S3-REST-API VERWENDEN

StorageGRID

NetApp
March 12, 2025

This PDF was generated from https://docs.netapp.com/de-de/storagegrid/s3/index.html on March 12, 2025. Always check docs.netapp.com for the latest.

Inhalt

S3-REST-API VERWENDEN

Von S3 REST API unterstützte Versionen und Updates

StorageGRID unterstützt die S3-API (Simple Storage Service), die als Satz Rest-Web-Services (Representational State Transfer) implementiert wird.

Dank der Unterstützung für die S3-REST-API können serviceorientierte Applikationen, die für S3-Web-Services entwickelt wurden, mit On-Premises-Objekt-Storage verbunden werden, der das StorageGRID-System verwendet. Es sind minimale Änderungen an der aktuellen Nutzung von S3-REST-API-Aufrufen einer Client-Applikation erforderlich.

Unterstützte Versionen

StorageGRID unterstützt die folgenden spezifischen Versionen von S3 und HTTP.

Element	Version
S3-API-Spezifikation	"Amazon Web Services (AWS) Dokumentation: Amazon Simple Storage Service API Reference"
HTTP	1,1 Weitere Informationen zu HTTP finden Sie unter HTTP/1.1 (RFCs 7230-35). "IETF RFC 2616: Hypertext Transfer Protocol (HTTP/1.1)" **Hinweis**: StorageGRID unterstützt HTTP/1.1-Pipelining nicht.

Updates für die S3-REST-API-Unterstützung

Freigabe	Kommentare
11,9	• Unterstützung für vorberechnete SHA-256-Prüfsummenwerte für die folgenden Anforderungen und unterstützten Header wurde hinzugefügt. Mit dieser Funktion können Sie die Integrität hochgeladener Objekte überprüfen: ◦ CompleteMultipartUpload: `x-amz-checksum-sha256` ◦ CreateMultipartUpload: `x-amz-checksum-algorithm` ◦ GetObject: `x-amz-checksum-mode` ◦ Kopfobjekt: `x-amz-checksum-mode` ◦ ListenTeile ◦ PutObject: `x-amz-checksum-sha256` ◦ UploadPart: `x-amz-checksum-sha256` • Der Grid-Administrator kann die Aufbewahrungs- und Compliance-Einstellungen auf Mandantenebene kontrollieren. Diese Einstellungen wirken sich auf die Einstellungen der S3-Objektsperre aus. ◦ Standardaufbewahrungsmodus und Objektaufbewahrungsmodus mit Buckets: Governance oder Compliance, sofern vom Grid-Administrator zugelassen. ◦ Standardaufbewahrungszeitraum für Bucket und Objektaufbewahrung bis Datum: Muss kleiner oder gleich dem sein, was durch den vom Grid-Administrator festgelegten maximalen Aufbewahrungszeitraum zulässig ist. • Verbesserte Unterstützung von `aws-chunked` Kodierungs- und Streaming-Werten für Inhalte `x-amz-content-sha256`. Einschränkungen: ◦ Falls vorhanden, `chunk-signature` ist optional und nicht validiert ◦ Wenn vorhanden, `x-amz-trailer` wird der Inhalt ignoriert
11,8	Die Namen der S3-Vorgänge wurden aktualisiert, um sie mit den in der verwendeten Namen "Amazon Web Services (AWS) Dokumentation: Amazon Simple Storage Service API Reference" zu vergleichen.
11,7	• Hinzugefügt "Schnelle Referenz: Unterstützte S3-API-Anforderungen". • Zusätzliche Unterstützung für die Verwendung DES GOVERNANCE-Modus mit S3 Object Lock. • Unterstützung für den StorageGRID-spezifischen Antwortheader für GET Object- und HEAD-Objektanforderungen wurde hinzugefügt `x-ntap-sg-cgr-replication-status`. Dieser Header stellt den Replikationsstatus eines Objekts für die Grid-übergreifende Replikation bereit. • SelectObjectContent Requests unterstützen nun Parkett-Objekte.

Freigabe	Kommentare
11,6	• Unterstützung für die Verwendung des Anforderungsparameters in GET Object und HEAD Object Requests hinzugefügt `partNumber`. • Zusätzliche Unterstützung für einen Standardaufbewahrungsmodus und einen Standardaufbewahrungszeitraum auf Bucket-Ebene für S3 Object Lock. • Unterstützung für den Richtlinienzustandsschlüssel hinzugefügt `s3:object-lock-remaining-retention-days`, um den Bereich der zulässigen Aufbewahrungsfristen für Ihre Objekte festzulegen. • Die maximale *recommended*-Größe für einen einzelnen PUT-Objekt-Vorgang wurde auf 5 gib (5,368,709,120 Bytes) geändert. Wenn Sie über Objekte mit einer Größe von mehr als 5 gib verfügen, verwenden Sie stattdessen mehrteilige Uploads.
11,5	• Zusätzliche Unterstützung für das Management der Bucket-Verschlüsselung • Unterstützung für S3 Object Lock und veraltete ältere Compliance-Anforderungen wurde hinzugefügt. • Zusätzliche Unterstützung beim LÖSCHEN mehrerer Objekte in versionierten Buckets. • Der `Content-MD5` Anforderungskopf wird jetzt korrekt unterstützt.
11,4	• Unterstützung für DELETE Bucket-Tagging, GET Bucket-Tagging und PUT Bucket-Tagging. Kostenzuordnungstags werden nicht unterstützt. • Bei in StorageGRID 11.4 erstellten Buckets ist keine Beschränkung der Objektschlüsselnamen auf Performance-Best-Practices mehr erforderlich. • Unterstützung für Bucket-Benachrichtigungen für den Ereignistyp hinzugefügt `s3:ObjectRestore:Post`. • Die Größenbeschränkungen von AWS für mehrere Teile werden nun durchgesetzt. Jedes Teil eines mehrteiligen Uploads muss zwischen 5 MiB und 5 gib liegen. Der letzte Teil kann kleiner als 5 MiB sein. • Unterstützung für TLS 1.3 hinzugefügt
11,3	• Zusätzliche Unterstützung für serverseitige Verschlüsselung von Objektdaten mit vom Kunden bereitgestellten Schlüsseln (SSE-C). • Unterstützung für DIE Lebenszyklusoperationen „DELETE“, „GET“ und „PUT“ (nur Ablaufaktion) und für den Antwortheader hinzugefügt `x-amz-expiration`. • Aktualisiertes PUT-Objekt, PUT-Objekt – Copy und Multipart-Upload, um die Auswirkungen von ILM-Regeln zu beschreiben, die synchrone Platzierung bei der Aufnahme verwenden. • TLS 1.1-Chiffren werden nicht mehr unterstützt.

Freigabe	Kommentare
11,2	Unterstützung für DIE WIEDERHERSTELLUNG NACH Objekten wurde hinzugefügt und kann in Cloud-Storage-Pools verwendet werden. Unterstützung für die Verwendung der AWS-Syntax für ARN, Richtlinienzustandsschlüssel und Richtlinienvariablen in Gruppen- und Bucket-Richtlinien Vorhandene Gruppen- und Bucket-Richtlinien, die die StorageGRID-Syntax verwenden, werden weiterhin unterstützt. **Hinweis:** die Verwendung von ARN/URN in anderen Konfigurationen JSON/XML, einschließlich derjenigen, die in benutzerdefinierten StorageGRID-Funktionen verwendet werden, hat sich nicht geändert.
11,1	Zusätzliche Unterstützung für die Cross-Origin Resource Sharing (CORS), HTTP für S3-Clientverbindungen zu Grid-Nodes und Compliance-Einstellungen für Buckets.
11,0	Unterstützung für die Konfiguration von Plattform-Services (CloudMirror Replizierung, Benachrichtigungen und Elasticsearch-Integration) für Buckets. Außerdem wurden die Unterstützung für Objekt-Tagging-Speicherortbeschränkungen für Buckets und die verfügbare Konsistenz hinzugefügt.
10,4	Unterstützung für ILM-Scanning-Änderungen an Versionierung, Seitenaktualisierungen von Endpoint Domain-Namen, Bedingungen und Variablen in Richtlinien, Richtlinienbeispiele und die Berechtigung PutOverwriteObject.
10,3	Zusätzliche Unterstützung für Versionierung
10,2	Unterstützung für Gruppen- und Bucket-Zugriffsrichtlinien und für mehrteilige Kopien (Upload Part - Copy) hinzugefügt
10,1	Unterstützung für mehrteilige Uploads, virtuelle Hosted-Style-Anforderungen und v4 Authentifizierung
10,0	Die erste Unterstützung der S3-REST-API durch das StorageGRID-System.die derzeit unterstützte Version der *Simple Storage Service API Reference* lautet 2006-03-01.

Schnelle Referenz: Unterstützte S3-API-Anforderungen

Auf dieser Seite wird zusammengefasst, wie StorageGRID Amazon Simple Storage Service (S3) APIs unterstützt.

Diese Seite umfasst nur die S3-Vorgänge, die von StorageGRID unterstützt werden.

Um die AWS Dokumentation für jeden Vorgang anzuzeigen, klicken Sie in der Überschrift auf den Link.

Allgemeine URI-Abfrageparameter und Anforderungsheader

Sofern nicht angegeben, werden die folgenden gängigen URI-Abfrageparameter unterstützt:

- `versionId` (Bei Bedarf für Objekt-Operationen)

Sofern nicht anders angegeben, werden die folgenden gängigen Anforderungsheader unterstützt:

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

Verwandte Informationen

- "Details zur S3-REST-API-Implementierung"
- "Amazon Simple Storage Service API-Referenz: Common Request Header"

"AbortMehrteilaUpload"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung sowie den zusätzlichen URI-Abfrageparameter:

- `uploadId`

Text anfordern

Keine

StorageGRID-Dokumentation

"Vorgänge für mehrteilige Uploads"

"CompleteMultipartUpload"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung sowie den zusätzlichen URI-Abfrageparameter:

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

Text-XML-Tags anfordern

StorageGRID unterstützt folgende XML-Tags für Anforderungstext:

- `ChecksumSHA256`
- `CompleteMultipartUpload`
- `ETag`
- `Part`
- `PartNumber`

StorageGRID-Dokumentation

"CompleteMultipartUpload"

"CopyObject"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung sowie die folgenden zusätzlichen Kopfzeilen:

- `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>`

Text anfordern

Keine

StorageGRID-Dokumentation

"CopyObject"

"CreateBucket"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung sowie die folgenden zusätzlichen Kopfzeilen:

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

Text anfordern

StorageGRID unterstützt alle Parameter des Abforderungskörpers, die zum Zeitpunkt der Implementierung von der Amazon S3 REST-API definiert wurden.

StorageGRID-Dokumentation

"Operationen auf Buckets"

"CreateMultipartUpload"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung sowie die folgenden zusätzlichen Kopfzeilen:

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

Text anfordern

Keine

StorageGRID-Dokumentation

"CreateMultipartUpload"

"DeleteBucket"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

StorageGRID-Dokumentation

"Operationen auf Buckets"

"DeleteBucketCors"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"DeleteBucketEncryption"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"DeleteBucketLifecycle"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

- "Operationen auf Buckets"
- "S3-Lebenszykluskonfiguration erstellen"

"DeleteBucketRichtlinien"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"DeleteBucketReplication"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"DeleteBucketTagging"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"DeleteObject"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung plus den folgenden zusätzlichen Anforderungsheader:

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

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen für Objekte"

"Objekte deObjekteObjekte"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung plus den folgenden zusätzlichen Anforderungsheader:

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

Text anfordern

StorageGRID unterstützt alle Parameter des Abforderungskörpers, die zum Zeitpunkt der Implementierung von der Amazon S3 REST-API definiert wurden.

StorageGRID-Dokumentation

"Operationen für Objekte"

"DeleteObjectTagging"

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen für Objekte"

"GetBucketAcl"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"GetBucketCors"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"GetBucketEncryption"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"GetBucketLifecycleKonfiguration"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

- "Operationen auf Buckets"
- "S3-Lebenszykluskonfiguration erstellen"

"GetBucketLocation"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"GetBucketNotificationConfiguration"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"GetBucketPolicy"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"GetBucketReplication"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"GetBucketTagging"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"GetBucketVersioning"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"GetObject"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilen für diese Anforderung plus die folgenden zusätzlichen URI-Abfrageparameter:

- `x-amz-checksum-mode`
- `partNumber`
- `response-cache-control`
- `response-content-disposition`

- `response-content-encoding`
- `response-content-language`
- `response-content-type`
- `response-expires`

Und diese zusätzlichen Anforderungsheader:

- `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`

Text anfordern

Keine

StorageGRID-Dokumentation

"GetObject"

"GetObjectAcl"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen für Objekte"

"GetObjectLegalHold"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"

"GetObjectLockConfiguration"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"

"GetObjectRetention"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"

"GetObjectTagging"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen für Objekte"

"HeadBucket"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"HeadObject"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung sowie die folgenden zusätzlichen Kopfzeilen:

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

Text anfordern

Keine

StorageGRID-Dokumentation

"HeadObject"

"ListBuchs"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Keine

StorageGRID-Dokumentation

Operationen für den Dienst › ListBuckets

"ListMultipartUploads"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung sowie die folgenden zusätzlichen Parameter:

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

Text anfordern

Keine

StorageGRID-Dokumentation

"ListMultipartUploads"

"ListObjekte"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung sowie die folgenden zusätzlichen Parameter:

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

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"ListObjekteV2"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung sowie die folgenden zusätzlichen Parameter:

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

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"ListObjectVersions"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung sowie die folgenden zusätzlichen Parameter:

- `delimiter`

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

Text anfordern

Keine

StorageGRID-Dokumentation

"Operationen auf Buckets"

"ListenTeile"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung sowie die folgenden zusätzlichen Parameter:

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

Text anfordern

Keine

StorageGRID-Dokumentation

"ListMultipartUploads"

"PutBucketCors"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

StorageGRID unterstützt alle Parameter des Abforderungskörpers, die zum Zeitpunkt der Implementierung von der Amazon S3 REST-API definiert wurden.

StorageGRID-Dokumentation

"Operationen auf Buckets"

"PutBucketEncryption"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text-XML-Tags anfordern

StorageGRID unterstützt folgende XML-Tags für Anforderungstext:

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

StorageGRID-Dokumentation

"Operationen auf Buckets"

"PutBucketLifecycleKonfiguration"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text-XML-Tags anfordern

StorageGRID unterstützt folgende XML-Tags für Anforderungstext:

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

StorageGRID-Dokumentation

- "Operationen auf Buckets"
- "S3-Lebenszykluskonfiguration erstellen"

"PutBucketNotificationKonfiguration"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text-XML-Tags anfordern

StorageGRID unterstützt folgende XML-Tags für Anforderungstext:

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

StorageGRID-Dokumentation

"Operationen auf Buckets"

"PutBucketPolicy"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Weitere Informationen zu den unterstützten JSON-Textfeldern finden Sie unter "Verwendung von Bucket- und Gruppenzugriffsrichtlinien".

"PutBucketReplication"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text-XML-Tags anfordern

- `Bucket`
- `Destination`
- `Prefix`
- `ReplicationConfiguration`

- `Rule`
- `Status`
- `StorageClass`

StorageGRID-Dokumentation

"Operationen auf Buckets"

"PutBucketTagging"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

StorageGRID unterstützt alle Parameter des Abforderungskörpers, die zum Zeitpunkt der Implementierung von der Amazon S3 REST-API definiert wurden.

StorageGRID-Dokumentation

"Operationen auf Buckets"

"PutBucketVersioning"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Body-Parameter anfordern

StorageGRID unterstützt die folgenden Parameter des Anfragenkörpers:

- `VersioningConfiguration`
- `Status`

StorageGRID-Dokumentation

"Operationen auf Buckets"

"PutObject"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung sowie die folgenden zusätzlichen Kopfzeilen:

- `Cache-Control`
- `Content-Disposition`
- `Content-Encoding`
- `Content-Language`
- `x-amz-checksum-sha256`
- `x-amz-server-side-encryption`
- `x-amz-storage-class`

- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`
- `x-amz-tagging`
- `x-amz-object-lock-mode`
- `x-amz-object-lock-retain-until-date`
- `x-amz-object-lock-legal-hold`
- `x-amz-meta-<metadata-name>`

Text anfordern

- Binäre Daten des Objekts

StorageGRID-Dokumentation

"PutObject"

"PutObjectLegalHold"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

StorageGRID unterstützt alle Parameter des Abforderungskörpers, die zum Zeitpunkt der Implementierung von der Amazon S3 REST-API definiert wurden.

StorageGRID-Dokumentation

"Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"

"PutObjectLockKonfiguration"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

StorageGRID unterstützt alle Parameter des Abforderungskörpers, die zum Zeitpunkt der Implementierung von der Amazon S3 REST-API definiert wurden.

StorageGRID-Dokumentation

"Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"

"PutObjectRetention"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung sowie diesen zusätzlichen Header:

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

Text anfordern

StorageGRID unterstützt alle Parameter des Abforderungskörpers, die zum Zeitpunkt der Implementierung von der Amazon S3 REST-API definiert wurden.

StorageGRID-Dokumentation

"Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"

"PutObjectTagging"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

StorageGRID unterstützt alle Parameter des Abforderungskörpers, die zum Zeitpunkt der Implementierung von der Amazon S3 REST-API definiert wurden.

StorageGRID-Dokumentation

"Operationen für Objekte"

"Objekt restoreObject"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Weitere Informationen zu den unterstützten Körperfeldern finden Sie unter "Objekt restoreObject".

"SelektierObjectContent"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilendieser Anforderung.

Text anfordern

Weitere Informationen zu den unterstützten Textfeldern finden Sie in den folgenden Informationen:

- "Verwenden Sie S3 Select"
- "SelektierObjectContent"

"UploadTeil"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilen für diese Anforderung plus die folgenden zusätzlichen URI-Abfrageparameter:

- `partNumber`
- `uploadId`

Und diese zusätzlichen Anforderungsheader:

- `x-amz-checksum-sha256`

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

Text anfordern

- Binäre Daten des Teils

StorageGRID-Dokumentation

"UploadTeil"

"UploadPartCopy"

URI-Abfrageparameter und Anforderungskopfzeilen

StorageGRID unterstützt alle Allgemeine Parameter und Kopfzeilen für diese Anforderung plus die folgenden zusätzlichen URI-Abfrageparameter:

- `partNumber`
- `uploadId`

Und diese zusätzlichen Anforderungsheader:

- `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`

Text anfordern

Keine

StorageGRID-Dokumentation

"UploadPartCopy"

Testen der S3-REST-API-Konfiguration

Sie können die Amazon Web Services Command Line Interface (AWS CLI) verwenden,

um die Verbindung zum System zu testen und zu überprüfen, ob Objekte gelesen und geschrieben werden können.

Bevor Sie beginnen

- Sie haben die AWS CLI von heruntergeladen und installiert "aws.amazon.com/cli".
- Optional haben Sie "Ein Load Balancer-Endpunkt wurde erstellt". Andernfalls kennen Sie die IP-Adresse des zu verbindenden Storage-Node und die zu verwendende Port-Nummer. Siehe "IP-Adressen und Ports für Client-Verbindungen".
- Sie haben "S3-Mandantenkonto wurde erstellt".
- Sie haben sich beim Mieter und angemeldet"Zugriffsschlüssel erstellt".

Weitere Informationen zu diesen Schritten finden Sie unter "Client-Verbindungen konfigurieren".

Schritte

1. Konfigurieren Sie die AWS-CLI-Einstellungen so, dass das im StorageGRID-System erstellte Konto verwendet wird:
   a. Konfigurationsmodus aufrufen: `aws configure`
   b. Geben Sie die Zugriffsschlüssel-ID für das von Ihnen erstellte Konto ein.
   c. Geben Sie den geheimen Zugriffsschlüssel für das von Ihnen erstellte Konto ein.
   d. Geben Sie die Standardregion ein, die verwendet werden soll. `us-east-1`Beispiel: .
   e. Geben Sie das zu verwendende Standardausgabeformat ein, oder drücken Sie **Enter**, um JSON auszuwählen.
2. Erstellen eines Buckets:

   In diesem Beispiel wird davon ausgegangen, dass Sie einen Load Balancer-Endpunkt für die Verwendung der IP-Adresse 10.96.101.17 und des Ports 10443 konfiguriert haben.

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

   Wenn der Bucket erfolgreich erstellt wurde, wird der Speicherort des Buckets zurückgegeben, wie im folgenden Beispiel zu sehen:

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

3. Hochladen eines Objekts.

   ```
   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
   ```

   Wenn das Objekt erfolgreich hochgeladen wurde, wird ein ETAG zurückgegeben, der ein Hash der Objektdaten ist.

4. Listen Sie den Inhalt des Buckets auf, um zu überprüfen, ob das Objekt hochgeladen wurde.

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

5. Löschen Sie das Objekt.

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

6. Löschen Sie den Bucket.

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

So implementiert StorageGRID die S3-REST-API

In Konflikt stehende Clientanforderungen

Widersprüchliche Clientanforderungen, wie z. B. zwei Clients, die in denselben Schlüssel schreiben, werden auf der Grundlage der „neuesten Wins“ gelöst.

Der Zeitpunkt für die Bewertung „neuester Erfolge“ basiert auf dem Zeitpunkt, an dem das StorageGRID System eine bestimmte Anforderung abgeschlossen hat und nicht auf dem Zeitpunkt, an dem S3-Clients einen Vorgang starten.

Konsistenzwerte

Konsistenz bietet ein Gleichgewicht zwischen der Verfügbarkeit der Objekte und der Konsistenz dieser Objekte über verschiedene Storage-Nodes und Standorte hinweg. Sie können die Konsistenz entsprechend den Anforderungen Ihrer Anwendung ändern.

Standardmäßig garantiert StorageGRID eine Lese-/Nachher-Konsistenz für neu erstellte Objekte. Jeder GET nach einem erfolgreich abgeschlossenen PUT wird in der Lage sein, die neu geschriebenen Daten zu lesen. Überschreibungen vorhandener Objekte, Metadatenaktualisierungen und -Löschungen sind schließlich konsistent. Überschreibungen dauern in der Regel nur wenige Sekunden oder Minuten, können jedoch bis zu 15 Tage in Anspruch nehmen.

Wenn Sie Objektoperationen mit einer anderen Konsistenz durchführen möchten, haben Sie folgende Möglichkeiten:

- Geben Sie eine Konsistenz für Jeden Eimer.
- Geben Sie eine Konsistenz für Jeder API-Vorgang.
- Ändern Sie die standardmäßige Konsistenz für das gesamte Grid, indem Sie eine der folgenden Aufgaben ausführen:

- Gehen Sie im Grid Manager zu **CONFIGURATION** > **System** > **Storage settings** > **Default Consistency**.
- .

> Eine Änderung der Konsistenz für das gesamte Grid gilt nur für Buckets, die nach der Änderung der Einstellung erstellt wurden. Informationen zu den Details einer Änderung finden Sie im Auditprotokoll unter `/var/local/log` (Suche nach **consistenzLevel**).

Konsistenzwerte

Die Konsistenz wirkt sich auf die Verteilung der Metadaten, die StorageGRID zum Nachverfolgen von Objekten verwendet, auf die Nodes aus und damit auf die Verfügbarkeit von Objekten für Client-Anforderungen.

Sie können die Konsistenz für einen Bucket oder eine API-Operation auf einen der folgenden Werte festlegen:

- **All**: Alle Knoten erhalten die Daten sofort, oder die Anfrage schlägt fehl.
- **Strong-global**: Garantiert Lese-nach-Schreiben-Konsistenz für alle Client-Anfragen über alle Standorte hinweg.
- **Strong-site**: Garantiert Lese-nach-Schreiben Konsistenz für alle Client-Anfragen innerhalb einer Site.
- **Read-after-New-write**: (Standard) bietet Read-after-write-Konsistenz für neue Objekte und eventuelle Konsistenz für Objektaktualisierungen. Hochverfügbarkeit und garantierte Datensicherung Empfohlen für die meisten Fälle.
- **Verfügbar**: Bietet eventuelle Konsistenz für neue Objekte und Objekt-Updates. Verwenden Sie für S3-Buckets nur nach Bedarf (z. B. für einen Bucket mit Protokollwerten, die nur selten gelesen werden, oder für HEAD- oder GET-Vorgänge für nicht vorhandene Schlüssel). Nicht unterstützt für S3 FabricPool-Buckets.

Verwenden Sie die Konsistenz „Read-after-New-write“ und „available“

Wenn ein HEAD- oder GET-Vorgang die Konsistenz von Read-after-New-write verwendet, führt StorageGRID die Suche in mehreren Schritten durch:

- Es sieht zunächst das Objekt mit einer niedrigen Konsistenz.
- Wenn diese Suche fehlschlägt, wiederholt sie die Suche beim nächsten Konsistenzwert, bis sie eine Konsistenz erreicht, die dem Verhalten für Strong-Global entspricht.

Wenn eine HEAD- oder GET-Operation die Konsistenz „Read-after-New-write“ verwendet, das Objekt aber nicht existiert, erreicht die Objekt-Lookup immer eine Konsistenz, die dem Verhalten für strong-global entspricht. Da für diese Konsistenz mehrere Kopien der Objektmetadaten an jedem Standort verfügbar sein müssen, können Sie eine hohe Anzahl von 500 internen Serverfehlern erhalten, wenn zwei oder mehr Storage-Nodes am selben Standort nicht verfügbar sind.

Sofern Sie keine Konsistenzgarantien ähnlich Amazon S3 benötigen, können Sie diese Fehler für HEAD- und GET-Operationen verhindern, indem Sie die Konsistenz auf „verfügbar“ setzen. Wenn ein HEAD- oder GET-Betrieb die „verfügbare“ Konsistenz verwendet, bietet StorageGRID letztendlich nur Konsistenz. Bei einem fehlgeschlagenen Vorgang wird nicht erneut versucht, die Konsistenz zu erhöhen, daher müssen nicht mehrere Kopien der Objekt-Metadaten verfügbar sein.

Geben Sie die Konsistenz für den API-Vorgang an

Um die Konsistenz für eine individuelle API-Operation festzulegen, müssen die Konsistenzwerte für den Vorgang unterstützt werden, und Sie müssen die Konsistenz in der Anforderungsheader angeben. In diesem Beispiel wird die Konsistenz für eine GetObject-Operation auf „strong-site“ gesetzt.

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

Sie müssen für die PutObject- und GetObject-Operationen dieselbe Konsistenz verwenden.

Geben Sie die Konsistenz für Bucket an

Zum Festlegen der Konsistenz für Bucket können Sie die StorageGRID-Anforderung verwenden"PUT Bucket-Konsistenz". Sie können dies aber auch "Ändern der Konsistenz eines Buckets"über den Tenant Manager tun.

Beachten Sie beim Festlegen der Konsistenz für einen Bucket Folgendes:

- Durch das Festlegen der Konsistenz für einen Bucket wird bestimmt, welche Konsistenz für S3-Vorgänge verwendet wird, die an den Objekten in der Bucket oder in der Bucket-Konfiguration durchgeführt werden. Er hat keine Auswirkungen auf die Vorgänge auf dem Bucket selbst.
- Die Konsistenz einer einzelnen API-Operation überschreibt die Konsistenz für den Bucket.
- Im Allgemeinen sollten Buckets die Standardkonsistenz „Read-after-New-write“ verwenden. Wenn die Anforderungen nicht korrekt funktionieren, ändern Sie das Client-Verhalten der Anwendung, wenn möglich. Oder konfigurieren Sie den Client so, dass die Konsistenz für jede API-Anforderung angegeben wird. Legen Sie die Konsistenz auf Bucket-Ebene nur als letzte Option fest.

wie Konsistenz- und ILM-Regeln interagieren, um den Datenschutz zu beeinträchtigen

Sowohl Ihre Wahl der Konsistenz als auch Ihre ILM-Regel beeinflussen die Art und Weise, wie Objekte geschützt werden. Diese Einstellungen können interagieren.

Beispielsweise wirkt sich die bei der Speicherung eines Objekts verwendete Konsistenz auf die anfängliche Platzierung von Objekt-Metadaten aus, während das für die ILM-Regel ausgewählte Aufnahmeverhalten sich auf die anfängliche Platzierung von Objektkopien auswirkt. StorageGRID benötigt zur Erfüllung von Clientanfragen Zugriff auf die Metadaten und die Daten eines Objekts. Durch die Auswahl einer passenden Sicherungsstufe für die Konsistenz und das Aufnahmeverhalten können die Daten am Anfang besser gesichert und Systemantworten besser vorhersehbar sein.

Folgende "Aufnahmeoptionen" Informationen sind für ILM-Regeln verfügbar:

Doppelte Provisionierung

StorageGRID erstellt sofort Zwischenkopien des Objekts und gibt den Erfolg an den Client zurück. Kopien, die in der ILM-Regel angegeben sind, werden nach Möglichkeit erstellt.

Streng

Bevor der Erfolg an den Client zurückgegeben wird, müssen alle in der ILM-Regel angegebenen Kopien erstellt werden.

Ausgeglichen

StorageGRID versucht, bei der Aufnahme alle in der ILM-Regel angegebenen Kopien zu erstellen. Ist dies nicht möglich, werden Zwischenkopien erstellt und der Erfolg wird an den Client zurückgegeben. Die Kopien, die in der ILM-Regel angegeben sind, werden, wenn möglich gemacht.

Beispiel für die Interaktion der Konsistenz- und ILM-Regel

Angenommen, Sie haben ein Grid mit zwei Standorten mit der folgenden ILM-Regel und folgender Konsistenz:

- **ILM-Regel**: Erstellen Sie zwei Objektkopien, eine am lokalen Standort und eine an einem entfernten Standort. Strikte Aufnahme-Verhaltensweise
- **Konsistenz**: Stark-global (Objektmetadaten werden sofort an alle Standorte verteilt).

Wenn ein Client ein Objekt im Grid speichert, erstellt StorageGRID sowohl Objektkopien als auch verteilt Metadaten an beiden Standorten, bevor der Kunde zum Erfolg zurückkehrt.

Das Objekt ist zum Zeitpunkt der Aufnahme der Nachricht vollständig gegen Verlust geschützt. Wenn beispielsweise der lokale Standort kurz nach der Aufnahme verloren geht, befinden sich Kopien der Objektdaten und der Objektmetadaten am Remote-Standort weiterhin. Das Objekt kann vollständig abgerufen werden.

Wenn Sie stattdessen dieselbe ILM-Regel und die Konsistenz für starke Standorte verwenden, erhält der Client möglicherweise eine Erfolgsmeldung, nachdem die Objektdaten am Remote-Standort repliziert wurden, jedoch bevor die Objektmetadaten dort verteilt werden. In diesem Fall entspricht die Sicherung von Objektmetadaten nicht dem Schutzniveau für Objektdaten. Falls der lokale Standort kurz nach der Aufnahme verloren geht, gehen Objektmetadaten verloren. Das Objekt kann nicht abgerufen werden.

Die Beziehung zwischen Konsistenz- und ILM-Regeln kann komplex sein. Wenden Sie sich an den NetApp, wenn Sie Hilfe benötigen.

Objektversionierung

Sie können den Versionsstatus eines Buckets festlegen, wenn Sie mehrere Versionen jedes Objekts beibehalten möchten. Die Aktivierung der Versionierung für einen Bucket kann zum Schutz vor versehentlichem Löschen von Objekten beitragen und ermöglicht es Ihnen, frühere Versionen eines Objekts abzurufen und wiederherzustellen.

Das StorageGRID System implementiert Versionierung mit Unterstützung für die meisten Funktionen und weist einige Einschränkungen auf. StorageGRID unterstützt bis zu 10,000 Versionen jedes Objekts.

Die Objektversionierung kann mit StorageGRID Information Lifecycle Management (ILM) oder mit der S3 Bucket Lifecycle-Konfiguration kombiniert werden. Sie müssen die Versionierung für jeden Bucket explizit aktivieren. Wenn die Versionierung für einen Bucket aktiviert ist, wird jedem dem Bucket hinzugefügten Objekt eine Versions-ID zugewiesen, die vom StorageGRID System generiert wird.

Die Verwendung von MFA (Multi-Faktor-Authentifizierung) Löschen wird nicht unterstützt.

Die Versionierung kann nur auf Buckets aktiviert werden, die mit StorageGRID Version 10.3 oder höher erstellt wurden.

ILM und Versionierung

ILM-Richtlinien werden auf jede Version eines Objekts angewendet. Ein ILM-Scanprozess scannt kontinuierlich alle Objekte und bewertet sie anhand der aktuellen ILM-Richtlinie neu. Alle Änderungen, die Sie an ILM-Richtlinien vornehmen, werden auf alle zuvor aufgenommenen Objekte angewendet. Dies umfasst bereits aufgenommene Versionen, wenn die Versionierung aktiviert ist. Beim ILM-Scannen werden neue ILM-Änderungen an zuvor aufgenommenen Objekten angewendet.

Bei S3-Objekten in versionierungsfähigen Buckets ermöglicht die Versionsunterstützung, ILM-Regeln zu erstellen, die als Referenzzeit „nicht aktuelle Zeit“ verwenden (wählen Sie **Ja** für die Frage „Diese Regel nur auf ältere Objektversionen anwenden?“ in "Schritt 1 des Assistenten zum Erstellen einer ILM-Regel"). Wenn ein Objekt aktualisiert wird, werden seine vorherigen Versionen nicht aktuell. Mithilfe eines Filters „nicht aktuelle Zeit“ können Sie Richtlinien erstellen, die die Auswirkungen vorheriger Objektversionen auf den Storage verringern.

> Wenn Sie eine neue Version eines Objekts über einen mehrteiligen Upload-Vorgang hochladen, wird der nicht aktuelle Zeitpunkt für die Originalversion des Objekts angezeigt, wenn der mehrteilige Upload für die neue Version erstellt wurde, nicht erst nach Abschluss des mehrteiligen Uploads. In begrenzten Fällen kann die nicht aktuelle Zeit der ursprünglichen Version Stunden oder Tage früher als die Zeit für die aktuelle Version sein.

Verwandte Informationen

- "Löschen von S3-versionierten Objekten"
- "ILM-Regeln und Richtlinien für versionierte S3-Objekte (Beispiel 4)".

Konfigurieren Sie die S3-Objektsperre über die S3-REST-API

Wenn die globale S3-Objektsperrung für Ihr StorageGRID-System aktiviert ist, können Sie Buckets mit aktivierter S3-Objektsperrung erstellen. Sie können für jeden Bucket oder die Aufbewahrungseinstellungen für jede Objektversion die Standardaufbewahrung festlegen.

Aktivieren der S3-Objektsperre für einen Bucket

Wenn die globale S3-Objektsperreneinstellung für Ihr StorageGRID-System aktiviert ist, können Sie bei der Erstellung jedes Buckets optional die S3-Objektsperre aktivieren.

S3 Object Lock ist eine permanente Einstellung, die nur beim Erstellen eines Buckets aktiviert werden kann. Sie können S3-Objektsperre nicht hinzufügen oder deaktivieren, nachdem ein Bucket erstellt wurde.

Verwenden Sie eine der folgenden Methoden, um S3 Object Lock für einen Bucket zu aktivieren:

- Erstellen Sie den Bucket mit Tenant Manager. Siehe "S3-Bucket erstellen".
- Erstellen Sie den Bucket mithilfe einer CreateBucket-Anforderung mit dem `x-amz-bucket-object-lock-enabled` Anforderungsheader. Siehe "Operationen auf Buckets".

S3 Object Lock erfordert eine Bucket-Versionierung, die beim Erstellen des Buckets automatisch aktiviert wird. Die Versionierung für den Bucket kann nicht unterbrochen werden. Siehe "Objektversionierung".

Standardeinstellungen für die Aufbewahrung eines Buckets

Wenn S3 Object Lock für einen Bucket aktiviert ist, können Sie optional die Standardaufbewahrung für den Bucket aktivieren und einen Standardaufbewahrungsmodus und die Standardaufbewahrungsdauer festlegen.

Standardaufbewahrungsmodus

- Im COMPLIANCE-Modus:
  - Das Objekt kann erst gelöscht werden, wenn das Aufbewahrungsdatum erreicht ist.
  - Das Aufbewahrungsdatum des Objekts kann erhöht, aber nicht verringert werden.
  - Das Aufbewahrungsdatum des Objekts kann erst entfernt werden, wenn dieses Datum erreicht ist.
- Im GOVERNANCE-Modus:
  - Benutzer mit der `s3:BypassGovernanceRetention` Berechtigung können den Anforderungskopf verwenden `x-amz-bypass-governance-retention: true`, um die Aufbewahrungseinstellungen zu umgehen.
  - Diese Benutzer können eine Objektversion löschen, bevor das Aufbewahrungsdatum erreicht ist.
  - Diese Benutzer können das Aufbewahrungsdatum eines Objekts erhöhen, verringern oder entfernen.

Standardaufbewahrungszeitraum

Für jeden Bucket kann ein Standardaufbewahrungszeitraum in Jahren oder Tagen angegeben werden.

Festlegen der Standardaufbewahrung für einen Bucket

Verwenden Sie eine der folgenden Methoden, um die Standardaufbewahrung für einen Bucket festzulegen:

- Managen Sie die Bucket-Einstellungen über den Tenant Manager. Siehe "Erstellen eines S3-Buckets" und "Aktualisieren Sie die S3 Object Lock-Standardaufbewahrung".
- Geben Sie eine PutObjectLockConfiguration-Anforderung für den Bucket aus, um den Standardmodus und die Standardanzahl von Tagen oder Jahren festzulegen.

PutObjectLockKonfiguration

Mit der PutObjectLockConfiguration-Anforderung können Sie den Standardaufbewahrungsmodus und den Standardaufbewahrungszeitraum für einen Bucket festlegen und ändern, für den S3 Object Lock aktiviert ist. Sie können auch zuvor konfigurierte Standardeinstellungen entfernen.

Wenn neue Objektversionen in den Bucket aufgenommen werden, wird der Standardaufbewahrungsmodus angewendet, sofern `x-amz-object-lock-mode` diese `x-amz-object-lock-retain-until-date` nicht angegeben sind. Der Standardaufbewahrungszeitraum wird verwendet, um das Aufbewahrungsdatum zu berechnen, wenn `x-amz-object-lock-retain-until-date` nicht angegeben ist.

Wenn der Standardaufbewahrungszeitraum nach der Aufnahme einer Objektversion geändert wird, bleibt das „bis-Aufbewahrung“-Datum der Objektversion identisch und wird im neuen Standardaufbewahrungszeitraum nicht neu berechnet.

Sie müssen über die Berechtigung verfügen oder Konto root sein, um `s3:PutBucketObjectLockConfiguration` diesen Vorgang abzuschließen.

Der `Content-MD5` Anforderungskopf muss in der PUT-Anforderung angegeben werden.

Anforderungsbeispiel

In diesem Beispiel wird S3 Object Lock für einen Bucket aktiviert und der Standardaufbewahrungsmodus auf COMPLIANCE und der Standardaufbewahrungszeitraum auf 6 Jahre festgelegt.

```
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>
```

Bestimmen der Standardaufbewahrung für einen Bucket

Verwenden Sie eine der folgenden Methoden, um zu ermitteln, ob S3 Object Lock für einen Bucket aktiviert ist und den Standardaufbewahrungsmodus und den Standardaufbewahrungszeitraum anzuzeigen:

- Zeigen Sie den Bucket im Tenant Manager an. Siehe "S3 Buckets anzeigen".
- Stellen Sie eine GetObjectLockConfiguration-Anforderung aus.

GetObjectLockConfiguration

Mit der GetObjectLockConfiguration-Anforderung können Sie festlegen, ob S3 Object Lock für einen Bucket aktiviert ist. Wenn diese Option aktiviert ist, können Sie prüfen, ob für den Bucket ein Standardaufbewahrungsmodus und eine Aufbewahrungsfrist konfiguriert sind.

Wenn neue Objektversionen in den Bucket aufgenommen werden, wird der Standardaufbewahrungsmodus angewendet, wenn `x-amz-object-lock-mode` nicht angegeben ist. Der Standardaufbewahrungszeitraum wird verwendet, um das Aufbewahrungsdatum zu berechnen, wenn `x-amz-object-lock-retain-until-date` nicht angegeben ist.

Sie müssen über die Berechtigung verfügen oder Konto root sein, um `s3:GetBucketObjectLockConfiguration` diesen Vorgang abzuschließen.

Anforderungsbeispiel

```
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
```

Antwortbeispiel

```
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>
```

Festlegen von Aufbewahrungseinstellungen für ein Objekt

Ein Bucket mit aktivierter S3-Objektsperrung kann eine Kombination von Objekten mit und ohne Aufbewahrungseinstellungen für S3-Objektsperrung enthalten.

Aufbewahrungseinstellungen auf Objektebene werden über die S3-REST-API angegeben. Die Aufbewahrungseinstellungen für ein Objekt überschreiben alle Standardaufbewahrungseinstellungen für den Bucket.

Sie können für jedes Objekt die folgenden Einstellungen festlegen:

- **Retention Mode**: Entweder COMPLIANCE oder GOVERNANCE.
- **Bis-Datum behalten**: Ein Datum, das angibt, wie lange die Objektversion von StorageGRID beibehalten

werden muss.

  - Wenn im COMPLIANCE-Modus das Aufbewahrungsdatum in der Zukunft liegt, kann das Objekt abgerufen, aber nicht geändert oder gelöscht werden. Das Aufbewahrungsdatum kann erhöht werden, aber dieses Datum kann nicht verringert oder entfernt werden.
  - Im GOVERNANCE-Modus können Benutzer mit besonderer Berechtigung die Einstellung „bis zum Datum behalten“ umgehen. Sie können eine Objektversion löschen, bevor der Aufbewahrungszeitraum abgelaufen ist. Außerdem können sie das Aufbewahrungsdatum erhöhen, verringern oder sogar entfernen.

- **Legal Hold**: Die Anwendung eines gesetzlichen Hold auf eine Objektversion sperrt diesen Gegenstand sofort. Beispielsweise müssen Sie ein Objekt, das mit einer Untersuchung oder einem Rechtsstreit zusammenhängt, rechtlich festhalten. Eine gesetzliche Aufbewahrungspflichten haben kein Ablaufdatum, bleiben aber bis zur ausdrücklichen Entfernung erhalten.

  Die Legal Hold-Einstellung für ein Objekt ist unabhängig vom Aufbewahrungsmodus und dem Aufbewahrungsdatum. Befindet sich eine Objektversion unter einem Legal Hold, kann diese Version nicht gelöscht werden.

Um die S3-Objektsperreinstellungen beim Hinzufügen einer Objektversion zu einem Bucket anzugeben, geben Sie eine "PutObject", , "CopyObject"oder "CreateMultipartUpload"-Anforderung aus.

Sie können Folgendes verwenden:

- `x-amz-object-lock-mode`, Die COMPLIANCE oder GOVERNANCE sein können (Groß-/Kleinschreibung beachten).

  > Wenn Sie angeben `x-amz-object-lock-mode`, müssen Sie auch angeben `x-amz-object-lock-retain-until-date`.

- `x-amz-object-lock-retain-until-date`
  - Der Wert „bis zum Datum behalten“ muss im Format `2020-08-10T21:46:00Z`vorliegen. Fraktionale Sekunden sind zulässig, aber nur 3 Dezimalstellen bleiben erhalten (Präzision in Millisekunden). Andere ISO 8601-Formate sind nicht zulässig.
  - Das „Aufbewahrung bis“-Datum muss in der Zukunft liegen.
- `x-amz-object-lock-legal-hold`

  Wenn die gesetzliche Aufbewahrungspflichten LIEGEN (Groß-/Kleinschreibung muss beachtet werden), wird das Objekt unter einer gesetzlichen Aufbewahrungspflichten platziert. Wenn die gesetzliche Aufbewahrungspflichten AUS DEM WEG gehen, wird keine gesetzliche Aufbewahrungspflichten platziert. Jeder andere Wert führt zu einem 400-Fehler (InvalidArgument).

Wenn Sie eine dieser Anfrageheadern verwenden, beachten Sie die folgenden Einschränkungen:

- Der `Content-MD5` Anforderungsheader ist erforderlich, wenn `x-amz-object-lock-*` in der PutObject-Anforderung ein Anforderungsheader vorhanden ist. `Content-MD5` Ist für CopyObject oder CreateMultipartUpload nicht erforderlich.
- Wenn im Bucket die S3-Objektsperre nicht aktiviert ist und eine `x-amz-object-lock-*` Anforderungsheader vorhanden ist, wird ein Fehler 400 Bad Request (InvalidRequest) zurückgegeben.
- Die PutObject-Anfrage unterstützt die Verwendung von `x-amz-storage-class: REDUCED_REDUNDANCY`, um AWS-Verhalten abzugleichen. Wird ein Objekt jedoch mit aktivierter S3-

Objektsperre in einen Bucket aufgenommen, führt StorageGRID immer eine Dual-Commit-Aufnahme durch.

- Eine nachfolgende GET- oder HeadObject-Versionsantwort enthält die Header `x-amz-object-lock-mode`, , `x-amz-object-lock-retain-until-date` und `x-amz-object-lock-legal-hold`, sofern konfiguriert und der Absender der Anfrage die richtigen Berechtigungen hat `s3:Get*`.

Sie können den Richtlinienkonditionsschlüssel verwenden `s3:object-lock-remaining-retention-days`, um die minimalen und maximal zulässigen Aufbewahrungsfristen für Ihre Objekte einzuschränken.

Aktualisieren von Aufbewahrungseinstellungen für ein Objekt

Wenn Sie die Einstellungen für die gesetzliche Aufbewahrungs- oder Aufbewahrungseinstellung einer vorhandenen Objektversion aktualisieren müssen, können Sie die folgenden Vorgänge der Unterressource des Objekts ausführen:

- `PutObjectLegalHold`

  Wenn der neue Legal-Hold-Wert AKTIVIERT ist, wird das Objekt unter einer gesetzlichen Aufbewahrungspflichten platziert. Wenn DER Rechtsvorenthalten-Wert DEAKTIVIERT ist, wird die gesetzliche Aufbewahrungspflichten aufgehoben.

- `PutObjectRetention`
  - Der Wert des Modus kann COMPLIANCE oder GOVERNANCE sein (Groß-/Kleinschreibung muss beachtet werden).
  - Der Wert „bis zum Datum behalten“ muss im Format `2020-08-10T21:46:00Z`vorliegen. Fraktionale Sekunden sind zulässig, aber nur 3 Dezimalstellen bleiben erhalten (Präzision in Millisekunden). Andere ISO 8601-Formate sind nicht zulässig.
  - Wenn eine Objektversion über ein vorhandenes Aufbewahrungsdatum verfügt, können Sie sie nur erhöhen. Der neue Wert muss in der Zukunft liegen.

So verwenden Sie DEN GOVERNANCE-Modus

Benutzer mit der `s3:BypassGovernanceRetention` Berechtigung können die aktiven Aufbewahrungseinstellungen eines Objekts umgehen, das den GOVERNANCE-Modus verwendet. Alle LÖSCHVORGÄNGE oder PutObjectRetention müssen den Anforderungsheader enthalten `x-amz-bypass-governance-retention:true`. Diese Benutzer können die folgenden zusätzlichen Vorgänge ausführen:

- Führen Sie DeleteObject- oder DeleteObjects-Vorgänge durch, um eine Objektversion vor Ablauf des Aufbewahrungszeitraums zu löschen.

  Objekte, die sich unter einem Legal Hold befinden, können nicht gelöscht werden. Legal Hold muss DEAKTIVIERT sein.

- Führen Sie PutObjectRetention-Vorgänge durch, die den Modus einer Objektversion vor Ablauf DER Aufbewahrungsfrist von GOVERNANCE in COMPLIANCE ändern.

  Die Änderung des Modus von COMPLIANCE zu GOVERNANCE ist niemals zulässig.

- Führen Sie PutObjectRetention-Operationen aus, um die Aufbewahrungsfrist einer Objektversion zu erhöhen, zu verringern oder zu entfernen.

Verwandte Informationen

- "Objekte managen mit S3 Object Lock"
- "Verwenden Sie S3 Objektsperre, um Objekte beizubehalten"
- "Amazon Simple Storage Service User Guide: Sperren Von Objekten"

S3-Lebenszykluskonfiguration erstellen

Sie können eine S3-Lebenszyklukonfiguration erstellen, um zu steuern, wann bestimmte Objekte aus dem StorageGRID System gelöscht werden.

Das einfache Beispiel in diesem Abschnitt veranschaulicht, wie eine S3-Lebenszykluskonfiguration das Löschen bestimmter Objekte aus bestimmten S3-Buckets kontrollieren kann. Das Beispiel in diesem Abschnitt dient nur zu Illustrationszwecken. Alle Details zum Erstellen von S3-Lebenszykluskonfigurationen finden Sie unter "Amazon Simple Storage Service User Guide: Objekt-Lifecycle-Management". Beachten Sie, dass StorageGRID nur Aktionen nach Ablauf unterstützt. Es werden keine Aktionen zur Transition unterstützt.

Welche Lifecycle-Konfiguration ist

Eine Lifecycle-Konfiguration ist ein Satz von Regeln, die auf die Objekte in bestimmten S3-Buckets angewendet werden. Jede Regel gibt an, welche Objekte betroffen sind und wann diese Objekte ablaufen (an einem bestimmten Datum oder nach einigen Tagen).

StorageGRID unterstützt in einer Lebenszykluskonfiguration bis zu 1,000 Lebenszyklusregeln. Jede Regel kann die folgenden XML-Elemente enthalten:

- Ablauf: Löschen eines Objekts, wenn ein bestimmtes Datum erreicht wird oder wenn eine bestimmte Anzahl von Tagen erreicht wird, beginnend mit dem Zeitpunkt der Aufnahme des Objekts.
- NoncurrentVersionExpiration: Löschen Sie ein Objekt, wenn eine bestimmte Anzahl von Tagen erreicht wird, beginnend ab dem Zeitpunkt, an dem das Objekt nicht mehr aktuell wurde.
- Filter (Präfix, Tag)
- Status
- ID

Jedes Objekt folgt den Aufbewahrungseinstellungen eines S3 Bucket-Lebenszyklus oder einer ILM-Richtlinie. Wenn ein S3-Bucket-Lebenszyklus konfiguriert ist, überschreiben die Lifecycle-Ablaufaktionen die ILM-Richtlinie für Objekte, die mit dem Bucket-Lifecycle-Filter übereinstimmen. Objekte, die nicht mit dem Bucket-Lebenszyklusfilter übereinstimmen, verwenden die Aufbewahrungseinstellungen der ILM-Richtlinie. Wenn ein Objekt mit einem Bucket-Lebenszyklusfilter übereinstimmt und keine Ablaufaktionen explizit angegeben werden, werden die Aufbewahrungseinstellungen der ILM-Richtlinie nicht verwendet, und es wird impliziert, dass Objektversionen für immer aufbewahrt werden. Siehe "Beispielprioritäten für den S3-Bucket-Lebenszyklus und die ILM-Richtlinie".

Aus diesem Grund kann ein Objekt aus dem Grid entfernt werden, obwohl die Speicheranweisungen in einer ILM-Regel noch auf das Objekt gelten. Alternativ kann ein Objekt auch dann im Grid aufbewahrt werden, wenn eine ILM-Platzierungsanleitung für das Objekt abgelaufen ist. Weitere Informationen finden Sie unter "Funktionsweise von ILM während der gesamten Nutzungsdauer eines Objekts".

> Die Bucket-Lifecycle-Konfiguration kann für Buckets verwendet werden, für die S3-Objektsperre aktiviert ist. Die Bucket-Lifecycle-Konfiguration wird jedoch für ältere Buckets, die Compliance verwenden, nicht unterstützt.

StorageGRID unterstützt den Einsatz der folgenden Bucket-Operationen zum Management der

Lebenszykluskonfigurationen:

- DeleteBucketLifecycle
- GetBucketLifecycleKonfiguration
- PutBucketLifecycleKonfiguration

Lebenszyklukonfiguration erstellen

Als erster Schritt beim Erstellen einer Lebenszykluskonfiguration erstellen Sie eine JSON-Datei mit einem oder mehreren Regeln. Diese JSON-Datei enthält beispielsweise drei Regeln:

1. Regel 1 gilt nur für Objekte, die dem Präfix/ entsprechen `category1` und den Wert `tag2` haben `key2`. Der `Expiration` Parameter gibt an, dass Objekte, die dem Filter entsprechen, am 22. August 2020 um Mitternacht ablaufen.
2. Regel 2 gilt nur für Objekte, die dem Präfix/ entsprechen `category2`. Der `Expiration` Parameter gibt an, dass Objekte, die dem Filter entsprechen, 100 Tage nach ihrer Aufnahme ablaufen.

   > Regeln, die eine Anzahl von Tagen angeben, sind relativ zu dem Zeitpunkt, an dem das Objekt aufgenommen wurde. Wenn das aktuelle Datum das Aufnahmedatum plus die Anzahl der Tage überschreitet, werden einige Objekte möglicherweise aus dem Bucket entfernt, sobald die Lebenszykluskonfiguration angewendet wird.

3. Regel 3 gilt nur für Objekte, die dem Präfix/ entsprechen `category3`. Der `Expiration` Parameter gibt an, dass alle nicht aktuellen Versionen übereinstimmender Objekte 50 Tage nach ihrer Nichtaktueller ablaufen.

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

Lifecycle-Konfiguration auf Bucket anwenden

Nachdem Sie die Lebenszykluskonfigurationsdatei erstellt haben, wenden Sie sie auf einen Bucket an, indem Sie eine Anforderung von PutkBucketLifecycleConfiguration ausgeben.

Diese Anforderung wendet die Lebenszykluskonfiguration in der Beispieldatei auf Objekte in einem Bucket mit dem Namen `testbucket`an.

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

Um zu überprüfen, ob eine Lebenszykluskonfiguration erfolgreich auf den Bucket angewendet wurde, geben Sie eine GetBucketLifecycleConfiguration-Anforderung aus. Beispiel:

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

Eine erfolgreiche Antwort zeigt die Konfiguration des Lebenszyklus, die Sie gerade angewendet haben.

Überprüfen, ob der Bucket-Lebenszyklus für das Objekt gilt

Sie können festlegen, ob eine Ablaufregel in der Lebenszykluskonfiguration für ein bestimmtes Objekt gilt, wenn Sie eine PutObject-, HeadObject- oder GetObject-Anforderung ausgeben. Wenn eine Regel angewendet wird, enthält die Antwort einen `Expiration` Parameter, der angibt, wann das Objekt abläuft und welche Ablaufregel abgeglichen wurde.

> Da der Bucket-Lebenszyklus ILM außer Kraft setzt, wird als tatsächliches Datum angezeigt, an dem `expiry-date` das Objekt gelöscht wird. Weitere Informationen finden Sie unter "Wie die Aufbewahrung von Objekten bestimmt wird".

Zum Beispiel wurde diese PutObject-Anforderung am 22. Juni 2020 ausgegeben und legt ein Objekt in den `testbucket` Bucket.

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

Die Erfolgsreaktion zeigt an, dass das Objekt in 100 Tagen (01. Oktober 2020) abläuft und dass es mit Regel 2 der Lebenszykluskonfiguration übereinstimmt.

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

Diese HeadObject-Anforderung wurde beispielsweise verwendet, um Metadaten für dasselbe Objekt im testbucket-Bucket zu erhalten.

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

Die Erfolgsreaktion umfasst die Metadaten des Objekts und gibt an, dass das Objekt in 100 Tagen abläuft und dass es mit Regel 2 übereinstimmt.

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

Bei Buckets mit aktivierter Versionierung gilt der `x-amz-expiration` Antwortheader nur für aktuelle Versionen von Objekten.

Empfehlungen für die Implementierung der S3-REST-API

Bei der Implementierung der S3-REST-API zur Verwendung mit StorageGRID sollten Sie diese Empfehlungen beachten.

Empfehlungen für Köpfe zu nicht vorhandenen Objekten

Wenn Ihre Anwendung regelmäßig prüft, ob ein Objekt an einem Pfad existiert, wo Sie nicht erwarten, dass das Objekt tatsächlich existiert, sollten Sie das "verfügbar" verwenden"Konsistenz". Sie sollten beispielsweise die Konsistenz „verfügbar“ verwenden, wenn Ihre Anwendung einen Speicherort vorgibt, bevor Sie ihn verwenden.

Wenn der HAUPTVORGANG das Objekt nicht findet, erhalten Sie möglicherweise eine hohe Anzahl von 500 internen Serverfehlern, wenn zwei oder mehr Storage Nodes am selben Standort nicht verfügbar sind oder ein Remote-Standort nicht erreichbar ist.

Sie können die „verfügbare“ Konsistenz für jeden Bucket mithilfe der Anforderung festlegen "PUT Bucket-Konsistenz"oder die Konsistenz in der Anforderungsheader für eine einzelne API-Operation angeben.

Empfehlungen für Objektschlüssel

Befolgen Sie diese Empfehlungen für Objektschlüsselnamen auf Basis des ersten Erstells des Buckets.

Buckets, die in StorageGRID 11.4 oder früher erstellt wurden

- Verwenden Sie keine Zufallswerte als die ersten vier Zeichen von Objektschlüsseln. Dies steht im Gegensatz zu der früheren AWS Empfehlung für wichtige Präfixe. Verwenden Sie stattdessen nicht zufällige, nicht eindeutige Präfixe, wiez. B. `image`.
- Wenn Sie der früheren AWS-Empfehlung folgen, zufällige und eindeutige Zeichen in Schlüsselpräfixen zu verwenden, setzen Sie den Objektschlüsseln einen Verzeichnisnamen vor. Verwenden Sie dieses Format:

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

  Anstelle dieses Formats:

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

Buckets, die in StorageGRID 11.4 oder höher erstellt wurden

Es ist nicht erforderlich, Objektschlüsselnamen auf die Best Practices für die Performance zu beschränken. In den meisten Fällen können Sie zufällige Werte für die ersten vier Zeichen von Objektschlüsselnamen verwenden.

> Eine Ausnahme ist ein S3-Workload, der nach kurzer Zeit kontinuierlich alle Objekte entfernt. Um die Auswirkungen auf die Performance in diesem Anwendungsfall zu minimieren, variieren Sie alle tausend Objekte mit einem ähnlichen Datum einen führenden Teil des Schlüsselnamens. Angenommen, ein S3-Client schreibt in der Regel 2,000 Objekte/Sekunde, und die ILM- oder Bucket-Lifecycle-Richtlinie entfernt alle Objekte nach drei Tagen. Um die Auswirkungen auf die Performance zu minimieren, können Sie Schlüssel anhand eines Musters wie folgt benennen: `/mybucket/mydir/yyyymmddhhmmss-random_UUID.jpg`

Empfehlungen für „Range Reads“

Wenn der "Globale Option zum Komprimieren gespeicherter Objekte"aktiviert ist, sollten S3-Client-Anwendungen die Ausführung von GetObject-Operationen vermeiden, die einen Bereich von Bytes angeben, die zurückgegeben werden sollen. Diese Vorgänge beim Lesen von Range sind ineffizient, da StorageGRID Objekte effektiv dekomprimieren muss, um auf die angeforderten Bytes zuzugreifen. GetObject Operationen, die einen kleinen Bereich von Bytes von einem sehr großen Objekt anfordern, sind besonders ineffizient; zum Beispiel ist es ineffizient, einen 10 MB Bereich von einem 50 GB komprimierten Objekt zu lesen.

Wenn Bereiche von komprimierten Objekten gelesen werden, können Client-Anforderungen eine Zeitdauer haben.

> Wenn Sie Objekte komprimieren müssen und Ihre Client-Applikation Bereichslesevorgänge verwenden muss, erhöhen Sie die Zeitüberschreitung beim Lesen der Anwendung.

Unterstützung für Amazon S3-REST-API

Details zur S3-REST-API-Implementierung

Das StorageGRID System implementiert die Simple Storage Service API (API Version 2006-03-01) mit Unterstützung der meisten Operationen und mit einigen Einschränkungen. Wenn Sie S3 REST-API-Client-Applikationen integrieren, sind die Implementierungsdetails bekannt.

Das StorageGRID System unterstützt sowohl Virtual-Hosted-Style-Anforderungen als auch Anforderungen im Pfadstil.

Umgang mit Daten

Die StorageGRID Implementierung der S3-REST-API unterstützt nur gültige HTTP-Datumsformate.

Das StorageGRID-System unterstützt nur gültige HTTP-Datumsformate für alle Header, die Datumswerte akzeptieren. Der Zeitbereich des Datums kann im Greenwich Mean Time (GMT)-Format oder im UTC-Format (Universal Coordinated Time) ohne Zeitzonenversatz angegeben werden (+0000 muss angegeben werden). Wenn Sie die Kopfzeile in Ihre Anfrage aufnehmen `x-amz-date`, wird ein Wert überschrieben, der in der Kopfzeile der Datumsanforderung angegeben ist. Bei Verwendung von AWS Signature Version 4 muss der `x-amz-date` Header in der signierten Anfrage vorhanden sein, da der Datumskopf nicht unterstützt wird.

Allgemeine Anfragemöpfe

Das StorageGRID-System unterstützt die von definierten allgemeinen Anforderungsheader "Amazon Simple Storage Service API-Referenz: Common Request Header" mit einer Ausnahme.

Kopfzeile der Anfrage	Implementierung
Autorisierung	Vollständige Unterstützung für AWS Signature Version 2 Unterstützung für AWS Signature Version 4, mit folgenden Ausnahmen: • Wenn Sie den tatsächlichen Wert der Payload Checksumme in angeben `x-amz-content-sha256`, wird der Wert ohne Validierung akzeptiert, als ob der Wert `UNSIGNED-PAYLOAD` für den Header angegeben worden wäre. Wenn Sie einen Header-Wert angeben `x-amz-content-sha256`, der Streaming impliziert `aws-chunked` (z. B. STREAMING-AWS4-HMAC-SHA256-PAYLOAD), werden die Chunk-Signaturen nicht gegen die Chunk-Daten verifiziert.
X-amz-Sicherheits-Token	Nicht implementiert. Kehrt Zurück. `XNotImplemented`

Allgemeine Antwortkopfzeilen

Das StorageGRID System unterstützt alle gängigen Antwortheader, die durch die *Simple Storage Service API Reference* definiert wurden. Eine Ausnahme bilden die Antwort.

Kopfzeile der Antwort	Implementierung
X-amz-id-2	Nicht verwendet

Authentifizieren von Anfragen

Das StorageGRID-System unterstützt über die S3-API sowohl authentifizierten als auch anonymen Zugriff auf Objekte.

Die S3-API unterstützt Signature Version 2 und Signature Version 4 zur Authentifizierung von S3-API-Anforderungen.

Authentifizierte Anfragen müssen mit Ihrer Zugriffsschlüssel-ID und Ihrem geheimen Zugriffsschlüssel signiert werden.

Das StorageGRID-System unterstützt zwei Authentifizierungsmethoden: Den HTTP- `Authorization`Header und die Abfrageparameter.

Verwenden Sie den HTTP-Autorisierungskopf

Der HTTP- `Authorization`Header wird von allen S3-API-Operationen außer „Anonyme Anfragen“ verwendet, sofern dies durch die Bucket-Richtlinie zulässig ist. Die `Authorization` Kopfzeile enthält alle erforderlichen Signaturinformationen zur Authentifizierung einer Anforderung.

Abfrageparameter verwenden

Sie können Abfrageparameter verwenden, um Authentifizierungsinformationen zu einer URL hinzuzufügen. Dies wird als Vorsignierung der URL bezeichnet, mit der ein temporärer Zugriff auf bestimmte Ressourcen gewährt werden kann. Benutzer mit der vorgeschichteten URL müssen den geheimen Zugriffsschlüssel nicht kennen, um auf die Ressource zuzugreifen. So können Sie beschränkten Zugriff von Drittanbietern auf eine Ressource bereitstellen.

Betrieb auf dem Service

Das StorageGRID System unterstützt die folgenden Vorgänge beim Service.

Betrieb	Implementierung
ListBuchs (Zuvor „GET Service“ genannt)	Wird mit dem gesamten Amazon S3-REST-API-Verhalten implementiert. Änderungen vorbehalten.
GET Storage-Auslastung	In der StorageGRID"GET Storage-Auslastung"-Anfrage wird der von einem Konto insgesamt und für jeden mit dem Konto verknüpften Bucket verwendete Storage angezeigt. Dies ist eine Operation auf dem Dienst mit einem Pfad von / und einem benutzerdefinierten Abfrageparameter (`?x-ntap-sg-usage`) hinzugefügt.
OPTIONEN /	Client-Anwendungen können `OPTIONS /` Anfragen an den S3-Port auf einem Storage-Node ausgeben, ohne S3-Authentifizierungsdaten bereitzustellen, um festzustellen, ob der Storage-Node verfügbar ist. Sie können diese Anforderung zum Monitoring verwenden oder um zu ermöglichen, dass externe Load Balancer eingesetzt werden, wenn ein Storage-Node ausfällt.

Operationen auf Buckets

Das StorageGRID System unterstützt für jedes S3-Mandantenkonto maximal 5,000 Buckets.

Jedes Grid kann maximal 100,000 Buckets enthalten.

Um 5,000 Buckets zu unterstützen, muss jeder Storage Node im Grid mindestens 64 GB RAM aufweisen.

Einschränkungen für Bucket-Namen folgen den regionalen Einschränkungen des AWS US Standard. Sie sollten sie jedoch weiter auf DNS-Namenskonventionen beschränken, um Anforderungen im virtuellen Hosted-Stil von S3 zu unterstützen.

Weitere Informationen finden Sie im Folgenden:

- "Amazon Simple Storage Service User Guide: Bucket-Kontingente, Einschränkungen und Einschränkungen"
- "Konfigurieren Sie die Domänennamen des S3-Endpunkts"

Die Operationen ListObjects (GET Bucket) und ListObjectVersions (GET Bucket Object Versions) unterstützen StorageGRID "Konsistenzwerte".

Sie können überprüfen, ob für einzelne Buckets Updates zur letzten Zugriffszeit aktiviert oder deaktiviert wurden. Siehe "ZEITPUNKT des letzten Zugriffs FÜR den Bucket ABRUFEN".

In der folgenden Tabelle wird beschrieben, wie StorageGRID S3-REST-API-Bucket-Operationen implementiert Um einen dieser Vorgänge durchzuführen, müssen die erforderlichen Anmeldedaten für den Zugriff für das Konto bereitgestellt werden.

Betrieb	Implementierung
CreateBucket	Erstellt einen neuen Bucket. Mit dem Erstellen des Buckets werden Sie zum Bucket-Eigentümer. • Bucket-Namen müssen die folgenden Regeln einhalten: ◦ Jedes StorageGRID System muss eindeutig sein (nicht nur innerhalb des Mandantenkontos). ◦ Muss DNS-konform sein. ◦ Muss mindestens 3 und nicht mehr als 63 Zeichen enthalten. ◦ Kann eine Reihe von einer oder mehreren Etiketten sein, wobei angrenzende Etiketten durch einen Zeitraum getrennt sind. Jedes Etikett muss mit einem Kleinbuchstaben oder einer Zahl beginnen und enden. Es können nur Kleinbuchstaben, Ziffern und Bindestriche verwendet werden. ◦ Darf nicht wie eine Text-formatierte IP-Adresse aussehen. ◦ Perioden sollten nicht in Anforderungen im virtuellen gehosteten Stil verwendet werden. Perioden verursachen Probleme bei der Überprüfung des Server-Platzhalterzertifikats. • Standardmäßig werden Buckets in der Region erstellt `us-east-1`. Sie können jedoch das Anforderungselement im Anforderungskörper verwenden `LocationConstraint`, um einen anderen Bereich anzugeben. Wenn Sie das Element verwenden `LocationConstraint`, müssen Sie den genauen Namen einer Region angeben, die mit dem Grid Manager oder der Grid Management API definiert wurde. Wenden Sie sich an Ihren Systemadministrator, wenn Sie den zu verwendenden Regionalnamen nicht kennen. **Hinweis**: Ein Fehler tritt auf, wenn Ihre CreateBucket-Anforderung eine Region verwendet, die nicht in StorageGRID definiert wurde. • Sie können den Anforderungsheader einschließen `x-amz-bucket-object-lock-enabled`, um einen Bucket mit aktivierter S3 Object Lock zu erstellen. Siehe "Konfigurieren Sie die S3-Objektsperre über die S3-REST-API". Sie müssen die S3-Objektsperre aktivieren, wenn Sie den Bucket erstellen. Sie können S3-Objektsperre nicht hinzufügen oder deaktivieren, nachdem ein Bucket erstellt wurde. Für die S3-Objektsperre ist eine Bucket-Versionierung erforderlich. Diese wird bei der Erstellung des Buckets automatisch aktiviert.
DeleteBucket	Löscht den Bucket.
DeleteBucketCors	Löscht die CORS-Konfiguration für den Bucket.
DeleteBucketEncryption	Löscht die Standardverschlüsselung aus dem Bucket. Vorhandene verschlüsselte Objekte bleiben verschlüsselt, neue Objekte, die dem Bucket hinzugefügt wurden, werden jedoch nicht verschlüsselt.

Betrieb	Implementierung
DeleteBucketLifecycle	Löscht die Lebenszykluskonfiguration aus dem Bucket. Siehe "S3-Lebenszykluskonfiguration erstellen".
DeleteBucketRichtlinien	Löscht die dem Bucket angehängte Richtlinie.
DeleteBucketReplication	Löscht die Replikationskonfiguration, die mit dem Bucket verbunden ist.
DeleteBucketTagging	Verwendet die `tagging` Unterressource, um alle Tags aus einem Bucket zu entfernen. **Achtung**: Wenn für diesen Bucket ein nicht standardmäßiges ILM-Policy-Tag gesetzt ist, wird ein Bucket-Tag mit einem ihm zugewiesenen Wert vorhanden sein `NTAP-SG-ILM-BUCKET-TAG`. Stellen Sie keine DeleteBucketTagging-Anforderung aus, wenn ein `NTAP-SG-ILM-BUCKET-TAG` Bucket-Tag vorhanden ist. Geben Sie stattdessen eine Anforderung für das PutkBucketTagging nur mit dem `NTAP-SG-ILM-BUCKET-TAG` Tag und dem ihm zugewiesenen Wert aus, um alle anderen Tags aus dem Bucket zu entfernen. Ändern oder entfernen Sie das Bucket-Tag nicht `NTAP-SG-ILM-BUCKET-TAG`.
GetBucketAcl	Gibt eine positive Antwort und die ID, den Anzeigenamen und die Berechtigung des Bucket-Eigentümers zurück, was darauf hinweist, dass der Besitzer vollen Zugriff auf den Bucket hat.
GetBucketCors	Gibt die Konfiguration für den Bucket zurück `cors`.
GetBucketEncryption	Gibt die Standardverschlüsselungskonfiguration für den Bucket zurück.
GetBucketLifecycleKonfiguration (Zuvor GET Bucket-Lebenszyklus genannt)	Gibt die Lebenszykluskonfiguration für den Bucket zurück. Siehe "S3-Lebenszykluskonfiguration erstellen".
GetBucketLocation	Gibt die Region zurück, die mit dem Element in der Anforderung CreateBucket festgelegt wurde `LocationConstraint`. Wenn der Bereich des Buckets ist `us-east-1`, wird eine leere Zeichenfolge für die Region zurückgegeben.
GetBucketNotificationConfiguration (Zuvor namens „GET Bucket“-Benachrichtigung)	Gibt die Benachrichtigungskonfiguration zurück, die mit dem Bucket verbunden ist.
GetBucketPolicy	Gibt die dem Bucket angehängte Richtlinie zurück.
GetBucketReplication	Gibt die Replikationskonfiguration zurück, die mit dem Bucket verbunden ist.

Betrieb	Implementierung
GetBucketTagging	Verwendet die `tagging` Unterressource, um alle Tags für einen Bucket zurückzugeben. **Achtung**: Wenn für diesen Bucket ein nicht standardmäßiges ILM-Policy-Tag gesetzt ist, wird ein Bucket-Tag mit einem ihm zugewiesenen Wert vorhanden sein `NTAP-SG-ILM-BUCKET-TAG`. Ändern oder entfernen Sie dieses Tag nicht.
GetBucketVersioning	Diese Implementierung verwendet die `versioning` Subressource, um den Versionsstatus eines Buckets zurückzugeben. • *Blank*: Die Versionierung wurde nie aktiviert (Bucket ist „unversioniert“) • Aktiviert: Versionierung ist aktiviert • Suspendiert: Die Versionierung war zuvor aktiviert und wird ausgesetzt
GetObjectLockConfiguration	Gibt den Standardaufbewahrungsmodus für Bucket und den Standardaufbewahrungszeitraum zurück, sofern konfiguriert. Siehe "Konfigurieren Sie die S3-Objektsperre über die S3-REST-API".
HeadBucket	Legt fest, ob ein Bucket vorhanden ist und Sie über die Berechtigung verfügen, darauf zuzugreifen. Dieser Vorgang liefert Folgendes zurück: • `x-ntap-sg-bucket-id`: Die UUID des Buckets im UUID-Format. • `x-ntap-sg-trace-id`: Die eindeutige Trace-ID der zugehörigen Anforderung.
ListObjects und ListObjectsV2 (Zuvor benannt nach „GET Bucket“)	Gibt einige oder alle (bis zu 1,000) Objekte in einem Bucket zurück. Die Storage-Klasse für Objekte kann einen der beiden Werte haben, selbst wenn das Objekt mit der Option Storage-Klasse aufgenommen wurde `REDUCED_REDUNDANCY`: • `STANDARD`, Das angibt, dass das Objekt in einem Speicherpool mit Storage Nodes gespeichert ist. • `GLACIER`, Das angibt, dass das Objekt in den externen Bucket verschoben wurde, der vom Cloud-Speicherpool angegeben wurde. Wenn der Bucket eine große Anzahl von gelöschten Schlüsseln mit dem gleichen Präfix enthält, kann die Antwort einige `CommonPrefixes` enthalten, die keine Schlüssel enthalten.
ListObjectVersions (Zuvor namens „GET Bucket Object Versions“)	Mit LESEZUGRIFF auf einen Bucket wird dieser Vorgang mit den Unterressourcen-Listen Metadaten aller Versionen von Objekten im Bucket verwendet `versions`.

Betrieb	Implementierung
PutBucketCors	Legt die CORS-Konfiguration für einen Bucket so fest, dass der Bucket Anfragen mit verschiedenen Ursprung bedienen kann. CORS (Cross-Origin Resource Sharing) ist ein Sicherheitsmechanismus, mit dem Client-Webanwendungen in einer Domäne auf Ressourcen in einer anderen Domäne zugreifen können. Angenommen, Sie verwenden einen S3-Bucket mit dem Namen `images` zum Speichern von Grafiken. Durch die Einstellung der CORS-Konfiguration für den `images` Bucket können Sie die Bilder in diesem Bucket auf der Website anzeigen lassen `http://www.example.com`.
PutBucketEncryption	Legt den Standardverschlüsselungsstatus eines vorhandenen Buckets fest. Bei aktivierter Verschlüsselung auf Bucket-Ebene sind alle neuen dem Bucket hinzugefügten Objekte verschlüsselt.StorageGRID unterstützt serverseitige Verschlüsselung mit von StorageGRID gemanagten Schlüsseln. Wenn Sie die serverseitige Verschlüsselungskonfigurationsregel angeben, setzen Sie den `SSEAlgorithm` Parameter auf `AES256`, und verwenden Sie den Parameter nicht `KMSMasterKeyID`. Die Standardverschlüsselungskonfiguration von Buckets wird ignoriert, wenn in der Objekt-Upload-Anforderung bereits Verschlüsselung angegeben ist (d. h. wenn die Anforderung den Anforderungsheader enthält `x-amz-server-side-encryption-*`).
PutBucketLifecycleKonfiguration (Zuvor PUT Bucket-Lebenszyklus genannt)	Erstellt eine neue Lebenszykluskonfiguration für den Bucket oder ersetzt eine vorhandene Lebenszykluskonfiguration. StorageGRID unterstützt in einer Lebenszykluskonfiguration bis zu 1,000 Lebenszyklusregeln. Jede Regel kann die folgenden XML-Elemente enthalten: • Ablauf (Tage, Datum, ErstrecktObjectDeleteMarker) • Nicht-aktuellVersionAblauf (NewerNichtaktuellVersionen, nicht aktuelleTage) • Filter (Präfix, Tag) • Status • ID StorageGRID bietet folgende Maßnahmen nicht: • AbortInsetteMultipartUpload • Übergang Siehe "S3-Lebenszykluskonfiguration erstellen". Informationen über die Interaktion der Aktion „Ablauf“ in einem Bucket-Lebenszyklus mit den Anweisungen zur ILM-Platzierung finden Sie unter "Wie ILM im gesamten Leben eines Objekts funktioniert". **Hinweis**: Die Konfiguration des Bucket-Lebenszyklus kann für Buckets verwendet werden, für die S3-Objektsperre aktiviert ist. Die Bucket-Lebenszykluskonfiguration wird jedoch für ältere kompatible Buckets nicht unterstützt.

Betrieb	Implementierung

PutBucketNotificationKonfiguration
 	
(Zuvor namens „PUT Bucket“-Benachrichtigung)
	Konfiguriert Benachrichtigungen für den Bucket mithilfe der XML-Benachrichtigungskonfiguration, die im Anforderungskörper enthalten ist. Sie sollten folgende Implementierungsdetails kennen:
	
	• StorageGRID unterstützt als Ziele Amazon Simple Notification Service (Amazon SNS) oder Kafka-Themen. SQS (Simple Queue Service)- oder Amazon Lambda-Endpunkte werden nicht unterstützt.
	• Das Ziel für Benachrichtigungen muss als URN eines StorageGRID-Endpunkts angegeben werden. Endpunkte können mit dem Mandanten-Manager oder der Mandanten-Management-API erstellt werden.
	
	Der Endpunkt muss vorhanden sein, damit die Benachrichtigungskonfiguration erfolgreich ausgeführt werden kann. Wenn der Endpunkt nicht vorhanden ist, wird ein `400 Bad Request` Fehler mit dem Code zurückgegeben `InvalidArgument`.
	
	• Sie können keine Benachrichtigung für die folgenden Ereignistypen konfigurieren. Diese Ereignistypen werden **nicht** unterstützt.
	◦ `s3:ReducedRedundancyLostObject`
	◦ `s3:ObjectRestore:Completed`
	• Aus StorageGRID gesendete Ereignisbenachrichtigungen verwenden das JSON-Standardformat, außer dass sie einige Schlüssel nicht enthalten und bestimmte Werte für andere verwenden, wie in der folgenden Liste gezeigt:
	◦ **EventSource**
	
	`sgws:s3`
	
	◦ **AwsRegion**
	
	Nicht enthalten
	
	◦ * X-amz-id-2*
	
	Nicht enthalten
	
	◦ **arn**
	
	`urn:sgws:s3:::bucket_name`

PutBucketPolicy	Legt die dem Bucket angehängte Richtlinie fest. Siehe "Verwendung von Bucket- und Gruppenzugriffsrichtlinien".

Betrieb	Implementierung
PutBucketReplication	Konfiguration "StorageGRID CloudMirror Replizierung" für den Bucket mithilfe der im Anforderungskörper bereitgestellten XML-Replikationskonfiguration Für die CloudMirror-Replikation sollten Sie die folgenden Implementierungsdetails beachten: • StorageGRID unterstützt nur V1 der Replizierungskonfiguration. Das bedeutet, dass StorageGRID die Verwendung des Elements für Regeln nicht unterstützt `Filter` und V1-Konventionen für das Löschen von Objektversionen befolgt. Weitere Informationen finden Sie unter "Amazon Simple Storage Service User Guide: Replizierungskonfiguration". • Die Bucket-Replizierung kann für versionierte oder nicht versionierte Buckets konfiguriert werden. • Sie können in jeder Regel der XML-Replikationskonfiguration einen anderen Ziel-Bucket angeben. Ein Quell-Bucket kann auf mehrere Ziel-Bucket replizieren. • Ziel-Buckets müssen als URN der StorageGRID-Endpunkte angegeben werden, wie im Mandantenmanager oder der Mandantenmanagement-API angegeben. Siehe "CloudMirror-Replizierung konfigurieren". Der Endpunkt muss vorhanden sein, damit die Replizierungskonfiguration erfolgreich ausgeführt werden kann. Wenn der Endpunkt nicht existiert, schlägt die Anforderung als fehl. Die Fehlermeldung lautet `400 Bad Request: Unable to save the replication policy. The specified endpoint URN does not exist: URN.` • Sie müssen kein in der Konfigurations-XML angeben `Role`. Dieser Wert wird von StorageGRID nicht verwendet und wird bei der Einreichung ignoriert. • Wenn Sie die Storage-Klasse aus dem Konfigurations-XML nicht angeben, verwendet StorageGRID standardmäßig die `STANDARD` Storage-Klasse. • Wenn Sie ein Objekt aus dem Quell-Bucket löschen oder den Quell-Bucket selbst löschen, sieht das Verhalten der regionsübergreifenden Replizierung wie folgt aus: ◦ Wenn Sie das Objekt oder den Bucket löschen, bevor es repliziert wurde, wird das Objekt/Bucket nicht repliziert, und Sie werden nicht benachrichtigt. ◦ Wenn Sie das Objekt oder Bucket nach der Replizierung löschen, befolgt StorageGRID das standardmäßige Löschverhalten von Amazon S3 für die V1 der regionsübergreifenden Replizierung.

Betrieb	Implementierung
PutBucketTagging	Verwendet die `tagging` Unterressource, um einen Satz von Tags für einen Bucket hinzuzufügen oder zu aktualisieren. Beachten Sie beim Hinzufügen von Bucket-Tags die folgenden Einschränkungen: • StorageGRID und Amazon S3 unterstützen für jeden Bucket bis zu 50 Tags. • Tags, die einem Bucket zugeordnet sind, müssen eindeutige Tag-Schlüssel haben. Ein Tag-Schlüssel kann bis zu 128 Unicode-Zeichen lang sein. • Die Tag-Werte können bis zu 256 Unicode-Zeichen lang sein. • Bei den Schlüsseln und Werten wird die Groß-/Kleinschreibung beachtet. **Achtung**: Wenn für diesen Bucket ein nicht standardmäßiges ILM-Policy-Tag gesetzt ist, wird ein Bucket-Tag mit einem ihm zugewiesenen Wert vorhanden sein `NTAP-SG-ILM-BUCKET-TAG`. Stellen Sie sicher, dass das `NTAP-SG-ILM-BUCKET-TAG` Bucket-Tag in allen PutBucketTagging-Anforderungen mit dem zugewiesenen Wert enthalten ist. Ändern oder entfernen Sie dieses Tag nicht. **Hinweis**: Dieser Vorgang überschreibt alle aktuellen Tags, die der Bucket bereits hat. Wenn vorhandene Tags aus dem Satz weggelassen werden, werden diese Tags für den Bucket entfernt.
PutBucketVersioning	Verwendet die `versioning` Unterressource, um den Versionsstatus eines vorhandenen Buckets festzulegen. Sie können den Versionierungsstatus mit einem der folgenden Werte festlegen: • Aktiviert: Versionierung für die Objekte im Bucket Alle dem Bucket hinzugefügten Objekte erhalten eine eindeutige Version-ID. • Suspendiert: Deaktiviert die Versionierung für die Objekte im Bucket. Alle dem Bucket hinzugefügten Objekte erhalten die Versions-ID `null`.
PutObjectLockKonfiguration	Konfiguriert oder entfernt den Standardaufbewahrungsmodus und den Standardaufbewahrungszeitraum für Bucket. Wenn der Standardaufbewahrungszeitraum geändert wird, bleiben die bisherigen Objektversionen unverändert und werden im neuen Standardaufbewahrungszeitraum nicht neu berechnet. Weitere Informationen finden Sie unter"Konfigurieren Sie die S3-Objektsperre über die S3-REST-API".

Operationen für Objekte

Operationen für Objekte

In diesem Abschnitt wird beschrieben, wie das StorageGRID System S3-REST-API-Vorgänge für Objekte implementiert.

Die folgenden Bedingungen gelten für alle Objektvorgänge:

- StorageGRID "Konsistenzwerte"werden von allen Operationen an Objekten unterstützt, mit Ausnahme der folgenden:
  - GetObjectAcl
  - `OPTIONS /`
  - PutObjectLegalHold
  - PutObjectRetention
  - SelektierObjectContent
- Widersprüchliche Clientanforderungen, wie z. B. zwei Clients, die in denselben Schlüssel schreiben, werden auf der Grundlage der „neuesten Wins“ gelöst. Der Zeitpunkt für die Bewertung „neuester Erfolge“ basiert auf dem Zeitpunkt, an dem das StorageGRID System eine bestimmte Anforderung abgeschlossen hat und nicht auf dem Zeitpunkt, an dem S3-Clients einen Vorgang starten.
- Alle Objekte in einem StorageGRID-Bucket sind im Eigentum des Bucket-Inhabers. Dies umfasst Objekte, die von einem anonymen Benutzer oder einem anderen Konto erstellt wurden.
- Der Zugriff auf Datenobjekte, die über Swift in das StorageGRID System aufgenommen wurden, ist nicht über S3 möglich.

In der folgenden Tabelle wird beschrieben, wie StorageGRID S3-REST-API-Objektvorgänge implementiert.

Betrieb	Implementierung
DeleteObject	Multi-Faktor-Authentifizierung (MFA) und der Antwortheader `x-amz-mfa` werden nicht unterstützt. Bei der Verarbeitung einer DeleteObject-Anforderung versucht StorageGRID sofort, alle Kopien des Objekts von allen gespeicherten Speicherorten zu entfernen. Wenn erfolgreich, gibt StorageGRID sofort eine Antwort an den Client zurück. Wenn nicht innerhalb von 30 Sekunden alle Kopien entfernt werden können (z. B. weil ein Speicherort vorübergehend nicht verfügbar ist), stellt StorageGRID die Kopien in eine Warteschlange zur Entfernung und zeigt dann den Erfolg des Clients an. **Versionierung** Zum Entfernen einer bestimmten Version muss der Anforderer der Bucket-Eigentümer sein und die Unterressource verwenden `versionId`. Durch die Verwendung dieser Unterressource wird die Version dauerhaft gelöscht. Wenn das `versionId` einer Löschmarkierung entspricht, wird die Antwortkopfzeile `x-amz-delete-marker` auf gesetzt zurückgegeben `true`. • Wenn ein Objekt ohne die Unterressource in einem Bucket gelöscht wird `versionId`, bei dem die Versionierung aktiviert ist, wird eine Löschmarkierung generiert. Der `versionId` für die Löschmarkierung wird mit dem Antwortheader zurückgegeben `x-amz-version-id`, und der `x-amz-delete-marker` Antwortheader wird auf gesetzt zurückgegeben `true`. • Wenn ein Objekt ohne die Unterressource in einem Bucket gelöscht wird `versionId`, bei dem die Versionierung ausgesetzt ist, führt dies zu einer dauerhaften Löschung einer bereits vorhandenen Null-Version oder einer Null-Löschmarkierung und zur Generierung einer neuen Null-Löschmarkierung. Der `x-amz-delete-marker` Antwortheader wird auf gesetzt zurückgegeben `true`. **Hinweis**: In bestimmten Fällen können für ein Objekt mehrere Löschen-Marker vorhanden sein. Weitere Informationen zum Löschen von Objektversionen im GOVERNANCE-Modus finden Sie unter"Konfigurieren Sie die S3-Objektsperre über die S3-REST-API".

Betrieb	Implementierung
Objekte deObjekteObjekte (Zuvor benanntes DELETE mehrere Objekte)	Multi-Faktor-Authentifizierung (MFA) und der Antwortheader `x-amz-mfa` werden nicht unterstützt. In derselben Anforderungsmeldung können mehrere Objekte gelöscht werden. Weitere Informationen zum Löschen von Objektversionen im GOVERNANCE-Modus finden Sie unter"Konfigurieren Sie die S3-Objektsperre über die S3-REST-API".
DeleteObjectTagging	Verwendet die `tagging` Unterressource, um alle Tags aus einem Objekt zu entfernen. **Versionierung** Wenn der `versionId` Abfrageparameter in der Anforderung nicht angegeben ist, werden alle Tags aus der neuesten Version des Objekts in einem versionierten Bucket gelöscht. Wenn es sich bei der aktuellen Version des Objekts um eine Löschmarkierung handelt, wird der Status „MethodenNotAllowed“ zurückgegeben, wobei der `x-amz-delete-marker` Antwortkopf auf gesetzt `true` ist.
GetObject	"GetObject"
GetObjectAcl	Wenn für das Konto die erforderlichen Zugangsdaten bereitgestellt werden, gibt der Vorgang eine positive Antwort und die ID, DisplayName und die Berechtigung des Objekteigentümers zurück und gibt an, dass der Eigentümer vollen Zugriff auf das Objekt hat.
GetObjectLegalHold	"Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"
GetObjectRetention	"Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"
GetObjectTagging	Verwendet die `tagging` Unterressource, um alle Tags für ein Objekt zurückzugeben. **Versionierung** Wenn der `versionId` Abfrageparameter in der Anforderung nicht angegeben ist, gibt der Vorgang alle Tags der neuesten Version des Objekts in einem versionierten Bucket zurück. Wenn es sich bei der aktuellen Version des Objekts um eine Löschmarkierung handelt, wird der Status „MethodenNotAllowed“ zurückgegeben, wobei der `x-amz-delete-marker` Antwortkopf auf gesetzt `true` ist.
HeadObject	"HeadObject"
Objekt restoreObject	"Objekt restoreObject"

Betrieb	Implementierung
PutObject	"PutObject"
CopyObject (Zuvor PUT Object – Copy genannt)	"CopyObject"
PutObjectLegalHold	"Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"
PutObjectRetention	"Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"

Betrieb	Implementierung
PutObjectTagging	Verwendet die `tagging` Unterressource, um einem vorhandenen Objekt einen Satz von Tags hinzuzufügen. **Grenzwerte für Objekt-Tags** Sie können neue Objekte mit Tags hinzufügen, wenn Sie sie hochladen, oder Sie können sie zu vorhandenen Objekten hinzufügen. StorageGRID und Amazon S3 unterstützen bis zu 10 Tags für jedes Objekt. Tags, die einem Objekt zugeordnet sind, müssen über eindeutige Tag-Schlüssel verfügen. Ein Tag-Schlüssel kann bis zu 128 Unicode-Zeichen lang sein, und Tag-Werte können bis zu 256 Unicode-Zeichen lang sein. Bei den Schlüsseln und Werten wird die Groß-/Kleinschreibung beachtet. **Tag-Updates und Ingest-Verhalten** Wenn Sie PutObjectTagging verwenden, um die Tags eines Objekts zu aktualisieren, nimmt StorageGRID das Objekt nicht erneut auf. Das bedeutet, dass die in der übereinstimmenden ILM-Regel angegebene Option für das Aufnahmeverhalten nicht verwendet wird. Sämtliche durch das Update ausgelösten Änderungen an der Objektplatzierung werden vorgenommen, wenn ILM durch normale ILM-Prozesse im Hintergrund neu bewertet wird. Das heißt, wenn die ILM-Regel die strikte Option für das Aufnahmeverhalten verwendet, werden keine Maßnahmen ergriffen, wenn die erforderlichen Objektplatzierungen nicht vorgenommen werden können (z. B. weil ein neu erforderlicher Speicherort nicht verfügbar ist). Das aktualisierte Objekt behält seine aktuelle Platzierung bei, bis die erforderliche Platzierung möglich ist. **Konflikte lösen** Widersprüchliche Clientanforderungen, wie z. B. zwei Clients, die in denselben Schlüssel schreiben, werden auf der Grundlage der „neuesten Wins“ gelöst. Der Zeitpunkt für die Bewertung „neuester Erfolge“ basiert auf dem Zeitpunkt, an dem das StorageGRID System eine bestimmte Anforderung abgeschlossen hat und nicht auf dem Zeitpunkt, an dem S3-Clients einen Vorgang starten. **Versionierung** Wenn der `versionId` Abfrageparameter in der Anforderung nicht angegeben ist, fügt der Vorgang Tags zur neuesten Version des Objekts in einem versionierten Bucket hinzu. Wenn es sich bei der aktuellen Version des Objekts um eine Löschmarkierung handelt, wird der Status „MethodenNotAllowed“ zurückgegeben, wobei der `x-amz-delete-marker` Antwortkopf auf gesetzt `true` ist.
SelektierObjectContent	"SelektierObjectContent"

Verwenden Sie S3 Select

StorageGRID unterstützt die folgenden Amazon S3 Select-Klauseln, Datentypen und

Operatoren für die "SelectObjectContent, Befehl".

Nicht aufgeführte Elemente werden nicht unterstützt.

Syntax siehe "SelektierObjectContent". Weitere Informationen zu S3 Select finden Sie im "AWS-Dokumentation für S3 Select".

Nur Mandantenkonten, für die S3 Select aktiviert ist, können SelectObjectContent-Abfragen ausgeben. Siehe "Überlegungen und Anforderungen bei der Verwendung von S3 Select".

Klauseln

- Wählen Sie die Liste aus
- FROM-Klausel
- WHERE-Klausel
- BEGRENZUNGSKLAUSEL

Datentypen

- bool
- Ganzzahl
- Zeichenfolge
- Schweben
- Dezimal, numerisch
- Zeitstempel

Operatoren

Logische Operatoren

- UND
- NICHT
- ODER

Vergleichsoperatoren

- <
- >
- ≪=
- >=
- =
- =
- <>
- !=
- ZWISCHEN

- IN

Operatoren für die Musteranpassung

- GEFÄLLT MIR
- _
- %

Einheitliche Operatoren

- IST NULL
- IST NICHT NULL

Mathematische Operatoren

- +
- -
- *
- /
- %

StorageGRID folgt der Priorität des Amazon S3 Select-Operators.

Aggregatfunktionen

- DURCHSCHN.()
- ANZAHL (*)
- MAX.()
- MIN.()
- SUMME()

Bedingte Funktionen

- FALL
- ZUSAMMENSCHMELZEN
- NULL LIF

Konvertierungsfunktionen

- CAST (für unterstützten Datentyp)

Datumsfunktionen

- DATUM_HINZUFÜGEN
- DATE_DIFF
- EXTRAHIEREN
- TO_STRING

- TO_ZEITSTEMPEL
- UTCNOW

Zeichenfolgenfunktionen

- CHAR_LENGTH, CHARACTER_LENGTH
- NIEDRIGER
- TEILSTRING
- TRIMMEN
- OBEN

Serverseitige Verschlüsselung

Die serverseitige Verschlüsselung schützt Ihre Objektdaten im Ruhezustand. StorageGRID verschlüsselt die Daten beim Schreiben des Objekts und entschlüsselt sie beim Zugriff auf das Objekt.

Wenn Sie die serverseitige Verschlüsselung verwenden möchten, können Sie eine der zwei Optionen auswählen, die sich gegenseitig ausschließen, je nachdem, wie die Verschlüsselungsschlüssel verwaltet werden:

- **SSE (serverseitige Verschlüsselung mit von StorageGRID verwalteten Schlüsseln)**: Bei der Ausgabe einer S3-Anfrage zum Speichern eines Objekts verschlüsselt StorageGRID das Objekt mit einem eindeutigen Schlüssel. Wenn Sie zum Abrufen des Objekts eine S3-Anforderung ausstellen, entschlüsselt StorageGRID das Objekt mithilfe des gespeicherten Schlüssels.
- **SSE-C (serverseitige Verschlüsselung mit vom Kunden bereitgestellten Schlüsseln)**: Wenn Sie eine S3-Anfrage zum Speichern eines Objekts ausgeben, geben Sie Ihren eigenen Verschlüsselungsschlüssel an. Wenn Sie ein Objekt abrufen, geben Sie denselben Verschlüsselungsschlüssel wie in Ihrer Anfrage ein. Stimmen die beiden Verschlüsselungsschlüssel überein, wird das Objekt entschlüsselt und die Objektdaten zurückgegeben.

  StorageGRID managt zwar alle Objektverschlüsselung und Entschlüsselungsvorgänge, muss aber die von Ihnen zur Verfügung gelegten Verschlüsselungsschlüssel verwalten.

> Die von Ihnen zur Verfügung gelegten Schlüssel werden niemals gespeichert. Wenn Sie einen Verschlüsselungsschlüssel verlieren, verlieren Sie das entsprechende Objekt.

> Wenn ein Objekt mit SSE oder SSE-C verschlüsselt wird, werden sämtliche Verschlüsselungseinstellungen auf Bucket- oder Grid-Ebene ignoriert.

Verwenden Sie SSE

Um ein Objekt mit einem eindeutigen, von StorageGRID gemanagten Schlüssel zu verschlüsseln, verwenden Sie die folgende Anforderungsüberschrift:

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

Der SSE-Anforderungsheader wird durch die folgenden Objektoperationen unterstützt:

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

Verwenden Sie SSE-C

Um ein Objekt mit einem eindeutigen Schlüssel zu verschlüsseln, den Sie verwalten, verwenden Sie drei Anforderungsheader:

Kopfzeile der Anfrage	Beschreibung
`x-amz-server-side-encryption-customer-algorithm`	Geben Sie den Verschlüsselungsalgorithmus an. Der Kopfzeilenwert muss sein `AES256`.
`x-amz-server-side-encryption-customer-key`	Geben Sie den Verschlüsselungsschlüssel an, der zum Verschlüsseln oder Entschlüsseln des Objekts verwendet wird. Der Wert für den Schlüssel muss 256-Bit, base64-codiert sein.
`x-amz-server-side-encryption-customer-key-MD5`	Geben Sie den MD5-Digest des Verschlüsselungsschlüssels gemäß RFC 1321 an, der dafür sorgt, dass der Verschlüsselungsschlüssel fehlerfrei übertragen wurde. Der Wert für das MD5 Digest muss base64-kodiert 128-Bit sein.

Die SSE-C-Anfrageheader werden durch die folgenden Objektoperationen unterstützt:

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

Überlegungen zur Verwendung serverseitiger Verschlüsselung mit vom Kunden bereitgestellten Schlüsseln (SSE-C)

Beachten Sie vor der Verwendung von SSE-C die folgenden Punkte:

- Sie müssen https verwenden.

> StorageGRID lehnt alle Anfragen ab, die über http bei Verwendung von SSE-C gestellt werden. Aus Sicherheitsgründen sollten Sie jeden Schlüssel, den Sie versehentlich mit http senden, als kompromittiert betrachten. Entsorgen Sie den Schlüssel, und drehen Sie ihn nach Bedarf.

- Der ETag in der Antwort ist nicht das MD5 der Objektdaten.
- Sie müssen die Zuordnung von Schlüsseln zu Objekten managen. StorageGRID speichert keine Schlüssel. Sie sind für die Nachverfolgung des Verschlüsselungsschlüssels verantwortlich, den Sie für

jedes Objekt bereitstellen.

- Wenn Ihr Bucket mit Versionierung aktiviert ist, sollte für jede Objektversion ein eigener Verschlüsselungsschlüssel vorhanden sein. Sie sind verantwortlich für das Tracking des Verschlüsselungsschlüssels, der für jede Objektversion verwendet wird.
- Da Sie Verschlüsselungsschlüssel auf Client-Seite verwalten, müssen Sie auch zusätzliche Schutzmaßnahmen, wie etwa die Rotation von Schlüsseln, auf Client-Seite verwalten.

> Die von Ihnen zur Verfügung gelegten Schlüssel werden niemals gespeichert. Wenn Sie einen Verschlüsselungsschlüssel verlieren, verlieren Sie das entsprechende Objekt.

- Wenn die Grid-übergreifende Replizierung oder CloudMirror Replizierung für den Bucket konfiguriert ist, können SSE-C-Objekte nicht aufgenommen werden. Der Aufnahmevorgang schlägt fehl.

Verwandte Informationen

"Amazon S3-Benutzerhandbuch: Verwenden der serverseitigen Verschlüsselung mit vom Kunden bereitgestellten Schlüsseln (SSE-C)"

CopyObject

Sie können die S3-CopyObject-Anforderung verwenden, um eine Kopie eines Objekts zu erstellen, das bereits in S3 gespeichert ist. Eine CopyObject-Operation ist die gleiche wie GetObject gefolgt von PutObject.

Konflikte lösen

Widersprüchliche Clientanforderungen, wie z. B. zwei Clients, die in denselben Schlüssel schreiben, werden auf der Grundlage der „neuesten Wins“ gelöst. Der Zeitpunkt für die Bewertung „neuester Erfolge“ basiert auf dem Zeitpunkt, an dem das StorageGRID System eine bestimmte Anforderung abgeschlossen hat und nicht auf dem Zeitpunkt, an dem S3-Clients einen Vorgang starten.

Objektgröße

Die maximale *recommended* Größe für eine einzelne PutObject-Operation beträgt 5 gib (5,368,709,120 Bytes). Falls Objekte größer als 5 gib sind, verwenden Sie "Mehrteiliges Hochladen" stattdessen.

Die maximale *supported*-Größe für eine einzelne PutObject-Operation beträgt 5 tib (5,497,558,138,880 Bytes).

> Wenn Sie ein Upgrade von StorageGRID 11.6 oder einer älteren Version durchgeführt haben, wird die Warnmeldung „S3 PUT Object size to Large“ ausgelöst, wenn Sie versuchen, ein Objekt hochzuladen, das mehr als 5 gib überschreitet. Wenn Sie eine neue Installation von StorageGRID 11.7 oder 11.8 haben, wird die Warnmeldung in diesem Fall nicht ausgelöst. Um sich jedoch auf den AWS S3-Standard abzustimmen, werden zukünftige Versionen von StorageGRID das Hochladen von Objekten, die mehr als 5 gib betragen, nicht unterstützen.

UTF-8 Zeichen in Benutzermetadaten

Wenn eine Anfrage UTF-8-Werte im Schlüsselnamen oder -Wert der benutzerdefinierten Metadaten enthält, ist das StorageGRID-Verhalten nicht definiert.

StorageGRID parst oder interpretiert keine entgangenen UTF-8-Zeichen, die im Schlüsselnamen oder -Wert der benutzerdefinierten Metadaten enthalten sind. Entgangenen UTF-8 Zeichen werden als ASCII-Zeichen behandelt:

- Anforderungen sind erfolgreich, wenn benutzerdefinierte Metadaten entgangenen UTF-8 Zeichen enthalten.
- StorageGRID gibt den Header nicht zurück `x-amz-missing-meta`, wenn der interpretierte Wert des Schlüsselnamens oder -Wertes nicht druckbare Zeichen enthält.

Unterstützte Anfrageheader

Die folgenden Anfragezeilen werden unterstützt:

- `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-`, Gefolgt von einem Name-Wert-Paar, das benutzerdefinierte Metadaten enthält
- `x-amz-metadata-directive`: Der Standardwert ist `COPY`, mit dem Sie das Objekt und die zugehörigen Metadaten kopieren können.

  Sie können angeben `REPLACE`, die vorhandenen Metadaten beim Kopieren des Objekts zu überschreiben oder die Objektmetadaten zu aktualisieren.
- `x-amz-storage-class`
- `x-amz-tagging-directive`: Der Standardwert ist `COPY`, mit dem Sie das Objekt und alle Tags kopieren können.

  Sie können festlegen `REPLACE`, dass die vorhandenen Tags beim Kopieren des Objekts überschrieben oder die Tags aktualisiert werden sollen.
- S3-Objektsperrungs-Anfrageheader:
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    Wenn eine Anforderung ohne diese Header ausgeführt wird, werden die Standardaufbewahrungseinstellungen für Buckets verwendet, um den Versionsmodus des Objekts zu berechnen und das „behalt-bis“-Datum zu erhalten. Siehe "Konfigurieren Sie die S3-Objektsperre über die S3-REST-API".
- SSE-Anfragezeilen:
  - `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`

  Siehe Anforderungsheader für serverseitige Verschlüsselung

Nicht unterstützte Anforderungsheader

Die folgenden Anforderungsheader werden nicht unterstützt:

- `Cache-Control`
- `Content-Disposition`
- `Content-Encoding`
- `Content-Language`
- `Expires`
- `x-amz-checksum-algorithm`

  Wenn Sie ein Objekt kopieren und das Quellobjekt eine Prüfsumme hat, kopiert StorageGRID diesen Prüfsummenwert nicht auf das neue Objekt. Dieses Verhalten gilt unabhängig davon, ob Sie versuchen, in der Objektanforderung zu verwenden `x-amz-checksum-algorithm`.

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

Optionen der Storage-Klasse

Der `x-amz-storage-class` Anforderungsheader wird unterstützt und beeinflusst, wie viele Objektkopien StorageGRID erstellt, wenn die passende ILM-Regel den doppelten Commit oder den ausgewogenen verwendet"Aufnahme-Option".

- `STANDARD`

  (Standard) gibt einen Dual-Commit-Aufnahmevorgang an, wenn die ILM-Regel die Option Dual Commit verwendet oder wenn die Option Balance auf das Erstellen von Zwischenkopien zurückgreift.

- `REDUCED_REDUNDANCY`

  Gibt einen Single-Commit-Aufnahmevorgang an, wenn die ILM-Regel die Option Dual Commit verwendet oder wenn die Option Balance zur Erstellung zwischenzeitaler Kopien zurückgreift.

  > Wenn Sie ein Objekt in einen Bucket mit aktivierter S3-Objektsperrung aufnehmen, wird die `REDUCED_REDUNDANCY` Option ignoriert. Wenn Sie ein Objekt in einen Legacy-konformen Bucket aufnehmen, gibt die `REDUCED_REDUNDANCY` Option einen Fehler zurück. StorageGRID führt immer eine doppelte Einspeisung durch, um Compliance-Anforderungen zu erfüllen.

Verwenden von x-amz-copy-source in CopyObject

Wenn sich Quell-Bucket und Schlüssel, wie in der Kopfzeile angegeben `x-amz-copy-source`, vom Ziel-

Bucket und Schlüssel unterscheiden, wird eine Kopie der Quell-Objektdaten auf das Ziel geschrieben.

Wenn die Quelle und das Ziel übereinstimmen und der `x-amz-metadata-directive` Header als angegeben `REPLACE` ist, werden die Metadaten des Objekts mit den in der Anfrage angegebenen Metadatenwerten aktualisiert. In diesem Fall nimmt StorageGRID das Objekt nicht erneut auf. Dies hat zwei wichtige Folgen:

- Sie können CopyObject nicht verwenden, um ein vorhandenes Objekt zu verschlüsseln oder die Verschlüsselung eines vorhandenen Objekts zu ändern. Wenn Sie den Header oder den `x-amz-server-side-encryption-customer-algorithm` Header liefern `x-amz-server-side-encryption`, lehnt StorageGRID die Anfrage ab und gibt zurück `XNotImplemented`.
- Die in der übereinstimmenden ILM-Regel angegebene Option für das Aufnahmeverhalten wird nicht verwendet. Sämtliche durch das Update ausgelösten Änderungen an der Objektplatzierung werden vorgenommen, wenn ILM durch normale ILM-Prozesse im Hintergrund neu bewertet wird.

  Das heißt, wenn die ILM-Regel die strikte Option für das Aufnahmeverhalten verwendet, werden keine Maßnahmen ergriffen, wenn die erforderlichen Objektplatzierungen nicht vorgenommen werden können (z. B. weil ein neu erforderlicher Speicherort nicht verfügbar ist). Das aktualisierte Objekt behält seine aktuelle Platzierung bei, bis die erforderliche Platzierung möglich ist.

Anforderungsheader für serverseitige Verschlüsselung

Wenn Sie "Serverseitige Verschlüsselung verwenden", die Anfrage Header Sie angeben, hängt davon ab, ob das Quellobjekt verschlüsselt ist und ob Sie planen, das Zielobjekt zu verschlüsseln.

- Wenn das Quellobjekt mit einem vom Kunden bereitgestellten Schlüssel (SSE-C) verschlüsselt wird, müssen Sie die folgenden drei Header in die CopyObject-Anforderung aufnehmen, damit das Objekt entschlüsselt und dann kopiert werden kann:
  - `x-amz-copy-source-server-side-encryption-customer-algorithm`: Spezifizieren `AES256`.
  - `x-amz-copy-source-server-side-encryption-customer-key`: Geben Sie den Verschlüsselungsschlüssel an, den Sie beim Erstellen des Quellobjekts angegeben haben.
  - `x-amz-copy-source-server-side-encryption-customer-key-MD5`: Geben Sie den MD5-Digest an, den Sie beim Erstellen des Quellobjekts angegeben haben.
- Wenn Sie das Zielobjekt (die Kopie) mit einem eindeutigen Schlüssel verschlüsseln möchten, den Sie bereitstellen und verwalten, müssen Sie die folgenden drei Header angeben:
  - `x-amz-server-side-encryption-customer-algorithm`: Spezifizieren `AES256`.
  - `x-amz-server-side-encryption-customer-key`: Geben Sie einen neuen Verschlüsselungsschlüssel für das Zielobjekt an.
  - `x-amz-server-side-encryption-customer-key-MD5`: Geben Sie den MD5-Digest des neuen Verschlüsselungsschlüssels an.

  > Die von Ihnen zur Verfügung gelegten Schlüssel werden niemals gespeichert. Wenn Sie einen Verschlüsselungsschlüssel verlieren, verlieren Sie das entsprechende Objekt. Bevor Sie vom Kunden bereitgestellte Schlüssel zum Schutz von Objektdaten verwenden, lesen Sie die Überlegungen für "Serverseitige Verschlüsselung".

- Wenn Sie das Zielobjekt (die Kopie) mit einem eindeutigen Schlüssel verschlüsseln möchten, der von StorageGRID (SSE) verwaltet wird, fügen Sie diesen Header in die CopyObject-Anforderung ein:

- x-amz-server-side-encryption

> Der server-side-encryption Wert des Objekts kann nicht aktualisiert werden. Erstellen Sie stattdessen eine Kopie mit einem neuen server-side-encryption Wert mit x-amz-metadata-directive: REPLACE.

Versionierung

Wenn der Quell-Bucket versioniert ist, können Sie die Kopfzeile verwenden x-amz-copy-source, um die neueste Version eines Objekts zu kopieren. Um eine bestimmte Version eines Objekts zu kopieren, müssen Sie explizit die Version angeben, die mit der Unterressource kopiert werden soll versionId. Wenn der Ziel-Bucket versioniert ist, wird die generierte Version im Antwortheader zurückgegeben x-amz-version-id. Wenn die Versionierung für den Ziel-Bucket unterbrochen wird, x-amz-version-id gibt der Wert „Null“ zurück.

GetObject

Mithilfe der S3-GetObject-Anforderung können Sie ein Objekt aus einem S3-Bucket abrufen.

GetObject- und mehrteilige Objekte

Mit dem Anforderungsparameter können partNumber Sie einen bestimmten Teil eines mehrteiligen oder segmentierten Objekts abrufen. Das x-amz-mp-parts-count Antwortelement gibt an, wie viele Teile das Objekt hat.

Sie können sowohl für segmentierte/mehrteilige Objekte als auch für nicht segmentierte/nicht mehrteilige Objekte auf 1 setzen partNumber. Das Antwortelement wird jedoch x-amz-mp-parts-count nur für segmentierte oder mehrteilige Objekte zurückgegeben.

UTF-8 Zeichen in Benutzermetadaten

StorageGRID parst oder interpretiert die entgangenen UTF-8-Zeichen nicht in benutzerdefinierten Metadaten. GET Requests for an object with escaped UTF-8 characters in user-defined metadata liefern den Header nicht zurück x-amz-missing-meta, wenn der Schlüsselname oder -Wert nicht druckbare Zeichen enthält.

Unterstützte Anforderungsheader

Der folgende Anforderungskopf wird unterstützt:

- x-amz-checksum-mode: Spezifizieren ENABLED

  Der Range Header wird für GetObject nicht unterstützt x-amz-checksum-mode. Wenn Sie die Anfrage mit x-amz-checksum-mode aktiviert einbeziehen Range, gibt StorageGRID keinen Prüfsummenwert in der Antwort zurück.

Nicht unterstützte Anforderungsüberschrift

Der folgende Anforderungsheader wird nicht unterstützt und gibt zurück XNotImplemented:

- x-amz-website-redirect-location

Versionierung

Wenn `versionId` keine Unterressource angegeben wird, ruft der Vorgang die aktuellste Version des Objekts in einem versionierten Bucket ab. Wenn es sich bei der aktuellen Version des Objekts um eine Löschmarkierung handelt, wird der Status „nicht gefunden“ zurückgegeben, wobei der `x-amz-delete-marker` Antwortkopf auf gesetzt `true` ist.

Kopfzeilen zur serverseitigen Verschlüsselung mit vom Kunden bereitgestellten Verschlüsselungsschlüsseln anfordern (SSE-C)

Verwenden Sie alle drei Kopfzeilen, wenn das Objekt mit einem eindeutigen Schlüssel verschlüsselt ist, den Sie angegeben haben.

- `x-amz-server-side-encryption-customer-algorithm`: Spezifizieren `AES256`.
- `x-amz-server-side-encryption-customer-key`: Geben Sie Ihren Verschlüsselungsschlüssel für das Objekt an.
- `x-amz-server-side-encryption-customer-key-MD5`: Geben Sie den MD5-Digest des Objektschlüssels an.

> Die von Ihnen zur Verfügung gelegten Schlüssel werden niemals gespeichert. Wenn Sie einen Verschlüsselungsschlüssel verlieren, verlieren Sie das entsprechende Objekt. Bevor Sie vom Kunden bereitgestellte Schlüssel zum Schutz von Objektdaten verwenden, lesen Sie die Überlegungen in "Serverseitige Verschlüsselung".

Verhalten von GetObject for Cloud Storage Pool Objects

Wenn ein Objekt in einem gespeichert wurde"Cloud-Storage-Pool", hängt das Verhalten einer GetObject-Anforderung vom Zustand des Objekts ab. Weitere Informationen finden Sie unter"HeadObject".

> Wenn ein Objekt in einem Cloud Storage-Pool gespeichert ist und eine oder mehrere Kopien des Objekts ebenfalls im Raster vorhanden sind, versucht GetObject Requests, die Daten aus dem Raster abzurufen, bevor sie aus dem Cloud Storage-Pool abgerufen werden.

Status des Objekts	Verhalten von GetObject
Objekt, das in StorageGRID aufgenommen wurde, durch ILM jedoch noch nicht evaluiert wurde, oder Objekt, das in einem herkömmlichen Storage-Pool gespeichert ist oder Erasure Coding verwendet	`200 OK` Eine Kopie des Objekts wird abgerufen.
Objekt in Cloud-Storage-Pool, ist aber noch nicht in einen Zustand übergegangen, der nicht abrufbar ist	`200 OK` Eine Kopie des Objekts wird abgerufen.
Das Objekt wurde in einen nicht aufrufbaren Zustand überführt	`403 Forbidden`, `InvalidObjectState` Verwenden Sie eine "Objekt restoreObject" Anforderung, um das Objekt in einem abrufbaren Zustand wiederherzustellen.

Status des Objekts	Verhalten von GetObject
Objekt wird aus einem nicht aufrufbaren Zustand wiederhergestellt	`403 Forbidden`, `InvalidObjectState` Warten Sie, bis die Anforderung „RestoreObject“ abgeschlossen ist.
Das Objekt wird im Cloud-Storage-Pool vollständig wiederhergestellt	`200 OK` Eine Kopie des Objekts wird abgerufen.

Mehrteilige oder segmentierte Objekte in einem Cloud Storage-Pool

Wenn Sie ein mehrteilige Objekt hochgeladen StorageGRID oder ein großes Objekt in Segmente aufgeteilt haben, bestimmt StorageGRID, ob das Objekt im Cloud-Storage-Pool verfügbar ist, indem Sie eine Teilmenge der Teile oder Segmente des Objekts testen. In einigen Fällen kann eine GetObject-Anforderung falsch zurückgegeben werden `200 OK`, wenn einige Teile des Objekts bereits in einen nicht abrufbaren Status überführt wurden oder wenn Teile des Objekts noch nicht wiederhergestellt wurden.

In diesen Fällen:

- Die GetObject-Anforderung gibt möglicherweise einige Daten zurück, hält jedoch während der Übertragung an.
- Eine nachfolgende GetObject-Anfrage kann zurückgegeben werden `403 Forbidden`.

GetObject- und Grid-übergreifende Replikation

Wenn Sie und "Grid-übergreifende Replizierung"für einen Bucket verwenden"Grid-Verbund", kann der S3-Client den Replikationsstatus eines Objekts überprüfen, indem er eine GetObject-Anforderung ausgibt. Die Antwort enthält den StorageGRID-spezifischen `x-ntap-sg-cgr-replication-status` Antwortheader, der einen der folgenden Werte enthält:

Raster	Replikationsstatus
Quelle	• **ABGESCHLOSSEN**: Die Replikation war erfolgreich. • **AUSSTEHEND**: Das Objekt wurde noch nicht repliziert. • **FAILURE**: Die Replikation ist mit einem permanenten Fehler fehlgeschlagen. Ein Benutzer muss den Fehler beheben.
Ziel	**REPLIKAT**: Das Objekt wurde aus dem Quellraster repliziert.

Der Header wird von StorageGRID nicht unterstützt `x-amz-replication-status`.

HeadObject

Sie können die S3 HeadObject-Anforderung verwenden, um Metadaten von einem Objekt abzurufen, ohne das Objekt selbst zurückzugeben. Wenn das Objekt in einem Cloud-Speicherpool gespeichert ist, können Sie HeadObject verwenden, um den

Übergangsstatus des Objekts zu bestimmen.

HeadObject- und mehrteilige Objekte

Mit dem Anforderungsparameter können Sie `partNumber` Metadaten für einen bestimmten Teil eines mehrteiligen oder segmentierten Objekts abrufen. Das `x-amz-mp-parts-count` Antwortelement gibt an, wie viele Teile das Objekt hat.

Sie können sowohl für segmentierte/mehrteilige Objekte als auch für nicht segmentierte/nicht mehrteilige Objekte auf 1 setzen `partNumber`. Das Antwortelement wird jedoch `x-amz-mp-parts-count` nur für segmentierte oder mehrteilige Objekte zurückgegeben.

UTF-8 Zeichen in Benutzermetadaten

StorageGRID parst oder interpretiert die entgangenen UTF-8-Zeichen nicht in benutzerdefinierten Metadaten. HEAD Requests for an object with escaped UTF-8 characters in user-defined metadata liefern den Header nicht zurück `x-amz-missing-meta`, wenn der Schlüsselname oder -Wert nicht druckbare Zeichen enthält.

Unterstützte Anforderungsheader

Der folgende Anforderungskopf wird unterstützt:

- `x-amz-checksum-mode`

  ```
  `partNumber`Parameter und `Range` Header werden für HeadObject nicht unterstützt `x-amz-checksum-mode`. Wenn Sie sie in die Anfrage mit aktiviertem aufnehmen `x-amz-checksum-mode`, gibt StorageGRID keinen Prüfsummenwert in der Antwort zurück.
  ```

Nicht unterstützte Anforderungsüberschrift

Der folgende Anforderungsheader wird nicht unterstützt und gibt zurück `XNotImplemented`:

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

Versionierung

Wenn `versionId` keine Unterressource angegeben wird, ruft der Vorgang die aktuellste Version des Objekts in einem versionierten Bucket ab. Wenn es sich bei der aktuellen Version des Objekts um eine Löschmarkierung handelt, wird der Status „nicht gefunden“ zurückgegeben, wobei der `x-amz-delete-marker` Antwortkopf auf gesetzt `true` ist.

Kopfzeilen zur serverseitigen Verschlüsselung mit vom Kunden bereitgestellten Verschlüsselungsschlüsseln anfordern (SSE-C)

Verwenden Sie alle drei dieser Kopfzeilen, wenn das Objekt mit einem eindeutigen Schlüssel verschlüsselt ist, den Sie angegeben haben.

- `x-amz-server-side-encryption-customer-algorithm`: Spezifizieren `AES256`.
- `x-amz-server-side-encryption-customer-key`: Geben Sie Ihren Verschlüsselungsschlüssel für das Objekt an.

- `x-amz-server-side-encryption-customer-key-MD5`: Geben Sie den MD5-Digest des Objektschlüssels an.

> Die von Ihnen zur Verfügung gelegten Schlüssel werden niemals gespeichert. Wenn Sie einen Verschlüsselungsschlüssel verlieren, verlieren Sie das entsprechende Objekt. Bevor Sie vom Kunden bereitgestellte Schlüssel zum Schutz von Objektdaten verwenden, lesen Sie die Überlegungen in "Serverseitige Verschlüsselung".

HeadObject-Antworten für Cloud-Storage-Pool-Objekte

Wenn das Objekt in einem gespeichert ist"Cloud-Storage-Pool", werden die folgenden Antwortkopfzeilen zurückgegeben:

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

Die Antwortheader liefern Informationen zum Status eines Objekts beim Verschieben in einen Cloud Storage Pool, beim Wechsel in einen nicht abrufbaren Zustand und wieder verfügbar.

Status des Objekts	Antwort auf HeadObject
Objekt, das in StorageGRID aufgenommen wurde, durch ILM jedoch noch nicht evaluiert wurde, oder Objekt, das in einem herkömmlichen Storage-Pool gespeichert ist oder Erasure Coding verwendet	`200 OK` (Es wird keine spezielle Antwortheader zurückgegeben.)

Objekt in Cloud-Storage-Pool, ist aber noch nicht in einen Zustand übergegangen, der nicht abrufbar ist	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"`
	
	Bis das Objekt in einen nicht abrufbaren Zustand übergeht, wird der Wert für `expiry-date` in der Zukunft auf eine ferne Zeit gesetzt. Die genaue Zeit der Transition wird nicht durch das StorageGRID System gesteuert.

Status des Objekts	Antwort auf HeadObject

Das Objekt ist in den nicht aufrufbaren Zustand übergegangen, aber mindestens eine Kopie ist auch im Grid vorhanden	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"`
	
	Der Wert für `expiry-date` wird auf eine ferne Zeit in der Zukunft gesetzt.
	
	Hinweis: Wenn die Kopie im Raster nicht verfügbar ist (z. B. ein Storage Node ist ausgefallen), müssen Sie eine Anforderung zur Wiederherstellung der Kopie aus dem Cloud Storage Pool ausgeben"Objekt restoreObject", bevor Sie das Objekt erfolgreich abrufen können.

Das Objekt wurde in einen nicht abrufbaren Zustand versetzt, und es ist keine Kopie im Grid vorhanden	`200 OK`
	
	`x-amz-storage-class: GLACIER`

Objekt wird aus einem nicht aufrufbaren Zustand wiederhergestellt	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="true"`

Das Objekt wird im Cloud-Storage-Pool vollständig wiederhergestellt	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2018 00:00:00 GMT"`
	
	Das `expiry-date` gibt an, wann das Objekt im Cloud-Speicherpool in einen nicht abrufbaren Zustand zurückkehrt.

Mehrteilige oder segmentierte Objekte in Cloud Storage Pool

Wenn Sie ein mehrteilige Objekt hochgeladen StorageGRID oder ein großes Objekt in Segmente aufgeteilt haben, bestimmt StorageGRID, ob das Objekt im Cloud-Storage-Pool verfügbar ist, indem Sie eine Teilmenge der Teile oder Segmente des Objekts testen. In einigen Fällen kann eine HeadObject-Anforderung falsch zurückgegeben werden `x-amz-restore: ongoing-request="false"`, wenn einige Teile des Objekts bereits in einen nicht abrufbaren Status überführt wurden oder wenn Teile des Objekts noch nicht wiederhergestellt wurden.

HeadObject- und Grid-übergreifende Replikation

Wenn Sie und "Grid-übergreifende Replizierung"für einen Bucket verwenden"Grid-Verbund", kann der S3-Client mit einer HeadObject-Anforderung den Replikationsstatus eines Objekts überprüfen. Die Antwort enthält den StorageGRID-spezifischen `x-ntap-sg-cgr-replication-status` Antwortheader, der einen der folgenden Werte enthält:

Raster	Replikationsstatus
Quelle	• **ABGESCHLOSSEN**: Die Replikation war erfolgreich. • **AUSSTEHEND**: Das Objekt wurde noch nicht repliziert. • **FAILURE**: Die Replikation ist mit einem permanenten Fehler fehlgeschlagen. Ein Benutzer muss den Fehler beheben.
Ziel	**REPLIKAT**: Das Objekt wurde aus dem Quellraster repliziert.

Der Header wird von StorageGRID nicht unterstützt `x-amz-replication-status`.

PutObject

Sie können die S3 PutObject-Anforderung verwenden, um einem Bucket ein Objekt hinzuzufügen.

Konflikte lösen

Widersprüchliche Clientanforderungen, wie z. B. zwei Clients, die in denselben Schlüssel schreiben, werden auf der Grundlage der „neuesten Wins“ gelöst. Der Zeitpunkt für die Bewertung „neuester Erfolge“ basiert auf dem Zeitpunkt, an dem das StorageGRID System eine bestimmte Anforderung abgeschlossen hat und nicht auf dem Zeitpunkt, an dem S3-Clients einen Vorgang starten.

Objektgröße

Die maximale *recommended* Größe für eine einzelne PutObject-Operation beträgt 5 gib (5,368,709,120 Bytes). Falls Objekte größer als 5 gib sind, verwenden Sie "Mehrteiliges Hochladen" stattdessen.

Die maximale *supported*-Größe für eine einzelne PutObject-Operation beträgt 5 tib (5,497,558,138,880 Bytes).

Wenn Sie ein Upgrade von StorageGRID 11.6 oder einer älteren Version durchgeführt haben, wird die Warnmeldung „S3 PUT Object size to Large“ ausgelöst, wenn Sie versuchen, ein Objekt hochzuladen, das mehr als 5 gib überschreitet. Wenn Sie eine neue Installation von StorageGRID 11.7 oder 11.8 haben, wird die Warnmeldung in diesem Fall nicht ausgelöst. Um sich jedoch auf den AWS S3-Standard abzustimmen, werden zukünftige Versionen von StorageGRID das Hochladen von Objekten, die mehr als 5 gib betragen, nicht unterstützen.

Größe der Benutzer-Metadaten

Amazon S3 begrenzt die Größe der benutzerdefinierten Metadaten innerhalb jeder PUT-Anforderung-Kopfzeile auf 2 KB. StorageGRID begrenzt die Benutzermetadaten auf 24 KiB. Die Größe der benutzerdefinierten Metadaten wird gemessen, indem die Summe der Anzahl Bytes in der UTF-8-Codierung jedes Schlüssels und jeden Wert angegeben wird.

UTF-8 Zeichen in Benutzermetadaten

Wenn eine Anfrage UTF-8-Werte im Schlüsselnamen oder -Wert der benutzerdefinierten Metadaten enthält, ist das StorageGRID-Verhalten nicht definiert.

StorageGRID parst oder interpretiert keine entgangenen UTF-8-Zeichen, die im Schlüsselnamen oder -Wert der benutzerdefinierten Metadaten enthalten sind. Entgangenen UTF-8 Zeichen werden als ASCII-Zeichen behandelt:

- PutObject-, CopyObject-, GetObject- und HeadObject-Anfragen werden erfolgreich ausgeführt, wenn benutzerdefinierte Metadaten UTF-8-Zeichen enthalten.
- StorageGRID gibt den Header nicht zurück `x-amz-missing-meta`, wenn der interpretierte Wert des Schlüsselnamens oder -Wertes nicht druckbare Zeichen enthält.

Grenzwerte für Objekt-Tags

Sie können neue Objekte mit Tags hinzufügen, wenn Sie sie hochladen, oder Sie können sie zu vorhandenen Objekten hinzufügen. StorageGRID und Amazon S3 unterstützen bis zu 10 Tags für jedes Objekt. Tags, die einem Objekt zugeordnet sind, müssen über eindeutige Tag-Schlüssel verfügen. Ein Tag-Schlüssel kann bis zu 128 Unicode-Zeichen lang sein, und Tag-Werte können bis zu 256 Unicode-Zeichen lang sein. Bei den Schlüsseln und Werten wird die Groß-/Kleinschreibung beachtet.

Objekteigentümer

In StorageGRID sind alle Objekte Eigentum des Bucket-Besitzers-Kontos, einschließlich der Objekte, die von einem Konto ohne Eigentümer oder einem anonymen Benutzer erstellt wurden.

Unterstützte Anfrageheader

Die folgenden Anfragezeilen werden unterstützt:

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

  Wenn Sie für `Content-Encoding`StorageGRID angeben, `aws-chunked` werden die folgenden Elemente nicht überprüft:

  - StorageGRID überprüft die nicht `chunk-signature` mit den Chunk-Daten.
  - StorageGRID überprüft nicht den Wert, den Sie für für für das Objekt angeben `x-amz-decoded-content-length`.
- `Content-Language`
- `Content-Length`
- `Content-MD5`
- `Content-Type`
- `Expires`
- `Transfer-Encoding`

  Die Chunked-Transferkodierung wird unterstützt, wenn `aws-chunked` auch das Signieren der Nutzlast

verwendet wird.

- `x-amz-checksum-sha256`
- `x-amz-meta-`, Gefolgt von einem Name-Wert-Paar, das benutzerdefinierte Metadaten enthält.

  Verwenden Sie bei der Angabe des Name-value-Paars für benutzerdefinierte Metadaten dieses allgemeine Format:

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

  Wenn Sie die Option **Benutzerdefinierte Erstellungszeit** als Referenzzeit für eine ILM-Regel verwenden möchten, müssen Sie als Name der Metadaten verwenden, `creation-time` die beim Erstellen des Objekts aufgezeichnet werden. Beispiel:

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

  Der Wert für `creation-time` wird seit dem 1. Januar 1970 als Sekunden ausgewertet.

  > Eine ILM-Regel kann nicht sowohl eine **benutzerdefinierte Erstellungszeit** für die Referenzzeit als auch die Option Balanced oder Strict Ingest verwenden. Beim Erstellen der ILM-Regel wird ein Fehler zurückgegeben.

- `x-amz-tagging`
- S3-Objektsperrungs-Anfrageheader
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    Wenn eine Anforderung ohne diese Header ausgeführt wird, werden die Standardaufbewahrungseinstellungen für Buckets verwendet, um den Versionsmodus des Objekts zu berechnen und das „behalt-bis“-Datum zu erhalten. Siehe "Konfigurieren Sie die S3-Objektsperre über die S3-REST-API".

- SSE-Anfragezeilen:
  - `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`

    Siehe Anforderungsheader für serverseitige Verschlüsselung

Nicht unterstützte Anforderungsheader

Die folgenden Anforderungsheader werden nicht unterstützt:

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

  Der `x-amz-website-redirect-location` Header gibt zurück `XNotImplemented`.

Optionen der Storage-Klasse

Der `x-amz-storage-class` Anforderungskopf wird unterstützt. Der für eingereichte Wert `x-amz-storage-class` hat einen Einfluss darauf, wie StorageGRID Objektdaten bei der Aufnahme schützt, und nicht darauf, wie viele persistente Kopien des Objekts im StorageGRID System gespeichert werden (durch ILM bestimmt).

Wenn die ILM-Regel, die einem aufgenommenen Objekt entspricht, die Option „Strict Ingest“ verwendet, hat der `x-amz-storage-class` Header keine Auswirkungen.

Folgende Werte können verwendet werden für `x-amz-storage-class`:

- `STANDARD` (Standard)
  - **Dual Commit**: Wenn die ILM-Regel die Dual Commit-Option für das Aufnahmeverhalten angibt, sobald ein Objekt aufgenommen wird, wird eine zweite Kopie dieses Objekts erstellt und auf einen anderen Storage Node verteilt (Dual Commit). Bei Bewertung des ILM bestimmt StorageGRID, ob diese ersten Zwischenkopien die Anweisungen zur Platzierung in der Regel erfüllen. Ist dies nicht der Fall, müssen möglicherweise neue Objektkopien an unterschiedlichen Standorten erstellt werden, und die ersten Zwischenkopien müssen eventuell gelöscht werden.
  - **Ausgeglichen**: Wenn die ILM-Regel die Option ausgeglichen angibt und StorageGRID nicht sofort alle in der Regel angegebenen Kopien erstellen kann, erstellt StorageGRID zwei Zwischenkopien auf verschiedenen Speicherknoten.

    Wenn StorageGRID sofort alle in der ILM-Regel angegebenen Objektkopien erstellen kann (synchrone Platzierung), hat der `x-amz-storage-class` Header keine Auswirkungen.

- `REDUCED_REDUNDANCY`
  - **Dual Commit**: Wenn die ILM-Regel die Dual Commit-Option für das Aufnahmeverhalten angibt, erstellt StorageGRID bei Aufnahme des Objekts eine einzelne Interimskopie (Single Commit).
  - **Ausgeglichen**: Wenn die ILM-Regel die Option ausgeglichen angibt, erstellt StorageGRID nur eine Zwischenkopie, wenn das System nicht sofort alle in der Regel angegebenen Kopien erstellen kann. Wenn StorageGRID eine synchrone Platzierung durchführen kann, hat diese Kopfzeile keine Auswirkung. Diese `REDUCED_REDUNDANCY` Option ist am besten geeignet, wenn die mit dem Objekt übereinstimmende ILM-Regel eine einzige replizierte Kopie erstellt. In diesem Fall `REDUCED_REDUNDANCY` entfällt bei jedem Einspielvorgang die unnötige Erstellung und Löschung einer zusätzlichen Objektkopie.

  Die Verwendung der `REDUCED_REDUNDANCY` Option wird in anderen Fällen nicht empfohlen. `REDUCED_REDUNDANCY` Erhöhtes Risiko von Objektdatenverlusten bei der Aufnahme. Beispielsweise können Sie Daten verlieren, wenn die einzelne Kopie zunächst auf einem Storage Node gespeichert wird, der ausfällt, bevor eine ILM-Evaluierung erfolgen kann.

Da nur eine Kopie zu einem beliebigen Zeitpunkt repliziert werden kann, sind Daten einem ständigen Verlust ausgesetzt. Wenn nur eine replizierte Kopie eines Objekts vorhanden ist, geht dieses Objekt verloren, wenn ein Speicherknoten ausfällt oder einen beträchtlichen Fehler hat. Während Wartungsarbeiten wie Upgrades verlieren Sie auch vorübergehend den Zugriff auf das Objekt.

Die Angabe `REDUCED_REDUNDANCY` wirkt sich nur darauf aus, wie viele Kopien erstellt werden, wenn ein Objekt zum ersten Mal aufgenommen wird. Sie wirkt sich nicht darauf aus, wie viele Kopien des Objekts erstellt werden, wenn das Objekt durch die aktiven ILM-Richtlinien evaluiert wird, und führt nicht dazu, dass Daten mit niedrigerer Redundanz im StorageGRID System gespeichert werden.

Wenn Sie ein Objekt in einen Bucket mit aktivierter S3-Objektsperrung aufnehmen, wird die `REDUCED_REDUNDANCY` Option ignoriert. Wenn Sie ein Objekt in einen Legacy-konformen Bucket aufnehmen, gibt die `REDUCED_REDUNDANCY` Option einen Fehler zurück. StorageGRID führt immer eine doppelte Einspeisung durch, um Compliance-Anforderungen zu erfüllen.

Anforderungsheader für serverseitige Verschlüsselung

Sie können die folgenden Anforderungsheader verwenden, um ein Objekt mit serverseitiger Verschlüsselung zu verschlüsseln. Die Optionen SSE und SSE-C schließen sich gegenseitig aus.

- **SSE**: Verwenden Sie den folgenden Header, wenn Sie das Objekt mit einem eindeutigen Schlüssel verschlüsseln möchten, der von StorageGRID verwaltet wird.
  - `x-amz-server-side-encryption`

    Wenn der `x-amz-server-side-encryption` Header nicht in der PutObject-Anforderung enthalten ist, wird das Grid-wide aus der PutObject-"Einstellung für die Verschlüsselung gespeicherter Objekte"Antwort weggelassen.
- **SSE-C**: Verwenden Sie alle drei dieser Header, wenn Sie das Objekt mit einem eindeutigen Schlüssel verschlüsseln möchten, den Sie bereitstellen und verwalten.
  - `x-amz-server-side-encryption-customer-algorithm`: Spezifizieren `AES256`.
  - `x-amz-server-side-encryption-customer-key`: Geben Sie Ihren Verschlüsselungsschlüssel für das neue Objekt an.
  - `x-amz-server-side-encryption-customer-key-MD5`: Geben Sie den MD5-Digest des Verschlüsselungsschlüssels des neuen Objekts an.

Die von Ihnen zur Verfügung gelegten Schlüssel werden niemals gespeichert. Wenn Sie einen Verschlüsselungsschlüssel verlieren, verlieren Sie das entsprechende Objekt. Bevor Sie vom Kunden bereitgestellte Schlüssel zum Schutz von Objektdaten verwenden, lesen Sie die Überlegungen für "Serverseitige Verschlüsselung".

Wenn ein Objekt mit SSE oder SSE-C verschlüsselt wird, werden sämtliche Verschlüsselungseinstellungen auf Bucket- oder Grid-Ebene ignoriert.

Versionierung

Wenn die Versionierung für einen Bucket aktiviert ist, wird automatisch ein eindeutiges `versionId` Objekt für die Version des gespeicherten Objekts generiert. Dies `versionId` wird auch in der Antwort über den Antwortheader zurückgegeben `x-amz-version-id`.

Wenn die Versionierung unterbrochen wird, wird die Objektversion mit einer Null gespeichert `versionId` und wenn bereits eine Null-Version vorhanden ist, wird sie überschrieben.

Signaturberechnungen für den Autorisierungskopf

Bei der Verwendung des `Authorization` Headers zur Authentifizierung von Anfragen unterscheidet sich StorageGRID von AWS folgendermaßen:

- StorageGRID erfordert nicht, `host` dass Header in enthalten `CanonicalHeaders` sind.
- StorageGRID muss nicht `Content-Type` in enthalten sein `CanonicalHeaders`.
- StorageGRID erfordert nicht, `x-amz-*` dass Header in enthalten `CanonicalHeaders` sind.

> Als allgemeine Best Practice sollten Sie diese Kopfzeilen immer in einschließen `CanonicalHeaders`, um sicherzustellen, dass sie verifiziert sind. Wenn Sie diese Kopfzeilen jedoch ausschließen, gibt StorageGRID keinen Fehler zurück.

Weitere Informationen finden Sie unter "Signaturberechnungen für den Autorisierungskopf: Payload in einem einzelnen Chunk übertragen (AWS Signature Version 4)".

Verwandte Informationen

- "Objektmanagement mit ILM"
- "Amazon Simple Storage Service API-Referenz: PutObject"

Objekt restoreObject

Sie können die S3-Wiederherstellungs-Objekt-Anforderung verwenden, um ein Objekt wiederherzustellen, das in einem Cloud-Storage-Pool gespeichert ist.

Unterstützter Anforderungstyp

StorageGRID unterstützt nur RestoreObject-Anfragen zur Wiederherstellung eines Objekts. Die Art der Wiederherstellung wird nicht unterstützt `SELECT`. Wählen Sie Rückgabeanforderungen `XNotImplemented`.

Versionierung

Optional können Sie angeben `versionId`, eine bestimmte Version eines Objekts in einem versionierten Bucket wiederherzustellen. Wenn Sie nicht angeben `versionId`, wird die neueste Version des Objekts wiederhergestellt

Verhalten von RestoreObject auf Cloud-Storage-Pool-Objekten

Wenn ein Objekt in einem gespeichert wurde"Cloud-Storage-Pool", hat eine RestoreObject-Anforderung das folgende Verhalten, basierend auf dem Zustand des Objekts. Weitere Informationen finden Sie unter"HeadObject".

> Wenn ein Objekt in einem Cloud-Storage-Pool gespeichert ist und eine oder mehrere Kopien des Objekts auch im Raster vorhanden sind, besteht keine Notwendigkeit, das Objekt durch Ausgabe einer RestoreObject-Anforderung wiederherzustellen. Stattdessen kann die lokale Kopie mithilfe einer GetObject-Anforderung direkt abgerufen werden.

Status des Objekts	Verhalten von RestoreObject
Objekt wird in StorageGRID aufgenommen, aber noch nicht durch ILM evaluiert oder Objekt befindet sich nicht in einem Cloud-Storage-Pool	`403 Forbidden`, `InvalidObjectState`
Objekt in Cloud-Storage-Pool, ist aber noch nicht in einen Zustand übergegangen, der nicht abrufbar ist	`200 OK` Es werden keine Änderungen vorgenommen. **Hinweis**: Bevor ein Objekt in einen nicht-abrufbaren Zustand überführt wurde, kann es nicht geändert werden `expiry-date`.
Das Objekt wurde in einen nicht aufrufbaren Zustand überführt	`202 Accepted` Stellt eine abrufbare Kopie des Objekts für die im Anforderungskörper angegebene Anzahl von Tagen im Cloud-Speicherpool wieder her. Am Ende dieses Zeitraums wird das Objekt in einen nicht aufrufbaren Zustand zurückgeführt. Verwenden Sie optional das `Tier` Anforderungselement, um festzulegen, wie lange der Wiederherstellungsjob dauern wird(`Expedited` `Standard`, bis er beendet ist, , oder `Bulk`). Wenn Sie nicht angeben `Tier`, wird der `Standard` Tier verwendet. **Wichtig**: Wenn ein Objekt in S3 Glacier Deep Archive migriert wurde oder der Cloud Storage Pool Azure Blob Storage verwendet, können Sie es nicht mithilfe der Tier wiederherstellen `Expedited`. Der folgende Fehler wird zurückgegeben `403 Forbidden`, `InvalidTier`: `Retrieval option is not supported by this storage class`.
Objekt wird aus einem nicht aufrufbaren Zustand wiederhergestellt	`409 Conflict`, `RestoreAlreadyInProgress`
Das Objekt wird im Cloud-Storage-Pool vollständig wiederhergestellt	`200 OK` **Hinweis:** Wenn ein Objekt in einen abrufbaren Zustand zurückgestellt wurde, können Sie es ändern `expiry-date`, indem Sie die RestoreObject-Anfrage mit einem neuen Wert für neu ausgeben `Days`. Das Wiederherstellungsdatum wird zum Zeitpunkt der Anfrage aktualisiert.

SelektierObjectContent

Sie können die S3 SelectObjectContent-Anfrage verwenden, um den Inhalt eines S3-Objekts anhand einer einfachen SQL-Anweisung zu filtern.

Weitere Informationen finden Sie unter "Amazon Simple Storage Service API Reference: SelectObjectContent".

Bevor Sie beginnen

- Das Mandantenkonto hat die S3 Select-Berechtigung.
- Sie haben `s3:GetObject` die Berechtigung für das Objekt, das Sie abfragen möchten.
- Das Objekt, das Sie abfragen möchten, muss eines der folgenden Formate aufweisen:
  - **CSV**. Kann wie ist verwendet oder in GZIP- oder BZIP2-Archiven komprimiert werden.
  - **Parkett**. Zusätzliche Anforderungen an Parkett-Objekte:
    - S3 Select unterstützt nur Spaltenkomprimierung mit GZIP oder Snappy. S3 Select unterstützt keine Komprimierung ganzer Objekte für Parkett-Objekte.
    - S3 Select unterstützt keine Parkett-Ausgabe. Sie müssen das Ausgabeformat als CSV oder JSON angeben.
    - Die maximale Größe der nicht komprimierten Zeilengruppe beträgt 512 MB.
    - Sie müssen die im Objektschema angegebenen Datentypen verwenden.
    - Sie können KEINE logischen TYPEN VON INTERVALL, JSON, LISTE, ZEIT oder UUID verwenden.
- Ihr SQL-Ausdruck hat eine maximale Länge von 256 KB.
- Jeder Datensatz im Eingang oder Ergebnis hat eine maximale Länge von 1 MiB.

Beispiel für eine CSV-Anfrage-Syntax

```
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>
```

Beispiel für die Syntax der Parkettanforderung

```
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>
```

Beispiel für eine SQL-Abfrage

Diese Abfrage erhält den Staatsnamen, 2010 Populationen, geschätzte 2015 Populationen und den Prozentsatz der Änderung von den Daten der US-Volkszählung. Datensätze in der Datei, die keine Status sind, werden ignoriert.

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

Die ersten Zeilen der Datei, die abgefragt werden sollen, `SUB-EST2020_ALL.csv` sehen wie folgt aus:

```
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
```

Beispiel für die Verwendung von AWS und CLI (CSV)

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

Die ersten Zeilen der Ausgabedatei, `changes.csv`, sehen wie folgt aus:

```
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
```

Beispiel für die Nutzung von AWS-CLI (Parkett)

```
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
```

Die ersten Zeilen der Ausgabedatei, changes.csv, sehen wie folgt aus:

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

Vorgänge für mehrteilige Uploads

Vorgänge für mehrteilige Uploads

In diesem Abschnitt wird beschrieben, wie StorageGRID Vorgänge für mehrteilige Uploads unterstützt.

Die folgenden Bedingungen und Hinweise gelten für alle mehrteiligen Uploadvorgänge:

- Sie sollten 1,000 gleichzeitige mehrteilige Uploads auf einen einzelnen Bucket nicht überschreiten, da die Ergebnisse von ListMultipartUploads Abfragen für diesen Bucket möglicherweise unvollständige Ergebnisse liefern.
- StorageGRID setzt AWS Größenbeschränkungen für mehrere Teile durch. S3-Clients müssen folgende Richtlinien einhalten:
  - Jedes Teil eines mehrteiligen Uploads muss zwischen 5 MiB (5,242,880 Byte) und 5 gib (5,368,709,120 Byte) liegen.
  - Der letzte Teil kann kleiner als 5 MiB (5,242,880 Byte) sein.
  - Im Allgemeinen sollten die Teilemaße so groß wie möglich sein. Verwenden Sie z. B. für ein Objekt mit 100 gib die Teilenummer 5 gib. Da jedes Teil als ein eindeutiges Objekt angesehen wird, sinkt der Overhead für StorageGRID Metadaten durch die Verwendung großer Teilgrößen.
  - Verwenden Sie für Objekte, die kleiner als 5 gib sind, stattdessen einen Upload ohne mehrere Teile.
- ILM wird für jeden Teil eines mehrteiligen Objekts in der Aufnahme und für das Objekt als Ganzes nach Abschluss des mehrteiligen Uploads ausgewertet, wenn die ILM-Regel die ausgewogene oder strikte verwendet"Aufnahme-Option". Sie sollten sich bewusst sein, wie dies die Objekt- und Teileplatzierung beeinflusst:
  - Wenn sich ILM ändert, während ein S3-Multipart-Upload durchgeführt wird, erfüllen einige Teile des

Objekts möglicherweise nicht die aktuellen ILM-Anforderungen, wenn der mehrteilige Upload abgeschlossen ist. Alle nicht korrekt platzierten Teile werden in die Warteschlange zur erneuten ILM-Bewertung gestellt und später an den richtigen Ort verschoben.

  - Bei der Evaluierung von ILM für ein Teil filtert StorageGRID nach der Größe des Teils und nicht der Größe des Objekts. Das bedeutet, dass Teile eines Objekts an Orten gespeichert werden können, die die ILM-Anforderungen für das gesamte Objekt nicht erfüllen. Wenn z. B. in einer Regel festgelegt wird, dass alle Objekte mit 10 GB oder mehr bei DC1 gespeichert werden, während alle kleineren Objekte bei DC2 gespeichert sind, wird jeder 1-GB-Teil eines 10-teiligen mehrteiligen Uploads bei DC2 beim Einspielen gespeichert. Wird ILM für das gesamte Objekt evaluiert, werden alle Teile des Objekts nach DC1 verschoben.

- Alle mehrteiligen Uploads unterstützen StorageGRID "Konsistenzwerte".
- Wenn ein Objekt mit mehrteiligen Uploads aufgenommen wird, wird das "Schwellenwert für Objektsegmentierung (1 gib)"nicht angewendet.
- Je nach Bedarf können Sie mit mehrteiligen Uploads verwenden"Serverseitige Verschlüsselung". Um SSE (serverseitige Verschlüsselung mit StorageGRID-verwalteten Schlüsseln) zu verwenden, fügen Sie den `x-amz-server-side-encryption` Anforderungsheader nur in die CreateMultipartUpload-Anforderung ein. Um SSE-C (serverseitige Verschlüsselung mit vom Kunden bereitgestellten Schlüsseln) zu verwenden, geben Sie in der CreateMultipartUpload-Anforderung und in jeder nachfolgenden UploadPart-Anforderung die gleichen drei Verschlüsselungsschlüsselanforderungsheader an.

Betrieb	Implementierung
AbortMehrteilaUpload	Wird mit dem gesamten Amazon S3-REST-API-Verhalten implementiert. Änderungen vorbehalten.
CompleteMultipartUpload	Siehe "CompleteMultipartUpload"
CreateMultipartUpload (Zuvor mehrteiliges Hochladen initiieren)	Siehe "CreateMultipartUpload"
ListMultipartUploads	Siehe "ListMultipartUploads"
ListenTeile	Wird mit dem gesamten Amazon S3-REST-API-Verhalten implementiert. Änderungen vorbehalten.
UploadTeil	Siehe "UploadTeil"
UploadPartCopy	Siehe "UploadPartCopy"

CompleteMultipartUpload

Der CompleteMultipartUpload-Vorgang führt einen mehrteiligen Upload eines Objekts durch, indem die zuvor hochgeladenen Teile zusammengelegt werden.

> StorageGRID unterstützt nicht aufeinander folgende Werte in aufsteigender Reihenfolge für den `partNumber` Anforderungsparameter mit CompleteMultipartUpload. Der Parameter kann mit einem beliebigen Wert beginnen.

Konflikte lösen

Widersprüchliche Clientanforderungen, wie z. B. zwei Clients, die in denselben Schlüssel schreiben, werden auf der Grundlage der „neuesten Wins“ gelöst. Der Zeitpunkt für die Bewertung „neuester Erfolge“ basiert auf dem Zeitpunkt, an dem das StorageGRID System eine bestimmte Anforderung abgeschlossen hat und nicht auf dem Zeitpunkt, an dem S3-Clients einen Vorgang starten.

Unterstützte Anfrageheader

Die folgenden Anfragezeilen werden unterstützt:

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

  Der `x-amz-storage-class` Header wirkt sich darauf aus, wie viele Objektkopien StorageGRID erstellt, wenn die passende ILM-Regel den angibt"Doppelte Provisionierung oder ausgewogene Aufnahmeoption".

- `STANDARD`

  (Standard) gibt einen Dual-Commit-Aufnahmevorgang an, wenn die ILM-Regel die Option Dual Commit verwendet oder wenn die Option Balance auf das Erstellen von Zwischenkopien zurückgreift.

- `REDUCED_REDUNDANCY`

  Gibt einen Single-Commit-Aufnahmevorgang an, wenn die ILM-Regel die Option Dual Commit verwendet oder wenn die Option Balance zur Erstellung zwischenzeitaler Kopien zurückgreift.

  > Wenn Sie ein Objekt in einen Bucket mit aktivierter S3-Objektsperrung aufnehmen, wird die `REDUCED_REDUNDANCY` Option ignoriert. Wenn Sie ein Objekt in einen Legacy-konformen Bucket aufnehmen, gibt die `REDUCED_REDUNDANCY` Option einen Fehler zurück. StorageGRID führt immer eine doppelte Einspeisung durch, um Compliance-Anforderungen zu erfüllen.

> Wenn ein mehrtei. Upload nicht innerhalb von 15 Tagen abgeschlossen wird, wird der Vorgang als inaktiv markiert und alle zugehörigen Daten werden aus dem System gelöscht.

> Der `ETag` zurückgegebene Wert ist keine MD5-Summe der Daten, sondern folgt der Amazon S3-API-Implementierung des `ETag` Werts für mehrteilige Objekte.

Nicht unterstützte Anforderungsheader

Die folgenden Anforderungsheader werden nicht unterstützt:

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

Versionierung

Durch diesen Vorgang ist ein mehrtei. Upload abgeschlossen. Wenn die Versionierung für einen Bucket aktiviert ist, wird die Objektversion nach Abschluss des mehrteiligen Uploads erstellt.

Wenn die Versionierung für einen Bucket aktiviert ist, wird automatisch ein eindeutiges `versionId` Objekt für die Version des gespeicherten Objekts generiert. Dies `versionId` wird auch in der Antwort über den Antwortheader zurückgegeben `x-amz-version-id`.

Wenn die Versionierung unterbrochen wird, wird die Objektversion mit einer Null gespeichert `versionId` und wenn bereits eine Null-Version vorhanden ist, wird sie überschrieben.

> Wenn die Versionierung für einen Bucket aktiviert ist, erstellt das Abschließen eines mehrteiligen Uploads immer eine neue Version, selbst wenn mehrere Teile gleichzeitig auf denselben Objektschlüssel hochgeladen wurden. Wenn die Versionierung für einen Bucket nicht aktiviert ist, ist es möglich, einen mehrteiligen Upload zu initiieren und dann einen weiteren mehrteiligen Upload zu initiieren und zuerst auf demselben Objektschlüssel abzuschließen. In Buckets, die nicht versioniert sind, hat der mehrteilige Upload, der den letzten Teil abschließt, Vorrang.

Fehlgeschlagene Replikation, Benachrichtigung oder Metadatenbenachrichtigung

Wenn der Bucket, in dem der mehrteilige Upload stattfindet, für einen Plattformdienst konfiguriert ist, ist der mehrteilige Upload erfolgreich, auch wenn die zugehörige Replizierungs- oder Benachrichtigungsaktion fehlschlägt.

Ein Mandant kann die fehlgeschlagene Replizierung oder Benachrichtigung auslösen, indem die Metadaten oder Tags des Objekts aktualisiert werden. Ein Mieter kann die vorhandenen Werte erneut einreichen, um unerwünschte Änderungen zu vermeiden.

Siehe "Fehlerbehebung bei Plattform-Services".

CreateMultipartUpload

Der Vorgang CreateMultipartUpload (zuvor Multipart-Upload initiieren) initiiert einen mehrteiligen Upload für ein Objekt und gibt eine Upload-ID zurück.

Der `x-amz-storage-class` Anforderungskopf wird unterstützt. Der für eingereichte Wert `x-amz-storage-class` hat einen Einfluss darauf, wie StorageGRID Objektdaten bei der Aufnahme schützt, und nicht darauf, wie viele persistente Kopien des Objekts im StorageGRID System gespeichert werden (durch ILM bestimmt).

Wenn die ILM-Regel, die einem aufgenommenen Objekt entspricht"Aufnahme-Option", das Strict verwendet, hat der `x-amz-storage-class` Header keine Wirkung.

Folgende Werte können verwendet werden für `x-amz-storage-class`:

- `STANDARD` (Standard)
  - **Dual Commit**: Wenn die ILM-Regel die Option Dual Commit Ingest angibt, wird, sobald ein Objekt aufgenommen wird, eine zweite Kopie dieses Objekts erstellt und an einen anderen Storage Node verteilt (Dual Commit). Bei Bewertung des ILM bestimmt StorageGRID, ob diese ersten Zwischenkopien die Anweisungen zur Platzierung in der Regel erfüllen. Ist dies nicht der Fall, müssen möglicherweise neue Objektkopien an unterschiedlichen Standorten erstellt werden, und die ersten Zwischenkopien müssen eventuell gelöscht werden.
  - **Ausgeglichen**: Wenn die ILM-Regel die Option ausgeglichen angibt und StorageGRID nicht sofort alle in der Regel angegebenen Kopien erstellen kann, erstellt StorageGRID zwei Zwischenkopien auf verschiedenen Speicherknoten.

Wenn StorageGRID sofort alle in der ILM-Regel angegebenen Objektkopien erstellen kann (synchrone Platzierung), hat der `x-amz-storage-class` Header keine Auswirkungen.

- `REDUCED_REDUNDANCY`
  - **Dual Commit**: Wenn die ILM-Regel die Option Dual Commit angibt, erstellt StorageGRID bei Aufnahme des Objekts eine einzige Zwischenkopie (Single Commit).
  - **Ausgeglichen**: Wenn die ILM-Regel die Option ausgeglichen angibt, erstellt StorageGRID nur eine Zwischenkopie, wenn das System nicht sofort alle in der Regel angegebenen Kopien erstellen kann. Wenn StorageGRID eine synchrone Platzierung durchführen kann, hat diese Kopfzeile keine Auswirkung. Diese `REDUCED_REDUNDANCY` Option ist am besten geeignet, wenn die mit dem Objekt übereinstimmende ILM-Regel eine einzige replizierte Kopie erstellt. In diesem Fall `REDUCED_REDUNDANCY` entfällt bei jedem Einspielvorgang die unnötige Erstellung und Löschung einer zusätzlichen Objektkopie.

  Die Verwendung der `REDUCED_REDUNDANCY` Option wird in anderen Fällen nicht empfohlen. `REDUCED_REDUNDANCY` Erhöhtes Risiko von Objektdatenverlusten bei der Aufnahme. Beispielsweise können Sie Daten verlieren, wenn die einzelne Kopie zunächst auf einem Storage Node gespeichert wird, der ausfällt, bevor eine ILM-Evaluierung erfolgen kann.

> Da nur eine Kopie zu einem beliebigen Zeitpunkt repliziert werden kann, sind Daten einem ständigen Verlust ausgesetzt. Wenn nur eine replizierte Kopie eines Objekts vorhanden ist, geht dieses Objekt verloren, wenn ein Speicherknoten ausfällt oder einen beträchtlichen Fehler hat. Während Wartungsarbeiten wie Upgrades verlieren Sie auch vorübergehend den Zugriff auf das Objekt.

Die Angabe `REDUCED_REDUNDANCY` wirkt sich nur darauf aus, wie viele Kopien erstellt werden, wenn ein Objekt zum ersten Mal aufgenommen wird. Sie wirkt sich nicht darauf aus, wie viele Kopien des Objekts erstellt werden, wenn das Objekt durch die aktiven ILM-Richtlinien evaluiert wird, und führt nicht dazu, dass Daten mit niedrigerer Redundanz im StorageGRID System gespeichert werden.

> Wenn Sie ein Objekt in einen Bucket mit aktivierter S3-Objektsperrung aufnehmen, wird die `REDUCED_REDUNDANCY` Option ignoriert. Wenn Sie ein Objekt in einen Legacy-konformen Bucket aufnehmen, gibt die `REDUCED_REDUNDANCY` Option einen Fehler zurück. StorageGRID führt immer eine doppelte Einspeisung durch, um Compliance-Anforderungen zu erfüllen.

Unterstützte Anfrageheader

Die folgenden Anfragezeilen werden unterstützt:

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

  Derzeit wird nur der SHA256-Wert für `x-amz-checksum-algorithm` unterstützt.

- `x-amz-meta-`, Gefolgt von einem Name-Wert-Paar, das benutzerdefinierte Metadaten enthält

  Verwenden Sie bei der Angabe des Name-value-Paars für benutzerdefinierte Metadaten dieses allgemeine Format:

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

Wenn Sie die Option **Benutzerdefinierte Erstellungszeit** als Referenzzeit für eine ILM-Regel verwenden möchten, müssen Sie als Name der Metadaten verwenden, `creation-time` die beim Erstellen des Objekts aufgezeichnet werden. Beispiel:

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

Der Wert für `creation-time` wird seit dem 1. Januar 1970 als Sekunden ausgewertet.

> Das Hinzufügen `creation-time` als benutzerdefinierte Metadaten ist nicht zulässig, wenn Sie einem Bucket ein Objekt hinzufügen, für das ältere Compliance-Funktionen aktiviert sind. Ein Fehler wird zurückgegeben.

- S3-Objektsperrungs-Anfrageheader:
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    Wenn eine Anfrage ohne diese Header erstellt wird, werden die Bucket-Standardeinstellungen zur Aufbewahrung der Objektversion herangezogen, um die Aufbewahrung bis dato zu berechnen.

  "Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"

- SSE-Anfragezeilen:
  - `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`

    Anforderungsheader für serverseitige Verschlüsselung

> Informationen darüber, wie StorageGRID UTF-8-Zeichen verarbeitet, finden Sie unter "PutObject".

Anforderungsheader für serverseitige Verschlüsselung

Sie können die folgenden Anforderungsheader verwenden, um ein mehrteitiges Objekt mit serverseitiger Verschlüsselung zu verschlüsseln. Die Optionen SSE und SSE-C schließen sich gegenseitig aus.

- **SSE**: Verwenden Sie den folgenden Header in der CreateMultipartUpload-Anfrage, wenn Sie das Objekt mit einem eindeutigen Schlüssel verschlüsseln möchten, der von StorageGRID verwaltet wird. Geben Sie diesen Header in keiner der UploadPart-Anforderungen an.

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

- **SSE-C**: Verwenden Sie alle drei dieser Header in der CreateMultipartUpload-Anfrage (und in jeder nachfolgenden UploadPart-Anfrage), wenn Sie das Objekt mit einem eindeutigen Schlüssel verschlüsseln möchten, den Sie bereitstellen und verwalten.
  - `x-amz-server-side-encryption-customer-algorithm`: Spezifizieren `AES256`.
  - `x-amz-server-side-encryption-customer-key`: Geben Sie Ihren Verschlüsselungsschlüssel für das neue Objekt an.
  - `x-amz-server-side-encryption-customer-key-MD5`: Geben Sie den MD5-Digest des Verschlüsselungsschlüssels des neuen Objekts an.

> Die von Ihnen zur Verfügung gelegten Schlüssel werden niemals gespeichert. Wenn Sie einen Verschlüsselungsschlüssel verlieren, verlieren Sie das entsprechende Objekt. Bevor Sie vom Kunden bereitgestellte Schlüssel zum Schutz von Objektdaten verwenden, lesen Sie die Überlegungen für "Serverseitige Verschlüsselung".

Nicht unterstützte Anforderungsheader

Der folgende Anforderungskopf wird nicht unterstützt:

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

  Der `x-amz-website-redirect-location` Header gibt zurück `XNotImplemented`.

Versionierung

Mehrteilige Uploads bestehen aus separaten Vorgängen zum Initiieren des Uploads, Auflisten von Uploads, Hochladen von Teilen, Zusammenbauen der hochgeladenen Teile und Abschließen des Uploads. Objekte werden erstellt (und ggf. versioniert), wenn der CompleteMultipartUpload-Vorgang ausgeführt wird.

ListMultipartUploads

Der Vorgang ListMultipartUploads listet mehrteilige Uploads für einen Bucket auf, die gerade ausgeführt werden.

Die folgenden Anforderungsparameter werden unterstützt:

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

Versionierung

Mehrteilige Uploads bestehen aus separaten Vorgängen zum Initiieren des Uploads, Auflisten von Uploads, Hochladen von Teilen, Zusammenbauen der hochgeladenen Teile und Abschließen des Uploads. Objekte werden erstellt (und ggf. versioniert), wenn der CompleteMultipartUpload-Vorgang ausgeführt wird.

UploadTeil

Der Vorgang UploadPart lädt ein Teil in einem mehrteiligen Upload für ein Objekt hoch.

Unterstützte Anfrageheader

Die folgenden Anfragezeilen werden unterstützt:

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

Anforderungsheader für serverseitige Verschlüsselung

Wenn Sie die SSE-C-Verschlüsselung für die CreateMultipartUpload-Anforderung angegeben haben, müssen Sie auch die folgenden Anforderungsheader in jede UploadPart-Anforderung einschließen:

- `x-amz-server-side-encryption-customer-algorithm`: Spezifizieren `AES256`.
- `x-amz-server-side-encryption-customer-key`: Geben Sie den gleichen Verschlüsselungsschlüssel an, den Sie in der CreateMultipartUpload-Anforderung angegeben haben.
- `x-amz-server-side-encryption-customer-key-MD5`: Geben Sie den gleichen MD5-Digest an, den Sie in der CreateMultipartUpload-Anfrage angegeben haben.

> Die von Ihnen zur Verfügung gelegten Schlüssel werden niemals gespeichert. Wenn Sie einen Verschlüsselungsschlüssel verlieren, verlieren Sie das entsprechende Objekt. Bevor Sie vom Kunden bereitgestellte Schlüssel zum Schutz von Objektdaten verwenden, lesen Sie die Überlegungen in "Serverseitige Verschlüsselung".

Wenn Sie während der CreateMultipartUpload-Anforderung eine SHA-256-Prüfsumme angegeben haben, müssen Sie in jeder UploadPart-Anforderung auch den folgenden Anforderungsheader einfügen:

- `x-amz-checksum-sha256`: Geben Sie die SHA-256-Prüfsumme für diesen Teil an.

Nicht unterstützte Anforderungsheader

Die folgenden Anforderungsheader werden nicht unterstützt:

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

Versionierung

Mehrteilige Uploads bestehen aus separaten Vorgängen zum Initiieren des Uploads, Auflisten von Uploads, Hochladen von Teilen, Zusammenbauen der hochgeladenen Teile und Abschließen des Uploads. Objekte werden erstellt (und ggf. versioniert), wenn der CompleteMultipartUpload-Vorgang ausgeführt wird.

UploadPartCopy

Der Vorgang UploadPartCopy lädt einen Teil eines Objekts hoch, indem Daten aus einem vorhandenen Objekt als Datenquelle kopiert werden.

Der UploadPartCopy-Vorgang wird mit dem gesamten Amazon S3-REST-API-Verhalten implementiert. Änderungen vorbehalten.

Diese Anforderung liest und schreibt die Objektdaten, die in im StorageGRID-System angegeben `x-amz-copy-source-range` sind.

Die folgenden Anfragezeilen werden unterstützt:

- `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`

Anforderungsheader für serverseitige Verschlüsselung

Wenn Sie die SSE-C-Verschlüsselung für die CreateMultipartUpload-Anforderung angegeben haben, müssen Sie auch die folgenden Anforderungsheader in jede UploadPartCopy-Anforderung einschließen:

- `x-amz-server-side-encryption-customer-algorithm`: Spezifizieren `AES256`.
- `x-amz-server-side-encryption-customer-key`: Geben Sie den gleichen Verschlüsselungsschlüssel an, den Sie in der CreateMultipartUpload-Anforderung angegeben haben.
- `x-amz-server-side-encryption-customer-key-MD5`: Geben Sie den gleichen MD5-Digest an, den Sie in der CreateMultipartUpload-Anfrage angegeben haben.

Wenn das Quellobjekt mit einem vom Kunden bereitgestellten Schlüssel (SSE-C) verschlüsselt wird, müssen Sie die folgenden drei Header in die Anforderung UploadPartCopy einbeziehen, damit das Objekt entschlüsselt und dann kopiert werden kann:

- `x-amz-copy-source-server-side-encryption-customer-algorithm`: Spezifizieren `AES256`.
- `x-amz-copy-source-server-side-encryption-customer-key`: Geben Sie den Verschlüsselungsschlüssel an, den Sie beim Erstellen des Quellobjekts angegeben haben.
- `x-amz-copy-source-server-side-encryption-customer-key-MD5`: Geben Sie den MD5-Digest an, den Sie beim Erstellen des Quellobjekts angegeben haben.

> Die von Ihnen zur Verfügung gelegten Schlüssel werden niemals gespeichert. Wenn Sie einen Verschlüsselungsschlüssel verlieren, verlieren Sie das entsprechende Objekt. Bevor Sie vom Kunden bereitgestellte Schlüssel zum Schutz von Objektdaten verwenden, lesen Sie die Überlegungen in "Serverseitige Verschlüsselung".

Versionierung

Mehrteilige Uploads bestehen aus separaten Vorgängen zum Initiieren des Uploads, Auflisten von Uploads, Hochladen von Teilen, Zusammenbauen der hochgeladenen Teile und Abschließen des Uploads. Objekte werden erstellt (und ggf. versioniert), wenn der CompleteMultipartUpload-Vorgang ausgeführt wird.

Fehlerantworten

Das StorageGRID System unterstützt alle zutreffenden S3-REST-API-Standardfehlerantworten. Darüber hinaus fügt die StorageGRID Implementierung mehrere individuelle Antworten hinzu.

Unterstützte S3-API-Fehlercodes

Name	HTTP-Status
AccessDenied	403 Verbotene
BadDigest	400 Fehlerhafte Anfrage
BucketAlreadyExists	409 Konflikt
BucketNotEmpty	409 Konflikt
IncompleteBody	400 Fehlerhafte Anfrage
Interner Fehler	500 Fehler Des Internen Servers
InvalidAccessKey ID	403 Verbotene
InvalidArgument	400 Fehlerhafte Anfrage
InvalidBucketName	400 Fehlerhafte Anfrage
InvalidBucketState	409 Konflikt
InvalidDigest	400 Fehlerhafte Anfrage
InvalidVerschlüsselungAlgorithmFehler	400 Fehlerhafte Anfrage
InvalidTeil	400 Fehlerhafte Anfrage
InvalidPartOrder	400 Fehlerhafte Anfrage
InvalidRange	416 Angeforderter Bereich Nicht Zu Unterprüfbar
InvalidRequest	400 Fehlerhafte Anfrage
InvalidStorageClass	400 Fehlerhafte Anfrage
InvalidTag	400 Fehlerhafte Anfrage

Name	HTTP-Status
InvalidURI	400 Fehlerhafte Anfrage
KeyTooLong	400 Fehlerhafte Anfrage
MalformedXML	400 Fehlerhafte Anfrage
MetadataTooLarge	400 Fehlerhafte Anfrage
MethodenAlled	405 Methode Nicht Zulässig
MissingContentLänge	411 Länge Erforderlich
MissingRequestBodyError	400 Fehlerhafte Anfrage
MissingSecurityHeader	400 Fehlerhafte Anfrage
NoSuchBucket	404 Nicht Gefunden
NoSuchKey	404 Nicht Gefunden
NoSuchUpload	404 Nicht Gefunden
NotImplemsted	501 Nicht Implementiert
NoSuchBucketRichtlinien	404 Nicht Gefunden
ObjektLockKonfigurationNotgefundenFehler	404 Nicht Gefunden
Vorbedingungen nicht möglich	412 Voraussetzung Fehlgeschlagen
AnforderungTimeTooSkewed	403 Verbotene
Servicenicht verfügbar	503 Service Nicht Verfügbar
SignalDoesNotMatch	403 Verbotene
TooManyDickets	400 Fehlerhafte Anfrage
UserKeyMustBespezifiziert	400 Fehlerhafte Anfrage

Benutzerdefinierte StorageGRID-Fehlercodes

Name	Beschreibung	HTTP-Status
XBucketLifecycleNotAlled	In einem zuvor konformen Bucket ist die Konfiguration des Bucket-Lebenszyklus nicht zulässig	400 Fehlerhafte Anfrage
XBucketPolicyParseException	Fehler beim Parsen der JSON der empfangenen Bucket-Richtlinie.	400 Fehlerhafte Anfrage
XComplianceKonflict	Vorgang aufgrund von Compliance-Einstellungen abgelehnt.	403 Verbotene
XComplianceReducedRAID-RedundanzVerbotenen	Reduzierte Redundanz ist in einem älteren, konformen Bucket nicht zulässig	400 Fehlerhafte Anfrage
XMaxBucketPolicyLengthexceed	Ihre Richtlinie überschreitet die maximal zulässige Länge der Bucket-Richtlinie.	400 Fehlerhafte Anfrage
XMissingInternRequestHeader	Eine Kopfzeile einer internen Anforderung fehlt.	400 Fehlerhafte Anfrage
XNoSuchBucketCompliance	Für den angegebenen Bucket ist die veraltete Compliance nicht aktiviert.	404 Nicht Gefunden
XNotAcceptable	Die Anforderung enthält mindestens einen Übernehmen-Header, der nicht erfüllt werden konnte.	406 Nicht Akzeptabel
XNotImplemsted	Die von Ihnen gestellte Anfrage beinhaltet Funktionen, die nicht implementiert sind.	501 Nicht Implementiert

Benutzerdefinierte Operationen von StorageGRID

Benutzerdefinierte Operationen von StorageGRID

Das StorageGRID System unterstützt benutzerdefinierte Vorgänge, die zur S3-REST-API hinzugefügt werden.

In der folgenden Tabelle sind die von StorageGRID unterstützten benutzerdefinierten Vorgänge aufgeführt.

Betrieb	Beschreibung
"Get Bucket-Konsistenz"	Gibt die Konsistenz zurück, die auf einen bestimmten Bucket angewendet wird.
"PUT Bucket-Konsistenz"	Legt die Konsistenz fest, die auf einen bestimmten Bucket angewendet wird.

Betrieb	Beschreibung
"ZEITPUNKT des letzten Zugriffs FÜR den Bucket ABRUFEN"	Gibt an, ob Updates der letzten Zugriffszeit für einen bestimmten Bucket aktiviert oder deaktiviert wurden.
"PUT Bucket-Zeit für den letzten Zugriff"	Hiermit können Sie Updates der letzten Zugriffszeit für einen bestimmten Bucket aktivieren oder deaktivieren.
"Konfiguration für die Benachrichtigung über Bucket-Metadaten LÖSCHEN"	Löscht die XML-Konfiguration für die Metadatenbenachrichtigung, die mit einem bestimmten Bucket verknüpft ist.
"Konfiguration der Bucket-Metadaten-Benachrichtigungen ABRUFEN"	Gibt die XML-XML-Benachrichtigungskonfiguration für Metadaten zurück, die einem bestimmten Bucket zugeordnet ist.
"PUT Bucket-Metadaten-Benachrichtigungskonfiguration"	Konfiguriert den Metadaten-Benachrichtigungsdienst für einen Bucket
"GET Storage-Auslastung"	Gibt an, wie viel Speicherplatz von einem Konto und für jeden mit dem Konto verknüpften Bucket insgesamt verwendet wird.
"Veraltet: CreateBucket mit Compliance-Einstellungen"	Veraltet und nicht unterstützt: Sie können keine neuen Buckets mit aktivierter Compliance mehr erstellen.
"Veraltet: EINHALTUNG von Bucket ABRUFEN"	Veraltet, aber unterstützt: Gibt die Compliance-Einstellungen zurück, die derzeit für einen vorhandenen Legacy-konformen Bucket wirksam sind.
"Veraltet: EINHALTUNG VON PUT Bucket"	Veraltet, aber unterstützt: Ermöglicht es Ihnen, die Compliance-Einstellungen für einen vorhandenen, älteren konformen Bucket zu ändern.

Get Bucket-Konsistenz

Mit der Konsistenzanforderung für GET Bucket können Sie die Konsistenz bestimmen, die auf einen bestimmten Bucket angewendet wird.

Die Standardkonsistenz ist so festgelegt, dass „Read-after-write“ für neu erstellte Objekte garantiert wird.

Sie müssen über die berechtigung s3:GetBucketConsistency verfügen oder als Account root vorliegen, um diesen Vorgang abzuschließen.

Anforderungsbeispiel

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

Antwort

In der XML-Antwort `<Consistency>` gibt einen der folgenden Werte zurück:

Konsistenz	Beschreibung
Alle	Alle Nodes erhalten die Daten sofort, sonst schlägt die Anfrage fehl.
Stark global	Garantierte Konsistenz bei Lese-nach-Schreibvorgängen für alle Client-Anfragen an allen Standorten.
Stark vor Ort	Garantiert Konsistenz bei Lese-nach-Schreibvorgängen für alle Client-Anfragen innerhalb eines Standorts.
Read-after-New-Write-Funktion	(Standard) konsistente Lese-/Schreibvorgänge für neue Objekte und eventuelle Konsistenz bei Objekt-Updates. Hochverfügbarkeit und garantierte Datensicherung Empfohlen für die meisten Fälle.
Verfügbar	Bietet eventuelle Konsistenz für neue Objekte und Objektaktualisierungen. Verwenden Sie für S3-Buckets nur nach Bedarf (z. B. für einen Bucket mit Protokollwerten, die nur selten gelesen werden, oder für HEAD- oder GET-Vorgänge für nicht vorhandene Schlüssel). Nicht unterstützt für S3 FabricPool-Buckets.

Antwortbeispiel

```
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>
```

Verwandte Informationen

"Konsistenzwerte"

PUT Bucket-Konsistenz

Mit der Konsistenzanforderung für PUT-Bucket können Sie die Konsistenz angeben, die auf Vorgänge angewendet werden soll, die auf einen Bucket ausgeführt wurden.

Die Standardkonsistenz ist so festgelegt, dass „Read-after-write“ für neu erstellte Objekte garantiert wird.

Bevor Sie beginnen

Sie müssen über die berechtigung s3:PutBucketConsistency verfügen oder als Account root vorliegen, um diesen Vorgang abzuschließen.

Anfrage

Der `x-ntap-sg-consistency` Parameter muss einen der folgenden Werte enthalten:

Konsistenz	Beschreibung
Alle	Alle Nodes erhalten die Daten sofort, sonst schlägt die Anfrage fehl.
Stark global	Garantierte Konsistenz bei Lese-nach-Schreibvorgängen für alle Client-Anfragen an allen Standorten.
Stark vor Ort	Garantiert Konsistenz bei Lese-nach-Schreibvorgängen für alle Client-Anfragen innerhalb eines Standorts.
Read-after-New-Write-Funktion	(Standard) konsistente Lese-/Schreibvorgänge für neue Objekte und eventuelle Konsistenz bei Objekt-Updates. Hochverfügbarkeit und garantierte Datensicherung Empfohlen für die meisten Fälle.
Verfügbar	Bietet eventuelle Konsistenz für neue Objekte und Objektaktualisierungen. Verwenden Sie für S3-Buckets nur nach Bedarf (z. B. für einen Bucket mit Protokollwerten, die nur selten gelesen werden, oder für HEAD- oder GET-Vorgänge für nicht vorhandene Schlüssel). Nicht unterstützt für S3 FabricPool-Buckets.

Anmerkung: im Allgemeinen sollten Sie die "Read-after-New-write" Konsistenz verwenden. Wenn die Anforderungen nicht korrekt funktionieren, ändern Sie das Client-Verhalten der Anwendung, wenn möglich. Oder konfigurieren Sie den Client so, dass die Konsistenz für jede API-Anforderung angegeben wird. Legen Sie die Konsistenz auf Bucket-Ebene nur als letzte Option fest.

Anforderungsbeispiel

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

Verwandte Informationen

"Konsistenzwerte"

ZEITPUNKT des letzten Zugriffs FÜR den Bucket ABRUFEN

In der Anforderung „letzte Bucket-Zugriffszeit“ KÖNNEN Sie festlegen, ob Updates der letzten Zugriffszeit für einzelne Buckets aktiviert oder deaktiviert sind.

Sie müssen über die berechtigung s3:GetBucketLastAccessTime verfügen oder als Kontostamm vorliegen, um diesen Vorgang abzuschließen.

Anforderungsbeispiel

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

Antwortbeispiel

Dieses Beispiel zeigt, dass Updates der letzten Zugriffszeit für den Bucket aktiviert sind.

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

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

PUT Bucket-Zeit für den letzten Zugriff

In der ANFORDERUNG PUT Bucket Last Access Time können Sie Updates der letzten Zugriffszeit für einzelne Buckets aktivieren oder deaktivieren. Durch das Deaktivieren von Updates der letzten Zugriffszeit wird die Performance verbessert. Dies ist die Standardeinstellung für alle Buckets, die mit Version 10.3 oder höher erstellt wurden.

Sie müssen über die s3:PutBucketLastAccessTime-Berechtigung für einen Bucket verfügen oder als Account-Root dienen, um diesen Vorgang abzuschließen.

Ab StorageGRID Version 10.3 sind Updates der letzten Zugriffszeit für alle neuen Buckets standardmäßig deaktiviert. Wenn Sie Buckets haben, die mit einer früheren Version von StorageGRID erstellt wurden und denen das neue Standardverhalten entsprechen möchten, müssen Sie für jeden dieser früheren Buckets explizit die Updates der letzten Zugriffszeit deaktivieren. Sie können Updates für die letzte Zugriffszeit mithilfe der Anforderung zum Zeitpunkt des letzten Zugriffs für Bucket oder über die Detailseite für einen Bucket im Tenant Manager aktivieren oder deaktivieren. Siehe "Aktiviert bzw. deaktiviert Updates der letzten Zugriffszeit".

Wenn Updates der letzten Zugriffszeit für einen Bucket deaktiviert wurden, wird das folgende Verhalten auf die Vorgänge auf dem Bucket angewendet:

- GetObject-, GetObjectAcl-, GetObjectTagging- und HeadObject-Anforderungen aktualisieren nicht die letzte Zugriffszeit. Das Objekt wird zur Bewertung des Information Lifecycle Management (ILM) nicht zu Warteschlangen hinzugefügt.
- CopyObject- und PutObjectTagging-Anfragen, die nur die Metadaten aktualisieren, aktualisieren ebenfalls die letzte Zugriffszeit. Das Objekt wird Warteschlangen für die ILM-Bewertung hinzugefügt.
- Wenn Updates zur letzten Zugriffszeit für den Quell-Bucket deaktiviert sind, aktualisieren CopyObject-Anforderungen nicht die letzte Zugriffszeit für den Quell-Bucket. Das kopierte Objekt wird nicht zu Warteschlangen für die ILM-Bewertung für den Quell-Bucket hinzugefügt. CopyObject-Anforderungen aktualisieren jedoch immer die letzte Zugriffszeit für das Ziel. Die Kopie des Objekts wird zu Warteschlangen für eine ILM-Bewertung hinzugefügt.
- CompleteMultipartUpload-Anforderungen werden zum Zeitpunkt des letzten Zugriffs aktualisiert. Das fertiggestellte Objekt wird zur ILM-Bewertung zu Warteschlangen hinzugefügt.

Beispiele anfordern

Dieses Beispiel ermöglicht die Zeit des letzten Zugriffs für einen Bucket.

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

Dieses Beispiel deaktiviert die Zeit des letzten Zugriffs für einen Bucket.

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

Konfiguration für die Benachrichtigung über Bucket-Metadaten LÖSCHEN

Mit der Konfigurationsanforderung FÜR DIE BENACHRICHTIGUNG „BUCKET-Metadaten LÖSCHEN“ können Sie den Suchintegrationsdienst für einzelne Buckets deaktivieren, indem Sie die Konfigurations-XML löschen.

Sie müssen über die berechtigung s3:DeleteBucketMetadataNotification für einen Bucket verfügen oder als Account-Root dienen, um diesen Vorgang abzuschließen.

Anforderungsbeispiel

Dieses Beispiel zeigt die Deaktivierung des Suchintegrationsservice für einen Bucket.

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

Konfiguration der Bucket-Metadaten-Benachrichtigungen ABRUFEN

Die Konfigurationsanforderung FÜR GET Bucket-Metadaten-Benachrichtigungen ermöglicht es Ihnen, die Konfigurations-XML abzurufen, die zur Konfiguration der Suchintegration für einzelne Buckets verwendet wird.

Sie müssen über die berechtigung s3:GetBucketMetadataNotification verfügen oder als Kontowurzel dienen, um diesen Vorgang abzuschließen.

Anforderungsbeispiel

Diese Anforderung ruft die Metadaten-Benachrichtigungskonfiguration für den Bucket namens `bucket`ab.

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

Antwort

Der Response Body umfasst die Konfiguration der Metadaten-Benachrichtigung für den Bucket. Anhand der Konfiguration der Metadatenbenachrichtigung können Sie festlegen, wie der Bucket für die Suchintegration konfiguriert ist. So können Unternehmen ermitteln, welche Objekte indiziert sind und an welche Endpunkte ihre Objektmetadaten gesendet werden.

```
<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>
```

Jede Konfiguration für die Metadatenbenachrichtigung enthält mindestens ein Regeln. Jede Regel gibt die Objekte an, die auf sie angewendet werden, und das Ziel, an dem StorageGRID Objekt-Metadaten senden soll. Ziele müssen mit dem URN eines StorageGRID-Endpunkts angegeben werden.

Name	Beschreibung	Erforderlich
MetadataNotificationKonfiguration	Container-Tag für Regeln zur Angabe von Objekten und Zielen für Metadatenbenachrichtigungen Enthält mindestens ein Regelelement.	Ja.
Regel	Container-Tag für eine Regel, die die Objekte identifiziert, deren Metadaten zu einem bestimmten Index hinzugefügt werden sollen. Regeln mit überlappenden Präfixen werden abgelehnt. Im MetadataNotificationConfiguration Element enthalten.	Ja.
ID	Eindeutige Kennung für die Regel. In das Element Regel aufgenommen.	Nein
Status	Der Status kann „aktiviert“ oder „deaktiviert“ sein. Für deaktivierte Regeln wird keine Aktion durchgeführt. In das Element Regel aufgenommen.	Ja.

Name	Beschreibung	Erforderlich
Präfix	Objekte, die mit dem Präfix übereinstimmen, werden von der Regel beeinflusst und ihre Metadaten werden an das angegebene Ziel gesendet. Geben Sie ein leeres Präfix an, um alle Objekte zu entsprechen. In das Element Regel aufgenommen.	Ja.
Ziel	Container-Tag für das Ziel einer Regel. In das Element Regel aufgenommen.	Ja.
Urne	URNE des Ziels, an dem Objektmetadaten gesendet werden. Muss der URN eines StorageGRID-Endpunkts mit den folgenden Eigenschaften sein: • `es` Muss das dritte Element sein. • Die URN muss mit dem Index und dem Typ enden, in dem die Metadaten gespeichert sind, in der Form `domain-name/myindex/mytype`. Endpunkte werden mithilfe der Mandanten-Manager oder der Mandanten-Management-API konfiguriert. Sie nehmen folgende Form: • `arn:aws:es:_region:account-ID_:domain/mydomain/myindex/mytype` • `urn:mysite:es:::mydomain/myindex/mytype` Der Endpunkt muss konfiguriert werden, bevor die Konfigurations-XML gesendet wird, oder die Konfiguration schlägt mit einem Fehler 404 fehl. Urne ist im Element Ziel enthalten.	Ja.

Antwortbeispiel

Die XML-Datei zwischen den `<MetadataNotificationConfiguration></MetadataNotificationConfiguration>` Tags zeigt, wie die Integration mit einem Endpunkt für die Suchintegration für den Bucket konfiguriert ist. In diesem Beispiel werden Objektmetadaten an einen Elasticsearch-Index mit dem Namen und dem Typ `2017` gesendet `current`, der in einer AWS-Domäne mit dem Namen gehostet wird `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>
```

Verwandte Informationen

"Verwenden Sie ein Mandantenkonto"

PUT Bucket-Metadaten-Benachrichtigungskonfiguration

Die Konfigurationsanforderung FÜR PUT Bucket-Metadaten-Benachrichtigungen ermöglicht es Ihnen, den Such-Integrationsservice für einzelne Buckets zu aktivieren. Die XML-XML-Konfiguration für die Metadatenbenachrichtigung, die Sie im Anforderungsindex angeben, gibt die Objekte an, deren Metadaten an den Zielsuchindex gesendet werden.

Sie müssen über die berechtigung s3:PutBucketMetadataNotification für einen Bucket verfügen oder als Account-Root dienen, um diesen Vorgang abzuschließen.

Anfrage

Die Anforderung muss die Konfiguration der Metadatenbenachrichtigung in der Anfraentext enthalten. Jede Konfiguration für die Metadatenbenachrichtigung enthält mindestens ein Regeln. Jede Regel gibt die Objekte an, auf die sie angewendet wird, und das Ziel, an dem StorageGRID Metadaten senden soll.

Objekte können nach dem Präfix des Objektnamens gefiltert werden. Beispielsweise können Sie Metadaten für Objekte mit dem Präfix an ein Ziel und Objekte mit dem `/videos` Präfix an ein anderes senden `/images`.

Konfigurationen mit überlappenden Präfixen sind nicht gültig und werden beim Einreichen abgelehnt. Eine Konfiguration, die beispielsweise eine Regel für Objekte mit dem Präfix und eine zweite Regel für Objekte mit dem `test2` Präfix enthält `test`, ist nicht zulässig.

Ziele müssen mit dem URN eines StorageGRID-Endpunkts angegeben werden. Der Endpunkt muss

vorhanden sein, wenn die Metadaten-Benachrichtigungskonfiguration übermittelt wird, oder die Anforderung schlägt als fehl `400 Bad Request`. Die Fehlermeldung lautet: `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>
```

In der Tabelle werden die Elemente in der XML-Konfiguration für die Metadatenbenachrichtigung beschrieben.

Name	Beschreibung	Erforderlich
MetadataNotificationKonfiguration	Container-Tag für Regeln zur Angabe von Objekten und Zielen für Metadatenbenachrichtigungen Enthält mindestens ein Regelelement.	Ja.
Regel	Container-Tag für eine Regel, die die Objekte identifiziert, deren Metadaten zu einem bestimmten Index hinzugefügt werden sollen. Regeln mit überlappenden Präfixen werden abgelehnt. Im MetadataNotificationConfiguration Element enthalten.	Ja.
ID	Eindeutige Kennung für die Regel. In das Element Regel aufgenommen.	Nein
Status	Der Status kann „aktiviert“ oder „deaktiviert“ sein. Für deaktivierte Regeln wird keine Aktion durchgeführt. In das Element Regel aufgenommen.	Ja.

Name	Beschreibung	Erforderlich
Präfix	Objekte, die mit dem Präfix übereinstimmen, werden von der Regel beeinflusst und ihre Metadaten werden an das angegebene Ziel gesendet. Geben Sie ein leeres Präfix an, um alle Objekte zu entsprechen. In das Element Regel aufgenommen.	Ja.
Ziel	Container-Tag für das Ziel einer Regel. In das Element Regel aufgenommen.	Ja.
Urne	URNE des Ziels, an dem Objektmetadaten gesendet werden. Muss der URN eines StorageGRID-Endpunkts mit den folgenden Eigenschaften sein: • `es` Muss das dritte Element sein. • Die URN muss mit dem Index und dem Typ enden, in dem die Metadaten gespeichert sind, in der Form `domain-name/myindex/mytype`. Endpunkte werden mithilfe der Mandanten-Manager oder der Mandanten-Management-API konfiguriert. Sie nehmen folgende Form: • `arn:aws:es:region:account-ID:domain/mydomain/myindex/mytype` • `urn:mysite:es:::mydomain/myindex/mytype` Der Endpunkt muss konfiguriert werden, bevor die Konfigurations-XML gesendet wird, oder die Konfiguration schlägt mit einem Fehler 404 fehl. Urne ist im Element Ziel enthalten.	Ja.

Beispiele anfordern

Dieses Beispiel zeigt die Aktivierung der Integration von Suchvorgängen für einen Bucket. In diesem Beispiel werden die Objektmetadaten für alle Objekte an dasselbe Ziel gesendet.

```
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>
```

In diesem Beispiel werden Objektmetadaten für Objekte mit dem Präfix `/images` an ein Ziel gesendet, während Objektmetadaten für Objekte mit dem Präfix `/videos` an ein zweites Ziel gesendet werden.

```
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>
```

Vom Suchintegrations-Service generierter JSON

Wenn Sie den Such-Integrationsservice für einen Bucket aktivieren, wird ein JSON-Dokument generiert und an den Zielendpunkt gesendet, wenn Metadaten oder Tags hinzugefügt, aktualisiert oder gelöscht werden.

Dieses Beispiel zeigt ein Beispiel für den JSON, der generiert werden könnte, wenn ein Objekt mit dem Schlüssel in einem Bucket mit `SGWS/Tagging.txt` dem Namen erstellt wird `test`. Der `test` Bucket ist nicht versioniert, daher ist das `versionId` Tag leer.

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

Objektmetadaten sind in Metadaten-Benachrichtigungen enthalten

In der Tabelle sind alle Felder aufgeführt, die im JSON-Dokument enthalten sind, die beim Aktivierung der Suchintegration an den Zielendpunkt gesendet werden.

Der Dokumentname umfasst, falls vorhanden, den Bucket-Namen, den Objektnamen und die Version-ID.

Typ	Elementname	Beschreibung
Bucket- und Objektinformationen	Eimer	Name des Buckets
Bucket- und Objektinformationen	Taste	Name des Objektschlüssels
Bucket- und Objektinformationen	VersionID	Objektversion für Objekte in versionierten Buckets
Bucket- und Objektinformationen	Werden	Beispiel: Bucket-Region `us-east-1`
System-Metadaten	Größe	Objektgröße (in Byte) wie für einen HTTP-Client sichtbar

Typ	Elementname	Beschreibung
System-Metadaten	md5	Objekt-Hash
Benutzer-Metadaten	Metadaten *key:value*	Alle Benutzer-Metadaten des Objekts als Schlüssel-Wert-Paare
Tags	Tags *key:value*	Alle für das Objekt definierten Objekt-Tags als Schlüsselwert-Paare

Für Tags und Benutzer-Metadaten gibt StorageGRID Daten und Nummern an Elasticsearch als Strings oder als S3-Ereignisbenachrichtigungen weiter. Um Elasticsearch so zu konfigurieren, dass diese Strings als Daten oder Zahlen interpretiert werden, befolgen Sie die Elasticsearch-Anweisungen für die dynamische Feldzuordnung und die Zuordnung von Datumsformaten. Sie müssen die dynamischen Feldzuordnungen im Index aktivieren, bevor Sie den Suchintegrationsdienst konfigurieren. Nachdem ein Dokument indiziert wurde, können Sie die Feldtypen des Dokuments im Index nicht mehr bearbeiten.

Verwandte Informationen

"Verwenden Sie ein Mandantenkonto"

Storage-Nutzungsanforderung ABRUFEN

Der Antrag ZUR GET Storage-Nutzung gibt Ihnen die Gesamtzahl des verwendeten Storage durch ein Konto und für jeden mit dem Account verknüpften Bucket an.

Die Menge des von einem Konto und seinen Buckets verwendeten Speichers kann durch eine modifizierte ListBuckets-Anforderung mit dem Abfrageparameter abgerufen `x-ntap-sg-usage` werden. Die Nutzung des Bucket-Storage wird getrennt von DEN PUT- und LÖSCHANFRAGEN, die vom System verarbeitet werden, nachverfolgt. Es kann zu einer gewissen Verzögerung kommen, bevor die Nutzungswerte auf der Grundlage der Verarbeitung von Anfragen den erwarteten Werten entsprechen, insbesondere wenn das System unter hoher Belastung steht.

StorageGRID versucht standardmäßig, Nutzungsdaten mithilfe einer starken globalen Konsistenz abzurufen. Wenn eine starke globale Konsistenz nicht erreicht werden kann, versucht StorageGRID, die Verwendungsinformationen in einer starken Site-Konsistenz abzurufen.

Sie müssen über die s3:ListAllMyBuchets-Berechtigung verfügen oder als Kontostamm vorliegen, um diese Operation abzuschließen.

Anforderungsbeispiel

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

Antwortbeispiel

Dieses Beispiel zeigt ein Konto, das vier Objekte und 12 Bytes Daten in zwei Buckets enthält. Jeder Bucket enthält zwei Objekte und sechs Bytes Daten.

```
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>
```

Versionierung

Jede gespeicherte Objektversion trägt zu den `ObjectCount` Werten und `DataBytes` in der Antwort bei. Löschmarkierungen werden nicht zur Gesamtmenge hinzugefügt `ObjectCount`.

Verwandte Informationen

"Konsistenzwerte"

Veraltete Bucket-Anforderungen für ältere Compliance

Veraltete Bucket-Anforderungen für ältere Compliance

Möglicherweise müssen Sie die StorageGRID S3 REST-API zum Management von Buckets verwenden, die mit der älteren Compliance-Funktion erstellt wurden.

Compliance-Funktion veraltet

Die in früheren StorageGRID-Versionen verfügbare Funktion für die StorageGRID-Konformität ist veraltet und wurde durch S3-Objektsperre ersetzt.

Wenn Sie zuvor die Einstellung für globale Konformität aktiviert haben, ist die globale S3-Objektsperre in StorageGRID 11.6 aktiviert. Neue Buckets können nicht mehr mit aktivierter Compliance erstellt werden. Trotzdem können Sie bei Bedarf die StorageGRID S3 REST-API verwenden, um alle vorhandenen, älteren, konformen Buckets zu managen.

- "Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"
- "Objektmanagement mit ILM"
- "NetApp Knowledge Base: Management älterer, konformer Buckets für StorageGRID 11.5"

Veraltete Compliance-Anforderungen:

- "Veraltet – PUT Bucket-Anforderung-Änderungen aus Compliance-Gründen"

  Das SGCompliance XML-Element ist veraltet. Zuvor könnten Sie dieses benutzerdefinierte StorageGRID-Element in das optionale XML-Anforderungsgremium VON PUT Bucket-Anforderungen integrieren, um einen konformen Bucket zu erstellen.

- "Veraltet – BUCKET-Compliance ABRUFEN"

  Die ANFORDERUNG „GET Bucket-Compliance“ ist veraltet. Sie können diese Anforderung jedoch weiterhin verwenden, um die derzeit für einen vorhandenen, älteren, konformen Bucket geltenden Compliance-Einstellungen zu bestimmen.

- "Veraltet – EINHALTUNG VON PUT Bucket"

  Die ANFORDERUNG „PUT Bucket-Compliance“ ist veraltet. Sie können diese Anforderung jedoch weiterhin verwenden, um die Compliance-Einstellungen für einen vorhandenen Bucket zu ändern, der die Compliance-Anforderungen erfüllt. Sie können beispielsweise einen vorhandenen Bucket auf „Legal Hold“ platzieren oder den Aufbewahrungszeitraum erhöhen.

Veraltet: CreateBucket fordert Änderungen für Compliance an

Das SGCompliance XML-Element ist veraltet. Zuvor könnten Sie dieses benutzerdefinierte StorageGRID-Element in den optionalen XML-Anforderungskörper von CreateBucket-Anforderungen aufnehmen, um einen konformen Bucket zu erstellen.

> Die in früheren StorageGRID-Versionen verfügbare Funktion für die StorageGRID-Konformität ist veraltet und wurde durch S3-Objektsperre ersetzt. Im Folgenden finden Sie weitere Informationen:
>
> - "Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"
> - "NetApp Knowledge Base: Management älterer, konformer Buckets für StorageGRID 11.5"

Mit aktivierter Compliance können keine neuen Buckets mehr erstellt werden. Die folgende Fehlermeldung wird zurückgegeben, wenn Sie versuchen, die Änderungen der CreateBucket-Anforderung für die Compliance zu verwenden, um einen neuen konformen Bucket zu erstellen:

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

Veraltet: Anforderung FÜR Bucket-Compliance ABRUFEN

Die ANFORDERUNG „GET Bucket-Compliance“ ist veraltet. Sie können diese Anforderung jedoch weiterhin verwenden, um die derzeit für einen vorhandenen, älteren, konformen Bucket geltenden Compliance-Einstellungen zu bestimmen.

> Die in früheren StorageGRID-Versionen verfügbare Funktion für die StorageGRID-Konformität ist veraltet und wurde durch S3-Objektsperre ersetzt. Im Folgenden finden Sie weitere Informationen:
>
> - "Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"
> - "NetApp Knowledge Base: Management älterer, konformer Buckets für StorageGRID 11.5"

Um diesen Vorgang abzuschließen, müssen Sie über die berechtigung s3:GetBucketCompliance verfügen oder als Stammverzeichnis für das Konto verfügen.

Anforderungsbeispiel

Mit dieser Beispielanforderung können Sie die Compliance-Einstellungen für den Bucket mit dem Namen festlegen `mybucket`.

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

Antwortbeispiel

In der XML-Antwort `<SGCompliance>` werden die für den Bucket verwendeten Compliance-Einstellungen aufgeführt. Dieses Beispiel zeigt die Compliance-Einstellungen für einen Bucket, in dem jedes Objekt ein Jahr lang (525,600 Minuten) aufbewahrt wird, beginnend mit der Aufnahme des Objekts in das Grid. Derzeit ist keine gesetzliche Aufbewahrungspflichten auf diesem Bucket vorhanden. Jedes Objekt wird nach einem Jahr automatisch gelöscht.

```
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	Beschreibung
WiederholungPeriodMinuten	Die Länge des Aufbewahrungszeitraums für Objekte, die diesem Bucket hinzugefügt wurden, in Minuten Der Aufbewahrungszeitraum beginnt, wenn das Objekt in das Raster aufgenommen wird.
LegalAlte	• Wahr: Dieser Bucket befindet sich derzeit in einer gesetzlichen Aufbewahrungspflichten. Objekte in diesem Bucket können erst gelöscht werden, wenn der Legal Hold aufgehoben wurde, auch wenn ihre Aufbewahrungsfrist abgelaufen ist. • Falsch: Dieser Eimer steht derzeit nicht unter einer gesetzlichen Aufbewahrungspflichten. Objekte in diesem Bucket können nach Ablauf ihres Aufbewahrungszeitraums gelöscht werden.
Automatisches Löschen	• Wahr: Die Objekte in diesem Bucket werden automatisch gelöscht, sobald ihre Aufbewahrungsfrist abgelaufen ist, es sei denn, der Bucket unterliegt einer gesetzlichen Aufbewahrungspflichten. • False: Die Objekte in diesem Bucket werden nicht automatisch gelöscht, wenn die Aufbewahrungsfrist abgelaufen ist. Sie müssen diese Objekte manuell löschen, wenn Sie sie löschen müssen.

Fehlerantworten

Wenn der Bucket nicht als konform angelegt wurde, lautet der HTTP-Statuscode für die Antwort `404 Not Found` , mit einem S3-Fehlercode von `XNoSuchBucketCompliance`.

Veraltet: PUT Bucket Compliance Request

Die ANFORDERUNG „PUT Bucket-Compliance“ ist veraltet. Sie können diese Anforderung jedoch weiterhin verwenden, um die Compliance-Einstellungen für einen vorhandenen Bucket zu ändern, der die Compliance-Anforderungen erfüllt. Sie können beispielsweise einen vorhandenen Bucket auf „Legal Hold“ platzieren oder den

Aufbewahrungszeitraum erhöhen.

> Die in früheren StorageGRID-Versionen verfügbare Funktion für die StorageGRID-Konformität ist veraltet und wurde durch S3-Objektsperre ersetzt. Im Folgenden finden Sie weitere Informationen:
>
> - "Konfigurieren Sie die S3-Objektsperre über die S3-REST-API"
> - "NetApp Knowledge Base: Management älterer, konformer Buckets für StorageGRID 11.5"

Sie müssen über die s3:PutBucketCompliance-Berechtigung verfügen oder als Kontoroot vorliegen, um diesen Vorgang abzuschließen.

Wenn Sie eine PUT Bucket-Compliance-Anforderung ausgeben, müssen Sie für jedes Feld der Compliance-Einstellungen einen Wert angeben.

Anforderungsbeispiel

In dieser Beispielanforderung werden die Compliance-Einstellungen für den Bucket mit dem Namen geändert `mybucket`. In diesem Beispiel werden Objekte in nun für zwei Jahre (1,051,200 Minuten) statt für ein Jahr aufbewahrt, beginnend bei der Aufnahme des Objekts in `mybucket` das Raster. Es gibt keine gesetzliche Aufbewahrungspflichten auf diesem Bucket. Jedes Objekt wird nach zwei Jahren automatisch gelöscht.

```
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	Beschreibung
WiederholungPeriodMinuten	Die Länge des Aufbewahrungszeitraums für Objekte, die diesem Bucket hinzugefügt wurden, in Minuten Der Aufbewahrungszeitraum beginnt, wenn das Objekt in das Raster aufgenommen wird. **Wichtig** Wenn Sie einen neuen Wert für RetentionPeriodMinutes angeben, müssen Sie einen Wert angeben, der der aktuellen Aufbewahrungsfrist des Buckets entspricht oder größer ist. Nachdem die Aufbewahrungsfrist des Buckets festgelegt wurde, können Sie diesen Wert nicht verringern, sondern nur erhöhen.

Name	Beschreibung
LegalAlte	• Wahr: Dieser Bucket befindet sich derzeit in einer gesetzlichen Aufbewahrungspflichten. Objekte in diesem Bucket können erst gelöscht werden, wenn der Legal Hold aufgehoben wurde, auch wenn ihre Aufbewahrungsfrist abgelaufen ist. • Falsch: Dieser Eimer steht derzeit nicht unter einer gesetzlichen Aufbewahrungspflichten. Objekte in diesem Bucket können nach Ablauf ihres Aufbewahrungszeitraums gelöscht werden.
Automatisches Löschen	• Wahr: Die Objekte in diesem Bucket werden automatisch gelöscht, sobald ihre Aufbewahrungsfrist abgelaufen ist, es sei denn, der Bucket unterliegt einer gesetzlichen Aufbewahrungspflichten. • False: Die Objekte in diesem Bucket werden nicht automatisch gelöscht, wenn die Aufbewahrungsfrist abgelaufen ist. Sie müssen diese Objekte manuell löschen, wenn Sie sie löschen müssen.

Konsistenz für Compliance-Einstellungen

Wenn Sie die Compliance-Einstellungen für einen S3-Bucket mit EINER PUT-Bucket-Compliance-Anforderung aktualisieren, versucht StorageGRID, die Metadaten des Buckets im Grid zu aktualisieren. Standardmäßig verwendet StorageGRID die **strong-global**-Konsistenz, um sicherzustellen, dass alle Datacenter-Standorte und alle Speicher-Nodes, die Bucket-Metadaten enthalten, für die geänderten Compliance-Einstellungen eine Lese-nach-Schreiben-Konsistenz aufweisen.

Wenn StorageGRID die **strong-global**-Konsistenz nicht erreichen kann, weil ein Rechenzentrum oder mehrere Speicherknoten an einem Standort nicht verfügbar sind, lautet der HTTP-Statuscode für die Antwort `503 Service Unavailable.`

Wenn Sie diese Antwort erhalten, müssen Sie sich an den Grid-Administrator wenden, um sicherzustellen, dass die erforderlichen Storage-Services so schnell wie möglich verfügbar gemacht werden. Wenn der Grid-Administrator nicht in der Lage ist, genügend Speicher-Nodes an jedem Standort zur Verfügung zu stellen, kann der technische Support Sie auffordern, die fehlgeschlagene Anforderung erneut zu versuchen, indem Sie die Konsistenz von **strong-site** erzwingen.

> Erzwingen Sie niemals die * strong-site* Konsistenz für PUT Bucket Compliance, es sei denn, Sie wurden von der technischen Unterstützung dazu angewiesen, und es sei denn, Sie verstehen die möglichen Konsequenzen, die sich aus der Verwendung dieses Levels ergeben.

Wenn die Konsistenz auf **strong-site** reduziert wird, garantiert StorageGRID, dass aktualisierte Compliance-Einstellungen nur für Client-Anforderungen innerhalb eines Standorts Lese-nach-Schreiben-Konsistenz aufweisen. Das bedeutet, dass das StorageGRID System vorübergehend mehrere inkonsistente Einstellungen für diesen Bucket bietet, bis alle Standorte und Storage-Nodes verfügbar sind. Die inkonsistenten Einstellungen können zu unerwarteten und unerwünschten Verhaltensweisen führen. Wenn Sie beispielsweise einen Bucket unter einen Legal Hold setzen und eine niedrigere Konsistenz erzwingen, könnten die vorherigen Compliance-Einstellungen des Buckets (d. h. Legal Hold off) an einigen Rechenzentrumsstandorten weiterhin wirksam sein. Aus diesem Grund können Objekte, die Ihrer Meinung nach in einer gesetzlichen Wartefrist liegen, nach Ablauf ihres Aufbewahrungszeitraums entweder durch den Benutzer oder durch AutoDelete gelöscht werden, sofern diese Option aktiviert ist.

Um die Verwendung der Konsistenz von **strong-site** zu erzwingen, geben Sie die Anforderung für die Einhaltung von PUT Bucket erneut aus und fügen Sie den `Consistency-Control` HTTP-

Anforderungsheader wie folgt ein:

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

Fehlerantworten

- Wenn der Bucket nicht für die Konformität erstellt wurde, lautet der HTTP-Statuscode für die Antwort `404 Not Found`.
- Wenn `RetentionPeriodMinutes` in der Anforderung weniger als die aktuelle Aufbewahrungsfrist des Buckets liegt, lautet der HTTP-Statuscode `400 Bad Request`.

Verwandte Informationen

"Veraltet: PUT Bucket-Request-Änderungen aus Compliance-Gründen"

Bucket- und Gruppenzugriffsrichtlinien

Verwendung von Bucket- und Gruppenzugriffsrichtlinien

StorageGRID verwendet die Richtliniensprache für Amazon Web Services (AWS), um S3-Mandanten die Kontrolle des Zugriffs auf Buckets und Objekte innerhalb dieser Buckets zu ermöglichen. Das StorageGRID System implementiert eine Untermenge der S3-REST-API-Richtliniensprache. Zugriffsrichtlinien für die S3 API werden in JSON geschrieben.

Zugriffsrichtlinien – Überblick

Von StorageGRID werden zwei Arten von Zugriffsrichtlinien unterstützt:

- **Bucket-Richtlinien**, die mit den Operationen GetBucket Policy, PutBucket Policy und DeleteBucket Policy S3 API oder der Tenant Manager- oder Tenant Management API verwaltet werden. Bucket-Richtlinien sind mit Buckets verknüpft, so dass sie so konfiguriert sind, dass sie den Zugriff durch Benutzer im Bucket-Eigentümerkonto oder andere Konten an den Bucket und die darin befindlichen Objekte steuern. Eine Bucket-Richtlinie gilt nur für einen Bucket und möglicherweise auch für mehrere Gruppen.
- **Gruppenrichtlinien**, die mit dem Tenant Manager oder der Mandantenmanagement-API konfiguriert sind. Gruppenrichtlinien sind einer Gruppe im Konto zugeordnet, sodass sie so konfiguriert sind, dass sie der Gruppe ermöglichen, auf bestimmte Ressourcen zuzugreifen, die dem Konto gehören. Eine Gruppenrichtlinie gilt nur für eine Gruppe und möglicherweise für mehrere Buckets.

> Es gibt keine Unterschiede in der Priorität zwischen Gruppen- und Bucket-Richtlinien.

StorageGRID Bucket und Gruppenrichtlinien folgen einer bestimmten Grammatik, die von Amazon definiert wurde. Innerhalb jeder Richtlinie gibt es eine Reihe von Richtlinienerklärungen, und jede Aussage enthält die folgenden Elemente:

- Statement-ID (Sid) (optional)
- Wirkung

- Principal/NotPrincipal
- Ressource/Ressource
- Aktion/Notaktion
- Bedingung (optional)

Richtlinienaussagen werden mithilfe dieser Struktur erstellt, um Berechtigungen anzugeben: <Effekt> gewähren, um <Principal> <Aktion> auf <Ressource> durchzuführen, wenn <Bedingung> angewendet wird.

Jedes Richtlinienelement wird für eine bestimmte Funktion verwendet:

Element	Beschreibung
Sid	Das Sid-Element ist optional. Der Sid ist nur als Beschreibung für den Benutzer gedacht. Diese wird vom StorageGRID System gespeichert, aber nicht interpretiert.
Wirkung	Verwenden Sie das Effektelement, um festzustellen, ob die angegebenen Vorgänge zulässig oder verweigert werden. Sie müssen anhand der Schlüsselwörter für unterstütztes Aktionselement Operationen identifizieren, die für Buckets oder Objekte zugelassen (oder verweigert) werden.
Principal/NotPrincipal	Benutzer, Gruppen und Konten können auf bestimmte Ressourcen zugreifen und bestimmte Aktionen ausführen. Wenn in der Anfrage keine S3-Signatur enthalten ist, ist ein anonymer Zugriff durch Angabe des Platzhalterzeichens (*) als Principal zulässig. Standardmäßig hat nur das Konto-Root Zugriff auf Ressourcen, die dem Konto gehören. Sie müssen nur das Hauptelement in einer Bucket-Richtlinie angeben. Bei Gruppenrichtlinien ist die Gruppe, der die Richtlinie zugeordnet ist, das implizite Prinzipalelement.
Ressource/Ressource	Das Ressourcenelement identifiziert Buckets und Objekte. Sie können Buckets und Objekten über den ARN (Amazon Resource Name) Berechtigungen gewähren oder verweigern, um die Ressource zu identifizieren.
Aktion/Notaktion	Die Elemente Aktion und Wirkung sind die beiden Komponenten von Berechtigungen. Wenn eine Gruppe eine Ressource anfordert, wird ihnen entweder der Zugriff auf die Ressource gewährt oder verweigert. Der Zugriff wird verweigert, es sei denn, Sie weisen ausdrücklich Berechtigungen zu, aber Sie können explizites Ablehnen verwenden, um eine von einer anderen Richtlinie gewährte Berechtigung zu überschreiben.
Zustand	Das Bedingungselement ist optional. Unter Bedingungen können Sie Ausdrücke erstellen, um zu bestimmen, wann eine Richtlinie angewendet werden soll.

Im Element Aktion können Sie das Platzhalterzeichen (*) verwenden, um alle Vorgänge oder eine Untermenge

von Vorgängen anzugeben. Diese Aktion entspricht beispielsweise Berechtigungen wie s3:GetObject, s3:PutObject und s3:DeleteObject.

```
s3:*Object
```

Im Element Ressource können Sie die Platzhalterzeichen (*) und (?) verwenden. Während das Sternchen (*) mit 0 oder mehr Zeichen übereinstimmt, ist das Fragezeichen (?) Entspricht einem beliebigen Zeichen.

Im Hauptelement werden Platzhalterzeichen nicht unterstützt, außer zum Festlegen eines anonymen Zugriffs, der allen Personen die Berechtigung gewährt. Sie legen beispielsweise den Platzhalter (*) als Principal-Wert fest.

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

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

Im folgenden Beispiel verwendet die Anweisung die Elemente „Effekt“, „Principal“, „Aktion“ und „Ressource“. Dieses Beispiel zeigt eine vollständige Bucket-Policy-Anweisung, die den Effekt „allow“ verwendet, um den Principals, der Admin-Gruppe und der Finanzgruppe `federated-group/finance` Berechtigungen zur Ausführung der Aktion `s3:ListBucket` für den Bucket namens und die Aktion `s3:GetObject` für alle Objekte innerhalb dieses Buckets `mybucket` zu geben `federated-group/admin`.

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

Die Bucket-Richtlinie hat eine Größenbeschränkung von 20,480 Byte, und die Gruppenrichtlinie hat ein Größenlimit von 5,120 Byte.

Konsistenz von Richtlinien

Standardmäßig sind alle Aktualisierungen, die Sie an Gruppenrichtlinien vornehmen, letztendlich konsistent. Wenn eine Gruppenrichtlinie konsistent wird, können die Änderungen aufgrund des Caching von Richtlinien weitere 15 Minuten in Anspruch nehmen. Standardmäßig sind alle Updates an Bucket-Richtlinien stark konsistent.

Sie können bei Bedarf die Konsistenzgarantien für Bucket-Richtlinienaktualisierungen ändern. Beispielsweise kann es vorkommen, dass eine Änderung an einer Bucket-Richtlinie bei einem Standortausfall verfügbar ist.

In diesem Fall können Sie entweder den Header in der Anforderung „PutBucket Policy“ festlegen `Consistency-Control` oder die Konsistenzanforderung „PUT Bucket“ verwenden. Wenn eine Bucket-Richtlinie konsistent wird, können die Änderungen durch das Caching von Richtlinien zusätzliche 8 Sekunden in Anspruch nehmen.

> Wenn Sie die Konsistenz auf einen anderen Wert setzen, um eine temporäre Situation zu umgehen, stellen Sie sicher, dass die Einstellung auf Bucket-Ebene wieder auf ihren ursprünglichen Wert zurückgesetzt wird, wenn Sie fertig sind. Andernfalls wird für alle zukünftigen Bucket-Anforderungen die geänderte Einstellung verwendet.

Verwenden Sie ARN in den Richtlinienerklärungen

In den Richtlinienerklärungen wird das ARN in Haupt- und Ressourcenelementen verwendet.

- Verwenden Sie diese Syntax, um die S3-Ressource ARN anzugeben:

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

- Verwenden Sie diese Syntax, um die Identitätressource ARN (Benutzer und Gruppen) festzulegen:

```
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
```

Weitere Überlegungen:

- Sie können das Sternchen (*) als Platzhalter verwenden, um Null oder mehr Zeichen im Objektschlüssel zu entsprechen.
- Internationale Zeichen, die im Objektschlüssel angegeben werden können, sollten mit JSON UTF-8 oder mit JSON \U Escape Sequenzen codiert werden. Die prozentuale Kodierung wird nicht unterstützt.

  "RFC 2141 URN Syntax"

Der HTTP-Anforderungskörper für den PutBucketPolicy-Vorgang muss mit charset=UTF-8 codiert werden.

Geben Sie Ressourcen in einer Richtlinie an

In Richtlinienausrechnungen können Sie mithilfe des Elements Ressourcen den Bucket oder das Objekt angeben, für das Berechtigungen zulässig oder verweigert werden.

- Jede Richtlinienanweisung erfordert ein Ressourcenelement. In einer Richtlinie werden Ressourcen durch das Element oder alternativ `NotResource` zum Ausschluss gekennzeichnet `Resource`.
- Sie legen Ressourcen mit einer S3-Ressource ARN fest. Beispiel:

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

- Sie können Richtlinienvariablen auch innerhalb des Objektschlüssels verwenden. Beispiel:

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

- Der Ressourcenwert kann einen Bucket angeben, der beim Erstellen einer Gruppenrichtlinie noch nicht vorhanden ist.

Principals in einer Policy angeben

Verwenden Sie das Hauptelement, um das Benutzer-, Gruppen- oder Mandantenkonto zu identifizieren, das über die Richtlinienanweisung Zugriff auf die Ressource erlaubt/verweigert wird.

- Jede Richtlinienanweisung in einer Bucket-Richtlinie muss ein Principal Element enthalten. Richtlinienanweisungen in einer Gruppenrichtlinie benötigen das Hauptelement nicht, da die Gruppe als Hauptelement verstanden wird.
- In einer Richtlinie werden Prinzipale durch das Element „Principal“ oder alternativ „NotPrincipal“ für den Ausschluss gekennzeichnet.
- Kontobasierte Identitäten müssen mit einer ID oder einem ARN angegeben werden:

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

- In diesem Beispiel wird die Mandanten-Account-ID 27233906934684427525 verwendet, die das Konto-Root und alle Benutzer im Konto enthält:

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

- Sie können nur das Konto-Root angeben:

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

- Sie können einen bestimmten föderierten Benutzer („Alex“) angeben:

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

- Sie können eine bestimmte föderierte Gruppe („Manager“) angeben:

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

- Sie können einen anonymen Principal angeben:

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

- Um Mehrdeutigkeiten zu vermeiden, können Sie die Benutzer-UUID anstelle des Benutzernamens verwenden:

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

  Angenommen, Alex verlässt die Organisation und der Benutzername `Alex` wird gelöscht. Wenn ein neuer Alex der Organisation Beitritt und demselben Benutzernamen zugewiesen wird `Alex`, erbt der neue Benutzer möglicherweise unbeabsichtigt die Berechtigungen, die dem ursprünglichen Benutzer gewährt wurden.

- Der Hauptwert kann einen Gruppen-/Benutzernamen angeben, der beim Erstellen einer Bucket-Richtlinie noch nicht vorhanden ist.

Legen Sie Berechtigungen in einer Richtlinie fest

In einer Richtlinie wird das Aktionselement verwendet, um Berechtigungen einer Ressource zuzulassen/zu verweigern. Es gibt eine Reihe von Berechtigungen, die Sie in einer Richtlinie festlegen können, die durch das Element „Aktion“ gekennzeichnet sind, oder alternativ durch „NotAction“ für den Ausschluss. Jedes dieser Elemente wird bestimmten S3-REST-API-Operationen zugeordnet.

In den Tabellen werden die Berechtigungen aufgeführt, die auf Buckets angewendet werden, sowie die Berechtigungen, die für Objekte gelten.

> Amazon S3 verwendet jetzt die s3:PutReplicationConfiguration-Berechtigung sowohl für die PutBucketReplication- als auch für die DeleteBucketReplication-Aktionen. StorageGRID verwendet für jede Aktion separate Berechtigungen, die mit der ursprünglichen Amazon S3 Spezifikation übereinstimmt.

> Ein Löschen wird durchgeführt, wenn ein Put zum Überschreiben eines vorhandenen Werts verwendet wird.

Berechtigungen, die für Buckets gelten

Berechtigungen	S3-REST-API-OPERATIONEN	Individuell für StorageGRID
s3:CreateBucket	CreateBucket	Ja. **Hinweis**: Nur in Gruppenrichtlinien verwenden.
s3:DeleteBucket	DeleteBucket	
s3:DeleteBucketMetadataBenachrichtigung	Konfiguration für die Benachrichtigung über Bucket-Metadaten LÖSCHEN	Ja.
s3:DeleteBucketPolicy	DeleteBucketRichtlinien	
s3:DeleteReplicationConfiguration	DeleteBucketReplication	Ja, separate Berechtigungen für PUT und DELETE
s3:GetBucketAcl	GetBucketAcl	
s3:GetBucketCompliance	GET Bucket-Compliance (veraltet)	Ja.
s3:GetBucketConsistency	Get Bucket-Konsistenz	Ja.
s3:GetBucketCORS	GetBucketCors	
s3:GetVerschlüsselungKonfiguration	GetBucketEncryption	
s3:GetBucketLastAccessTime	ZEITPUNKT des letzten Zugriffs FÜR den Bucket ABRUFEN	Ja.
s3:GetBucketLocation	GetBucketLocation	
s3:GetBucketMetadataBenachrichtigung	Konfiguration der Bucket-Metadaten-Benachrichtigungen ABRUFEN	Ja.
s3:GetBucketBenachrichtigung	GetBucketNotificationConfiguration	
s3:GetBucketObjectLockConfiguration	GetObjectLockConfiguration	
s3:GetBucketPolicy	GetBucketPolicy	
s3:GetBucketTagging	GetBucketTagging	

Berechtigungen	S3-REST-API-OPERATIONEN	Individuell für StorageGRID
s3:GetBucketVersionierung	GetBucketVersioning	
s3:GetLifecycleKonfiguration	GetBucketLifecycleKonfiguration	
s3:GetReplicationConfiguration	GetBucketReplication	
s3:ListAllMyBuchs	• ListBuchs • GET Storage-Auslastung	Ja, für DIE GET Storage-Nutzung. **Hinweis**: Nur in Gruppenrichtlinien verwenden.
s3:ListBucket	• ListObjekte • HeadBucket • Objekt restoreObject	
s3:ListBucketMultipartUploads	• ListMultipartUploads • Objekt restoreObject	
s3:ListBucketVersions	Get Bucket-Versionen	
s3:PutBucketCompliance	PUT Bucket-Compliance (veraltet)	Ja.
s3:PutBucketConsistency	PUT Bucket-Konsistenz	Ja.
s3:PutBucketCORS	• DeleteBucketCors† • PutBucketCors	
s3:PutVerschlüsselungKonfiguration	• DeleteBucketEncryption • PutBucketEncryption	
s3:PutBucketLastAccessTime	PUT Bucket-Zeit für den letzten Zugriff	Ja.
s3:PutBucketMetadataBenachrichtigung	PUT Bucket-Metadaten-Benachrichtigungskonfiguration	Ja.
s3:PutBucketNotification	PutBucketNotificationKonfiguration	

Berechtigungen	S3-REST-API-OPERATIONEN	Individuell für StorageGRID
s3:PutBucketObjectLockConfiguration	• CreateBucket mit dem `x-amz-bucket-object-lock-enabled: true` Anforderungsheader (erfordert auch die Berechtigung s3:CreateBucket) • PutObjectLockKonfiguration	
s3:PutBucketPolicy	PutBucketPolicy	
s3:PutBucketTagging	• DeleteBucketTagging† • PutBucketTagging	
s3:PutBucketVersionierung	PutBucketVersioning	
s3:PutLifecycleKonfiguration	• DeleteBucketLifecycle† • PutBucketLifecycleKonfiguration	
s3:PuteReplikationKonfiguration	PutBucketReplication	Ja, separate Berechtigungen für PUT und DELETE

Berechtigungen, die sich auf Objekte beziehen

Berechtigungen	S3-REST-API-OPERATIONEN	Individuell für StorageGRID
s3:AbortMehrteilaUpload	• AbortMehrteilaUpload • Objekt restoreObject	
s3:BypassGovernanceAufbewahrung	• DeleteObject • Objekte deObjekteObjekte • PutObjectRetention	
s3:DeleteObject	• DeleteObject • Objekte deObjekteObjekte • Objekt restoreObject	
s3:DeleteObjectTagging	DeleteObjectTagging	
s3:DeleteObjectVersionTagging	DeleteObjectTagging (eine spezifische Version des Objekts)	

Berechtigungen	S3-REST-API-OPERATIONEN	Individuell für StorageGRID
s3:DeleteObjectVersion	DeleteObject (eine bestimmte Version des Objekts)	
s3:GetObject	• GetObject • HeadObject • Objekt restoreObject • SelektierObjectContent	
s3:GetObjectAcl	GetObjectAcl	
s3:GetObjectLegalOld	GetObjectLegalHold	
s3:GetObjectRetention	GetObjectRetention	
s3:GetObjectTagging	GetObjectTagging	
s3:GetObjectVersionTagging	GetObjectTagging (eine spezifische Version des Objekts)	
s3:GetObjectVersion	GetObject (eine spezifische Version des Objekts)	
s3:ListeMultipartUploadParts	ListParts, RestoreObject	
s3:PutObject	• PutObject • CopyObject • Objekt restoreObject • CreateMultipartUpload • CompleteMultipartUpload • UploadTeil • UploadPartCopy	
s3:PuttObjectLegalOld	PutObjectLegalHold	
s3:PutObjectRetention	PutObjectRetention	
s3:PuttObjectTagging	PutObjectTagging	
s3:PuttObjectVersionTagging	PutObjectTagging (eine spezifische Version des Objekts)	

Berechtigungen	S3-REST-API-OPERATIONEN	Individuell für StorageGRID
s3:PutOverwrite Object	• PutObject • CopyObject • PutObjectTagging • DeleteObjectTagging • CompleteMultipartUpload	Ja.
s3:RestoreObject	Objekt restoreObject	

Verwenden Sie PutOverwriteObject-Berechtigung

die s3:PutOverwriteObject-Berechtigung ist eine benutzerdefinierte StorageGRID-Berechtigung, die für Vorgänge gilt, die Objekte erstellen oder aktualisieren. Durch diese Berechtigung wird festgelegt, ob der Client die Daten, benutzerdefinierte Metadaten oder S3-Objekt-Tagging überschreiben kann.

Mögliche Einstellungen für diese Berechtigung sind:

- **Zulassen**: Der Client kann ein Objekt überschreiben. Dies ist die Standardeinstellung.
- **Deny**: Der Client kann ein Objekt nicht überschreiben. Wenn die Option „Ablehnen“ eingestellt ist, funktioniert die Berechtigung „PutOverwriteObject“ wie folgt:
  - Wenn ein vorhandenes Objekt auf demselben Pfad gefunden wird:
    - Die Daten, benutzerdefinierten Metadaten oder S3-Objekt-Tagging des Objekts können nicht überschrieben werden.
    - Alle laufenden Aufnahmevorgänge werden abgebrochen und ein Fehler wird zurückgegeben.
    - Wenn die S3-Versionierung aktiviert ist, verhindert die Einstellung Deny, dass PutObjectTagging- oder DeleteObjectTagging-Operationen das TagSet für ein Objekt und seine nicht aktuellen Versionen ändern.
  - Wenn ein vorhandenes Objekt nicht gefunden wird, hat diese Berechtigung keine Wirkung.
- Wenn diese Berechtigung nicht vorhanden ist, ist der Effekt der gleiche, als ob Allow-were gesetzt wurden.

> Wenn die aktuelle S3-Richtlinie Überschreiben zulässt und die PutOverwriteObject-Berechtigung auf Deny festgelegt ist, kann der Client die Daten, benutzerdefinierten Metadaten oder Objekt-Tagging eines Objekts nicht überschreiben. Wenn zusätzlich das Kontrollkästchen **Client-Änderung verhindern** aktiviert ist (**KONFIGURATION** > **Sicherheitseinstellungen** > **Netzwerk und Objekte**), setzt diese Einstellung die Einstellung der PutOverwriteObject-Berechtigung außer Kraft.

Legen Sie Bedingungen in einer Richtlinie fest

Die Bedingungen legen fest, wann eine Richtlinie in Kraft sein wird. Die Bedingungen bestehen aus Bedienern und Schlüsselwertpaaren.

Bedingungen Verwenden Sie Key-Value-Paare für die Auswertung. Ein Bedingungselement kann mehrere Bedingungen enthalten, und jede Bedingung kann mehrere Schlüsselwert-Paare enthalten. Der Bedingungsblock verwendet das folgende Format:

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

Im folgenden Beispiel verwendet die IPAddress-Bedingung den SourceIp-Bedingungsschlüssel.

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

Unterstützte Bedingungsoperatoren

Bedingungsoperatoren werden wie folgt kategorisiert:

- Zeichenfolge
- Numerisch
- Boolesch
- IP-Adresse
- Null-Prüfung

Bedingungsoperatoren	Beschreibung
StringEquals	Vergleicht einen Schlüssel mit einem Zeichenfolgenwert, der auf exakter Übereinstimmung basiert (Groß-/Kleinschreibung wird beachtet).
StringNotEquals	Vergleicht einen Schlüssel mit einem Zeichenfolgenwert, der auf negatives Matching basiert (Groß-/Kleinschreibung wird beachtet).
StringEqusIgnoreCase	Vergleicht einen Schlüssel mit einem Zeichenfolgenwert, der auf exakter Übereinstimmung basiert (Groß-/Kleinschreibung wird ignoriert).
StringNotEqualesIgnoreCase	Vergleicht einen Schlüssel mit einem String-Wert, der auf negatives Matching basiert (Groß-/Kleinschreibung wird ignoriert).
StringLike	Vergleicht einen Schlüssel mit einem Zeichenfolgenwert, der auf exakter Übereinstimmung basiert (Groß-/Kleinschreibung wird beachtet). Kann * und ? Platzhalterzeichen enthalten.
StringNotLike	Vergleicht einen Schlüssel mit einem Zeichenfolgenwert, der auf negatives Matching basiert (Groß-/Kleinschreibung wird beachtet). Kann * und ? Platzhalterzeichen enthalten.

Bedingungsoperatoren	Beschreibung
Ziffern	Vergleicht einen Schlüssel mit einem numerischen Wert, der auf exakter Übereinstimmung basiert.
ZiffernNotequals	Vergleicht einen Schlüssel mit einem numerischen Wert, der auf negatives Matching basiert.
NumericGreaterThan	Vergleicht einen Schlüssel mit einem numerischen Wert basierend auf dem „größer als“-Vergleich.
ZahlungGreaterThanEquals	Vergleicht einen Schlüssel mit einem numerischen Wert basierend auf dem „größer als oder gleich“-Vergleich.
NumericLessThan	Vergleicht einen Schlüssel mit einem numerischen Wert basierend auf „weniger als“-Übereinstimmung.
ZahlungWenigerThanEquals	Vergleicht einen Schlüssel mit einem numerischen Wert basierend auf dem „kleiner als oder gleich“-Vergleich.
Bool	Vergleicht einen Schlüssel mit einem booleschen Wert basierend auf „true“ oder „false“-Matching.
IP-Adresse	Vergleicht einen Schlüssel mit einer IP-Adresse oder einem IP-Adressbereich.
NotIpAddress	Vergleicht einen Schlüssel mit einer IP-Adresse oder einem IP-Adressbereich, basierend auf negatiertem Abgleich.
Null	Überprüft, ob im aktuellen Anforderungskontext ein Bedingungsschlüssel vorhanden ist.

Unterstützte Bedingungsschlüssel

Zustandsschlüssel	Aktionen	Beschreibung
aws:SourceIp	IP-Operatoren	Vergleicht mit der IP-Adresse, von der die Anfrage gesendet wurde. Kann für Bucket- oder Objektvorgänge verwendet werden **Hinweis:** wurde die S3-Anfrage über den Lastbalancer-Dienst auf Admin-Knoten und Gateways-Knoten gesendet, wird dies mit der IP-Adresse verglichen, die vor dem Load Balancer Service liegt. **Hinweis**: Wenn ein Drittanbieter-, nicht-transparenter Load Balancer verwendet wird, wird dies mit der IP-Adresse dieses Load Balancer verglichen. Jede `X-Forwarded-For` Kopfzeile wird ignoriert, da ihre Gültigkeit nicht ermittelt werden kann.
aws:Benutzername	Ressource/Identität	Vergleicht mit dem Benutzernamen des Absenders, von dem die Anfrage gesendet wurde. Kann für Bucket- oder Objektvorgänge verwendet werden
s3:Trennzeichen	s3:ListBucket und s3:ListBucketVersions Berechtigungen	Wird mit dem in einer ListObjects- oder ListObjectVersions-Anforderung angegebenen Trennzeichen-Parameter verglichen.
s3:ExistingObjectTag/<tag-key>	s3:DeleteObjectTagging s3:DeleteObjectVersionTagging s3:GetObject s3:GetObjectAcl 3:GetObjectTagging s3:GetObjectVersion s3:GetObjectVersionAkl s3:GetObjectVersionTagging s3:PutObjectAcl s3:PuttObjectTagging s3:PutObjectVersionAcl s3:PuttObjectVersionTagging	Erfordert, dass das vorhandene Objekt über den spezifischen Tag-Schlüssel und -Wert verfügt.

Zustandsschlüssel	Aktionen	Beschreibung
s3:max-keys	s3:ListBucket und s3:ListBucketVersions Berechtigungen	Wird mit dem Parameter max-keys verglichen, der in einer ListObjects- oder ListObjectVersions-Anforderung angegeben ist.
s3:verbleibende Object-Lock-Retention-Tage	s3:PutObject	Vergleicht das im Anforderungskopf angegebene oder aus dem Standardaufbewahrungszeitraum berechnete Aufbewahrungsdatum `x-amz-object-lock-retain-until-date`, um sicherzustellen, dass diese Werte innerhalb des zulässigen Bereichs für die folgenden Anforderungen liegen: • PutObject • CopyObject • CreateMultipartUpload
s3:verbleibende Object-Lock-Retention-Tage	s3:PutObjectRetention	Vergleicht das in der PutObjectRetention-Anfrage angegebene Aufbewahrungsdatum, um sicherzustellen, dass es innerhalb des zulässigen Bereichs liegt.
s3:Präfix	s3:ListBucket und s3:ListBucketVersions Berechtigungen	Wird mit dem Präfix-Parameter verglichen, der in einer ListObjects- oder ListObjectVersions-Anforderung angegeben ist.
s3:RequestObjectTag/<tag-key>	s3:PutObject s3:PuttObjectTagging s3:PuttObjectVersionTagging	Erfordert einen bestimmten Tag-Schlüssel und einen bestimmten Wert, wenn die Objektanforderung Tagging beinhaltet.

Geben Sie Variablen in einer Richtlinie an

Sie können Variablen in Richtlinien verwenden, um die Richtlinieninformationen auszufüllen, wenn sie verfügbar sind. Sie können Richtlinienvariablen im Element und in Stringvergleiche im `Condition` Element verwenden `Resource`.

In diesem Beispiel ist die Variable `${aws:username}` Teil des Elements Ressource:

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

In diesem Beispiel ist die Variable `${aws:username}` Teil des Bedingungswerts im Bedingungsblock:

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

Variabel	Beschreibung
`${aws:SourceIp}`	Verwendet den SourceIp-Schlüssel als bereitgestellte Variable.
`${aws:username}`	Verwendet den Benutzernamen-Schlüssel als bereitgestellte Variable.
`${s3:prefix}`	Verwendet den Service-spezifischen Präfixschlüssel als bereitgestellte Variable.
`${s3:max-keys}`	Verwendet die Service-spezifische max-keys als die angegebene Variable.
`${*}`	Sonderzeichen. Verwendet das Zeichen als Literal * -Zeichen.
`${?}`	Sonderzeichen. Verwendet das Zeichen als Literal ? Zeichen.
`${$}`	Sonderzeichen. Verwendet das Zeichen als Literal USD Zeichen.

Erstellen von Richtlinien, die eine spezielle Handhabung erfordern

Manchmal kann eine Richtlinie Berechtigungen erteilen, die für die Sicherheit oder die Gefahr für einen fortgesetzten Betrieb gefährlich sind, z. B. das Sperren des Root-Benutzers des Kontos. Die StorageGRID S3-REST-API-Implementierung ist bei der Richtlinienvalidierung weniger restriktiv als Amazon, aber auch bei der Richtlinienbewertung streng.

Richtlinienbeschreibung	Richtlinientyp	Verhalten von Amazon	Verhalten von StorageGRID
Verweigern Sie sich selbst irgendwelche Berechtigungen für das Root-Konto	Eimer	Gültig und durchgesetzt, aber das Root-Benutzerkonto behält die Berechtigung für alle S3 Bucket-Richtlinienvorgänge bei	Gleich
Verweigern Sie selbst jegliche Berechtigungen für Benutzer/Gruppe	Gruppieren	Gültig und durchgesetzt	Gleich

Richtlinienbeschreibung	Richtlinientyp	Verhalten von Amazon	Verhalten von StorageGRID
Erlauben Sie einer fremden Kontogruppe jegliche Berechtigung	Eimer	Ungültiger Principal	Gültig, aber die Berechtigungen für alle S3-Bucket-Richtlinienvorgänge geben bei Richtlinienzugelassen durch eine Richtlinie einen nicht zugelassenen 405-Method-Fehler zurück
Berechtigung für ein ausländisches Konto oder einen Benutzer zulassen	Eimer	Gültig, aber die Berechtigungen für alle S3-Bucket-Richtlinienvorgänge geben bei Richtlinienzugelassen durch eine Richtlinie einen nicht zugelassenen 405-Method-Fehler zurück	Gleich
Alle Berechtigungen für alle Aktionen zulassen	Eimer	Gültig, aber Berechtigungen für alle S3-Bucket-Richtlinienvorgänge geben einen 405 Methode nicht erlaubten Fehler für das ausländische Konto Root und Benutzer zurück	Gleich
Alle Berechtigungen für alle Aktionen verweigern	Eimer	Gültig und durchgesetzt, aber das Root-Benutzerkonto behält die Berechtigung für alle S3 Bucket-Richtlinienvorgänge bei	Gleich
Principal ist ein nicht existierender Benutzer oder eine Gruppe	Eimer	Ungültiger Principal	Gültig
Die Ressource ist ein nicht existierender S3-Bucket	Gruppieren	Gültig	Gleich
Principal ist eine lokale Gruppe	Eimer	Ungültiger Principal	Gültig

Richtlinienbeschreibung	Richtlinientyp	Verhalten von Amazon	Verhalten von StorageGRID
Die Richtlinie gewährt einem Konto ohne Eigentümer (einschließlich anonymer Konten) Berechtigungen zum Setzen von Objekten.	Eimer	Gültig. Objekte sind Eigentum des Erstellerkontos, und die Bucket-Richtlinie gilt nicht. Das Ersteller-Konto muss über Objekt-ACLs Zugriffsrechte für das Objekt gewähren.	Gültig. Der Eigentümer der Objekte ist das Bucket-Owner-Konto. Bucket-Richtlinie gilt.

WORM-Schutz (Write Once, Read Many)

Sie können WORM-Buckets (Write-Once-Read-Many) erstellen, um Daten, benutzerdefinierte Objekt-Metadaten und S3-Objekt-Tagging zu sichern. SIE konfigurieren die WORM-Buckets, um das Erstellen neuer Objekte zu ermöglichen und Überschreibungen oder das Löschen vorhandener Inhalte zu verhindern. Verwenden Sie einen der hier beschriebenen Ansätze.

Um sicherzustellen, dass Überschreibungen immer verweigert werden, können Sie:

- Gehen Sie im Grid Manager zu **CONFIGURATION** > **Security** > **Security settings** > **Network and Objects** und aktivieren Sie das Kontrollkästchen **Client-Änderung verhindern**.
- Wenden Sie die folgenden Regeln und S3-Richtlinien an:
  - Fügen Sie der S3-Richtlinie einen PutOverwriteObject DENY-Vorgang hinzu.
  - Fügen Sie der S3-Richtlinie einen DeleteObject DENY-Vorgang hinzu.
  - Fügen Sie der S3-Richtlinie einen PutObject ALLOW-Vorgang hinzu.

> Wenn in einer S3-Richtlinie DeleteObject auf DENY festgelegt wird, verhindert dies nicht, dass ILM Objekte löscht, wenn eine Regel wie „Zero Copies after 30 days“ vorhanden ist.

> Selbst wenn alle diese Regeln und Richtlinien angewendet werden, schützen sie sich nicht vor gleichzeitigen Schreibvorgängen (siehe Situation A). Sie schützen vor sequenziellen Überschreibungen (siehe Situation B).

Situation A: Gleichzeitige Schreibvorgänge (nicht bewacht)

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

Situation B: Sequentielle abgeschlossene Überschreibungen (bewacht gegen)

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

Verwandte Informationen

- "Managen von Objekten durch StorageGRID ILM-Regeln"
- "Beispiel für Bucket-Richtlinien"
- "Beispiel für Gruppenrichtlinien"
- "Objektmanagement mit ILM"
- "Verwenden Sie ein Mandantenkonto"

Beispiel für Bucket-Richtlinien

Mithilfe der Beispiele in diesem Abschnitt können Sie StorageGRID-Zugriffsrichtlinien für Buckets erstellen.

Bucket-Richtlinien geben die Zugriffsberechtigungen für den Bucket an, mit dem die Richtlinie verknüpft ist. Sie konfigurieren eine Bucket-Richtlinie mithilfe der S3-PutBucketPolicy-API über eines der folgenden Tools:

- "Mandanten-Manager".
- AWS CLI mit diesem Befehl (siehe "Operationen auf Buckets"):

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

Beispiel: Lesezugriff auf einen Bucket zulassen

In diesem Beispiel darf jeder, auch anonym, Objekte im Bucket auflisten und GetObject-Operationen für alle Objekte im Bucket ausführen. Alle anderen Operationen werden abgelehnt. Beachten Sie, dass diese Richtlinie möglicherweise nicht besonders nützlich ist, da niemand außer dem Konto root über Berechtigungen zum Schreiben in den Bucket verfügt.

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

Beispiel: Jeder in einem Konto Vollzugriff zulassen, und jeder in einem anderen Konto hat nur Lesezugriff auf einen Bucket

In diesem Beispiel hat jeder in einem bestimmten Konto vollen Zugriff auf einen Bucket, während jeder in einem anderen angegebenen Konto nur berechtigt ist, den Bucket aufzulisten und GetObject-Operationen für

Objekte im Bucket durchzuführen, beginnend mit dem `shared/` Objektschlüsselpräfix.

> In StorageGRID sind Objekte, die von einem nicht-Inhaberkonto erstellt wurden (einschließlich anonymer Konten), Eigentum des Bucket-Inhaberkontos. Die Bucket-Richtlinie gilt für diese Objekte.

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

Beispiel: Lesezugriff für einen Bucket und vollständiger Zugriff durch angegebene Gruppe

In diesem Beispiel kann jeder, einschließlich anonym, den Bucket auflisten und GetObject-Operationen für alle Objekte im Bucket ausführen, während nur Benutzer, die der Gruppe im angegebenen Konto angehören, `Marketing` vollen Zugriff erhalten.

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

Beispiel: Jeder Lese- und Schreibzugriff auf einen Bucket zulassen, wenn Client im IP-Bereich ist

In diesem Beispiel darf jeder, einschließlich anonym, den Bucket auflisten und beliebige Objektvorgänge an allen Objekten im Bucket durchführen, vorausgesetzt, dass die Anforderungen aus einem bestimmten IP-Bereich stammen (54.240.143.0 bis 54.240.143.255, außer 54.240.143.188). Alle anderen Vorgänge werden abgelehnt, und alle Anfragen außerhalb des IP-Bereichs werden abgelehnt.

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

Beispiel: Vollständigen Zugriff auf einen Bucket zulassen, der ausschließlich von einem festgelegten föderierten Benutzer verwendet wird

In diesem Beispiel hat der föderierte Benutzer Alex vollen Zugriff auf den `examplebucket` Bucket und seine Objekte. Alle anderen Benutzer, einschließlich ‘root’, werden ausdrücklich allen Operationen verweigert. Beachten Sie jedoch, dass ‘root’ niemals die Berechtigungen zum Put/get/DeleteBucketPolicy verweigert wird.

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

Beispiel: PutOverwriteObject-Berechtigung

In diesem Beispiel stellt der `Deny` Effekt für PutOverwriteObject und DeleteObject sicher, dass niemand die Objektdaten, benutzerdefinierten Metadaten und S3-Objekt-Tagging überschreiben oder löschen kann.

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

Beispiel für Gruppenrichtlinien

Verwenden Sie die Beispiele in diesem Abschnitt, um StorageGRID-Zugriffsrichtlinien für Gruppen zu erstellen.

Gruppenrichtlinien legen die Zugriffsberechtigungen für die Gruppe fest, der die Richtlinie zugeordnet ist. Es gibt kein `Principal` Element in der Richtlinie, weil es implizit ist. Gruppenrichtlinien werden mit dem Tenant Manager oder der API konfiguriert.

Beispiel: Legen Sie eine Gruppenrichtlinie mit Tenant Manager fest

Wenn Sie eine Gruppe im Tenant Manager hinzufügen oder bearbeiten, können Sie eine Gruppenrichtlinie auswählen, um festzulegen, über welche S3-Zugriffsberechtigungen die Mitglieder dieser Gruppe verfügen. Siehe "Erstellen von Gruppen für einen S3-Mandanten".

- **Kein S3-Zugriff**: Standardoption. Benutzer in dieser Gruppe haben keinen Zugriff auf S3-Ressourcen, es sei denn, der Zugriff wird über eine Bucket-Richtlinie gewährt. Wenn Sie diese Option auswählen, hat nur der Root-Benutzer standardmäßig Zugriff auf S3-Ressourcen.
- **Schreibgeschützter Zugriff**: Benutzer in dieser Gruppe haben schreibgeschützten Zugriff auf S3-Ressourcen. Benutzer in dieser Gruppe können beispielsweise Objekte auflisten und Objektdaten, Metadaten und Tags lesen. Wenn Sie diese Option auswählen, wird im Textfeld der JSON-String für eine schreibgeschützte Gruppenrichtlinie angezeigt. Diese Zeichenfolge kann nicht bearbeitet werden.
- **Vollzugriff**: Benutzer in dieser Gruppe haben vollen Zugriff auf S3-Ressourcen, einschließlich Buckets. Wenn Sie diese Option auswählen, wird im Textfeld der JSON-String für eine Richtlinie mit vollem Zugriff angezeigt. Diese Zeichenfolge kann nicht bearbeitet werden.
- **Ransomware Mitigation**: Diese Beispielrichtlinie gilt für alle Buckets für diesen Mandanten. Benutzer in dieser Gruppe können allgemeine Aktionen ausführen, aber Objekte aus Buckets, für die die Objektversionierung aktiviert ist, nicht dauerhaft löschen.

  Mandanten-Manager-Benutzer mit der Berechtigung zum Verwalten aller Buckets können diese Gruppenrichtlinie überschreiben. Beschränken Sie die Berechtigung zum Verwalten aller Buckets auf vertrauenswürdige Benutzer und verwenden Sie die Multi-Faktor-Authentifizierung (MFA), sofern verfügbar.

- **Benutzerdefiniert**: Benutzern in der Gruppe werden die Berechtigungen erteilt, die Sie im Textfeld angeben.

Beispiel: Vollständigen Zugriff auf alle Buckets zulassen

In diesem Beispiel sind alle Mitglieder der Gruppe berechtigt, vollständigen Zugriff auf alle Buckets des Mandantenkontos zu erhalten, sofern nicht ausdrücklich von der Bucket-Richtlinie abgelehnt wurde.

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

Beispiel: Schreibgeschützter Zugriff auf alle Buckets für Gruppen zulassen

In diesem Beispiel haben alle Mitglieder der Gruppe schreibgeschützten Zugriff auf S3-Ressourcen, sofern nicht ausdrücklich von der Bucket-Richtlinie abgelehnt wird. Benutzer in dieser Gruppe können beispielsweise Objekte auflisten und Objektdaten, Metadaten und Tags lesen.

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

Beispiel: Gruppenmitgliedern vollen Zugriff nur auf ihren "Ordner" in einem Bucket erlauben

In diesem Beispiel dürfen Mitglieder der Gruppe nur ihren spezifischen Ordner (Schlüsselpräfix) im angegebenen Bucket auflisten und darauf zugreifen. Beachten Sie, dass bei der Festlegung der Privatsphäre dieser Ordner Zugriffsberechtigungen aus anderen Gruppenrichtlinien und der Bucket-Richtlinie berücksichtigt werden sollten.

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

S3-Vorgänge werden in den Audit-Protokollen protokolliert

Audit-Meldungen werden von StorageGRID-Diensten generiert und in Text-Log-Dateien gespeichert. Sie können die S3-spezifischen Audit-Meldungen im Revisionsprotokoll prüfen, um Details zu Bucket- und Objektvorgängen zu abrufen.

Bucket-Vorgänge werden in den Audit-Protokollen protokolliert

- CreateBucket
- DeleteBucket
- DeleteBucketTagging
- Objekte deObjekteObjekte
- GetBucketTagging
- HeadBucket
- ListObjekte
- ListObjectVersions
- BUCKET-Compliance
- PutBucketTagging
- PutBucketVersioning

Objektvorgänge werden in den Audit-Protokollen protokolliert

- CompleteMultipartUpload
- CopyObject
- DeleteObject
- GetObject
- HeadObject
- PutObject
- Objekt restoreObject
- Wählen Sie Objekt aus
- UploadPart (wenn eine ILM-Regel ausgeglichene oder strikte Aufnahme verwendet)
- UploadPartCopy (wenn eine ILM-Regel ausgeglichene oder strikte Aufnahme verwendet)

Verwandte Informationen

- "Zugriff auf die Audit-Log-Datei"
- "Audit-Meldungen des Clients schreiben"
- "Client liest Audit-Meldungen"

Copyright-Informationen

Copyright © 2025 NetApp. Alle Rechte vorbehalten. Gedruckt in den USA. Dieses urheberrechtlich geschützte Dokument darf ohne die vorherige schriftliche Genehmigung des Urheberrechtsinhabers in keiner Form und durch keine Mittel – weder grafische noch elektronische oder mechanische, einschließlich Fotokopieren, Aufnehmen oder Speichern in einem elektronischen Abrufsystem – auch nicht in Teilen, vervielfältigt werden.

Software, die von urheberrechtlich geschütztem NetApp Material abgeleitet wird, unterliegt der folgenden Lizenz und dem folgenden Haftungsausschluss:

DIE VORLIEGENDE SOFTWARE WIRD IN DER VORLIEGENDEN FORM VON NETAPP ZUR VERFÜGUNG GESTELLT, D. H. OHNE JEGLICHE EXPLIZITE ODER IMPLIZITE GEWÄHRLEISTUNG, EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE STILLSCHWEIGENDE GEWÄHRLEISTUNG DER MARKTGÄNGIGKEIT UND EIGNUNG FÜR EINEN BESTIMMTEN ZWECK, DIE HIERMIT AUSGESCHLOSSEN WERDEN. NETAPP ÜBERNIMMT KEINERLEI HAFTUNG FÜR DIREKTE, INDIREKTE, ZUFÄLLIGE, BESONDERE, BEISPIELHAFTE SCHÄDEN ODER FOLGESCHÄDEN (EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE BESCHAFFUNG VON ERSATZWAREN ODER -DIENSTLEISTUNGEN, NUTZUNGS-, DATEN- ODER GEWINNVERLUSTE ODER UNTERBRECHUNG DES GESCHÄFTSBETRIEBS), UNABHÄNGIG DAVON, WIE SIE VERURSACHT WURDEN UND AUF WELCHER HAFTUNGSTHEORIE SIE BERUHEN, OB AUS VERTRAGLICH FESTGELEGTER HAFTUNG, VERSCHULDENSUNABHÄNGIGER HAFTUNG ODER DELIKTSHAFTUNG (EINSCHLIESSLICH FAHRLÄSSIGKEIT ODER AUF ANDEREM WEGE), DIE IN IRGENDEINER WEISE AUS DER NUTZUNG DIESER SOFTWARE RESULTIEREN, SELBST WENN AUF DIE MÖGLICHKEIT DERARTIGER SCHÄDEN HINGEWIESEN WURDE.

NetApp behält sich das Recht vor, die hierin beschriebenen Produkte jederzeit und ohne Vorankündigung zu ändern. NetApp übernimmt keine Verantwortung oder Haftung, die sich aus der Verwendung der hier beschriebenen Produkte ergibt, es sei denn, NetApp hat dem ausdrücklich in schriftlicher Form zugestimmt. Die Verwendung oder der Erwerb dieses Produkts stellt keine Lizenzierung im Rahmen eines Patentrechts, Markenrechts oder eines anderen Rechts an geistigem Eigentum von NetApp dar.

Das in diesem Dokument beschriebene Produkt kann durch ein oder mehrere US-amerikanische Patente, ausländische Patente oder anhängige Patentanmeldungen geschützt sein.

ERLÄUTERUNG ZU „RESTRICTED RIGHTS“: Nutzung, Vervielfältigung oder Offenlegung durch die US-Regierung unterliegt den Einschränkungen gemäß Unterabschnitt (b)(3) der Klausel „Rights in Technical Data – Noncommercial Items“ in DFARS 252.227-7013 (Februar 2014) und FAR 52.227-19 (Dezember 2007).

Die hierin enthaltenen Daten beziehen sich auf ein kommerzielles Produkt und/oder einen kommerziellen Service (wie in FAR 2.101 definiert) und sind Eigentum von NetApp, Inc. Alle technischen Daten und die Computersoftware von NetApp, die unter diesem Vertrag bereitgestellt werden, sind gewerblicher Natur und wurden ausschließlich unter Verwendung privater Mittel entwickelt. Die US-Regierung besitzt eine nicht ausschließliche, nicht übertragbare, nicht unterlizenzierbare, weltweite, limitierte unwiderrufliche Lizenz zur Nutzung der Daten nur in Verbindung mit und zur Unterstützung des Vertrags der US-Regierung, unter dem die Daten bereitgestellt wurden. Sofern in den vorliegenden Bedingungen nicht anders angegeben, dürfen die Daten ohne vorherige schriftliche Genehmigung von NetApp, Inc. nicht verwendet, offengelegt, vervielfältigt, geändert, aufgeführt oder angezeigt werden. Die Lizenzrechte der US-Regierung für das US-Verteidigungsministerium sind auf die in DFARS-Klausel 252.227-7015(b) (Februar 2014) genannten Rechte beschränkt.

Markeninformationen

NETAPP, das NETAPP Logo und die unter http://www.netapp.com/TM aufgeführten Marken sind Marken von NetApp, Inc. Andere Firmen und Produktnamen können Marken der jeweiligen Eigentümer sein.