketVersioning
    - PutObjectLegalHold
    - PutObjectLockConfiguration
    - PutObjectRetention

  Se necessário, você pode "configurar a semântica de quorum do StorageGRID para consistência global forte".

- **Strong-site**: os metadados do objeto são distribuídos imediatamente para outros nós no site. Garante a consistência de leitura após gravação para todas as solicitações do cliente dentro de um site.
- **Leitura após nova gravação**: Oferece consistência de leitura após gravação para novos objetos e consistência eventual para atualizações de objetos. Oferece alta disponibilidade e garantias de proteção de dados. Recomendado para a maioria dos casos.
- **Disponível**: Oferece consistência eventual tanto para novos objetos quanto para atualizações de objetos. Para buckets do S3, use somente quando necessário (por exemplo, para um bucket que contém valores de log que são lidos raramente ou para operações HEAD ou GET em chaves que não existem). Não é compatível com buckets do S3 FabricPool.

Use as consistências "Read-after-new-write" e "Available"

Quando uma operação HEAD ou GET utiliza a consistência "Leitura após nova gravação", StorageGRID realiza a pesquisa em várias etapas, da seguinte forma:

- Primeiro, ele busca o objeto usando uma consistência baixa.
- Caso essa busca falhe, ela se repete no próximo valor de consistência até atingir uma consistência equivalente ao comportamento de strong-global.

Se uma operação HEAD ou GET usar a consistência "Read-after-new-write", mas o objeto não existir, a busca pelo objeto sempre atingirá uma consistência equivalente ao comportamento para strong-global. Como essa consistência exige que várias cópias dos metadados do objeto estejam disponíveis em cada site, você pode receber um grande número de erros 500 Internal Server se dois ou mais Storage Nodes no mesmo site estiverem indisponíveis.

A menos que você precise de garantias de consistência semelhantes às da Amazon S3, você pode evitar esses erros para operações HEAD e GET definindo a consistência como "Disponível". Quando uma operação HEAD ou GET usa a consistência "Disponível", StorageGRID fornece apenas consistência eventual. Não tenta novamente uma operação com falha em níveis de consistência crescentes, portanto, não exige que várias cópias dos metadados do objeto estejam disponíveis.

Especifique a consistência para a operação da API

Para definir a consistência de uma operação de API específica, os valores de consistência devem ser suportados pela operação e você deve especificar a consistência no cabeçalho da solicitação. Este exemplo define a consistência como "Strong-site" para uma operação GetObject.

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

> Você deve usar a mesma consistência para as operações PutObject e GetObject.

Especifique a consistência para o bucket

Para definir a consistência do bucket, você pode usar a solicitação "Consistência de PUT Bucket" do StorageGRID. Ou você pode "alterar a consistência de um bucket" a partir do Tenant Manager.

Ao definir a consistência de um bucket, esteja ciente do seguinte:

- A configuração de consistência para um bucket determina qual consistência é usada para as operações S3 realizadas nos objetos no bucket ou na configuração do bucket. Ela não afeta as operações no próprio bucket.
- A consistência de uma operação individual da API tem precedência sobre a consistência do bucket.
- Em geral, os buckets devem usar a consistência padrão, "Leitura após nova gravação". Se as solicitações não estiverem funcionando corretamente, altere o comportamento do cliente do aplicativo, se possível. Ou configure o cliente para especificar a consistência para cada solicitação de API. Defina a consistência no nível do bucket somente como último recurso.

Como as regras de consistência e ILM interagem para afetar a proteção de dados

Tanto a sua escolha de consistência quanto a sua regra ILM afetam a forma como os objetos são protegidos. Essas configurações podem interagir.

Por exemplo, a consistência usada quando um objeto é armazenado afeta o posicionamento inicial dos metadados do objeto, enquanto o comportamento de ingestão selecionado para a regra de gerenciamento de ciclo de vida (ILM) afeta o posicionamento inicial das cópias do objeto. Como StorageGRID requer acesso tanto aos metadados quanto aos dados de um objeto para atender às solicitações do cliente, selecionar níveis de proteção correspondentes para a consistência e o comportamento de ingestão pode proporcionar melhor proteção inicial dos dados e respostas mais previsíveis do sistema.

As seguintes "opções de ingestão" estão disponíveis para as regras do ILM:

Commit duplo
: StorageGRID cria imediatamente cópias provisórias do objeto e retorna sucesso ao cliente. Cópias especificadas na regra ILM são feitas quando possível.

Estrito
: Todas as cópias especificadas na regra ILM devem ser feitas antes que a mensagem de sucesso seja retornada ao cliente.

Equilibrado

StorageGRID tenta criar todas as cópias especificadas na regra ILM durante a ingestão; se isso não for possível, cópias provisórias são criadas e uma mensagem de sucesso é retornada ao cliente. As cópias especificadas na regra ILM são criadas sempre que possível.

Exemplo de como a consistência e a regra de ILM podem interagir

Suponha que você tenha uma grade de três sites com a seguinte regra ILM e a seguinte consistência:

- **Regra ILM**: Crie três cópias do objeto, uma no local e uma em cada local remoto. Use o comportamento de ingestão estrito.
- **Consistência**: Global forte (os metadados do objeto são distribuídos imediatamente para vários sites).

Quando um cliente armazena um objeto na grid, StorageGRID cria as três cópias do objeto e distribui os metadados para vários sites antes de retornar sucesso ao cliente.

O objeto fica totalmente protegido contra perda no momento em que a mensagem de ingestão bem-sucedida é recebida. Por exemplo, se o local remoto for perdido logo após a ingestão, cópias dos dados do objeto e dos metadados do objeto ainda existirão nos locais remotos. O objeto poderá ser recuperado integralmente dos outros locais.

Se, em vez disso, você usasse a mesma regra ILM e a consistência forte do site, o cliente poderia receber uma mensagem de sucesso após os dados do objeto serem replicados para os locais remotos, mas antes que os metadados do objeto fossem distribuídos lá. Nesse caso, o nível de proteção dos metadados do objeto não corresponde ao nível de proteção dos dados do objeto. Se o local remoto for perdido logo após a ingestão, os metadados do objeto serão perdidos. O objeto não pode ser recuperado.

A inter-relação entre as regras de consistência e as regras de ILM pode ser complexa. Entre em contato com a NetApp se você precisar de assistência.

Como StorageGRID usa o versionamento de objetos

Você pode definir o estado de versionamento de um bucket se desejar manter várias versões de cada objeto. Habilitar o versionamento para um bucket pode ajudar a proteger contra a exclusão acidental de objetos e permite que você recupere e restaure versões anteriores de um objeto.

O sistema StorageGRID implementa o versionamento com suporte para a maioria dos recursos, com algumas limitações. StorageGRID suporta até 10.000 versões de cada objeto.

O versionamento de objetos pode ser combinado com o gerenciamento do ciclo de vida das informações (ILM) do StorageGRID ou com a configuração do ciclo de vida do bucket S3. Você deve habilitar explicitamente o versionamento para cada bucket. Quando o versionamento está habilitado para um bucket, cada objeto adicionado ao bucket recebe um ID de versão, que é gerado pelo sistema StorageGRID.

O uso de MFA (autenticação multifator) com a função Delete não é suportado.

> O versionamento só pode ser ativado em buckets criados com StorageGRID versão 10.3 ou posterior.

ILM e versionamento

As políticas de ILM são aplicadas a cada versão de um objeto. Um processo de varredura de ILM examina continuamente todos os objetos e os reavalia em relação à política de ILM atual. Quaisquer alterações que você fizer nas políticas de ILM são aplicadas a todos os objetos previamente ingeridos. Isso inclui versões previamente ingeridas se o versionamento estiver habilitado. A varredura de ILM aplica as novas alterações de ILM aos objetos previamente ingeridos.

Para objetos S3 em buckets com versionamento habilitado, o suporte a versionamento permite que você crie regras ILM que usam "Tempo não atual" como tempo de referência (selecione **Sim** para a pergunta "Aplicar esta regra somente a versões antigas do objeto?" em "Etapa 1 do assistente Criar uma regra ILM"). Quando um objeto é atualizado, suas versões anteriores se tornam não atuais. Usar um filtro de "Tempo não atual" permite que você crie políticas que reduzem o impacto de armazenamento das versões anteriores dos objetos.

> Ao carregar uma nova versão de um objeto usando uma operação de upload multipart, a hora não atual da versão original do objeto reflete o momento em que o upload multipart foi criado para a nova versão, não quando o upload multipart foi concluído. Em casos limitados, a hora não atual da versão original pode ser horas ou dias anterior à hora da versão atual.

Informações relacionadas

- "Como os objetos versionados do S3 são excluídos"
- "Regras e políticas do ILM para objetos versionados do S3 (Exemplo 4)".

Use a API REST S3 para configurar o S3 Object Lock do StorageGRID

Se a configuração global de Bloqueio de Objetos S3 estiver habilitada para o seu sistema StorageGRID, você pode criar buckets com o Bloqueio de Objetos S3 ativado. Você pode especificar o período de retenção padrão para cada bucket ou as configurações de retenção para cada versão de objeto.

Como habilitar o S3 Object Lock para um bucket

Se a configuração global de S3 Object Lock estiver ativada para o seu sistema StorageGRID, você pode, opcionalmente, ativar o S3 Object Lock ao criar cada bucket.

O Bloqueio de Objetos do S3 é uma configuração permanente que só pode ser ativada ao criar um bucket. Você não pode adicionar ou desativar o Bloqueio de Objetos do S3 depois que um bucket é criado.

Para ativar o S3 Object Lock para um bucket, use um destes métodos:

- Crie o bucket usando o Gerenciador de Locatários. Consulte "Criar bucket S3".
- Crie o bucket usando uma solicitação CreateBucket com o cabeçalho de solicitação `x-amz-bucket-object-lock-enabled`. Veja "Operações em buckets".

O S3 Object Lock requer o versionamento do bucket, que é ativado automaticamente quando o bucket é criado. Você não pode suspender o versionamento do bucket. Consulte "Versionamento de objetos".

Configurações de retenção padrão para um bucket

Quando o S3 Object Lock está habilitado para um bucket, você pode opcionalmente habilitar a retenção padrão para o bucket e especificar um modo de retenção padrão e um período de retenção padrão.

Modo de retenção padrão

- No modo COMPLIANCE:
  - O objeto não pode ser excluído até que sua retain-until-date seja atingida.
  - A data de retenção do objeto pode ser aumentada, mas não pode ser diminuída.
  - A data de retenção do objeto não pode ser removida até que essa data seja atingida.
- No modo GOVERNANÇA:
  - Usuários com a `s3:BypassGovernanceRetention` permissão podem usar o `x-amz-bypass-governance-retention: true` cabeçalho da solicitação para ignorar as configurações de retenção.
  - Esses usuários podem excluir uma versão de objeto antes que a data de retenção seja atingida.
  - Esses usuários podem aumentar, diminuir ou remover a retain-until-date de um objeto.

Período de retenção padrão

Cada bucket pode ter um período de retenção padrão especificado em anos ou dias.

Como definir a retenção padrão para um bucket

Para definir a retenção padrão de um bucket, use um destes métodos:

- Gerencie as configurações do bucket a partir do Tenant Manager. Consulte "Crie um bucket S3" e "Atualizar retenção padrão do S3 Object Lock".
- Envie uma solicitação PutObjectLockConfiguration para o bucket para especificar o modo padrão e o número padrão de dias ou anos.

PutObjectLockConfiguration

A solicitação PutObjectLockConfiguration permite que você defina e modifique o modo de retenção padrão e o período de retenção padrão para um bucket com o S3 Object Lock ativado. Você também pode remover as configurações de retenção padrão previamente definidas.

Quando novas versões de objetos são ingeridas no bucket, o modo de retenção padrão é aplicado se `x-amz-object-lock-mode` e `x-amz-object-lock-retain-until-date` não forem especificados. O período de retenção padrão é usado para calcular a data de retenção até se `x-amz-object-lock-retain-until-date` não for especificado.

Se o período de retenção padrão for modificado após a ingestão de uma versão de objeto, a data de retenção da versão do objeto permanecerá a mesma e não será recalculada usando o novo período de retenção padrão.

Você precisa ter a `s3:PutBucketObjectLockConfiguration` permissão, ou ser o root da conta, para concluir esta operação.

O `Content-MD5`cabeçalho da requisição deve ser especificado na solicitação PUT.

Exemplo de solicitação

Este exemplo habilita o S3 Object Lock para um bucket e define o modo de retenção padrão como COMPLIANCE e o período de retenção padrão como 6 anos.

```
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>
```

Como determinar a retenção padrão para um bucket

Para determinar se o S3 Object Lock está ativado para um bucket e para ver o modo de retenção padrão e o período de retenção, use um destes métodos:

- Veja o bucket no Tenant Manager. Consulte "Visualizar buckets do S3".
- Emita uma solicitação GetObjectLockConfiguration.

GetObjectLockConfiguration

A solicitação GetObjectLockConfiguration permite que você determine se o S3 Object Lock está habilitado para um bucket e, se estiver habilitado, veja se há um modo de retenção padrão e um período de retenção configurados para o bucket.

Quando novas versões de objetos são ingeridas no bucket, o modo de retenção padrão é aplicado se `x-amz-object-lock-mode` não for especificado. O período de retenção padrão é usado para calcular a retain-until-date se `x-amz-object-lock-retain-until-date` não for especificado.

Você precisa ter a `s3:GetBucketObjectLockConfiguration` permissão, ou ser account root, para concluir esta operação.

Exemplo de solicitação

```
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
```

Exemplo de resposta

```
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>
```

Como especificar as configurações de retenção para um objeto

Um bucket com o S3 Object Lock ativado pode conter uma combinação de objetos com e sem configurações de retenção do S3 Object Lock.

As configurações de retenção em nível de objeto são especificadas usando a API REST do S3. As configurações de retenção para um objeto substituem quaisquer configurações de retenção padrão para o bucket.

Você pode especificar as seguintes configurações para cada objeto:

- **Modo de retenção**: COMPLIANCE ou GOVERNANÇA.
- **Retain-until-date**: Uma data que especifica por quanto tempo a versão do objeto deve ser retida pelo StorageGRID.

    - No modo COMPLIANCE, se a retain-until-date for futura, o objeto poderá ser recuperado, mas não poderá ser modificado ou excluído. A retain-until-date pode ser aumentada, mas essa data não pode ser diminuída ou removida.
    - No modo GOVERNANÇA, usuários com permissões especiais podem ignorar a configuração de retain-until-date. Eles podem excluir uma versão de objeto antes que seu período de retenção tenha expirado. Eles também podem aumentar, diminuir ou até mesmo remover o retain-until-date.
- **Guarda legal**: Aplicar uma guarda legal a uma versão de objeto bloqueia imediatamente esse objeto. Por exemplo, você pode precisar aplicar uma guarda legal a um objeto relacionado a uma investigação ou disputa legal. Uma guarda legal não tem data de expiração, mas permanece em vigor até ser explicitamente removida.

  A configuração de guarda legal para um objeto é independente do modo de retenção e da data limite de retenção. Se uma versão de um objeto estiver sob guarda legal, ninguém pode excluir essa versão.

Para especificar as configurações de S3 Object Lock ao adicionar uma versão de objeto a um bucket, emita uma solicitação "PutObject", "CopyObject" ou "CreateMultipartUpload".

Você pode usar o seguinte:

- `x-amz-object-lock-mode`, que pode ser COMPLIANCE ou GOVERNANCE (diferencia maiúsculas de minúsculas).

  > Se você especificar `x-amz-object-lock-mode`, você também deve especificar `x-amz-object-lock-retain-until-date`.

- `x-amz-object-lock-retain-until-date`
    - O valor de retain-until-date deve estar no formato `2020-08-10T21:46:00Z`. Frações de segundo são permitidas, mas apenas 3 casas decimais são preservadas (precisão em milissegundos). Outros formatos ISO 8601 não são permitidos.
    - A data de retenção deve estar no futuro.
- `x-amz-object-lock-legal-hold`

  Se guarda legal estiver ON (diferencia maiúsculas de minúsculas), o objeto é colocado sob guarda legal. Se guarda legal estiver OFF, nenhuma guarda legal é aplicada. Qualquer outro valor resulta em um erro 400 Bad Request (InvalidArgument).

Se você usar algum desses cabeçalhos de solicitação, esteja ciente destas restrições:

- O `Content-MD5`cabeçalho da requisição é obrigatório se qualquer `x-amz-object-lock-*`cabeçalho de requisição estiver presente na solicitação PutObject. `Content-MD5`Não é obrigatório para CopyObject ou CreateMultipartUpload.
- Se o bucket não tiver o S3 Object Lock ativado e um `x-amz-object-lock-*` cabeçalho de solicitação estiver presente, será retornado um erro 400 Bad Request (InvalidRequest).
- A solicitação PutObject dá suporte ao uso de `x-amz-storage-class: REDUCED_REDUNDANCY` para corresponder ao comportamento da AWS. No entanto, quando um objeto é ingerido em um bucket com S3 Object Lock ativado, StorageGRID sempre realizará uma ingestão com confirmação dupla.
- Uma resposta GET ou de versão HeadObject subsequente incluirá os cabeçalhos `x-amz-object-lock-mode`, `x-amz-object-lock-retain-until-date` e `x-amz-object-lock-legal-hold`, se configurados e se o remetente da solicitação tiver as permissões corretas de `s3:Get*`.

Você pode usar a `s3:object-lock-remaining-retention-days`chave de condição de política para limitar os períodos de retenção mínimos e máximos permitidos para seus objetos.

Como atualizar as configurações de retenção de um objeto

Se você precisar atualizar as configurações de guarda legal ou de retenção para uma versão de objeto existente, pode executar as seguintes operações de sub-recurso de objeto:

- `PutObjectLegalHold`

  Se o novo valor de guarda legal estiver ATIVADO, o objeto fica sujeito a guarda legal. Se o valor de guarda legal estiver DESATIVADO, a guarda legal é suspensa.

- `PutObjectRetention`
  - O valor do modo pode ser COMPLIANCE ou GOVERNANCE (diferencia maiúsculas de minúsculas).
  - O valor de retain-until-date deve estar no formato `2020-08-10T21:46:00Z`. Frações de segundo são permitidas, mas apenas 3 casas decimais são preservadas (precisão em milissegundos). Outros formatos ISO 8601 não são permitidos.
  - Se uma versão de um objeto já possui uma data de retenção definida, você só pode aumentá-la. O novo valor deve estar no futuro.

Como usar o modo GOVERNANCE

Usuários que têm a permissão `s3:BypassGovernanceRetention` podem ignorar as configurações de retenção ativa de um objeto que utiliza o modo GOVERNANCE. Quaisquer operações DELETE ou PutObjectRetention devem incluir o cabeçalho `x-amz-bypass-governance-retention:true` da solicitação. Esses usuários podem executar as seguintes operações adicionais:

- Execute operações DeleteObject ou DeleteObjects para excluir uma versão de objeto antes que seu período de retenção tenha expirado.

  Objetos que estão sob guarda legal não podem ser excluídos. A guarda legal deve estar DESATIVADA.

- Execute operações PutObjectRetention que alteram o modo de uma versão de objeto de GOVERNANCE para COMPLIANCE antes que o período de retenção tenha expirado.

  A mudança do modo de COMPLIANCE para GOVERNANCE nunca é permitida.

- Realize operações PutObjectRetention para aumentar, diminuir ou remover o período de retenção de uma versão do objeto.

Informações relacionadas

- "Gerencie objetos com S3 Object Lock"
- "Use o S3 Object Lock para reter objetos"
- "Guia do usuário do Amazon Simple Storage Service: bloqueando objetos"

Saiba mais sobre a configuração do ciclo de vida do S3 para StorageGRID

Você pode criar uma configuração de ciclo de vida do S3 para controlar quando objetos específicos são excluídos do sistema StorageGRID.

O exemplo simples nesta seção ilustra como uma configuração de ciclo de vida do S3 pode controlar quando determinados objetos são excluídos (expirados) de buckets específicos do S3. O exemplo nesta seção é apenas para fins ilustrativos. Para obter detalhes completos sobre a criação de configurações de ciclo de vida do S3, consulte "Guia do usuário do Amazon Simple Storage Service: gerenciamento de ciclo de vida de objetos". Observe que StorageGRID oferece suporte apenas a ações de Expiração; não oferece suporte a ações de Transição.

O que é configuração de ciclo de vida

Uma configuração de ciclo de vida é um conjunto de regras aplicadas aos objetos em buckets específicos do S3. Cada regra especifica quais objetos são afetados e quando esses objetos expirarão (em uma data específica ou após um determinado número de dias).

StorageGRID suporta até 1.000 regras de ciclo de vida em uma configuração de ciclo de vida. Cada regra pode incluir os seguintes elementos XML:

- Expiração: exclua um objeto quando uma data específica for atingida ou quando um número específico de dias for alcançado, a partir do momento em que o objeto foi ingerido.
- NoncurrentVersionExpiration: Exclua um objeto quando um número específico de dias for atingido, a partir do momento em que o objeto deixou de ser atual.
- Filtro (Prefixo, Tag)
- Status
- ID

Cada objeto segue as configurações de retenção de um ciclo de vida do bucket S3 ou de uma política de ILM. Quando um ciclo de vida do bucket S3 é configurado, as ações de expiração do ciclo de vida substituem a política de ILM para objetos que correspondem ao filtro do ciclo de vida do bucket. Objetos que não correspondem ao filtro do ciclo de vida do bucket usam as configurações de retenção da política de ILM. Se um objeto corresponder a um filtro do ciclo de vida do bucket e nenhuma ação de expiração for explicitamente especificada, as configurações de retenção da política de ILM não são usadas e entende-se que as versões do objeto são retidas para sempre. Consulte "Exemplos de prioridades para ciclo de vida de bucket S3 e política ILM".

Como resultado, um objeto pode ser removido da grid mesmo que as instruções de posicionamento em uma regra ILM ainda se apliquem ao objeto. Ou, um objeto pode ser mantido na grid mesmo depois que quaisquer instruções de posicionamento ILM para o objeto tiverem expirado. Para obter detalhes, consulte "Como o ILM opera ao longo do ciclo de vida de um objeto".

> A configuração de ciclo de vida do bucket pode ser usada com buckets que têm o S3 Object Lock ativado, mas a configuração de ciclo de vida do bucket não é compatível com buckets legados Compliant.

StorageGRID suporta o uso das seguintes operações de bucket para gerenciar configurações de ciclo de vida:

- DeleteBucketLifecycle
- GetBucketLifecycleConfiguration
- PutBucketLifecycleConfiguration

Criar configuração de ciclo de vida

Como primeiro passo para criar uma configuração de ciclo de vida, você cria um arquivo JSON que inclui uma ou mais regras. Por exemplo, este arquivo JSON inclui três regras, como segue:

1. A regra 1 aplica-se apenas a objetos que correspondam ao prefixo `category1/` e que tenham um valor de `key2 tag2`. O parâmetro `Expiration` especifica que os objetos que correspondam ao filtro expirarão à meia-noite de 22 de agosto de 2020.
2. A regra 2 aplica-se apenas a objetos que correspondam ao prefixo `category2/`. O `Expiration` parâmetro especifica que objetos que correspondem ao filtro expirarão 100 dias após serem ingeridos.

> As regras que especificam um número de dias são relativas à data de ingestão do objeto. Se a data atual for posterior à data de ingestão mais o número de dias, alguns objetos podem ser removidos do bucket assim que a configuração do ciclo de vida for aplicada.

3. A regra 3 aplica-se apenas a objetos que correspondam ao prefixo `category3/`. O `Expiration`parâmetro especifica que quaisquer versões não atuais de objetos correspondentes expirarão 50 dias após se tornarem não atuais.

```
{
    "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 configuração de ciclo de vida ao bucket

Após criar o arquivo de configuração de ciclo de vida, você o aplica a um bucket emitindo uma solicitação PutBucketLifecycleConfiguration.

Esta solicitação aplica a configuração de ciclo de vida no arquivo de exemplo aos objetos em um bucket chamado `testbucket`.

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

Para validar se uma configuração de ciclo de vida foi aplicada com sucesso ao bucket, emita uma solicitação GetBucketLifecycleConfiguration. Por exemplo:

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

Uma resposta bem-sucedida lista a configuração de ciclo de vida que você acabou de aplicar.

Valide se a expiração do ciclo de vida do bucket se aplica ao objeto

Você pode determinar se uma regra de expiração na configuração de ciclo de vida se aplica a um objeto específico ao emitir uma solicitação PutObject, HeadObject ou GetObject. Se uma regra se aplicar, a resposta incluirá um parâmetro `Expiration` que indica quando o objeto expira e qual regra de expiração foi correspondida.

> Como o ciclo de vida do bucket substitui o ILM, a `expiry-date` data exibida é a data real em que o objeto será excluído. Para obter detalhes, consulte "Como a retenção de objeto é determinada".

Por exemplo, esta solicitação PutObject foi emitida em 22 de junho de 2020 e coloca um objeto no bucket `testbucket`.

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

A resposta de sucesso indica que o objeto expirará em 100 dias (01 out 2020) e que corresponde à Regra 2 da configuração do ciclo de vida.

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

Por exemplo, esta solicitação HeadObject foi usada para obter metadados para o mesmo objeto no bucket testbucket.

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

A resposta de sucesso inclui os metadados do objeto e indica que o objeto expirará em 100 dias e que corresponde à Regra 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": {}
}
```

Para buckets com controle de versão ativado, o `x-amz-expiration` cabeçalho de resposta se aplica somente às versões atuais dos objetos.

Recomendações para implementar a API REST S3 com StorageGRID

Você deve seguir estas recomendações ao implementar a API REST para uso com StorageGRID.

Recomendações para HEADs para objetos inexistentes

Se sua aplicação verifica rotineiramente se um objeto existe em um caminho onde você não espera que ele exista, você deve usar a consistência "Disponível" "consistência". Por exemplo, você deve usar a consistência "Disponível" se sua aplicação executar um HEAD em um local antes de executar um PUT nele.

Caso contrário, se a operação HEAD não encontrar o objeto, você poderá receber um grande número de erros 500 Internal Server se dois ou mais nós de armazenamento no mesmo site estiverem indisponíveis ou se um local remoto estiver inacessível.

Você pode definir a consistência "Disponível" para cada bucket usando a "Consistência de PUT Bucket"

solicitação, ou pode especificar a consistência no cabeçalho da solicitação para uma operação de API individual.

Recomendações para chaves de objeto

Siga estas recomendações para nomes de chaves de objetos, com base em quando o bucket foi criado pela primeira vez.

Buckets criados no StorageGRID 11.4 ou anterior

- Não utilize valores aleatórios como os quatro primeiros caracteres das chaves de objetos. Isso contrasta com a recomendação anterior da AWS para prefixos de chave. Em vez disso, utilize prefixos não aleatórios e não exclusivos, como `image`.
- Se você seguir a recomendação anterior da AWS de usar caracteres aleatórios e exclusivos nos prefixos das chaves, prefixe as chaves dos objetos com um nome de diretório. Ou seja, use este formato:

  `mybucket/mydir/f8e3-image3132.jpg`

  Em vez deste formato:

  `mybucket/f8e3-image3132.jpg`

Buckets criados no StorageGRID 11.4 ou posterior

Não é necessário restringir os nomes das chaves de objeto para atender às melhores práticas de desempenho. Na maioria dos casos, você pode usar valores aleatórios para os quatro primeiros caracteres dos nomes das chaves de objeto.

> Uma exceção a isso é uma carga de trabalho do S3 que remove continuamente todos os objetos após um curto período de tempo. Para minimizar o impacto no desempenho nesse caso de uso, varie a parte inicial do nome da chave a cada alguns milhares de objetos com algo como a data. Por exemplo, suponha que um cliente S3 normalmente grave 2.000 objetos por segundo e que a política de ciclo de vida do ILM ou do bucket remova todos os objetos após três dias. Para minimizar o impacto no desempenho, você pode nomear as chaves usando um padrão como este: `/mybucket/mydir/yyyymmddhhmmss-random_UUID.jpg`

Recomendações para "leituras de intervalo"

Se a "opção global para compactar objetos armazenados" estiver habilitada, os aplicativos cliente do S3 devem evitar realizar operações GetObject que especifiquem um intervalo de bytes a ser retornado. Essas operações de "leitura de intervalo" são ineficientes porque o StorageGRID precisa efetivamente descompactar os objetos para acessar os bytes solicitados. Operações GetObject que solicitam um pequeno intervalo de bytes de um objeto muito grande são especialmente ineficientes; por exemplo, é ineficiente ler um intervalo de 10 MB de um objeto compactado de 50 GB.

Se intervalos forem lidos de objetos compactados, as solicitações do cliente podem expirar.

> Se você precisar compactar objetos e seu aplicativo cliente precisar usar leituras de intervalo, aumente o tempo limite de leitura do aplicativo.

Suporte para Amazon S3 API REST

Operações e limitações da API REST S3 no StorageGRID

O sistema StorageGRID implementa a API Simple Storage Service (API Version 2006-03-01) com suporte para a maioria das operações, e com algumas limitações. Você precisa entender os detalhes da implementação ao integrar aplicativos cliente da API REST do S3.

O sistema StorageGRID suporta tanto solicitações virtuais no estilo de hospedagem quanto solicitações no estilo de caminho.

Tratamento de datas

A implementação do StorageGRID da API REST S3 suporta apenas formatos de data HTTP válidos.

O sistema StorageGRID suporta apenas formatos de data HTTP válidos para quaisquer cabeçalhos que aceitam valores de data. A parte da hora da data pode ser especificada no formato de Greenwich Mean Time (GMT) ou no formato de Universal Coordinated Time (UTC) sem deslocamento de fuso horário (+0000 deve ser especificado). Se você incluir o `x-amz-date` cabeçalho na sua solicitação, ele substituirá qualquer valor especificado no cabeçalho Date da solicitação. Ao usar AWS Signature Version 4, o `x-amz-date` cabeçalho deve estar presente na solicitação assinada porque o cabeçalho Date não é compatível.

Cabeçalhos de solicitação comuns

O sistema StorageGRID suporta os cabeçalhos de solicitação comuns definidos por "Referência da API do Amazon Simple Storage Service: cabeçalhos de solicitação comuns", com uma exceção.

Cabeçalho da solicitação	Implementação
Autorização	Suporte completo para AWS Signature Version 2 Suporte para AWS Signature Version 4, com as seguintes exceções: • Quando você fornece o valor real do checksum da carga em `x-amz-content-sha256`, o valor é aceito sem validação, como se o valor `UNSIGNED-PAYLOAD` tivesse sido fornecido para o cabeçalho. Quando você fornece um `x-amz-content-sha256` valor de cabeçalho que implica `aws-chunked` streaming (por exemplo, STREAMING-AWS4-HMAC-SHA256-PAYLOAD), as assinaturas dos fragmentos não são verificadas em relação aos dados dos fragmentos.

Cabeçalhos de resposta comuns

O sistema StorageGRID suporta todos os cabeçalhos de resposta comuns definidos pela *Referência da API do Simple Storage Service*, com uma exceção.

Cabeçalho de resposta	Implementação
x-amz-id-2	Não utilizado

Como StorageGRID S3 API REST autentica solicitações

O sistema StorageGRID suporta tanto o acesso autenticado quanto o acesso anônimo a objetos usando a API S3.

A API S3 suporta Signature version 2 e Signature version 4 para autenticar solicitações da API S3.

As solicitações autenticadas devem ser assinadas usando seu access key ID e secret access key.

O sistema StorageGRID suporta dois métodos de autenticação: o cabeçalho HTTP `Authorization` e o uso de parâmetros de consulta.

Use o cabeçalho Authorization HTTP

O cabeçalho `Authorization`HTTP é usado por todas as operações da API S3, exceto para solicitações de acesso anônimo quando permitido pela política do bucket. O cabeçalho `Authorization`contém todas as informações de assinatura necessárias para autenticar uma solicitação.

Use parâmetros de consulta

Você pode usar parâmetros de consulta para adicionar informações de autenticação a uma URL. Isso é conhecido como pré-assinar a URL, o que pode ser usado para conceder acesso temporário a recursos específicos. Usuários com a URL pré-assinada não precisam saber a chave de acesso secreta para acessar o recurso, o que permite que você forneça acesso restrito de terceiro a um recurso.

Operações de serviço suportadas no StorageGRID

O sistema StorageGRID suporta as seguintes operações no serviço.

Operação	Implementação
ListBuckets (anteriormente chamado GET Service)	Implementado com todo o comportamento da API REST do Amazon S3. Sujeito a alterações sem aviso prévio.
GET Uso de Armazenamento	A solicitação "GET Uso de Armazenamento" StorageGRID informa a quantidade total de armazenamento em uso por uma conta e para cada bucket associado à conta. Esta é uma operação no serviço com o caminho / e um parâmetro de consulta personalizado (`?x-ntap-sg-usage` adicionado.
OPÇÕES /	Os aplicativos cliente podem enviar `OPTIONS /`solicitações para a porta S3 em um nó de armazenamento, sem fornecer credenciais de autenticação S3, para determinar se o nó de armazenamento está disponível. Você pode usar essa solicitação para monitoramento ou para permitir que balanceadores de carga externos identifiquem quando um nó de armazenamento está inativo.

Operações e detalhes de implementação do bucket S3 no StorageGRID

O sistema StorageGRID suporta um máximo de 5.000 buckets por conta de locatário do S3.

Cada grid pode ter no máximo 100.000 buckets.

Para suportar 5.000 buckets, cada Storage Node na grid deve ter no mínimo 64 GB de RAM.

As restrições de nome de bucket seguem as restrições da região AWS US Standard, mas você deve restringi-las ainda mais às convenções de nomenclatura DNS para dar suporte a solicitações no estilo de hospedagem virtual do S3.

Veja a seguir para mais informações:

- "Guia do usuário do Amazon Simple Storage Service: cotas, restrições e limitações de buckets"
- "Configurar nomes de domínio de endpoint S3"

As operações ListObjects (GET Bucket) e ListObjectVersions (GET Bucket object versions) são compatíveis com StorageGRID "valores de consistência".

Você pode verificar se as atualizações do último tempo de acesso estão habilitadas ou desabilitadas para buckets individuais. Veja "Obter o último tempo de acesso do bucket".

A tabela a seguir descreve como o StorageGRID implementa as operações de bucket da API REST do S3. Para executar qualquer uma dessas operações, as credenciais de acesso necessárias devem ser fornecidas para a conta.

Operação	Implementação
CreateBucket	Cria um novo bucket. Ao criar o bucket, você se torna o proprietário dele. • Os nomes dos buckets devem obedecer às seguintes regras: ◦ Deve ser único em cada sistema StorageGRID (e não apenas único dentro da conta do locatário). ◦ Deve ser compatível com DNS. ◦ Deve conter no mínimo 3 e no máximo 63 caracteres. ◦ Pode ser uma série de um ou mais rótulos, com rótulos adjacentes separados por um ponto. Cada rótulo deve começar e terminar com uma letra minúscula ou um número e só pode usar letras minúsculas, números e hífens. ◦ Não deve ter a aparência de um endereço IP formatado como texto. ◦ Não se deve usar pontos em solicitações no estilo de hospedagem virtual. Os pontos causarão problemas com a verificação do certificado curinga do servidor. • Por padrão, os buckets são criados na `us-east-1` região; no entanto, você pode usar o `LocationConstraint` elemento no corpo da requisição para especificar uma região diferente. Ao usar o `LocationConstraint` elemento, você deve especificar o nome exato de uma região que tenha sido definida usando o Grid Manager ou a Grid Management API. Entre em contato com o administrador do sistema se não souber o nome da região que deve usar. **Nota**: ocorrerá um erro se a sua solicitação CreateBucket usar uma região que não foi definida no StorageGRID. • Você pode incluir o `x-amz-bucket-object-lock-enabled` cabeçalho da solicitação para criar um bucket com o S3 Object Lock ativado. Consulte "Use a API REST S3 para configurar o S3 Object Lock". Você deve habilitar o S3 Object Lock ao criar o bucket. Você não pode adicionar ou desabilitar o S3 Object Lock depois que um bucket é criado. O S3 Object Lock requer o versionamento de buckets, que é habilitado automaticamente ao criar o bucket.
DeleteBucket	Exclui o bucket.
DeleteBucketCors	Remove a configuração CORS do bucket.
DeleteBucketEncryption	Remove a criptografia padrão do bucket. Os objetos criptografados existentes permanecem criptografados, mas quaisquer novos objetos adicionados ao bucket não são criptografados.
DeleteBucketLifecycle	Exclui a configuração de ciclo de vida do bucket. Consulte "Criar configuração de ciclo de vida do S3".

Operação	Implementação
DeleteBucketPolicy	Exclui a política associada ao bucket.
DeleteBucketReplication	Exclui a configuração de replicação associada ao bucket.
DeleteBucketTagging	Utiliza o `tagging` subrecurso para remover todas as tags de um bucket. **Atenção**: Se uma tag de política ILM não padrão estiver definida para este bucket, haverá uma `NTAP-SG-ILM-BUCKET-TAG` tag de bucket com um valor atribuído a ela. Não envie uma solicitação DeleteBucketTagging se houver uma `NTAP-SG-ILM-BUCKET-TAG` tag de bucket. Em vez disso, envie uma solicitação PutBucketTagging contendo apenas a `NTAP-SG-ILM-BUCKET-TAG` tag e seu valor atribuído para remover todas as outras tags do bucket. Não modifique nem remova a `NTAP-SG-ILM-BUCKET-TAG` tag de bucket.
GetBucketAcl	Retorna uma resposta positiva e o ID, DisplayName e a permissão do proprietário do bucket, indicando que o proprietário tem acesso total ao bucket.
GetBucketCors	Retorna a `cors`configuração do bucket.
GetBucketEncryption	Retorna a configuração de criptografia padrão para o bucket.
GetBucketLifecycleConfiguration (anteriormente chamado GET Bucket lifecycle)	Retorna a configuração do ciclo de vida do bucket. Consulte "Criar configuração de ciclo de vida do S3".
GetBucketLocation	Retorna a região que foi definida usando o elemento `LocationConstraint` na solicitação CreateBucket. Se a região do bucket for `us-east-1`, uma string vazia será retornada para a região.
GetBucketNotificationConfiguration (anteriormente chamado de GET Bucket notification)	Retorna a configuração de notificação associada ao bucket.
GetBucketPolicy	Retorna a política associada ao bucket.
GetBucketReplication	Retorna a configuração de replicação associada ao bucket.
GetBucketTagging	Utiliza o `tagging` subrecurso para retornar todas as tags de um bucket. **Atenção**: Se uma tag de política ILM não padrão estiver definida para este bucket, haverá uma tag de bucket `NTAP-SG-ILM-BUCKET-TAG` com um valor atribuído a ela. Não modifique nem remova esta tag.

Operação	Implementação
GetBucketVersioning	Esta implementação utiliza o `versioning` subrecurso para retornar o estado de versionamento de um bucket. • *blank*: O versionamento nunca foi ativado (o bucket está "Não versionado") • Ativado: o versionamento está ativado • Suspenso: o versionamento estava ativado anteriormente e está suspenso
GetObjectLockConfiguration	Retorna o modo de retenção padrão do bucket e o período de retenção padrão, se configurados. Consulte "Use a API REST S3 para configurar o S3 Object Lock".
HeadBucket	Determina se um bucket existe e se você tem permissão para acessá-lo. Esta operação retorna: • `x-ntap-sg-bucket-id`: o UUID do bucket no formato UUID. • `x-ntap-sg-trace-id`: o ID de rastreamento exclusivo da solicitação associada.
ListObjects e ListObjectsV2 (anteriormente chamado GET Bucket)	Retorna alguns ou todos (até 1.000) os objetos em um bucket. A Storage Class para objetos pode ter um dos dois valores, mesmo que o objeto tenha sido ingerido com a `REDUCED_REDUNDANCY` storage class option: • `STANDARD`, o que indica que o objeto está armazenado em um pool de storage composto por Storage Nodes. • `GLACIER`, o que indica que o objeto foi movido para o bucket externo especificado pelo Cloud Storage Pool. Se o bucket contiver um grande número de chaves excluídas com o mesmo prefixo, a resposta pode incluir algumas `CommonPrefixes` que não contêm chaves. Para as solicitações HeadObject e ListObject, o StorageGRID retorna os timestamps LastModified com precisões diferentes, enquanto a AWS retorna os timestamps com a mesma precisão, como mostrado nos exemplos a seguir: • StorageGRID HeadObject: ""LastModified": "2024-09-26T16:43:24+00:00" • StorageGRID ListObject: "LastModified": "2024-09-26T16:43:24.931000+00:00" • AWS HeadObject: "LastModified": "2023-10-17T00:19:54+00:00" • AWS ListObject: "LastModified": "2023-10-17T00:19:54+00:00"

Operação	Implementação
ListObjectVersions (anteriormente chamado de GET Bucket Object versions)	Com acesso de LEITURA em um bucket, usar essa operação com o `versions` subrecurso lista os metadados de todas as versões dos objetos no bucket.
PutBucketCors	Define a configuração CORS para um bucket, permitindo que ele atenda solicitações de origem cruzada. O compartilhamento de recursos de origem cruzada (CORS) é um mecanismo de segurança que permite que aplicativos web cliente em um domínio acessem recursos em um domínio diferente. Por exemplo, suponha que você use um bucket S3 chamado `images` para armazenar imagens. Ao definir a configuração CORS para o bucket `images`, você pode permitir que as imagens nesse bucket sejam exibidas no site `http://www.example.com`.
PutBucketEncryption	Define o estado de criptografia padrão de um bucket existente. Quando a criptografia no nível do bucket está habilitada, todos os novos objetos adicionados ao bucket são criptografados. O StorageGRID oferece suporte à criptografia do lado do servidor com chaves gerenciadas pelo StorageGRID. Ao especificar a regra de configuração de criptografia do lado do servidor, defina o parâmetro `SSEAlgorithm` como `AES256` e não use o parâmetro `KMSMasterKeyID`. A configuração de criptografia padrão do bucket é ignorada se a solicitação de upload do objeto já especificar criptografia (ou seja, se a solicitação incluir o `x-amz-server-side-encryption-*` cabeçalho de solicitação).

Operação	Implementação
PutBucketLifecycleConfiguration (anteriormente denominado PUT Bucket lifecycle)	Cria uma nova configuração de ciclo de vida para o bucket ou substitui uma configuração de ciclo de vida existente. O StorageGRID suporta até 1.000 regras de ciclo de vida em uma configuração de ciclo de vida. Cada regra pode incluir os seguintes elementos XML: • Validade (dias, data, ExpiredObjectDeleteMarker) • NoncurrentVersionExpiration (NewerNoncurrentVersions, NoncurrentDays) • Filtro (Prefixo, Tag) • Status • ID StorageGRID não suporta estas ações: • AbortIncompleteMultipartUpload • Transição Consulte "Criar configuração de ciclo de vida do S3". Para entender como a ação de expiração no ciclo de vida do bucket interage com as instruções de posicionamento do ILM, consulte "Como o ILM opera ao longo do ciclo de vida de um objeto". **Nota**: a configuração do ciclo de vida do bucket pode ser usada com buckets que têm o S3 Object Lock ativado, mas a configuração do ciclo de vida do bucket não é compatível com buckets legados Compliant.

Operação	Implementação

PutBucketNotificationConfiguration
 	
(anteriormente chamada de PUT Bucket notification)
	Configura as notificações para o bucket usando o XML de configuração de notificações incluído no corpo da solicitação. Você deve estar ciente dos seguintes detalhes de implementação:
	
	• StorageGRID suporta tópicos do Amazon Simple Notification Service (Amazon SNS), tópicos do Kafka ou endpoints de webhook como destinos. Endpoints do Simple Queue Service (SQS) ou do AWS Lambda não são suportados.
	• O destino das notificações deve ser especificado como o URN de um endpoint do StorageGRID. Os endpoints podem ser criados usando o Tenant Manager ou a Tenant Management API.
	
	O endpoint deve existir para que a configuração de notificação seja bem-sucedida. Se o endpoint não existir, um erro `400 Bad Request` é retornado com o código `InvalidArgument`.
	
	• Não é possível configurar uma notificação para os seguintes tipos de evento. Esses tipos de evento **não** são suportados.
	◦ `s3:ReducedRedundancyLostObject`
	◦ `s3:ObjectRestore:Completed`
	• As notificações de eventos enviadas pelo StorageGRID usam o formato JSON padrão, exceto que não incluem algumas chaves e usam valores específicos para outras, conforme mostrado na lista a seguir:
	◦ **eventSource**
	
	`sgws:s3`
	
	◦ **awsRegion**
	
	não incluído
	
	◦ **x-amz-id-2**
	
	não incluído
	
	◦ **arn**
	
	`urn:sgws:s3:::bucket_name`

PutBucketPolicy	Define a política associada ao bucket. Consulte "Use políticas de acesso a buckets e grupos".

Operação	Implementação
PutBucketReplication	Configura "StorageGRID CloudMirror replicação" para o bucket usando o XML de configuração de replicação fornecido no corpo da solicitação. Para a replicação CloudMirror, você deve estar ciente dos seguintes detalhes de implementação: • StorageGRID suporta apenas a V1 da configuração de replicação. Isso significa que StorageGRID não suporta o uso do `Filter` elemento para regras e segue as convenções da V1 para exclusão de versões de objetos. Para obter detalhes, consulte "Guia do usuário do Amazon Simple Storage Service: configuração de replicação". • A replicação de buckets pode ser configurada em buckets versionados ou não versionados. • Você pode especificar um bucket de destino diferente em cada regra do XML de configuração de replicação. Um bucket de origem pode replicar para mais de um bucket de destino. • Os buckets de destino devem ser especificados como o URN dos endpoints do StorageGRID, conforme especificado no Tenant Manager ou na API de Gerenciamento de Tenant. Consulte "Configurar a replicação CloudMirror". O endpoint deve existir para que a configuração de replicação seja bem-sucedida. Se o endpoint não existir, a solicitação falhará como um `400 Bad Request`. A mensagem de erro indica: `Unable to save the replication policy. The specified endpoint URN does not exist: URN`. • Você não precisa especificar um `Role` no XML de configuração. Esse valor não é usado pelo StorageGRID e será ignorado se for enviado. • Se você omitir a classe de armazenamento do XML de configuração, StorageGRID usará a classe de armazenamento `STANDARD` por padrão. • Se você excluir um objeto do bucket de origem ou excluir o próprio bucket de origem, o comportamento da replicação entre regiões será o seguinte: ◦ Se você excluir o objeto ou bucket antes que ele seja replicado, o objeto/bucket não será replicado e você não será notificado. ◦ Se você excluir o objeto ou bucket após ele ter sido replicado, o StorageGRID segue o comportamento padrão de exclusão do Amazon S3 para a V1 da replicação entre regiões.

Operação	Implementação
PutBucketTagging	Utiliza o `tagging` sub-recurso para adicionar ou atualizar um conjunto de tags para um bucket. Ao adicionar tags ao bucket, esteja ciente das seguintes limitações: • Tanto StorageGRID quanto Amazon S3 suportam até 50 tags por bucket. • As tags associadas a um bucket devem ter chaves de tag exclusivas. Uma chave de tag pode ter até 128 caracteres Unicode. • Os valores das tags podem ter até 256 caracteres Unicode. • As chaves e os valores diferenciam maiúsculas de minúsculas. **Atenção**: Se uma tag de política ILM não padrão for definida para este bucket, haverá uma `NTAP-SG-ILM-BUCKET-TAG` tag de bucket com um valor atribuído a ela. Certifique-se de que a `NTAP-SG-ILM-BUCKET-TAG` tag de bucket esteja incluída com o valor atribuído em todas as solicitações PutBucketTagging. Não modifique nem remova esta tag. **Nota**: Esta operação irá sobrescrever quaisquer tags existentes no bucket. Se alguma tag existente for omitida do conjunto, essa tag será removida do bucket.
PutBucketVersioning	Utiliza o `versioning` subrecurso para definir o estado de versionamento de um bucket existente. Você pode definir o estado de versionamento com um dos seguintes valores: • Ativado: habilita o versionamento para os objetos no bucket. Todos os objetos adicionados ao bucket recebem um ID de versão exclusivo. • Suspenso: desativa o versionamento para os objetos no bucket. Todos os objetos adicionados ao bucket recebem o ID da versão `null`.
PutObjectLockConfiguration	Configura ou remove o modo de retenção padrão do bucket e o período de retenção padrão. Se o período de retenção padrão for modificado, a data de retenção das versões existentes do objeto permanece a mesma e não é recalculada usando o novo período de retenção padrão. Veja "Use a API REST S3 para configurar o S3 Object Lock" para obter informações detalhadas.

Operações em objetos

Como o StorageGRID implementa operações da API REST do S3 para objetos

Esta seção descreve como o sistema StorageGRID implementa as operações da API REST do S3 para objetos.

As seguintes condições aplicam-se a todas as operações de objeto:

- StorageGRID "valores de consistência" é compatível com todas as operações em objetos, com exceção

das seguintes:

- GetObjectAcl
- `OPTIONS /`
- PutObjectLegalHold
- PutObjectRetention
- SelectObjectContent

- Solicitações conflitantes de clientes, como dois clientes gravando na mesma chave, são resolvidas com base no princípio "latest-wins". O momento da avaliação do "latest-wins" é determinado quando o sistema StorageGRID conclui uma determinada solicitação, e não quando os clientes S3 iniciam uma operação.
- Todos os objetos em um bucket do StorageGRID pertencem ao proprietário do bucket, incluindo objetos criados por um usuário anônimo ou por outra conta.

A tabela a seguir descreve como o StorageGRID implementa as operações de objetos da API REST do S3.

Operação	Implementação
DeleteObject	Ao processar uma solicitação DeleteObject, StorageGRID tenta remover imediatamente todas as cópias do objeto de todos os locais de armazenamento. Se for bem-sucedido, StorageGRID retorna uma resposta ao cliente imediatamente. Se todas as cópias não puderem ser removidas em 30 segundos (por exemplo, porque um local está temporariamente indisponível), StorageGRID coloca as cópias em fila para remoção e então indica sucesso ao cliente. **Restrições** • A autenticação multifator (MFA) e o cabeçalho de resposta `x-amz-mfa` não são suportados. • Os `If-None` e `If-None-Match` cabeçalhos são aceitos, mas não funcionais. **Controle de versões** Para remover uma versão específica, o solicitante deve ser o proprietário do bucket e usar o `versionId` sub-recurso. Usar esse sub-recurso exclui permanentemente a versão. Se o `versionId` corresponder a um marcador de exclusão, o cabeçalho de resposta `x-amz-delete-marker` é retornado definido como `true`. • Se um objeto for excluído sem o `versionId` sub-recurso em um bucket com versionamento ativado, isso resulta na geração de um marcador de exclusão. O `versionId` marcador de exclusão é retornado usando o `x-amz-version-id` cabeçalho de resposta, e o `x-amz-delete-marker` cabeçalho de resposta é retornado definido como `true`. • Se um objeto for excluído sem o `versionId` sub-recurso em um bucket com versionamento suspenso, isso resulta na exclusão permanente de uma versão "null" já existente ou de um marcador de exclusão "null", e na geração de um novo marcador de exclusão "null". O cabeçalho de resposta `x-amz-delete-marker` é retornado definido como `true`. **Nota**: Em certos casos, vários marcadores de exclusão podem existir para um objeto. Veja "Use a API REST S3 para configurar o S3 Object Lock" para saber como excluir versões de objetos no modo GOVERNANÇA.
DeleteObjects (anteriormente chamado de DELETE Multiple Objects)	A autenticação multifator (MFA) e o cabeçalho de resposta `x-amz-mfa` não são suportados. Vários objetos podem ser excluídos na mesma mensagem de solicitação. Veja "Use a API REST S3 para configurar o S3 Object Lock" para saber como excluir versões de objetos no modo GOVERNANÇA.

Operação	Implementação
DeleteObjectTagging	Utiliza o `tagging` subrecurso para remover todas as tags de um objeto. **Controle de versões** Se o `versionId`parâmetro de consulta não for especificado na solicitação, a operação exclui todas as tags da versão mais recente do objeto em um bucket versionado. Se a versão atual do objeto for um marcador de exclusão, um status " MethodNotAllowed " será retornado com o cabeçalho de resposta `x-amz-delete-marker` definido como `true`.
GetObject	"GetObject"
GetObjectAcl	Se as credenciais de acesso necessárias forem fornecidas para a conta, a operação retorna uma resposta positiva e o ID, DisplayName e a Permissão do proprietário do objeto, indicando que o proprietário tem acesso total ao objeto.
GetObjectLegalHold	"Use a API REST S3 para configurar o S3 Object Lock"
GetObjectRetention	"Use a API REST S3 para configurar o S3 Object Lock"
GetObjectTagging	Utiliza o `tagging` subrecurso para retornar todas as tags de um objeto. **Controle de versões** Se o `versionId` parâmetro de consulta não for especificado na solicitação, a operação retorna todas as tags da versão mais recente do objeto em um bucket versionado. Se a versão atual do objeto for um marcador de exclusão, um status " MethodNotAllowed " será retornado com o cabeçalho de resposta `x-amz-delete-marker` definido como `true`.
HeadObject	"HeadObject"
RestoreObject	"RestoreObject"
PutObject	"PutObject"
CopyObject (anteriormente denominado PUT Object - Copy)	"CopyObject"
PutObjectLegalHold	"Use a API REST S3 para configurar o S3 Object Lock"
PutObjectRetention	"Use a API REST S3 para configurar o S3 Object Lock"

Operação	Implementação
PutObjectTagging	Utiliza o `tagging` subrecurso para adicionar um conjunto de tags a um objeto existente. **Limites de tags de objeto** Você pode adicionar tags a novos objetos ao carregá-los ou pode adicioná-las a objetos existentes. Tanto StorageGRID quanto Amazon S3 suportam até 10 tags por objeto. As tags associadas a um objeto devem ter chaves de tag exclusivas. Uma chave de tag pode ter até 128 caracteres Unicode e os valores de tag podem ter até 256 caracteres Unicode. Chaves e valores diferenciam maiúsculas de minúsculas. **Atualizações de tags e comportamento de ingestão** Quando você usa PutObjectTagging para atualizar as tags de um objeto, o StorageGRID não o ingere novamente. Isso significa que a opção de Comportamento de Ingestão especificada na regra ILM correspondente não é usada. Quaisquer alterações no posicionamento do objeto acionadas pela atualização são feitas quando o ILM é reavaliado pelos processos normais de ILM em segundo plano. Isso significa que, se a regra ILM usar a opção "Estrita" para o comportamento de ingestão, nenhuma ação será tomada se os posicionamentos de objeto necessários não puderem ser feitos (por exemplo, porque um local recém-necessário não está disponível). O objeto atualizado mantém seu posicionamento atual até que o posicionamento necessário seja possível. **Resolução de conflitos** Solicitações conflitantes de clientes, como dois clientes gravando na mesma chave, são resolvidas com base no princípio "latest-wins". O momento da avaliação do "latest-wins" é determinado quando o sistema StorageGRID conclui uma determinada solicitação, e não quando os clientes S3 iniciam uma operação. **Controle de versões** Se o `versionId`parâmetro de consulta não for especificado na solicitação, a operação adiciona tags à versão mais recente do objeto em um bucket versionado. Se a versão atual do objeto for um marcador de exclusão, um status ""MethodNotAllowed"" é retornado com o cabeçalho de resposta `x-amz-delete-marker`definido como `true`.
SelectObjectContent	"SelectObjectContent"

Operações Amazon S3 Select suportadas no StorageGRID

StorageGRID oferece suporte às seguintes cláusulas SELECT do Amazon S3, tipos de

dados e operadores para o "Comando SelectObjectContent".

Quaisquer itens não listados não são suportados.

Para sintaxe, consulte "SelectObjectContent". Para mais informações sobre S3 Select, consulte o "Documentação da AWS para S3 Select".

Somente contas de locatário com o S3 Select ativado podem emitir consultas SelectObjectContent. Consulte o "Considerações e requisitos para usar o S3 Select".

Cláusulas

- Lista de seleção
- Cláusula FROM
- Cláusula WHERE
- Cláusula LIMIT

Tipos de dados

- bool
- inteiro
- string
- float
- decimal, numérico
- carimbo de data/hora

Operadores

Operadores lógicos

- E
- NÃO
- OU

Operadores de comparação

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

- EM

Operadores de correspondência de padrões

- CURTIR
- _
- %

Operadores unitários

- É NULO
- NÃO É NULO

Operadores matemáticos

- +
- -
- *
- /
- %

StorageGRID segue a precedência do operador Amazon S3 Select.

Funções agregadas

- AVG()
- COUNT(*)
- MAX()
- MIN()
- SOMA()

Funções condicionais

- CASO
- COALESCE
- NULLIF

Funções de conversão

- CAST (para tipo de dados suportado)

Funções de data

- DATE_ADD
- DATE_DIFF
- EXTRAIR
- TO_STRING

- TO_TIMESTAMP
- UTCNOW

Funções de string

- CHAR_LENGTH, CHARACTER_LENGTH
- MAIS BAIXO
- SUBSTRING
- TRIM
- SUPERIOR

Como StorageGRID usa criptografia do lado do servidor

A criptografia do lado do servidor permite que você proteja seus dados de objeto em repouso. StorageGRID criptografa os dados ao gravar o objeto e os descriptografa quando você acessa o objeto.

Se você deseja usar criptografia do lado do servidor, pode escolher uma das duas opções mutuamente exclusivas, com base em como as chaves de criptografia são gerenciadas:

- **SSE (criptografia do lado do servidor com chaves gerenciadas pelo StorageGRID)**: Quando você envia uma solicitação S3 para armazenar um objeto, o StorageGRID criptografa o objeto com uma chave exclusiva. Quando você envia uma solicitação S3 para recuperar o objeto, o StorageGRID usa a chave armazenada para descriptografar o objeto.
- **SSE-C (server-side encryption with customer-provided keys)**: Ao enviar uma solicitação ao S3 para armazenar um objeto, você fornece sua própria chave de criptografia. Ao recuperar um objeto, você fornece a mesma chave de criptografia como parte da sua solicitação. Se as duas chaves de criptografia coincidirem, o objeto é descriptografado e seus dados do objeto são retornados.

  Embora o StorageGRID gerencie todas as operações de criptografia e descriptografia de objetos, você deve gerenciar as chaves de criptografia que fornecer.

  As chaves de criptografia que você fornece nunca são armazenadas. Se você perder uma chave de criptografia, perderá o objeto correspondente.

  Se um objeto for criptografado com SSE ou SSE-C, quaisquer configurações de criptografia em nível de bucket ou de grid serão ignoradas.

Use SSE

Para criptografar um objeto com uma chave exclusiva gerenciada pelo StorageGRID, você usa o seguinte cabeçalho de solicitação:

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

O cabeçalho de solicitação SSE é compatível com as seguintes operações de objeto:

- "PutObject"
- "CopyObject"

- "CreateMultipartUpload"

Use SSE-C

Para criptografar um objeto com uma chave exclusiva que você gerencia, você usa três cabeçalhos de solicitação:

Cabeçalho da solicitação	Descrição
`x-amz-server-side-encryption-customer-algorithm`	Especifique o algoritmo de criptografia. O valor do cabeçalho deve ser `AES256`.
`x-amz-server-side-encryption-customer-key`	Especifique a chave de criptografia que será usada para criptografar ou descriptografar o objeto. O valor da chave deve ser de 256 bits, codificado em base64.
`x-amz-server-side-encryption-customer-key-MD5`	Especifique o resumo MD5 da chave de criptografia de acordo com a RFC 1321, que é usado para garantir que a chave de criptografia foi transmitida sem erro. O valor do resumo MD5 deve ser codificado em base64 com 128 bits.

Os cabeçalhos de solicitação SSE-C são suportados pelas seguintes operações de objeto:

- "GetObject"
- "HeadObject"
- "PutObject"
- "CopyObject"
- "CreateMultipartUpload"
- "UploadPart"
- "UploadPartCopy"

Considerações sobre o uso de criptografia do lado do servidor com chaves fornecidas pelo cliente (SSE-C)

Antes de usar o SSE-C, esteja ciente das seguintes considerações:

- Você deve usar https.

> ⚠ StorageGRID rejeita quaisquer solicitações feitas via http ao usar SSE-C. Por questões de segurança, você deve considerar qualquer chave enviada acidentalmente via http como comprometida. Descarte a chave e faça a rotação conforme apropriado.

- O ETag na resposta não é o MD5 dos dados do objeto.
- Você deve gerenciar o mapeamento das chaves de criptografia para os objetos. StorageGRID não armazena chaves de criptografia. Você é responsável por controlar a chave de criptografia que fornecer para cada objeto.
- Se o seu bucket tiver o versionamento ativado, cada versão do objeto deverá ter sua própria chave de criptografia. Você é responsável por controlar a chave de criptografia usada para cada versão do objeto.

- Como você gerencia as chaves de criptografia no lado do cliente, também deve gerenciar quaisquer medidas de segurança adicionais, como a rotação de chaves, no lado do cliente.

> As chaves de criptografia que você fornece nunca são armazenadas. Se você perder uma chave de criptografia, perderá o objeto correspondente.

- Se a replicação entre grades ou a replicação CloudMirror estiver configurada para o bucket, você não poderá ingerir objetos SSE-C. A operação de ingestão falhará.

Informações relacionadas

"Guia do usuário do Amazon S3: usando criptografia do lado do servidor com chaves fornecidas pelo cliente (SSE-C)"

Crie uma cópia de um objeto no StorageGRID com a solicitação S3 CopyObject

Você pode usar a solicitação S3 CopyObject para criar uma cópia de um objeto que já está armazenado no S3. Uma operação CopyObject é a mesma que executar GetObject seguido de PutObject.

Resolver conflitos

Solicitações conflitantes de clientes, como dois clientes gravando na mesma chave, são resolvidas com base no princípio "latest-wins". O momento da avaliação do "latest-wins" é determinado quando o sistema StorageGRID conclui uma determinada solicitação, e não quando os clientes S3 iniciam uma operação.

Tamanho do objeto

O tamanho máximo *recomendado* para uma única operação PutObject é de 5 GiB (5.368.709.120 bytes). Se você tiver objetos maiores que 5 GiB, use "upload multipart" em vez disso.

O tamanho máximo *suportado* para uma única operação PutObject é de 5 TiB (5.497.558.138.880 bytes).

> Se você atualizou do StorageGRID 11.6 ou anterior, o alerta "Tamanho do objeto S3 PUT muito grande" será acionado se você tentar fazer o upload de um objeto que exceda 5 GiB. Se você tiver uma nova instalação do StorageGRID 11.7 ou 11.8, o alerta não será acionado nesse caso. No entanto, para alinhar com o padrão AWS S3, as versões futuras do StorageGRID não suportarão uploads de objetos maiores que 5 GiB.

Caracteres UTF-8 nos metadados do usuário

Se uma solicitação incluir valores UTF-8 (sem escape) no nome da chave ou no valor dos metadados definidos pelo usuário, o comportamento do StorageGRID será indefinido.

StorageGRID não analisa nem interpreta caracteres UTF-8 escapados incluídos no nome da chave ou no valor dos metadados definidos pelo usuário. Caracteres UTF-8 escapados são tratados como caracteres ASCII:

- As solicitações são bem-sucedidas se os metadados definidos pelo usuário incluírem caracteres UTF-8 escapados.
- StorageGRID não retorna o `x-amz-missing-meta` cabeçalho se o valor interpretado do nome ou valor da chave incluir caracteres não imprimíveis.

Cabeçalhos de solicitação suportados

Os seguintes cabeçalhos de solicitação são suportados:

- `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 por um par nome-valor contendo metadados definidos pelo usuário
- `x-amz-metadata-directive`: O valor padrão é `COPY`, o que permite que você copie o objeto e os metadados associados.

  Você pode especificar `REPLACE` para sobrescrever os metadados existentes ao copiar o objeto ou para atualizar os metadados do objeto.

- `x-amz-storage-class`
- `x-amz-tagging-directive`: O valor padrão é `COPY`, que permite copiar o objeto e todas as tags.

  Você pode especificar `REPLACE`para sobrescrever as tags existentes ao copiar o objeto ou para atualizar as tags.

- Cabeçalhos da solicitação S3 Object Lock:
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    Se uma solicitação for feita sem esses cabeçalhos, as configurações de retenção padrão do bucket serão usadas para calcular o modo de versão do objeto e a data de retenção. Consulte "Use a API REST S3 para configurar o S3 Object Lock".

- Cabeçalhos de solicitação 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 Cabeçalhos de solicitação para criptografia do lado do servidor

Cabeçalhos de solicitação não suportados

Os seguintes cabeçalhos de solicitação não são suportados:

- `Cache-Control`
- `Content-Disposition`
- `Content-Encoding`
- `Content-Language`
- `Expires`
- `If-Match`

  O `If-Match header` é aceito, mas não funcional.

- `If-None-Match`

  O `If-None-Match header` é aceito, mas não funcional.

- `x-amz-checksum-algorithm`

  Ao copiar um objeto, se o objeto de origem tiver um checksum, StorageGRID não copia esse valor de checksum para o novo objeto. Esse comportamento se aplica independentemente de você tentar usar `x-amz-checksum-algorithm` na solicitação do objeto.

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

Opções de classe de armazenamento

O `x-amz-storage-class`cabeçalho da solicitação é compatível e afeta quantas cópias de objetos o StorageGRID cria se a regra ILM correspondente usar o Dual commit ou Balanced "opção de ingestão".

- `STANDARD`

  (Padrão) Especifica uma operação de ingestão de confirmação dupla quando a regra ILM usa a opção Dual commit ou quando a opção Balanced recorre à criação de cópias intermediárias.

- `REDUCED_REDUNDANCY`

  Especifica uma operação de ingestão de confirmação única quando a regra ILM usa a opção de confirmação dupla ou quando a opção balanceada recorre à criação de cópias intermediárias.

  > ⓘ Se você estiver ingerindo um objeto em um bucket com S3 Object Lock ativado, a `REDUCED_REDUNDANCY` opção será ignorada. Se você estiver ingerindo um objeto em um bucket legado Compliant, a `REDUCED_REDUNDANCY` opção retornará um erro. StorageGRID sempre realizará uma ingestão com confirmação dupla para garantir que os requisitos de conformidade sejam atendidos.

Usando x-amz-copy-source em CopyObject

Se o bucket e a chave de origem, especificados no `x-amz-copy-source` cabeçalho, forem diferentes do bucket de destino e da chave de destino, uma cópia dos dados do objeto de origem é gravada no destino.

Se a origem e o destino coincidirem e o `x-amz-metadata-directive` cabeçalho for especificado como `REPLACE`, os metadados do objeto serão atualizados com os valores de metadados fornecidos na solicitação. Nesse caso, StorageGRID não reingere o objeto. Isso tem duas consequências importantes:

- Não é possível usar CopyObject para criptografar um objeto existente no local ou para alterar a criptografia de um objeto existente no local. Se você fornecer o `x-amz-server-side-encryption` cabeçalho ou o `x-amz-server-side-encryption-customer-algorithm` cabeçalho, StorageGRID rejeita a solicitação e retorna `XNotImplemented`.
- A opção de Comportamento de Ingestão especificada na regra ILM correspondente não é utilizada. Quaisquer alterações no posicionamento do objeto que sejam acionadas pela atualização são feitas quando o ILM é reavaliado pelos processos ILM normais em segundo plano.

  Isso significa que, se a regra ILM usar a opção "Estrita" para o comportamento de ingestão, nenhuma ação será tomada se os posicionamentos de objeto necessários não puderem ser feitos (por exemplo, porque um local recém-necessário não está disponível). O objeto atualizado mantém seu posicionamento atual até que o posicionamento necessário seja possível.

Cabeçalhos de solicitação para criptografia do lado do servidor

Se você "usar criptografia do lado do servidor" fizer isso, os cabeçalhos da solicitação que você fornecer dependerão de o objeto de origem estar criptografado e de você planejar criptografar o objeto de destino.

- Se o objeto de origem estiver criptografado usando uma chave fornecida pelo cliente (SSE-C), você deverá incluir os três cabeçalhos a seguir na solicitação CopyObject, para que o objeto possa ser descriptografado e copiado:
  - `x-amz-copy-source-server-side-encryption-customer-algorithm`: especifique `AES256`.
  - `x-amz-copy-source-server-side-encryption-customer-key`: Especifique a chave de criptografia que você forneceu ao criar o objeto de origem.
  - `x-amz-copy-source-server-side-encryption-customer-key-MD5`: Especifique o resumo MD5 que você forneceu ao criar o objeto de origem.
- Se você deseja criptografar o objeto de destino (a cópia) com uma chave exclusiva que você fornece e gerencia, inclua os três cabeçalhos a seguir:
  - `x-amz-server-side-encryption-customer-algorithm`: Especifique `AES256`.
  - `x-amz-server-side-encryption-customer-key`: Especifique uma nova chave de criptografia para o objeto de destino.
  - `x-amz-server-side-encryption-customer-key-MD5`: Especifique o hash MD5 da nova chave de criptografia.

  > As chaves de criptografia fornecidas por você nunca são armazenadas. Se você perder uma chave de criptografia, perderá o objeto correspondente. Antes de usar chaves fornecidas pelo cliente para proteger dados de objetos, revise as considerações para "usando criptografia do lado do servidor".

- Se você deseja criptografar o objeto de destino (a cópia) com uma chave exclusiva gerenciada pelo StorageGRID (SSE), inclua este cabeçalho na solicitação CopyObject:
  - `x-amz-server-side-encryption`

O `server-side-encryption` valor do objeto não pode ser atualizado. Em vez disso, faça uma cópia com um novo `server-side-encryption` valor usando `x-amz-metadata-directive`: `REPLACE`.

Controle de versões

Se o bucket de origem for versionado, você pode usar o `x-amz-copy-source` cabeçalho para copiar a versão mais recente de um objeto. Para copiar uma versão específica de um objeto, você deve especificar explicitamente a versão a ser copiada usando o `versionId` sub-recurso. Se o bucket de destino for versionado, a versão gerada será retornada no `x-amz-version-id` cabeçalho da resposta. Se o versionamento estiver suspenso para o bucket de destino, então `x-amz-version-id` retorna o valor "null".

Recuperar um objeto de um bucket no StorageGRID com a solicitação GetObject

Você pode usar a solicitação S3 GetObject para recuperar um objeto de um bucket S3.

GetObject e objetos multipartes

Você pode usar o `partNumber`parâmetro de solicitação para recuperar uma parte específica de um objeto multipart ou segmentado. O `x-amz-mp-parts-count`elemento de resposta indica quantas partes o objeto possui.

Você pode definir `partNumber` como 1 tanto para objetos segmentados/multipartes quanto para objetos não segmentados/não multipartes; no entanto, o elemento de resposta `x-amz-mp-parts-count` só é retornado para objetos segmentados ou multipartes.

Caracteres UTF-8 nos metadados do usuário

StorageGRID não analisa nem interpreta caracteres UTF-8 escapados em metadados definidos pelo usuário. Solicitações GET para um objeto com caracteres UTF-8 escapados em metadados definidos pelo usuário não retornam o `x-amz-missing-meta` cabeçalho se o nome da chave ou o valor incluir caracteres não imprimíveis.

Cabeçalho de solicitação compatível

O seguinte cabeçalho de solicitação é compatível:

- `x-amz-checksum-mode`: especificar `ENABLED`

  O `Range`cabeçalho não é compatível com `x-amz-checksum-mode` para GetObject. Quando você inclui `Range` na solicitação com `x-amz-checksum-mode` ativado, StorageGRID não retorna um valor de checksum na resposta.

Cabeçalho de solicitação não suportado

O seguinte cabeçalho de solicitação não é compatível e retorna `XNotImplemented`:

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

Controle de versões

Se um `versionId` subrecurso não for especificado, a operação busca a versão mais recente do objeto em

um bucket versionado. Se a versão atual do objeto for um marcador de exclusão, o status "Não encontrado" é retornado com o `x-amz-delete-marker` cabeçalho de resposta definido como `true`.

Cabeçalhos de solicitação para criptografia do lado do servidor com chaves de criptografia fornecidas pelo cliente (SSE-C)

Utilize os três cabeçalhos se o objeto estiver criptografado com uma chave exclusiva que você forneceu.

- `x-amz-server-side-encryption-customer-algorithm`: Especifique `AES256`.
- `x-amz-server-side-encryption-customer-key`: Especifique sua chave de criptografia para o objeto.
- `x-amz-server-side-encryption-customer-key-MD5`: Especifique o resumo MD5 da chave de criptografia do objeto.

> As chaves de criptografia que você fornece nunca são armazenadas. Se você perder uma chave de criptografia, perderá o objeto correspondente. Antes de usar chaves fornecidas pelo cliente para proteger dados de objetos, revise as considerações em "Use criptografia do lado do servidor".

Comportamento de GetObject para objetos do Cloud Storage Pool

Se um objeto foi armazenado em um "Pool de storage em nuvem", o comportamento de uma solicitação GetObject depende do estado do objeto. Veja "HeadObject" para mais detalhes.

> Se um objeto estiver armazenado em um Cloud Storage Pool e uma ou mais cópias do objeto também existirem na grid, as solicitações GetObject tentarão recuperar os dados da grid antes de recuperá-los do Cloud Storage Pool.

Estado do objeto	Comportamento de GetObject
Objeto inserido no StorageGRID, mas ainda não avaliado pelo ILM, ou objeto armazenado em um pool de storage tradicional ou usando codificação de apagamento	`200 OK` Uma cópia do objeto é recuperada.
Objeto no pool de Cloud Storage, mas ainda não foi transferido para um estado não recuperável	`200 OK` Uma cópia do objeto é recuperada.
Objeto passou para um estado não recuperável	`403 Forbidden`, `InvalidObjectState` Utilize uma solicitação "RestoreObject" para restaurar o objeto a um estado recuperável.
Objeto em processo de restauração a partir de um estado irrecuperável	`403 Forbidden`, `InvalidObjectState` Aguarde a conclusão da solicitação RestoreObject.

Estado do objeto	Comportamento de GetObject
Objeto totalmente restaurado ao pool de storage em nuvem	`200 OK` Uma cópia do objeto é recuperada.

Objetos multipartes ou segmentados em um pool de storage em nuvem

Se você carregou um objeto multipart ou se StorageGRID dividiu um objeto grande em segmentos, StorageGRID determina se o objeto está disponível no Cloud Storage Pool ao amostrar um subconjunto das partes ou segmentos do objeto. Em alguns casos, uma solicitação GetObject pode retornar incorretamente `200 OK` quando algumas partes do objeto já foram transferidas para um estado não recuperável ou quando algumas partes do objeto ainda não foram restauradas.

Nesses casos:

- A solicitação GetObject pode retornar alguns dados, mas ser interrompida no meio da transferência.
- Uma solicitação GetObject subsequente pode retornar `403 Forbidden`.

GetObject e replicação entre grades

Se você estiver usando "federação de grid" e "replicação entre grids" estiver habilitado para um bucket, o cliente pode verificar o status de replicação de um objeto emitindo uma solicitação GetObject. A resposta inclui o cabeçalho de resposta específico do StorageGRID `x-ntap-sg-cgr-replication-status`, que terá um dos seguintes valores:

Grid	Status de replicação
Fonte	• **CONCLUÍDO**: A replicação foi bem-sucedida. • **PENDENTE**: O objeto ainda não foi replicado. • **FALHA**: A replicação falhou com uma falha permanente. Um usuário deve resolver o erro.
Destino	**RÉPLICA**: O objeto foi replicado da grid de origem.

> StorageGRID não suporta o `x-amz-replication-status` header.

Recuperar metadados de um objeto no StorageGRID com a solicitação S3 HeadObject

Você pode usar a solicitação S3 HeadObject para recuperar metadados de um objeto sem retornar o próprio objeto. Se o objeto estiver armazenado em um pool de storage em nuvem, você pode usar HeadObject para determinar o estado de transição do objeto.

HeadObject e objetos multipartes

Você pode usar o `partNumber`parâmetro de solicitação para recuperar metadados de uma parte específica de um objeto multipart ou segmentado. O `x-amz-mp-parts-count`elemento de resposta indica quantas partes o objeto possui.

Você pode definir `partNumber` como 1 tanto para objetos segmentados/multipartes quanto para objetos não segmentados/não multipartes; no entanto, o elemento de resposta `x-amz-mp-parts-count` só é retornado para objetos segmentados ou multipartes.

Caracteres UTF-8 nos metadados do usuário

StorageGRID não analisa nem interpreta caracteres UTF-8 escapados em metadados definidos pelo usuário. Requisições HEAD para um objeto com caracteres UTF-8 escapados em metadados definidos pelo usuário não retornam o `x-amz-missing-meta` cabeçalho se o nome da chave ou o valor incluir caracteres não imprimíveis.

Cabeçalho de solicitação compatível

O seguinte cabeçalho de solicitação é compatível:

- `x-amz-checksum-mode`

  O `partNumber`parâmetro e o `Range`cabeçalho não são suportados com `x-amz-checksum-mode`para HeadObject. Quando você os inclui na solicitação com `x-amz-checksum-mode`ativado, StorageGRID não retorna um valor de checksum na resposta.

Cabeçalho de solicitação não suportado

O seguinte cabeçalho de solicitação não é compatível e retorna `XNotImplemented`:

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

Controle de versões

Se um `versionId` subrecurso não for especificado, a operação busca a versão mais recente do objeto em um bucket versionado. Se a versão atual do objeto for um marcador de exclusão, o status "Não encontrado" é retornado com o `x-amz-delete-marker` cabeçalho de resposta definido como `true`.

Cabeçalhos de solicitação para criptografia do lado do servidor com chaves de criptografia fornecidas pelo cliente (SSE-C)

Utilize todos os três cabeçalhos se o objeto estiver criptografado com uma chave exclusiva que você forneceu.

- `x-amz-server-side-encryption-customer-algorithm`: Especifique `AES256`.
- `x-amz-server-side-encryption-customer-key`: Especifique sua chave de criptografia para o objeto.
- `x-amz-server-side-encryption-customer-key-MD5`: Especifique o resumo MD5 da chave de criptografia do objeto.

> ⚠ As chaves de criptografia que você fornece nunca são armazenadas. Se você perder uma chave de criptografia, perderá o objeto correspondente. Antes de usar chaves fornecidas pelo cliente para proteger dados de objetos, revise as considerações em "Use criptografia do lado do servidor".

HeadObject respostas para objetos do Cloud Storage Pool

Se o objeto estiver armazenado em um "Pool de storage em nuvem", os seguintes cabeçalhos de resposta serão retornados:

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

Os cabeçalhos de resposta fornecem informações sobre o estado de um objeto à medida que ele é movido para um pool de storage em nuvem, opcionalmente transitado para um estado não recuperável e restaurado.

Estado do objeto	Resposta a HeadObject
Objeto inserido no StorageGRID, mas ainda não avaliado pelo ILM, ou objeto armazenado em um pool de storage tradicional ou usando codificação de apagamento	`200 OK` (Nenhum cabeçalho de resposta especial é retornado.)

Objeto no pool de Cloud Storage, mas ainda não foi transferido para um estado não recuperável	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"`
	
	Até que o objeto seja levado a um estado irrecuperável, o valor para `expiry-date` é definido para um momento distante no futuro. O momento exato da transição não é controlado pelo sistema StorageGRID.

O objeto passou para um estado não recuperável, mas pelo menos uma cópia também existe na grid	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"`
	
	O valor para `expiry-date` é definido para um momento distante no futuro.
	
	Nota: Se a cópia na grade não estiver disponível (por exemplo, se um Storage Node estiver inativo), você deve emitir uma "RestoreObject" solicitação para restaurar a cópia do Cloud Storage Pool antes de conseguir recuperar o objeto.

O objeto passou para um estado não recuperável e não existe nenhuma cópia no grid	`200 OK`
	
	`x-amz-storage-class: GLACIER`

Estado do objeto	Resposta a HeadObject

Objeto em processo de restauração a partir de um estado irrecuperável	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="true"`

Objeto totalmente restaurado ao pool de storage em nuvem	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2018 00:00:00 GMT"`
	
	O `expiry-date` indica quando o objeto no pool de storage em nuvem será retornado a um estado não recuperável.

Objetos multipartes ou segmentados no Cloud Storage Pool

Se você carregou um objeto multipart ou se StorageGRID dividiu um objeto grande em segmentos, StorageGRID determina se o objeto está disponível no Cloud Storage Pool ao amostrar um subconjunto das partes ou segmentos do objeto. Em alguns casos, uma solicitação HeadObject pode retornar incorretamente `x-amz-restore: ongoing-request="false"` quando algumas partes do objeto já foram transferidas para um estado não recuperável ou quando algumas partes do objeto ainda não foram restauradas.

HeadObject e replicação entre grades

Se você estiver usando "federação de grid" e "replicação entre grids" estiver habilitado para um bucket, o cliente S3 pode verificar o status de replicação de um objeto emitindo uma solicitação HeadObject. A resposta inclui o cabeçalho de resposta específico do StorageGRID `x-ntap-sg-cgr-replication-status`, que terá um dos seguintes valores:

Grid	Status de replicação
Fonte	• **CONCLUÍDO**: A replicação foi bem-sucedida. • **PENDENTE**: O objeto ainda não foi replicado. • **FALHA**: A replicação falhou com uma falha permanente. Um usuário deve resolver o erro.
Destino	**RÉPLICA**: O objeto foi replicado da grid de origem.

> StorageGRID não suporta o `x-amz-replication-status` header.

Adicionar um objeto a um bucket no StorageGRID com a solicitação S3 PutObject

Você pode usar a solicitação S3 PutObject para adicionar um objeto a um bucket.

Resolver conflitos

Solicitações conflitantes de clientes, como dois clientes gravando na mesma chave, são resolvidas com base no princípio "latest-wins". O momento da avaliação do "latest-wins" é determinado quando o sistema StorageGRID conclui uma determinada solicitação, e não quando os clientes S3 iniciam uma operação.

Tamanho do objeto

O tamanho máximo *recomendado* para uma única operação PutObject é de 5 GiB (5.368.709.120 bytes). Se você tiver objetos maiores que 5 GiB, use "upload multipart" em vez disso.

O tamanho máximo *suportado* para uma única operação PutObject é de 5 TiB (5.497.558.138.880 bytes).

> Se você atualizou do StorageGRID 11.6 ou anterior, o alerta "Tamanho do objeto S3 PUT muito grande" será acionado se você tentar fazer o upload de um objeto que exceda 5 GiB. Se você tiver uma nova instalação do StorageGRID 11.7 ou 11.8, o alerta não será acionado nesse caso. No entanto, para alinhar com o padrão AWS S3, as versões futuras do StorageGRID não suportarão uploads de objetos maiores que 5 GiB.

Tamanho dos metadados do usuário

Amazon S3 limita o tamanho dos metadados definidos pelo usuário em cada cabeçalho de solicitação PUT a 2 KB. StorageGRID limita os metadados do usuário a 24 KiB. O tamanho dos metadados definidos pelo usuário é medido pela soma do número de bytes na codificação UTF-8 de cada chave e valor.

Caracteres UTF-8 nos metadados do usuário

Se uma solicitação incluir valores UTF-8 (sem escape) no nome da chave ou no valor dos metadados definidos pelo usuário, o comportamento do StorageGRID será indefinido.

StorageGRID não analisa nem interpreta caracteres UTF-8 escapados incluídos no nome da chave ou no valor dos metadados definidos pelo usuário. Caracteres UTF-8 escapados são tratados como caracteres ASCII:

- PutObject, CopyObject, GetObject, e HeadObject as solicitações são bem-sucedidas se os metadados definidos pelo usuário incluírem caracteres UTF-8 escapados.
- StorageGRID não retorna o `x-amz-missing-meta` cabeçalho se o valor interpretado do nome ou valor da chave incluir caracteres não imprimíveis.

Limites de tags de objeto

Você pode adicionar tags a novos objetos ao carregá-los ou pode adicioná-las a objetos existentes. Tanto StorageGRID quanto Amazon S3 suportam até 10 tags por objeto. As tags associadas a um objeto devem ter chaves de tag exclusivas. Uma chave de tag pode ter até 128 caracteres Unicode e os valores de tag podem ter até 256 caracteres Unicode. Chaves e valores diferenciam maiúsculas de minúsculas.

Propriedade do objeto

No StorageGRID, todos os objetos pertencem à conta proprietária do bucket, incluindo objetos criados por uma conta que não seja proprietária ou por um usuário anônimo.

Cabeçalhos de solicitação suportados

Os seguintes cabeçalhos de solicitação são suportados:

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

  Ao especificar `aws-chunked`para `Content-Encoding`StorageGRID, o StorageGRID não verifica os seguintes itens:

  - StorageGRID não verifica o `chunk-signature` em relação aos dados em blocos.
  - StorageGRID não verifica o valor que você fornece para `x-amz-decoded-content-length` em relação ao objeto.
- `Content-Language`
- `Content-Length`
- `Content-MD5`
- `Content-Type`
- `Expires`
- `Transfer-Encoding`

  A codificação de transferência em partes é suportada se `aws-chunked` a assinatura da carga também for usada.
- `x-amz-checksum-sha256`
- `x-amz-meta-`, seguido por um par nome-valor contendo metadados definidos pelo usuário.

  Ao especificar o par nome-valor para metadados definidos pelo usuário, utilize este formato geral:

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

  Se você deseja usar a opção **Hora de criação definida pelo usuário** como a hora de referência para uma regra ILM, você deve usar `creation-time` como o nome do metadado que registra quando o objeto foi criado. Por exemplo:

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

  O valor de `creation-time` é avaliado em segundos desde 1º de janeiro de 1970.

  > Uma regra ILM não pode usar simultaneamente um **horário de criação definido pelo usuário** para o horário de referência e a opção de ingestão balanceada ou estrita. Um erro é retornado quando a regra ILM é criada.

- `x-amz-tagging`
- Cabeçalhos de solicitação de bloqueio de objeto S3
  - `x-amz-object-lock-mode`

- `x-amz-object-lock-retain-until-date`
- `x-amz-object-lock-legal-hold`

  Se uma solicitação for feita sem esses cabeçalhos, as configurações de retenção padrão do bucket serão usadas para calcular o modo de versão do objeto e a data de retenção. Consulte "Use a API REST S3 para configurar o S3 Object Lock".

- Cabeçalhos de solicitação 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 Cabeçalhos de solicitação para criptografia do lado do servidor

Cabeçalhos de solicitação não suportados

Os seguintes cabeçalhos de solicitação não são suportados:

- `If-Match`

  O `If-Match header` é aceito, mas não funcional.

- `If-None-Match`

  O `If-None-Match header` é aceito, mas não funcional.

- `x-amz-acl`
- `x-amz-sdk-checksum-algorithm`
- `x-amz-trailer`
- `x-amz-website-redirect-location`

  O `x-amz-website-redirect-location`cabeçalho retorna `XNotImplemented`.

Opções de classe de armazenamento

O `x-amz-storage-class`cabeçalho da solicitação é compatível. O valor enviado para `x-amz-storage-class`afeta como o StorageGRID protege os dados do objeto durante a ingestão e não quantas cópias persistentes do objeto são armazenadas no sistema StorageGRID (o que é determinado pelo ILM).

Se a regra ILM correspondente a um objeto ingerido usar a opção de ingestão estrita, o `x-amz-storage-class` cabeçalho não terá efeito.

Os seguintes valores podem ser usados para `x-amz-storage-class`:

- `STANDARD` (Padrão)
  - **Confirmação dupla**: Se a regra ILM especificar a opção de confirmação dupla para o comportamento

de ingestão, assim que um objeto for ingerido, uma segunda cópia desse objeto será criada e distribuída para um nó de armazenamento diferente (confirmação dupla). Quando a ILM é avaliada, o StorageGRID determina se essas cópias intermediárias iniciais atendem às instruções de posicionamento na regra. Se não atenderem, novas cópias do objeto podem precisar ser criadas em locais diferentes e as cópias intermediárias iniciais podem precisar ser excluídas.

  - **Balanceado**: Se a regra ILM especificar a opção Balanceado e StorageGRID não puder criar imediatamente todas as cópias especificadas na regra, StorageGRID faz duas cópias intermediárias em diferentes nós de armazenamento.

    Se StorageGRID puder criar imediatamente todas as cópias de objetos especificadas na regra ILM (posicionamento síncrono), o `x-amz-storage-class` cabeçalho não terá efeito.

- `REDUCED_REDUNDANCY`

  - **Confirmação dupla**: Se a regra ILM especificar a opção de confirmação dupla para o comportamento de ingestão, StorageGRID cria uma única cópia intermediária à medida que o objeto é ingerido (confirmação única).
  - **Balanceado**: Se a regra ILM especificar a opção Balanceado, StorageGRID cria uma única cópia intermediária somente se o sistema não puder criar imediatamente todas as cópias especificadas na regra. Se StorageGRID puder realizar o posicionamento síncrono, este cabeçalho não terá efeito. A `REDUCED_REDUNDANCY` opção é mais adequada quando a regra ILM correspondente ao objeto cria uma única cópia replicada. Nesse caso, usar `REDUCED_REDUNDANCY` essa opção elimina a criação e exclusão desnecessárias de uma cópia extra do objeto para cada operação de ingestão.

  O uso da `REDUCED_REDUNDANCY`opção não é recomendado em outras circunstâncias. `REDUCED_REDUNDANCY` Isso aumenta o risco de perda de dados de objeto durante a ingestão. Por exemplo, você pode perder dados se a cópia única for armazenada inicialmente em um Storage Node que falhe antes que a avaliação do ILM possa ocorrer.

> Ter apenas uma cópia replicada para qualquer período de tempo coloca os dados em risco de perda permanente. Se existir apenas uma cópia replicada de um objeto, esse objeto é perdido se um Storage Node falhar ou apresentar um erro significativo. Você também perde temporariamente o acesso ao objeto durante procedimentos de manutenção, como upgrades.

A especificação `REDUCED_REDUNDANCY`afeta apenas quantas cópias são criadas quando um objeto é ingerido pela primeira vez. Não afeta quantas cópias do objeto são feitas quando o objeto é avaliado pelas políticas ILM ativas e não resulta no armazenamento de dados em níveis mais baixos de redundância no sistema StorageGRID.

> Se você estiver ingerindo um objeto em um bucket com S3 Object Lock ativado, a `REDUCED_REDUNDANCY` opção será ignorada. Se você estiver ingerindo um objeto em um bucket legado Compliant, a `REDUCED_REDUNDANCY` opção retornará um erro. StorageGRID sempre realizará uma ingestão com confirmação dupla para garantir que os requisitos de conformidade sejam atendidos.

Cabeçalhos de solicitação para criptografia do lado do servidor

Você pode usar os seguintes cabeçalhos de solicitação para criptografar um objeto com criptografia do lado do servidor. As opções SSE e SSE-C são mutuamente exclusivas.

- **SSE**: Use o seguinte cabeçalho se você quiser criptografar o objeto com uma chave exclusiva gerenciada pelo StorageGRID.

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

  Quando o `x-amz-server-side-encryption`cabeçalho não está incluído na solicitação PutObject, o "configuração de criptografia de objeto armazenado" é omitido da resposta PutObject.

- **SSE-C**: Utilize todos os três cabeçalhos se você quiser criptografar o objeto com uma chave exclusiva que você fornece e gerencia.
  - `x-amz-server-side-encryption-customer-algorithm`: Especifique `AES256`.
  - `x-amz-server-side-encryption-customer-key`: Especifique sua chave de criptografia para o novo objeto.
  - `x-amz-server-side-encryption-customer-key-MD5`: Especifique o resumo MD5 da chave de criptografia do novo objeto.

> As chaves de criptografia fornecidas por você nunca são armazenadas. Se você perder uma chave de criptografia, perderá o objeto correspondente. Antes de usar chaves fornecidas pelo cliente para proteger dados de objetos, revise as considerações para "usando criptografia do lado do servidor".

> Se um objeto for criptografado com SSE ou SSE-C, quaisquer configurações de criptografia em nível de bucket ou de grid serão ignoradas.

Controle de versões

Se o versionamento estiver habilitado para um bucket, um identificador único `versionId` é gerado automaticamente para a versão do objeto armazenado. Esse `versionId` também é retornado na resposta usando o `x-amz-version-id` cabeçalho da resposta.

Se o versionamento estiver suspenso, a versão do objeto será armazenada com um valor nulo `versionId` e, se já existir uma versão nula, ela será sobrescrita.

Cálculos de assinatura para o cabeçalho Authorization

Ao usar o `Authorization` cabeçalho para autenticar solicitações, StorageGRID difere da AWS das seguintes maneiras:

- StorageGRID não exige que os `host` cabeçalhos sejam incluídos em `CanonicalHeaders`.
- StorageGRID não precisa `Content-Type`ser incluído em `CanonicalHeaders`.
- StorageGRID não exige que os cabeçalhos `x-amz-*` sejam incluídos em `CanonicalHeaders`.

> Como prática recomendada geral, sempre inclua esses cabeçalhos dentro de `CanonicalHeaders` para garantir que sejam verificados; no entanto, se você excluir esses cabeçalhos, StorageGRID não retorna um erro.

Para obter detalhes, consulte "Cálculos de assinatura para o cabeçalho de autorização: transferência de carga em um único bloco (AWS Signature Version 4)".

Informações relacionadas

- "Gerencie objetos com ILM"
- "Referência da API do Amazon Simple Storage Service: PutObject"

Restaurar um objeto no StorageGRID com a solicitação S3 RestoreObject

Você pode usar a solicitação S3 RestoreObject para restaurar um objeto armazenado em um Cloud Storage Pool.

Tipo de solicitação compatível

StorageGRID suporta apenas solicitações RestoreObject para restaurar um objeto. Ele não suporta o tipo de restauração `SELECT`. As solicitações Select retornam `XNotImplemented`.

Controle de versões

Opcionalmente, especifique `versionId` para restaurar uma versão específica de um objeto em um bucket versionado. Se você não especificar `versionId`, a versão mais recente do objeto é restaurada

Comportamento de RestoreObject em objetos do Cloud Storage Pool

Se um objeto foi armazenado em um "Pool de storage em nuvem", uma solicitação RestoreObject terá o seguinte comportamento, com base no estado do objeto. Veja "HeadObject" para mais detalhes.

> Se um objeto estiver armazenado em um Cloud Storage Pool e uma ou mais cópias desse objeto também existirem na grid, não é necessário restaurar o objeto emitindo uma solicitação RestoreObject. Em vez disso, a cópia local pode ser recuperada diretamente, usando uma solicitação GetObject.

Estado do objeto	Comportamento de RestoreObject
Objeto ingerido no StorageGRID, mas ainda não avaliado pelo ILM, ou objeto não está em um Cloud Storage Pool	`403 Forbidden, InvalidObjectState`
Objeto no pool de Cloud Storage, mas ainda não foi transferido para um estado não recuperável	`200 OK`Nenhuma alteração foi feita. **Nota**: Antes que um objeto tenha sido transferido para um estado não recuperável, você não pode alterar seu `expiry-date`.
Objeto passou para um estado não recuperável	`202 Accepted`Restaura uma cópia recuperável do objeto no Cloud Storage Pool pelo número de dias especificado no corpo da solicitação. Ao final desse período, o objeto retorna a um estado não recuperável. Opcionalmente, use o `Tier` elemento de solicitação para determinar quanto tempo a tarefa de restauração levará para ser concluída (`Expedited`, `Standard`, ou `Bulk`). Se você não especificar `Tier`, o `Standard` nível será usado. **Importante**: Se um objeto tiver sido migrado para o S3 Glacier Deep Archive ou se o Cloud Storage Pool usar Azure Blob storage, você não poderá restaurá-lo usando a `Expedited` camada. O seguinte erro é retornado `403 Forbidden, InvalidTier: Retrieval option is not supported by this storage class.`

Estado do objeto	Comportamento de RestoreObject
Objeto em processo de restauração a partir de um estado irrecuperável	`409 Conflict`, `RestoreAlreadyInProgress`
Objeto totalmente restaurado ao pool de storage em nuvem	`200 OK` **Nota:** Se um objeto foi restaurado a um estado recuperável, você pode alterar seu `expiry-date` reenviando a solicitação RestoreObject com um novo valor para `Days`. A data de restauração é atualizada em relação ao momento da solicitação.

Filtrar dados de objetos no StorageGRID com a solicitação S3 SelectObjectContent

Você pode usar a solicitação S3 SelectObjectContent para filtrar o conteúdo de um objeto S3 com base em uma instrução SQL simples.

Para obter mais informações, consulte "Referência da API do Amazon Simple Storage Service: SelectObjectContent".

Antes de começar

- A conta do locatário possui a permissão S3 Select.
- Você tem `s3:GetObject` permissão para o objeto que você deseja consultar.
- O objeto que você deseja consultar deve estar em um dos seguintes formatos:
  - **CSV**. Pode ser usado como está ou compactado em arquivos GZIP ou BZIP2.
  - **Parquet**. Requisitos adicionais para objetos Parquet:
    - O S3 Select suporta apenas compressão colunar usando GZIP ou Snappy. O S3 Select não suporta compressão de objeto inteiro para objetos Parquet.
    - O S3 Select não suporta saída em Parquet. Você deve especificar o formato de saída como CSV ou JSON.
    - O tamanho máximo de um grupo de linhas não compactadas é de 512 MB.
    - Você deve usar os tipos de dados especificados no esquema do objeto.
    - Não é possível usar os tipos lógicos INTERVAL, JSON, LIST, TIME ou UUID.
- Sua expressão SQL tem um comprimento máximo de 256 KB.
- Qualquer registro na entrada ou nos resultados tem um comprimento máximo de 1 MiB.

Exemplo de sintaxe de solicitação CSV

```
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>
```

Exemplo de sintaxe de solicitação Parquet

```
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>
      <PARQUET>
      </PARQUET>
   </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>
```

Exemplo de consulta SQL

Esta consulta obtém o nome do estado, a população em 2010, a população estimada para 2015 e a porcentagem de variação a partir de dados do censo dos EUA. Registros no arquivo que não são estados são ignorados.

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

As primeiras linhas do arquivo a ser consultado, `SUB-EST2020_ALL.csv` são semelhantes a estas:

```
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
```

Exemplo de uso do AWS-CLI (CSV)

```
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
```

As primeiras linhas do arquivo de saída, `changes.csv`, têm a seguinte aparência:

```
Alabama,4779736,4854803,1.570526070895965802295356898372629785 4
Alaska,710231,738430,3.970398363349389142405780654463125377 5
Arizona,6392017,6832810,6.895992297892824753125656580700583243 1
Arkansas,2915918,2979732,2.188470320495981025529524492801237894 9
California,37253956,38904296,4.429972483996062055798852610444914897 1
Colorado,5029196,5454328,8.453279609703022113276157859029554624 6
```

Exemplo de uso da AWS-CLI (Parquet)

```
aws s3api select-object-content  -endpoint-url https://10.224.7.44:10443
--bucket 619c0755-9e38-42e0-a614-05064f74126d --key SUB-
EST2020_ALL.parquet --expression "SELECT STNAME, CENSUS2010POP,
POPESTIMATE2015, CAST((POPESTIMATE2015 - CENSUS2010POP) AS DECIMAL) /
CENSUS2010POP * 100.0 FROM S3Object WHERE NAME = STNAME" --expression-type
'SQL' --input-serialization '{"Parquet":{}}'  --output-serialization
'{"CSV": {}}' changes.csv
```

As primeiras linhas do arquivo de saída, changes.csv, têm a seguinte aparência:

```
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
```

Operações para uploads multipartes

Operações de upload multipart suportadas no StorageGRID

Esta seção descreve como StorageGRID oferece suporte a operações para uploads multipartes.

As seguintes condições e observações aplicam-se a todas as operações de upload multipart:

- Você não deve exceder 1.000 uploads multipart simultâneos para um único bucket, pois os resultados das consultas ListMultipartUploads para esse bucket podem retornar resultados incompletos.
- StorageGRID impõe limites de tamanho da AWS para partes multipartes. Os clientes S3 devem seguir estas diretrizes:
  - Cada parte em um upload multipart deve ter entre 5 MiB (5,242,880 bytes) e 5 GiB (5,368,709,120 bytes).
  - A última parte pode ser menor que 5 MiB (5,242,880 bytes).
  - Em geral, os tamanhos das partes devem ser os maiores possíveis. Por exemplo, use tamanhos de parte de 5 GiB para um objeto de 100 GiB. Como cada parte é considerada um objeto único, o uso de tamanhos de parte grandes reduz a sobrecarga de metadados do StorageGRID.
  - Para objetos menores que 5 GiB, considere usar o upload não multipart em vez disso.
- O ILM é avaliado para cada parte de um objeto multipart à medida que é ingerido e para o objeto como um todo quando o upload multipart é concluído, caso a regra do ILM utilize Balanced ou Strict "opção de ingestão". Você deve estar ciente de como isso afeta o posicionamento do objeto e de suas partes:
  - Se o ILM for alterado durante um upload multipart do S3, algumas partes do objeto podem não atender aos requisitos atuais do ILM quando o upload multipart for concluído. Qualquer parte que não estiver posicionada corretamente é enfileirada para reavaliação do ILM e movida para o local correto

posteriormente.

    - Ao avaliar o ILM para uma parte, StorageGRID filtra pelo tamanho da parte, não pelo tamanho do objeto. Isso significa que partes de um objeto podem ser armazenadas em locais que não atendem aos requisitos de ILM para o objeto como um todo. Por exemplo, se uma regra especificar que todos os objetos com 10 GB ou mais sejam armazenados em DC1, enquanto todos os objetos menores sejam armazenados em DC2, cada parte de 1 GB de um upload multipart de 10 partes é armazenada em DC2 durante a ingestão. No entanto, quando o ILM é avaliado para o objeto como um todo, todas as partes do objeto são movidas para DC1.

- Todas as operações de upload multipart são compatíveis com StorageGRID "valores de consistência".
- Quando um objeto é ingerido usando upload multipart, o "Limiar de segmentação de objetos (1 GiB)" não é aplicado.
- Conforme necessário, você pode usar "criptografia do lado do servidor" com uploads multipartes. Para usar SSE (criptografia do lado do servidor com chaves gerenciadas pelo StorageGRID), você inclui o cabeçalho de solicitação `x-amz-server-side-encryption` apenas na solicitação CreateMultipartUpload. Para usar SSE-C (criptografia do lado do servidor com chaves fornecidas pelo cliente), você especifica os mesmos três cabeçalhos de solicitação de chave de criptografia na solicitação CreateMultipartUpload e em cada solicitação subsequente UploadPart.
- Um objeto carregado em várias partes é incluído em um "bucket de branch" se a ingestão foi iniciada antes do carimbo de data/hora "Antes" do bucket base, independentemente de quando o upload for concluído.

Operação	Implementação
AbortMultipartUpload	Implementado com todo o comportamento da API REST do Amazon S3. Sujeito a alterações sem aviso prévio.
CompleteMultipartUpload	Consulte "CompleteMultipartUpload"
CreateMultipartUpload (anteriormente chamado de Initiate Multipart Upload)	Consulte "CreateMultipartUpload"
ListMultipartUploads	Consulte "ListMultipartUploads"
ListParts	Implementado com todo o comportamento da API REST do Amazon S3. Sujeito a alterações sem aviso prévio.
UploadPart	Consulte "UploadPart"
UploadPartCopy	Consulte "UploadPartCopy"

Como StorageGRID S3 API REST conclui uploads multipartes

A operação CompleteMultipartUpload completa um upload de objeto em várias partes, montando as partes previamente carregadas.

StorageGRID suporta valores não consecutivos em ordem crescente para o parâmetro de solicitação `partNumber`CompleteMultipartUpload. O parâmetro pode começar com qualquer valor.

Resolver conflitos

Solicitações conflitantes de clientes, como dois clientes gravando na mesma chave, são resolvidas com base no princípio "latest-wins". O momento da avaliação do "latest-wins" é determinado quando o sistema StorageGRID conclui uma determinada solicitação, e não quando os clientes S3 iniciam uma operação.

Cabeçalhos de solicitação suportados

Os seguintes cabeçalhos de solicitação são suportados:

- `x-amz-checksum-sha256`
- `x-amz-storage-class`

  O `x-amz-storage-class` cabeçalho afeta quantas cópias de objetos o StorageGRID cria se a regra ILM correspondente especificar o "Opção de confirmação dupla ou ingestão balanceada".

- `STANDARD`

  (Padrão) Especifica uma operação de ingestão de confirmação dupla quando a regra ILM usa a opção Dual commit ou quando a opção Balanced recorre à criação de cópias intermediárias.

- `REDUCED_REDUNDANCY`

  Especifica uma operação de ingestão de confirmação única quando a regra ILM usa a opção de confirmação dupla ou quando a opção balanceada recorre à criação de cópias intermediárias.

  Se você estiver ingerindo um objeto em um bucket com S3 Object Lock ativado, a `REDUCED_REDUNDANCY` opção será ignorada. Se você estiver ingerindo um objeto em um bucket legado Compliant, a `REDUCED_REDUNDANCY` opção retornará um erro. StorageGRID sempre realizará uma ingestão com confirmação dupla para garantir que os requisitos de conformidade sejam atendidos.

Se um upload multipart não for concluído em 15 dias, a operação é marcada como inativa e todos os dados associados são excluídos do sistema.

O `ETag`valor retornado não é uma soma MD5 dos dados, mas segue a implementação da API do Amazon S3 do valor `ETag`para objetos multipartes.

Cabeçalhos de solicitação não suportados

Os seguintes cabeçalhos de solicitação não são suportados:

- `If-Match`

  O `If-Match header` é aceito, mas não funcional.

- `If-None-Match`

O `If-None-Match header` é aceito, mas não funcional.

- `x-amz-sdk-checksum-algorithm`
- `x-amz-trailer`

Controle de versões

Esta operação conclui um upload multipart. Se o versionamento estiver habilitado para um bucket, a versão do objeto é criada após a conclusão do upload multipart.

Se o versionamento estiver habilitado para um bucket, um identificador único `versionId` é gerado automaticamente para a versão do objeto armazenado. Esse `versionId` também é retornado na resposta usando o `x-amz-version-id` cabeçalho da resposta.

Se o versionamento estiver suspenso, a versão do objeto será armazenada com um valor nulo `versionId` e, se já existir uma versão nula, ela será sobrescrita.

> Quando o versionamento está habilitado para um bucket, a conclusão de um upload multipart sempre cria uma nova versão, mesmo que haja uploads multipart simultâneos concluídos na mesma chave de objeto. Quando o versionamento não está habilitado para um bucket, é possível iniciar um upload multipart e depois outro upload multipart pode ser iniciado e concluído primeiro na mesma chave de objeto. Em buckets sem versionamento, o upload multipart que for concluído por último terá precedência.

Falha na replicação, notificação ou notificação de metadados

Se o bucket onde ocorre o upload multipart estiver configurado para um serviço de plataforma, o upload multipart será bem-sucedido mesmo que a replicação ou notificação associada falhe.

Um locatário pode acionar a falha na replicação ou a notificação atualizando os metadados ou as tags do objeto. Um locatário pode reenviar os valores existentes para evitar alterações indesejadas.

Consulte "Solucionar problemas de serviços da plataforma".

Cabeçalhos e opções de solicitação S3 CreateMultipartUpload no StorageGRID

A operação CreateMultipartUpload (anteriormente chamada de Initiate Multipart Upload) inicia um upload multipart para um objeto e retorna um ID de upload.

O `x-amz-storage-class`cabeçalho da solicitação é compatível. O valor enviado para `x-amz-storage-class`afeta como o StorageGRID protege os dados do objeto durante a ingestão e não quantas cópias persistentes do objeto são armazenadas no sistema StorageGRID (o que é determinado pelo ILM).

Se a regra ILM correspondente a um objeto ingerido usar o Strict "opção de ingestão", o cabeçalho `x-amz-storage-class` não terá efeito.

Os seguintes valores podem ser usados para `x-amz-storage-class`:

- `STANDARD` (Padrão)
  - **Confirmação dupla**: Se a regra ILM especificar a opção de ingestão de confirmação dupla, assim que um objeto for ingerido, uma segunda cópia desse objeto será criada e distribuída para um Storage Node diferente (confirmação dupla). Quando a ILM é avaliada, o StorageGRID determina se essas

cópias intermediárias iniciais atendem às instruções de posicionamento na regra. Se não atenderem, novas cópias do objeto podem precisar ser criadas em locais diferentes e as cópias intermediárias iniciais podem precisar ser excluídas.

- **Balanceado**: Se a regra ILM especificar a opção Balanceado e StorageGRID não puder criar imediatamente todas as cópias especificadas na regra, StorageGRID faz duas cópias intermediárias em diferentes nós de armazenamento.

  Se StorageGRID puder criar imediatamente todas as cópias de objetos especificadas na regra ILM (posicionamento síncrono), o `x-amz-storage-class` cabeçalho não terá efeito.

- `REDUCED_REDUNDANCY`

  - **Confirmação dupla**: Se a regra ILM especificar a opção de confirmação dupla, StorageGRID cria uma única cópia intermediária à medida que o objeto é ingerido (confirmação única).
  - **Balanceado**: Se a regra ILM especificar a opção Balanceado, StorageGRID cria uma única cópia intermediária somente se o sistema não puder criar imediatamente todas as cópias especificadas na regra. Se StorageGRID puder realizar o posicionamento síncrono, este cabeçalho não terá efeito. A `REDUCED_REDUNDANCY` opção é mais adequada quando a regra ILM correspondente ao objeto cria uma única cópia replicada. Nesse caso, usar `REDUCED_REDUNDANCY` essa opção elimina a criação e exclusão desnecessárias de uma cópia extra do objeto para cada operação de ingestão.

  O uso da `REDUCED_REDUNDANCY`opção não é recomendado em outras circunstâncias. `REDUCED_REDUNDANCY` Isso aumenta o risco de perda de dados de objeto durante a ingestão. Por exemplo, você pode perder dados se a cópia única for armazenada inicialmente em um Storage Node que falhe antes que a avaliação do ILM possa ocorrer.

> Ter apenas uma cópia replicada para qualquer período de tempo coloca os dados em risco de perda permanente. Se existir apenas uma cópia replicada de um objeto, esse objeto é perdido se um Storage Node falhar ou apresentar um erro significativo. Você também perde temporariamente o acesso ao objeto durante procedimentos de manutenção, como upgrades.

A especificação `REDUCED_REDUNDANCY`afeta apenas quantas cópias são criadas quando um objeto é ingerido pela primeira vez. Não afeta quantas cópias do objeto são feitas quando o objeto é avaliado pelas políticas ILM ativas e não resulta no armazenamento de dados em níveis mais baixos de redundância no sistema StorageGRID.

> Se você estiver ingerindo um objeto em um bucket com S3 Object Lock ativado, a `REDUCED_REDUNDANCY` opção será ignorada. Se você estiver ingerindo um objeto em um bucket legado Compliant, a `REDUCED_REDUNDANCY` opção retornará um erro. StorageGRID sempre realizará uma ingestão com confirmação dupla para garantir que os requisitos de conformidade sejam atendidos.

Cabeçalhos de solicitação suportados

Os seguintes cabeçalhos de solicitação são suportados:

- `Content-Type`
- `x-amz-checksum-algorithm`

  Atualmente, apenas o valor SHA256 para `x-amz-checksum-algorithm` é suportado.

- `x-amz-meta-`, seguido por um par nome-valor contendo metadados definidos pelo usuário

  Ao especificar o par nome-valor para metadados definidos pelo usuário, utilize este formato geral:

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

  Se você deseja usar a opção **Hora de criação definida pelo usuário** como a hora de referência para uma regra ILM, você deve usar `creation-time` como o nome do metadado que registra quando o objeto foi criado. Por exemplo:

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

  O valor de `creation-time` é avaliado em segundos desde 1º de janeiro de 1970.

  > Adicionar `creation-time` como metadados definidos pelo usuário não é permitido se você estiver adicionando um objeto a um bucket com a Compliance legada ativada. Um erro será retornado.

- Cabeçalhos da solicitação S3 Object Lock:

  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    Se uma solicitação for feita sem esses cabeçalhos, as configurações de retenção padrão do bucket serão usadas para calcular a data de retenção da versão do objeto.

  "Use a API REST S3 para configurar o S3 Object Lock"

- Cabeçalhos de solicitação 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`

    Cabeçalhos de solicitação para criptografia do lado do servidor

  > Para obter informações sobre como StorageGRID lida com caracteres UTF-8, consulte "PutObject".

Cabeçalhos de solicitação para criptografia do lado do servidor

Você pode usar os seguintes cabeçalhos de solicitação para criptografar um objeto multipart com criptografia do lado do servidor. As opções SSE e SSE-C são mutuamente exclusivas.

- **SSE**: Utilize o seguinte cabeçalho na solicitação CreateMultipartUpload se você quiser criptografar o objeto com uma chave exclusiva gerenciada pelo StorageGRID. Não especifique este cabeçalho em nenhuma das solicitações UploadPart.
  - `x-amz-server-side-encryption`
- **SSE-C**: Utilize todos os três cabeçalhos na solicitação CreateMultipartUpload (e em cada solicitação UploadPart subsequente) se você quiser criptografar o objeto com uma chave exclusiva que você fornece e gerencia.
  - `x-amz-server-side-encryption-customer-algorithm`: Especifique `AES256`.
  - `x-amz-server-side-encryption-customer-key`: Especifique sua chave de criptografia para o novo objeto.
  - `x-amz-server-side-encryption-customer-key-MD5`: Especifique o resumo MD5 da chave de criptografia do novo objeto.

> ⚠ As chaves de criptografia fornecidas por você nunca são armazenadas. Se você perder uma chave de criptografia, perderá o objeto correspondente. Antes de usar chaves fornecidas pelo cliente para proteger dados de objetos, revise as considerações para "usando criptografia do lado do servidor".

Cabeçalhos de solicitação não suportados

O seguinte cabeçalho de solicitação não é compatível:

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

  O `x-amz-website-redirect-location`cabeçalho retorna `XNotImplemented`.

Controle de versões

O upload multipart consiste em operações separadas para iniciar o upload, listar os uploads, enviar as partes, montar as partes enviadas e concluir o upload. Os objetos são criados (e versionados, se aplicável) quando a operação CompleteMultipartUpload é realizada.

Operação S3 ListMultipartUploads e parâmetros suportados no StorageGRID

A operação ListMultipartUploads lista os uploads multipartes em andamento para um bucket.

Os seguintes parâmetros de solicitação são suportados:

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

- `Authorization`

Controle de versões

O upload multipart consiste em operações separadas para iniciar o upload, listar os uploads, enviar as partes, montar as partes enviadas e concluir o upload. Os objetos são criados (e versionados, se aplicável) quando a operação CompleteMultipartUpload é realizada.

Cabeçalhos de solicitação suportados e não suportados para operações UploadPart do S3 no StorageGRID

A operação UploadPart carrega uma parte em um upload multipartes de um objeto.

Cabeçalhos de solicitação suportados

Os seguintes cabeçalhos de solicitação são suportados:

- `x-amz-checksum-sha256`
- `Content-Length`
- `Content-MD5`

Cabeçalhos de solicitação para criptografia do lado do servidor

Se você especificou criptografia SSE-C para a solicitação CreateMultipartUpload, também deverá incluir os seguintes cabeçalhos de solicitação em cada solicitação UploadPart:

- `x-amz-server-side-encryption-customer-algorithm`: Especifique `AES256`.
- `x-amz-server-side-encryption-customer-key`: Especifique a mesma chave de criptografia que você forneceu na solicitação CreateMultipartUpload.
- `x-amz-server-side-encryption-customer-key-MD5`: Especifique o mesmo resumo MD5 que você forneceu na solicitação CreateMultipartUpload.

> As chaves de criptografia que você fornece nunca são armazenadas. Se você perder uma chave de criptografia, perderá o objeto correspondente. Antes de usar chaves fornecidas pelo cliente para proteger dados de objetos, revise as considerações em "Use criptografia do lado do servidor".

Se você especificou um checksum SHA-256 durante a solicitação CreateMultipartUpload, também deverá incluir o seguinte cabeçalho de solicitação em cada solicitação UploadPart:

- `x-amz-checksum-sha256`: Especifique o checksum SHA-256 para esta parte.

Cabeçalhos de solicitação não suportados

Os seguintes cabeçalhos de solicitação não são suportados:

- `x-amz-sdk-checksum-algorithm`
- `x-amz-trailer`

Controle de versões

O upload multipart consiste em operações separadas para iniciar o upload, listar os uploads, enviar as partes, montar as partes enviadas e concluir o upload. Os objetos são criados (e versionados, se aplicável) quando a operação CompleteMultipartUpload é realizada.

Carregar uma parte de um objeto no StorageGRID com a operação S3 UploadPartCopy

A operação UploadPartCopy carrega uma parte de um objeto copiando dados de um objeto existente como fonte de dados.

A operação UploadPartCopy é implementada com todo o comportamento da API REST do Amazon S3. Sujeito a alterações sem aviso prévio.

Esta solicitação lê e grava os dados do objeto especificados em `x-amz-copy-source-range` no sistema StorageGRID.

Os seguintes cabeçalhos de solicitação são suportados:

- `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`

Cabeçalhos de solicitação para criptografia do lado do servidor

Se você especificou criptografia SSE-C para a solicitação CreateMultipartUpload, também deverá incluir os seguintes cabeçalhos de solicitação em cada solicitação UploadPartCopy:

- `x-amz-server-side-encryption-customer-algorithm`: Especifique `AES256`.
- `x-amz-server-side-encryption-customer-key`: Especifique a mesma chave de criptografia que você forneceu na solicitação CreateMultipartUpload.
- `x-amz-server-side-encryption-customer-key-MD5`: Especifique o mesmo resumo MD5 que você forneceu na solicitação CreateMultipartUpload.

Se o objeto de origem estiver criptografado usando uma chave fornecida pelo cliente (SSE-C), você deve incluir os três cabeçalhos a seguir na solicitação UploadPartCopy, para que o objeto possa ser descriptografado e copiado:

- `x-amz-copy-source-server-side-encryption-customer-algorithm`: especifique `AES256`.
- `x-amz-copy-source-server-side-encryption-customer-key`: Especifique a chave de criptografia que você forneceu ao criar o objeto de origem.
- `x-amz-copy-source-server-side-encryption-customer-key-MD5`: Especifique o resumo MD5 que você forneceu ao criar o objeto de origem.

> As chaves de criptografia que você fornece nunca são armazenadas. Se você perder uma chave de criptografia, perderá o objeto correspondente. Antes de usar chaves fornecidas pelo cliente para proteger dados de objetos, revise as considerações em "Use criptografia do lado do servidor".

Controle de versões

O upload multipart consiste em operações separadas para iniciar o upload, listar os uploads, enviar as partes, montar as partes enviadas e concluir o upload. Os objetos são criados (e versionados, se aplicável) quando a operação CompleteMultipartUpload é realizada.

Respostas de erro da API REST do StorageGRID S3

O sistema StorageGRID suporta todas as respostas de erro padrão da API REST S3 aplicáveis. Além disso, a implementação do StorageGRID adiciona diversas respostas personalizadas.

Códigos de erro da API S3 suportados

Nome	Status HTTP
AccessDenied	403 Proibido
BadDigest	400 Solicitação inválida
BucketAlreadyExists	409 Conflito
BucketNotEmpty	409 Conflito
IncompleteBody	400 Solicitação inválida
InternalError	Erro interno do servidor 500
InvalidAccessKeyId	403 Proibido
InvalidArgument	400 Solicitação inválida
InvalidBucketName	400 Solicitação inválida
InvalidBucketState	409 Conflito
InvalidDigest	400 Solicitação inválida
InvalidEncryptionAlgorithmError	400 Solicitação inválida
InvalidPart	400 Solicitação inválida
InvalidPartOrder	400 Solicitação inválida
InvalidRange	416 Intervalo solicitado não satisfatório
InvalidRequest	400 Solicitação inválida

Nome	Status HTTP
InvalidStorageClass	400 Solicitação inválida
InvalidTag	400 Solicitação inválida
URI inválido	400 Solicitação inválida
KeyTooLong	400 Solicitação inválida
XML malformado	400 Solicitação inválida
MetadataTooLarge	400 Solicitação inválida
MethodNotAllowed	405 Método Não Permitido
MissingContentLength	411 Comprimento necessário
MissingRequestBodyError	400 Solicitação inválida
MissingSecurityHeader	400 Solicitação inválida
NoSuchBucket	404 Não encontrado
NoSuchKey	404 Não encontrado
NoSuchUpload	404 Não encontrado
NotImplemented	501 Não Implementado
NoSuchBucketPolicy	404 Não encontrado
ObjectLockConfigurationNotFoundError	404 Não encontrado
PreconditionFailed	412 Pré-condição falhou
RequestTimeTooSkewed	403 Proibido
ServiceUnavailable	503 Serviço indisponível
SignatureDoesNotMatch	403 Proibido
TooManyBuckets	400 Solicitação inválida
UserKeyMustBeSpecified	400 Solicitação inválida

Códigos de erro personalizados do StorageGRID

Nome	Descrição	Status HTTP
XBucketLifecycleNotAllowed	A configuração do ciclo de vida do bucket não é permitida em um bucket Compliant legado	400 Solicitação inválida
XBucketPolicyParseException	Falha ao analisar o JSON da política de bucket recebido.	400 Solicitação inválida
XComplianceConflict	Operação negada devido a configurações de Compliance legadas.	403 Proibido
XConformidadeRedundânciaReduzidaProibido	Redução de redundância não é permitida no bucket Compliant legado	400 Solicitação inválida
XMaxBucketPolicyLengthExceeded	Sua política excede o comprimento máximo permitido para a política de bucket.	400 Solicitação inválida
XMissingInternalRequestHeader	Falta um cabeçalho em uma solicitação interna.	400 Solicitação inválida
XNoSuchBucketCompliance	O bucket especificado não tem Compliance legada ativada.	404 Não encontrado
XNotAcceptable	A solicitação contém um ou mais cabeçalhos Accept que não puderam ser atendidos.	406 Não Aceitável
XNãoImplementado	A solicitação que você forneceu implica uma funcionalidade que não está implementada.	501 Não Implementado

Operações personalizadas do StorageGRID

Operações de bucket personalizadas suportadas no StorageGRID

O sistema StorageGRID suporta operações personalizadas que são adicionadas à API REST S3.

A tabela a seguir lista as operações personalizadas suportadas pelo StorageGRID.

Operação	Descrição
"GET consistência do bucket"	Retorna a consistência que está sendo aplicada a um determinado bucket.

Operação	Descrição
"Consistência de PUT Bucket"	Define a consistência aplicada a um determinado bucket.
"Obter o último tempo de acesso do bucket"	Retorna se as atualizações de tempo de acesso estão habilitadas ou desabilitadas para um determinado bucket.
"Último tempo de acesso do PUT Bucket"	Permite que você ative ou desative as atualizações do tempo de acesso para um bucket específico.
"EXCLUIR configuração de notificação de metadados do bucket"	Exclui o arquivo XML de configuração de notificação de metadados associado a um bucket específico.
"GET Configuração de notificação de metadados do bucket"	Retorna o XML de configuração de notificação de metadados associado a um bucket específico.
"PUT Configuração de notificação de metadados do bucket"	Configura o serviço de notificação de metadados para um bucket.
"GET Uso de Armazenamento"	Mostra a quantidade total de armazenamento em uso por uma conta e para cada bucket associado à conta.
"Obsoleto: CreateBucket com configurações de conformidade"	Obsoleto e sem suporte: você não pode mais criar novos buckets com Compliance ativado.
"Obsoleto: GET Bucket compliance"	Obsoleto, mas ainda suportado: retorna as configurações de conformidade atualmente em vigor para um bucket Compliant legado existente.
"Obsoleto: PUT Bucket compliance"	Obsoleto, mas ainda compatível: permite modificar as configurações de conformidade de um bucket legado compatível.

Recupere o nível de consistência do bucket no StorageGRID com a solicitação GET Bucket consistency

A solicitação GET Bucket consistency permite determinar a consistência que está sendo aplicada a um bucket específico.

A consistência padrão está definida para garantir a leitura após a gravação de objetos recém-criados.

Você deve ter a permissão s3:GetBucketConsistency, ou ser o usuário raiz da conta, para concluir esta operação.

Exemplo de solicitação

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

Resposta

Na resposta XML, `<Consistency>` retornará um dos seguintes valores:

Consistência	Descrição
all	Todos os nós recebem os dados imediatamente ou a solicitação falhará.
strong-global	Garante a consistência de leitura após gravação para todas as solicitações do cliente em todos os sites.
site forte	Garante a consistência de leitura após gravação para todas as solicitações do cliente dentro de um site.
leitura-após-nova-gravação	(Padrão) Oferece consistência de leitura após gravação para novos objetos e consistência eventual para atualizações de objetos. Oferece alta disponibilidade e garantias de proteção de dados. Recomendado para a maioria dos casos.
disponível	Garante consistência eventual tanto para novos objetos quanto para atualizações de objetos. Para buckets do S3, use somente quando necessário (por exemplo, para um bucket que contém valores de log que são lidos raramente ou para operações HEAD ou GET em chaves que não existem). Não é compatível com buckets do S3 FabricPool.

Exemplo de resposta

```
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>
```

Informações relacionadas

"Consistência"

Especifique os níveis de consistência no StorageGRID com a solicitação PUT Bucket consistency

A solicitação PUT Bucket de consistência permite que você especifique a consistência a ser aplicada às operações realizadas em um bucket.

A consistência padrão está definida para garantir a leitura após a gravação de objetos recém-criados.

Antes de começar

Você deve ter a permissão s3:PutBucketConsistency ou ser o root da conta para concluir esta operação.

Solicitação

O `x-ntap-sg-consistency`parâmetro deve conter um dos seguintes valores:

Consistência	Descrição
all	Todos os nós recebem os dados imediatamente ou a solicitação falhará.
strong-global	Garante a consistência de leitura após gravação para todas as solicitações do cliente em todos os sites.
site forte	Garante a consistência de leitura após gravação para todas as solicitações do cliente dentro de um site.
leitura-após-nova-gravação	(Padrão) Oferece consistência de leitura após gravação para novos objetos e consistência eventual para atualizações de objetos. Oferece alta disponibilidade e garantias de proteção de dados. Recomendado para a maioria dos casos.
disponível	Garante consistência eventual tanto para novos objetos quanto para atualizações de objetos. Para buckets do S3, use somente quando necessário (por exemplo, para um bucket que contém valores de log que são lidos raramente ou para operações HEAD ou GET em chaves que não existem). Não é compatível com buckets do S3 FabricPool.

Nota: Em geral, você deve usar a consistência "Leitura após nova gravação". Se as solicitações não estiverem funcionando corretamente, altere o comportamento do cliente do aplicativo, se possível. Ou configure o cliente para especificar a consistência para cada solicitação de API. Defina a consistência no nível do bucket somente como último recurso.

Exemplo de solicitação

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

Informações relacionadas

"Consistência"

Recupere o status do tempo de acesso ao bucket no StorageGRID com a solicitação GET Bucket last access time

A solicitação GET Bucket tempo de acesso permite que você determine se as atualizações de tempo de acesso estão ativadas ou desativadas para buckets individuais.

Você deve ter a permissão s3:GetBucketLastAccessTime, ou ser o usuário raiz da conta, para concluir esta operação.

Exemplo de solicitação

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

Exemplo de resposta

Este exemplo mostra que as atualizações de tempo de acesso do último acesso estão habilitadas para o bucket.

```
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>
```

Ativar e desativar atualizações do tempo de acesso no StorageGRID com a solicitação PUT Bucket last access time

A solicitação PUT Bucket de tempo de acesso permite que você habilite ou desabilite as atualizações de tempo de acesso para buckets individuais. Desabilitar as atualizações de tempo de acesso melhora o desempenho e é a configuração padrão para todos os buckets criados com a versão 10.3.0 ou posterior.

Você precisa ter a permissão s3:PutBucketLastAccessTime para um bucket ou ser o root da conta para

concluir esta operação.

> A partir da versão 10.3 do StorageGRID, as atualizações do tempo de acesso estão desativadas por padrão para todos os novos buckets. Se você tiver buckets criados com uma versão anterior do StorageGRID e quiser que eles sigam o novo comportamento padrão, você deve desativar explicitamente as atualizações do tempo de acesso para cada um desses buckets. Você pode ativar ou desativar as atualizações do tempo de acesso usando a solicitação PUT Bucket last access time ou na página de detalhes de um bucket no Tenant Manager. Consulte "Ativar ou desativar atualizações do tempo de acesso mais recente".

Se as atualizações de tempo de acesso estiverem desativadas para um bucket, o seguinte comportamento será aplicado às operações no bucket:

- GetObject, GetObjectAcl, GetObjectTagging, e HeadObject não atualizam o tempo de acesso. O objeto não é adicionado às filas para avaliação do gerenciamento do ciclo de vida das informações (ILM).
- CopyObject e PutObjectTagging solicitações que atualizam apenas os metadados também atualizam o tempo de acesso. O objeto é adicionado às filas para avaliação do ILM.
- Se as atualizações do último tempo de acesso estiverem desativadas para o bucket de origem, as solicitações CopyObject não atualizam o último tempo de acesso para o bucket de origem. O objeto copiado não é adicionado às filas de avaliação de ILM para o bucket de origem. No entanto, para o destino, as solicitações CopyObject sempre atualizam o último tempo de acesso. A cópia do objeto é adicionada às filas de avaliação de ILM.
- CompleteMultipartUpload solicitações atualizam o tempo de acesso. O objeto concluído é adicionado às filas para avaliação do ILM.

Exemplos de solicitação

Este exemplo ativa o tempo de último acesso para um bucket.

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

Este exemplo desativa o tempo de acesso para um bucket.

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

Desative o serviço de integração de pesquisa com a solicitação de configuração de notificação de metadados DELETE Bucket no StorageGRID

A solicitação DELETE Bucket metadata notification configuration permite que você desative o serviço de integração de pesquisa para buckets individuais, excluindo o XML de configuração.

Você precisa ter a permissão s3:DeleteBucketMetadataNotification para um bucket ou ser o administrador da conta para concluir esta operação.

Exemplo de solicitação

Este exemplo mostra como desativar o serviço de integração de pesquisa para um bucket.

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

Configurar a integração de pesquisa no StorageGRID com a solicitação GET Bucket metadata notification configuration

A solicitação GET Bucket metadata notification configuration permite que você recupere o XML de configuração usado para configurar a integração de pesquisa para buckets individuais.

Você deve ter a permissão s3:GetBucketMetadataNotification, ou ser o usuário raiz da conta, para concluir esta operação.

Exemplo de solicitação

Esta solicitação recupera a configuração de notificação de metadados para o bucket chamado `bucket`.

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

Resposta

O corpo da resposta inclui a configuração de notificação de metadados para o bucket. A configuração de notificação de metadados permite que você determine como o bucket está configurado para integração de pesquisa. Ou seja, permite que você determine quais objetos são indexados e para quais endpoints os metadados dos objetos estão sendo enviados.

```
<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 configuração de notificação de metadados inclui uma ou mais regras. Cada regra especifica os objetos aos quais se aplica e o destino para onde o StorageGRID deve enviar os metadados dos objetos. Os destinos devem ser especificados usando o URN de um endpoint do StorageGRID.

Nome	Descrição	Obrigatório
MetadataNotificationConfiguration	Tag de contêiner para regras usadas para especificar os objetos e o destino para notificações de metadados. Contém um ou mais elementos de Rule.	Sim
Regra	Etiqueta de contêiner para uma regra que identifica os objetos cujos metadados devem ser adicionados a um índice específico. Regras com prefixos sobrepostos são rejeitadas. Incluído no elemento MetadataNotificationConfiguration.	Sim
ID	Identificador único para a regra. Incluído no elemento Rule.	Não
Status	O status pode ser 'Ativado' ou 'Desativado'. Nenhuma ação é executada para regras desativadas. Incluído no elemento Rule.	Sim

Nome	Descrição	Obrigatório
Prefixo	Os objetos que correspondem ao prefixo são afetados pela regra e seus metadados são enviados para o destino especificado. Para corresponder a todos os objetos, especifique um prefixo vazio. Incluído no elemento Rule.	Sim
Destino	Etiqueta de contêiner para o destino de uma regra. Incluído no elemento Rule.	Sim
Urna	URN do destino para onde os metadados do objeto são enviados. Deve ser o URN de um endpoint do StorageGRID com as seguintes propriedades: • `es`deve ser o terceiro elemento. • O URN deve terminar com o índice e o tipo onde os metadados estão armazenados, no formato `domain-name/myindex/mytype`. Os endpoints são configurados usando o Tenant Manager ou a Tenant Management API. Eles têm o seguinte formato: • `arn:aws:es:_region:account-ID_:domain/mydomain/myindex/mytype` • `urn:mysite:es:::mydomain/myindex/mytype` O endpoint deve ser configurado antes do envio do XML de configuração, ou a configuração falhará com um erro 404. Urn está incluído no elemento Destination.	Sim

Exemplo de resposta

O XML incluído entre as `<MetadataNotificationConfiguration></MetadataNotificationConfiguration>` tags mostra como a integração com um endpoint de integração de pesquisa é configurada para o bucket. Neste exemplo, os metadados do objeto estão sendo enviados para um índice do Elasticsearch chamado `current` e tipo chamado `2017` que está hospedado em um domínio da AWS chamado `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>
```

Informações relacionadas

"Use uma conta de tenant"

Habilite o serviço de integração de pesquisa no StorageGRID com a solicitação de configuração de notificação de metadados do bucket PUT

A solicitação PUT Bucket metadata notification configuration permite que você habilite o serviço de integração de pesquisa para buckets individuais. O XML de configuração de notificação de metadados que você fornece no corpo da solicitação especifica os objetos cujos metadados são enviados para o índice de pesquisa de destino.

Você precisa ter a permissão s3:PutBucketMetadataNotification para um bucket ou ser o root da conta para concluir esta operação.

Solicitação

A solicitação deve incluir a configuração de notificação de metadados no corpo da solicitação. Cada configuração de notificação de metadados inclui uma ou mais regras. Cada regra especifica os objetos aos quais se aplica e o destino para onde StorageGRID deve enviar os metadados dos objetos.

Os objetos podem ser filtrados pelo prefixo do nome do objeto. Por exemplo, você pode enviar metadados para objetos com o prefixo `/images` para um destino e objetos com o prefixo `/videos` para outro.

Configurações com prefixos sobrepostos não são válidas e são rejeitadas no momento do envio. Por exemplo, uma configuração que incluísse uma regra para objetos com o prefixo `test` e uma segunda regra para objetos com o prefixo `test2` não seria permitida.

Os destinos devem ser especificados usando o URN de um endpoint do StorageGRID. O endpoint deve existir

quando a configuração de notificação de metadados for enviada, ou a solicitação falhará como um `400 Bad Request`. A mensagem de erro 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>
```

A tabela descreve os elementos no XML de configuração da notificação de metadados.

Nome	Descrição	Obrigatório
MetadataNotificationConfiguration	Tag de contêiner para regras usadas para especificar os objetos e o destino para notificações de metadados. Contém um ou mais elementos de Rule.	Sim
Regra	Etiqueta de contêiner para uma regra que identifica os objetos cujos metadados devem ser adicionados a um índice específico. Regras com prefixos sobrepostos são rejeitadas. Incluído no elemento MetadataNotificationConfiguration.	Sim
ID	Identificador único para a regra. Incluído no elemento Rule.	Não
Status	O status pode ser 'Ativado' ou 'Desativado'. Nenhuma ação é executada para regras desativadas. Incluído no elemento Rule.	Sim

Nome	Descrição	Obrigatório
Prefixo	Os objetos que correspondem ao prefixo são afetados pela regra e seus metadados são enviados para o destino especificado. Para corresponder a todos os objetos, especifique um prefixo vazio. Incluído no elemento Rule.	Sim
Destino	Etiqueta de contêiner para o destino de uma regra. Incluído no elemento Rule.	Sim
Urna	URN do destino para onde os metadados do objeto são enviados. Deve ser o URN de um endpoint do StorageGRID com as seguintes propriedades: • `es`deve ser o terceiro elemento. • O URN deve terminar com o índice e o tipo onde os metadados estão armazenados, no formato `domain-name/myindex/mytype`. Os endpoints são configurados usando o Tenant Manager ou a Tenant Management API. Eles têm o seguinte formato: • `arn:aws:es:region:account-ID:domain/mydomain/myindex/mytype` • `urn:mysite:es:::mydomain/myindex/mytype` O endpoint deve ser configurado antes do envio do XML de configuração, ou a configuração falhará com um erro 404. Urn está incluído no elemento Destination.	Sim

Exemplos de solicitação

Este exemplo mostra como habilitar a integração de pesquisa para um bucket. Neste exemplo, os metadados dos objetos são enviados para o mesmo destino.

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

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

Neste exemplo, os metadados dos objetos que correspondem ao prefixo `/images` são enviados para um destino, enquanto os metadados dos objetos que correspondem ao prefixo `/videos` são enviados para um segundo destino.

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

<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 gerado pelo serviço de integração de pesquisa

Ao ativar o serviço de integração de pesquisa para um bucket, um documento JSON é gerado e enviado para o endpoint de destino sempre que os metadados ou tags de um objeto são adicionados, atualizados ou excluídos.

Este exemplo mostra um exemplo do JSON que poderia ser gerado quando um objeto com a chave `SGWS/Tagging.txt` é criado em um bucket chamado `test`. O `test` bucket não é versionado, portanto a `versionId` tag está vazia.

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

Metadados de objeto incluídos em notificações de metadados

A tabela lista todos os campos incluídos no documento JSON que é enviado ao endpoint de destino quando a integração de pesquisa está habilitada.

O nome do documento inclui o nome do bucket, o nome do objeto e o ID da versão, se presente.

Tipo	Nome do item	Descrição
Informações sobre buckets e objetos	bucket	Nome do bucket
Informações sobre buckets e objetos	chave	Nome da chave do objeto
Informações sobre buckets e objetos	ID da versão	Versão do objeto, para objetos em buckets versionados
Informações sobre buckets e objetos	região	Região do bucket, por exemplo `us-east-1`
Metadados do sistema	tamanho	Tamanho do objeto (em bytes) conforme visível para um cliente HTTP
Metadados do sistema	md5	Hash do objeto
Metadados do usuário	metadados *`key:value`*	Todos os metadados do usuário para o objeto, como pares de chave-valor

Tipo	Nome do item	Descrição
Tags	tags *key:value*	Todas as tags de objeto definidas para o objeto, como pares chave-valor

> Para tags e metadados de usuário, StorageGRID envia datas e números para o Elasticsearch como strings ou como notificações de eventos do S3. Para configurar o Elasticsearch para interpretar essas strings como datas ou números, siga as instruções do Elasticsearch para mapeamento dinâmico de campos e para mapeamento de formatos de data. Você deve habilitar os mapeamentos dinâmicos de campos no índice antes de configurar o serviço de integração de pesquisa. Depois que um documento é indexado, você não pode editar os tipos de campo do documento no índice.

Informações relacionadas

"Use uma conta de tenant"

Recupere o uso de storage da conta e do bucket no StorageGRID com a solicitação GET Storage Usage

A solicitação GET Storage Usage informa a quantidade total de storage em uso por uma conta e para cada bucket associado à conta.

A quantidade de armazenamento utilizada por uma conta e seus buckets pode ser obtida por meio de uma solicitação ListBuckets modificada com o parâmetro de consulta `x-ntap-sg-usage`. O uso de armazenamento dos buckets é rastreado separadamente das solicitações PUT e DELETE processadas pelo sistema. Pode haver algum atraso até que os valores de uso correspondam aos valores esperados com base no processamento das solicitações, principalmente se o sistema estiver sob carga elevada.

Por padrão, StorageGRID tenta recuperar informações de uso utilizando consistência global forte. Se a consistência global forte não puder ser alcançada, StorageGRID tenta recuperar as informações de uso com consistência de site forte.

Você deve ter a permissão s3:ListAllMyBuckets, ou ser o root da conta, para concluir esta operação.

Exemplo de solicitação

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

Exemplo de resposta

Este exemplo mostra uma conta que possui quatro objetos e 12 bytes de dados em dois buckets. Cada bucket contém dois objetos e seis bytes de dados.

```
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>
```

Controle de versões

Cada versão de objeto armazenada contribuirá para os valores de `ObjectCount` e `DataBytes` na resposta. Os marcadores de exclusão não são adicionados ao total de `ObjectCount`.

Informações relacionadas

"Consistência"

Solicitações de bucket obsoletas para Compliance legada

Solicitações de bucket obsoletas para StorageGRID Compliance legado

Você pode precisar usar a API REST S3 do StorageGRID para gerenciar buckets que foram criados usando o recurso de Compliance legado.

Recurso de conformidade descontinuado

O recurso de conformidade do StorageGRID, disponível em versões anteriores do StorageGRID, foi descontinuado e substituído pelo S3 Object Lock.

Se você habilitou anteriormente a configuração de Conformidade global, a configuração de Bloqueio de Objeto S3 global também estará habilitada no StorageGRID 11.6. Você não pode mais criar novos buckets com a Conformidade habilitada; no entanto, se necessário, você pode usar a API REST S3 do StorageGRID para gerenciar quaisquer buckets legados em conformidade existentes.

- "Use a API REST S3 para configurar o S3 Object Lock"
- "Gerencie objetos com ILM"
- "NetApp Knowledge Base: como gerenciar buckets legados compatíveis no StorageGRID 11.5"

Solicitações de conformidade obsoletas:

- "Obsoleto - Modificações na solicitação PUT Bucket para fins de conformidade"

  O elemento XML SGCompliance está obsoleto. Anteriormente, você podia incluir esse elemento personalizado do StorageGRID no corpo da requisição XML opcional das solicitações PUT Bucket para criar um bucket em conformidade.

- "Obsoleto - GET Bucket compliance"

  A solicitação GET Bucket compliance foi descontinuada. No entanto, você pode continuar usando essa solicitação para determinar as configurações de conformidade atualmente em vigor para um bucket legado Compliant.

- "Obsoleto - PUT Bucket compliance"

  A solicitação de conformidade de bucket PUT está obsoleta. No entanto, você pode continuar usando essa solicitação para modificar as configurações de conformidade de um bucket legado Compliant existente. Por exemplo, você pode colocar um bucket existente em guarda legal ou aumentar seu período de retenção.

Elemento SGCompliance obsoleto no StorageGRID

O elemento XML SGCompliance está obsoleto. Anteriormente, você podia incluir esse elemento personalizado do StorageGRID no corpo opcional da solicitação XML de CreateBucket para criar um bucket compatível.

> ⓘ O recurso de conformidade do StorageGRID, disponível em versões anteriores do StorageGRID, foi descontinuado e substituído pelo S3 Object Lock. Veja mais detalhes a seguir:
>
> - "Use a API REST S3 para configurar o S3 Object Lock"
> - "NetApp Knowledge Base: como gerenciar buckets legados compatíveis no StorageGRID 11.5"

Não é mais possível criar novos buckets com a Compliance ativada. A seguinte mensagem de erro é exibida se você tentar usar as modificações de solicitação CreateBucket para compliance para criar um novo bucket em compliance:

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

Solicitação GET de conformidade de bucket obsoleta no StorageGRID

A solicitação GET Bucket compliance foi descontinuada. No entanto, você pode continuar usando essa solicitação para determinar as configurações de conformidade atualmente em vigor para um bucket legado Compliant.

> O recurso de conformidade do StorageGRID, disponível em versões anteriores do StorageGRID, foi descontinuado e substituído pelo S3 Object Lock. Veja mais detalhes a seguir:
>
> - "Use a API REST S3 para configurar o S3 Object Lock"
> - "NetApp Knowledge Base: como gerenciar buckets legados compatíveis no StorageGRID 11.5"

Você deve ter a permissão s3:GetBucketCompliance ou ser o root da conta para concluir esta operação.

Exemplo de solicitação

Esta solicitação de exemplo permite que você determine as configurações de conformidade para o bucket chamado `mybucket`.

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

Exemplo de resposta

Na resposta XML, `<SGCompliance>` lista as configurações de conformidade em vigor para o bucket. Este exemplo de resposta mostra as configurações de conformidade para um bucket no qual cada objeto será retido por um ano (525.600 minutos), a partir do momento em que o objeto é inserido na grid. Atualmente, não há guarda legal neste bucket. Cada objeto será excluído automaticamente após um ano.

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

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

Nome	Descrição
RetentionPeriodMinutes	O período de retenção dos objetos adicionados a este bucket, em minutos. O período de retenção começa quando o objeto é ingerido no grid.
LegalHold	• Verdadeiro: Este bucket está atualmente sob guarda legal. Os objetos neste bucket não podem ser excluídos até que a guarda legal seja suspensa, mesmo que o período de retenção tenha expirado. • Falso: Este bucket não está atualmente sob guarda legal. Os objetos neste bucket podem ser excluídos quando o período de retenção expirar.
AutoDelete	• Verdadeiro: Os objetos neste bucket serão excluídos automaticamente quando o período de retenção expirar, a menos que o bucket esteja sob guarda legal. • Falso: Os objetos neste bucket não serão excluídos automaticamente quando o período de retenção expirar. Você deve excluir esses objetos manualmente se precisar excluí-los.

Respostas de erro

Se o bucket não foi criado para ser compatível, o código de status HTTP da resposta é `404 Not Found`, com um código de erro S3 de `XNoSuchBucketCompliance`.

Solicitação PUT de conformidade de bucket obsoleta no StorageGRID

A solicitação de conformidade de bucket PUT está obsoleta. No entanto, você pode continuar usando essa solicitação para modificar as configurações de conformidade de um bucket legado Compliant existente. Por exemplo, você pode colocar um bucket existente em guarda legal ou aumentar seu período de retenção.

> O recurso de conformidade do StorageGRID, disponível em versões anteriores do StorageGRID, foi descontinuado e substituído pelo S3 Object Lock. Veja mais detalhes a seguir:
>
> • "Use a API REST S3 para configurar o S3 Object Lock"
> • "NetApp Knowledge Base: como gerenciar buckets legados compatíveis no StorageGRID 11.5"

Você deve ter a permissão s3:PutBucketCompliance ou ser o root da conta para concluir esta operação.

Você deve especificar um valor para cada campo das configurações de conformidade ao emitir uma solicitação PUT Bucket compliance.

Exemplo de solicitação

Esta solicitação de exemplo modifica as configurações de conformidade para o bucket denominado `mybucket`. Neste exemplo, os objetos em `mybucket` serão retidos por dois anos (1.051.200 minutos) em vez de um ano, a partir do momento em que o objeto for ingerido na grid. Não há guarda legal neste bucket. Cada

objeto será excluído automaticamente após dois anos.

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

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

Nome	Descrição
RetentionPeriodMinutes	O período de retenção dos objetos adicionados a este bucket, em minutos. O período de retenção começa quando o objeto é ingerido no grid. **Importante** Ao especificar um novo valor para RetentionPeriodMinutes, você deve especificar um valor igual ou maior que o período de retenção atual do bucket. Depois que o período de retenção do bucket for definido, você não poderá diminuir esse valor; você só poderá aumentá-lo.
LegalHold	• Verdadeiro: Este bucket está atualmente sob guarda legal. Os objetos neste bucket não podem ser excluídos até que a guarda legal seja suspensa, mesmo que o período de retenção tenha expirado. • Falso: Este bucket não está atualmente sob guarda legal. Os objetos neste bucket podem ser excluídos quando o período de retenção expirar.
AutoDelete	• Verdadeiro: Os objetos neste bucket serão excluídos automaticamente quando o período de retenção expirar, a menos que o bucket esteja sob guarda legal. • Falso: Os objetos neste bucket não serão excluídos automaticamente quando o período de retenção expirar. Você deve excluir esses objetos manualmente se precisar excluí-los.

Consistência para configurações de conformidade

Ao atualizar as configurações de conformidade de um bucket S3 com uma solicitação PUT Bucket compliance, StorageGRID tenta atualizar os metadados do bucket em toda a grid. Por padrão, StorageGRID usa a consistência **Strong-global** para garantir que todos os sites do data center e todos os Storage Nodes que contêm metadados do bucket tenham consistência de leitura após gravação para as configurações de conformidade alteradas.

Se StorageGRID não conseguir atingir a consistência **Strong-global** porque um site de data center ou vários nós de armazenamento em um site estão indisponíveis, o código de status HTTP para a resposta é `503 Service Unavailable`.

Se você receber esta resposta, entre em contato com o administrador da grid para garantir que os serviços de armazenamento necessários sejam disponibilizados o mais rápido possível. Caso o administrador da grid não consiga disponibilizar Storage Nodes suficientes em cada site, o suporte técnico pode orientar você a tentar novamente a solicitação com falha, forçando a consistência **Strong-site**.

> Nunca force a consistência **Strong-site** para conformidade do bucket PUT, a menos que você tenha sido instruído a fazê-lo pelo suporte técnico e a menos que compreenda as possíveis consequências do uso desse nível.

Quando a consistência é reduzida para **Strong-site**, StorageGRID garante que as configurações de conformidade atualizadas terão consistência de leitura após gravação apenas para solicitações de clientes dentro de um site. Isso significa que o sistema StorageGRID pode ter temporariamente várias configurações inconsistentes para este bucket até que todos os sites e Storage Nodes estejam disponíveis. As configurações inconsistentes podem resultar em comportamentos inesperados e indesejados. Por exemplo, se você estiver colocando um bucket sob guarda legal e forçar uma consistência menor, as configurações de conformidade anteriores do bucket (ou seja, guarda legal desativada) podem continuar em vigor em alguns sites do data center. Como resultado, objetos que você acredita estarem sob guarda legal podem ser excluídos quando o período de retenção expirar, seja pelo usuário ou pelo AutoDelete, se ativado.

Para forçar o uso da consistência **Strong-site**, reenvie a solicitação de conformidade do bucket PUT e inclua o `Consistency-Control` cabeçalho da solicitação HTTP, conforme mostrado a seguir:

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

Respostas de erro

- Se o bucket não foi criado para estar em conformidade, o código de status HTTP para a resposta é `404 Not Found`.
- Se `RetentionPeriodMinutes` na solicitação for menor que o período de retenção atual do bucket, o código de status HTTP é `400 Bad Request`.

Informações relacionadas

"Obsoleto: modificações de solicitação PUT Bucket para conformidade"

Gerenciar políticas de acesso

Como StorageGRID usa políticas da AWS para controlar o acesso a buckets e objetos do S3

StorageGRID usa a linguagem de políticas da Amazon Web Services (AWS) para permitir que os clientes S3 controlem o acesso a buckets e objetos dentro desses buckets. O sistema StorageGRID implementa um subconjunto da linguagem de políticas da API REST do S3. As políticas de acesso para a API S3 são escritas em JSON.

Visão geral da política de acesso

StorageGRID suporta três tipos de políticas de acesso:

- **Políticas de bucket**, que são gerenciadas usando as operações da API S3 GetBucketPolicy, PutBucketPolicy e DeleteBucketPolicy ou pelo Tenant Manager ou Tenant Management API. As políticas de bucket são associadas aos buckets, então são configuradas para controlar o acesso de usuários da conta proprietária do bucket ou de outras contas ao bucket e aos objetos nele. Uma política de bucket se aplica a apenas um bucket e, possivelmente, a vários grupos.
- **Políticas de grupo**, que são configuradas usando o Tenant Manager ou a Tenant Management API. As políticas de grupo são associadas a um grupo na conta, então são configuradas para permitir que esse grupo acesse recursos específicos pertencentes a essa conta. Uma política de grupo se aplica a apenas um grupo e possivelmente a vários buckets.
- **Políticas de sessão**, que são incluídas ao fazer uma solicitação AssumeRole. As políticas de sessão aplicam-se apenas à sessão em questão, definindo com mais detalhes as permissões que o usuário possui, além daquelas concedidas pelas políticas de grupo e de bucket.

Não há diferença de prioridade entre as políticas de grupo, bucket e sessão.

As políticas de bucket e grupo do StorageGRID seguem uma gramática específica definida pela Amazon. Dentro de cada política há uma matriz de declarações de política, e cada declaração contém os seguintes elementos:

- ID da declaração (Sid) (opcional)
- Efeito
- Principal/NotPrincipal
- Recurso/NotResource
- Ação/NotAction
- Condição (opcional)

As declarações de política são construídas usando esta estrutura para especificar permissões: conceder <Effect> para permitir/negar <Principal> executar <Action> em <Resource> quando <Condition> se aplica.

Cada elemento da política é usado para uma função específica:

Elemento	Descrição
Sid	O elemento Sid é opcional. O Sid serve apenas como uma descrição para o usuário. Ele é armazenado, mas não interpretado pelo sistema StorageGRID.
Efeito	Use o elemento Effect para determinar se as operações especificadas são permitidas ou negadas. Você deve identificar as operações que permite (ou nega) em buckets ou objetos usando as palavras-chave do elemento Action.

Elemento	Descrição
Principal/NotPrincipal	Você pode permitir que usuários, grupos e contas acessem recursos específicos e executem ações específicas. Se nenhuma assinatura S3 estiver incluída na solicitação, o acesso anônimo será permitido especificando o caractere curinga (*) como o principal. Por padrão, somente o root da conta tem acesso aos recursos pertencentes à conta. Você só precisa especificar o elemento Principal em uma política de bucket. Para políticas de grupo, o grupo ao qual a política está associada é o elemento Principal implícito.
Recurso/NotResource	O elemento Resource identifica buckets e objetos. Você pode conceder ou negar permissões a buckets e objetos usando o Nome de Recurso da Amazon (ARN) para identificar o recurso.
Ação/NotAction	Os elementos Ação e Efeito são os dois componentes das permissões. Quando um grupo solicita um recurso, o acesso a ele é concedido ou negado. O acesso é negado a menos que você atribua permissões especificamente, mas você pode usar a negação explícita para substituir uma permissão concedida por outra política.
Condição	O elemento Condição é opcional. As condições permitem criar expressões para determinar quando uma política deve ser aplicada.

No elemento Action, você pode usar o caractere curinga (*) para especificar todas as operações ou um subconjunto de operações. Por exemplo, esta Action corresponde a permissões como s3:GetObject, s3:PutObject e s3:DeleteObject.

```
s3:*Object
```

No elemento Resource, você pode usar os caracteres curinga (*) e (?). Enquanto o asterisco (*) corresponde a 0 ou mais caracteres, o ponto de interrogação (?) corresponde a qualquer caractere único.

No elemento Principal, caracteres curinga não são suportados, exceto para definir acesso anônimo, que concede permissão a todos. Por exemplo, você define o curinga (*) como o valor Principal.

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

```
"Principal":{"AWS":"*"}
```

No exemplo a seguir, a declaração utiliza os elementos Efeito, Principal, Ação e Recurso. Este exemplo mostra uma declaração de política de bucket completa que usa o Efeito "Permitir" para conceder aos Principais, ao grupo de administradores `federated-group/admin` e ao grupo de finanças `federated-group/finance`, permissões para executar a Ação `s3:ListBucket` no bucket nomeado `mybucket` e a Ação `s3:GetObject` em todos os objetos dentro desse bucket.

```
{
  "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:s3:::mybucket",
        "arn:aws:s3:::mybucket/*"
      ]
    }
  ]
}
```

A política de bucket tem um limite de tamanho de 20.480 bytes e a política de grupo tem um limite de tamanho de 5.120 bytes.

Consistência para políticas

Por padrão, todas as atualizações feitas nas políticas de grupo são eventualmente consistentes. Quando uma política de grupo se torna consistente, as alterações podem levar até 15 minutos adicionais para entrar em vigor devido ao armazenamento em cache. Por padrão, todas as atualizações feitas nas políticas de bucket são fortemente consistentes.

Conforme necessário, você pode alterar as garantias de consistência para atualizações de políticas de bucket. Por exemplo, você pode querer que uma alteração em uma política de bucket esteja disponível durante uma falha no local.

Nesse caso, você pode definir o `Consistency-Control` cabeçalho na solicitação PutBucketPolicy ou pode usar a solicitação de consistência PUT Bucket. Quando uma política de bucket se torna consistente, as alterações podem levar até 8 segundos adicionais para entrar em vigor devido ao armazenamento em cache.

> Se você definir a consistência para um valor diferente para contornar uma situação temporária, certifique-se de definir a configuração do bucket de volta ao valor original quando terminar. Caso contrário, todas as solicitações futuras ao bucket usarão a configuração modificada.

O que é política de sessão?

Uma política de sessão é uma política de acesso que restringe temporariamente as permissões disponíveis durante uma sessão específica, como quando um usuário assume um grupo. Uma política de sessão só pode

permitir um subconjunto de permissões e não pode conceder permissões adicionais. O próprio grupo pode ter permissões mais amplas.

Use o ARN em declarações de política

Nas declarações de política, o ARN é usado nos elementos Principal e Resource.

- Use esta sintaxe para especificar o ARN do recurso S3:

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

- Use esta sintaxe para especificar o ARN do recurso de identidade (usuários e 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
```

Outras considerações:

- Você pode usar o asterisco (*) como um caractere curinga para corresponder a zero ou mais caracteres dentro da chave do objeto.
- Caracteres internacionais, que podem ser especificados na chave do objeto, devem ser codificados usando JSON UTF-8 ou usando sequências de escape JSON \u. A codificação percentual não é suportada.

  "Sintaxe URN da RFC 2141"

  O corpo da requisição HTTP para a operação PutBucketPolicy deve ser codificado com charset=UTF-8.

Especifique recursos em uma política

Nas declarações de política, você pode usar o elemento Resource para especificar o bucket ou objeto para o qual as permissões são permitidas ou negadas.

- Cada declaração de política requer um elemento Resource. Em uma política, os recursos são indicados pelo elemento `Resource`, ou, alternativamente, `NotResource` para exclusão.
- Você especifica recursos com um ARN de recurso S3. Por exemplo:

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

- Você também pode usar variáveis de política dentro da chave do objeto. Por exemplo:

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

- O valor do recurso pode especificar um bucket que ainda não existe quando uma política de grupo é criada.

Especifique os principais em uma política

Utilize o elemento Principal para identificar o usuário, grupo ou conta de locatário que tem permissão/negação de acesso ao recurso pela declaração de política.

- Cada declaração de política em uma política de bucket deve incluir um elemento Principal. Declarações de política em uma política de grupo não precisam do elemento Principal porque o grupo é entendido como o principal.
- Em uma política, os principais são designados pelo elemento "Principal" ou, alternativamente, "NotPrincipal" para exclusão.
- As identidades baseadas em contas devem ser especificadas usando um ID ou um ARN:

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

- Este exemplo usa o ID da conta de locatário 27233906934684427525, que inclui a raiz da conta e todos os usuários da conta:

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

- Você pode especificar apenas a raiz da conta:

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

- Você pode especificar um usuário federado específico ("Alex"):

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

- Você pode especificar um grupo federado específico ("Managers"):

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

- Você pode especificar um principal anônimo:

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

- Para evitar ambiguidade, você pode usar o UUID do usuário em vez do nome de usuário:

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

  Por exemplo, suponha que Alex deixe a organização e o nome de usuário `Alex` seja excluído. Se um novo Alex entrar na organização e receber o mesmo `Alex` nome de usuário, o novo usuário poderá herdar involuntariamente as permissões concedidas ao usuário original.

- O valor principal pode especificar um nome de grupo/usuário que ainda não existe quando uma política de bucket é criada.

Especifique as permissões em uma política

Em uma política, o elemento Action é usado para permitir/negar permissões a um recurso. Há um conjunto de permissões que você pode especificar em uma política, que são indicadas pelo elemento "Action" ou, alternativamente, por "NotAction" para exclusão. Cada um desses elementos corresponde a operações específicas da API REST do S3.

A tabela lista as permissões que se aplicam aos buckets e as permissões que se aplicam aos objetos.

> Amazon S3 agora usa a permissão s3:PutReplicationConfiguration tanto para as ações PutBucketReplication e DeleteBucketReplication. StorageGRID usa permissões separadas para cada ação, o que corresponde à especificação original do Amazon S3.

> Uma exclusão é realizada quando uma operação de inserção (put) é usada para sobrescrever um valor existente.

Permissões que se aplicam aos buckets

Permissões	Operações da API REST do S3	Personalizado para StorageGRID
s3:CreateBucket	CreateBucket	Sim. **Nota**: use somente em políticas de grupo.
s3:DeleteBucket	DeleteBucket	
s3:DeleteBucketMetadataNotification	EXCLUIR configuração de notificação de metadados do bucket	Sim
s3:DeleteBucketPolicy	DeleteBucketPolicy	

Permissões	Operações da API REST do S3	Personalizado para StorageGRID
s3:DeleteReplicationConfiguration	DeleteBucketReplication	Sim, permissões separadas para PUT e DELETE
s3:GetBucketAcl	GetBucketAcl	
s3:GetBucketCompliance	GET conformidade do bucket (obsoleto)	Sim
s3:GetBucketConsistency	GET consistência do bucket	Sim
s3:GetBucketCORS	GetBucketCors	
s3:GetEncryptionConfiguration	GetBucketEncryption	
s3:GetBucketLastAccessTime	Obter o último tempo de acesso do bucket	Sim
s3:GetBucketLocation	GetBucketLocation	
s3:GetBucketMetadataNotification	GET Configuração de notificação de metadados do bucket	Sim
s3:GetBucketNotification	GetBucketNotificationConfiguration	
s3:GetBucketObjectLockConfiguration	GetObjectLockConfiguration	
s3:GetBucketPolicy	GetBucketPolicy	
s3:GetBucketTagging	GetBucketTagging	
s3:GetBucketVersioning	GetBucketVersioning	
s3:GetLifecycleConfiguration	GetBucketLifecycleConfiguration	
s3:GetReplicationConfiguration	GetBucketReplication	
s3:ListAllMyBuckets	• ListBuckets • GET Uso de Armazenamento	Sim, para GET Storage Usage. **Nota**: use somente em políticas de grupo.

Permissões	Operações da API REST do S3	Personalizado para StorageGRID
s3:ListBucket	• ListObjects • HeadBucket • RestoreObject	
s3:ListBucketMultipartUploads	• ListMultipartUploads • RestoreObject	
s3:ListBucketVersions	GET versões do Bucket	
s3:PutBucketCompliance	PUT Bucket compliance (obsoleto)	Sim
s3:PutBucketConsistency	Consistência de PUT Bucket	Sim
s3:PutBucketCORS	• DeleteBucketCors† • PutBucketCors	
s3:PutEncryptionConfiguration	• DeleteBucketEncryption • PutBucketEncryption	
s3:PutBucketLastAccessTime	Último tempo de acesso do PUT Bucket	Sim
s3:PutBucketMetadataNotification	PUT Configuração de notificação de metadados do bucket	Sim
s3:PutBucketNotification	PutBucketNotificationConfiguration	
s3:PutBucketObjectLockConfiguration	• CreateBucket com o `x-amz-bucket-object-lock-enabled: true` cabeçalho da solicitação (também requer a permissão s3:CreateBucket) • PutObjectLockConfiguration	
s3:PutBucketPolicy	PutBucketPolicy	
s3:PutBucketTagging	• DeleteBucketTagging† • PutBucketTagging	
s3:PutBucketVersioning	PutBucketVersioning	
s3:PutLifecycleConfiguration	• DeleteBucketLifecycle† • PutBucketLifecycleConfiguration	

Permissões	Operações da API REST do S3	Personalizado para StorageGRID
s3:PutReplicationConfiguration	PutBucketReplication	Sim, permissões separadas para PUT e DELETE

Permissões que se aplicam a objetos

Permissões	Operações da API REST do S3	Personalizado para StorageGRID
s3:AbortMultipartUpload	• AbortMultipartUpload • RestoreObject	
s3:BypassGovernanceRetention	• DeleteObject • DeleteObjects • PutObjectRetention	
s3:DeleteObject	• DeleteObject • DeleteObjects • RestoreObject	
s3:DeleteObjectTagging	DeleteObjectTagging	
s3:DeleteObjectVersionTagging	DeleteObjectTagging (uma versão específica do objeto)	
s3:DeleteObjectVersion	DeleteObject (uma versão específica do objeto)	
s3:GetObject	• GetObject • HeadObject • RestoreObject • SelectObjectContent	
s3:GetObjectAcl	GetObjectAcl	
s3:GetObjectLegalHold	GetObjectLegalHold	
s3:GetObjectRetention	GetObjectRetention	
s3:GetObjectTagging	GetObjectTagging	

Permissões	Operações da API REST do S3	Personalizado para StorageGRID
s3:GetObjectVersionTagging	GetObjectTagging (uma versão específica do objeto)	
s3:GetObjectVersion	GetObject (uma versão específica do objeto)	
s3:ListMultipartUploadParts	ListParts, RestoreObject	
s3:PutObject	• PutObject • CopyObject • RestoreObject • CreateMultipartUpload • CompleteMultipartUpload • UploadPart • UploadPartCopy	
s3:PutObjectLegalHold	PutObjectLegalHold	
s3:PutObjectRetention	PutObjectRetention	
s3:PutObjectTagging	PutObjectTagging	
s3:PutObjectVersionTagging	PutObjectTagging (uma versão específica do objeto)	
s3:PutOverwriteObject	• PutObject • CopyObject • PutObjectTagging • DeleteObjectTagging • CompleteMultipartUpload	Sim
s3:RestoreObject	RestoreObject	

Use a permissão PutOverwriteObject

A permissão s3:PutOverwriteObject é uma permissão personalizada do StorageGRID que se aplica a operações que criam ou atualizam objetos. A configuração dessa permissão determina se o cliente pode sobrescrever os dados de um objeto, metadados definidos pelo usuário ou as tags de objetos S3.

As configurações possíveis para essa permissão incluem:

- **Permitir**: O cliente pode sobrescrever um objeto. Esta é a configuração padrão.

- **Negar**: O cliente não pode sobrescrever um objeto. Quando definido como Negar, a permissão PutOverwriteObject funciona da seguinte forma:
  - Se um objeto existente for encontrado no mesmo caminho:
    - Os dados do objeto, metadados definidos pelo usuário ou as etiquetas do objeto S3 não podem ser sobrescritos.
    - Todas as operações de ingestão em andamento são canceladas e um erro é retornado.
    - Se o versionamento do S3 estiver ativado, a configuração Deny impede que as operações PutObjectTagging ou DeleteObjectTagging modifiquem o TagSet de um objeto e suas versões não atuais.
  - Se um objeto existente não for encontrado, esta permissão não terá efeito.
- Quando essa permissão não está presente, o efeito é o mesmo que se Allow estivesse definido.

> Se a política atual do S3 permitir sobrescrever e a permissão PutOverwriteObject estiver definida como Negar, o cliente não poderá sobrescrever os dados de um objeto, metadados definidos pelo usuário ou a marcação do objeto. Além disso, se a caixa de seleção **Impedir modificação pelo cliente** estiver marcada (**Configuração** > **Configurações de segurança** > **Rede e objetos**), essa configuração substituirá a configuração da permissão PutOverwriteObject.

Especificar condições em uma política

As condições definem quando uma política entrará em vigor. As condições consistem em operadores e pares de chave-valor.

As condições utilizam pares de chave-valor para avaliação. Um elemento de condição pode conter múltiplas condições, e cada condição pode conter múltiplos pares de chave-valor. O bloco de condição utiliza o seguinte formato:

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

No exemplo a seguir, a condição IpAddress usa a chave de condição SourceIp.

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

Operadores de condição suportados

Os operadores condicionais são categorizados da seguinte forma:

- String

- Numérico
- Booleano
- Endereço IP
- Verificação de valor nulo

Operadores de condição	Descrição
StringEquals	Compara uma chave com um valor de string com base na correspondência exata (diferencia maiúsculas de minúsculas).
StringNotEquals	Compara uma chave com um valor de string com base na correspondência negada (diferencia maiúsculas de minúsculas).
StringEqualsIgnoreCase	Compara uma chave com um valor de string com base em uma correspondência exata (ignora maiúsculas e minúsculas).
StringNotEqualsIgnoreCase	Compara uma chave com um valor de string com base na correspondência negada (ignora maiúsculas e minúsculas).
StringLike	Compara uma chave com um valor de string com base na correspondência exata (diferencia maiúsculas de minúsculas). Pode incluir os caracteres curinga * e ?.
StringNotLike	Compara uma chave com um valor de string com base na correspondência negada (diferencia maiúsculas de minúsculas). Pode incluir os caracteres curinga * e ?.
NumericEquals	Compara uma chave a um valor numérico com base em uma correspondência exata.
NumericNotEquals	Compara uma chave a um valor numérico com base na correspondência negada.
NumericGreaterThan	Compara uma chave a um valor numérico com base na correspondência "maior que".
NumericGreaterThanEquals	Compara uma chave a um valor numérico com base na correspondência "maior ou igual a".
NumericLessThan	Compara uma chave a um valor numérico com base na correspondência "menor que".
NumericLessThanEquals	Compara uma chave a um valor numérico com base na correspondência "menor ou igual a".
Bool	Compara uma chave a um valor booleano com base na correspondência "verdadeiro ou falso".

Operadores de condição	Descrição
IpAddress	Compara uma chave a um endereço IP ou intervalo de endereços IP.
NotIpAddress	Compara uma chave a um endereço IP ou intervalo de endereços IP com base na correspondência negada.
Nulo	Verifica se uma chave de condição está presente no contexto da solicitação atual.
IfExists	Adicionado a qualquer operador de condição, exceto a condição Null, para verificar a ausência da chave de condição. Retorna VERDADEIRO se a chave de condição não estiver presente.

Chaves de condição suportadas

Chaves de condição	Ações	Descrição
aws:SourceIp	Operadores de IP	Será comparado ao endereço IP de onde a solicitação foi enviada. Pode ser usado para operações de bucket ou objeto. **Nota:** Se a solicitação S3 foi enviada através do serviço Load Balancer nos nós Admin e Gateway, ela será comparada com o endereço IP upstream do serviço Load Balancer. **Nota**: Se um balanceador de carga de terceiro não transparente for utilizado, a comparação será feita com o endereço IP desse balanceador de carga. Qualquer `X-Forwarded-For` cabeçalho será ignorado, pois sua validade não pode ser verificada.
aws:username	Recurso/Identidade	Será comparado ao nome de usuário do remetente da solicitação. Pode ser usado para operações de bucket ou objeto.
s3:delimitador	s3:ListBucket e s3:ListBucketVersions permissões	Será comparado ao parâmetro delimitador especificado em uma solicitação ListObjects ou ListObjectVersions.

Chaves de condição	Ações	Descrição
s3:ExistingObjectTag/<tag-key>	s3:DeleteObjectTagging s3:DeleteObjectVersionTagging s3:GetObject s3:GetObjectAcl 3:GetObjectTagging s3:GetObjectVersion s3:GetObjectVersionAcl s3:GetObjectVersionTagging s3:PutObjectAcl s3:PutObjectTagging s3:PutObjectVersionAcl s3:PutObjectVersionTagging	Será necessário que o objeto existente possua a chave e o valor de tag específicos.
s3:max-keys	s3:ListBucket e s3:ListBucketVersions permissões	Será comparado ao parâmetro max-keys especificado em uma solicitação ListObjects ou ListObjectVersions.
s3:modo de bloqueio de objeto	s3:PutObject	Compara com o `object-lock-mode` expandido do cabeçalho da solicitação nas solicitações PutObject, CopyObject e CreateMultipartUpload.
s3:modo de bloqueio de objeto	s3:PutObjectRetention	Compara com a `object-lock-mode` expandida do corpo XML na solicitação PutObjectRetention.
s3:object-lock-remaining-retention-days	s3:PutObject	Compara com a data de retenção especificada no `x-amz-object-lock-retain-until-date`cabeçalho da solicitação ou calculada a partir do período de retenção padrão do bucket para garantir que esses valores estejam dentro do intervalo permitido para as seguintes solicitações: • PutObject • CopyObject • CreateMultipartUpload

Chaves de condição	Ações	Descrição
s3:object-lock-remaining-retention-days	s3:PutObjectRetention	Compara com a data de retenção especificada na solicitação PutObjectRetention para garantir que esteja dentro do intervalo permitido.
s3:prefix	s3:ListBucket e s3:ListBucketVersions permissões	Será comparado ao parâmetro de prefixo especificado em uma solicitação ListObjects ou ListObjectVersions.
s3:RequestObjectTag/<tag-key>	s3:PutObject s3:PutObjectTagging s3:PutObjectVersionTagging	Será necessário especificar uma chave e um valor de tag quando a solicitação do objeto incluir tagging.
s3:x-amz-server-side-encryption-customer-algorithm	s3:PutObject	Compara com o `sse-customer-algorithm` ou com o `copy-source-sse-customer-algorithm` expandido do cabeçalho da solicitação nas solicitações PutObject, CopyObject, CreateMultipartUpload, UploadPart, UploadPartCopy e CompleteMultipartUpload.

Especifique variáveis em uma política

Você pode usar variáveis em políticas para preencher informações de política quando elas estiverem disponíveis. Você pode usar variáveis de política no elemento `Resource` e em comparações de strings no elemento `Condition`.

Neste exemplo, a variável `${aws:username}` faz parte do elemento Resource:

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

Neste exemplo, a variável `${aws:username}` faz parte do valor da condição no bloco de condição:

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

Variável	Descrição
`${aws:SourceIp}`	Utiliza a chave SourceIp como a variável fornecida.
`${aws:username}`	Utiliza a chave de nome de usuário como a variável fornecida.
`${s3:prefix}`	Utiliza a chave de prefixo específica do serviço como variável fornecida.
`${s3:max-keys}`	Utiliza a chave max-keys específica do serviço como variável fornecida.
`${*}`	Caractere especial. Usa o caractere como um caractere literal *.
`${?}`	Caractere especial. Usa o caractere como um caractere literal "?"
`${$}`	Caractere especial. Usa o caractere como um caractere $ literal.

Criar políticas que exigem tratamento especial

Às vezes, uma política pode conceder permissões que são perigosas para a segurança ou para a continuidade das operações, como bloquear o usuário raiz da conta. A implementação da API REST do StorageGRID S3 é menos restritiva durante a validação da política do que a da Amazon, mas igualmente rigorosa durante a avaliação da política.

Descrição da política	Tipo de política	Comportamento da Amazon	Comportamento do StorageGRID
Negue a si mesmo quaisquer permissões para a conta root	Bucket	Válido e aplicado, mas a conta de usuário raiz mantém permissão para todas as operações de política do bucket S3	Mesmo
Negar a si mesmo quaisquer permissões para usuário/grupo	Grupo	Válido e em vigor	Mesmo
Permitir que um grupo de contas estrangeiras tenha qualquer permissão	Bucket	Principal inválido	Válido, mas as permissões para todas as operações de política do bucket S3 retornam um erro 405 Method Not Allowed quando permitidas por uma política

Descrição da política	Tipo de política	Comportamento da Amazon	Comportamento do StorageGRID
Permitir que o usuário raiz ou usuário de uma conta estrangeira tenha qualquer permissão	Bucket	Válido, mas as permissões para todas as operações de política do bucket S3 retornam um erro 405 Method Not Allowed quando permitidas por uma política	Mesmo
Permitir que todos tenham permissão para realizar todas as ações	Bucket	Válido, mas as permissões para todas as operações de política do bucket S3 retornam um erro 405 Method Not Allowed para o usuário raiz e usuários da conta externa	Mesmo
Negar permissões a todos para todas as ações	Bucket	Válido e aplicado, mas a conta de usuário raiz mantém permissão para todas as operações de política do bucket S3	Mesmo
Principal é um usuário ou grupo inexistente	Bucket	Principal inválido	Válido
O recurso é um bucket S3 inexistente	Grupo	Válido	Mesmo
Principal é um grupo local	Bucket	Principal inválido	Válido
A política concede permissões a contas que não são proprietárias (incluindo contas anônimas) para colocar objetos.	Bucket	Válido. Os objetos pertencem à conta do criador e a política do bucket não se aplica. A conta do criador deve conceder permissões de acesso ao objeto usando ACLs de objeto.	Válido. Os objetos pertencem à conta proprietária do bucket. A política do bucket se aplica.

Proteção WORM (gravação única e leitura múltipla)

Você pode criar buckets WORM (write-once-read-many) para proteger dados, metadados de objetos definidos pelo usuário e a marcação de objetos S3. Você configura os buckets WORM para permitir a criação de novos objetos e impedir a sobrescrita ou exclusão de conteúdo existente. Use uma das abordagens descritas aqui.

Para garantir que as sobrescritas sejam sempre negadas, você pode:

- No Grid Manager, acesse **Configuração** > **Segurança** > **Configurações de segurança** > **Rede e objetos** e selecione a caixa de seleção **Impedir modificação do cliente**.

- Aplique as seguintes regras e políticas do S3:
  - Adicione uma operação DENY PutOverwriteObject à política do S3.
  - Adicione uma operação DENY DeleteObject à política do S3.
  - Adicione uma operação ALLOW PutObject à política do S3.

> Configurar DeleteObject como NEGAR em uma política do S3 não impede que o ILM exclua objetos quando existe uma regra como "zero cópias após 30 dias".

> Mesmo quando todas essas regras e políticas são aplicadas, elas não impedem gravações simultâneas (veja a Situação A). Elas impedem sobrescritas sequenciais concluídas (veja a Situação B).

Situação A: Escritas simultâneas (sem proteção)

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

Situação B: Sobrescritas sequenciais concluídas (protegidas contra)

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

Informações relacionadas

- "Como as regras do StorageGRID ILM gerenciam objetos"
- "Exemplos de políticas de bucket"
- "Exemplos de políticas de grupo"
- "Exemplo de política de sessão"
- "Gerencie objetos com ILM"
- "Use uma conta de tenant"

Exemplo de política de sessão da API REST do StorageGRID S3

Utilize o exemplo a seguir para criar uma política de sessão do StorageGRID.

Exemplo: configure uma política de sessão que permita a recuperação de objetos

Neste exemplo, o usuário principal da sessão só tem permissão para recuperar objetos do bucket1. Todas as outras ações são implicitamente negadas, exceto as ações específicas do StorageGRID, como o uso da "s3:PutOverwriteObject" permissão. A política de sessão pode ser fornecida como um arquivo JSON ao chamar a AssumeRole API.

```
{
  "Statement": [
    {
      "Action": "s3:GetObject",
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::bucket1/*"
    }
  ]
}
```

Exemplos de políticas de bucket da API REST S3 do StorageGRID

Utilize os exemplos desta seção para criar políticas de acesso do StorageGRID para buckets.

As políticas de bucket especificam as permissões de acesso para o bucket ao qual a política está associada. Você configura uma política de bucket usando a API PutBucketPolicy do S3 por meio de uma destas ferramentas:

- "Tenant Manager".
- AWS CLI usando este comando (consulte "Operações em buckets"):

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

Exemplo: permitir que todos tenham acesso somente leitura a um bucket

Neste exemplo, todos, incluindo anônimos, têm permissão para listar objetos no bucket e realizar operações GetObject em todos os objetos no bucket. Todas as outras operações serão negadas. Observe que essa política pode não ser particularmente útil porque ninguém, exceto o usuário raiz, tem permissões para gravar no bucket.

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

Exemplo: permitir acesso total a todos em uma conta e acesso somente leitura a um bucket para todos em outra conta

Neste exemplo, todos em uma conta específica têm permissão para acesso total a um bucket, enquanto todos em outra conta específica têm permissão apenas para listar o bucket e realizar operações GetObject em objetos no bucket que começam com o prefixo da chave do objeto `shared/`.

> No StorageGRID, os objetos criados por uma conta que não seja a proprietária (incluindo contas anônimas) pertencem à conta proprietária do bucket. A política do bucket se aplica a esses 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/*"
        }
      }
    }
  ]
}
```

Exemplo: permitir acesso somente leitura a um bucket para todos e acesso total para um grupo específico

Neste exemplo, todos, incluindo usuários anônimos, têm permissão para listar o bucket e realizar GetObject operações em todos os objetos no bucket, enquanto apenas os usuários pertencentes ao grupo `Marketing` na conta especificada têm acesso total.

```
{
  "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/*"
      ]
    }
  ]
}
```

Exemplo: permitir acesso de leitura e gravação a um bucket para todos os clientes se estiverem no intervalo de IP

Neste exemplo, todos, incluindo usuários anônimos, têm permissão para listar o bucket e executar quaisquer operações de objeto em todos os objetos do bucket, desde que as solicitações venham de um intervalo de IP especificado (54.240.143.0 a 54.240.143.255, exceto 54.240.143.188). Todas as outras operações serão negadas e todas as solicitações fora do intervalo de IP serão negadas.

```
{
  "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"}
      }
    }
  ]
}
```

Exemplo: permitir acesso total a um bucket exclusivamente por um usuário federado específico

Neste exemplo, o usuário federado Alex tem permissão total de acesso ao `examplebucket` bucket e aos seus objetos. Todos os outros usuários, incluindo 'root', têm todas as operações explicitamente negadas. Observe, no entanto, que 'root' nunca tem as permissões de Put/Get/DeleteBucketPolicy negadas.

```
{
  "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/*"
      ]
    }
  ]
}
```

Exemplo: permissão PutOverwriteObject

Neste exemplo, o `Deny`Effect para PutOverwriteObject e DeleteObject garante que ninguém possa sobrescrever ou excluir os dados do objeto, os metadados definidos pelo usuário e a marcação do objeto 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/*"
    }
  ]
}
```

Exemplos de política de grupo da API REST do StorageGRID S3

Utilize os exemplos desta seção para criar políticas de acesso do StorageGRID para grupos.

As políticas de grupo especificam as permissões de acesso para o grupo ao qual a política está associada. Não há nenhum `Principal` elemento na política, pois ela é implícita. As políticas de grupo são configuradas usando o Tenant Manager ou a API.

Exemplo: defina a política de grupo usando o Tenant Manager

Ao adicionar ou editar um grupo no Gerenciador de Locatários, você pode selecionar uma política de grupo para determinar quais permissões de acesso ao S3 os membros desse grupo terão. Consulte "Crie grupos para um locatário do S3".

- **Sem acesso ao S3**: Opção padrão. Os usuários deste grupo não têm acesso aos recursos do S3, a menos que o acesso seja concedido por meio de uma política de bucket. Se você selecionar esta opção, somente o usuário raiz terá acesso aos recursos do S3 por padrão.
- **Acesso somente leitura**: Os usuários deste grupo têm acesso somente leitura aos recursos do S3. Por exemplo, os usuários deste grupo podem listar objetos e ler dados, metadados e tags dos objetos. Ao selecionar esta opção, a string JSON para uma política de grupo somente leitura aparece na caixa de texto. Você não pode editar esta string.
- **Acesso total**: Os usuários deste grupo têm acesso total aos recursos do S3, incluindo buckets. Ao selecionar esta opção, a string JSON para uma política de grupo de acesso total aparece na caixa de texto. Você não pode editar esta string.
- **Mitigação de ransomware**: Esta política de exemplo se aplica a todos os buckets deste locatário. Os usuários deste grupo podem executar ações comuns, mas não podem excluir permanentemente objetos de buckets que tenham o versionamento de objetos ativado.

  Os usuários do Tenant Manager que possuem a permissão Manage all buckets podem substituir esta política de grupo. Limite a permissão Manage all buckets a usuários confiáveis e utilize a autenticação multifator (MFA) quando disponível.

- **Personalizado**: Os usuários do grupo recebem as permissões que você especificar na caixa de texto.

Exemplo: permitir que o grupo tenha acesso total a todos os buckets

Neste exemplo, todos os membros do grupo têm permissão de acesso total a todos os buckets pertencentes à conta do tenant, a menos que seja explicitamente negado pela política do bucket.

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

Exemplo: permitir acesso somente leitura ao grupo em todos os buckets

Neste exemplo, todos os membros do grupo têm acesso somente leitura aos recursos do S3, a menos que seja explicitamente negado pela política do bucket. Por exemplo, os usuários deste grupo podem listar objetos e ler dados, metadados e tags dos objetos.

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

Exemplo: Permitir que os membros do grupo tenham acesso total apenas à sua "pasta" em um bucket

Neste exemplo, os membros do grupo só têm permissão para listar e acessar sua pasta específica (prefixo da chave) no bucket especificado. Observe que as permissões de acesso de outras políticas de grupo e da política do bucket devem ser consideradas ao determinar a privacidade dessas pastas.

```
{
  "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}/*"
    }
  ]
}
```

Operações S3 rastreadas nos logs de auditoria do StorageGRID

As mensagens de auditoria são geradas pelos serviços do StorageGRID e armazenadas em arquivos de log de texto. Você pode revisar as mensagens de auditoria específicas do S3 no log de auditoria para obter detalhes sobre as operações de bucket e objeto.

Operações de bucket rastreadas nos logs de auditoria

- CreateBucket
- DeleteBucket
- DeleteBucketTagging
- DeleteObjects
- GetBucketTagging
- HeadBucket
- ListObjects
- ListObjectVersions
- Conformidade de bucket PUT
- PutBucketTagging
- PutBucketVersioning

Operações de objetos registradas nos logs de auditoria

- CompleteMultipartUpload
- CopyObject
- DeleteObject
- GetObject
- HeadObject
- PutObject
- RestoreObject
- SelectObject
- UploadPart (quando uma regra ILM usa ingestão balanceada ou estrita)
- UploadPartCopy (quando uma regra ILM usa ingestão balanceada ou estrita)

Informações relacionadas

- "Acesse o arquivo de log de auditoria"
- "Cliente escreve mensagens de log de auditoria"
- "Mensagens de leitura do log de auditoria do cliente"

Informações sobre direitos autorais

Informações sobre marcas comerciais