使用S3 REST API

StorageGRID

NetApp
November 04, 2025

This PDF was generated from https://docs.netapp.com/zh-cn/storagegrid-118/s3/index.html on November 04, 2025. Always check docs.netapp.com for the latest.

目录

使用S3 REST API 1
S3 REST API支持的版本和更新 1
支持的版本 1
对S3 REST API支持进行了更新 1
快速参考：支持的S3 API请求 3
通用URI查询参数和请求标头 3
"AbortMultipartUpload" 4
"CompleteMultipartUpload" 4
"CopyObject" 4
"CreateBucket" 5
"CreateMultipartUpload" 5
"DeleteBucket" 6
"DeleteBucketCors" 6
"DeleteBucketEncryption" 6
"DeleteBucketLifecycle" 7
"DeleteBucketPolicy" 7
"DeleteBucketReplication" 7
"DeleteBucketTbaging" 7
"DeleteObject" 8
"DeleteObjects" 8
"DeleteObjectTagging" 8
"GetBucketAcl" 8
"GetBucketCors" 9
"GetBucketEncryption" 9
"GetBucketLifecycleConfiguration" 9
"GetBucketLocation" 9
"GetBucketNotizationConfiguration" 10
"GetBucketPolicy" 10
"GetBucketReplication" 10
"GetBucketTaging" 10
"GetBucketVersioning" 11
"GetObject" 11
"GetObjectAcl" 12
"GetObjectLegalHold" 12
"GetObjectLockConfiguration" 12
"GetObject保留" 12
"GetObjectTagging" 13
"HeadBucket" 13
"HeadObject" 13
"List桶" 14
"ListMultipartUploads" 14
"ListObjects" 14

"ListObjectsV2" 15
"ListObjectVersies" 15
"ListParts" 15
"PutBucketCors" 16
"PutBucketEncryption" 16
"PutBucketLifecycleConfiguration" 16
"PutBucketNotizationConfiguration" 17
"PutBucketPolicy" 18
"PutBucketReplication" 18
"PutBucketTaging" 18
"PutBucketVersioning" 19
"PutObject" 19
"PutObjectLegalHold" 20
"PutObjectLockConfiguration" 20
"PutObject保留" 20
"PutObjectTagging" 20
"RestorEObject" 21
"SelectObjectContent" 21
"上传部件" 21
"上传PartCopy" 21
测试S3 REST API配置 22
StorageGRID 如何实施 S3 REST API 23
客户端请求冲突 24
一致性值 24
对象版本控制 26
使用S3 REST API配置S3对象锁定 27
创建 S3 生命周期配置 32
实施 S3 REST API 的建议 36
支持Amazon S3 REST API 37
S3 REST API实施详细信息 37
对请求进行身份验证 38
对服务执行的操作 38
对存储分段执行的操作 39
对对象执行的操作 46
多部分上传操作 71
错误响应 78
StorageGRID自定义操作 80
StorageGRID自定义操作：概述 80
获取存储分段一致性 81
PUT 存储分段一致性 82
获取存储分段上次访问时间 83
PUT 分段上次访问时间 84
删除存储分段元数据通知配置 85

获取存储分段元数据通知配置 . . . 85
PUT 存储分段元数据通知配置 . . . 88
获取存储使用情况请求 . . . 93
已弃用旧合规性存储分段请求 . . . 94
存储分段和组访问策略 . . . 99
使用存储分段和组访问策略 . . . 99
存储分段策略示例 . . . 115
组策略示例 . . . 120
审核日志中跟踪的 S3 操作 . . . 123
审核日志中跟踪的存储分段操作 . . . 123
审核日志中跟踪的对象操作 . . . 124

使用S3 REST API

S3 REST API支持的版本和更新

StorageGRID 支持简单存储服务（ S3 ） API ，该 API 作为一组表示状态传输（ Representational State Transfer ， REST ） Web 服务来实施。

通过对S3 REST API的支持、您可以将为S3 Web服务开发的面向服务的应用程序与使用StorageGRID 系统的内部对象存储连接起来。需要对客户端应用程序当前使用S3 REST API调用的情况进行最小更改。

支持的版本

StorageGRID 支持以下特定版本的 S3 和 HTTP 。

项目	**version**
S3 API规范	"Amazon Web Services （AWS ）文档： Amazon Simple Storage Service API 参考"
HTTP	1.1 有关 HTTP 的详细信息，请参见 HTTP/1.1 （ RFC 7230-35 ）。 "IETF RFC 2616 ： 超文本传输协议（ HTTP/1.1 ）" • 注 * ： StorageGRID 不支持 HTTP/1.1 管道化。

对S3 REST API支持进行了更新

版本。	注释
11.8.	更新了S3操作的名称、以便与中使用的名称匹配 "Amazon Web Services （AWS ）文档： Amazon Simple Storage Service API 参考"。
11.7	• 已添加 "快速参考：支持的S3 API请求"。 • 增加了对将监管模式与S3对象锁定结合使用的支持。 • 增加了对特定于StorageGRID的支持 `x-ntap-sg-cgr-replication-status` GET对象和HEAD对象请求的响应标头。此标头可提供跨网格复制的对象复制状态。 • 现在，选择对象内容请求支持镶木图对象。

版本。	注释
11.6.	• 增加了对使用的支持 `partNumber` GET对象和HEAD对象请求中的Request参数。 • 增加了对 S3 对象锁定的默认保留模式和存储分段级别的默认保留期限的支持。 • 增加了对的支持 `s3:object-lock-remaining-retention-days` 策略条件键、用于设置对象允许的保留期限范围。 • 已将单个Put对象操作的最大_Recommended_大小更改为5 GiB (5、368、709、120字节)。如果对象大于 5 GiB ，请改用多部分上传。
11.5	• 增加了对管理存储分段加密的支持。 • 增加了对 S3 对象锁定和已弃用旧合规性请求的支持。 • 增加了对在版本控制的存储分段上使用删除多个对象的支持。 • 。 `Content-MD5` 现在已正确支持请求标头。
11.4	• 增加了对删除存储分段标记，获取存储分段标记和放置存储分段标记的支持。不支持成本分配标记。 • 对于在 StorageGRID 11.4 中创建的分段，不再需要限制对象密钥名称以满足性能最佳实践。 • 增加了对存储分段通知的支持 `s3:ObjectRestore:Post` 事件类型。 • 现在，多部件的 AWS 大小限制已强制实施。多部分上传中的每个部件必须介于 5 MiB 和 5 GiB 之间。最后一个部件可以小于 5 MiB 。 • 增加了对TLS 1.3的支持
11.3	• 增加了对使用客户提供的密钥（ SSI-C ）对对象数据进行服务器端加密的支持。 • 增加了对删除、获取和放置分段生命周期操作(仅限到期操作)和的支持 `x-amz-expiration` 响应标头。 • 更新了 PUT 对象， PUT 对象 - 复制和多部件上传，以说明在载入时使用同步放置的 ILM 规则的影响。 • 不再支持 TLS 1.1 密码。
11.2.	增加了对用于云存储池的后对象还原的支持。增加了对在组和存储分段策略中使用 AWS 语法来处理 ARN ，策略条件密钥和策略变量的支持。仍支持使用 StorageGRID 语法的现有组和存储分段策略。 • 注意： * 在其他配置 JSON/XML 中使用 ARN/URN 的情况没有改变，包括在自定义 StorageGRID 功能中使用的情况。
11.1	增加了对跨源站资源共享(CORS)、用于S3客户端连接到网格节点的HTTP以及分段合规性设置的支持。
11.0	增加了对为存储分段配置平台服务（ CloudMirror 复制，通知和 Elasticsearch 搜索集成）的支持。此外、还增加了对存储分段的对象标记位置限制以及可用一致性的支持。

版本。	注释
10.4.	增加了对版本控制，端点域名页面更新，策略中的条件和变量，策略示例以及 PutOverwriteObject 权限的 ILM 扫描更改的支持。
10.3	增加了对版本控制的支持。
10.2	增加了对组和存储分段访问策略以及多部件副本（上传部件 - 复制）的支持。
10.1	增加了对多部分上传，虚拟托管模式请求和 v4 身份验证的支持。
10.0	StorageGRID 系统最初支持 S3 REST API 。当前支持的 _Simple Storage Service API 参考版本为 2006-03-01 。

快速参考：支持的S3 API请求

此页面汇总了StorageGRID 如何支持Amazon Simple Storage Service (S3) API。

此页面仅包含StorageGRID 支持的S3操作。

要查看每个操作的AWS文档、请选择标题中的链接。

通用URI查询参数和请求标头

除非另有说明、否则支持以下通用URI查询参数：

- `versionId` (根据对象操作的需要)

除非另有说明、否则支持以下通用请求标头：

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

相关信息

- "S3 REST API实施详细信息"

- "Amazon Simple Storage Service API参考：通用请求标头"

"AbortMultipartUpload"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、加上此附加URI查询参数：

- `uploadId`

请求正文

无

StorageGRID 文档

"多部分上传操作"

"CompleteMultipartUpload"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、加上此附加URI查询参数：

- `uploadId`

请求正文**XML**标记

StorageGRID 支持以下请求正文XML标记：

- `CompleteMultipartUpload`
- `ETag`
- `Part`
- `PartNumber`

StorageGRID 文档

"CompleteMultipartUpload"

"CopyObject"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、加上以下附加标头：

- `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>`

请求正文

无

StorageGRID 文档

"CopyObject"

"CreateBucket"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、加上以下附加标头：

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

请求正文

StorageGRID 支持在实施时由Amazon S3 REST API定义的所有请求正文参数。

StorageGRID 文档

"对存储分段执行的操作"

"CreateMultipartUpload"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、加上以下附加标头：

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

- `Content-Language`
- `Expires`
- `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>`

请求正文

无

StorageGRID 文档

"CreateMultipartUpload"

"DeleteBucket"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

StorageGRID 文档

"对存储分段执行的操作"

"DeleteBucketCors"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"DeleteBucketEncryption"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"DeleteBucketLifecycle"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

- "对存储分段执行的操作"
- "创建 S3 生命周期配置"

"DeleteBucketPolicy"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"DeleteBucketReplication"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"DeleteBucketTbaging"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"DeleteObject"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、加上此附加请求标头：

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

请求正文

无

StorageGRID 文档

"对对象执行的操作"

"DeleteObjects"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、加上此附加请求标头：

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

请求正文

StorageGRID 支持在实施时由Amazon S3 REST API定义的所有请求正文参数。

StorageGRID 文档

"对对象执行的操作"

"DeleteObjectTagging"

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对对象执行的操作"

"GetBucketAcl"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"GetBucketCors"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"GetBucketEncryption"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"GetBucketLifecycleConfiguration"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

- "对存储分段执行的操作"
- "创建 S3 生命周期配置"

"GetBucketLocation"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"GetBucketNotizationConfiguration"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"GetBucketPolicy"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"GetBucketReplication"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"GetBucketTaging"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"GetBucketVersioning"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"GetObject"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、以及以下附加URI查询参数：

- `partNumber`
- `response-cache-control`
- `response-content-disposition`
- `response-content-encoding`
- `response-content-language`
- `response-content-type`
- `response-expires`

以及以下附加请求标头：

- `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`

请求正文

无

StorageGRID 文档

"GetObject"

"GetObjectAcl"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对对象执行的操作"

"GetObjectLegalHold"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"使用S3 REST API配置S3对象锁定"

"GetObjectLockConfiguration"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"使用S3 REST API配置S3对象锁定"

"GetObject保留"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"使用S3 REST API配置S3对象锁定"

"GetObjectTagging"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对对象执行的操作"

"HeadBucket"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"HeadObject"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、加上以下附加标头：

- `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`

请求正文

无

StorageGRID 文档

"HeadObject"

"List桶"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

无

StorageGRID 文档

"服务 上的操作"

"ListMultipartUploads"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、以及以下附加参数：

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

请求正文

无

StorageGRID 文档

"ListMultipartUploads"

"ListObjects"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、以及以下附加参数：

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

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"ListObjectsV2"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、以及以下附加参数：

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

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"ListObjectVersies"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、以及以下附加参数：

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

请求正文

无

StorageGRID 文档

"对存储分段执行的操作"

"ListParts"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、以及以下附加参数：

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

请求正文

无

StorageGRID 文档

"ListMultipartUploads"

"PutBucketCors"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

StorageGRID 支持在实施时由Amazon S3 REST API定义的所有请求正文参数。

StorageGRID 文档

"对存储分段执行的操作"

"PutBucketEncryption"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文**XML**标记

StorageGRID 支持以下请求正文XML标记：

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

StorageGRID 文档

"对存储分段执行的操作"

"PutBucketLifecycleConfiguration"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文**XML**标记

StorageGRID 支持以下请求正文XML标记：

- `And`

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

StorageGRID 文档

- "对存储分段执行的操作"
- "创建 S3 生命周期配置"

"PutBucketNotizationConfiguration"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文**XML**标记

StorageGRID 支持以下请求正文XML标记：

- `Event`
- `Filter`
- `FilterRule`
- `Id`
- `Name`
- `NotificationConfiguration`
- `Prefix`
- `S3Key`
- `Suffix`
- `Topic`

- `TopicConfiguration`
- `Value`

StorageGRID 文档

"对存储分段执行的操作"

"PutBucketPolicy"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

有关支持的JSON正文字段的详细信息、请参见 "使用存储分段和组访问策略"。

"PutBucketReplication"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文**XML**标记

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

StorageGRID 文档

"对存储分段执行的操作"

"PutBucketTaging"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

StorageGRID 支持在实施时由Amazon S3 REST API定义的所有请求正文参数。

StorageGRID 文档

"对存储分段执行的操作"

"PutBucketVersioning"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文参数

StorageGRID 支持以下请求正文参数：

- `VersioningConfiguration`
- `Status`

StorageGRID 文档

"对存储分段执行的操作"

"PutObject"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、加上以下附加标头：

- `Cache-Control`
- `Content-Disposition`
- `Content-Encoding`
- `Content-Language`
- `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>`

请求正文

- 对象的二进制数据

StorageGRID 文档

"PutObject"

"PutObjectLegalHold"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

StorageGRID 支持在实施时由Amazon S3 REST API定义的所有请求正文参数。

StorageGRID 文档

"使用S3 REST API配置S3对象锁定"

"PutObjectLockConfiguration"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

StorageGRID 支持在实施时由Amazon S3 REST API定义的所有请求正文参数。

StorageGRID 文档

"使用S3 REST API配置S3对象锁定"

"PutObject保留"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、加上以下附加标题：

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

请求正文

StorageGRID 支持在实施时由Amazon S3 REST API定义的所有请求正文参数。

StorageGRID 文档

"使用S3 REST API配置S3对象锁定"

"PutObjectTagging"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

StorageGRID 支持在实施时由Amazon S3 REST API定义的所有请求正文参数。

StorageGRID 文档

"对对象执行的操作"

"RestorEObject"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

有关支持的正文字段的详细信息、请参见 "RestorEObject"。

"SelectObjectContent"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 。

请求正文

有关支持的正文字段的详细信息、请参见以下内容：

- "使用 S3 Select"
- "SelectObjectContent"

"上传部件"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、以及以下附加URI查询参数：

- `partNumber`
- `uploadId`

以及以下附加请求标头：

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

请求正文

- 零件的二进制数据

StorageGRID 文档

"上传部件"

"上传PartCopy"

URI查询参数和请求标头

StorageGRID 支持所有 通用参数和标头 对于此请求、以及以下附加URI查询参数：

- `partNumber`
- `uploadId`

以及以下附加请求标头：

- `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`

请求正文

无

StorageGRID 文档

"上传PartCopy"

测试S3 REST API配置

您可以使用Amazon Web Services命令行界面(AWS CLI)测试与系统的连接、并验证是否可以读取和写入对象。

开始之前

- 您已从下载并安装 AWS 命令行界面 "aws.amazon.com/cli"。
- 您也可以选择 "已创建负载平衡器端点"。否则、您知道要连接到的存储节点的IP地址以及要使用的端口号。请参见 "客户端连接的IP地址和端口"。
- 您已拥有 "已创建S3租户帐户"。
- 您已登录到租户和 "已创建访问密钥"。

有关这些步骤的详细信息、请参见 "配置客户端连接"。

步骤

1. 配置AWS命令行界面设置以使用您在StorageGRID 系统中创建的帐户：
   a. 进入配置模式： `aws configure`
   b. 输入您创建的帐户的访问密钥ID。
   c. 输入您创建的帐户的机密访问密钥。

d. 输入要使用的默认区域。例如： `us-east-1`。

e. 输入要使用的默认输出格式，或者按 * 输入 * 选择 JSON 。

2. 创建存储分段。

   此示例假设您已将负载平衡器端点配置为使用IP地址10.96.101.17和端口10443。

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

   如果已成功创建存储分段，则会返回存储分段的位置，如以下示例所示：

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

3. 上传对象。

   ```
   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
   ```

   如果对象上传成功，则返回一个 Etag ，该 Etag 是对象数据的哈希。

4. 列出存储分段的内容以验证是否已上传此对象。

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

5. 删除对象。

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

6. 删除存储分段。

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

StorageGRID 如何实施 S3 REST API

客户端请求冲突

冲突的客户端请求（例如，两个客户端写入同一密钥）将以 " 最新成功 " 为基础进行解决。

" 最新赢单 " 评估的时间取决于 StorageGRID 系统何时完成给定请求，而不是 S3 客户端何时开始操作。

一致性值

一致性可在不同存储节点和站点之间的对象可用性与这些对象的一致性之间实现平衡。您可以根据应用程序的要求更改一致性。

默认情况下， StorageGRID 保证新创建的对象的写入后读一致性。成功完成 PUT 后的任何 GET 都将能够读取新写入的数据。对现有对象的覆盖，元数据更新和删除最终保持一致。覆盖通常需要几秒钟或几分钟才能传播，但可能需要长达 15 天的时间。

如果要以不同的一致性执行对象操作、您可以：

- 指定的一致性 每个存储分段。
- 指定的一致性 每个API操作。
- 通过执行以下任务之一更改默认的网格范围一致性：
  - 在网格管理器中，转至*configuration*>*System*>*Storage settings **>*Default s**一致 性。
  - 。

> 对网格范围一致性的更改仅适用于在更改设置后创建的分段。要确定更改的详细信息、请参见位于的审核日志 `/var/local/log` (搜索*consencyLevel*)。

一致性值

一致性会影响StorageGRID用于跟踪对象的元数据在节点之间的分布方式、从而影响对象对客户端请求的可用性。

您可以将存储分段或API操作的一致性设置为以下值之一：

- **all**：所有节点都会立即接收数据，否则请求将失败。
- 强-全局：保证所有站点中所有客户端请求的写入后读一致性。
- 强站点：保证站点内所有客户端请求的写入后读一致性。
- **read-after-new-write**：(默认)为新对象提供写后读一致性、并最终为对象更新提供一致性。提供高可用性和数据保护保证。建议用于大多数情况。
- 可用：为新对象和对象更新提供最终一致性。对于S3存储分段、请仅在需要时使用(例如、对于包含很少读取的日志值的存储分段、或者对于不存在的密钥执行HEAD或GET操作)。S3 FabricPool 存储分段不支持。

使用"新写后读取"和"可用"一致性

如果head或get操作使用"新写入后读取"一致性、则StorageGRID会通过多个步骤执行查找、如下所示：

- 它首先使用低一致性查找对象。
- 如果此查找失败、它将在下一个一致性值处重复此查找、直到达到与强全局行为等效的一致性为止。

如果head或get操作使用"新写入后读取"一致性、但对象不存在、则对象查找将始终达到与强全局行为等效的一致性。由于这种一致性要求每个站点上都有多个对象元数据副本、因此、如果同一站点上的两个或更多存储节点不可用、您可能会收到大量500个内部服务器错误。

除非您需要与Amazon S3类似的一致性保证、否则可以通过将一致性设置为"available "来防止HEAD和GET操作出现这些错误。 如果head或get操作使用"可用"一致性、则StorageGRID仅提供最终的一致性。它不会在一致性提高时重试失败的操作、因此不需要提供对象元数据的多个副本。

[[API-operation]]指定API操作的一致性

要为单个API操作设置一致性、此操作必须支持一致性值、并且必须在请求标头中指定一致性。此示例将GetObject操作的一致性设置为"strong-site"。

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

对于PutObject和GetObject操作、必须使用相同的一致性。

指定存储分段的一致性

要设置存储分段的一致性、可以使用StorageGRID "PUT 存储分段一致性" 请求。您也可以 "更改存储分段的一致性" 从租户管理器。

设置存储分段的一致性时、请注意以下事项：

- 设置存储分段的一致性可确定对存储分段或存储分段配置中的对象执行的S3操作所使用的一致性。它不会影响存储分段本身的操作。
- 单个API操作的一致性会覆盖存储分段的一致性。
- 通常、分段应使用默认一致性"read-after-new-write"。 如果请求无法正常工作、请尽可能更改应用程序客户端的行为。或者、将客户端配置为为为每个API请求指定一致性。只能在最后一种方法下、在存储分段级别设置一致性。

[[how－sistic-controls-and-ilm-뭐 聶-interact]]如何通过一致性和ILM规则交互来影响数据保护

您选择的一致性和ILM规则都会影响对象的保护方式。这些设置可以进行交互。

例如、存储对象时使用的一致性会影响对象元数据的初始放置、而为ILM规则选择的加载行为会影响对象副本的初始放置。由于StorageGRID需要同时访问对象的元数据及其数据才能满足客户端请求、因此为一致性和载入行为选择匹配的保护级别可以提供更好的初始数据保护、并提高系统响应的可预测性。

以下内容 "加热选项" 可用于ILM规则：

双提交

StorageGRID会立即创建对象的临时副本、并将成功结果返回给客户端。如果可能，将创建 ILM 规则中指定的副本。

严格

必须先创建ILM规则中指定的所有副本、然后才能将成功返回到客户端。

平衡

StorageGRID会在加载时尝试创建ILM规则中指定的所有副本；如果无法创建、则会创建临时副本、并将成功结果返回给客户端。在可能的情况下，将创建 ILM 规则中指定的副本。

一致性规则和**ILM**规则如何交互的示例

假设您有一个双站点网格、该网格具有以下ILM规则、并且具有以下一致性：

- * ILM 规则 * ：创建两个对象副本，一个在本地站点，一个在远程站点。使用严格的加热行为。
- 一致性：强全局(对象元数据立即分发到所有站点)。

当客户端将对象存储到网格时， StorageGRID 会创建两个对象副本并将元数据分发到两个站点，然后再向客户端返回成功。

在载入成功消息时，此对象将受到完全保护，不会丢失。例如，如果本地站点在载入后不久丢失，则远程站点上仍存在对象数据和对象元数据的副本。此对象完全可检索。

如果您改用相同的ILM规则和强站点一致性、则在将对象数据复制到远程站点之后、在远程站点分发对象元数据之前、客户端可能会收到一条成功消息。在这种情况下，对象元数据的保护级别与对象数据的保护级别不匹配。如果本地站点在载入后不久丢失，则对象元数据将丢失。无法检索此对象。

一致性和ILM规则之间的相互关系可能很复杂。如果需要帮助、请联系NetApp。

对象版本控制

如果要保留每个对象的多个版本、可以设置分段的版本控制状态。为分段启用版本控制有助于防止意外删除对象、并可用于检索和还原对象的早期版本。

StorageGRID 系统实施版本控制，并支持大多数功能，但存在一些限制。StorageGRID 最多支持 1 ， 000 个对象版本。

对象版本控制可以与 StorageGRID 信息生命周期管理（ ILM ）或 S3 存储分段生命周期配置结合使用。您必须明确为每个存储分段启用版本控制。为分段启用版本控制后、添加到分段的每个对象都会分配一个版本ID、该ID由StorageGRID系统生成。

不支持使用 MFA （多因素身份验证） Delete 。

只能在使用 StorageGRID 10.3 或更高版本创建的存储分段上启用版本控制。

ILM 和版本控制

ILM 策略将应用于对象的每个版本。ILM 扫描过程会持续扫描所有对象，并根据当前 ILM 策略重新评估这些对象。对 ILM 策略所做的任何更改都会应用于先前载入的所有对象。如果启用了版本控制，则包括先前载入的版

本。ILM 扫描会将新的 ILM 更改应用于先前输入的对象。

对于启用了版本控制的分段中的S3对象、版本控制支持允许您创建使用"非当前时间"作为参考时间的ILM规则(对于问题"仅将此规则应用于较早对象版本？"、请选择*是* 在中 "创建ILM规则向导的第1步"）。对象更新后，其先前版本将变为非最新版本。通过使用"非当前时间"筛选器、您可以创建可减少先前版本对象对存储的影响的策略。

> 使用多部分上传操作上传新版本的对象时，原始版本对象的非当前时间反映为新版本创建多部分上传的时间，而不是多部分上传完成的时间。在有限情况下，原始版本的非当前时间可能比当前版本的时间早数小时或数天。

相关信息

- "如何删除受版本控制的 S3 对象"
- "S3 版本对象的 ILM 规则和策略（示例 4 ）"。

使用S3 REST API配置S3对象锁定

如果为StorageGRID 系统启用了全局S3对象锁定设置、则可以在启用S3对象锁定的情况下创建分段。您可以为每个存储分段指定默认保留、也可以为每个对象版本指定保留设置。

如何为存储分段启用S3对象锁定

如果为 StorageGRID 系统启用了全局 S3 对象锁定设置，则可以选择在创建每个分段时启用 S3 对象锁定。

S3对象锁定是一种永久性设置、只有在创建存储分段时才能启用。创建分段后、您无法添加或禁用S3对象锁定。

要为存储分段启用S3对象锁定、请使用以下方法之一：

- 使用租户管理器创建存储分段。请参见 "创建 S3 存储分段"。
- 通过使用CreateBucket.创建存储分段 `x-amz-bucket-object-lock-enabled` 请求标题。请参见 "对存储分段执行的操作"。

S3对象锁定需要分段版本控制、创建分段时会自动启用此功能。您不能暂停分段的版本控制。请参见 "对象版本控制"。

存储分段的默认保留设置

为存储分段启用S3对象锁定后、您可以选择为存储分段启用默认保留、并指定默认保留模式和默认保留期限。

默认保留模式

- 在合规模式下：
  - 在达到保留截止日期之前、无法删除此对象。
  - 对象的保留截止日期可以增加、但不能减少。
  - 在达到该日期之前、无法删除对象的保留截止日期。
- 在监管模式下：

- 使用的用户 `s3:BypassGovernanceRetention` 权限可以使用 `x-amz-bypass-governance-retention: true` 请求标头以绕过保留设置。
- 这些用户可以在达到保留截止日期之前删除对象版本。
- 这些用户可以增加、减少或删除对象的保留截止日期。

默认保留期限

每个存储分段都可以指定默认保留期限(以年或天为单位)。

如何设置存储分段的默认保留

要设置存储分段的默认保留时间、请使用以下方法之一：

- 通过租户管理器管理存储分段设置。请参见 "创建 S3 存储区。" 和 "更新S3对象锁定默认保留"。
- 问题描述存储分段的PutObjectLockConfiguration请求、用于指定默认模式和默认天数或年数。

PutObjectLockConfiguration

通过PutObjectLockConfiguration请求、您可以设置和修改启用了S3对象锁定的存储分段的默认保留模式和默认保留期限。您还可以删除先前配置的默认保留设置。

如果将新对象版本写入存储分段、则会应用默认保留模式 `x-amz-object-lock-mode` 和 `x-amz-object-lock-retain-until-date` 未指定。默认保留期限用于计算保留截止日期IF `x-amz-object-lock-retain-until-date` 未指定。

如果在载入对象版本后修改了默认保留期限，则对象版本的保留日期将保持不变，不会使用新的默认保留期限重新计算。

您必须具有 `s3:PutBucketObjectLockConfiguration` 权限、或者作为帐户root用户来完成此操作。

。 `Content-MD5` 必须在Put请求中指定请求标头。

请求示例

此示例为存储分段启用S3对象锁定、并将默认保留模式设置为合规、将默认保留期限设置为6年。

```
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>
```

如何确定存储分段的默认保留

要确定是否为存储分段启用了S3对象锁定并查看默认保留模式和保留期限、请使用以下方法之一：

- 在租户管理器中查看存储分段。请参见 "查看S3存储分段"。
- 问题描述一个GetObjectLockConfiguration请求。

GetObjectLockConfiguration

通过GetObjectLockConfiguration请求、您可以确定是否为存储分段启用了S3对象锁定、如果已启用、则查看是否为存储分段配置了默认保留模式和保留期限。

如果将新对象版本写入存储分段、则会应用默认保留模式 `x-amz-object-lock-mode` 未指定。默认保留期限用于计算保留截止日期IF `x-amz-object-lock-retain-until-date` 未指定。

您必须具有 `s3:GetBucketObjectLockConfiguration` 权限、或者作为帐户root用户来完成此操作。

请求示例

```
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
```

响应示例

```
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>
```

如何指定对象的保留设置

启用了S3对象锁定的存储分段可以包含具有和不具有S3对象锁定保留设置的对象组合。

对象级保留设置可通过S3 REST API来指定。对象的保留设置将覆盖存储分段的任何默认保留设置。

您可以为每个对象指定以下设置：

- 保留模式：合规性或监管。
- *retain－until－date *：指定StorageGRID 必须保留对象版本多长时间的日期。
  - 在合规模式下、如果保留截止日期为未来日期、则可以检索对象、但无法修改或删除它。保留截止日期可以增加、但不能减少或删除此日期。
  - 在监管模式下、具有特殊权限的用户可以绕过保留截止日期设置。他们可以在对象版本的保留期限到期

之前将其删除。它们还可以增加、减少甚至删除保留截止日期。

- * 合法保留 * ：对对象版本应用合法保留时，会立即锁定该对象。例如，您可能需要对与调查或法律争议相关的对象进行法律保留。合法保留没有到期日期，但在明确删除之前始终有效。

  对象的合法保留设置与保留模式和保留截止日期无关。如果某个对象版本处于合法保留状态、则任何人都无法删除该版本。

要在向存储分段添加对象版本时指定S3对象锁定设置、请使用问题描述 A "PutObject"， "CopyObject"或 "CreateMultipartUpload" 请求。

您可以使用以下命令：

- `x-amz-object-lock-mode`，可以是合规性或监管(区分大小写)。

  > 如果指定 `x-amz-object-lock-mode`、您还必须指定 `x-amz-object-lock-retain-until-date`。

- `x-amz-object-lock-retain-until-date`
  - 保留截止日期值必须采用格式 `2020-08-10T21:46:00Z`。允许使用小数秒，但仅保留 3 位小数（精确度为毫秒）。不允许使用其他ISO 8601格式。
  - 保留截止日期必须为未来日期。
- `x-amz-object-lock-legal-hold`

  如果处于合法保留状态（区分大小写），则对象将置于合法保留状态。如果关闭了合法保留，则不会进行合法保留。任何其他值都会导致 400 错误请求（ InvalidArgument ）错误。

如果您使用上述任一请求标头，请注意以下限制：

- 。 `Content-MD5` 如果有、则请求标头为必填项 `x-amz-object-lock-*` PutObject请求中存在请求标头。 `Content-MD5` 对于CopyObject或CreateMultipartUpload不是必需项。
- 如果存储分段未启用S3对象锁定和 `x-amz-object-lock-*` 存在请求标头、返回400错误请求(InvalidRequest)错误。
- PutObject请求支持使用 `x-amz-storage-class: REDUCED_REDUNDANCY` 以匹配AWS行为。但是，如果在启用了 S3 对象锁定的情况下将对象载入存储分段，则 StorageGRID 将始终执行双提交载入。
- 后续的GET或HeadObject版本响应将包括标题 `x-amz-object-lock-mode`， `x-amz-object-lock-retain-until-date`，和 `x-amz-object-lock-legal-hold`(如果已配置)以及请求发送方是否正确 `s3:Get*` 权限。

您可以使用 `s3:object-lock-remaining-retention-days` 策略条件关键字、用于限制对象允许的最短和最长保留期限。

如何更新对象的保留设置

如果需要更新现有对象版本的合法保留或保留设置，可以执行以下对象子资源操作：

- `PutObjectLegalHold`

如果新的合法保留值为 on ，则对象将置于合法保留状态。如果合法保留值为 off ，则取消合法保留。

- `PutObjectRetention`
  - 模式值可以是合规性或监管(区分大小写)。
  - 保留截止日期值必须采用格式 `2020-08-10T21:46:00Z`。允许使用小数秒，但仅保留 3 位小数（精确度为毫秒）。不允许使用其他ISO 8601格式。
  - 如果对象版本具有现有的保留日期，则只能增加此保留日期。新的价值必须是未来的。

如何使用监管模式

拥有的用户 `s3:BypassGovernanceRetention` 权限可以绕过使用监管模式的对象的活动保留设置。任何删除或PutObject保留 操作都必须包含 `x-amz-bypass-governance-retention:true` 请求标题。这些用户可以执行以下附加操作：

- 执行DeleteObject或DeleteObjects操作以在对象保留期限到期之前删除该对象版本。

  无法删除处于合法保留状态的对象。合法保留必须关闭。

- 执行PutObject锟 刻 操作、以便在对象的保留期限结束之前将对象版本的模式从监管更改为合规。

  绝不允许将模式从合规性更改为监管。

- 执行PutObject锟 刻 操作以增加、减少或删除对象版本的保留期限。

相关信息

- "使用 S3 对象锁定管理对象"
- "使用S3对象锁定保留对象"
- "《 Amazon Simple Storage Service 用户指南：使用 S3 对象锁定》"

创建 **S3** 生命周期配置

您可以创建 S3 生命周期配置，以控制何时从 StorageGRID 系统中删除特定对象。

本节中的简单示例说明了 S3 生命周期配置如何控制从特定 S3 存储分段中删除（过期）某些对象的时间。本节中的示例仅供说明。有关创建 S3 生命周期配置的完整详细信息，请参见 "《Amazon Simple Storage Service用户指南：对象生命周期管理》"。请注意， StorageGRID 仅支持到期操作，不支持过渡操作。

什么是生命周期配置

生命周期配置是一组应用于特定 S3 分段中的对象的规则。每个规则都指定受影响的对象以及这些对象的到期时间（在特定日期或一定天数后）。

StorageGRID 在一个生命周期配置中最多支持 1 ， 000 条生命周期规则。每个规则可以包含以下 XML 元素：

- 到期日期：从对象载入开始，在达到指定日期或达到指定天数时删除对象。
- NoncurrentVersionExpiration ：从对象变为非最新状态开始，在达到指定天数时删除对象。
- 筛选器（前缀，标记）

- Status
- ID

每个对象都遵循S3存储分段生命周期或ILM策略的保留设置。配置S3存储分段生命周期后、对于与存储分段生命周期筛选器匹配的对象、生命周期到期操作将覆盖ILM策略。与存储分段生命周期筛选器不匹配的对象将使用ILM策略的保留设置。如果某个对象与存储分段生命周期筛选器匹配、并且未明确指定到期操作、则不会使用ILM策略的保留设置、这意味着对象版本将永久保留。请参见 "S3存储分段生命周期和ILM策略的优先级示例"。

因此，即使 ILM 规则中的放置说明仍适用于某个对象，该对象也可能会从网格中删除。或者，即使对象的任何 ILM 放置指令已失效，该对象也可能会保留在网格中。有关详细信息，请参见 "ILM 如何在对象的整个生命周期内运行"。

> 存储分段生命周期配置可用于启用了 S3 对象锁定的存储分段，但旧版合规存储分段不支持存储分段生命周期配置。

StorageGRID 支持使用以下存储分段操作来管理生命周期配置：

- DeleteBucketLifecycle
- GetBucketLifecycleConfiguration
- PutBucketLifecycleConfiguration

创建生命周期配置

作为创建生命周期配置的第一步，您需要创建一个包含一个或多个规则的 JSON 文件。例如，此 JSON 文件包含三个规则，如下所示：

1. 规则1仅适用于与前缀匹配的对象 `category1/`并且具有 `key2` 的值 `tag2`。。 `Expiration` 参数指定与筛选器匹配的对象将在2020年8月22日午夜到期。
2. 规则2仅适用于与前缀匹配的对象 `category2/`。。 `Expiration` 参数指定与筛选器匹配的对象将在载入后100天过期。

   > 指定天数的规则与对象的载入时间相关。如果当前日期超过载入日期加上天数，则在应用生命周期配置后，可能会立即从存储分段中删除某些对象。

3. 规则3仅适用于与前缀匹配的对象 `category3/`。。 `Expiration` 参数指定任何非最新版本的匹配对象将在其变为非最新状态50天后过期。

```
{
    "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"
        }
    ]
}
```

将生命周期配置应用于存储分段

创建生命周期配置文件后、您可以通过发出PutBucketLifecycleConfiguration请求将其应用于存储分段。

此请求会将示例文件中的生命周期配置应用于名为的存储分段中的对象 `testbucket`。

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

要验证生命周期配置是否已成功应用于存储分段、请发送问题描述a GetBucketLifecycleConfiguration请求。例如：

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

成功的响应将列出您刚刚应用的生命周期配置。

验证存储分段生命周期到期适用场景 对象

在发出PutObject、HeadObject或GetObject请求时、您可以确定生命周期配置适用场景中的到期规则是否为特定对象。如果规则适用、响应将包括 `Expiration` 此参数用于指示对象何时到期以及匹配的到期规则。

> 由于存储分段生命周期会覆盖ILM、因此 `expiry-date` 显示的是删除对象的实际日期。有关详细信息，请参见 "如何确定对象保留"。

例如、此PutObject请求是在2020年6月22日发出的、并将对象放置在中 `testbucket` 存储分段。

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

成功响应表示此对象将在 100 天后（ 2020 年 10 月 1 日）过期，并且与生命周期配置的规则 2 匹配。

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

例如、此HeadObject请求用于获取testb分 段中同一对象的元数据。

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

成功响应包括对象的元数据，并指示对象将在 100 天后过期，并且与规则 2 匹配。

```
{
      "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": {}
}
```

对于启用了版本控制的分段、 `x-amz-expiration` 响应标头仅适用于当前版本的对象。

实施 S3 REST API 的建议

在实施用于 StorageGRID 的 S3 REST API 时，应遵循以下建议。

针对不存在的对象的建议

如果您的应用程序定期检查某个对象是否位于您不希望该对象实际存在的路径上、则应使用"可用" "一致性"。例如、如果您的应用程序在放置之前指向某个位置、则应使用"可用"一致性。

否则、如果HEAD操作未找到对象、则在同一站点上的两个或更多存储节点不可用或某个远程站点不可访问时、您可能会收到大量500个内部服务器错误。

您可以使用为每个存储分段设置"可用"一致性 "PUT 存储分段一致性" 请求、也可以在单个API操作的请求标头中指定一致性。

对象密钥建议

根据首次创建分段的时间、请遵循这些对象键名建议。

在StorageGRID 11.4或更早版本中创建的分段

- 不要使用随机值作为对象键的前四个字符。这与 AWS 以前针对密钥前缀的建议不同。请改用非随机、非唯一的前缀、例如 `image`。
- 如果按照以前的AWS建议在密钥前缀中使用随机和唯一字符、请在对象密钥前添加目录名称。也就是说，请使用以下格式：

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

而不是以下格式：

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

在**StorageGRID 11.4**或更高版本中创建的分段

不需要限制对象密钥名称以满足性能最佳实践。在大多数情况下、对象密钥名称的前四个字符可以使用随机值。

> 但S3工作负载例外、它会在短时间后持续删除所有对象。为了最大限度地降低此使用情形对性能的影响、请每隔数千个对象更改一次密钥名称的前导部分、并使用日期之类的内容。例如、假设S3客户端通常每秒写入2、000个对象、而ILM或存储分段生命周期策略将在三天后删除所有对象。为了最大限度地降低对性能的影响、您可以使用如下模式命名密钥：
> `/mybucket/mydir/yyyymmddhhmmss-random_UUID.jpg`

"范围读取"建议

如果 "用于压缩存储对象的全局选项" 已启用、则S3客户端应用程序应避免执行指定要返回的字节数范围的GetObject操作。这些"范围读取"操作效率低下、因为StorageGRID必须有效地解压缩对象才能访问请求的字节。从非常大的对象请求少量字节的GetObject操作效率特别低；例如、从50 GB压缩对象读取10 MB的范围是效率低下的。

如果从压缩对象读取范围，则客户端请求可能会超时。

> 如果需要压缩对象，并且客户端应用程序必须使用范围读取，请增加应用程序的读取超时时间。

支持Amazon S3 REST API

S3 REST API实施详细信息

StorageGRID 系统实施简单存储服务 API （ API 版本 2006-03-01 ），支持大多数操作，但有一些限制。在集成 S3 REST API 客户端应用程序时，您需要了解实施详细信息。

StorageGRID 系统既支持虚拟托管模式请求，也支持路径模式请求。

日期处理

S3 REST API 的 StorageGRID 实施仅支持有效的 HTTP 日期格式。

对于接受日期值的任何标头， StorageGRID 系统仅支持有效的 HTTP 日期格式。日期的时间部分可以使用格林威治标准时间（ GMT ）格式或通用协调时间（ UTC ）格式指定，并且不存在时区偏移（必须指定 +0000 ）。如果包括 `x-amz-date` 标题中指定的任何值。使用AWS签名版本4时、将显示 `x-amz-date` 签名请求中必须存在标题、因为不支持日期标题。

通用请求标头

StorageGRID 系统支持定义的通用请求标头 "Amazon Simple Storage Service API参考：通用请求标头"，但有一个例外。

请求标题	实施
Authorization	完全支持 AWS 签名版本 2 支持 AWS 签名版本 4 ，但以下情况除外： • 不会为请求正文计算 SHA256 值。接受用户提交的值而不进行验证、就像该值一样 `UNSIGNED-PAYLOAD` 已为提供 `x-amz-content-sha256` 标题。
X-AMZ-securation-token	未实施。返回 `XNotImplemented`。

通用响应标头

StorageGRID 系统支持由 _Simple Storage Service API 参考 _ 定义的所有通用响应标头，但有一个例外。

响应标头	实施
X-AMZ-ID-2	未使用

对请求进行身份验证

StorageGRID 系统支持使用 S3 API 对对象进行身份验证和匿名访问。

S3 API 支持签名版本 2 和签名版本 4 对 S3 API 请求进行身份验证。

经过身份验证的请求必须使用您的访问密钥 ID 和机密访问密钥进行签名。

StorageGRID 系统支持两种身份验证方法：HTTP `Authorization` 标题和使用查询参数。

使用 **HTTP** 授权标头

HTTP `Authorization` 标头由所有S3 API操作使用、但在存储分段策略允许的情况下使用匿名请求除外。。`Authorization` 标头包含对请求进行身份验证所需的所有签名信息。

使用查询参数

您可以使用查询参数向 URL 添加身份验证信息。这称为对 URL 进行预签名，可用于授予对特定资源的临时访问权限。具有预先签名URL的用户无需知道访问资源的机密访问密钥、这样您就可以为资源提供第三方受限访问权限。

对服务执行的操作

StorageGRID 系统支持对该服务执行以下操作。

操作	实施
List桶 (以前称为GET服务)	在所有 Amazon S3 REST API 行为下实施。如有更改、恕不另行通知。
获取存储使用量	StorageGRID "获取存储使用量" Request (请求)用于告知您帐户使用的总存储量以及与帐户关联的每个存储分段的存储量。这是对服务执行的操作、路径为/、并具有自定义查询参数 (`?x-ntap-sg-usage`)。
选项 /	客户端应用程序可以使用问题描述 `OPTIONS /` 向存储节点上的S3端口发出请求、但不提供S3身份验证凭据、以确定存储节点是否可用。您可以使用此请求进行监控，也可以允许外部负载平衡器确定存储节点何时关闭。

对存储分段执行的操作

对于每个 S3 租户帐户， StorageGRID 系统最多支持 1 ， 000 个分段。

存储分段名称限制遵循AWS US Standard区域限制、但您应进一步将其限制为DNS命名约定、以支持S3虚拟托管模式请求。

有关详细信息，请参见以下内容：

- "《Amazon Simple Storage Service用户指南：存储分段限制》"
- "配置S3端点域名"

ListObjects (GET Bucket)和ListObjectVersies (GET Bucket)对象版本)操作支持StorageGRID "一致性值"。

您可以检查是否已为各个存储分段启用上次访问时间更新。请参见 "获取存储分段上次访问时间"。

下表介绍了 StorageGRID 如何实施 S3 REST API 存储分段操作。要执行其中任何操作，必须为帐户提供必要的访问凭据。

操作	实施
CreateBucket	创建新存储分段。创建存储分段后，您就会成为存储分段所有者。 • 存储分段名称必须符合以下规则： ◦ 每个 StorageGRID 系统必须是唯一的（而不仅仅是租户帐户中的唯一）。 ◦ 必须符合 DNS 要求。 ◦ 必须至少包含 3 个字符，并且不能超过 63 个字符。 ◦ 可以是一个或多个标签的序列，并使用一个句点分隔相邻标签。每个标签必须以小写字母或数字开头和结尾，并且只能使用小写字母，数字和连字符。 ◦ 不能与文本格式的 IP 地址类似。 ◦ 不应在虚拟托管模式请求中使用句点。句点会在验证服务器通配符证书时出现发生原因 问题。 • 默认情况下、将在中创建分段 `us-east-1` 区域；但是、您可以使用 `LocationConstraint` 请求正文中的请求元素以指定其他区域。使用时 `LocationConstraint` Element中、您必须指定已使用网格管理器或网格管理API定义的区域的确切名称。如果您不知道应使用的区域名称、请联系您的系统管理员。 注意：如果CreateBucket(创建存储分段)请求使用的区域尚未在StorageGRID中定义，则会发生错误。 • 您可以包括 `x-amz-bucket-object-lock-enabled` 请求标题以创建启用了S3对象锁定的存储分段。请参见 "使用S3 REST API配置S3对象锁定"。 创建存储分段时，必须启用 S3 对象锁定。创建分段后、您无法添加或禁用S3对象锁定。S3 对象锁定需要分段版本控制，在创建分段时会自动启用分段版本控制。
DeleteBucket	删除存储分段。
DeleteBucketCors	删除存储分段的CORS配置。
DeleteBucketEncryption	从存储分段中删除默认加密。现有加密对象将保持加密状态、但添加到存储分段的任何新对象不会加密。
DeleteBucketLifecycle	从存储分段中删除生命周期配置。请参见 "创建 S3 生命周期配置"。
DeleteBucketPolicy	删除附加到存储分段的策略。
DeleteBucketReplication	删除附加到存储分段的复制配置。

操作	实施
DeleteBucketTbaging	使用 `tagging` 用于从存储分段中删除所有标记的子资源。 注意：如果为此存储分段设置了非默认ILM策略标记、则会出现 `NTAP-SG-ILM-BUCKET-TAG` 具有分配给存储分段的值的存储分段标记。如果存在、请勿问题描述一个DeleteBucketTbagingRequest `NTAP-SG-ILM-BUCKET-TAG` 存储分段标签。而是使用问题描述发出仅包含的PutBucketTagingRequest `NTAP-SG-ILM-BUCKET-TAG` 用于从存储分段中删除所有其他标记的标记及其分配值。请勿修改或删除 `NTAP-SG-ILM-BUCKET-TAG` 存储分段标签。
GetBucketAcl	返回肯定响应以及存储分段所有者的ID、DisplayName和权限、指示所有者对存储分段具有完全访问权限。
GetBucketCors	返回 `cors` 存储分段的配置。
GetBucketEncryption	返回存储分段的默认加密配置。
GetBucketLifecycleConfiguration (以前称为GET分段生命周期)	返回存储分段的生命周期配置。请参见 "创建 S3 生命周期配置"。
GetBucketLocation	返回使用设置的区域 `LocationConstraint` CreateBucket.如果存储分段的区域为 `us-east-1`、将返回该区域的空字符串。
GetBucketNotizationConfiguration (以前称为GET分段通知)	返回附加到存储分段的通知配置。
GetBucketPolicy	返回附加到存储分段的策略。
GetBucketReplication	返回附加到存储分段的复制配置。
GetBucketTaging	使用 `tagging` 用于返回存储分段的所有标记的子资源。 注意：如果为此存储分段设置了非默认ILM策略标记、则会出现 `NTAP-SG-ILM-BUCKET-TAG` 具有分配给存储分段的值的存储分段标记。请勿修改或删除此标记。

操作	实施
GetBucketVersioning	此实施使用 `versioning` 用于返回存储分段版本控制状态的子资源。 • *blank*：从未启用版本控制(分段已"取消版本控制") • Enabled ： 已启用版本控制 • suspended ： 先前已启用版本控制并已暂停
GetObjectLockConfiguration	返回存储分段默认保留模式和默认保留期限(如果已配置)。 请参见 "使用S3 REST API配置S3对象锁定"。
HeadBucket	确定存储分段是否存在、以及您是否有权访问该存储分段。 此操作将返回： • `x-ntap-sg-bucket-id`：UUID格式的存储分段的UUID。 • `x-ntap-sg-trace-id`：关联请求的唯一跟踪ID。
ListObjects和ListObjectsV2 (以前称为GET分段)	返回分段中的部分或全部对象(最多1、000个)。对象的存储类可以具有两个值之一、即使对象是随一起载入的 `REDUCED_REDUNDANCY` 存储类选项： • `STANDARD`、表示对象存储在由存储节点组成的存储池中。 • `GLACIER`、表示对象已移至云存储池指定的外部存储分段。 如果存储分段包含大量前缀相同的已删除密钥、则响应可能包括一些密钥 `CommonPrefixes` 不包含密钥。
ListObjectVersies (以前称为Get BucketObject Version)	在存储分段上具有读取访问权限时、将此操作与结合使用 `versions` 子资源列出了存储分段中所有版本对象的元数据。
PutBucketCors	设置存储分段的CORS配置、以便存储分段可以处理跨源站请求。跨源资源共享（ CORS ）是一种安全机制，允许一个域中的客户端 Web 应用程序访问不同域中的资源。例如、假设您使用名为的S3存储分段 `images` 以存储图形。通过设置的CORS配置 `images` 存储分段中的图像、您可以在网站上显示该存储分段中的图像 `http://www.example.com`。
PutBucketEncryption	设置现有存储分段的默认加密状态。启用存储分段级别加密后，添加到存储分段中的任何新对象都会进行加密。 StorageGRID 支持使用 StorageGRID 管理的密钥进行服务器端加密。指定服务器端加密配置规则时、请设置 `SSEAlgorithm` 参数设置为 `AES256`，并且不要使用 `KMSMasterKeyID` 参数。 如果对象上传请求已指定加密(即、如果请求包含)、则存储分段默认加密配置将被忽略 `x-amz-server-side-encryption-*` 请求标题)。

操作	实施
PutBucketLifecycleConfiguration (以前称为"放置分段生命周期")	为存储分段创建新的生命周期配置或替换现有生命周期配置。StorageGRID 在一个生命周期配置中最多支持 1 ， 000 条生命周期规则。每个规则可以包含以下 XML 元素： • 到期日期(天数、日期、ExpireObjectDeleteMarker) • 非当前版本到期(新非当前版本、非当前日期) • 筛选器（前缀，标记） • Status • ID StorageGRID 不支持以下操作： • AbortIncompleteMultipartUpload • 过渡 请参见 "创建 S3 生命周期配置"。要了解存储分段生命周期中的到期操作如何与ILM放置说明交互、请参见 "ILM 如何在对象的整个生命周期内运行"。 • 注 * ：存储分段生命周期配置可用于启用了 S3 对象锁定的存储分段，但传统合规存储分段不支持存储分段生命周期配置。

操作	实施

PutBucketNotizationConfiguration
 	
(以前称为Put Bucket"通知)
	使用请求正文中包含的通知配置XML配置分段的通知。您应了解以下实施详细信息：
	
	• StorageGRID支持将Amazon Simple Notification Service (Amazon SNS)或Kafka主题作为目标。不支持简单队列服务(SQS)或Amazon Lamba端点。
	• 必须将通知目标指定为 StorageGRID 端点的 URN 。可以使用租户管理器或租户管理 API 创建端点。
	
	要成功配置通知，端点必须存在。如果端点不存在、则为 `400 Bad Request` 返回错误并显示代码 `InvalidArgument`。
	
	• 您不能为以下事件类型配置通知。这些事件类型 * 不 * 受支持。
	
	◦ `s3:ReducedRedundancyLostObject`
	
	◦ `s3:ObjectRestore:Completed`
	
	• 从StorageGRID 发送的事件通知使用标准JSON格式、不同之处在于它们不包含某些密钥、而对其他密钥使用特定值、如以下列表所示：
	
	◦ * 事件源 *
	
	`sgws:s3`
	
	◦ * awsRegion*
	
	不包括
	
	◦ * 。 x-AMZ-id-2*
	
	不包括
	
	◦ * arn*
	
	`urn:sgws:s3:::bucket_name`

PutBucketPolicy	设置附加到存储分段的策略。请参见 "使用存储分段和组访问策略"。

操作	实施
PutBucketReplication	配置 "StorageGRID CloudMirror复制" 使用请求正文中提供的复制配置XML的存储分段。对于 CloudMirror 复制，您应了解以下实施详细信息： • StorageGRID 仅支持复制配置的 V1 。这意味着、StorageGRID 不支持使用 `Filter` Element中的规则、并遵循V1中有关删除对象版本的约定。有关详细信息，请参见 "《Amazon Simple Storage Service用户指南：复制配置》"。 • 分段复制可以在分版本或未分版本的分段上配置。 • 您可以在复制配置 XML 的每个规则中指定不同的目标存储分段。一个源存储分段可以复制到多个目标存储分段。 • 必须将目标分段指定为租户管理器或租户管理 API 中指定的 StorageGRID 端点的 URN 。请参见 "配置 CloudMirror 复制"。 要成功进行复制配置，必须存在此端点。如果端点不存在、则请求将以失败的形式出现 `400 Bad Request`。错误消息显示： `Unable to save the replication policy. The specified endpoint URN does not exist: URN.` • 您无需指定 `Role` 在配置XML中。StorageGRID 不使用此值，如果提交，则会忽略此值。 • 如果在配置XML中省略存储类、则StorageGRID 将使用 `STANDARD` 默认情况下、存储类。 • 如果从源存储分段中删除对象或删除源存储分段本身，则跨区域复制行为如下： ◦ 如果在复制对象或存储分段之前将其删除、则不会复制该对象或存储分段、也不会通知您。 ◦ 如果您在复制对象或存储分段后将其删除，则 StorageGRID 会对跨区域复制的 V1 遵循标准 Amazon S3 删除行为。
PutBucketTaging	使用 `tagging` 用于为存储分段添加或更新一组标记的子资源。添加存储分段标记时，请注意以下限制： • StorageGRID 和 Amazon S3 为每个存储分段最多支持 50 个标签。 • 与存储分段关联的标记必须具有唯一的标记密钥。一个标记密钥的长度最多可包含 128 个 Unicode 字符。 • 标记值的长度最多可以为 256 个 Unicode 字符。 • 密钥和值区分大小写。 注意：如果为此存储分段设置了非默认ILM策略标记、则会出现 `NTAP-SG-ILM-BUCKET-TAG` 具有分配给存储分段的值的存储分段标记。确保 `NTAP-SG-ILM-BUCKET-TAG` 存储分段标记包含在所有PutBucketTag请求中的已分配值中。请勿修改或删除此标记。 注意：此操作将覆盖存储分段已有的任何当前标记。如果在集合中省略了任何现有标记、则会删除存储分段中的这些标记。

操作	实施
PutBucketVersioning	使用 `versioning` 用于设置现有存储分段的版本控制状态的子资源。您可以使用以下值之一设置版本控制状态： • Enabled ： 为存储分段中的对象启用版本控制。添加到存储分段中的所有对象都会收到唯一的版本 ID 。 • suspended ： 为存储分段中的对象禁用版本控制。添加到存储分段中的所有对象都会收到版本ID `null`。
PutObjectLockConfiguration	配置或删除存储分段默认保留模式和默认保留期限。 如果修改了默认保留期限，则现有对象版本的保留日期将保持不变，不会使用新的默认保留期限重新计算。 请参见 "使用S3 REST API配置S3对象锁定" 了解详细信息。

对对象执行的操作

对对象执行的操作

本节介绍 StorageGRID 系统如何对对象实施 S3 REST API 操作。

以下条件适用于所有对象操作：

- StorageGRID "一致性值" 支持对对象执行的所有操作，但以下操作除外：
  - GetObjectAcl
  - `OPTIONS /`
  - PutObjectLegalHold
  - PutObject保留
  - SelectObjectContent
- 冲突的客户端请求（例如，两个客户端写入同一密钥）将以 " 最新成功 " 为基础进行解决。" 最新赢单 " 评估的时间取决于 StorageGRID 系统何时完成给定请求，而不是 S3 客户端何时开始操作。
- StorageGRID 存储分段中的所有对象均归存储分段所有者所有，包括由匿名用户或其他帐户创建的对象。
- 通过Swift加热到StorageGRID 系统的数据对象无法通过S3进行访问。

下表介绍了 StorageGRID 如何实施 S3 REST API 对象操作。

操作	实施
DeleteObject	多因素身份验证(MFA)和响应标头 `x-amz-mfa` 不受支持。 处理DeleteObject请求时、StorageGRID会尝试立即从所有存储位置删除对象的所有副本。如果成功， StorageGRID 会立即向客户端返回响应。如果无法在30秒内删除所有副本(例如、由于某个位置暂时不可用)、则StorageGRID 会将这些副本排队等待删除、然后向客户端指示删除成功。 版本控制 要删除特定版本、请求者必须是存储分段所有者并使用 `versionId` 子资源。使用此子资源将永久删除此版本。如果 `versionId` 对应于删除标记、即响应标头 `x-amz-delete-marker` 返回时设置为 `true`。 • 删除对象时不使用 `versionId` 子资源在已启用版本的存储分段上、将生成删除标记。。 `versionId` 对于删除标记、使用返回 `x-amz-version-id` 响应标头和 `x-amz-delete-marker` 返回的响应标头设置为 `true`。 • 删除对象时不使用 `versionId` 子资源在版本暂停的分段上、它会永久删除已存在的"null"版本或"null"删除标记、并生成新的"null"删除标记。。 `x-amz-delete-marker` 返回的响应标头设置为 `true`。 • 注意 * ：在某些情况下，一个对象可能存在多个删除标记。 请参见 "使用S3 REST API配置S3对象锁定" 了解如何在监管模式下删除对象版本。
DeleteObjects (以前称为删除多个对象)	多因素身份验证(MFA)和响应标头 `x-amz-mfa` 不受支持。 可以在同一请求消息中删除多个对象。 请参见 "使用S3 REST API配置S3对象锁定" 了解如何在监管模式下删除对象版本。
DeleteObjectTagging	使用 `tagging` 用于从对象中删除所有标记的子资源。 版本控制 如果 `versionId` 请求中未指定查询参数、此操作将从受版本控制的存储分段中的对象的最新版本中删除所有标记。如果对象的当前版本是删除标记、则会返回"NDotAllowed"状态 `x-amz-delete-marker` 响应标头设置为 `true`。
GetObject	"GetObject"

操作	实施
GetObjectAcl	如果为帐户提供了必要的访问凭据，则此操作将返回肯定响应以及对象所有者的 ID ， DisplayName 和权限，指示所有者对对象具有完全访问权限。
GetObjectLegalHold	"使用S3 REST API配置S3对象锁定"
GetObject保留	"使用S3 REST API配置S3对象锁定"
GetObjectTagging	使用 `tagging` 子资源以返回对象的所有标记。 版本控制 如果 `versionId` 请求中未指定查询参数、此操作将返回受版本控制的存储分段中对象的最新版本中的所有标记。如果对象的当前版本是删除标记、则会返回"NDotAllowed"状态 `x-amz-delete-marker` 响应标头设置为 `true`。
HeadObject	"HeadObject"
RestorEObject	"RestorEObject"
PutObject	"PutObject"
CopyObject (以前称为Put Object - Copy)	"CopyObject"
PutObjectLegalHold	"使用S3 REST API配置S3对象锁定"
PutObject保留	"使用S3 REST API配置S3对象锁定"

操作	实施
PutObjectTagging	使用 `tagging` 用于向现有对象添加一组标记的子资源。 对象标记限制 您可以在上传新对象时为其添加标记，也可以将其添加到现有对象中。StorageGRID 和 Amazon S3 对每个对象最多支持 10 个标记。与对象关联的标记必须具有唯一的标记密钥。一个标记密钥的长度最多可以是 128 个 Unicode 字符，而标记值的长度最多可以是 256 个 Unicode 字符。密钥和值区分大小写。 标记更新和加热行为 使用PutObjectTags更新对象的标记时、StorageGRID不会重新加载对象。这意味着不会使用匹配 ILM 规则中指定的 " 载入行为 " 选项。通过正常后台 ILM 进程重新评估 ILM 时，更新触发的任何对象放置更改都会进行。 这意味着、如果ILM规则使用stricting选项执行加数据操作、则在无法放置所需对象(例如、新需要的位置不可用)时不会执行任何操作。更新后的对象会保留其当前位置，直到可以进行所需的位置为止。 解决冲突 冲突的客户端请求（例如，两个客户端写入同一密钥）将以 " 最新成功 " 为基础进行解决。" 最新赢单 " 评估的时间取决于 StorageGRID 系统何时完成给定请求，而不是 S3 客户端何时开始操作。 版本控制 如果 `versionId` 未在此请求中指定查询参数、此操作会将标记添加到受版本控制的存储分段中的对象的最新版本。如果对象的当前版本是删除标记、则会返回"NDotAllowed"状态 `x-amz-delete-marker` 响应标头设置为 `true`。
SelectObjectContent	"SelectObjectContent"

使用 **S3 Select**

StorageGRID 支持的以下Amazon S3 Select子句、数据类型和运算符 "SelectObjectContent 命令"。

不支持未列出的任何项目。

有关语法，请参见 "SelectObjectContent"。有关 S3 Select 的详细信息，请参见 "适用于 S3 Select 的 AWS 文档"。

只有启用了 S3 Select 的租户帐户才能进行问题描述 SelectObjectContent 查询。请参见 "使用 S3 Select 的注意事项和要求"。

条款

- 选择列表
- from 子句
- Where 子句
- Limit 子句

数据类型

- 池
- 整型
- string
- 浮点
- 小数点，数字
- timestamp

运算符

逻辑运算符

- 和
- 不是
- 或

比较运算符

- <
- >
- &lt； =
- &gt； =
- =
- =
- <>
- ！ =
- 介于之间
- 在中

模式匹配运算符

- 例如
- _
- %

统一运算符

- 为空
- 不为空

数学运算符

- +
- -
- *
- /
- %

StorageGRID 遵循Amazon S3 Select操作员优先级。

聚合函数

- 平均（）
- 计数（ * ）
- 最大值（）
- 最小值（）
- sum（ ）

条件函数

- 案例
- 合并
- NULLIF

转换函数

- cast （用于受支持的数据类型）

date 函数

- 日期添加
- 日期差异
- 提取
- to_string
- to_timestamp
- UTCNOW

字符串函数

- char_length ， character_length

- 更低
- 子字符串
- 剪切
- 上限

使用服务器端加密

服务器端加密可用于保护空闲对象数据。StorageGRID 会在写入对象时对数据进行加密，并在您访问对象时对数据进行解密。

如果要使用服务器端加密，可以根据加密密钥的管理方式从两个互斥选项中选择任一选项：

- *SSE （使用 StorageGRID 管理的密钥进行服务器端加密） *：在问题描述 S3 请求以存储对象时，StorageGRID 会使用唯一密钥对对象进行加密。在问题描述 S3 请求以检索对象时， StorageGRID 会使用存储的密钥对对象进行解密。
- * SSI-C （使用客户提供的密钥进行服务器端加密） *：在问题描述 S3 请求以存储对象时，您可以提供自己的加密密钥。检索对象时，您可以在请求中提供相同的加密密钥。如果这两个加密密钥匹配，则会对对象进行解密，并返回您的对象数据。

  虽然 StorageGRID 负责管理所有对象加密和解密操作，但您必须管理提供的加密密钥。

  > 您提供的加密密钥永远不会存储。如果丢失加密密钥，则会丢失相应的对象。

  > 如果使用 SSE 或 SSI-C 对对象进行加密，则会忽略任何分段级别或网格级别的加密设置。

使用 **SS**

要使用 StorageGRID 管理的唯一密钥对对象进行加密，请使用以下请求标头：

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

以下对象操作支持此命令头：

- "PutObject"
- "CopyObject"
- "CreateMultipartUpload"

使用 **SSI-C**

要使用您管理的唯一密钥对对象进行加密，请使用三个请求标头：

请求标题	**Description**
`x-amz-server-side-encryption-customer-algorithm`	指定加密算法。标题值必须为 `AES256`。

请求标题	Description
`x-amz-server-side-encryption-customer-key`	指定用于对对象进行加密或解密的加密密钥。密钥的值必须为 256 位 base64 编码。
`x-amz-server-side-encryption-customer-key-MD5`	根据 RFC 1321 指定加密密钥的 MD5 摘要，用于确保加密密钥的传输没有错误。MD5 摘要的值必须为 base64 编码的 128 位。

以下对象操作支持 SSI-C 请求标头：

- "GetObject"
- "HeadObject"
- "PutObject"
- "CopyObject"
- "CreateMultipartUpload"
- "上传部件"
- "上传PartCopy"

将服务器端加密与客户提供的密钥（ **SSI-C** ）结合使用的注意事项

在使用 SSI-C 之前，请注意以下注意事项：

- 必须使用 https 。

> 使用 SSI-C 时， StorageGRID 会拒绝通过 http 发出的任何请求出于安全考虑，您应考虑使用 http 意外发送的任何密钥受到损坏。丢弃该密钥，并根据需要旋转。

- 响应中的 ETag 不是对象数据的 MD5 。
- 您必须管理加密密钥到对象的映射。StorageGRID 不存储加密密钥。您负责跟踪为每个对象提供的加密密钥。
- 如果您的存储分段已启用版本控制，则每个对象版本都应具有自己的加密密钥。您负责跟踪每个对象版本使用的加密密钥。
- 由于您在客户端上管理加密密钥，因此您还必须在客户端上管理任何其他保护措施，例如密钥轮换。

> 您提供的加密密钥永远不会存储。如果丢失加密密钥，则会丢失相应的对象。

- 如果为存储分段配置了跨网格复制或CloudMirror复制、则无法加载SSE-C对象。载入操作将失败。

相关信息

"Amazon S3用户指南：使用客户提供的密钥进行服务器端加密(SSE-C)"

CopyObject

您可以使用S3 CopyObject请求为已存储在S3中的对象创建副本。CopyObject操作与依次

执行GetObject和PutObject相同。

解决冲突

冲突的客户端请求（例如，两个客户端写入同一密钥）将以 " 最新成功 " 为基础进行解决。" 最新赢单 " 评估的时间取决于 StorageGRID 系统何时完成给定请求，而不是 S3 客户端何时开始操作。

对象大小

一个PutObject操作的最大_Recommended_大小为5 GiB (5、368、709、120字节)。如果您的对象大于5 GiB、请使用 "多部分上传" 而是。

一个PutObject操作的最大_supported_大小为5 TiB (5、497、555、138、880字节)。

> 如果您从StorageGRID 11.5或更早版本升级、则在尝试上传超过5 GiB的对象时、将触发S3 Put Object Size Too Liger警报。如果您全新安装了StorageGRID 11.7或11.7、则在这种情况下不会触发警报。但是、为了符合AWS S3标准、未来版本的StorageGRID不支持上传超过5 GiB的对象。

用户元数据中的 **UTF-8** 字符

如果某个请求在用户定义的元数据的密钥名称或值中包含（未转义） UTF-8 值，则会未定义 StorageGRID 行为。

StorageGRID 不会解析或解释用户定义的元数据的密钥名称或值中包含的转义 UTF-8 字符。转义的 UTF-8 字符被视为 ASCII 字符：

- 如果用户定义的元数据包含转义的 UTF-8 字符，则请求将成功。
- StorageGRID 不会返回 `x-amz-missing-meta` 如果对密钥名称或值的解释值包含不可打印的字符、则为标题。

支持的请求标头

支持以下请求标头：

- `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-`、后跟一个名称-值对、该对包含用户定义的元数据
- `x-amz-metadata-directive`：默认值为 `COPY`、用于复制对象和关联的元数据。

  您可以指定 `REPLACE` 复制对象时覆盖现有元数据、或者更新对象元数据。

- `x-amz-storage-class`
- `x-amz-tagging-directive`：默认值为 `COPY`、用于复制对象和所有标记。

您可以指定 REPLACE 可在复制对象时覆盖现有标记、或更新标记。

- S3 对象锁定请求标头：
  - x-amz-object-lock-mode
  - x-amz-object-lock-retain-until-date
  - x-amz-object-lock-legal-hold

    如果在发出请求时没有这些标头、则会使用存储分段默认保留设置来计算对象版本模式和保留截止日期。请参见 "使用S3 REST API配置S3对象锁定"。

- SSA 请求标头：
  - 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

    请参见 [服务器端加密的请求标头]

请求标头不受支持

不支持以下请求标头：

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

存储类选项

。 x-amz-storage-class 支持请求标头、如果匹配的ILM规则使用"双提交"或"已平衡"、则会影响StorageGRID创建的对象副本数 ""INGest"选项"。

- STANDARD

  （默认）指定在 ILM 规则使用双提交选项或 balanced-option 回退到创建中间副本时执行双提交载入操作。

- REDUCED_REDUNDANCY

  指定在 ILM 规则使用双提交选项或 balanced-option 回退为创建中间副本时执行单提交载入操作。

如果要在启用了S3对象锁定的情况下将对象载入存储分段、则会显示 `REDUCED_REDUNDANCY` 选项将被忽略。如果要将对象载入旧的合规存储分段、则会显示 `REDUCED_REDUNDANCY` 选项返回错误。StorageGRID 将始终执行双提交载入，以确保满足合规性要求。

在CopyObject中使用x-AMZ-copy-source

如果源存储分段和密钥、请在中指定 `x-amz-copy-source` 标头与目标分段和密钥不同、源对象数据的副本将写入目标。

如果源和目标匹配、则使用和 `x-amz-metadata-directive` 标头指定为 `REPLACE`、对象的元数据将使用请求中提供的元数据值进行更新。在这种情况下， StorageGRID 不会重新载入对象。这有两个重要后果：

- 不能使用CopyObject原位加密现有对象、也不能更改原位现有对象的加密。如果您提供 `x-amz-server-side-encryption` 标题或 `x-amz-server-side-encryption-customer-algorithm` 标头、StorageGRID 拒绝请求并返回 `XNotImplemented`。
- 不会使用匹配 ILM 规则中指定的 " 载入行为 " 选项。通过正常后台 ILM 进程重新评估 ILM 时，更新触发的任何对象放置更改都会进行。

  这意味着、如果ILM规则使用stricting选项执行加数据操作、则在无法放置所需对象(例如、新需要的位置不可用)时不会执行任何操作。更新后的对象会保留其当前位置，直到可以进行所需的位置为止。

服务器端加密的请求标头

如果您 "使用服务器端加密"，您提供的请求标头取决于源对象是否已加密以及是否计划加密目标对象。

- 如果源对象使用客户提供的密钥(SSE-C)进行加密、则必须在CopyObject请求中包含以下三个标头、以便可以对该对象进行解密、然后进行复制：
  - `x-amz-copy-source-server-side-encryption-customer-algorithm`：指定 `AES256`。
  - `x-amz-copy-source-server-side-encryption-customer-key`：指定在创建源对象时提供的加密密钥。
  - `x-amz-copy-source-server-side-encryption-customer-key-MD5`：指定在创建源对象时提供的MD5摘要。
- 如果要使用您提供和管理的唯一密钥对目标对象（副本）进行加密，请包含以下三个标题：
  - `x-amz-server-side-encryption-customer-algorithm`：指定 `AES256`。
  - `x-amz-server-side-encryption-customer-key`：为目标对象指定新的加密密钥。
  - `x-amz-server-side-encryption-customer-key-MD5`：指定新加密密钥的MD5摘要。

  您提供的加密密钥永远不会存储。如果丢失加密密钥，则会丢失相应的对象。在使用客户提供的密钥保护对象数据之前、请查看的注意事项 "使用服务器端加密"。
- 如果要使用由StorageGRID (SSE)管理的唯一密钥对目标对象(副本)进行加密，请在CopyObject请求中包括此标头：
  - `x-amz-server-side-encryption`

。 server-side-encryption 无法更新对象的值。而是使用新创建副本 server-side-encryption 价值使用 x-amz-metadata-directive： REPLACE。

版本控制

如果源存储分段已版本控制、则可以使用 x-amz-copy-source 用于复制最新版本对象的标题。要复制对象的特定版本、必须使用明确指定要复制的版本 versionId 子资源。如果目标存储分段已进行版本控制、则会在中返回生成的版本 x-amz-version-id 响应标头。如果目标分段的版本控制已暂停、则 x-amz-version-id 返回"null"值。

GetObject

您可以使用S3 GetObject请求从S3存储分段中检索对象。

GetObject和多部分对象

您可以使用 partNumber 用于检索多部分或分段对象的特定部分的请求参数。 。 x-amz-mp-parts-count 响应元素指示对象有多少个零件。

您可以设置 partNumber 对于分段/多部分对象和非分段/非多部分对象、均为1；但是、 x-amz-mp-parts-count 只有分段对象或多部分对象才会返回响应元素。

用户元数据中的 UTF-8 字符

StorageGRID 不会解析或解释用户定义的元数据中的转义 UTF-8 字符。对用户定义的元数据中具有转义UTF-8字符的对象发出的获取请求不会返回 x-amz-missing-meta 如果密钥名称或值包含不可打印的字符、则为标题。

请求标头不受支持

不支持以下请求标头、并返回 XNotImplemented：

- x-amz-website-redirect-location

版本控制

如果为 versionId 未指定子资源、此操作将提取受版本控制的存储分段中的对象的最新版本。如果对象的当前版本是删除标记、则会随返回"未找到"状态 x-amz-delete-marker 响应标头设置为 true。

使用客户提供的加密密钥（ SSI-C ）进行服务器端加密的请求标头

如果使用您提供的唯一密钥对对象进行加密，请使用所有三个标头。

- x-amz-server-side-encryption-customer-algorithm：指定 AES256。
- x-amz-server-side-encryption-customer-key：指定对象的加密密钥。
- x-amz-server-side-encryption-customer-key-MD5：指定对象加密密钥的MD5摘要。

您提供的加密密钥永远不会存储。如果丢失加密密钥，则会丢失相应的对象。在使用客户提供的密钥保护对象数据之前、请查看中的注意事项 "使用服务器端加密"。

GetObject for Cloud Storage Pool对象的行为

对象已存储在中 "云存储池"，GetObject请求的行为取决于对象的状态。请参见 "HeadObject" 有关详细信息：

> 如果对象存储在云存储池中、并且该对象的一个或多个副本也位于网格中、则GetObject请求将尝试从网格中检索数据、然后再从云存储池中检索数据。

对象的状态	**GetObject**的行为
对象已载入 StorageGRID 但尚未通过 ILM 进行评估，或者存储在传统存储池中的对象或使用纠删编码	`200 OK` 检索对象的副本。
云存储池中的对象，但尚未过渡到无法检索的状态	`200 OK` 检索对象的副本。
对象已过渡到无法检索的状态	`403 Forbidden， InvalidObjectState` 使用 "RestorEObject" 请求将对象还原到可检索状态。
正在从不可检索状态还原的对象	`403 Forbidden， InvalidObjectState` 等待RestorEObject请求完成。
对象已完全还原到云存储池	`200 OK` 检索对象的副本。

云存储池中的多部分或分段对象

如果您上传的是多部分对象或 StorageGRID 将一个大型对象拆分为多个区块，则 StorageGRID 会通过取样该对象的部分或区块来确定该对象是否在云存储池中可用。在某些情况下、GetObject请求可能会错误地返回 `200 OK` 对象的某些部分已过渡到无法检索的状态、或者对象的某些部分尚未还原。

在这些情况下：

- GetObject请求可能会返回一些数据、但会在传输中途停止。
- 可能会返回后续GetObject请求 `403 Forbidden`。

GetObject和跨网格复制

如果您使用的是 … "网格联盟" 和 "跨网格复制" 已为分段启用、则S3客户端可以通过发出GetObject请求来验证对象的复制状态。响应包括特定于StorageGRID的 `x-ntap-sg-cgr-replication-status` 响应标头、它将具有以下值之一：

网格	复制状态
源	• *SUCCESS *：复制成功。 • *pending *：对象尚未复制。 • 失败：复制失败并出现永久故障。用户必须解决此错误。
目标	**REPRAM**：对象已从源网格复制。

StorageGRID 不支持 `x-amz-replication-status` 标题。

HeadObject

您可以使用S3 HeadObject请求从对象中检索元数据、而无需返回对象本身。如果对象存储在云存储池中、则可以使用HeadObject确定对象的过渡状态。

HeadObject和多部分对象

您可以使用 `partNumber` 用于检索多部分或分段对象特定部分的元数据的请求参数。 。 `x-amz-mp-parts-count` 响应元素指示对象有多少个零件。

您可以设置 `partNumber` 对于分段/多部分对象和非分段/非多部分对象、均为1；但是、 `x-amz-mp-parts-count` 只有分段对象或多部分对象才会返回响应元素。

用户元数据中的 UTF-8 字符

StorageGRID 不会解析或解释用户定义的元数据中的转义 UTF-8 字符。对用户定义的元数据中具有转义UTF-8字符的对象发出的HEAD请求不会返回 `x-amz-missing-meta` 如果密钥名称或值包含不可打印的字符、则为标题。

请求标头不受支持

不支持以下请求标头、并返回 `XNotImplemented`：

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

版本控制

如果为 `versionId` 未指定子资源、此操作将提取受版本控制的存储分段中的对象的最新版本。如果对象的当前版本是删除标记、则会随返回"未找到"状态 `x-amz-delete-marker` 响应标头设置为 `true`。

使用客户提供的加密密钥（ SSI-C ）进行服务器端加密的请求标头

如果对象使用您提供的唯一密钥进行加密，请使用所有这三个标头。

- `x-amz-server-side-encryption-customer-algorithm`：指定 `AES256`。
- `x-amz-server-side-encryption-customer-key`：指定对象的加密密钥。
- `x-amz-server-side-encryption-customer-key-MD5`：指定对象加密密钥的MD5摘要。

您提供的加密密钥永远不会存储。如果丢失加密密钥，则会丢失相应的对象。在使用客户提供的密钥保护对象数据之前、请查看中的注意事项 "使用服务器端加密"。

云存储池对象的**HeadObject**响应

对象存储在中 "云存储池"，将返回以下响应标头：

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

响应标头提供了有关对象移动到云存储池，可选择过渡到不可检索状态并已还原时的状态的信息。

对象的状态	对**HeadObject**的响应
对象已载入 StorageGRID 但尚未通过 ILM 进行评估，或者存储在传统存储池中的对象或使用纠删编码	`200 OK` (不返回任何特殊的响应标头。)

云存储池中的对象，但尚未过渡到无法检索的状态	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-AMZ-restore：ongoid-request="false"、thy-date="Sat, 23 7-20 203000:00:00 GMT"`
	
	在将对象过渡到无法检索的状态之前、为提供的值 `expiry-date` 设置为未来的某个远程时间。确切的过渡时间不受 StorageGRID 系统控制。

对象已过渡到不可检索状态，但网格上至少也存在一个副本	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-AMZ-restore：ongoid-request="false"、thy-date="Sat, 23 7-20 203000:00:00 GMT"`
	
	的值 `expiry-date` 设置为未来的某个远程时间。
	
	注意：如果网格上的副本不可用(例如、存储节点已关闭)、则必须使用问题描述 A "RestorEObject" 请求先从云存储池还原副本、然后才能成功检索对象。

对象已过渡到无法检索的状态，网格上不存在任何副本	`200 OK`
	
	`x-amz-storage-class: GLACIER`

对象的状态	对**HeadObject**的响应

正在从不可检索状态还原的对象	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-AMZ-restore：ongoy-request="true`

对象已完全还原到云存储池	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-AMZ-restore：ongoid-request="false"、thy-date="Sat, 23 7-20 2018 00：00 GMT"`
	
	。 `expiry-date` 指示何时将云存储池中的对象返回到无法检索的状态。

云存储池中的多部分或分段对象

如果您上传的是多部分对象或 StorageGRID 将一个大型对象拆分为多个区块，则 StorageGRID 会通过取样该对象的部分或区块来确定该对象是否在云存储池中可用。在某些情况下、如果某个HeadObject请求的某些部分已被转换为不可检索状态、或者该对象的某些部分尚未还原、则该请求可能会错误地返回`x-AMZ-restore：ongued-request="false"。

HeadObject和跨网格复制

如果您使用的是 … "网格联盟" 和 "跨网格复制" 已为分段启用、则S3客户端可以通过发出HeadObject请求来验证对象的复制状态。响应包括特定于StorageGRID的 `x-ntap-sg-cgr-replication-status` 响应标头、它将具有以下值之一：

网格	复制状态
源	• *SUCCESS *：复制成功。 • *pending *：对象尚未复制。 • 失败：复制失败并出现永久故障。用户必须解决此错误。
目标	**REPRAM**：对象已从源网格复制。

StorageGRID 不支持 `x-amz-replication-status` 标题。

PutObject

您可以使用S3 PutObject请求将对象添加到分段。

解决冲突

冲突的客户端请求（例如，两个客户端写入同一密钥）将以 " 最新成功 " 为基础进行解决。" 最新赢单 " 评估的

时间取决于 StorageGRID 系统何时完成给定请求，而不是 S3 客户端何时开始操作。

对象大小

一个PutObject操作的最大_Recommended_大小为5 GiB (5、368、709、120字节)。如果您的对象大于5 GiB、请使用 "多部分上传" 而是。

一个PutObject操作的最大_supported_大小为5 TiB (5、497、555、138、880字节)。

> 如果您从StorageGRID 11.5或更早版本升级、则在尝试上传超过5 GiB的对象时、将触发S3 Put Object Size Too Liger警报。如果您全新安装了StorageGRID 11.7或11.7、则在这种情况下不会触发警报。但是、为了符合AWS S3标准、未来版本的StorageGRID不支持上传超过5 GiB的对象。

用户元数据大小

Amazon S3 将每个 PUT 请求标头中用户定义的元数据的大小限制为 2 KB 。StorageGRID 将用户元数据限制为 24 KiB 。用户定义的元数据的大小是通过采用 UTF-8 编码的每个键和值的字节数之和来衡量的。

用户元数据中的 **UTF-8** 字符

如果某个请求在用户定义的元数据的密钥名称或值中包含（未转义） UTF-8 值，则会未定义 StorageGRID 行为。

StorageGRID 不会解析或解释用户定义的元数据的密钥名称或值中包含的转义 UTF-8 字符。转义的 UTF-8 字符被视为 ASCII 字符：

- 如果用户定义的元数据包含转义的UTF-8字符、则PutObject、CopyObject、GetObject和HeadObject请求会成功。
- StorageGRID 不会返回 `x-amz-missing-meta` 如果对密钥名称或值的解释值包含不可打印的字符、则为标题。

对象标记限制

您可以在上传新对象时为其添加标记，也可以将其添加到现有对象中。StorageGRID 和 Amazon S3 对每个对象最多支持 10 个标记。与对象关联的标记必须具有唯一的标记密钥。一个标记密钥的长度最多可以是 128 个 Unicode 字符，而标记值的长度最多可以是 256 个 Unicode 字符。密钥和值区分大小写。

对象所有权

在 StorageGRID 中，所有对象均归存储分段所有者帐户所有，包括由非所有者帐户或匿名用户创建的对象。

支持的请求标头

支持以下请求标头：

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

  指定时 `aws-chunked` 适用于 `Content-Encoding`StorageGRID 不会验证以下各项：

- StorageGRID 不会验证 `chunk-signature` 针对区块数据。
- StorageGRID 不会验证您为提供的值 `x-amz-decoded-content-length` 针对对象。

- `Content-Language`
- `Content-Length`
- `Content-MD5`
- `Content-Type`
- `Expires`
- `Transfer-Encoding`

  如果出现、则支持分块传输编码 `aws-chunked` 此外、还会使用有效负载签名。

- `x-amz-meta-`、后跟一个名称-值对、该对包含用户定义的元数据。

  为用户定义的元数据指定名称 - 值对时，请使用以下通用格式：

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

  如果要使用*用户定义的创建时间*选项作为ILM规则的参考时间、则必须使用 `creation-time` 作为创建对象时记录的元数据的名称。例如：

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

  的值 `creation-time` 评估为自1970年1月1日以来的秒数。

  > ILM规则不能同时使用*用户定义的创建时间*作为参考时间和平衡或严格的加注选项。创建 ILM 规则时返回错误。

- `x-amz-tagging`
- S3 对象锁定请求标头
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    如果在发出请求时没有这些标头、则会使用存储分段默认保留设置来计算对象版本模式和保留截止日期。请参见 "使用S3 REST API配置S3对象锁定"。

- SSA 请求标头：
  - `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

  请参见 [服务器端加密的请求标头]

请求标头不受支持

不支持以下请求标头：

- 。 x-amz-acl 不支持请求标头。
- 。 x-amz-website-redirect-location 不支持请求标头、将返回 XNotImplemented。

存储类选项

。 x-amz-storage-class 支持请求标头。为提交的值 x-amz-storage-class 影响StorageGRID 在载入期间保护对象数据的方式、而不影响StorageGRID 系统中存储的对象持久副本数(由ILM决定)。

如果与已加热对象匹配的ILM规则使用了严格加热选项、则 x-amz-storage-class 标题无效。

可以使用以下值 x-amz-storage-class：

- STANDARD （默认）
  - * 双提交 * ：如果 ILM 规则为载入行为指定了双提交选项，则在载入对象后，系统会立即创建该对象的第二个副本并将其分发到其他存储节点（双提交）。评估ILM时、StorageGRID 会确定这些初始临时副本是否符合规则中的放置说明。否则、可能需要在不同位置创建新对象副本、并且可能需要删除初始临时副本。
  - 已平衡：如果ILM规则指定了已平衡选项、而StorageGRID 无法立即创建规则中指定的所有副本、则StorageGRID 会在不同的存储节点上创建两个临时副本。

    如果StorageGRID 可以立即创建ILM规则(同步放置)中指定的所有对象副本、则会显示 x-amz-storage-class 标题无效。
- REDUCED_REDUNDANCY
  - * 双提交 * ：如果 ILM 规则为载入行为指定了双提交选项，则 StorageGRID 会在载入对象时创建一个临时副本（单个提交）。
  - 均衡：如果ILM规则指定了均衡选项，则只有当系统无法立即创建规则中指定的所有副本时，StorageGRID 才会创建一个临时副本。如果 StorageGRID 可以执行同步放置，则此标头不起作用。。 REDUCED_REDUNDANCY 如果与对象匹配的ILM规则创建一个复制副本、则最好使用选项。在这种情况下、使用 REDUCED_REDUNDANCY 无需在每次载入操作中创建和删除额外的对象副本。

  使用 REDUCED_REDUNDANCY 在其他情况下、不建议使用此选项。 REDUCED_REDUNDANCY 增加载入期间对象数据丢失的风险。例如，如果最初将单个副本存储在发生故障的存储节点上，而此存储节点未能进行 ILM 评估，则可能会丢失数据。

> 在任何一段时间内只复制一个副本会使数据面临永久丢失的风险。如果某个对象只存在一个复制副本，则在存储节点出现故障或出现严重错误时，该对象将丢失。在升级等维护过程中，您还会暂时失去对对象的访问权限。

指定 REDUCED_REDUNDANCY 仅影响首次载入对象时创建的副本数。它不会影响通过活动ILM策略评估对象时为

对象创建的副本数、也不会导致数据在StorageGRID系统中以较低的冗余级别进行存储。

> 如果要在启用了S3对象锁定的情况下将对象载入存储分段、则会显示 `REDUCED_REDUNDANCY` 选项将被忽略。如果要将对象载入旧的合规存储分段、则会显示 `REDUCED_REDUNDANCY` 选项返回错误。StorageGRID 将始终执行双提交载入，以确保满足合规性要求。

服务器端加密的请求标头

您可以使用以下请求标头通过服务器端加密对对象进行加密。SSE 和 SSI-C 选项是互斥的。

- * SSE* ：如果要使用 StorageGRID 管理的唯一密钥对对象进行加密，请使用以下标题。
  - `x-amz-server-side-encryption`
- * SSI-C* ：如果要使用您提供和管理的唯一密钥对对象进行加密，请使用所有这三个标头。
  - `x-amz-server-side-encryption-customer-algorithm`：指定 `AES256`。
  - `x-amz-server-side-encryption-customer-key`：指定新对象的加密密钥。
  - `x-amz-server-side-encryption-customer-key-MD5`：指定新对象加密密钥的MD5摘要。

> 您提供的加密密钥永远不会存储。如果丢失加密密钥，则会丢失相应的对象。在使用客户提供的密钥保护对象数据之前、请查看的注意事项 "使用服务器端加密"。

> 如果使用 SSE 或 SSI-C 对对象进行加密，则会忽略任何分段级别或网格级别的加密设置。

版本控制

如果为存储分段启用了版本控制、则为唯一的 `versionId` 会自动为所存储对象的版本生成。这 `versionId` 也会使用在响应中返回 `x-amz-version-id` 响应标头。

如果版本控制已暂停、则存储对象版本时为空 `versionId` 如果已存在空版本、则该版本将被覆盖。

授权标题的签名计算

使用时 `Authorization` 用于对请求进行身份验证的标头、StorageGRID 与AWS在以下方面有所不同：

- StorageGRID 不需要 `host` 要包含在中的标题 `CanonicalHeaders`。
- StorageGRID 不需要 `Content-Type` 将包含在中 `CanonicalHeaders`。
- StorageGRID 不需要 `x-amz-*` 要包含在中的标题 `CanonicalHeaders`。

> 作为一般最佳实践、请始终将这些标题包含在中 `CanonicalHeaders` 为了确保它们已通过验证；但是、如果排除这些标头、StorageGRID 不会返回错误。

有关详细信息，请参见 "授权标头的签名计算：传输单个区块中的有效负载(AWS签名版本4)"。

相关信息

"使用 ILM 管理对象"

RestorEObject

您可以使用S3 RestorEObject请求还原存储在云存储池中的对象。

支持的请求类型

StorageGRID仅支持用于还原对象的RestorEObject请求。它不支持 `SELECT` 还原类型。选择返回请求 `XNotImplemented`。

版本控制

(可选)指定 `versionId` 还原受版本控制的存储分段中特定版本的对象。如果未指定 `versionId`、将还原对象的最新版本

云存储池对象上的**RestorEObject**的行为

对象已存储在中 "云存储池"，则根据对象的状态，RestorEObject请求具有以下行为。请参见 "HeadObject" 有关详细信息：

> 如果某个对象存储在云存储池中、并且该对象的一个或多个副本也位于网格中、则无需发出RestorEObject请求来还原该对象。而是可以使用GetObject请求直接检索本地副本。

对象的状态	**RestorEObject**的行为
对象已载入 StorageGRID ，但尚未通过 ILM 进行评估，或者对象不在云存储池中	`403 Forbidden， InvalidObjectState`
云存储池中的对象，但尚未过渡到无法检索的状态	`200 OK` 不会进行任何更改。 注意：在将对象转换为不可检索状态之前，您不能更改它 `expiry-date`。
对象已过渡到无法检索的状态	`202 Accepted` 在请求正文中指定的天数内将对象的可检索副本还原到云存储池。在此期间结束时，对象将返回到无法检索的状态。 或者、也可以使用 `Tier` 请求元素以确定还原作业完成所需的时间(`Expedited， Standard`或 `Bulk`）。如果未指定 `Tier`，`Standard` 已使用层。 重要：如果对象已迁移到S3 Glaciereep Archive或云存储池使用Azure Blb存储、则无法使用还原它 `Expedited` 层。返回以下错误 `403 Forbidden， InvalidTier: Retrieval option is not supported by this storage class`。
正在从不可检索状态还原的对象	`409 Conflict， RestoreAlreadyInProgress`

对象的状态	**RestorEObject**的行为
对象已完全还原到云存储池	`200 OK` *注意：*如果对象已还原到可检索状态、则可以更改其 `expiry-date` 使用的新值重新发出RestorreObject请求 `Days`。还原日期将相对于请求时间进行更新。

SelectObjectContent

您可以使用 S3 SelectObjectContent 请求根据简单的 SQL 语句筛选 S3 对象的内容。

有关详细信息，请参见 "Amazon Simple Storage Service API参考：选择对象内容"。

开始之前

- 此租户帐户具有 S3 Select 权限。
- 您已拥有 `s3:GetObject` 要查询的对象的权限。
- 要查询的对象必须采用以下格式之一：
  - **CSX**。可以按原样使用、也可以压缩到GZIP或bzip2归档中。
  - 镶木地板。对镶木地板对象的其他要求：
    - S3 Select仅支持使用GZIP或Snappy进行列式压缩。S3 Select不支持对镶木地板对象进行整体对象压缩。
    - S3 Select不支持镶木地板输出。必须将输出格式指定为CSV或JSON。
    - 最大未压缩行组大小为512 MB。
    - 您必须使用对象架构中指定的数据类型。
    - 不能使用间隔、JSON、列表、时间或UUID逻辑类型。
- SQL 表达式的最大长度为 256 KB 。
- 输入或结果中的任何记录的最大长度为 1 MiB 。

CSV请求语法示例

```
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>
```

镶木地板请求语法示例

```
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>
```

SQL 查询示例

此查询可从美国人口统计数据中获取状态名称， 2010 年人口， 2015 年估计人口以及变更百分比。文件中非状态的记录将被忽略。

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

要查询的文件的前几行、 `SUB-EST2020_ALL.csv`，如下所示：

```
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
```

AWS命令行界面使用示例(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
```

输出文件的前几行、 `changes.csv`，如下所示：

```
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
```

AWS命令行界面使用示例(镶木地板)

```
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
```

输出文件的前几行changes.csv如下所示：

```
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,5029196,5454328,8.453279609703022113276157859029554624 6
```

多部分上传操作

多部分上传操作：概述

本节介绍 StorageGRID 如何支持多部件上传操作。

以下条件和注释适用于所有多部件上传操作：

- 一个分段的并发多部分上传不应超过1、000次、因为ListMultipartUploADS查询结果可能返回不完整的结果。
- StorageGRID 对多部件部件强制实施 AWS 大小限制。S3 客户端必须遵循以下准则：
  - 多部分上传中的每个部分必须介于 5 MiB （ 5 ， 242 ， 880 字节）和 5 GiB （ 5 ， 368 ， 709 ， 120 字节）之间。
  - 最后一部分可以小于 5 MiB （ 5 ， 242 ， 880 字节）。
  - 通常，部件大小应尽可能大。例如，对于 100 GiB 对象，请使用部件大小 5 GiB 。由于每个部件都被视为唯一的对象、因此使用较大的部件可降低StorageGRID 元数据开销。
  - 对于小于 5 GiB 的对象，请考虑使用非多部分上传。
- 如果ILM规则使用平衡或严格、则在载入多部分对象时、会针对该对象的每个部分以及在多部分上传完成后、针对整个对象评估ILM ""INGest"选项"。您应了解这会对对象和部件放置产生何种影响：
  - 如果在进行S3多部分上传时ILM发生更改、则在多部分上传完成后、对象的某些部分可能无法满足当前ILM要求。未正确放置的任何部件将排队等待ILM重新评估、并在稍后移至正确位置。
  - 在评估某个部件的 ILM 时， StorageGRID 会筛选该部件的大小，而不是对象的大小。这意味着、对象的某些部分可以存储在不满足对象整体ILM要求的位置。例如、如果规则指定所有10 GB或更大的对象存储在DC1、而所有较小的对象存储在DC2、则载入时、10部分多部分上传的每个1 GB部分都存储在

DC2。但是、在为对象整体评估ILM时、对象的所有部分都会移至DC1。

- 所有多部分上传操作均支持StorageGRID "一致性值"。
- 使用多部分上传载入对象时、 "对象分段阈值(1 GiB)" 未应用。
- 您可以根据需要使用 "服务器端加密" 多部分上传。要使用SSE (服务器端加密与StorageGRID管理的密钥)、您需要包括 `x-amz-server-side-encryption` 仅CreateMultipartUpload请求中的请求标头。要使用SSE-C (使用客户提供的密钥进行服务器端加密)、您可以在CreateMultipartUpload请求和后续每个UploadPart请求中指定相同的三个加密密钥请求标头。

操作	实施
AbortMultipartUpload	在所有 Amazon S3 REST API 行为下实施。如有更改、恕不另行通知。
CompleteMultipartUpload	请参见 "CompleteMultipartUpload"
CreateMultipartUpload (以前称为"启动多部分上传")	请参见 "CreateMultipartUpload"
ListMultipartUploads	请参见 "ListMultipartUploads"
ListParts	在所有 Amazon S3 REST API 行为下实施。如有更改、恕不另行通知。
上传部件	请参见 "上传部件"
上传PartCopy	请参见 "上传PartCopy"

CompleteMultipartUpload

CompleteMultipartUpload操作通过整合先前上传的部件来完成对象的多部分上传。

解决冲突

冲突的客户端请求（例如，两个客户端写入同一密钥）将以 " 最新成功 " 为基础进行解决。" 最新赢单 " 评估的时间取决于 StorageGRID 系统何时完成给定请求，而不是 S3 客户端何时开始操作。

请求标题

。 `x-amz-storage-class` 支持请求标头、如果匹配的ILM规则指定"双提交"或"已平衡"、则会影响StorageGRID创建的对象副本数 ""INGest"选项"。

- `STANDARD`

  （默认）指定在 ILM 规则使用双提交选项或 balanced-option 回退到创建中间副本时执行双提交载入操作。

- `REDUCED_REDUNDANCY`

指定在 ILM 规则使用双提交选项或 balanced-option 回退为创建中间副本时执行单提交载入操作。

> 如果要在启用了S3对象锁定的情况下将对象载入存储分段、则会显示 `REDUCED_REDUNDANCY` 选项将被忽略。如果要将对象载入旧的合规存储分段、则会显示 `REDUCED_REDUNDANCY` 选项返回错误。StorageGRID 将始终执行双提交载入，以确保满足合规性要求。

> 如果多部分上传未在 15 天内完成，则此操作将标记为非活动，并从系统中删除所有关联数据。

> 。 `ETag` 返回的值不是数据的MD5之和、而是遵循的Amazon S3 API实施 `ETag` 多部分对象的值。

版本控制

此操作将完成多部分上传。如果为分段启用了版本控制、则在完成多部分上传后创建对象版本。

如果为存储分段启用了版本控制、则为唯一的 `versionId` 会自动为所存储对象的版本生成。这 `versionId` 也会使用在响应中返回 `x-amz-version-id` 响应标头。

如果版本控制已暂停、则存储对象版本时为空 `versionId` 如果已存在空版本、则该版本将被覆盖。

> 如果为存储分段启用了版本控制，则完成多部分上传始终会创建新版本，即使在同一对象密钥上同时完成多部分上传也是如此。如果某个存储分段未启用版本控制，则可以先启动多部分上传，然后再对同一对象密钥启动并完成另一个多部分上传。在非版本控制的存储分段上，最后完成的多部分上传将优先。

复制，通知或元数据通知失败

如果为平台服务配置了进行多部分上传的存储分段，则即使关联的复制或通知操作失败，多部分上传也会成功。

如果发生这种情况，则会在网格管理器中针对总事件（ SMT ）发出警报。对于其通知失败的最后一个对象、最后一条事件消息将显示"Ffailed to puber-nameobject key的通知"。（要查看此消息，请选择 * 节点 * > * 存储节点 _* > * 事件 * 。在表顶部查看上次事件。)事件消息也会在中列出 `/var/local/log/bycast-err.log`。

租户可以通过更新对象的元数据或标记来触发失败的复制或通知。租户可以重新提交现有值，以避免进行不必要的更改。

CreateMultipartUpload

CreateMultipartUpload (以前称为启动多部分上传)操作会为对象启动多部分上传、并返回上传ID。

。 `x-amz-storage-class` 支持请求标头。为提交的值 `x-amz-storage-class` 影响StorageGRID 在载入期间保护对象数据的方式、而不影响StorageGRID 系统中存储的对象持久副本数(由ILM决定)。

如果与已引入对象匹配的ILM规则使用了strict ""INGest"选项"， `x-amz-storage-class` 标题无效。

可以使用以下值 `x-amz-storage-class`：

- `STANDARD` （默认）
  - *Dual Commit *：如果ILM规则指定了Dual Commit INGEST选项、则在一个对象被加注后、系统将创建该对象的第二个副本并将其分发到其他存储节点(Dual Commit)。评估ILM时、StorageGRID 会确定这些初始临时副本是否符合规则中的放置说明。否则、可能需要在不同位置创建新对象副本、并且可能需要删除初始临时副本。
  - 已平衡：如果ILM规则指定了已平衡选项、而StorageGRID 无法立即创建规则中指定的所有副本、则StorageGRID 会在不同的存储节点上创建两个临时副本。

    如果StorageGRID 可以立即创建ILM规则(同步放置)中指定的所有对象副本、则会显示 `x-amz-storage-class` 标题无效。

- `REDUCED_REDUNDANCY`
  - *Dual Commit *：如果ILM规则指定了Dual Commit选项、则StorageGRID会在对象被引入时创建一个临时副本(单个提交)。
  - 均衡：如果ILM规则指定了均衡选项，则只有当系统无法立即创建规则中指定的所有副本时，StorageGRID 才会创建一个临时副本。如果 StorageGRID 可以执行同步放置，则此标头不起作用。。 `REDUCED_REDUNDANCY` 如果与对象匹配的ILM规则创建一个复制副本、则最好使用选项。在这种情况下、使用 `REDUCED_REDUNDANCY` 无需在每次载入操作中创建和删除额外的对象副本。

  使用 `REDUCED_REDUNDANCY` 在其他情况下、不建议使用此选项。 `REDUCED_REDUNDANCY` 增加载入期间对象数据丢失的风险。例如，如果最初将单个副本存储在发生故障的存储节点上，而此存储节点未能进行 ILM 评估，则可能会丢失数据。

> 在任何一段时间内只复制一个副本会使数据面临永久丢失的风险。如果某个对象只存在一个复制副本，则在存储节点出现故障或出现严重错误时，该对象将丢失。在升级等维护过程中，您还会暂时失去对对象的访问权限。

指定 `REDUCED_REDUNDANCY` 仅影响首次载入对象时创建的副本数。它不会影响通过活动ILM策略评估对象时为对象创建的副本数、也不会导致数据在StorageGRID系统中以较低的冗余级别进行存储。

> 如果要在启用了S3对象锁定的情况下将对象载入存储分段、则会显示 `REDUCED_REDUNDANCY` 选项将被忽略。如果要将对象载入旧的合规存储分段、则会显示 `REDUCED_REDUNDANCY` 选项返回错误。StorageGRID 将始终执行双提交载入，以确保满足合规性要求。

支持以下请求标头：

- `Content-Type`
- `x-amz-meta-`、后跟一个名称-值对、该对包含用户定义的元数据

  为用户定义的元数据指定名称 - 值对时，请使用以下通用格式：

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

  如果要使用*用户定义的创建时间*选项作为ILM规则的参考时间、则必须使用 `creation-time` 作为创建对象时记录的元数据的名称。例如：

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

的值 `creation-time` 评估为自1970年1月1日以来的秒数。

> 正在添加 `creation-time` 由于在将对象添加到启用了旧合规性的存储分段时不允许使用用户定义的元数据。此时将返回错误。

- S3 对象锁定请求标头：
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    如果在不使用这些标题的情况下发出请求，则存储分段默认保留设置用于计算对象版本 retain-until 日期。

"使用S3 REST API配置S3对象锁定"

- SSA 请求标头：
  - `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`

    [服务器端加密的请求标头]

> 有关StorageGRID如何处理UTF-8字符的信息、请参见 "PutObject"。

服务器端加密的请求标头

您可以使用以下请求标头通过服务器端加密对多部分对象进行加密。SSE 和 SSI-C 选项是互斥的。

- **SSE**：如果要使用由StorageGRID管理的唯一密钥对对象进行加密，请在CreateMultipartUpload请求中使用以下标头。请勿在任何UploadPart请求中指定此标题。
  - `x-amz-server-side-encryption`
- **SSE-C**：如果要使用提供和管理的唯一密钥对对象进行加密，请在CreateMultipartUpload请求(以及后续的每个UploadPart请求)中使用所有这三个标头。
  - `x-amz-server-side-encryption-customer-algorithm`：指定 `AES256`。
  - `x-amz-server-side-encryption-customer-key`：指定新对象的加密密钥。
  - `x-amz-server-side-encryption-customer-key-MD5`：指定新对象加密密钥的MD5摘要。

您提供的加密密钥永远不会存储。如果丢失加密密钥，则会丢失相应的对象。在使用客户提供的密钥保护对象数据之前、请查看的注意事项 "使用服务器端加密"。

请求标头不受支持

不支持以下请求标头、并返回 `XNotImplemented`

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

版本控制

多部分上传包括启动上传，发布上传，上传部件，组装上传的部件以及完成上传的操作。执行CompleteMultipartUpload操作时、系统会创建对象(如果适用、还会对其进行版本管理)。

ListMultipartUploads

ListMultipartUploads操作可列出分段的正在进行的多部分上传。

支持以下请求参数：

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

版本控制

多部分上传包括启动上传，发布上传，上传部件，组装上传的部件以及完成上传的操作。执行CompleteMultipartUpload操作时、系统会创建对象(如果适用、还会对其进行版本管理)。

上传部件

UploadPart操作在对象的多部分上传中上传部件。

支持的请求标头

支持以下请求标头：

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

服务器端加密的请求标头

如果为CreateMultipartUpload请求指定了SSE-C加密、则还必须在每个UploadPart请求中包含以下请求标头：

- `x-amz-server-side-encryption-customer-algorithm`：指定 `AES256`。
- `x-amz-server-side-encryption-customer-key`：指定与CreateMultipartUpload请求中提供的加密密钥相同的加密密钥。
- `x-amz-server-side-encryption-customer-key-MD5`：指定与CreateMultipartUpload请求中提供的MD5摘要相同的MD5摘要。

> 您提供的加密密钥永远不会存储。如果丢失加密密钥，则会丢失相应的对象。在使用客户提供的密钥保护对象数据之前、请查看中的注意事项 "使用服务器端加密"。

版本控制

多部分上传包括启动上传，发布上传，上传部件，组装上传的部件以及完成上传的操作。执行CompleteMultipartUpload操作时、系统会创建对象(如果适用、还会对其进行版本管理)。

上传**PartCopy**

UploadPartCopy操作通过从现有对象作为数据源复制数据来上传部分对象。

所有Amazon S3 REST API行为均会实施UploadPartCopy操作。如有更改、恕不另行通知。

此请求读取和写入中指定的对象数据 `x-amz-copy-source-range` 在StorageGRID 系统中。

支持以下请求标头：

- `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`

服务器端加密的请求标头

如果为CreateMultipartUpload请求指定了SSE-C加密、则还必须在每个UploadPartCopy请求中包含以下请求标头：

- `x-amz-server-side-encryption-customer-algorithm`：指定 `AES256`。
- `x-amz-server-side-encryption-customer-key`：指定与CreateMultipartUpload请求中提供的加密密钥相同的加密密钥。
- `x-amz-server-side-encryption-customer-key-MD5`：指定与CreateMultipartUpload请求中提供的MD5摘要相同的MD5摘要。

如果源对象使用客户提供的密钥(SSE-C)进行加密、则必须在UploadPartCopy请求中包含以下三个标头、以便可以解密并复制该对象：

- `x-amz-copy-source-server-side-encryption-customer-algorithm`：指定 `AES256`。

- `x-amz-copy-source-server-side-encryption-customer-key`：指定在创建源对象时提供的加密密钥。
- `x-amz-copy-source-server-side-encryption-customer-key-MD5`：指定在创建源对象时提供的MD5摘要。

> 您提供的加密密钥永远不会存储。如果丢失加密密钥，则会丢失相应的对象。在使用客户提供的密钥保护对象数据之前、请查看中的注意事项 "使用服务器端加密"。

版本控制

多部分上传包括启动上传，发布上传，上传部件，组装上传的部件以及完成上传的操作。执行CompleteMultipartUpload操作时、系统会创建对象(如果适用、还会对其进行版本管理)。

错误响应

StorageGRID 系统支持所有适用的标准 S3 REST API 错误响应。此外， StorageGRID 实施还添加了多个自定义响应。

支持的 **S3 API** 错误代码

Name	**HTTP** 状态
ACCESSDENIED	403 已禁用
BadDigest	400 个错误请求
BucketAlreadyExists	409 冲突
BucketNotEmpagty	409 冲突
实体不完整	400 个错误请求
内部错误	500 内部服务器错误
InvalidAccessKeyId	403 已禁用
InvalidArgument	400 个错误请求
InvalidBucketName	400 个错误请求
InvalidBucketState	409 冲突
InvalidDigest	400 个错误请求
InvalidEncryptionAlgorithmError	400 个错误请求

Name	HTTP 状态
InvalidPart	400 个错误请求
InvalidPartOrder	400 个错误请求
InvalidRange	416 无法满足请求的范围
InvalidRequest	400 个错误请求
InvalidStorageClass	400 个错误请求
InvalidTag	400 个错误请求
InvalidURI	400 个错误请求
KeyTooLong	400 个错误请求
MalformedXML	400 个错误请求
MetadataTooLarge	400 个错误请求
方法未使用	不允许使用 405 方法
MissingContent长度	411 需要长度
MissingRequestBodyError	400 个错误请求
MissingSecurityHeader	400 个错误请求
NoSuchBucket	未找到 404
NoSuchKey	未找到 404
NoSuchUpload	未找到 404
未实施	501 未实施
NoSuchBucketPolicy	未找到 404
ObjectLockConfigurationNotFoundError	未找到 404
预条件已启用	412- 前提条件失败

Name	HTTP 状态
已请求超时	403 已禁用
服务不可用	503 服务不可用
SignatureDoesNotMatch	403 已禁用
TooMany桶	400 个错误请求
用户密钥已规范	400 个错误请求

StorageGRID 自定义错误代码

Name	Description	HTTP 状态
XBucketLifeycleNotAllowed	旧版合规存储分段不支持存储分段生命周期配置	400 个错误请求
XBucketPolicyParseException	无法解析收到的存储分段策略 JSON 。	400 个错误请求
XComplianceConflict	操作因原有合规性设置而被拒绝。	403 已禁用
XComplianceReducedRedundancyFor禁用	原有的合规存储分段不允许减少冗余	400 个错误请求
XMaxBucketPolicyLengthExceeded	您的策略超出了允许的最大存储分段策略长度。	400 个错误请求
XMissingInternalRequestHeader	缺少内部请求的标题。	400 个错误请求
XNoSuchBucketCompliance	指定的存储分段未启用原有合规性。	未找到 404
XNotAcceptable	此请求包含一个或多个无法满足的接受标头。	406 不可接受
未实施	您提供的请求意味着未实施的功能。	501 未实施

StorageGRID自定义操作

StorageGRID自定义操作：概述

StorageGRID系统支持添加到S3 REST API中的自定义操作。

下表列出了StorageGRID支持的自定义操作。

操作	Description
"获取存储分段一致性"	返回应用于特定存储分段的一致性。
"PUT 存储分段一致性"	设置应用于特定存储分段的一致性。
"获取存储分段上次访问时间"	返回为特定存储分段启用还是禁用上次访问时间更新。
"PUT 分段上次访问时间"	用于启用或禁用特定存储分段的上次访问时间更新。
"删除存储分段元数据通知配置"	删除与特定存储分段关联的元数据通知配置 XML 。
"获取存储分段元数据通知配置"	返回与特定存储分段关联的元数据通知配置 XML 。
"PUT 存储分段元数据通知配置"	配置存储分段的元数据通知服务。
"获取存储使用量"	告诉您某个帐户以及与该帐户关联的每个存储分段使用的总存储量。
"已弃用：具有合规性设置的CreateBucket"	已弃用且不支持：您无法再在启用合规性的情况下创建新存储分段。
"已弃用：GET分段合规性"	已弃用但受支持：返回当前对现有旧版合规存储分段有效的合规性设置。
"已弃用：Put Bucket"	已弃用但受支持：允许您修改现有旧版合规存储分段的合规性设置。

获取存储分段一致性

通过GET分段一致性请求、您可以确定应用于特定分段的一致性。

默认一致性设置为保证新创建的对象在写入后进行读取。

要完成此操作、您必须具有S3：GetBucketConsistency权限或帐户root。

请求示例

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

响应

在响应XML中、 `<Consistency>` 将返回以下值之一：

一致性	Description
全部	所有节点都会立即接收数据，否则请求将失败。
强大的全局功能	保证所有站点中所有客户端请求的写入后读一致性。
强大的站点	保证站点内所有客户端请求的写入后读一致性。
读后写	（默认）为新对象提供写入后读一致性，并为对象更新提供最终一致性。提供高可用性和数据保护保证。建议用于大多数情况。
可用	为新对象和对象更新提供最终一致性。对于S3存储分段、请仅在需要时使用(例如、对于包含很少读取的日志值的存储分段、或者对于不存在的密钥执行HEAD或GET操作)。S3 FabricPool 存储分段不支持。

响应示例

```
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>
```

相关信息

"一致性值"

PUT 存储分段一致性

通过"放置分段一致性请求"、您可以指定要应用于对分段执行的操作的一致性。

默认一致性设置为保证新创建的对象在写入后进行读取。

开始之前

要完成此操作、您必须具有S3：PutBucketConsistency权限或帐户root。

请求

。 `x-ntap-sg-consistency` 参数必须包含以下值之一：

一致性	Description
全部	所有节点都会立即接收数据，否则请求将失败。
强大的全局功能	保证所有站点中所有客户端请求的写入后读一致性。
强大的站点	保证站点内所有客户端请求的写入后读一致性。
读后写	（默认）为新对象提供写入后读一致性，并为对象更新提供最终一致性。提供高可用性和数据保护保证。建议用于大多数情况。
可用	为新对象和对象更新提供最终一致性。对于S3存储分段、请仅在需要时使用(例如、对于包含很少读取的日志值的存储分段、或者对于不存在的密钥执行HEAD或GET操作)。S3 FabricPool 存储分段不支持。

*注意：*通常、应使用"新写入后读取"一致性。如果请求无法正常工作、请尽可能更改应用程序客户端的行为。或者、将客户端配置为为为每个API请求指定一致性。只能在最后一种方法下、在存储分段级别设置一致性。

请求示例

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

相关信息

"一致性值"

获取存储分段上次访问时间

通过获取分段上次访问时间请求，您可以确定是为单个分段启用还是禁用了上次访问时间更新。

要完成此操作、您必须具有S3：GetBucketLastAccessTime权限或帐户root。

请求示例

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

响应示例

此示例显示已为存储分段启用上次访问时间更新。

```
HTTP/1.1 200 OK
Date: Sat, 29 Nov 2015 01:02:18 GMT
Connection: CLOSE
Server: StorageGRID/10.3.0
x-amz-request-id: 12345
Content-Length: 127
Content-Type: application/xml

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

PUT 分段上次访问时间

通过 PUT 分段上次访问时间请求，您可以为各个分段启用或禁用上次访问时间更新。禁用上次访问时间更新可提高性能，它是使用 10.3.0 或更高版本创建的所有存储分段的默认设置。

要完成此操作、您必须对某个存储分段拥有S3：PutBucketLastAccessTime权限、或者以root帐户身份登录。

> 从 StorageGRID 10.3 版开始，默认情况下，所有新存储分段都会禁用对上次访问时间的更新。如果您的存储分段是使用早期版本的 StorageGRID 创建的，并且您希望与新的默认行为匹配，则必须明确禁用上述每个存储分段的上次访问时间更新。您可以使用"放置分段上次访问时间"请求或从租户管理器中某个分段的详细信息页面启用或禁用对上次访问时间的更新。请参见 "启用或禁用上次访问时间更新"。

如果禁用了某个存储分段的上次访问时间更新，则会对存储分段上的操作应用以下行为：

- GetObject、GetObjectAcl、GetObjectTaging和HeadObject请求不更新上次访问时间。此对象不会添加到用于信息生命周期管理（ ILM ）评估的队列中。
- 仅更新元数据的CopyObject和PutObjectTagingRequests也会更新上次访问时间。对象将添加到队列中以进行 ILM 评估。
- 如果对源存储分段禁用了对上次访问时间的更新、则CopyObject请求不会更新源存储分段的上次访问时间。复制的对象不会添加到源存储分段的 ILM 评估队列中。但是、对于目标、CopyObject请求始终会更新上次访问时间。对象副本将添加到队列中以进行 ILM 评估。
- CompleteMultipartUpload请求更新上次访问时间。已完成的对象将添加到队列中以进行 ILM 评估。

请求示例

此示例将为存储分段启用上次访问时间。

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

此示例将禁用存储分段的上次访问时间。

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

删除存储分段元数据通知配置

通过删除存储分段元数据通知配置请求，您可以通过删除配置 XML 来禁用各个存储分段的搜索集成服务。

要完成此操作、您必须对某个存储分段拥有S3：DeleteBucketMetadataNotification权限、或者以root帐户身份登录。

请求示例

此示例显示了禁用存储分段的搜索集成服务。

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

获取存储分段元数据通知配置

使用获取分段元数据通知配置请求，您可以检索用于为各个分段配置搜索集成的配置 XML 。

要完成此操作、您必须具有S3：GetBucketMetadataNotification权限或帐户root。

请求示例

此请求将检索名为的存储分段的元数据通知配置 `bucket`。

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

响应

响应正文包括存储分段的元数据通知配置。通过元数据通知配置，您可以确定如何配置存储分段以进行搜索集成。也就是说，您可以通过它确定哪些对象已编制索引，以及将其对象元数据发送到哪些端点。

```
<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>
```

每个元数据通知配置都包含一个或多个规则。每个规则都指定其适用场景 的对象以及 StorageGRID 应将对象元数据发送到的目标。必须使用 StorageGRID 端点的 URN 指定目标。

Name	Description	Required
MetadataNotificationConfiguration	用于指定元数据通知的对象和目标的规则的容器标记。 包含一个或多个规则元素。	是的。
规则	用于标识应将其元数据添加到指定索引中的对象的规则的容器标记。 拒绝前缀重叠的规则。 包含在 MetadataNotificationConfiguration 元素中。	是的。

Name	Description	Required
ID	规则的唯一标识符。 包含在 Rule 元素中。	否
Status	状态可以是 " 已启用 " 或 " 已禁用 " 。不会对已禁用的规则执行任何操作。 包含在 Rule 元素中。	是的。
前缀	与前缀匹配的对象受此规则的影响，其元数据将发送到指定目标。 要匹配所有对象，请指定一个空前缀。 包含在 Rule 元素中。	是的。
目标	规则目标的容器标记。 包含在 Rule 元素中。	是的。
URN	发送对象元数据的目标的 urn 。必须是具有以下属性的 StorageGRID 端点的 URN ： • `es` 必须是第三个元素。 • URN必须以存储元数据的索引和类型结尾、格式为 `domain-name/myindex/mytype`。 端点使用租户管理器或租户管理 API 进行配置。它们的形式如下： • `arn:aws:es:_region:account-ID_:domain/mydomain/myindex/mytype` • `urn:mysite:es:::mydomain/myindex/mytype` 必须在提交配置 XML 之前配置端点，否则配置将失败并显示 404 错误。 urn 包含在目标元素中。	是的。

响应示例

包含在之间的XML `<MetadataNotificationConfiguration></MetadataNotificationConfiguration>` 标记显示了如何为存储分段配置与搜索集成端点的集成。在此示例中、对象元数据将发送到名为的Elasticsearch索引 `current` 并键入named `2017` 托管在名为的AWS域中 `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>
```

相关信息

"使用租户帐户"

PUT 存储分段元数据通知配置

通过 PUT Bucket 元数据通知配置请求，您可以为各个存储分段启用搜索集成服务。您在请求正文中提供的元数据通知配置 XML 用于指定将其元数据发送到目标搜索索引的对象。

要完成此操作、您必须对某个存储分段拥有S3：PutBucketMetadataNotification权限、或者以root帐户身份登录。

请求

此请求必须在请求正文中包含元数据通知配置。每个元数据通知配置都包含一个或多个规则。每个规则都指定其适用场景 的对象以及 StorageGRID 应将对象元数据发送到的目标。

可以按对象名称的前缀筛选对象。例如、您可以发送具有前缀的对象的元数据 `/images` 到一个目标、以及具有前缀的对象 `/videos` 另一个。

前缀重叠的配置无效、在提交时将被拒绝。例如、一种配置、其中包含一个规则、用于具有前缀的对象 `test` 和第二个规则、用于具有前缀的对象 `test2` 不允许。

必须使用 StorageGRID 端点的 URN 指定目标。如果提交元数据通知配置、或者请求以失败的形式出现故障、则端点必须存在 `400 Bad Request`。错误消息显示： `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>
```

下表介绍了元数据通知配置 XML 中的元素。

Name	Description	Required
MetadataNotificationConfiguration	用于指定元数据通知的对象和目标的规则的容器标记。 包含一个或多个规则元素。	是的。
规则	用于标识应将其元数据添加到指定索引中的对象的规则的容器标记。 拒绝前缀重叠的规则。 包含在 MetadataNotificationConfiguration 元素中。	是的。
ID	规则的唯一标识符。 包含在 Rule 元素中。	否
Status	状态可以是 " 已启用 " 或 " 已禁用 " 。不会对已禁用的规则执行任何操作。 包含在 Rule 元素中。	是的。

Name	Description	Required
前缀	与前缀匹配的对象受此规则的影响，其元数据将发送到指定目标。 要匹配所有对象，请指定一个空前缀。 包含在 Rule 元素中。	是的。
目标	规则目标的容器标记。 包含在 Rule 元素中。	是的。
URN	发送对象元数据的目标的 urn 。必须是具有以下属性的 StorageGRID 端点的 URN： • `es` 必须是第三个元素。 • URN必须以存储元数据的索引和类型结尾、格式为 `domain-name/myindex/mytype`。 端点使用租户管理器或租户管理 API 进行配置。它们的形式如下： • `arn:aws:es:region:account-ID:domain/mydomain/myindex/mytype` • `urn:mysite:es:::mydomain/myindex/mytype` 必须在提交配置 XML 之前配置端点，否则配置将失败并显示 404 错误。 urn 包含在目标元素中。	是的。

请求示例

此示例显示了为存储分段启用搜索集成。在此示例中，所有对象的对象元数据都将发送到同一目标。

```
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>
```

在此示例中、是指与前缀匹配的对象的对象元数据 `/images` 发送到一个目标、而与前缀匹配的对象的对象元数据则发送到一个目标 `/videos` 发送到另一个目标。

```
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>
```

由搜索集成服务生成的 **JSON**

为存储分段启用搜索集成服务后，每次添加，更新或删除对象元数据或标记时，系统都会生成一个 JSON 文档并将其发送到目标端点。

此示例显示了使用密钥的对象时可能生成的JSON示例 `SGWS/Tagging.txt` 在名为的存储分段中创建 `test`。。 `test` 存储分段未进行版本控制、因此 `versionId` 标记为空。

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

元数据通知中包含的对象元数据

下表列出了启用搜索集成后发送到目标端点的 JSON 文档中包含的所有字段。

文档名称包括存储分段名称，对象名称和版本 ID （如果存在）。

Type	项目名称	Description
存储分段和对象信息	存储分段	存储分段的名称
存储分段和对象信息	key	对象密钥名称
存储分段和对象信息	版本 ID	对象版本，用于受版本控制的分段中的对象
存储分段和对象信息	region	分段区域、例如 `us-east-1`
系统元数据	size	HTTP 客户端可见的对象大小（以字节为单位）
系统元数据	MD5	对象哈希

Type	项目名称	Description
用户元数据	元数据 `key:value`	对象的所有用户元数据，作为键值对
Tags	tags `key:value`	为对象定义的所有对象标记，作为键值对

对于标记和用户元数据， StorageGRID 会将日期和数字作为字符串或 S3 事件通知传递给 Elasticsearch 。要配置 Elasticsearch 以将这些字符串解释为日期或数字，请按照 Elasticsearch 说明进行动态字段映射和映射日期格式。在配置搜索集成服务之前，必须在索引上启用动态字段映射。为文档编制索引后、无法在索引中编辑文档的域类型。

相关信息

"使用租户帐户"

获取存储使用情况请求

" 获取存储使用量 " 请求会告知您帐户正在使用的存储总量以及与帐户关联的每个存储分段的存储总量。

通过修改后的ListBucb计 请求、可以获取帐户及其存储分段使用的存储量 `x-ntap-sg-usage` 查询参数。存储分段使用量与系统处理的 PUT 和 DELETE 请求分开跟踪。根据请求处理情况，使用量值与预期值匹配可能会有一定的延迟，尤其是在系统负载较重时。

默认情况下， StorageGRID 会尝试使用强全局一致性检索使用情况信息。如果无法实现强全局一致性、StorageGRID 会尝试在强站点一致性处检索使用情况信息。

要完成此操作、您必须具有S3：ListAllMy桶 权限或帐户root。

请求示例

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

响应示例

此示例显示了一个帐户，该帐户在两个存储分段中包含四个对象和 12 字节的数据。每个存储分段包含两个对象和六个字节的数据。

```
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>
```

版本控制

存储的每个对象版本都将用于 `ObjectCount` 和 `DataBytes` 响应中的值。删除标记未添加到中 `ObjectCount` 总计。

相关信息

"一致性值"

已弃用旧合规性存储分段请求

已弃用旧合规性存储分段请求

您可能需要使用 StorageGRID S3 REST API 来管理使用原有合规性功能创建的分段。

已弃用合规性功能

先前 StorageGRID 版本中提供的 StorageGRID 合规性功能已弃用，并已被 S3 对象锁定取代。

如果您之前启用了全局合规性设置，则会在 StorageGRID 11.6 中启用全局 S3 对象锁定设置。您不能再在启用

了合规性的情况下创建新的存储分段；但是，您可以根据需要使用 StorageGRID S3 REST API 管理任何现有的旧合规存储分段。

- "使用S3 REST API配置S3对象锁定"
- "使用 ILM 管理对象"
- "NetApp 知识库：如何在 StorageGRID 11.5 中管理原有的合规存储分段"

已弃用的合规性请求：

- "已弃用 - 为符合性修改存储分段请求"

  SGCompliance XML 元素已弃用。以前，您可以将此 StorageGRID 自定义元素包含在 PUT 存储分段请求的可选 XML 请求正文中，以创建合规存储分段。

- "已弃用—获取存储分段合规性"

  获取存储分段合规性请求已弃用。但是，您可以继续使用此请求来确定当前对现有旧版合规存储分段有效的合规性设置。

- "已弃用—放置分段合规性"

  PUT 存储分段合规性请求已弃用。但是，您可以继续使用此请求修改现有旧版合规存储分段的合规性设置。例如，您可以将现有存储分段置于合法保留状态或延长其保留期限。

已弃用：为合规性修改**CreateBucket**

SGCompliance XML 元素已弃用。以前、您可以将此StorageGRID自定义元素包含在CreateBuckets请求的可选XML请求正文中、以创建兼容分段。

> 先前 StorageGRID 版本中提供的 StorageGRID 合规性功能已弃用，并已被 S3 对象锁定取代。有关详细信息、请参见以下内容：
>
> - "使用S3 REST API配置S3对象锁定"
> - "NetApp 知识库：如何在 StorageGRID 11.5 中管理原有的合规存储分段"

您不能再在已启用合规性的情况下创建新存储分段。如果尝试使用CreateBucket"请求修改以实现合规性"来创建新的合规分段、则会返回以下错误消息：

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

已弃用：获取存储分段合规性请求

获取存储分段合规性请求已弃用。但是，您可以继续使用此请求来确定当前对现有旧版合规存储分段有效的合规性设置。

先前 StorageGRID 版本中提供的 StorageGRID 合规性功能已弃用，并已被 S3 对象锁定取代。有关详细信息、请参见以下内容：

- "使用S3 REST API配置S3对象锁定"
- "NetApp 知识库：如何在 StorageGRID 11.5 中管理原有的合规存储分段"

要完成此操作、您必须具有S3：GetBucketCompliance权限或帐户root。

请求示例

通过此示例请求、您可以确定名为的存储分段的合规性设置 `mybucket`。

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

响应示例

在响应XML中、 `<SGCompliance>` 列出了对存储分段有效的合规性设置。此示例响应显示了一个存储分段的合规性设置，从将对象载入网格开始，每个对象将保留一年（ 525600 分钟）。此存储分段当前没有法律上的保留。每个对象将在一年后自动删除。

```
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>
```

Name	Description
RetentionPeriodMinutes	添加到此存储分段的对象的保留期限长度，以分钟为单位。保留期限从将对象载入网格时开始。

Name	Description
乐高积木	• true ： 此存储分段当前处于合法保留状态。在解除合法保留之前、无法删除此存储分段中的对象、即使其保留期限已到期也是如此。 • false ： 此存储分段当前未处于合法保留状态。此存储分段中的对象可以在保留期限到期时删除。
自动删除	• true ： 此存储分段中的对象将在保留期限到期时自动删除，除非此存储分段处于合法保留状态。 • false ： 保留期限到期后，不会自动删除此存储分段中的对象。如果需要删除这些对象，必须手动将其删除。

错误响应

如果未创建符合要求的存储分段、则响应的HTTP状态代码为 `404 Not Found`、带有S3错误代码 `XNoSuchBucketCompliance`。

已弃用： **PUT** 存储分段合规性请求

PUT 存储分段合规性请求已弃用。但是，您可以继续使用此请求修改现有旧版合规存储分段的合规性设置。例如，您可以将现有存储分段置于合法保留状态或延长其保留期限。

> 先前 StorageGRID 版本中提供的 StorageGRID 合规性功能已弃用，并已被 S3 对象锁定取代。有关详细信息、请参见以下内容：
>
> • "使用S3 REST API配置S3对象锁定"
> • "NetApp 知识库：如何在 StorageGRID 11.5 中管理原有的合规存储分段"

要完成此操作、您必须具有S3：PutBucketCompliance权限或帐户root。

发出 PUT 存储分段合规性请求时，必须为合规性设置的每个字段指定一个值。

请求示例

此示例请求修改名为的存储分段的合规性设置 `mybucket`。在此示例中、对象位于中 `mybucket` 现在将保留两年(1、051、200分钟)、而不是一年、从将对象载入网格开始。此存储分段没有法律上的保留。每个对象将在两年后自动删除。

```
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>
```

Name	Description
RetentionPeriodMinutes	添加到此存储分段的对象的保留期限长度，以分钟为单位。保留期限从将对象载入网格时开始。 *重要*为RettionPeriodMinutes指定新值时、必须指定一个等于或大于存储分段的当前保留期限的值。设置存储分段的保留期限后、您不能减小该值、只能增加该值。
乐高积木	• true ： 此存储分段当前处于合法保留状态。在解除合法保留之前、无法删除此存储分段中的对象、即使其保留期限已到期也是如此。 • false ： 此存储分段当前未处于合法保留状态。此存储分段中的对象可以在保留期限到期时删除。
自动删除	• true ： 此存储分段中的对象将在保留期限到期时自动删除，除非此存储分段处于合法保留状态。 • false ： 保留期限到期后，不会自动删除此存储分段中的对象。如果需要删除这些对象，必须手动将其删除。

合规性设置的一致性

当您使用 PUT 存储分段合规性请求更新 S3 存储分段的合规性设置时， StorageGRID 会尝试更新整个网格中存储分段的元数据。默认情况下、StorageGRID会使用*强-全局*一致性来保证所有数据中心站点和包含存储分段元数据的所有存储节点在更改合规性设置后都具有读写后一致性。

如果由于一个数据中心站点或一个站点上的多个存储节点不可用而导致StorageGRID无法实现*强全局*一致性、则响应的HTTP状态代码为 `503 Service Unavailable.`

如果收到此响应，您必须联系网格管理员，以确保所需的存储服务尽快可用。如果网格管理员无法使每个站点上的存储节点足够可用、技术支持可能会通过强制保持*强站点*一致性来指示您重试失败的请求。

除非技术支持指示您执行此操作、并且您了解使用此级别可能会产生的后果、否则切勿强制实施*强站点*一致性以满足放入存储分段合规性要求。

当一致性降低到*强站点*时，StorageGRID保证更新后的合规性设置将仅对站点内的客户端请求具有写后读的一致性。这意味着，在所有站点和存储节点均可用之前， StorageGRID 系统可能会暂时为此存储分段设置多个不一致的设置。设置不一致可能导致意外和意外的行为。例如、如果您将存储分段置于合法保留状态、而您强制实施较低的一致性、则存储分段的先前合规性设置(即合法保留)可能仍会在某些数据中心站点有效。因此，您认为处于合法保留状态的对象可能会在保留期限到期时被用户删除，或者如果启用了自动删除，也可以删除。

要强制使用*强站点*一致性、请重新发出Put Bucket*合规性请求并包含 `Consistency-Control` HTTP请求标头、如下所示：

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

错误响应

- 如果未创建符合要求的存储分段、则响应的HTTP状态代码为 `404 Not Found`。
- 条件 `RetentionPeriodMinutes` 在请求小于存储分段的当前保留期限时、HTTP状态代码为 `400 Bad Request`。

相关信息

"已弃用：为满足合规性而修改存储分段请求"

存储分段和组访问策略

使用存储分段和组访问策略

StorageGRID 使用 Amazon Web Services （ AWS ）策略语言允许 S3 租户控制对这些存储分段和对象的访问。StorageGRID 系统实施 S3 REST API 策略语言的一个子集。S3 API 的访问策略以 JSON 格式写入。

访问策略概述

StorageGRID 支持两种访问策略。

- 存储分段策略，使用GetBucketPolicy、PutBucketPolicy和DeleteBucketPolicy S3 API操作进行管理。存储分段策略附加到存储分段，因此，可以对其进行配置，以控制存储分段所有者帐户或其他帐户中的用户对存储分段及其对象的访问。一个存储分段策略适用场景 只能包含一个存储分段，并且可能包含多个组。
- * 组策略 * ，使用租户管理器或租户管理 API 配置。组策略会附加到帐户中的某个组，因此，这些策略会配置为允许该组访问该帐户拥有的特定资源。一个组策略只对一个组进行适用场景 ，并且可能对多个存储分段进行。

> 组策略和存储分段策略之间的优先级没有差别。

StorageGRID 存储分段和组策略遵循由 Amazon 定义的特定语法。每个策略中都包含一组策略语句，每个语句都包含以下元素：

- 语句 ID （ SID ）（可选）

- 影响
- 主体 / 不重要
- 资源 /NotResource
- 操作 / 未操作
- 条件（可选）

策略语句是使用此结构构建的，用于指定权限： Grant <Effic> to allow/deny <Principe> to Perform <Action> on <Resource> when <condition> applies 。

每个策略元素都用于特定功能：

Element	Description
SID	Sid 元素是可选的。SID 仅用作用户的问题描述 。它会被存储，但不会被 StorageGRID 系统解释。
影响	使用 Effect 元素确定是否允许或拒绝指定的操作。您必须使用支持的 Action Element 关键字来确定允许（或拒绝）对存储分段或对象执行的操作。
主体 / 不重要	您可以允许用户，组和帐户访问特定资源并执行特定操作。如果请求中不包含 S3 签名，则可以通过指定通配符（ * ）作为主体来进行匿名访问。默认情况下，只有帐户 root 有权访问该帐户拥有的资源。 您只需要在存储分段策略中指定主体元素。对于组策略，附加该策略的组为隐式主体元素。
资源 /NotResource	资源元素用于标识分段和对象。您可以使用 Amazon 资源名称（ ARN ）来标识资源，从而允许或拒绝对存储分段和对象的权限。
操作 / 未操作	操作和效果元素是权限的两个组成部分。当组请求资源时，它们会被授予或拒绝访问该资源。除非您明确分配权限，否则访问将被拒绝，但您可以使用显式拒绝覆盖由其他策略授予的权限。
条件	条件元素是可选的。通过条件，您可以构建表达式以确定何时应用策略。

在 Action 元素中，您可以使用通配符（ * ）指定所有操作或部分操作。例如，此操作与 S3： GetObject ， S3 ： PutObject 和 S3： DeleteObject 等权限匹配。

```
s3:*Object
```

在资源元素中，可以使用通配符（* ）和（？ ）。星号（ * ）与 0 个或更多字符匹配时，问号（？ ） 匹配任意单个字符。

在Principal元素 中、不支持使用通配符、但设置匿名访问除外、此操作会向所有人授予权限。例如，您将通配符（ * ）设置为 Principal 值。

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

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

在以下示例中，该语句使用的是 " 影响 " ， " 主体 " ， " 操作 " 和 " 资源 " 元素。此示例显示了一个完整的存储分段策略语句、该语句使用"allow"的效果为Principals即管理组 `federated-group/admin` 和财务团队 `federated-group/finance`、执行操作的权限 `s3:ListBucket` 位于名为的存储分段上 `mybucket` 和操作 `s3:GetObject` 存储在该存储分段内的所有对象上。

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

存储分段策略的大小限制为 20 ， 480 字节，而组策略的大小限制为 5 ， 120 字节。

策略一致性

默认情况下，对组策略所做的任何更新最终都是一致的。当组策略保持一致时、由于策略缓存、更改可能需要额外15分钟才能生效。默认情况下、您对存储分段策略进行的任何更新都具有强烈的一致性。

您可以根据需要更改存储分段策略更新的一致性保证。例如、您可能希望在站点中断期间对存储分段策略进行更改。

在这种情况下、您可以设置 `Consistency-Control` 标题、也可以使用Put BucketPolicy请求。如果存储分段策略保持一致、则由于策略缓存、所做的更改可能需要额外8秒才能生效。

如果您将一致性设置为其他值以解决临时情况、请务必在完成后将存储分段级别设置恢复为其原始值。否则、所有未来存储分段请求都将使用修改后的设置。

在策略语句中使用 **ARN**

在策略语句中， ARN 用于 Principal 和 Resource Element 。

- 使用以下语法指定 S3 资源 ARN ：

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

- 使用以下语法指定身份资源 ARN （用户和组）：

```
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
```

其他注意事项：

- 您可以使用星号（ * ）作为通配符，以匹配对象密钥中的零个或多个字符。
- 可以在对象密钥中指定的国际字符应使用 JSON UTF-8 或 JSON \u 转义序列进行编码。不支持百分比编码。

  "RFC 2141 URN 语法"

  PutBucketPolicy操作的HTTP请求正文必须使用charset=UTF-8进行编码。

在策略中指定资源

在策略语句中，您可以使用资源元素指定允许或拒绝权限的分段或对象。

- 每个策略语句都需要一个资源元素。在策略中、资源由元素表示 `Resource`或者、 `NotResource` 以排除。
- 您可以使用 S3 资源 ARN 指定资源。例如：

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

- 您也可以在对象密钥中使用策略变量。例如：

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

- 资源值可以指定创建组策略时尚不存在的存储分段。

指定策略中的主体

使用 Principal 元素标识策略语句允许 / 拒绝访问资源的用户，组或租户帐户。

- 存储分段策略中的每个策略语句都必须包含一个主体元素。组策略中的策略语句不需要Principal元素、因为该组被理解为主体。
- 在策略中、主体由元素"Principal"或"NotPrincipal"表示以供排除。
- 必须使用 ID 或 ARN 指定基于帐户的身份：

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

- 此示例使用租户帐户 ID 27233906934684427525 ，其中包括帐户 root 和帐户中的所有用户：

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

- 您只能指定帐户 root ：

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

- 您可以指定一个特定的联合用户（ "Alex" ）：

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

- 您可以指定特定的联合组（ "Managers" ）：

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

- 您可以指定匿名主体：

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

- 为避免歧义，您可以使用用户 UUID ，而不是用户名：

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

例如、假设Alex离开了组织和用户名 `Alex` 已删除。如果新的Alex加入了该组织并获得了相同的分配 `Alex` 用户名、新用户可能会意外继承授予原始用户的权限。

- 主体值可以指定在创建存储分段策略时尚不存在的组 / 用户名称。

在策略中指定权限

在策略中， Action 元素用于允许 / 拒绝对资源的权限。您可以在策略中指定一组权限，这些权限由元素 "Action" 或 "NotAction" 表示以表示排除。其中每个元素都映射到特定的 S3 REST API 操作。

下表列出了应用于存储分段的权限以及应用于对象的权限。

现在、Amazon S3会对PutBucketReplication和DeleteBucketReplication操作使用S3：PutReplication配置权限。StorageGRID 对每个操作使用单独的权限，这些权限与原始 Amazon S3 规范匹配。

使用放置覆盖现有值时执行删除。

应用于存储分段的权限

权限	**S3 REST API** 操作	为 **StorageGRID** 自定义
S3： CreateBucket	CreateBucket	是的。 注意：仅用于组策略。
S3： DeleteBucket	DeleteBucket	
S3： DeleteBucketMetadataNotification	删除存储分段元数据通知配置	是的。
S3： DeleteBucketPolicy	DeleteBucketPolicy	
S3： DeleteReplicationConfiguration	DeleteBucketReplication	可以、分开放置和删除权限
S3： GetBucketAcl	GetBucketAcl	
S3： GetBucketCompliance	获取存储分段合规性（已弃用）	是的。
S3： GetBucketConsistency	获取存储分段一致性	是的。

权限	S3 REST API 操作	为 StorageGRID 自定义
S3： GetBucketCORS	GetBucketCors	
S3： GetEncryptionConfiguration	GetBucketEncryption	
S3： GetBucketLastAccessTime	获取存储分段上次访问时间	是的。
S3： GetBucketLocation	GetBucketLocation	
S3： GetBucketMetadataNotification	获取存储分段元数据通知配置	是的。
S3： GetBucketNotification	GetBucketNotizationConfiguration	
S3： GetBucketObjectLockConfiguration	GetObjectLockConfiguration	
S3： GetBucketPolicy	GetBucketPolicy	
S3： GetBucketTagging	GetBucketTaging	
S3： GetBucketVersioning	GetBucketVersioning	
S3： GetLifeycleConfiguration	GetBucketLifecycleConfiguration	
S3： GetReplicationConfiguration	GetBucketReplication	
S3： ListAllMy桶	• List桶 • 获取存储使用量	是、对于GET存储使用情况。 注意：仅用于组策略。
S3： ListBucket	• ListObjects • HeadBucket • RestorEObject	
S3： ListBucketMultipartUploads	• ListMultipartUploads • RestorEObject	
S3： ListBucketVersions	获取存储分段版本	
S3： PutBucketCompliance	PUT 存储分段合规性（已弃用）	是的。

权限	**S3 REST API** 操作	为 **StorageGRID** 自定义
S3： PutBucketConsistency	PUT 存储分段一致性	是的。
S3： PutBucketCORS	• DeleteBucketCors†ñ a • PutBucketCors	
S3： PutEncryptionConfiguration	• DeleteBucketEncryption • PutBucketEncryption	
S3： PutBucketLastAccessTime	PUT 分段上次访问时间	是的。
S3： PutBucketMetadataNotification	PUT 存储分段元数据通知配置	是的。
S3： PutBucketNotification	PutBucketNotizationConfiguration	
S3： PutBucketObjectLockConfiguration	• CreateBucket `x-amz-bucket-object-lock-enabled: true` 请求标头(也需要S3：CreateBucket权限) • PutObjectLockConfiguration	
S3： PutBucketPolicy	PutBucketPolicy	
S3： PutBucketTagging	• DeleteBucketTbaging*LW_AT*† • PutBucketTaging	
S3： PutBucketVersioning	PutBucketVersioning	
S3： PutLifeycleConfiguration	• DeleteBucketLifecycle† • PutBucketLifecycleConfiguration	
S3： PutReplicationConfiguration	PutBucketReplication	可以、分开放置和删除权限

应用于对象的权限

权限	**S3 REST API** 操作	为 **StorageGRID** 自定义
S3： AbortMultipartUpload	• AbortMultipartUpload • RestorEObject	

权限	**S3 REST API** 操作	为 **StorageGRID** 自定义
S3：BypassGovernanceRetention	• DeleteObject • DeleteObjects • PutObject保留	
S3： DeleteObject	• DeleteObject • DeleteObjects • RestorEObject	
S3： DeleteObjectTagging	DeleteObjectTagging	
S3： DeleteObjectVersionTagging	DeleteObjectTaging(对象的特定版本)	
S3： DeleteObjectVersion	DeleteObject (对象的特定版本)	
S3： GetObject	• GetObject • HeadObject • RestorEObject • SelectObjectContent	
S3： GetObjectAcl	GetObjectAcl	
S3： GetObjectLegend	GetObjectLegalHold	
S3： GetObjectRetention	GetObject保留	
S3： GetObjectTagging	GetObjectTagging	
S3： GetObjectVersionTagging	GetObjectTaging(对象的特定版本)	
S3： GetObjectVersion	GetObject (对象的特定版本)	
S3： ListMultipartUploadPart	ListParts、RestorEObject	

权限	**S3 REST API** 操作	为 **StorageGRID** 自定义
S3： PutObject	• PutObject • CopyObject • RestorEObject • CreateMultipartUpload • CompleteMultipartUpload • 上传部件 • 上传PartCopy	
S3： PutObjectLegend	PutObjectLegalHold	
S3： PutObjectRetention	PutObject保留	
S3： PutObjectTagging	PutObjectTagging	
S3： PutObjectVersionTagging	PutObjectTaging(对象的特定版本)	
S3： PutOverwriteObject	• PutObject • CopyObject • PutObjectTagging • DeleteObjectTagging • CompleteMultipartUpload	是的。
S3： RestoreObject	RestorEObject	

使用 **PutOverwriteObject** 权限

S3： PutOverwriteObject 权限是一种自定义 StorageGRID 权限，适用场景 可通过此权限创建或更新对象。此权限的设置可确定客户端是否可以覆盖对象的数据，用户定义的元数据或 S3 对象标记。

此权限的可能设置包括：

- * 允许 *： 客户端可以覆盖对象。这是默认设置。
- **deny**:客户端无法覆盖对象。如果设置为 deny ，则 PutOverwriteObject 权限的工作原理如下：
  - 如果在同一路径中找到现有对象：
    - 无法覆盖对象的数据、用户定义的元数据或S3对象标记。
    - 正在执行的任何载入操作均会取消，并返回错误。
    - 如果启用了S3版本控制、则拒绝设置将阻止PutObjectTaging或DeleteObjectTaging操作修改对象及其非最新版本的标记集。
  - 如果未找到现有对象，此权限将不起作用。

- 如果不存在此权限，则效果与设置了 allow 时相同。

> 如果当前S3策略允许覆盖、并且PutOverwriteObject权限设置为deny、则客户端无法覆盖对象的数据、用户定义的元数据或对象标记。此外，如果选中了*禁止修改客户端*复选框(配置>*安全设置*>*网络和对象*)，则该设置将覆盖PutOverwriteObject权限的设置。

指定策略中的条件

条件用于定义策略何时生效。条件包括运算符和键值对。

条件使用键值对进行评估。一个条件元素可以包含多个条件，每个条件可以包含多个键值对。条件块使用以下格式：

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

在以下示例中， ipaddress 条件使用 SourceIp 条件密钥。

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

支持的条件运算符

条件运算符分为以下几类：

- string
- 数字
- 布尔值
- IP 地址
- 空检查

条件运算符	**Description**
StringEquals	根据完全匹配（区分大小写）将键与字符串值进行比较。
StringNotEquals	根据否定匹配（区分大小写）将键与字符串值进行比较。
StringEqualsIgnoreCase	根据完全匹配将键与字符串值进行比较（忽略大小写）。

条件运算符	Description
StringNotEqualsIgnoreCase	根据否定的匹配将键与字符串值进行比较（忽略大小写）。
StringLike	根据完全匹配（区分大小写）将键与字符串值进行比较。可以包括 * 和? 通配符。
StringNotLike	根据否定匹配（区分大小写）将键与字符串值进行比较。可以包括 * 和? 通配符。
数值方程式	根据精确匹配将键与数字值进行比较。
NumericNotEquals	根据否定匹配将键与数字值进行比较。
数值 GreaterThan	将键与基于"大于"匹配的数值进行比较。
NumericGreaterThals.	将键与基于"大于或等于"匹配的数值进行比较。
数值细小	将键与基于"小于"匹配的数值进行比较。
数值 ThalEquals	将键与基于"小于或等于"匹配的数值进行比较。
池	根据"true或false"匹配将键与布尔值进行比较。
IP 地址	将密钥与 IP 地址或 IP 地址范围进行比较。
NotIpAddress	根据否定匹配将密钥与 IP 地址或 IP 地址范围进行比较。
空	检查当前请求上下文中是否存在条件密钥。

支持的条件密钥

条件键	操作	Description
AWS ： 源 Ip	IP 运算符	将与发送请求的 IP 地址进行比较。可用于存储分段或对象操作。 • 注意： * 如果 S3 请求是通过管理节点和网关节点上的负载平衡器服务发送的，则此请求将与负载平衡器服务上游的 IP 地址进行比较。 • 注 * ： 如果使用第三方非透明负载平衡器，则此负载平衡器将与该负载平衡器的 IP 地址进行比较。任意 `X-Forwarded-For` 标头将被忽略、因为无法确定其有效性。

条件键	操作	Description
AWS ： 用户名	资源 / 身份	将与发送请求的发件人用户名进行比较。可用于存储分段或对象操作。
S3 ： 分隔符	S3 ： ListBucket 和 S3 ： ListBucketVersions 权限	将与在ListObjects或ListObjectVersies请求中指定的delifier参数进行比较。
S3：<tag-key>	S3 ： DeleteObjectTagging S3 ： DeleteObjectVersionTaggi ng S3 ： GetObject S3 ： GetObjectAcl 3：GetObjectTaging S3 ： GetObjectVersion S3：GetObjectVersionAcl S3 ： GetObjectVersionTagging S3：PutObjectAcl S3 ： PutObjectTagging S3：PutObjectVersion对象 S3 ： PutObjectVersionTagging	将要求现有对象具有特定的标记键和值。
S3 ： 最大密钥	S3 ： ListBucket 和 S3 ： ListBucketVersions 权限	将与ListObjects或ListObjectVersies请求中指定的最大键数参数进行比较。

条件键	操作	Description
S3： object-lock-real-retenation-days	S3： PutObject	与中指定的保留截止日期进行比较 `x-amz-object-lock-retain-until-date` 请求标头或根据存储分段默认保留期限计算得出、以确保这些值处于以下请求允许的范围内： • PutObject • CopyObject • CreateMultipartUpload
S3： object-lock-real-retenation-days	S3： PutObjectRetention	与PutObjectRetain请求中指定的保留截止日期进行比较、以确保该日期在允许的范围内。
S3：前缀	S3： ListBucket 和 S3： ListBucketVersions 权限	将与ListObjects或ListObjectVersies请求中指定的前缀参数进行比较。
S3：<tag-key>	S3： PutObject S3： PutObjectTagging S3： PutObjectVersionTagging	如果对象请求包含标记、则需要特定的标记密钥和值。

指定策略中的变量

您可以在策略中使用变量填充可用的策略信息。您可以在中使用策略变量 `Resource` 中的元素和字符串比较 `Condition` Element。

在此示例中、为变量 `${aws:username}` 是资源元素的一部分：

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

在此示例中、为变量 `${aws:username}` 是条件块中条件值的一部分：

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

变量	Description
`${aws:SourceIp}`	使用 SourceIp 键作为提供的变量。
`${aws:username}`	使用 username 密钥作为提供的变量。
`${s3:prefix}`	使用特定于服务的前缀密钥作为提供的变量。
`${s3:max-keys}`	使用特定于服务的 max-keys 键作为提供的变量。
`${*}`	特殊字符。使用字符作为文字 * 字符。
`${?}`	特殊字符。使用字符作为文字？字符。
`${$}`	特殊字符。使用字符作为文字 $ 字符。

创建需要特殊处理的策略

有时，策略可能会授予对安全性有危险或对持续操作（例如锁定帐户的 root 用户）有危险的权限。在策略验证期间， StorageGRID S3 REST API 实施的限制性要低于 Amazon ，但在策略评估期间同样严格。

策略问题描述	Policy type	Amazon 行为	StorageGRID 行为
拒绝向自己授予对 root 帐户的任何权限	存储分段	有效且强制实施，但 root 用户帐户保留所有 S3 存储分段策略操作的权限	相同
拒绝用户 / 组的任何权限	组	有效且强制实施	相同
允许外部帐户组拥有任何权限	存储分段	主体无效	有效，但如果某个策略允许，则所有 S3 存储分段策略操作的权限均会返回 405 Method not allowed 错误
允许外部帐户 root 或用户拥有任何权限	存储分段	有效，但如果某个策略允许，则所有 S3 存储分段策略操作的权限均会返回 405 Method not allowed 错误	相同
允许所有人对所有操作拥有权限	存储分段	有效，但对所有 S3 存储分段策略操作的权限会为外部帐户 root 和用户返回 405 Method not allowed 错误	相同

策略问题描述	**Policy type**	**Amazon** 行为	**StorageGRID** 行为
拒绝任何人对所有操作的权限	存储分段	有效且强制实施，但 root 用户帐户保留所有 S3 存储分段策略操作的权限	相同
主体是不存在的用户或组	存储分段	主体无效	有效
资源不是 S3 存储分段	组	有效	相同
主体是一个本地组	存储分段	主体无效	有效
策略授予非所有者帐户(包括匿名帐户)放置对象的权限。	存储分段	有效。对象由创建者帐户拥有，并且存储分段策略不适用。创建者帐户必须使用对象 ACL 为对象授予访问权限。	有效。对象由存储分段所有者帐户拥有。存储分段策略适用。

一次写入多读（ **WORM** ）保护

您可以创建一次写入多读（ Write Once Read-Many ， WORM ）分段来保护数据，用户定义的对象元数据和 S3 对象标记。您可以配置 WORM 分段，以便创建新对象并防止覆盖或删除现有内容。请使用此处所述的方法之一。

为了确保覆盖始终被拒绝，您可以：

- 在网格管理器中，转到*configuration*>*Security*>*Security settings **>*Network and objects**，然后选中*prevent client修改*复选框。
- 应用以下规则和 S3 策略：
  - 向 S3 策略添加 PutOverwriteObject deny 操作。
  - 将 DeleteObject deny 操作添加到 S3 策略中。
  - 将PutObject Allow操作添加到S3策略中。

> 在S3策略中将DeleteObject设置为deny不会阻止ILM在存在"30天后将副本置零"等规则时删除对象。

> 即使应用了所有这些规则和策略、它们也无法防止并发写入(请参见情形A)。它们可以防止顺序完成的覆盖（请参见情况 B ）。

- 情形 A*： 并发写入（不受保护）

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

- 情形 B*： 顺序完成的覆盖（防止）

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

相关信息

- "StorageGRID ILM 规则如何管理对象"
- "存储分段策略示例"
- "组策略示例"
- "使用 ILM 管理对象"
- "使用租户帐户"

存储分段策略示例

使用本节中的示例为分段构建StorageGRID 访问策略。

存储分段策略用于指定附加此策略的存储分段的访问权限。存储分段策略使用 S3 PutBucketPolicy API 进行配置。请参见 "对存储分段执行的操作"。

可以按照以下命令使用 AWS 命令行界面配置存储分段策略：

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

示例：允许每个人对某个存储分段进行只读访问

在此示例中、允许所有人(包括匿名用户)列出分段中的对象、并对分段中的所有对象执行GetObject操作。所有其他操作都将被拒绝。请注意、此策略可能并不特别有用、因为除了帐户root之外、没有其他人有权向存储分段写入数据。

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

示例：允许一个帐户中的每个人完全访问某个存储分段，而另一帐户中的每个人只读访问某个存储分段

在此示例中、一个指定帐户中的每个人都可以完全访问某个存储分段、而另一个指定帐户中的每个人只能列出存储分段并对以开头的存储分段中的对象执行GetObject操作 `shared/` 对象密钥前缀。

> 在 StorageGRID 中，非所有者帐户创建的对象（包括匿名帐户）归存储分段所有者帐户所有。存储分段策略适用场景 这些对象。

```
{
  "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/*"
        }
      }
    }
  ]
}
```

示例：允许每个人对某个存储分段进行只读访问，并允许指定组进行完全访问

在此示例中、允许包括匿名用户在内的所有人列出分段并对分段中的所有对象执行GetObject操作、而仅限属于组的用户 `Marketing` 在指定帐户中、允许完全访问。

```
{
  "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/*"
      ]
    }
  ]
}
```

示例：如果客户端位于 **IP** 范围内，则允许每个人对存储分段进行读写访问

在此示例中，允许包括匿名用户在内的所有人列出存储分段并对存储分段中的所有对象执行任何对象操作，前提是这些请求来自指定的 IP 范围（ 54.240.143.0 到 54.240.143.255 ，但 54.240.143.188 除外）。所有其他操作都将被拒绝，并且 IP 范围以外的所有请求都将被拒绝。

```
{
  "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"}
      }
    }
  ]
}
```

示例：允许指定的联合用户完全访问某个存储分段

在此示例中、允许联合用户Alex完全访问 `examplebucket` 存储分段及其对象。包括 "`root` " 在内的所有其他用户均被明确拒绝所有操作。但请注意， "`root` " 从不会被拒绝 PUT ， Get/DeleteBucketPolicy 的权限。

```
{
  "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/*"
      ]
    }
  ]
}
```

示例： **PutOverwriteObject** 权限

在此示例中、将显示 `Deny` 对PutOverwriteObject和DeleteObject的影响可确保任何人都不能覆盖或删除对象的数据、用户定义的元数据和S3对象标记。

```
{
  "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/*"
    }
  ]
}
```

组策略示例

使用本节中的示例为组构建StorageGRID 访问策略。

组策略用于指定附加此策略的组的访问权限。没有 `Principal` 元素、因为它是隐式的。组策略可使用租户管理器或 API 进行配置。

示例：使用租户管理器设置组策略

在租户管理器中添加或编辑组时、您可以选择组策略来确定此组的成员将具有哪些S3访问权限。请参见 "为 S3 租户创建组"。

- * 无 S3 访问 *：默认选项。此组中的用户无权访问S3资源、除非使用存储分段策略授予访问权限。如果选择此选项，则默认情况下，只有 root 用户才能访问 S3 资源。
- * 只读访问 *：此组中的用户对 S3 资源具有只读访问权限。例如，此组中的用户可以列出对象并读取对象数据，元数据和标记。选择此选项后，只读组策略的 JSON 字符串将显示在文本框中。您无法编辑此字符串。
- * 完全访问 *：此组中的用户对 S3 资源（包括分段）具有完全访问权限。选择此选项后，完全访问组策略的 JSON 字符串将显示在文本框中。您无法编辑此字符串。
- **Ransmans**要 缓解：此示例策略适用场景 all b分 段for this租户。此组中的用户可以执行常见操作、但无法从启用了对象版本控制的分段中永久删除对象。

  具有"管理所有存储分段"权限的租户管理器用户可以覆盖此组策略。将"管理所有分段"权限限制为受信任用户、并在可用时使用多因素身份验证(Multi-FactorAuthentication、MFA)。

- * 自定义 *：组中的用户将获得您在文本框中指定的权限。

示例：允许组完全访问所有存储分段

在此示例中，除非 bucket 策略明确拒绝，否则允许组中的所有成员对租户帐户拥有的所有分段进行完全访问。

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

示例：允许组对所有分段进行只读访问

在此示例中，组的所有成员都对 S3 资源具有只读访问权限，除非 bucket 策略明确拒绝。例如，此组中的用户可以列出对象并读取对象数据，元数据和标记。

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

示例：仅允许组成员对存储分段中的"文件夹"具有完全访问权限

在此示例中，组成员只能列出并访问指定存储分段中的特定文件夹（密钥前缀）。请注意，在确定其他组策略和存储分段策略的隐私时，应考虑这些文件夹的访问权限。

```
{
  "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}/*"
    }
  ]
}
```

审核日志中跟踪的 **S3** 操作

审核消息由 StorageGRID 服务生成并存储在文本日志文件中。您可以在审核日志中查看特定于S3的审核消息、以获取有关分段和对象操作的详细信息。

审核日志中跟踪的存储分段操作

- CreateBucket
- DeleteBucket
- DeleteBucketTbaging
- DeleteObjects
- GetBucketTaging
- HeadBucket
- ListObjects
- ListObjectVersies
- PUT 存储分段合规性
- PutBucketTaging
- PutBucketVersioning

审核日志中跟踪的对象操作

- CompleteMultipartUpload
- CopyObject
- DeleteObject
- GetObject
- HeadObject
- PutObject
- RestorEObject
- 选择对象
- UploadPart (当ILM规则使用平衡或严格的加载时)
- UploadPartCopy (当ILM规则使用平衡或严格的加载时)

相关信息

- "访问审核日志文件"
- "客户端写入审核消息"
- "客户端读取审核消息"

版权信息

版权所有 © 2025 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。 美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

商标信息

NetApp、NetApp 标识和 http://www.netapp.com/TM 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。