



# **S3 REST API**を使用する StorageGRID software

NetApp  
December 03, 2025

# 目次

S3 REST APIを使用する	1
S3 REST API のサポート対象バージョンとアップデート	1
サポート対象のバージョン	1
S3 REST API サポートの更新	1
クイックリファレンス: サポートされている S3 API リクエスト	4
一般的なURIクエリパラメータとリクエストヘッダー	4
"AbortMultipartUpload"	5
"CompleteMultipartUpload"	5
"CopyObject"	6
"CreateBucket"	7
"CreateMultipartUpload"	7
>DeleteBucket"	8
"バケットCorsの削除"	8
"バケット暗号化の削除"	8
>DeleteBucketLifecycle"	8
>DeleteBucketPolicy"	9
"バケットレプリケーションの削除"	9
"バケットタグの削除"	9
>DeleteObject"	9
>DeleteObjects"	10
>DeleteObjectTagging"	10
"GetBucketAcl"	10
"GetBucketCors"	10
"バケット暗号化を取得"	11
"GetBucketLifecycleConfiguration"	11
"GetBucketLocation"	11
"GetBucketNotificationConfiguration"	11
"GetBucketPolicy"	12
"GetBucketReplication"	12
"GetBucketTagging"	12
"GetBucketVersioning"	12
"GetObject"	12
"GetObjectAcl"	13
"GetObjectLegalHold"	14
"GetObjectLockConfiguration"	14
"GetObjectRetention"	14
"GetObjectTagging"	14
"HeadBucket"	15
"HeadObject"	15
>ListBuckets"	15
"リストマルチパートアップロード"	16

"ListObjects" .....	16
"リストオブジェクトV2" .....	16
"ListObjectVersions" .....	17
"ListParts" .....	17
"PutBucketCors" .....	18
"PutBucketEncryption" .....	18
"PutBucketLifecycleConfiguration" .....	18
"PutBucketNotificationConfiguration" .....	19
"PutBucketPolicy" .....	20
"PutBucketレプリケーション" .....	20
"PutBucketTagging" .....	20
"PutBucketVersioning" .....	20
"PutObject" .....	21
"PutObjectLegalHold" .....	22
"PutObjectLockConfiguration" .....	22
"PutObjectRetention" .....	22
"PutObjectTagging" .....	22
"オブジェクトの復元" .....	23
"オブジェクトコンテンツの選択" .....	23
"UploadPart" .....	23
"UploadPartCopy" .....	24
S3 REST API 構成のテスト .....	24
StorageGRIDがS3 REST APIを実装する方法 .....	26
競合するクライアント要求 .....	26
一貫性の値 .....	26
オブジェクトのバージョン管理 .....	29
S3 REST API を使用して S3 オブジェクトロックを設定する .....	30
S3ライフサイクル設定を作成する .....	35
S3 REST API 実装に関する推奨事項 .....	40
Amazon S3 REST APIのサポート .....	41
S3 REST API実装の詳細 .....	41
リクエストを認証する .....	42
サービス上の操作 .....	43
バケットの操作 .....	43
オブジェクトに対する操作 .....	51
マルチパートアップロードの操作 .....	80
エラー応答 .....	89
StorageGRIDカスタム操作 .....	91
StorageGRIDカスタム操作 .....	91
GETバケットの一貫性 .....	92
PUTバケットの一貫性 .....	94
GETバケットの最終アクセス時刻 .....	95
バケットの最終アクセス時刻をPUTする .....	95

バケットメタデータ通知設定の削除	97
バケットメタデータ通知設定の取得	97
PUT バケットメタデータ通知設定	100
GETストレージ使用量リクエスト	106
従来のコンプライアンスに対するバケット リクエストのサポート終了	107
バケットとグループのアクセスポリシー	112
バケットとグループのアクセスポリシーを使用する	112
バケットポリシーの例	130
グループポリシーの例	136
監査ログで追跡されるS3操作	139
監査ログで追跡されるバケット操作	139
監査ログで追跡されるオブジェクト操作	140

# S3 REST APIを使用する

## S3 REST API のサポート対象バージョンとアップデート

StorageGRID は、Representational State Transfer (REST) Web サービスのセットとして実装される Simple Storage Service (S3) API をサポートしています。

S3 REST API のサポートにより、S3 Web サービス用に開発されたサービス指向アプリケーションを、StorageGRIDシステムを使用するオンプレミスのオブジェクトストレージに接続できるようになります。クライアント アプリケーションの現在の S3 REST API 呼び出しの使用に対する最小限の変更が必要です。

### サポート対象のバージョン

StorageGRID は、次の特定のバージョンの S3 と HTTP をサポートしています。

項目	version
S3 API仕様	<a href="#">"Amazon Web Services (AWS) ドキュメント: Amazon Simple Storage Service API リファレンス"</a>
HTTP	1.1  HTTP の詳細については、HTTP/1.1 (RFC 7230-35) を参照してください。  <a href="#">"IETF RFC 2616: ハイパーテキスト転送プロトコル (HTTP/1.1)"</a>  注意: StorageGRID はHTTP/1.1 パイプラインをサポートしていません。

### S3 REST API サポートの更新

リリース	コメント
11.9	<ul style="list-style-type: none"> <li>• 次のリクエストとサポートされているヘッダーに対して、事前計算された SHA-256 チェックサム値のサポートが追加されました。この機能を使用すると、アップロードされたオブジェクトの整合性を検証できます。 <ul style="list-style-type: none"> <li>◦ 完全なマルチパートアップロード: <code>x-amz-checksum-sha256</code></li> <li>◦ マルチパートアップロードの作成: <code>x-amz-checksum-algorithm</code></li> <li>◦ オブジェクトを取得: <code>x-amz-checksum-mode</code></li> <li>◦ ヘッドオブジェクト: <code>x-amz-checksum-mode</code></li> <li>◦ ListParts</li> <li>◦ オブジェクトを配置: <code>x-amz-checksum-sha256</code></li> <li>◦ アップロードパーツ: <code>x-amz-checksum-sha256</code></li> </ul> </li> <li>• グリッド管理者がテナントレベルの保持とコンプライアンス設定を制御できるようになりました。これらの設定は S3 オブジェクトロックの設定に影響します。 <ul style="list-style-type: none"> <li>◦ バケットのデフォルトの保持モードとオブジェクト保持モード: グリッド管理者が許可している場合、ガバナンスまたはコンプライアンス。</li> <li>◦ バケットのデフォルトの保持期間とオブジェクトの保持期限: グリッド管理者が設定した最大保持期間で許可されている値以下である必要があります。</li> </ul> </li> <li>• サポートの改善 `aws-chunked` コンテンツのエンコーディングとストリーミング `x-amz-content-sha256` 値値観。制限事項: <ul style="list-style-type: none"> <li>◦ 存在する場合、`chunk-signature` オプションであり検証されません</li> <li>◦ 存在する場合、`x-amz-trailer` コンテンツは無視されます</li> </ul> </li> </ul>
11.8	<p>S3オペレーションの名前を、<a href="#">"Amazon Web Services (AWS) ドキュメント: Amazon Simple Storage Service API リファレンス"</a>。</p>
11.7	<ul style="list-style-type: none"> <li>• 追加した<a href="#">"クイックリファレンス: サポートされている S3 API リクエスト"</a>。</li> <li>• S3 オブジェクトロックで GOVERNANCE モードを使用するためのサポートが追加されました。</li> <li>• StorageGRID固有のサポートを追加 `x-ntap-sg-cgr-replication-status` GET Object および HEAD Object リクエストの応答ヘッダー。このヘッダーは、クロスグリッドレプリケーションのオブジェクトのレプリケーションステータスを提供します。</li> <li>• SelectObjectContent リクエストは Parquet オブジェクトをサポートするようになりました。</li> </ul>

リリース	コメント
11.6	<ul style="list-style-type: none"> <li>• 使用するためのサポートを追加しました `partNumber` GET Object リクエストと HEAD Object リクエストのリクエスト パラメータ。</li> <li>• S3 オブジェクト ロックのバケット レベルでのデフォルトの保持モードとデフォルトの保持期間のサポートが追加されました。</li> <li>• のサポートを追加しました `s3:object-lock-remaining-retention-days` オブジェクトの許容保持期間の範囲を設定するためのポリシー条件キー。</li> <li>• 単一の PUT Object 操作の最大 推奨 サイズを 5 GiB (5,368,709,120 バイト) に変更しました。5 GiB を超えるオブジェクトがある場合は、代わりにマルチパートアップロードを使用します。</li> </ul>
11.5	<ul style="list-style-type: none"> <li>• バケット暗号化の管理のサポートが追加されました。</li> <li>• S3 オブジェクト ロックと廃止された従来のコンプライアンス リクエストのサポートが追加されました。</li> <li>• バージョン管理されたバケットで DELETE Multiple Objects を使用するためのサポートが追加されました。</li> <li>• その `Content-MD5` リクエスト ヘッダーが正しくサポートされるようになりました。</li> </ul>
11.4	<ul style="list-style-type: none"> <li>• DELETE バケット タグ付け、GET バケット タグ付け、PUT バケット タグ付けのサポートが追加されました。コスト配分タグはサポートされていません。</li> <li>• StorageGRID 11.4 で作成されたバケットの場合、パフォーマンスのベスト プラクティスを満たすためにオブジェクト キー名を制限する必要がなくなりました。</li> <li>• バケット通知のサポートを追加しました `s3:ObjectRestore:Post` イベントの種類。</li> <li>• マルチパートパーツに対する AWS サイズ制限が適用されるようになりました。マルチパートアップロードの各パートは 5 MiB ~ 5 GiB の範囲でなければなりません。最後の部分は 5 MiB より小さくても構いません。</li> <li>• TLS 1.3のサポートを追加しました</li> </ul>
11.3	<ul style="list-style-type: none"> <li>• 顧客提供のキーを使用したオブジェクト データのサーバー側暗号化 (SSE-C) のサポートが追加されました。</li> <li>• DELETE、GET、PUTバケットライフサイクル操作（有効期限アクションのみ）のサポートが追加されました。`x-amz-expiration`レスポンス ヘッダー。</li> <li>• 取り込み時に同期配置を使用する ILM ルールの影響を説明するために、PUT オブジェクト、PUT オブジェクト - コピー、およびマルチパートアップロードを更新しました。</li> <li>• TLS 1.1 暗号はサポートされなくなりました。</li> </ul>

リリース	コメント
11.2	Cloud Storage Pools で使用するための POST Object restore のサポートが追加されました。グループおよびバケットポリシーの ARN、ポリシー条件キー、およびポリシー変数に AWS 構文を使用するためのサポートが追加されました。StorageGRID構文を使用する既存のグループおよびバケット ポリシーは引き続きサポートされます。  注: カスタムStorageGRID機能で使用されるものも含め、他の構成 JSON/XML での ARN/URN の使用は変更されていません。
11.1	クロスオリジン リソース共有 (CORS)、グリッド ノードへの S3 クライアント接続用の HTTP、バケットのコンプライアンス設定のサポートが追加されました。
11.0	バケットのプラットフォーム サービス (CloudMirror レプリケーション、通知、Elasticsearch 検索統合) を構成するためのサポートが追加されました。また、バケットのオブジェクトタグ付け場所制約と、利用可能な一貫性のサポートも追加されました。
10.4	ILM スキャンのバージョン管理の変更、エンドポイント ドメイン名ページの更新、ポリシーの条件と変数、ポリシーの例、および PutOverwriteObject 権限のサポートが追加されました。
10.3	バージョン管理のサポートが追加されました。
10.2	グループおよびバケット アクセス ポリシー、およびマルチパート コピー (アップロード部分 - コピー) のサポートが追加されました。
10.1	マルチパートアップロード、仮想ホストスタイルのリクエスト、v4 認証のサポートが追加されました。
10.0	StorageGRIDシステムによる S3 REST API の初期サポート。現在サポートされている <i>Simple Storage Service API Reference</i> のバージョンは 2006-03-01 です。

## クイックリファレンス: サポートされている **S3 API** リクエスト

このページでは、StorageGRID が Amazon Simple Storage Service (S3) API をサポートする方法についてまとめています。

このページには、StorageGRIDでサポートされている S3 操作のみが含まれています。



各操作の AWS ドキュメントを表示するには、見出しのリンクを選択します。

### 一般的なURIクエリパラメータとリクエストヘッダー

特に記載がない限り、次の一般的な URI クエリ パラメータがサポートされています。

- versionId (オブジェクト操作に必要な場合)

特に記載がない限り、次の一般的なリクエスト ヘッダーがサポートされています。

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

関連情報

- ["S3 REST API実装の詳細"](#)
- ["Amazon Simple Storage Service API リファレンス: 共通リクエストヘッダー"](#)

## "AbortMultipartUpload"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#)このリクエストには、次の追加の URI クエリ パラメータが含まれます。

- uploadId

リクエスト本文

なし

StorageGRID ドキュメント

["マルチパートアップロードの操作"](#)

## "CompleteMultipartUpload"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#)このリクエストには、次の追加の URI クエリ パラメータが含まれます。

- uploadId
- x-amz-checksum-sha256

リクエストボディのXMLタグ

StorageGRID は、次のリクエスト ボディ XML タグをサポートしています。

- ChecksumSHA256
- CompleteMultipartUpload

- ETag
- Part
- PartNumber

## StorageGRID ドキュメント

### "CompleteMultipartUpload"

## "CopyObject"

### URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加ヘッダーが含まれます。

- 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>`

### リクエスト本文

なし

## StorageGRID ドキュメント

### "CopyObject"

## "CreateBucket"

### URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加ヘッダーが含まれます。

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

### リクエスト本文

StorageGRID は、実装時に Amazon S3 REST API によって定義されたすべてのリクエスト ボディ パラメータをサポートします。

### StorageGRIDドキュメント

["バケットの操作"](#)

## "CreateMultipartUpload"

### URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加ヘッダーが含まれます。

- 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>`

### リクエスト本文

なし

### StorageGRIDドキュメント

"CreateMultipartUpload"

## "DeleteBucket"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

StorageGRIDドキュメント

["バケットの操作"](#)

## "バケットCorsの削除"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["バケットの操作"](#)

## "バケット暗号化の削除"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["バケットの操作"](#)

## "DeleteBucketLifecycle"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

- ["バケットの操作"](#)
- ["S3ライフサイクル設定を作成する"](#)

## "DeleteBucketPolicy"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

**StorageGRID**ドキュメント

["バケットの操作"](#)

## "バケットレプリケーションの削除"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

**StorageGRID**ドキュメント

["バケットの操作"](#)

## "バケットタグの削除"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

**StorageGRID**ドキュメント

["バケットの操作"](#)

## "DeleteObject"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加のリクエストヘッダーが含まれます。

- x-amz-bypass-governance-retention

リクエスト本文

なし

**StorageGRID**ドキュメント

["オブジェクトに対する操作"](#)

## "DeleteObjects"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加のリクエストヘッダーが含まれます。

- x-amz-bypass-governance-retention

リクエスト本文

StorageGRID は、実装時に Amazon S3 REST API によって定義されたすべてのリクエスト ボディ パラメータをサポートします。

StorageGRIDドキュメント

["オブジェクトに対する操作"](#)

## "DeleteObjectTagging"

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["オブジェクトに対する操作"](#)

## "GetBucketAcl"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["バケットの操作"](#)

## "GetBucketCors"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["バケットの操作"](#)

## "バケット暗号化を取得"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート **共通パラメータとヘッダー**このリクエストに対して。

リクエスト本文

なし

**StorageGRID**ドキュメント

["バケットの操作"](#)

## "GetBucketLifecycleConfiguration"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート **共通パラメータとヘッダー**このリクエストに対して。

リクエスト本文

なし

**StorageGRID**ドキュメント

- ["バケットの操作"](#)
- ["S3ライフサイクル設定を作成する"](#)

## "GetBucketLocation"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート **共通パラメータとヘッダー**このリクエストに対して。

リクエスト本文

なし

**StorageGRID**ドキュメント

["バケットの操作"](#)

## "GetBucketNotificationConfiguration"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート **共通パラメータとヘッダー**このリクエストに対して。

リクエスト本文

なし

**StorageGRID**ドキュメント

["バケットの操作"](#)

## "GetBucketPolicy"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["バケットの操作"](#)

## "GetBucketReplication"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["バケットの操作"](#)

## "GetBucketTagging"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["バケットの操作"](#)

## "GetBucketVersioning"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["バケットの操作"](#)

## "GetObject"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加の URI クエリ パ

ラメータが含まれます。

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

さらに次の追加のリクエスト ヘッダーがあります。

- 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

リクエスト本文

なし

**StorageGRID** ドキュメント

["GetObject"](#)

## "GetObjectAcl"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

**StorageGRID** ドキュメント

["オブジェクトに対する操作"](#)

## "GetObjectLegalHold"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["S3 REST API を使用して S3 オブジェクトロックを設定する"](#)

## "GetObjectLockConfiguration"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["S3 REST API を使用して S3 オブジェクトロックを設定する"](#)

## "GetObjectRetention"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["S3 REST API を使用して S3 オブジェクトロックを設定する"](#)

## "GetObjectTagging"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["オブジェクトに対する操作"](#)

## "HeadBucket"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

["バケットの操作"](#)

## "HeadObject"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加ヘッダーが含まれます。

- 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

リクエスト本文

なし

StorageGRIDドキュメント

["HeadObject"](#)

## "ListBuckets"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

なし

StorageGRIDドキュメント

[サービス上の操作](#) > [ListBuckets](#)

## "リストマルチパートアップロード"

### URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加パラメータが含まれます。

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

リクエスト本文

なし

### StorageGRIDドキュメント

["リストマルチパートアップロード"](#)

## "ListObjects"

### URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加パラメータが含まれます。

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

リクエスト本文

なし

### StorageGRIDドキュメント

["バケットの操作"](#)

## "リストオブジェクトV2"

### URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加パラメータが含まれます。

- continuation-token
- delimiter

- encoding-type
- fetch-owner
- max-keys
- prefix
- start-after

リクエスト本文

なし

**StorageGRID**ドキュメント

["バケットの操作"](#)

## "ListObjectVersions"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加パラメータが含まれます。

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

リクエスト本文

なし

**StorageGRID**ドキュメント

["バケットの操作"](#)

## "ListParts"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加パラメータが含まれます。

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

リクエスト本文

なし

StorageGRID ドキュメント

["リストマルチパートアップロード"](#)

## "PutBucketCors"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

StorageGRID は、実装時に Amazon S3 REST API によって定義されたすべてのリクエスト ボディ パラメータをサポートします。

StorageGRID ドキュメント

["バケットの操作"](#)

## "PutBucketEncryption"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエストボディのXMLタグ

StorageGRID は、次のリクエスト ボディ XML タグをサポートしています。

- ApplyServerSideEncryptionByDefault
- Rule
- ServerSideEncryptionConfiguration
- SSEAlgorithm

StorageGRID ドキュメント

["バケットの操作"](#)

## "PutBucketLifecycleConfiguration"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエストボディのXMLタグ

StorageGRID は、次のリクエスト ボディ XML タグをサポートしています。

- And
- Days
- Expiration
- ExpiredObjectDeleteMarker
- Filter

- ID
- Key
- LifecycleConfiguration
- NewerNoncurrentVersions
- NoncurrentDays
- NoncurrentVersionExpiration
- Prefix
- Rule
- Status
- Tag
- Value

#### StorageGRID ドキュメント

- ["バケットの操作"](#)
- ["S3ライフサイクル設定を作成する"](#)

## "PutBucketNotificationConfiguration"

#### URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

#### リクエストボディのXMLタグ

StorageGRID は、次のリクエスト ボディ XML タグをサポートしています。

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

#### StorageGRID ドキュメント

["バケットの操作"](#)

## "PutBucketPolicy"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

サポートされているJSON本体フィールドの詳細については、以下を参照してください。 ["バケットとグループのアクセスポリシーを使用する"](#)。

## "PutBucketレプリケーション"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエストボディの**XML**タグ

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

**StorageGRID** ドキュメント

["バケットの操作"](#)

## "PutBucketTagging"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

StorageGRID は、実装時に Amazon S3 REST API によって定義されたすべてのリクエスト ボディ パラメータをサポートします。

**StorageGRID** ドキュメント

["バケットの操作"](#)

## "PutBucketVersioning"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエストボディパラメータ

StorageGRID は次のリクエスト本体パラメータをサポートしています。

- VersioningConfiguration
- Status

**StorageGRID**ドキュメント

["バケットの操作"](#)

## "PutObject"

**URI**クエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#)このリクエストには、次の追加ヘッダーが含まれます。

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

リクエスト本文

- オブジェクトのバイナリデータ

**StorageGRID**ドキュメント

["PutObject"](#)

## "PutObjectLegalHold"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

StorageGRID は、実装時に Amazon S3 REST API によって定義されたすべてのリクエスト ボディ パラメータをサポートします。

**StorageGRID** ドキュメント

["S3 REST API を使用して S3 オブジェクトロックを設定する"](#)

## "PutObjectLockConfiguration"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

StorageGRID は、実装時に Amazon S3 REST API によって定義されたすべてのリクエスト ボディ パラメータをサポートします。

**StorageGRID** ドキュメント

["S3 REST API を使用して S3 オブジェクトロックを設定する"](#)

## "PutObjectRetention"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには次の追加ヘッダーが含まれません:

- x-amz-bypass-governance-retention

リクエスト本文

StorageGRID は、実装時に Amazon S3 REST API によって定義されたすべてのリクエスト ボディ パラメータをサポートします。

**StorageGRID** ドキュメント

["S3 REST API を使用して S3 オブジェクトロックを設定する"](#)

## "PutObjectTagging"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

リクエスト本文

StorageGRID は、実装時に Amazon S3 REST API によって定義されたすべてのリクエスト ボディ パラメータをサポートします。

**StorageGRID** ドキュメント

## "オブジェクトに対する操作"

### "オブジェクトの復元"

#### URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

#### リクエスト本文

サポートされている本文フィールドの詳細については、["オブジェクトの復元"](#)。

### "オブジェクトコンテンツの選択"

#### URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストに対して。

#### リクエスト本文

サポートされている本文フィールドの詳細については、以下を参照してください。

- ["S3 Selectを使用する"](#)
- ["オブジェクトコンテンツの選択"](#)

### "UploadPart"

#### URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加の URI クエリ パラメータが含まれます。

- `partNumber`
- `uploadId`

さらに次の追加のリクエスト ヘッダーがあります。

- `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`

#### リクエスト本文

- 部品のバイナリデータ

#### StorageGRIDドキュメント

##### ["UploadPart"](#)

## "UploadPartCopy"

URIクエリパラメータとリクエストヘッダー

StorageGRIDはすべてをサポート [共通パラメータとヘッダー](#) このリクエストには、次の追加の URI クエリ パラメータが含まれます。

- partNumber
- uploadId

さらに次の追加のリクエスト ヘッダーがあります。

- 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

リクエスト本文

なし

StorageGRIDドキュメント

["UploadPartCopy"](#)

## S3 REST API 構成のテスト

Amazon Web Services コマンドラインインターフェイス (AWS CLI) を使用して、システムへの接続をテストし、オブジェクトの読み取りと書き込みができることを確認できます。

開始する前に

- AWS CLIをダウンロードしてインストールしました。 ["aws.amazon.com/cli"](https://aws.amazon.com/cli) 。
- オプションとして、["ロードバランサエンドポイントを作成した"](#)。それ以外の場合は、接続するストレージノードの IP アドレスと使用するポート番号がわかっています。見る ["クライアント接続用のIPアドレスとポート"](#)。
- あなたが持っている ["S3テナントアカウントを作成しました"](#)。

- テナントにサインインし、"アクセスキーを作成しました"。

これらの手順の詳細については、"クライアント接続を構成する"。

#### 手順

1. StorageGRIDシステムで作成したアカウントを使用するように AWS CLI 設定を構成します。
  - a. 設定モードに入ります: `aws configure`
  - b. 作成したアカウントのアクセス キー ID を入力します。
  - c. 作成したアカウントのシークレットアクセスキーを入力します。
  - d. 使用するデフォルトのリージョンを入力します。例: `us-east-1`。
  - e. 使用するデフォルトの出力形式を入力するか、**Enter** キーを押して JSON を選択します。
2. バケットを作成

この例では、IP アドレス 10.96.101.17 とポート 10443 を使用するようにロード バランサー エンドポイントを構成していることを前提としています。

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

バケットが正常に作成されると、次の例に示すように、バケットの場所が返されます。

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

3. オブジェクトをアップロードします。

```
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
```

オブジェクトが正常にアップロードされると、オブジェクト データのハッシュである Etag が返されま

4. バケットの内容を一覧表示して、オブジェクトがアップロードされたことを確認します。

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

5. オブジェクトを削除します。

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

6. バケットを削除します。

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

## StorageGRIDがS3 REST APIを実装する方法

### 競合するクライアント要求

2つのクライアントが同じキーに書き込むなど、競合するクライアント要求は、「最新のもの優先」して解決されます。

「最新のもの優先」評価のタイミングは、S3クライアントが操作を開始した時点ではなく、StorageGRIDシステムが特定のリクエストを完了した時点に基づきます。

### 一貫性の値

一貫性は、オブジェクトの可用性と、異なるストレージノードおよびサイト間でのオブジェクトの一貫性との間のバランスを提供します。アプリケーションの必要に応じて一貫性を変更できます。

デフォルトでは、StorageGRIDは新しく作成されたオブジェクトの書き込み後の読み取り一貫性を保証します。正常に完了したPUTに続くGETでは、新しく書き込まれたデータを読み取ることができます。既存のオブジェクトの上書き、メタデータの更新、および削除は、最終的に一貫性が保たれます。上書きが反映されるまでには通常数秒または数分かかりますが、最大15日かかる場合があります。

異なる一貫性でオブジェクト操作を実行する場合は、次の操作を実行できます。

- 一貫性を指定する [各バケツ](#)。
- 一貫性を指定する [各API操作](#)。
- 次のいずれかのタスクを実行して、グリッド全体のデフォルトの一貫性を変更します。
  - グリッドマネージャーで、構成 > システム > ストレージ設定 > デフォルトの一貫性 に移動します。
  - 。



グリッド全体の一貫性の変更は、設定の変更後に作成されたバケットにのみ適用されません。変更の詳細を確認するには、次の場所にある監査ログを参照してください。  
`/var/local/log (*consistencyLevel*`を検索してください)。

## 一貫性の値

一貫性は、StorageGRID がオブジェクトを追跡するために使用するメタデータがノード間でどのように分散されるかに影響し、したがってクライアント要求に対するオブジェクトの可用性に影響します。

バケットまたは API 操作の一貫性を次のいずれかの値に設定できます。

- **すべて:** すべてのノードがデータを直ちに受信します。そうでない場合、要求は失敗します。
- **強力なグローバル:** すべてのサイトにわたるすべてのクライアント要求の書き込み後の読み取り一貫性を保証します。
- **強力なサイト:** サイト内のすべてのクライアント要求に対して、書き込み後の読み取りの一貫性を保証します。
- **新規書き込み後の読み取り:** (デフォルト) 新しいオブジェクトに対して書き込み後の読み取りの一貫性を提供し、オブジェクトの更新に対して最終的な一貫性を提供します。高可用性とデータ保護の保証を提供します。ほとんどの場合に推奨されます。
- **利用可能:** 新しいオブジェクトとオブジェクトの更新の両方に対して最終的な一貫性を提供します。S3 バケットの場合は、必要な場合にのみ使用してください (たとえば、めったに読み取られないログ値を含むバケットの場合や、存在しないキーに対する HEAD または GET 操作の場合など)。S3 FabricPool バケットではサポートされていません。

「新規書き込み後の読み取り」と「利用可能」の一貫性を使用する

HEAD または GET 操作で「Read-after-new-write」一貫性を使用する場合、StorageGRID は次のように複数のステップで検索を実行します。

- まず、低い一貫性を使用してオブジェクトを検索します。
- その検索が失敗した場合、strong-global の動作と同等の一貫性に達するまで、次の一貫性値での検索を繰り返します。

HEAD または GET 操作で「新規書き込み後の読み取り」一貫性が使用され、オブジェクトが存在しない場合は、オブジェクトの検索は常に strong-global の動作と同等の一貫性に到達します。この一貫性を保つには、各サイトでオブジェクト メタデータの複数のコピーが使用可能である必要があるため、同じサイトにある 2 つ以上のストレージ ノードが使用できない場合、多数の 500 内部サーバー エラーが発生する可能性があります。

Amazon S3 と同様の一貫性の保証が必要ない場合は、一貫性を「利用可能」に設定することで、HEAD および GET 操作でこれらのエラーを防ぐことができます。HEAD または GET 操作で「使用可能」な一貫性が使用される場合、StorageGRID は最終的な一貫性のみを提供します。一貫性が増すにつれて失敗した操作を再試行することはないため、オブジェクト メタデータの複数のコピーが利用可能である必要はありません。

## API操作の一貫性を指定する

個々の API 操作の一貫性を設定するには、その操作に対して一貫性の値がサポートされている必要があり、リクエスト ヘッダーで一貫性を指定する必要があります。この例では、GetObject 操作の一貫性を「Strong-site」に設定します。

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



PutObject 操作と GetObject 操作の両方に同じ一貫性を使用する必要があります。

### バケットの一貫性を指定する

バケットの一貫性を設定するには、StorageGRIDを使用します。**"PUT バケットの一貫性"**リクエスト。あるいは**"バケットの一貫性を変更する"**テナントマネージャーより。

バケットの一貫性を設定するときは、次の点に注意してください。

- バケットの一貫性を設定すると、バケット内のオブジェクトまたはバケット設定に対して実行される S3 操作に使用される一貫性が決まります。バケット自体の操作には影響しません。
- 個々の API 操作の一貫性は、バケットの一貫性よりも優先されます。
- 一般に、バケットではデフォルトの一貫性である「新規書き込み後の読み取り」を使用する必要があります。リクエストが正しく機能しない場合は、可能であればアプリケーション クライアントの動作を変更します。または、各 API リクエストの一貫性を指定するようにクライアントを構成します。最後の手段としてのみ、バケット レベルで一貫性を設定します。

### 一貫性とILMルールがどのように相互作用してデータ保護に影響を与えるか

一貫性の選択と ILM ルールの両方が、オブジェクトの保護方法に影響します。これらの設定は相互作用する可能性があります。

たとえば、オブジェクトの保存時に使用される一貫性は、オブジェクト メタデータの初期配置に影響し、ILM ルールに選択された取り込み動作は、オブジェクト コピーの初期配置に影響します。StorageGRID はクライアントの要求を満たすためにオブジェクトのメタデータとそのデータの両方にアクセスするため、一貫性と取り込み動作に一致する保護レベルを選択すると、初期データ保護が向上し、システム応答がより予測可能になります。

次の**"取り込みオプション"**ILM ルールに使用できるもの:

#### Dual commit

StorageGRID は直ちにオブジェクトの中間コピーを作成し、クライアントに成功を返します。ILM ルールで指定されたコピーは、可能な場合に作成されます。

#### 厳しい

クライアントに成功が返される前に、ILM ルールで指定されたすべてのコピーを作成する必要があります。

#### バランスの取れた

StorageGRID は、取り込み時に ILM ルールで指定されたすべてのコピーを作成しようとします。これが不可能な場合は、中間コピーが作成され、クライアントに成功が返されます。ILM ルールで指定されたコピーは、可能な場合に作成されます。

一貫性とILMルールがどのように相互作用するかの例

次の ILM ルールと次の一貫性を持つ 2 つのサイト グリッドがあるとします。

- **ILM ルール:** ローカル サイトとリモート サイトに 1 つずつ、合計 2 つのオブジェクト コピーを作成します。厳密な取り込み動作を使用します。
- **一貫性:** 強力なグローバル (オブジェクト メタデータはすべてのサイトに直ちに配布されます)。

クライアントがオブジェクトをグリッドに保存すると、StorageGRID は両方のオブジェクトのコピーを作成し、両方のサイトにメタデータを配布してから、クライアントに成功を返します。

オブジェクトは、取り込み成功メッセージの時点で損失から完全に保護されます。たとえば、取り込み直後にローカル サイトが失われた場合でも、オブジェクト データとオブジェクト メタデータの両方のコピーがリモート サイトに残ります。オブジェクトは完全に取得可能です。

代わりに同じ ILM ルールと強力なサイト一貫性を使用した場合、オブジェクト データがリモート サイトにレプリケートされた後、オブジェクト メタデータがそこに配布される前に、クライアントは成功メッセージを受信する可能性があります。この場合、オブジェクト メタデータの保護レベルは、オブジェクト データの保護レベルと一致しません。取り込み直後にローカル サイトが失われた場合、オブジェクト メタデータは失われます。オブジェクトを取得できません。

一貫性と ILM ルール間の相互関係は複雑になる可能性があります。サポートが必要な場合は、NetAppにお問い合わせください。

## オブジェクトのバージョン管理

各オブジェクトの複数のバージョンを保持する場合は、バケットのバージョン管理状態を設定できます。バケットのバージョン管理を有効にすると、オブジェクトが誤って削除されるのを防ぎ、オブジェクトの以前のバージョンを取得して復元できるようになります。

StorageGRIDシステムは、ほとんどの機能をサポートするバージョン管理を実装していますが、いくつかの制限があります。StorageGRID は、各オブジェクトの最大 10,000 バージョンをサポートします。

オブジェクトのバージョン管理は、StorageGRID情報ライフサイクル管理 (ILM) または S3 バケット ライフサイクル構成と組み合わせることができます。各バケットに対してバージョン管理を明示的に有効にする必要があります。バケットのバージョン管理が有効になっている場合、バケットに追加された各オブジェクトには、StorageGRIDシステムによって生成されるバージョン ID が割り当てられます。

MFA (多要素認証) を使用した削除はサポートされていません。



バージョン管理は、StorageGRIDバージョン 10.3 以降で作成されたバケットでのみ有効にできます。

## ILMとバージョン管理

ILM ポリシーはオブジェクトの各バージョンに適用されます。ILM スキャン プロセスは、すべてのオブジェクトを継続的にスキャンし、現在の ILM ポリシーに対して再評価します。ILM ポリシーに加えた変更は、以前に取り込まれたすべてのオブジェクトに適用されます。バージョン管理が有効になっている場合は、以前に取り込まれたバージョンも含まれます。ILM スキャンは、以前に取り込まれたオブジェクトに新しい ILM 変更を適用します。

バージョン管理が有効なバケット内のS3オブジェクトの場合、バージョン管理のサポートにより、「非現在時刻」を参照時刻として使用するILMルールを作成できます（「このルールを古いオブジェクトバージョンにのみ適用しますか？」という質問に対して\*はい\*を選択します）。「[ILMルールの作成ウィザードのステップ1](#)」。オブジェクトが更新されると、以前のバージョンは非最新になります。「非現在の時間」フィルターを使用すると、オブジェクトの以前のバージョンによるストレージへの影響を軽減するポリシーを作成できます。



マルチパートアップロード操作を使用してオブジェクトの新しいバージョンをアップロードすると、オブジェクトの元のバージョンの非現在の時間は、マルチパートアップロードが完了したときではなく、新しいバージョンのマルチパートアップロードが作成されたときに反映されます。場合によっては、元のバージョンの非現在の時刻が現在のバージョンの時刻よりも数時間または数日前になることがあります。

#### 関連情報

- ["S3 バージョン管理オブジェクトの削除方法"](#)
- ["S3 バージョン管理オブジェクトの ILM ルールとポリシー \(例 4\)"](#)。

## S3 REST API を使用して S3 オブジェクトロックを設定する

StorageGRIDシステムでグローバル S3 オブジェクト ロック設定が有効になっている場合は、S3 オブジェクト ロックが有効になっているバケットを作成できます。各バケットのデフォルトの保持期間、または各オブジェクト バージョンの保持期間設定を指定できます。

### バケットのS3オブジェクトロックを有効にする方法

StorageGRIDシステムでグローバル S3 オブジェクト ロック設定が有効になっている場合は、各バケットを作成するときにオプションで S3 オブジェクト ロックを有効にできます。

S3 オブジェクト ロックは、バケットを作成するときのみ有効にできる永続的な設定です。バケットの作成後に S3 オブジェクトロックを追加または無効にすることはできません。

バケットに対して S3 オブジェクトロックを有効にするには、次のいずれかの方法を使用します。

- テナント マネージャーを使用してバケットを作成します。見る["S3バケットを作成する"](#)。
- CreateBucketリクエストを使用してバケットを作成します。`x-amz-bucket-object-lock-enabled` リクエスト ヘッダー。見る["バケットの操作"](#)。

S3 オブジェクト ロックにはバケットのバージョン管理が必要です。これはバケットの作成時に自動的に有効になります。バケットのバージョン管理を一時停止することはできません。見る["オブジェクトのバージョン管理"](#)。

### バケットのデフォルトの保持設定

バケットに対して S3 オブジェクトロックが有効になっている場合は、オプションでバケットのデフォルトの保存を有効にし、デフォルトの保存モードとデフォルトの保存期間を指定できます。

## デフォルトの保持モード

- コンプライアンスモードの場合:
  - オブジェクトは、保持期限に達するまで削除できません。
  - オブジェクトの保持期限を増やすことはできますが、減らすことはできません。
  - オブジェクトの保持期限は、その日付に達するまで削除できません。
- ガバナンスモードの場合:
  - ユーザーは `s3:BypassGovernanceRetention` 許可は `x-amz-bypass-governance-retention: true` 保持設定をバイパスするためのリクエスト ヘッダー。
  - これらのユーザーは、オブジェクト バージョンを、その保持期限に達する前に削除できます。
  - これらのユーザーは、オブジェクトの保持期限を増減または削除できます。

## デフォルト保持期間

各バケットには、年または日数で指定されたデフォルトの保持期間を設定できます。

## バケットのデフォルトの保持期間を設定する方法

バケットのデフォルトの保持期間を設定するには、次のいずれかの方法を使用します。

- テナント マネージャーからバケット設定を管理します。見る["S3バケットを作成する"](#)そして["S3 オブジェクトロックのデフォルト保持を更新"](#)。
- バケットに対して PutObjectLockConfiguration リクエストを発行し、デフォルトのモードとデフォルトの日数または年数を指定します。

## PutObjectLockConfiguration

PutObjectLockConfiguration リクエストを使用すると、S3 オブジェクトロックが有効になっているバケットのデフォルトの保持モードとデフォルトの保持期間を設定および変更できます。以前に構成したデフォルトの保持設定を削除することもできます。

新しいオブジェクトバージョンがバケットに取り込まれると、次の場合、デフォルトの保持モードが適用されます。 `x-amz-object-lock-mode` そして `x-amz-object-lock-retain-until-date` 指定されていません。デフォルトの保存期間は、次の場合に保存期限日を計算するために使用されます。 `x-amz-object-lock-retain-until-date` 指定されていません。

オブジェクト バージョンの取り込み後にデフォルトの保持期間が変更された場合、オブジェクト バージョンの保持期限は同じままとなり、新しいデフォルトの保持期間を使用して再計算されることはありません。

あなたは `s3:PutBucketObjectLockConfiguration` この操作を完了するには、権限を持っているか、アカウントの root である必要があります。

その `Content-MD5` PUT リクエストではリクエスト ヘッダーを指定する必要があります。

## リクエスト例

この例では、バケットの S3 オブジェクトロックを有効にし、デフォルトの保持モードを COMPLIANCE に、デフォルトの保持期間を 6 年に設定します。

```
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>
```

## バケットのデフォルトの保持期間を決定する方法

バケットに対して S3 オブジェクト ロックが有効になっているかどうかを確認し、デフォルトの保持モードと保持期間を確認するには、次のいずれかの方法を使用します。

- テナント マネージャーでバケットを表示します。見る["S3バケットを表示"](#)。
- `GetObjectLockConfiguration` 要求を発行します。

### **GetObjectLockConfiguration**

`GetObjectLockConfiguration` リクエストを使用すると、バケットに対して S3 オブジェクト ロックが有効になっているかどうかを確認し、有効になっている場合は、バケットにデフォルトの保持モードと保持期間が設定されているかどうかを確認できます。

新しいオブジェクトバージョンがバケットに取り込まれると、次の場合、デフォルトの保持モードが適用されます。`x-amz-object-lock-mode`指定されていません。デフォルトの保存期間は、次の場合に保存期限日を計算するために使用されます。`x-amz-object-lock-retain-until-date`指定されていません。

あなたは`s3:GetBucketObjectLockConfiguration`この操作を完了するには、権限を持っているか、アカウントの root である必要があります。

### リクエスト例

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

## 応答例

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

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

## オブジェクトの保持設定を指定する方法

S3 オブジェクトロックが有効になっているバケットには、S3 オブジェクトロックの保持設定があるオブジェクトとないオブジェクトの組み合わせを含めることができます。

オブジェクト レベルの保持設定は、S3 REST API を使用して指定されます。オブジェクトの保持設定は、バケットのデフォルトの保持設定よりも優先されます。

各オブジェクトに対して次の設定を指定できます。

- 保持モード: コンプライアンスまたはガバナンスのいずれか。
- **Retain-until-date:** オブジェクト バージョンをStorageGRIDで保持する必要がある期間を指定する日付。
  - COMPLIANCE モードでは、保持期限が将来の日付である場合、オブジェクトを取得することはでき

ますが、変更または削除することはできません。保持期限を増やすことはできますが、この日付を減らしたり削除したりすることはできません。

- GOVERNANCE モードでは、特別な権限を持つユーザーは、retain-until-date 設定をバイパスできません。保持期間が経過する前にオブジェクト バージョンを削除できます。また、保持期限を延長、短縮、または削除することもできます。
- 法的保留: オブジェクト バージョンに法的保留を適用すると、そのオブジェクトは直ちにロックされます。たとえば、調査や法的紛争に関連するオブジェクトに対して法的保留を設定する必要がある場合があります。法的保留には有効期限はありませんが、明示的に削除されるまで有効のままになります。

オブジェクトの法的保留設定は、保持モードおよび保持期限とは無関係です。オブジェクト バージョンが法的保留中の場合、そのバージョンを削除することは誰にもできません。

バケットにオブジェクトバージョンを追加するときにS3オブジェクトロック設定を指定するには、"[PutObject](#)"、"[CopyObject](#)"、または"[CreateMultipartUpload](#)"リクエスト。

以下を使用できます。

- ``x-amz-object-lock-mode`` COMPLIANCE または GOVERNANCE (大文字と小文字が区別されます) のいずれかになります。



指定する場合 `x-amz-object-lock-mode`、また指定する必要があります `x-amz-object-lock-retain-until-date`。

- `x-amz-object-lock-retain-until-date`
  - 保持期限の値は、次の形式でなければなりません。2020-08-10T21:46:00Z。小数秒は許可されませんが、保持されるのは小数点第 3 位のみです (ミリ秒の精度)。その他の ISO 8601 形式は許可されません。
  - 保持期限は将来の日付にする必要があります。
- `x-amz-object-lock-legal-hold`

法的保留がオンの場合 (大文字と小文字が区別されます)、オブジェクトは法的保留の対象となります。法的保留がオフの場合、法的保留は設定されません。その他の値を指定すると、400 Bad Request (InvalidArgument) エラーが発生します。

これらのリクエスト ヘッダーのいずれかを使用する場合は、次の制限に注意してください。

- その Content-MD5 リクエストヘッダーは必須です ``x-amz-object-lock-*`` リクエスト ヘッダーは PutObject リクエストに存在します。 ``Content-MD5 CopyObject`` または `CreateMultipartUpload`` には必要ありません。
- バケットにS3オブジェクトロックが有効になっていない場合、 ``x-amz-object-lock-*`` リクエスト ヘッダーが存在する場合、400 Bad Request (InvalidRequest) エラーが返されます。
- PutObjectリクエストは、 `x-amz-storage-class: REDUCED_REDUNDANCY AWS` の動作と一致させるためです。ただし、S3 オブジェクト ロックが有効になっているバケットにオブジェクトが取り込まれると、StorageGRID は常にデュアルコミット取り込みを実行します。
- 後続のGETまたはHeadObjectバージョンのレスポンスにはヘッダーが含まれます `x-amz-object-lock-mode``、 `x-amz-object-lock-retain-until-date``、そして ``x-amz-object-lock-legal-hold`` 設定されていて、リクエスト送信者が正しい ``s3:Get*`` 権限。

使用することができます `s3:object-lock-remaining-retention-days` オブジェクトの許容される最小および最大保持期間を制限するポリシー条件キー。

## オブジェクトの保持設定を更新する方法

既存のオブジェクト バージョンの法的保留または保持設定を更新する必要がある場合は、次のオブジェクト サブリソース操作を実行できます。

- `PutObjectLegalHold`

新しい法的保留値がオンの場合、オブジェクトは法的保留下に置かれます。法的保留値が OFF の場合、法的保留は解除されます。

- `PutObjectRetention`

- モード値は、COMPLIANCE または GOVERNANCE (大文字と小文字が区別されます) になります。
- 保持期限の値は、次の形式でなければなりません。2020-08-10T21:46:00Z。小数秒は許可されませんが、保持されるのは小数点第 3 位のみです (ミリ秒の精度)。その他の ISO 8601 形式は許可されません。
- オブジェクト バージョンに既存の保持期限がある場合は、それを増やすことしかできません。新しい値は将来のものである必要があります。

## ガバナンスモードの使い方

ユーザーは `s3:BypassGovernanceRetention` 権限により、GOVERNANCE モードを使用するオブジェクトのアクティブな保持設定がバイパスされる可能性があります。DELETE または `PutObjectRetention` 操作には、`x-amz-bypass-governance-retention:true` リクエスト ヘッダー。これらのユーザーは次の追加操作を実行できます。

- 保持期間が経過する前にオブジェクト バージョンを削除するには、`DeleteObject` または `DeleteObjects` 操作を実行します。

法的保留中のオブジェクトは削除できません。法的保留はオフにする必要があります。

- オブジェクトの保持期間が経過する前に、オブジェクト バージョンのモードを GOVERNANCE から COMPLIANCE に変更する `PutObjectRetention` 操作を実行します。

モードを COMPLIANCE から GOVERNANCE に変更することは許可されません。

- オブジェクト バージョンの保持期間を増加、減少、または削除するには、`PutObjectRetention` 操作を実行します。

## 関連情報

- ["S3 オブジェクトロックでオブジェクトを管理する"](#)
- ["S3 オブジェクトロックを使用してオブジェクトを保持する"](#)
- ["Amazon Simple Storage Service ユーザーガイド: オブジェクトのロック"](#)

## S3 ライフサイクル設定を作成する

S3 ライフサイクル設定を作成して、特定のオブジェクトが StorageGRID システムから

つ削除されるかを制御することができます。

このセクションの簡単な例は、S3 ライフサイクル設定によって、特定の S3 バケットから特定のオブジェクトがいつ削除される (期限切れになる) かを制御する方法を示しています。このセクションの例は説明のみを目的としています。S3 ライフサイクル設定の作成の詳細については、以下を参照してください。"[Amazon Simple Storage Service ユーザーガイド: オブジェクトのライフサイクル管理](#)"。StorageGRID は有効期限アクションのみをサポートし、遷移アクションはサポートしていないことに注意してください。

ライフサイクル構成とは

ライフサイクル設定は、特定の S3 バケット内のオブジェクトに適用される一連のルールです。各ルールでは、影響を受けるオブジェクトと、それらのオブジェクトの有効期限 (特定の日付または一定の日数後) を指定します。

StorageGRID は、ライフサイクル構成で最大 1,000 個のライフサイクル ルールをサポートします。各ルールには次の XML 要素を含めることができます。

- 有効期限: 指定した日付に達したとき、またはオブジェクトが取り込まれた時点から指定した日数に達したときに、オブジェクトを削除します。
- NoncurrentVersionExpiration: オブジェクトが非最新になった時点から指定された日数が経過すると、オブジェクトを削除します。
- フィルター (プレフィックス、タグ)
- ステータス
- ID

各オブジェクトは、S3 バケットのライフサイクルまたは ILM ポリシーの保持設定に従います。S3 バケットのライフサイクルが設定されている場合、ライフサイクルの有効期限アクションは、バケットのライフサイクル フィルターに一致するオブジェクトの ILM ポリシーをオーバーライドします。バケットのライフサイクル フィルターに一致しないオブジェクトは、ILM ポリシーの保持設定を使用します。オブジェクトがバケットのライフサイクル フィルタに一致し、有効期限アクションが明示的に指定されていない場合、ILM ポリシーの保持設定は使用されず、オブジェクト バージョンは永久に保持されるものとみなされます。見る"[S3 バケットライフサイクルと ILM ポリシーの優先順位の例](#)"。

その結果、ILM ルールの配置指示がオブジェクトに適用されているにもかかわらず、オブジェクトがグリッドから削除される可能性があります。または、オブジェクトに対する ILM 配置指示が失効した後でも、オブジェクトがグリッド上に保持されることがあります。詳細については、"[ILM がオブジェクトのライフサイクル全体にわたってどのように機能するか](#)"。



バケットのライフサイクル設定は、S3 オブジェクト ロックが有効になっているバケットで使用できますが、従来の準拠バケットではバケットのライフサイクル設定はサポートされていません。

StorageGRID は、ライフサイクル構成を管理するために次のバケット操作の使用をサポートしています。

- DeleteBucketLifecycle
- GetBucketLifecycleConfiguration
- PutBucketLifecycleConfiguration

## ライフサイクル構成を作成する

ライフサイクル構成を作成する最初の手順として、1つ以上のルールを含む JSON ファイルを作成します。たとえば、この JSON ファイルには次の3つのルールが含まれています。

1. ルール1は接頭辞に一致するオブジェクトにのみ適用される `category1` / そして、`key2` の値 `tag2`。その `Expiration` パラメーターは、フィルターに一致するオブジェクトが 2020 年 8 月 22 日の深夜に期限切れになることを指定します。
2. ルール2は接頭辞に一致するオブジェクトにのみ適用される `category2`。その `Expiration` パラメーターは、フィルターに一致するオブジェクトが取り込まれてから 100 日後に期限切れになることを指定します。



日数を指定するルールは、オブジェクトが取り込まれた時点を基準とします。現在の日付が取り込み日に日数を加えた日付を超えている場合、ライフサイクル設定が適用されるとすぐに一部のオブジェクトがバケットから削除される可能性があります。

3. ルール3は接頭辞に一致するオブジェクトにのみ適用される `category3`。その `Expiration` パラメーターは、一致するオブジェクトの非現在のバージョンが非現在のバージョンになってから 50 日後に期限切れになることを指定します。

```

{
  "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"
    }
  ]
}

```

バケットにライフサイクル設定を適用する

ライフサイクル設定ファイルを作成したら、PutBucketLifecycleConfiguration リクエストを発行してそれをバケットに適用します。

このリクエストは、サンプルファイルのライフサイクル設定をバケット内のオブジェクトに適用します。testbucket。

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

ライフサイクル構成がバケットに正常に適用されたことを検証するには、GetBucketLifecycleConfiguration リクエストを発行します。例えば：

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

成功した応答には、適用したライフサイクル構成が一覧表示されます。

バケットのライフサイクルの有効期限がオブジェクトに適用されていることを検証する

PutObject、HeadObject、または GetObject リクエストを発行するときに、ライフサイクル構成の有効期限ルールが特定のオブジェクトに適用されるかどうかを判断できます。ルールが適用される場合、応答には `Expiration` オブジェクトの有効期限と、どの有効期限ルールに一致したかを示すパラメーター。



バケットのライフサイクルはILMよりも優先されるため、`expiry-date`表示されるのは、オブジェクトが削除される実際の日付です。詳細については、["物体の保持がどのように決定されるか"](#)。

例えば、このPutObjectリクエストは2020年6月22日に発行され、オブジェクトを `testbucket` バケツ。

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

成功応答は、オブジェクトが 100 日後 (2020 年 10 月 1 日) に期限切れになり、ライフサイクル構成のルール 2 に一致したことを示します。

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

たとえば、この HeadObject リクエストは、testbucket バケット内の同じオブジェクトのメタデータを取得するために使用されました。

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

成功応答にはオブジェクトのメタデータが含まれており、オブジェクトが 100 日後に期限切れになることと、ルール 2 に一致したことが示されます。

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



バージョン対応バケットの場合、`x-amz-expiration` 応答ヘッダーはオブジェクトの現在のバージョンにのみ適用されます。

### S3 REST API 実装に関する推奨事項

StorageGRIDで使用するために S3 REST API を実装する場合は、次の推奨事項に従う必要があります。

#### 存在しないオブジェクトへのHEADの推奨事項

アプリケーションが、実際には存在しないと思われるパスにオブジェクトが存在するかどうかを定期的に確認する場合は、「使用可能」を使用する必要があります。「[一貫性](#)」。たとえば、アプリケーションが PUT を実行する前に場所を HEAD する場合は、「Available」一貫性を使用する必要があります。

そうでない場合、HEAD 操作でオブジェクトが見つからないと、同じサイトにある 2 つ以上のストレージ ノードが使用できないか、リモート サイトにアクセスできないときに、多数の 500 内部サーバー エラーが発生する可能性があります。

各バケットの「利用可能」一貫性を設定するには、「[PUT バケットの一貫性](#)」リクエストごとに一貫性を指定したり、個々の API 操作のリクエスト ヘッダーで一貫性を指定したりすることもできます。

#### オブジェクトキーに関する推奨事項

バケットが最初に作成された時期に基づいて、オブジェクト キー名に関する次の推奨事項に従ってください。

#### StorageGRID 11.4 以前で作成されたバケット

- オブジェクト キーの最初の 4 文字としてランダムな値を使用しないでください。これは、キープレフィックスに関する以前の AWS 推奨事項とは対照的です。代わりに、ランダムではない、ユニークではない接頭辞を使用してください。 image。
- キープレフィックスにランダムで一意的文字を使用するという以前の AWS 推奨事項に従う場合は、オブジェクトキーの前にディレクトリ名を付けます。つまり、次の形式を使用します。

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

この形式の代わりに:

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

## StorageGRID 11.4以降で作成されたバケット

パフォーマンスのベスト プラクティスを満たすためにオブジェクト キー名を制限する必要はありません。ほとんどの場合、オブジェクト キー名の最初の 4 文字にはランダムな値を使用できます。



例外は、短時間ですべてのオブジェクトを継続的に削除する S3 ワークロードです。このユースケースでのパフォーマンスへの影響を最小限に抑えるには、数千個のオブジェクトごとにキー名の先頭部分を日付などに変更します。たとえば、S3 クライアントが通常 1 秒あたり 2,000 個のオブジェクトを書き込み、ILM またはバケット ライフサイクル ポリシーによって 3 日後にすべてのオブジェクトが削除されるとします。パフォーマンスへの影響を最小限に抑えるには、次のようなパターンを使用してキーに名前を付けます。

```
/mybucket/mydir/yyyyymmddhhmmss-random_UUID.jpg
```

## 「レンジリード」の推奨事項

もし"保存されたオブジェクトを圧縮するグローバルオプション"が有効になっている場合、S3 クライアントアプリケーションは、返されるバイトの範囲を指定する GetObject 操作の実行を避ける必要があります。これらの「範囲読み取り」操作は、StorageGRID が要求されたバイトにアクセスするためにオブジェクトを効果的に解凍する必要があるため、非効率的です。非常に大きなオブジェクトから小さな範囲のバイトを要求する GetObject 操作は特に非効率的です。たとえば、50 GB の圧縮されたオブジェクトから 10 MB の範囲を読み取るのは非効率的です。

範囲が圧縮されたオブジェクトから読み取られる場合、クライアント要求がタイムアウトする可能性があります。



オブジェクトを圧縮する必要があり、クライアントアプリケーションで範囲読み取りを使用する必要がある場合は、アプリケーションの読み取りタイムアウトを増やします。

# Amazon S3 REST APIのサポート

## S3 REST API実装の詳細

StorageGRIDシステムは、ほとんどの操作をサポートする Simple Storage Service API (API バージョン 2006-03-01) を実装していますが、いくつかの制限があります。S3 REST API クライアント アプリケーションを統合する場合は、実装の詳細を理解する必要があります。

StorageGRIDシステムは、仮想ホスト スタイルのリクエストとパス スタイルのリクエストの両方をサポートします。

## 日付処理

S3 REST API のStorageGRID実装では、有効な HTTP 日付形式のみがサポートされます。

StorageGRIDシステムは、日付値を受け入れるヘッダーに対して有効な HTTP 日付形式のみをサポートします。日付の時間部分は、グリニッジ標準時 (GMT) 形式、またはタイムゾーン オフセットなしの世界協定時刻 (UTC) 形式で指定できます (+0000 を指定する必要があります)。含めると `x-amz-date` リクエストのヘッダーに Date を指定すると、Date リクエスト ヘッダーに指定された値が上書きされます。AWS署名バージョン4を使用する場合、`x-amz-date`日付ヘッダーはサポートされていないため、署名されたリクエストには日付ヘッダーが存在する必要があります。

## 一般的なリクエストヘッダー

StorageGRIDシステムは、以下で定義された共通リクエストヘッダーをサポートしています。"[Amazon Simple Storage Service API リファレンス: 共通リクエストヘッダー](#)"ただし、1つの例外があります。

リクエストヘッダー	導入
承認	AWS 署名バージョン 2 の完全サポート  AWS 署名バージョン 4 のサポート (ただし、次の例外があります)。 <ul style="list-style-type: none"><li>実際のペイロードチェックサム値を <code>x-amz-content-sha256</code>、値は検証なしで受け入れられ、値が <code>UNSIGNED-PAYLOAD</code> ヘッダーに提供されていました。あなたが <code>x-amz-content-sha256</code> ヘッダー値の意味 <code>aws-chunked</code> ストリーミング (たとえば、<code>STREAMING-AWS4-HMAC-SHA256-PAYLOAD</code>) の場合、チャンク署名はチャンク データに対して検証されません。</li></ul>
x-amz-セキュリティトークン	実装されていません。返品 XNotImplemented。

## 一般的なレスポンスヘッダー

StorageGRIDシステムは、1つの例外を除き、「Simple Storage Service API リファレンス」で定義されているすべての共通応答ヘッダーをサポートします。

レスポンスヘッダー	導入
x-amz-id-2	未使用

## リクエストを認証する

StorageGRIDシステムは、S3 API を使用したオブジェクトへの認証アクセスと匿名アクセスの両方をサポートします。

S3 API は、S3 API リクエストの認証に署名バージョン 2 と署名バージョン 4 をサポートしています。

認証されたリクエストは、アクセス キー ID とシークレット アクセス キーを使用して署名する必要があります。

StorageGRIDシステムは、HTTPとHTTPSの2つの認証方法をサポートしています。`Authorization`ヘッダーとクエリパラメータを使用します。

### HTTP認証ヘッダーを使用する

HTTP `Authorization`ヘッダーは、バケット ポリシーで許可されている匿名リクエストを除くすべての S3 API 操作で使用されます。その `Authorization`ヘッダーには、リクエストを認証するために必要な署名情報がすべて含まれています。

### クエリパラメータを使用する

クエリ パラメータを使用して、URL に認証情報を追加できます。これは URL の事前署名と呼ばれ、特定の リソースへの一時的なアクセスを許可するために使用できます。署名済み URL を持つユーザーは、リソースにアクセスするために秘密アクセス キーを知る必要がないため、リソースへのサードパーティの制限付きアクセスを提供できます。

## サービス上の操作

StorageGRIDシステムは、サービス上で次の操作をサポートします。

処理	導入
ListBuckets (以前の名称は GET サービス)	すべての Amazon S3 REST API 動作が実装されています。予告なく変更されることがあります。
ストレージ使用量の取得	StorageGRID" <a href="#">ストレージ使用量の取得</a> "リクエストにより、アカウントで使用されているストレージの合計量と、アカウントに関連付けられている各バケットのストレージ量がわかります。これは、/ のパスとカスタムクエリパラメータを使用したサービスに対する操作です。( ?x-ntap-sg-usage ) 追加した。
OPTIONS /	クライアントアプリケーションは、OPTIONS / S3 認証資格情報を提供せずに、ストレージ ノードの S3 ポートに要求を送信し、ストレージ ノードが使用可能かどうかを判断します。このリクエストは監視に使用したり、外部ロードバランサーがストレージノードがダウンしたかどうかを識別できるようにするために使用できます。

## バケットの操作

StorageGRIDシステムは、S3 テナント アカウントごとに最大 5,000 個のバケットをサポートします。

各グリッドには最大 100,000 個のバケットを含めることができます。

5,000 個のバケットをサポートするには、グリッド内の各ストレージ ノードに少なくとも 64 GB の RAM が必要です。

バケット名の制限は AWS US 標準リージョンの制限に従いますが、S3 仮想ホスト形式のリクエストをサポートするには、DNS 命名規則にさらに制限する必要があります。

詳細については、以下を参照してください。

- ["Amazon Simple Storage Service ユーザーガイド: バケットのクォータ、制限、制約事項"](#)
- ["S3エンドポイントのドメイン名を設定する"](#)

ListObjects (GET Bucket) および ListObjectVersions (GET Bucket object versions) オペレーションはStorageGRIDをサポートします"[一貫性値](#)"。

個々のバケットに対して最終アクセス時刻の更新が有効になっているか無効になっているかを確認できます。見る"[GET バケットの最終アクセス時刻](#)"。

次の表は、StorageGRID がS3 REST API バケット操作を実装する方法を示しています。これらの操作を実行するには、アカウントに必要なアクセス資格情報を提供する必要があります。

処理	導入
CreateBucket	<p>新しいバケットを作成します。バケットを作成すると、バケットの所有者になります。</p> <ul style="list-style-type: none"> <li>• バケット名は次の規則に従う必要があります。 <ul style="list-style-type: none"> <li>◦ 各StorageGRIDシステム全体で一意である必要があります (テナント アカウント内で一意だけでなく)。</li> <li>◦ DNS に準拠している必要があります。</li> <li>◦ 3 文字以上 63 文字以下でなければなりません。</li> <li>◦ 隣接するラベルをピリオドで区切った、1つ以上のラベルの連続にすることができます。各ラベルは小文字または数字で始まり、終わる必要があります。小文字、数字、ハイフンのみを使用できます。</li> <li>◦ テキスト形式の IP アドレスのように見えてはいけません。</li> <li>◦ 仮想ホスト形式のリクエストではピリオドを使用しないでください。ピリオドを使用すると、サーバーのワイルドカード証明書の検証で問題が発生します。</li> </ul> </li> <li>• デフォルトでは、バケットは `us-east-1` 地域;ただし、`LocationConstraint` 別のリージョンを指定するには、リクエスト本文にリクエスト要素を追加します。使用する際は `LocationConstraint` 要素を使用する場合は、グリッド マネージャーまたはグリッド管理 API を使用して定義されたリージョンの正確な名前を指定する必要があります。使用すべき地域名がわからない場合は、システム管理者に問い合わせてください。</li> </ul> <p>注意: CreateBucket リクエストでStorageGRIDで定義されていないリージョンを使用すると、エラーが発生します。</p> <ul style="list-style-type: none"> <li>• 以下のものを含めることができます `x-amz-bucket-object-lock-enabled` S3 オブジェクト ロックが有効になっているバケットを作成するためのリクエストヘッダー。見る<a href="#">"S3 REST API を使用して S3 オブジェクトロックを設定する"</a>。</li> </ul> <p>バケットを作成するときに、S3 オブジェクトロックを有効にする必要があります。バケットの作成後に S3 オブジェクトロックを追加または無効にすることはできません。S3 オブジェクトロックにはバケットのバージョン管理が必要です。これはバケットを作成すると自動的に有効になります。</p>
DeleteBucket	バケットを削除します。
バケットCorsの削除	バケットの CORS 構成を削除します。
バケット暗号化の削除	バケットからデフォルトの暗号化を削除します。既存の暗号化されたオブジェクトは暗号化されたままですが、バケットに追加された新しいオブジェクトは暗号化されません。

処理	導入
DeleteBucketLifecycle	バケットからライフサイクル構成を削除します。見る <a href="#">"S3ライフサイクル設定を作成する"</a> 。
DeleteBucketPolicy	バケットに添付されたポリシーを削除します。
バケットレプリケーションの削除	バケットに添付されたレプリケーション設定を削除します。
バケットタグの削除	<p>使用するのは `tagging` バケットからすべてのタグを削除するサブリソース。</p> <p>注意: このバケットにデフォルト以外のILMポリシータグが設定されている場合は、`NTAP-SG-ILM-BUCKET-TAG` 値が割り当てられたバケット タグ。DeleteBucketTagging リクエストを発行しないでください。`NTAP-SG-ILM-BUCKET-TAG` バケットタグ。代わりに、PutBucketTagging リクエストを発行し、`NTAP-SG-ILM-BUCKET-TAG` タグとそれに割り当てられた値を入力すると、バケットから他のすべてのタグが削除されます。変更したり削除したりしないでください `NTAP-SG-ILM-BUCKET-TAG` バケットタグ。</p>
GetBucketAcl	肯定応答とバケット所有者の ID、DisplayName、および権限を返し、所有者がバケットへのフルアクセス権を持っていることを示します。
GetBucketCors	返します `cors` バケットの構成。
バケット暗号化を取得	バケットのデフォルトの暗号化設定を返します。
GetBucketLifecycleConfiguration  (以前はGET Bucket ライフサイクルと呼ばれていました)	バケットのライフサイクル構成を返します。見る <a href="#">"S3ライフサイクル設定を作成する"</a> 。
GetBucketLocation	設定された領域を返します。LocationConstraint CreateBucket リクエスト内の要素。バケットのリージョンが us-east-1、その領域に対して空の文字列が返されます。
GetBucketNotificationConfiguration  (以前の名称は GET Bucket 通知)	バケットに添付された通知設定を返します。
GetBucketPolicy	バケットに添付されたポリシーを返します。
GetBucketReplication	バケットに添付されたレプリケーション設定を返します。

処理	導入
GetBucketTagging	<p>使用するのは `tagging` バケットのすべてのタグを返すサブリソース。</p> <p>注意: このバケットにデフォルト以外のILMポリシータグが設定されている場合は、`NTAP-SG-ILM-BUCKET-TAG`値が割り当てられたバケット タグ。このタグを変更したり削除したりしないでください。</p>
GetBucketVersioning	<p>この実装では、`versioning`バケットのバージョン管理状態を返すサブリソース。</p> <ul style="list-style-type: none"> <li>• <i>blank</i>: バージョン管理が有効になったことがありません (バケットは「バージョン管理なし」です)</li> <li>• 有効: バージョン管理が有効です</li> <li>• 一時停止: バージョン管理は以前は有効でしたが、一時停止されています</li> </ul>
GetObjectLockConfiguration	<p>設定されている場合は、バケットのデフォルトの保持モードとデフォルトの保持期間を返します。</p> <p>見る"<a href="#">S3 REST API を使用して S3 オブジェクトロックを設定する</a>"。</p>
HeadBucket	<p>バケットが存在するかどうか、またそれにアクセスする権限があるかどうかを判断します。</p> <p>この操作は次を返します:</p> <ul style="list-style-type: none"> <li>• <code>x-ntap-sg-bucket-id</code>: UUID 形式のバケットの UUID。</li> <li>• <code>x-ntap-sg-trace-id</code>: 関連付けられたリクエストの一意のトレース ID。</li> </ul>
ListObjects と ListObjectsV2  (以前の名称は GET Bucket)	<p>バケット内のオブジェクトの一部またはすべて (最大 1,000 個) を返します。オブジェクトのストレージクラスは、オブジェクトが `REDUCED_REDUNDANCY` ストレージクラスオプション:</p> <ul style="list-style-type: none"> <li>• `STANDARD` これは、オブジェクトがストレージ ノードで構成されるストレージ プールに保存されていることを示します。</li> <li>• `GLACIER` これは、オブジェクトが Cloud Storage Pool によって指定された外部バケットに移動されたことを示します。</li> </ul> <p>バケットに同じプレフィックスを持つ削除されたキーが多数含まれている場合、レスポンスには `CommonPrefixes` キーが含まれないもの。</p>
ListObjectVersions  (以前の名称は GET Bucket Object バージョン)	<p>バケットへのREADアクセスで、この操作を `versions` サブリソースは、バケット内のオブジェクトのすべてのバージョンのメタデータを一覧表示します。</p>

処理	導入
PutBucketCors	<p>バケットがクロスオリジン リクエストを処理できるように、バケットの CORS 構成を設定します。クロスオリジン リソース共有 (CORS) は、あるドメイン内のクライアント Web アプリケーションが別のドメイン内のリソースにアクセスできるようにするセキュリティ メカニズムです。たとえば、S3バケットの名前が images`グラフィックを保存します。 CORS設定を設定することで `images`バケット内の画像をウェブサイトに表示できるようにすることができます `http://www.example.com。</p>
PutBucketEncryption	<p>既存のバケットのデフォルトの暗号化状態を設定します。バケットレベルの暗号化を有効にすると、バケットに追加された新しいオブジェクトはすべて暗号化されます。StorageGRIDは、StorageGRID管理キーを使用したサーバー側の暗号化をサポートしています。サーバー側暗号化設定ルールを指定するときは、SSEAlgorithm`パラメータに `AES256、そして、`KMSMasterKeyID`パラメータ。</p> <p>オブジェクトのアップロードリクエストで既に暗号化が指定されている場合 (つまり、リクエストに `x-amz-server-side-encryption-*`リクエスト ヘッダー)。</p>
PutBucketLifecycleConfiguration  (以前はPUT Bucketライフサイクルと呼ばれていました)	<p>バケットの新しいライフサイクル設定を作成するか、既存のライフサイクル設定を置き換えます。StorageGRID は、ライフサイクル構成で最大 1,000 個のライフサイクル ルールをサポートします。各ルールには次の XML 要素を含めることができます。</p> <ul style="list-style-type: none"> <li>• 有効期限 (日数、日付、ExpiredObjectDeleteMarker)</li> <li>• NoncurrentVersionExpiration (NewerNoncurrentVersions、NoncurrentDays)</li> <li>• フィルター (プレフィックス、タグ)</li> <li>• ステータス</li> <li>• ID</li> </ul> <p>StorageGRID は次のアクションをサポートしていません。</p> <ul style="list-style-type: none"> <li>• AbortIncompleteMultipartUpload</li> <li>• 移行</li> </ul> <p>見る<a href="#">"S3ライフサイクル設定を作成する"</a>。バケットライフサイクルの有効期限アクションがILM配置指示とどのように相互作用するかを理解するには、以下を参照してください。"<a href="#">ILMがオブジェクトのライフサイクル全体にわたってどのように機能するか</a>"。</p> <p>注意: バケットのライフサイクル設定は、S3 オブジェクト ロックが有効になっているバケットで使用できますが、従来の準拠バケットではバケットのライフサイクル設定はサポートされていません。</p>

処理	導入
PutBucketNotificationConfiguration  (以前の名称はPUT Bucket 通知)	<p>リクエスト本文に含まれる通知構成 XML を使用して、バケットの通知を構成します。次の実装の詳細に注意する必要があります。</p> <ul style="list-style-type: none"> <li>• StorageGRID は、送信先として Amazon Simple Notification Service (Amazon SNS) または Kafka トピックをサポートしています。 Simple Queue Service (SQS) または Amazon Lambda エンドポイントはサポートされていません。</li> <li>• 通知の宛先は、 StorageGRID エンドポイントの URN として指定する必要があります。 エンドポイントは、テナント マネージャーまたはテナント管理 API を使用して作成できます。</li> </ul> <p>通知構成を成功させるには、エンドポイントが存在している必要があります。 エンドポイントが存在しない場合は、 400 Bad Request `コードとともにエラーが返されます `InvalidArgument。</p> <ul style="list-style-type: none"> <li>• 次のイベント タイプに対しては通知を構成できません。これらのイベントタイプはサポートされていません。 <ul style="list-style-type: none"> <li>◦ s3:ReducedRedundancyLostObject</li> <li>◦ s3:ObjectRestore:Completed</li> </ul> </li> <li>• StorageGRID から送信されるイベント通知では、次のリストに示すように、一部のキーが含まれず、他のキーには特定の値を使用することを除いて、標準の JSON 形式が使用されます。 <ul style="list-style-type: none"> <li>◦ イベントソース  sgws:s3</li> <li>◦ <b>aws</b> リージョン  含まれません</li> <li>◦ <b>x-amz-id-2</b>  含まれません</li> <li>◦ アーーン  urn:sgws:s3:::bucket_name</li> </ul> </li> </ul>
PutBucketPolicy	バケットに添付されたポリシーを設定します。見る <a href="#">"バケットとグループのアクセスポリシーを使用する"</a> 。

処理	導入
PutBucketレプリケーション	<p>構成"<a href="#">StorageGRID CloudMirror レプリケーション</a>"リクエスト本文で提供されるレプリケーション構成 XML を使用してバケットに対して行います。CloudMirrorレプリケーションの場合、次の実装の詳細に注意する必要があります。</p> <ul style="list-style-type: none"> <li>StorageGRID はレプリケーション構成の V1 のみをサポートします。これは、StorageGRIDが `Filter` ルールの要素であり、オブジェクトバージョンの削除については V1 規則に従います。詳細については、"<a href="#">Amazon Simple Storage Service ユーザーガイド: レプリケーション設定</a>"。</li> <li>バケットのレプリケーションは、バージョン管理されたバケットでもバージョン管理されていないバケットでも設定できます。</li> <li>レプリケーション構成 XML の各ルールで異なる宛先バケットを指定できます。ソース バケットは複数の宛先バケットに複製できます。</li> <li>宛先バケットは、テナント マネージャまたはテナント管理 API で指定されているStorageGRIDエンドポイントの URN として指定する必要があります。見る"<a href="#">CloudMirrorレプリケーションを構成する</a>"。</li> </ul> <p>レプリケーション構成を成功させるには、エンドポイントが存在している必要があります。エンドポイントが存在しない場合は、リクエストは失敗し、400 Bad Request エラーメッセージは次のようになります。 `Unable to save the replication policy. The specified endpoint URN does not exist: URN.`</p> <ul style="list-style-type: none"> <li>指定する必要はありません `Role` 構成 XML 内。この値はStorageGRIDでは使用されず、送信されても無視されます。</li> <li>設定XMLからストレージクラスを省略した場合、StorageGRIDは `STANDARD` デフォルトのストレージ クラス。</li> <li>ソースバケットからオブジェクトを削除するか、ソースバケット自体を削除した場合、リージョン間のレプリケーションの動作は次のようになります。 <ul style="list-style-type: none"> <li>オブジェクトまたはバケットが複製される前に削除した場合、オブジェクト/バケットは複製されず、通知も行われません。</li> <li>オブジェクトまたはバケットをレプリケートした後に削除する場合、StorageGRID はクロスリージョンレプリケーションの V1 の標準の Amazon S3 削除動作に従います。</li> </ul> </li> </ul>

処理	導入
PutBucketTagging	<p>使用するのは `tagging` バケットのタグのセットを追加または更新するためのサブリソース。バケット タグを追加するときは、次の制限に注意してください。</p> <ul style="list-style-type: none"> <li>StorageGRIDと Amazon S3 はどちらも、バケットごとに最大 50 個のタグをサポートします。</li> <li>バケットに関連付けられたタグには、一意のタグ キーが必要です。タグ キーの長さは最大 128 文字の Unicode 文字です。</li> <li>タグ値の長さは最大 256 文字の Unicode 文字です。</li> <li>キーと値は大文字と小文字が区別されます。</li> </ul> <p>注意: このバケットにデフォルト以外のILMポリシータグが設定されている場合は、`NTAP-SG-ILM-BUCKET-TAG`値が割り当てられたバケット タグ。必ず `NTAP-SG-ILM-BUCKET-TAG`バケット タグは、すべての PutBucketTagging リクエストで割り当てられた値に含まれます。このタグを変更したり削除したりしないでください。</p> <p>注意: この操作により、バケットにすでに存在するタグが上書きされます。セットから既存のタグが省略されている場合、それらのタグはバケットから削除されます。</p>
PutBucketVersioning	<p>使用するのは `versioning` 既存のバケットのバージョン管理状態を設定するためのサブリソース。バージョン管理の状態は、次のいずれかの値で設定できます。</p> <ul style="list-style-type: none"> <li>有効: バケット内のオブジェクトのバージョン管理を有効にします。バケットに追加されたすべてのオブジェクトには、一意のバージョン ID が付与されます。</li> <li>停止: バケット内のオブジェクトのバージョン管理を無効にします。バケットに追加されたすべてのオブジェクトにはバージョンIDが付与されず null。</li> </ul>
PutObjectLockConfiguration	<p>バケットのデフォルトの保持モードとデフォルトの保持期間を設定または削除します。</p> <p>デフォルトの保持期間が変更された場合、既存のオブジェクト バージョンの保持期限は同じままになり、新しいデフォルトの保持期間を使用して再計算されることはありません。</p> <p>見る<a href="#">"S3 REST API を使用して S3 オブジェクトロックを設定する"</a>詳細情報については。</p>

## オブジェクトに対する操作

### オブジェクトに対する操作

このセクションでは、StorageGRIDシステムがオブジェクトに対して S3 REST API 操作を実装する方法について説明します。

すべてのオブジェクト操作には次の条件が適用されます。

- StorageGRID"一貫性値"次の例外を除き、オブジェクトに対するすべての操作でサポートされます。
  - GetObjectAcl
  - OPTIONS /
  - PutObjectLegalHold
  - PutObjectRetention
  - オブジェクトコンテンツの選択
- 2つのクライアントが同じキーに書き込むなど、競合するクライアント要求は、「最新のものを優先」して解決されます。「最新のものを優先」評価のタイミングは、S3クライアントが操作を開始した時点ではなく、StorageGRIDシステムが特定のリクエストを完了した時点に基づきます。
- StorageGRIDバケット内のすべてのオブジェクトは、匿名ユーザーまたは別のアカウントによって作成されたオブジェクトを含め、バケット所有者によって所有されます。
- Swift を介してStorageGRIDシステムに取り込まれたデータ オブジェクトには、S3 を介してアクセスできません。

次の表は、StorageGRID がS3 REST API オブジェクト操作を実装する方法を示しています。

処理	導入
DeleteObject	<p>多要素認証 (MFA) とレスポンスヘッダー `x-amz-mfa` サポートされていません。</p> <p>DeleteObject 要求を処理する場合、StorageGRID は保存されているすべての場所からオブジェクトのすべてのコピーを直ちに削除しようとします。成功した場合、StorageGRID はすぐにクライアントに応答を返します。30 秒以内にすべてのコピーを削除できない場合 (たとえば、場所が一時的に利用できないなど)、StorageGRID はコピーを削除キューに入れてから、クライアントに成功を通知します。</p> <p>バージョン管理</p> <p>特定のバージョンを削除するには、リクエスト者はバケットの所有者であり、`versionId` サブリソース。このサブリソースを使用すると、バージョンが完全に削除されます。もし `versionId` 削除マーカーに対応するレスポンスヘッダー `x-amz-delete-marker` に設定されて返されます `true`。</p> <ul style="list-style-type: none"> <li>オブジェクトが削除された場合、`versionId` バージョン管理が有効になっているバケットのサブリソースを削除すると、削除マーカーが生成されます。その `versionId` 削除マーカーは、`x-amz-version-id` レスポンスヘッダー、および `x-amz-delete-marker` レスポンスヘッダーは次のように設定されて返されます `true`。</li> <li>オブジェクトが削除された場合、`versionId` バージョン管理が一時停止されているバケットのサブリソースを削除すると、既存の「null」バージョンまたは「null」削除マーカーが永久に削除され、新しい「null」削除マーカーが生成されます。その `x-amz-delete-marker` レスポンスヘッダーは次のように設定されて返されます `true`。</li> </ul> <p>注意: 場合によっては、オブジェクトに対して複数の削除マーカーが存在することがあります。</p> <p><a href="#">見る"S3 REST API を使用して S3 オブジェクトロックを設定する</a> "GOVERNANCE モードでオブジェクト バージョンを削除する方法を学習します。</p>
DeleteObjects (以前の名称は DELETE Multiple Objects)	<p>多要素認証 (MFA) とレスポンスヘッダー `x-amz-mfa` サポートされていません。</p> <p>同じリクエスト メッセージで複数のオブジェクトを削除できます。</p> <p><a href="#">見る"S3 REST API を使用して S3 オブジェクトロックを設定する</a> "GOVERNANCE モードでオブジェクト バージョンを削除する方法を学習します。</p>

処理	導入
DeleteObjectTagging	<p>使用するのは `tagging` オブジェクトからすべてのタグを削除するサブリソース。</p> <p>バージョン管理</p> <p>もし `versionId` クエリ パラメータがリクエストに指定されていない場合、この操作では、バージョン管理されたバケット内のオブジェクトの最新バージョンからすべてのタグが削除されます。オブジェクトの現在のバージョンが削除マーカである場合、「MethodNotAllowed」ステータスが返されます。 `x-amz-delete-marker` レスポンスヘッダーを `true`。</p>
GetObject	"GetObject"
GetObjectAcl	アカウントに必要なアクセス資格情報が提供されている場合、操作は肯定応答と、オブジェクト所有者の ID、表示名、および権限を返し、所有者がオブジェクトへのフルアクセス権を持っていることを示します。
GetObjectLegalHold	"S3 REST API を使用して S3 オブジェクトロックを設定する"
GetObjectRetention	"S3 REST API を使用して S3 オブジェクトロックを設定する"
GetObjectTagging	<p>使用するのは `tagging` オブジェクトのすべてのタグを返すサブリソース。</p> <p>バージョン管理</p> <p>もし `versionId` クエリ パラメータがリクエストに指定されていない場合、この操作ではバージョン管理されたバケット内のオブジェクトの最新バージョンからすべてのタグが返されます。オブジェクトの現在のバージョンが削除マーカである場合、「MethodNotAllowed」ステータスが返されます。 `x-amz-delete-marker` レスポンスヘッダーを `true`。</p>
HeadObject	"HeadObject"
オブジェクトの復元	"オブジェクトの復元"
PutObject	"PutObject"
CopyObject (以前の名前は PUT Object - Copy)	"CopyObject"
PutObjectLegalHold	"S3 REST API を使用して S3 オブジェクトロックを設定する"
PutObjectRetention	"S3 REST API を使用して S3 オブジェクトロックを設定する"

処理	導入
PutObjectTagging	<p>使用するのは `tagging` 既存のオブジェクトにタグのセットを追加するためのサブリソース。</p> <p><b>オブジェクトタグの制限</b></p> <p>新しいオブジェクトをアップロードするときにタグを追加したり、既存のオブジェクトにタグを追加したりできます。StorageGRIDとAmazon S3 はどちらも、オブジェクトごとに最大 10 個のタグをサポートします。オブジェクトに関連付けられたタグには、一意のタグキーが必要です。タグキーの長さは最大 128 文字の Unicode 文字、タグ値の長さは最大 256 文字の Unicode 文字です。キーと値は大文字と小文字が区別されます。</p> <p><b>タグの更新と取り込み動作</b></p> <p>PutObjectTagging を使用してオブジェクトのタグを更新すると、StorageGRID はオブジェクトを再取り込みしません。これは、一致する ILM ルールで指定された取り込み動作のオプションが使用されないことを意味します。更新によってトリガーされるオブジェクト配置の変更は、通常のバックグラウンド ILM プロセスによって ILM が再評価されるときに行われます。</p> <p>つまり、ILM ルールが取り込み動作に厳密オプションを使用する場合、必要なオブジェクトを配置できない場合 (たとえば、新しく必要な場所が利用できないなど)、アクションは実行されません。更新されたオブジェクトは、必要な配置が可能になるまで現在の配置を保持します。</p> <p><b>紛争の解決</b></p> <p>2 つのクライアントが同じキーに書き込むなど、競合するクライアント要求は、「最新のものを優先」して解決されます。「最新のものの優先」評価のタイミングは、S3 クライアントが操作を開始した時点ではなく、StorageGRIDシステムが特定のリクエストを完了した時点に基づきます。</p> <p><b>バージョン管理</b></p> <p>もし `versionId` クエリ パラメータがリクエストに指定されていない場合、この操作では、バージョン管理されたバケット内のオブジェクトの最新バージョンにタグが追加されます。オブジェクトの現在のバージョンが削除マーカーである場合、「MethodNotAllowed」ステータスが返されます。 `x-amz-delete-marker` レスポンスヘッダーを `true`。</p>
オブジェクトコンテンツの選択	<a href="#">"オブジェクトコンテンツの選択"</a>

### S3 Selectを使用する

StorageGRIDは、以下のAmazon S3 Select句、データ型、演算子をサポートしています。"[SelectObjectContent コマンド](#)"。



リストされていない項目はサポートされません。

構文については、"[オブジェクトコンテンツの選択](#)"。S3 Selectの詳細については、"[S3 Select の AWS ドキュメント](#)"。

S3 Select が有効になっているテナント アカウントのみが SelectObjectContent クエリを発行できます。参照"[S3 Select の使用に関する考慮事項と要件](#)"。

#### 条項

- SELECTリスト
- FROM句
- WHERE句
- LIMIT句

#### データ型

- ブール
- integer
- string
- float
- 小数、数値
- タイムスタンプ

#### 有人対応

#### 論理演算子

- および
- ない
- または

#### 比較演算子

- <
- >
- <=
- >=
- =
- =
- <>
- !=
- 間

- で

#### パターンマッチング演算子

- のように
- \_
- %

#### ユニタリ演算子

- NULLです
- NULLではない

#### 数学演算子

- +
- -
- \*
- /
- %

StorageGRID はAmazon S3 Select 演算子の優先順位に従います。

#### 集計関数

- 平均()
- カウント (\*)
- 最大()
- 最小値()
- 和 ()

#### 条件関数

- 場合
- 合体
- NULLIF

#### 変換関数

- CAST (サポートされているデータ型の場合)

#### 日付関数

- 日付\_追加
- 日付差

- EXTRACT
- TO\_STRING
- TO\_タイムスタンプ
- UTCNOW

#### 文字列関数

- CHAR\_LENGTH、CHARACTER\_LENGTH
- より低い
- 部分文字列
- トリム
- アッパー

#### サーバー側の暗号化を使用する

サーバー側の暗号化により、保存中のオブジェクト データを保護できます。StorageGRID はオブジェクトを書き込むときにデータを暗号化し、オブジェクトにアクセスするときにデータを復号化します。

サーバー側の暗号化を使用する場合は、暗号化キーの管理方法に基づいて、相互に排他的な 2 つのオプションのいずれかを選択できます。

- **SSE ( StorageGRID管理キーによるサーバー側暗号化)**: オブジェクトを保存するための S3 リクエストを発行すると、StorageGRID は一意のキーを使用してオブジェクトを暗号化します。オブジェクトを取得するために S3 リクエストを発行すると、StorageGRID は保存されたキーを使用してオブジェクトを復号化します。
- **SSE-C (顧客提供のキーを使用したサーバー側暗号化)**: オブジェクトを保存するための S3 リクエストを発行するときに、独自の暗号化キーを提供します。オブジェクトを取得するときは、リクエストの一部として同じ暗号化キーを提供します。2 つの暗号化キーが一致する場合、オブジェクトは復号化され、オブジェクト データが返されます。

StorageGRID はすべてのオブジェクトの暗号化および復号化操作を管理しますが、提供する暗号化キーはユーザーが管理する必要があります。



提供された暗号化キーは保存されません。暗号化キーを紛失すると、対応するオブジェクトも失われます。



オブジェクトが SSE または SSE-C で暗号化されている場合、バケット レベルまたはグリッド レベルの暗号化設定はすべて無視されます。

#### SSE を使用

StorageGRIDによって管理される一意のキーを使用してオブジェクトを暗号化するには、次のリクエスト ヘッダーを使用します。

`x-amz-server-side-encryption`

SSE 要求ヘッダーは、次のオブジェクト操作でサポートされます。

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

#### SSE-Cを使用する

管理する一意のキーを使用してオブジェクトを暗号化するには、次の 3 つのリクエスト ヘッダーを使用します。

リクエストヘッダー	説明
x-amz-server-side-encryption-customer-algorithm	暗号化アルゴリズムを指定します。ヘッダー値は AES256。
x-amz-server-side-encryption-customer-key	オブジェクトの暗号化または復号化に使用する暗号化キーを指定します。キーの値は 256 ビット、base64 でエンコードされている必要があります。
x-amz-server-side-encryption-customer-key-MD5	RFC 1321 に従って暗号化キーの MD5 ダイジェストを指定します。これは、暗号化キーがエラーなく送信されたことを確認するために使用されます。MD5 ダイジェストの値は、base64 でエンコードされた 128 ビットである必要があります。

SSE-C 要求ヘッダーは、次のオブジェクト操作でサポートされます。

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

#### 顧客提供キーによるサーバー側暗号化 (SSE-C) の使用に関する考慮事項

SSE-C を使用する前に、次の点に注意してください。

- https を使用する必要があります。



StorageGRIDは、SSE-Cを使用する場合、HTTP経由のすべてのリクエストを拒否します。セキュリティ上の考慮事項として、HTTP経由で誤って送信したキーは侵害される可能性があります。キーを破棄し、必要に応じてローテーションします。

- 応答内の ETag はオブジェクト データの MD5 ではありません。

- 暗号化キーとオブジェクトのマッピングを管理する必要があります。StorageGRID は暗号化キーを保存しません。各オブジェクトに提供した暗号化キーを追跡するのはユーザーの責任です。
- バケットでバージョン管理が有効になっている場合は、各オブジェクトバージョンに独自の暗号化キーが必要です。各オブジェクトバージョンに使用される暗号化キーを追跡するのはユーザーの責任です。
- 暗号化キーはクライアント側で管理するため、キーのローテーションなどの追加の安全対策もクライアント側で管理する必要があります。



提供された暗号化キーは保存されません。暗号化キーを紛失すると、対応するオブジェクトも失われます。

- バケットにクロスグリッド レプリケーションまたは CloudMirror レプリケーションが設定されている場合、SSE-C オブジェクトを取り込むことはできません。取り込み操作は失敗します。

#### 関連情報

["Amazon S3 ユーザーガイド: 顧客提供キーによるサーバー側暗号化 \(SSE-C\) の使用"](#)

### CopyObject

S3 CopyObject リクエストを使用して、S3 にすでに保存されているオブジェクトのコピーを作成できます。CopyObject 操作は、GetObject に続いて PutObject を実行するのと同じです。

#### 紛争を解決する

2つのクライアントが同じキーに書き込むなど、競合するクライアント要求は、「最新のを優先」して解決されます。「最新のを優先」評価のタイミングは、S3 クライアントが操作を開始した時点ではなく、StorageGRIDシステムが特定のリクエストを完了した時点に基づきます。

#### オブジェクトのサイズ

1回の PutObject 操作の最大推奨サイズは 5 GiB (5,368,709,120 バイト) です。5GiBを超えるオブジェクトがある場合は、["マルチパートアップロード"](#)その代わり。

1回の PutObject 操作でサポートされる最大サイズは 5 TiB (5,497,558,138,880 バイト) です。



StorageGRID 11.6 以前からアップグレードした場合、5 GiB を超えるオブジェクトをアップロードしようとする、S3 PUT オブジェクトのサイズが大きすぎるというアラートがトリガーされます。StorageGRID 11.7 または 11.8 を新規にインストールした場合、この場合アラートはトリガーされません。ただし、AWS S3 標準に準拠するため、StorageGRIDの将来のリリースでは 5 GiB を超えるオブジェクトのアップロードはサポートされません。

#### ユーザーメタデータの UTF-8 文字

リクエストに、キー名またはユーザー定義メタデータの値に (エスケープされていない) UTF-8 値が含まれている場合、StorageGRID の動作は未定義になります。

StorageGRID は、ユーザー定義メタデータのキー名または値に含まれるエスケープされた UTF-8 文字を解析または解釈しません。エスケープされた UTF-8 文字は ASCII 文字として扱われます。

- ユーザー定義のメタデータにエスケープされた UTF-8 文字が含まれている場合、リクエストは成功しま

す。

- StorageGRIDは `x-amz-missing-meta` キー名または値の解釈された値に印刷できない文字が含まれている場合は、ヘッダーにエラーが発生します。

サポートされているリクエストヘッダー

次のリクエスト ヘッダーがサポートされています。

- 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-、その後ユーザー定義のメタデータを含む名前と値のペアが続く
- x-amz-metadata-directive: デフォルト値は `COPY` これにより、オブジェクトと関連メタデータをコピーできるようになります。

指定できます `REPLACE` オブジェクトをコピーするときに既存のメタデータを上書きするか、オブジェクトのメタデータを更新します。

- x-amz-storage-class
- x-amz-tagging-directive: デフォルト値は `COPY` これにより、オブジェクトとすべてのタグをコピーできるようになります。

指定できます `REPLACE` オブジェクトをコピーするときに既存のタグを上書きしたり、タグを更新したりします。

- S3 オブジェクトロックリクエストヘッダー:

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

これらのヘッダーなしでリクエストが行われた場合、バケットのデフォルトの保持設定を使用して、オブジェクト バージョン モードと保持期限が計算されます。見る "[S3 REST API を使用して S3 オブジェクトロックを設定する](#)"。

- SSE リクエスト ヘッダー:

- 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

見る[\[サーバー側暗号化のリクエストヘッダー\]](#)

サポートされていないリクエストヘッダー

次のリクエスト ヘッダーはサポートされていません。

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

オブジェクトをコピーするときに、ソース オブジェクトにチェックサムがある場合、StorageGRID はそのチェックサム値を新しいオブジェクトにコピーしません。この動作は、使用しようとしたかどうかに関係なく適用されます。`x-amz-checksum-algorithm` オブジェクト要求内。

- x-amz-website-redirect-location

ストレージクラスのオプション

その `x-amz-storage-class` リクエストヘッダーがサポートされており、一致する ILM ルールがデュアルコミットまたはバランスコミットを使用している場合に StorageGRID が作成するオブジェクトコピーの数に影響します。"[取り込みオプション](#)"。

- STANDARD

(デフォルト) ILM ルールがデュアル コミット オプションを使用する場合、またはバランス オプションが中間コピーの作成にフォールバックする場合に、デュアル コミット インジエスト操作を指定します。

- REDUCED\_REDUNDANCY

ILM ルールがデュアル コミット オプションを使用する場合、またはバランス オプションが中間コピーの作成にフォールバックする場合に、単一コミットの取り込み操作を指定します。



S3 オブジェクトロックが有効になっているバケットにオブジェクトを取り込む場合、`REDUCED\_REDUNDANCY` オプションは無視されます。レガシーコンプライアンスバケットにオブジェクトを取り込む場合、`REDUCED\_REDUNDANCY` オプションはエラーを返します。StorageGRID は、コンプライアンス要件が満たされていることを確認するために、常にデュアルコミット取り込みを実行します。

**CopyObject** で **x-amz-copy-source** を使用する

ソースバケットとキーが `x-amz-copy-source` ヘッダー、値、およびキーが宛先バケットとキーと異なる場合、ソース オブジェクト データのコピーが宛先に書き込まれます。

送信元と送信先が一致し、`x-amz-metadata-directive`ヘッダーは次のように指定されます。`REPLACE`オブジェクトのメタデータは、リクエストで提供されたメタデータ値で更新されます。この場合、StorageGRIDはオブジェクトを再取り込みしません。これには2つの重要な結果があります。

- CopyObjectを使用して、既存のオブジェクトをその場で暗号化したり、既存のオブジェクトの暗号化を変更したりすることはできません。あなたが`x-amz-server-side-encryption`ヘッダーまたは`x-amz-server-side-encryption-customer-algorithm`ヘッダー、StorageGRIDはリクエストを拒否し、`XNotImplemented`。
- 一致するILMルールで指定された取り込み動作のオプションは使用されません。更新によってトリガーされるオブジェクト配置の変更は、通常のバックグラウンドILMプロセスによってILMが再評価される時に行われます。

つまり、ILMルールが取り込み動作に厳密オプションを使用する場合、必要なオブジェクトを配置できない場合(たとえば、新しく必要な場所が利用できないなど)、アクションは実行されません。更新されたオブジェクトは、必要な配置が可能になるまで現在の配置を保持します。

#### サーバー側暗号化のリクエストヘッダー

もしあなたが"[サーバー側の暗号化を使用する](#)"指定するリクエストヘッダーは、ソースオブジェクトが暗号化されているかどうか、およびターゲットオブジェクトを暗号化する予定があるかどうかによって異なります。

- ソースオブジェクトが顧客提供のキー(SSE-C)を使用して暗号化されている場合は、オブジェクトを復号化してコピーできるように、CopyObject要求に次の3つのヘッダーを含める必要があります。
  - `x-amz-copy-source-server-side-encryption-customer-algorithm`：特定AES256。
  - `x-amz-copy-source-server-side-encryption-customer-key`：ソースオブジェクトの作成時に指定した暗号化キーを指定します。
  - `x-amz-copy-source-server-side-encryption-customer-key-MD5`：ソースオブジェクトの作成時に指定したMD5ダイジェストを指定します。
- 提供および管理する一意のキーを使用してターゲットオブジェクト(コピー)を暗号化する場合は、次の3つのヘッダーを含めます。
  - `x-amz-server-side-encryption-customer-algorithm`：特定AES256。
  - `x-amz-server-side-encryption-customer-key`：対象オブジェクトの新しい暗号化キーを指定します。
  - `x-amz-server-side-encryption-customer-key-MD5`：新しい暗号化キーのMD5ダイジェストを指定します。



提供された暗号化キーは保存されません。暗号化キーを紛失すると、対応するオブジェクトも失われます。顧客提供のキーを使用してオブジェクトデータを保護する前に、以下の考慮事項を確認してください。"[サーバー側暗号化を使用する](#)"。

- StorageGRID(SSE)によって管理される一意のキーを使用してターゲットオブジェクト(コピー)を暗号化する場合は、CopyObject要求に次のヘッダーを含めます。
  - `x-amz-server-side-encryption`



その `server-side-encryption` オブジェクトの値を更新できません。代わりに、新しいコピーを作成してください。`server-side-encryption` 値を使用する `x-amz-metadata-directive: REPLACE`。

## バージョン管理

ソースバケットがバージョン管理されている場合は、`x-amz-copy-source` オブジェクトの最新バージョンをコピーするためのヘッダー。オブジェクトの特定のバージョンをコピーするには、`versionId` サブリソース。宛先バケットがバージョン管理されている場合、生成されたバージョンは `x-amz-version-id` レスポンスヘッダー。ターゲットバケットのバージョン管理が停止されている場合は、`x-amz-version-id` 「null」値を返します。

## GetObject

S3 GetObject リクエストを使用して、S3 バケットからオブジェクトを取得できます。

### GetObjectとマルチパートオブジェクト

使用することができます `partNumber` マルチパートまたはセグメント化されたオブジェクトの特定の部分を取得するためのリクエストパラメータ。その `x-amz-mp-parts-count` レスポンス要素は、オブジェクトに含まれるパーツの数を示します。

設定できます `partNumber` セグメント化/マルチパートオブジェクトと非セグメント化/非マルチパートオブジェクトの両方で1になりますが、`x-amz-mp-parts-count` 応答要素は、セグメント化されたオブジェクトまたはマルチパートオブジェクトに対してのみ返されます。

### ユーザーメタデータの UTF-8 文字

StorageGRID は、ユーザー定義のメタデータ内のエスケープされた UTF-8 文字を解析または解釈しません。ユーザー定義のメタデータにエスケープされた UTF-8 文字を含むオブジェクトに対する GET リクエストは、`x-amz-missing-meta` キー名または値に印刷できない文字が含まれている場合は、ヘッダーを作成します。

### サポートされているリクエストヘッダー

次のリクエストヘッダーがサポートされています。

- `x-amz-checksum-mode` : 特定 ENABLED

その `Range` ヘッダーはサポートされていません `x-amz-checksum-mode` GetObject 用。含める場合 `Range` リクエストに `x-amz-checksum-mode` 有効にすると、StorageGRID は応答でチェックサム値を返しません。

### サポートされていないリクエストヘッダー

次のリクエストヘッダーはサポートされていないため、XNotImplemented:

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

## バージョン管理

もし `versionId` サブリソースが指定されていない場合、この操作ではバージョン管理されたバケット内の

オブジェクトの最新バージョンが取得されます。オブジェクトの現在のバージョンが削除マーカである場合、「見つかりません」ステータスが返されます。`x-amz-delete-marker`レスポンスヘッダーを`true`。

#### 顧客提供の暗号化キーを使用したサーバー側暗号化のリクエスト ヘッダー (SSE-C)

オブジェクトが指定した一意のキーで暗号化されている場合は、3つのヘッダーすべてを使用します。

- `x-amz-server-side-encryption-customer-algorithm`: 特定 AES256。
- `x-amz-server-side-encryption-customer-key`: オブジェクトの暗号化キーを指定します。
- `x-amz-server-side-encryption-customer-key-MD5`: オブジェクトの暗号化キーの MD5 ダイジェストを指定します。



提供された暗号化キーは保存されません。暗号化キーを紛失すると、対応するオブジェクトも失われます。顧客提供のキーを使用してオブジェクトデータを保護する前に、以下の考慮事項を確認してください。["サーバー側の暗号化を使用する"](#)。

#### Cloud Storage Pool オブジェクトに対する `GetObject` の動作

オブジェクトが["クラウド ストレージ プール"](#)`GetObject` 要求の動作はオブジェクトの状態によって異なります。見る["HeadObject"](#)詳細についてはこちらをご覧ください。



オブジェクトがクラウド ストレージ プールに格納され、そのオブジェクトの1つ以上のコピーがグリッド上にも存在する場合、`GetObject` リクエストは、クラウド ストレージ プールからデータを取得する前に、グリッドからデータを取得しようとします。

オブジェクトの状態	<code>GetObject</code> の動作
StorageGRIDに取り込まれたが ILM によってまだ評価されていないオブジェクト、または従来のストレージ プールに保存されているか消去コーディングを使用しているオブジェクト	200 OK  オブジェクトのコピーが取得されます。
Cloud Storage Pool 内のオブジェクトですが、まだ取得不可の状態に移行していません	200 OK  オブジェクトのコピーが取得されます。
オブジェクトは取得不可能な状態に移行しました	403 Forbidden、InvalidObjectState  使用 <a href="#">"オブジェクトの復元"</a> オブジェクトを取得可能な状態に復元するよう要求します。
回復不可能な状態から復元中のオブジェクト	403 Forbidden、InvalidObjectState  RestoreObject 要求が完了するまで待ちます。

オブジェクトの状態	GetObjectの動作
オブジェクトがクラウド ストレージ プールに完全に復元されました	200 OK  オブジェクトのコピーが取得されます。

クラウド ストレージ プール内のマルチパートまたはセグメント化されたオブジェクト

マルチパート オブジェクトをアップロードした場合、またはStorageGRID が大きなオブジェクトをセグメントに分割した場合、StorageGRID はオブジェクトのパートまたはセグメントのサブセットをサンプリングして、オブジェクトが Cloud Storage Pool で使用可能かどうかを判断します。場合によっては、GetObjectリクエストが誤って返されることがあります。`200 OK`オブジェクトの一部がすでに回復不可能な状態に移行している場合、またはオブジェクトの一部がまだ復元されていない場合。

このような場合:

- GetObject 要求はデータを返す可能性があります、転送の途中で停止することがあります。
- 後続のGetObjectリクエストは、403 Forbidden。

#### GetObjectとクロスグリッドレプリケーション

使用している場合"**グリッドフェデレーション**"そして"**クロスグリッドレプリケーション**"バケットに対して有効になっている場合、S3 クライアントは GetObject リクエストを発行してオブジェクトのレプリケーションステータスを確認できます。応答にはStorageGRID固有の `x-ntap-sg-cgr-replication-status` レスポンス ヘッダーは次のいずれかの値を持ちます。

Grid	レプリケーションステータス
ソース	<ul style="list-style-type: none"> <li>• 完了: レプリケーションは成功しました。</li> <li>• 保留中: オブジェクトはまだ複製されていません。</li> <li>• 失敗: レプリケーションは永続的な障害により失敗しました。ユーザーはエラーを解決する必要があります。</li> </ul>
デスティネーション	<b>REPLICA:</b> オブジェクトはソース グリッドから複製されました。



StorageGRIDはサポートしていません `x-amz-replication-status` ヘッダー。

#### HeadObject

S3 HeadObject リクエストを使用すると、オブジェクト自体を返さずにオブジェクトからメタデータを取得できます。オブジェクトが Cloud Storage プールに保存されている場合は、HeadObject を使用してオブジェクトの遷移状態を判別できます。

#### HeadObjectとマルチパートオブジェクト

使用することができます `partNumber` マルチパートまたはセグメント化されたオブジェクトの特定の部分のメタデータを取得するためのリクエスト パラメータ。その `x-amz-mp-parts-count` レスポンス要素は、オブジェクトに含まれるパーツの数を示します。

設定できます `partNumber` セグメント化/マルチパートオブジェクトと非セグメント化/非マルチパートオブジェクトの両方で1になりますが、`x-amz-mp-parts-count` 応答要素は、セグメント化されたオブジェクトまたはマルチパート オブジェクトに対してのみ返されます。

#### ユーザーメタデータの UTF-8 文字

StorageGRID は、ユーザー定義のメタデータ内のエスケープされた UTF-8 文字を解析または解釈しません。ユーザー定義のメタデータにエスケープされた UTF-8 文字を含むオブジェクトに対する HEAD リクエストは、`x-amz-missing-meta` キー名または値に印刷できない文字が含まれている場合は、ヘッダーを作成します。

#### サポートされているリクエストヘッダー

次のリクエスト ヘッダーがサポートされています。

- `x-amz-checksum-mode`

その `partNumber` パラメータと `Range` ヘッダーはサポートされていません `x-amz-checksum-mode` HeadObject 用。リクエストにそれらを含めると `x-amz-checksum-mode` 有効にすると、StorageGRID は応答でチェックサム値を返しません。

#### サポートされていないリクエストヘッダー

次のリクエストヘッダーはサポートされていないため、XNotImplemented:

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

#### バージョン管理

もし `versionId` サブリソースが指定されていない場合、この操作ではバージョン管理されたバケット内のオブジェクトの最新バージョンが取得されます。オブジェクトの現在のバージョンが削除マーカである場合、「見つかりません」ステータスが返されます。`x-amz-delete-marker` レスポンスヘッダーを `true`。

#### 顧客提供の暗号化キーを使用したサーバー側暗号化のリクエスト ヘッダー (SSE-C)

オブジェクトが指定した一意のキーで暗号化されている場合は、これら 3 つのヘッダーをすべて使用します。

- `x-amz-server-side-encryption-customer-algorithm`: 特定 AES256。
- `x-amz-server-side-encryption-customer-key`: オブジェクトの暗号化キーを指定します。
- `x-amz-server-side-encryption-customer-key-MD5`: オブジェクトの暗号化キーの MD5 ダイジェストを指定します。



提供された暗号化キーは保存されません。暗号化キーを紛失すると、対応するオブジェクトも失われます。顧客提供のキーを使用してオブジェクトデータを保護する前に、以下の考慮事項を確認してください。"[サーバー側の暗号化を使用する](#)"。

#### Cloud Storage Pool オブジェクトの HeadObject 応答

オブジェクトが"[クラウド ストレージ プール](#)"、次の応答ヘッダーが返されます。

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

レスポンス ヘッダーには、オブジェクトが Cloud Storage プールに移動され、オプションで取得不可能な状態に移行され、復元される際のオブジェクトの状態に関する情報が提供されます。

オブジェクトの状態	HeadObjectへの応答
StorageGRIDに取り込まれたが ILM によってまだ評価されていないオブジェクト、または従来のストレージ プールに保存されているか消去コーディングを使用しているオブジェクト	200 OK(特別な応答ヘッダーは返されません。)
Cloud Storage Pool 内のオブジェクトですが、まだ取得不可の状態に移行していません	200 OK  x-amz-storage-class: GLACIER  x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"  オブジェクトが取得不可能な状態に移行するまで、`expiry-date` 遠い将来のある時点に設定されています。移行の正確な時間はStorageGRIDシステムによって制御されません。
オブジェクトは取得不可能な状態に移行しましたが、少なくとも1つのコピーがグリッド上に存在します	200 OK  x-amz-storage-class: GLACIER  x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"  の価値 `expiry-date` 遠い将来のある時点に設定されています。  注意: グリッド上のコピーが利用できない場合 (たとえば、ストレージノードがダウンしている場合)、" <a href="#">オブジェクトの復元</a> "オブジェクトを正常に取得するには、クラウド ストレージ プールからコピーを復元するようにリクエストする必要があります。
オブジェクトは取得不可能な状態に移行し、グリッド上にコピーが存在しません	200 OK  x-amz-storage-class: GLACIER

オブジェクトの状態	HeadObjectへの応答
回復不可能な状態から復元中のオブジェクト	<pre>200 OK  x-amz-storage-class: GLACIER  x-amz-restore: ongoing-request="true"</pre>
オブジェクトがクラウド ストレージ プールに完全に復元されました	<pre>200 OK  x-amz-storage-class: GLACIER  x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2018 00:00:00 GMT"  その `expiry-date` クラウド ストレージ プール内のオブジェクトが取得不可能な状態に戻されるタイミングを示します。</pre>

クラウド ストレージ プール内のマルチパートまたはセグメント化されたオブジェクト

マルチパート オブジェクトをアップロードした場合、またはStorageGRID が大きなオブジェクトをセグメントに分割した場合、StorageGRID はオブジェクトのパートまたはセグメントのサブセットをサンプリングして、オブジェクトが Cloud Storage Pool で使用可能かどうかを判断します。場合によっては、HeadObjectリクエストが誤って返されることがあります。`x-amz-restore: ongoing-request="false"` オブジェクトの一部がすでに回復不可能な状態に移行している場合、またはオブジェクトの一部がまだ復元されていない場合。

#### HeadObjectとクロスグリッドレプリケーション

使用している場合"[グリッドフェデレーション](#)"そして"[クロスグリッドレプリケーション](#)"バケットに対して有効になっている場合、S3 クライアントは HeadObject リクエストを発行してオブジェクトのレプリケーション ステータスを確認できます。応答にはStorageGRID固有の `x-ntap-sg-cgr-replication-status` レスポンス ヘッダーは次のいずれかの値を持ちます。

Grid	レプリケーションステータス
ソース	<ul style="list-style-type: none"> <li>完了: レプリケーションは成功しました。</li> <li>保留中: オブジェクトはまだ複製されていません。</li> <li>失敗: レプリケーションは永続的な障害により失敗しました。ユーザーはエラーを解決する必要があります。</li> </ul>
デスティネーション	<b>REPLICA:</b> オブジェクトはソース グリッドから複製されました。



StorageGRIDはサポートしていません `x-amz-replication-status` ヘッダ。

## PutObject

S3 PutObject リクエストを使用して、バケットにオブジェクトを追加できます。

### 紛争を解決する

2 つのクライアントが同じキーに書き込むなど、競合するクライアント要求は、「最新のを優先」して解決されます。「最新のを優先」評価のタイミングは、S3 クライアントが操作を開始した時点ではなく、StorageGRID システムが特定のリクエストを完了した時点に基づきます。

### オブジェクトのサイズ

1 回の PutObject 操作の最大 推奨 サイズは 5 GiB (5,368,709,120 バイト) です。5GiB を超えるオブジェクトがある場合は、"[マルチパートアップロード](#)"その代わり。

1 回の PutObject 操作でサポートされる最大サイズは 5 TiB (5,497,558,138,880 バイト) です。



StorageGRID 11.6 以前からアップグレードした場合、5 GiB を超えるオブジェクトをアップロードしようとする、S3 PUT オブジェクトのサイズが大きすぎるというアラートがトリガーされます。StorageGRID 11.7 または 11.8 を新規にインストールした場合、この場合アラートはトリガーされません。ただし、AWS S3 標準に準拠するため、StorageGRID の将来のリリースでは 5 GiB を超えるオブジェクトのアップロードはサポートされません。

### ユーザーメタデータのサイズ

Amazon S3 では、各 PUT リクエスト ヘッダー内のユーザー定義メタデータのサイズが 2 KB に制限されています。StorageGRID はユーザー メタデータを 24 KiB に制限します。ユーザー定義メタデータのサイズは、各キーと値の UTF-8 エンコードのバイト数の合計によって測定されます。

### ユーザーメタデータの UTF-8 文字

リクエストに、キー名またはユーザー定義メタデータの値に (エスケープされていない) UTF-8 値が含まれている場合、StorageGRID の動作は未定義になります。

StorageGRID は、ユーザー定義メタデータのキー名または値に含まれるエスケープされた UTF-8 文字を解析または解釈しません。エスケープされた UTF-8 文字は ASCII 文字として扱われます。

- ユーザー定義のメタデータにエスケープされた UTF-8 文字が含まれている場合、PutObject、CopyObject、GetObject、および HeadObject 要求は成功します。
- StorageGRID は `x-amz-missing-meta` キー名または値の解釈された値に印刷できない文字が含まれている場合は、ヘッダーにエラーが発生します。

### オブジェクトタグの制限

新しいオブジェクトをアップロードするときにタグを追加したり、既存のオブジェクトにタグを追加したりできます。StorageGRID と Amazon S3 はどちらも、オブジェクトごとに最大 10 個のタグをサポートします。オブジェクトに関連付けられたタグには、一意のタグ キーが必要です。タグ キーの長さは最大 128 文字の Unicode 文字、タグ値の長さは最大 256 文字の Unicode 文字です。キーと値は大文字と小文字が区別されません。

## オブジェクトの所有権

StorageGRIDでは、所有者以外のアカウントまたは匿名ユーザーによって作成されたオブジェクトも含め、すべてのオブジェクトはバケット所有者アカウントによって所有されます。

サポートされているリクエストヘッダー

次のリクエストヘッダーがサポートされています。

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

指定する場合 `aws-chunked` のために Content-Encoding StorageGRID は次の項目を検証しません。

- StorageGRIDは、`chunk-signature` チャンクデータに対して。
- StorageGRIDは、指定された値を検証しません。`x-amz-decoded-content-length` オブジェクトに対して。
- Content-Language
- Content-Length
- Content-MD5
- Content-Type
- Expires
- Transfer-Encoding

チャンク転送エンコーディングは、以下の場合にサポートされます。`aws-chunked` ペイロード署名も使用されます。

- x-amz-checksum-sha256
- x-amz-meta-、その後ユーザー定義のメタデータを含む名前と値のペアが続きます。

ユーザー定義のメタデータの名前と値のペアを指定する場合は、次の一般的な形式を使用します。

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

ILMルールの参照時間として\*ユーザー定義作成時間\*オプションを使用する場合は、`creation-time` オブジェクトが作成された日時を記録するメタデータの名前として。例えば：

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

の価値 `creation-time` 1970 年 1 月 1 日からの秒数として評価されます。



ILM ルールでは、参照時間の ユーザー定義の作成時間 と、バランスまたは厳密な取り込みオプションの両方を使用することはできません。ILM ルールの作成時にエラーが返されません。

- `x-amz-tagging`
- S3 オブジェクトロックリクエストヘッダー
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

これらのヘッダーなしでリクエストが行われた場合、バケットのデフォルトの保持設定を使用して、オブジェクトバージョンモードと保持期限が計算されます。見る["S3 REST API を使用して S3 オブジェクトロックを設定する"](#)。

- SSE リクエストヘッダー:
  - `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`

見る[\[サーバー側暗号化のリクエストヘッダー\]](#)

サポートされていないリクエストヘッダー

次のリクエストヘッダーはサポートされていません。

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

その `x-amz-website-redirect-location``ヘッダーリターン ``XNotImplemented`。

ストレージクラスのオプション

その ``x-amz-storage-class``リクエストヘッダーがサポートされています。提出された値は ``x-amz-storage-class`StorageGRID` が取り込み中にオブジェクトデータを保護する方法に影響しますが、オブジェクトの永続コピーがStorageGRIDシステムにいくつ保存されるかには影響しません (これは ILM によって決定されます)。

取り込まれたオブジェクトに一致するILMルールが厳密な取り込みオプションを使用している場合、``x-amz-storage-class``ヘッダーは効果がありません。

以下の値は、`x-amz-storage-class`` :

- STANDARD (デフォルト)

- デュアル コミット: ILM ルールで取り込み動作にデュアル コミット オプションが指定されている場合は、オブジェクトが取り込まれるとすぐにそのオブジェクトの 2 番目のコピーが作成され、別のストレージ ノードに配布されます (デュアル コミット)。ILM が評価されると、StorageGRID はこれらの初期の中間コピーがルール内の配置指示を満たしているかどうかを判断します。そうでない場合は、別の場所に新しいオブジェクトのコピーを作成し、初期の中間コピーを削除する必要があるかもしれません。
- バランス: ILM ルールでバランス オプションが指定されていて、StorageGRID がルールで指定されたすべてのコピーをすぐに作成できない場合、StorageGRID は異なるストレージ ノードに 2 つの中間コピーを作成します。

StorageGRIDがILMルールで指定されたすべてのオブジェクトコピーを即時に作成できる場合 (同期配置)、`x-amz-storage-class`ヘッダーは効果がありません。

- REDUCED\_REDUNDANCY

- デュアル コミット: ILM ルールで取り込み動作にデュアル コミット オプションが指定されている場合、StorageGRID はオブジェクトの取り込み時に単一の中間コピーを作成します (単一コミット)。
- バランス: ILM ルールでバランス オプションが指定されている場合、StorageGRID は、システムがルールで指定されたすべてのコピーをすぐに作成できない場合にのみ、単一の中間コピーを作成します。StorageGRID が同期配置を実行できる場合、このヘッダーは効果がありません。その `REDUCED\_REDUNDANCY` このオプションは、オブジェクトに一致する ILM ルールによって単一の複製コピーが作成される場合に最適です。この場合、`REDUCED\_REDUNDANCY` すべての取り込み操作で余分なオブジェクト コピーが不必要に作成および削除されるのを防ぎます。

使用して `REDUCED\_REDUNDANCY` 他の状況ではこのオプションは推奨されません。

`REDUCED\_REDUNDANCY` 取り込み中にオブジェクト データが失われるリスクが高まります。たとえば、ILM 評価が行われる前に障害が発生したストレージ ノードに単一のコピーが最初に保存された場合、データが失われる可能性があります。



任意の期間に複製されたコピーが 1 つしかない場合、データが永久に失われるリスクがあります。オブジェクトの複製されたコピーが 1 つしか存在しない場合、ストレージ ノードに障害が発生したり重大なエラーが発生すると、そのオブジェクトは失われます。また、アップグレードなどのメンテナンス手順中は、オブジェクトへのアクセス権が一時的に失われます。

指定 `REDUCED\_REDUNDANCY` オブジェクトが最初に取り込まれたときに作成されるコピーの数にのみ影響します。これは、オブジェクトがアクティブな ILM ポリシーによって評価されるときに作成されるオブジェクトのコピー数には影響せず、StorageGRIDシステムに低いレベルの冗長性でデータが保存されることにはなりません。



S3オブジェクトロックが有効になっているバケットにオブジェクトを取り込む場合、`REDUCED\_REDUNDANCY` オプションは無視されます。レガシーコンプライアンスバケットにオブジェクトを取り込む場合、`REDUCED\_REDUNDANCY` オプションはエラーを返します。StorageGRID は、コンプライアンス要件が満たされていることを確認するために、常にデュアルコミット取り込みを実行します。

#### サーバー側暗号化のリクエストヘッダー

次のリクエスト ヘッダーを使用して、サーバー側暗号化でオブジェクトを暗号化できます。SSE オプションと SSE-C オプションは相互に排他的です。

- **SSE:** StorageGRIDによって管理される一意のキーを使用してオブジェクトを暗号化する場合は、次のヘッダーを使用します。
  - `x-amz-server-side-encryption`

いつ ``x-amz-server-side-encryption`PutObject` リクエストにヘッダーが含まれていない場合、グリッド全体の["保存オブジェクトの暗号化設定"](#)PutObject 応答から省略されます。
- **SSE-C:** 提供および管理する一意のキーを使用してオブジェクトを暗号化する場合は、これら 3 つのヘッダーをすべて使用します。
  - `x-amz-server-side-encryption-customer-algorithm`: 特定 AES256。
  - `x-amz-server-side-encryption-customer-key`: 新しいオブジェクトの暗号化キーを指定します。
  - `x-amz-server-side-encryption-customer-key-MD5`: 新しいオブジェクトの暗号化キーの MD5 ダイジェストを指定します。



提供された暗号化キーは保存されません。暗号化キーを紛失すると、対応するオブジェクトも失われます。顧客提供のキーを使用してオブジェクトデータを保護する前に、以下の考慮事項を確認してください。["サーバー側暗号化を使用する"](#)。



オブジェクトが SSE または SSE-C で暗号化されている場合、バケット レベルまたはグリッドレベルの暗号化設定はすべて無視されます。

## バージョン管理

バケットのバージョン管理が有効になっている場合、一意の ``versionId`` 保存されるオブジェクトのバージョンに応じて自動的に生成されます。これ ``versionId`` レスポンスでは、``x-amz-version-id`` レスポンス ヘッダー。

バージョン管理が中断されている場合、オブジェクトのバージョンは null で保存されます。``versionId`` nul バージョンがすでに存在する場合は上書きされます。

## Authorizationヘッダーの署名計算

使用する際は ``Authorization`` ヘッダーを使用してリクエストを認証する点において、StorageGRID は AWS と次の点で異なります。

- StorageGRIDは `host`` 含まれるヘッダー ``CanonicalHeaders``。
- StorageGRIDは `Content-Type`` 含まれる ``CanonicalHeaders``。
- StorageGRIDは `x-amz-*`` 含まれるヘッダー ``CanonicalHeaders``。



一般的なベストプラクティスとして、これらのヘッダーを常に含めてください。``CanonicalHeaders`` 確実に検証するためです。ただし、これらのヘッダーを除外すると、StorageGRID はエラーを返しません。

詳細については、["認証ヘッダーの署名計算: ペイロードを単一チャンクで転送する \(AWS 署名バージョン 4\)"](#)。

## 関連情報

- ["ILMでオブジェクトを管理する"](#)
- ["Amazon Simple Storage Service API リファレンス: PutObject"](#)

## オブジェクトの復元

S3 RestoreObject リクエストを使用して、クラウド ストレージ プールに保存されているオブジェクトを復元できます。

### サポートされているリクエストタイプ

StorageGRID は、オブジェクトを復元するための RestoreObject 要求のみをサポートします。サポートされていません SELECT `修復の種類。選択したリクエストの戻り` `XNotImplemented。

### バージョン管理

オプションで指定 `versionId`バージョン管理されたバケット内のオブジェクトの特定のバージョンを復元します。指定しない場合 `versionId`オブジェクトの最新バージョンが復元されます

### Cloud Storage Pool オブジェクトにおける RestoreObject の動作

オブジェクトが["クラウド ストレージ プール"](#)RestoreObject 要求は、オブジェクトの状態に基づいて次の動作をします。見る["HeadObject"](#)詳細についてはこちらをご覧ください。



オブジェクトがクラウド ストレージ プールに保存され、そのオブジェクトの1つ以上のコピーがグリッド上に存在する場合は、RestoreObject リクエストを発行してオブジェクトを復元する必要はありません。代わりに、GetObject 要求を使用してローカル コピーを直接取得できます。

オブジェクトの状態	RestoreObjectの動作
オブジェクトはStorageGRIDに取り込まれましたが、ILMによってまだ評価されていません。または、オブジェクトが Cloud Storage Pool にありません。	403 Forbidden、InvalidObjectState
Cloud Storage Pool 内のオブジェクトですが、まだ取得不可の状態に移行していません	`200 OK`変更は行われません。 注意: オブジェクトが取得不可能な状態に移行する前に、そのオブジェクトを変更することはできません。 expiry-date 。

オブジェクトの状態	RestoreObjectの動作
オブジェクトは取得不可能な状態に移行しました	<p>`202 Accepted` リクエスト本文で指定された日数の間、オブジェクトの取得可能なコピーを Cloud Storage プールに復元します。この期間が終了すると、オブジェクトは取得不可能な状態に戻されます。</p> <p>オプションとして、Tier`復元ジョブの完了にかかる時間を決定するためのリクエスト要素(`Expedited、Standard、またはBulk)。指定しない場合 Tier、`Standard`ティアが使用されます。</p> <p>重要: オブジェクトがS3 Glacier Deep Archiveに移行されている場合、またはクラウドストレージプールがAzure Blobストレージを使用している場合は、Expedited`層。次のエラーが返されます `403 Forbidden、InvalidTier:Retrieval option is not supported by this storage class。</p>
回復不可能な状態から復元中のオブジェクト	409 Conflict、RestoreAlreadyInProgress
オブジェクトがクラウド ストレージ プールに完全に復元されました	<p>200 OK</p> <p>注: オブジェクトが取得可能な状態に復元された場合、その expiry-date`新しい値でRestoreObjectリクエストを再発行することにより、`Days。復元日付はリクエストの時刻に応じて更新されます。</p>

## オブジェクトコンテンツの選択

S3 SelectObjectContent リクエストを使用すると、単純な SQL ステートメントに基づいて S3 オブジェクトの内容をフィルタリングできます。

詳細については、"[Amazon Simple Storage Service API リファレンス: SelectObjectContent](#)"。

### 開始する前に

- テナント アカウントには S3 Select 権限があります。
- あなたが持っている `s3:GetObject` クエリを実行するオブジェクトに対する権限。
- クエリするオブジェクトは、次のいずれかの形式である必要があります。
  - **CSV**。そのまま使用することも、GZIP または BZIP2 アーカイブに圧縮することもできます。
  - 寄木細工。 Parquet オブジェクトの追加要件:
    - S3 Select は、GZIP または Snappy を使用した列指向の圧縮のみをサポートします。 S3 Select は、Parquet オブジェクトのオブジェクト全体の圧縮をサポートしていません。
    - S3 Select は Parquet 出力をサポートしていません。出力形式を CSV または JSON として指定する必要があります。
    - 圧縮されていない行グループの最大サイズは 512 MB です。
    - オブジェクトのスキーマで指定されたデータ型を使用する必要があります。
    - INTERVAL、JSON、LIST、TIME、または UUID 論理型は使用できません。

- SQL 式の最大長は 256 KB です。
- 入力または結果のレコードの最大長は 1 MiB です。

#### CSVリクエスト構文の例

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

## Parquetリクエスト構文の例

```
POST /{Key+}?select&select-type=2 HTTP/1.1
Host: Bucket.s3.abc-company.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<SelectObjectContentRequest xmlns=http://s3.amazonaws.com/doc/2006-03-01/>
  <Expression>string</Expression>
  <ExpressionType>string</ExpressionType>
  <RequestProgress>
    <Enabled>boolean</Enabled>
  </RequestProgress>
  <InputSerialization>
    <CompressionType>GZIP</CompressionType>
    <PARQUET>
    </PARQUET>
  </InputSerialization>
  <OutputSerialization>
    <CSV>
      <FieldDelimiter>string</FieldDelimiter>
      <QuoteCharacter>string</QuoteCharacter>
      <QuoteEscapeCharacter>string</QuoteEscapeCharacter>
      <QuoteFields>string</QuoteFields>
      <RecordDelimiter>string</RecordDelimiter>
    </CSV>
  </OutputSerialization>
  <ScanRange>
    <End>long</End>
    <Start>long</Start>
  </ScanRange>
</SelectObjectContentRequest>
```

## SQLクエリの例

このクエリは、米国国勢調査データから州名、2010年の人口、2015年の推定人口、および変化率を取得します。ファイル内の状態ではないレコードは無視されます。

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

照会するファイルの最初の数行、SUB-EST2020\_ALL.csv、次のようになります。

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

#### AWS-CLI の使用例 (CSV)

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

出力ファイルの最初の数行は、changes.csv、次のようになります。

```
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
```

## AWS-CLI の使用例 (Parquet)

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

出力ファイル changes.csv の最初の数行は次のようになります。

```
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
```

## マルチパートアップロードの操作

### マルチパートアップロードの操作

このセクションでは、StorageGRID がマルチパートアップロードの操作をサポートする方法について説明します。

すべてのマルチパートアップロード操作には、次の条件と注意事項が適用されます。

- バケットに対する ListMultipartUploads クエリの結果が不完全な結果を返す可能性があるため、1つのバケットへの同時マルチパートアップロードは 1,000 件を超えないようにしてください。
- StorageGRID は、マルチパートパーツに対して AWS サイズ制限を適用します。S3 クライアントは次のガイドラインに従う必要があります。
  - マルチパートアップロードの各パーツは、5 MiB (5,242,880 バイト) ~ 5 GiB (5,368,709,120 バイト) の範囲にする必要があります。
  - 最後の部分は 5 MiB (5,242,880 バイト) より小さくても構いません。
  - 一般に、部品のサイズは可能な限り大きくする必要があります。たとえば、100 GiB のオブジェクトには 5 GiB のパーツ サイズを使用します。各パーツは一意的オブジェクトと見なされるため、大きなパーツ サイズを使用すると StorageGRID メタデータのオーバーヘッドが削減されます。
  - 5 GiB 未満のオブジェクトの場合は、代わりに非マルチパートアップロードの使用を検討してください。
- ILM ルールがバランス型または厳密型を使用している場合、ILM は、マルチパートオブジェクトの各部分が取り込まれる際に評価され、マルチパートアップロードが完了するとオブジェクト全体に対して評価されます。**"取り込みオプション"**。これがオブジェクトとパーツの配置にどのように影響するかは注意する必要があります。

- S3 マルチパートアップロードの進行中に ILM が変更されると、マルチパートアップロードが完了したときにオブジェクトの一部が現在の ILM 要件を満たさなくなる可能性があります。正しく配置されていないパーツは ILM 再評価のキューに入れられ、後で正しい場所に移動されます。
- パーツの ILM を評価する際、StorageGRID はオブジェクトのサイズではなくパーツのサイズに基づいてフィルタリングします。つまり、オブジェクトの一部は、オブジェクト全体の ILM 要件を満たさない場所に保存される可能性があります。たとえば、10 GB 以上のすべてのオブジェクトを DC1 に保存し、それより小さいすべてのオブジェクトを DC2 に保存するようにルールで指定すると、10 パートのマルチパートアップロードの各 1 GB の部分は、取り込み時に DC2 に保存されます。ただし、オブジェクト全体に対して ILM が評価されると、オブジェクトのすべての部分が DC1 に移動されます。
- すべてのマルチパートアップロード操作はStorageGRIDをサポートしています"**一貫性値**"。
- マルチパートアップロードを使用してオブジェクトを取り込む場合、"**オブジェクトセグメンテーションしきい値 (1 GiB)**" 適用されません。
- 必要に応じて、"**サーバー側の暗号化**"マルチパートアップロードを使用します。SSE (StorageGRID管理キーによるサーバー側暗号化)を使用するには、`x-amz-server-side-encryption` CreateMultipartUpload リクエスト内のリクエストヘッダーのみ。SSE-C (顧客提供のキーを使用したサーバー側暗号化)を使用するには、CreateMultipartUpload リクエストと後続の各 UploadPart リクエストで同じ 3 つの暗号化キー リクエストヘッダーを指定します。

処理	導入
AbortMultipartUpload	すべての Amazon S3 REST API 動作が実装されています。予告なく変更されることがあります。
CompleteMultipartUpload	見る" <a href="#">CompleteMultipartUpload</a> "
CreateMultipartUpload (以前の名称は「マルチパートアップロードの開始」)	見る" <a href="#">CreateMultipartUpload</a> "
リストマルチパートアップロード	見る" <a href="#">リストマルチパートアップロード</a> "
ListParts	すべての Amazon S3 REST API 動作が実装されています。予告なく変更されることがあります。
UploadPart	見る" <a href="#">UploadPart</a> "
UploadPartCopy	見る" <a href="#">UploadPartCopy</a> "

## CompleteMultipartUpload

CompleteMultipartUpload 操作は、以前にアップロードされた部分を組み立てることで、オブジェクトのマルチパートアップロードを完了します。



StorageGRIDは昇順で連続しない値をサポートします。partNumber CompleteMultipartUpload のリクエスト パラメータ。パラメータは任意の値で始めることができます。

紛争を解決する

2つのクライアントが同じキーに書き込むなど、競合するクライアント要求は、「最新のを優先」して解決されます。「最新のを優先」評価のタイミングは、S3 クライアントが操作を開始した時点ではなく、StorageGRIDシステムが特定のリクエストを完了した時点に基づきます。

サポートされているリクエストヘッダー

次のリクエスト ヘッダーがサポートされています。

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

その `x-amz-storage-class` ヘッダーは、一致するILMルールで指定されている場合にStorageGRIDが作成するオブジェクトコピーの数に影響します。["デュアルコミットまたはバランス取り込みオプション"](#)。

- STANDARD

(デフォルト) ILM ルールがデュアル コミット オプションを使用する場合、またはバランス オプションが中間コピーの作成にフォールバックする場合に、デュアル コミット インジェスト操作を指定します。

- REDUCED\_REDUNDANCY

ILM ルールがデュアル コミット オプションを使用する場合、またはバランス オプションが中間コピーの作成にフォールバックする場合に、単一コミットの取り込み操作を指定します。



S3オブジェクトロックが有効になっているバケットにオブジェクトを取り込む場合、`REDUCED\_REDUNDANCY`オプションは無視されます。レガシーコンプライアンスバケットにオブジェクトを取り込む場合、`REDUCED\_REDUNDANCY`オプションはエラーを返します。StorageGRID は、コンプライアンス要件が満たされていることを確認するために、常にデュアルコミット取り込みを実行します。



マルチパートアップロードが15日以内に完了しない場合、操作は非アクティブとしてマークされ、関連するすべてのデータがシステムから削除されます。



その `ETag` 返される値はデータのMD5合計ではなく、Amazon S3 API実装に従います。`ETag` マルチパートオブジェクトの値。

サポートされていないリクエストヘッダー

次のリクエスト ヘッダーはサポートされていません。

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

この操作により、マルチパートアップロードが完了します。バケットのバージョン管理が有効になっている場合、マルチパートアップロードの完了後にオブジェクトバージョンが作成されます。

バケットのバージョン管理が有効になっている場合、一意の `versionId` 保存されるオブジェクトのバージョンに応じて自動的に生成されます。これ `versionId` レスポンスでは、`x-amz-version-id` レスポンス ヘッダー。

バージョン管理が中断されている場合、オブジェクトのバージョンは null で保存されます。`versionId` 複数バージョンがすでに存在する場合は上書きされます。



バケットのバージョン管理が有効になっている場合、同じオブジェクト キーで同時にマルチパートアップロードが完了している場合でも、マルチパートアップロードを完了すると常に新しいバージョンが作成されます。バケットでバージョン管理が有効になっていない場合、マルチパートアップロードを開始し、次に同じオブジェクト キーで別のマルチパートアップロードを開始して完了させることが可能です。バージョン管理されていないバケットでは、最後に完了したマルチパートアップロードが優先されます。

レプリケーション、通知、またはメタデータ通知が失敗しました

マルチパートアップロードが行われるバケットがプラットフォーム サービス用に構成されている場合、関連付けられたレプリケーションまたは通知アクションが失敗しても、マルチパートアップロードは成功します。

テナントは、オブジェクトのメタデータまたはタグを更新することで、失敗したレプリケーションまたは通知をトリガーできます。テナントは、不要な変更を避けるために既存の値を再送信できます。

。"プラットフォームサービスのトラブルシューティング"。

### CreateMultipartUpload

CreateMultipartUpload (以前の名前は Initiate Multipart Upload) 操作は、オブジェクトのマルチパートアップロードを開始し、アップロード ID を返します。

その `x-amz-storage-class` リクエスト ヘッダーがサポートされています。提出された値は `x-amz-storage-class` StorageGRID が取り込み中にオブジェクト データを保護する方法に影響しますが、オブジェクトの永続コピーが StorageGRID システムにいくつ保存されるかには影響しません (これは ILM によって決定されます)。

取り込まれたオブジェクトに一致する ILM ルールが厳密な"取り込みオプション"、`x-amz-storage-class` ヘッダーは効果がありません。

以下の値は、`x-amz-storage-class` :

- STANDARD (デフォルト)
  - デュアル コミット: ILM ルールでデュアル コミット取り込みオプションが指定されている場合は、オブジェクトが取り込まれるとすぐにそのオブジェクトの 2 番目のコピーが作成され、別のストレージ ノードに配布されます (デュアル コミット)。ILM が評価されると、StorageGRID はこれらの初期の中間コピーがルール内の配置指示を満たしているかどうかを判断します。そうでない場合は、別の場所に新しいオブジェクトのコピーを作成し、初期の中間コピーを削除する必要があるかもしれません。
  - バランス: ILM ルールでバランス オプションが指定されていて、StorageGRID がルールで指定された

すべてのコピーをすぐに作成できない場合、StorageGRID は異なるストレージ ノードに 2 つの中間コピーを作成します。

StorageGRIDがILMルールで指定されたすべてのオブジェクトコピーを即時に作成できる場合（同期配置）、`x-amz-storage-class`ヘッダーは効果がありません。

- REDUCED\_REDUNDANCY

- デュアル コミット: ILM ルールでデュアル コミット オプションが指定されている場合、StorageGRID はオブジェクトが取り込まれると単一の中間コピーを作成します (シングル コミット)。
- バランス: ILM ルールでバランス オプションが指定されている場合、StorageGRID は、システムがルールで指定されたすべてのコピーをすぐに作成できない場合にのみ、単一の中間コピーを作成します。StorageGRID が同期配置を実行できる場合、このヘッダーは効果がありません。その `REDUCED\_REDUNDANCY` このオプションは、オブジェクトに一致する ILM ルールによって単一の複製コピーが作成される場合に最適です。この場合、`REDUCED\_REDUNDANCY` すべての取り込み操作で余分なオブジェクト コピーが不必要に作成および削除されるのを防ぎます。

使用して `REDUCED\_REDUNDANCY` 他の状況ではこのオプションは推奨されません。  
`REDUCED\_REDUNDANCY` 取り込み中にオブジェクト データが失われるリスクが高まります。たとえば、ILM 評価が行われる前に障害が発生したストレージ ノードに単一のコピーが最初に保存された場合、データが失われる可能性があります。



任意の期間に複製されたコピーが 1 つしかない場合、データが永久に失われるリスクがあります。オブジェクトの複製されたコピーが 1 つしか存在しない場合、ストレージ ノードに障害が発生したり重大なエラーが発生すると、そのオブジェクトは失われます。また、アップグレードなどのメンテナンス手順中は、オブジェクトへのアクセス権が一時的に失われます。

指定 `REDUCED\_REDUNDANCY` オブジェクトが最初に取り込まれたときに作成されるコピーの数にのみ影響します。これは、オブジェクトがアクティブな ILM ポリシーによって評価されるときに作成されるオブジェクトのコピー数には影響せず、StorageGRIDシステムに低いレベルの冗長性でデータが保存されることにはなりません。



S3オブジェクトロックが有効になっているバケットにオブジェクトを取り込む場合、`REDUCED\_REDUNDANCY` オプションは無視されます。レガシーコンプライアンスバケットにオブジェクトを取り込む場合、`REDUCED\_REDUNDANCY` オプションはエラーを返します。StorageGRID は、コンプライアンス要件が満たされていることを確認するために、常にデュアルコミット取り込みを実行します。

サポートされているリクエストヘッダー

次のリクエスト ヘッダーがサポートされています。

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

現在、SHA256値のみが `x-amz-checksum-algorithm` サポートされています。

- x-amz-meta-、その後ユーザー定義のメタデータを含む名前と値のペアが続く

ユーザー定義のメタデータの名前と値のペアを指定する場合は、次の一般的な形式を使用します。

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

ILMルールの参照時間として\*ユーザー定義作成時間\*オプションを使用する場合は、`creation-time`オブジェクトが作成された日時を記録するメタデータの名前として。例えば：

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

の価値 `creation-time` 1970 年 1 月 1 日からの秒数として評価されます。



追加 `creation-time` 従来のコンプライアンスが有効になっているバケットにオブジェクトを追加する場合、ユーザー定義のメタデータは許可されません。エラーが返されます。

- S3 オブジェクトロックリクエストヘッダー:

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

これらのヘッダーなしでリクエストが行われた場合、バケットのデフォルトの保持設定を使用して、オブジェクト バージョンの保持期限が計算されます。

### "S3 REST API を使用して S3 オブジェクトロックを設定する"

- SSE リクエスト ヘッダー:

- 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

#### [サーバー側暗号化のリクエストヘッダー]



StorageGRIDがUTF-8文字を処理する方法については、以下を参照してください。"PutObject"。

#### サーバー側暗号化のリクエストヘッダー

次のリクエスト ヘッダーを使用して、サーバー側暗号化でマルチパート オブジェクトを暗号化できます。SSE オプションと SSE-C オプションは相互に排他的です。

- **SSE:** StorageGRIDによって管理される一意のキーを使用してオブジェクトを暗号化する場合は、CreateMultipartUpload リクエストで次のヘッダーを使用します。いずれの UploadPart リクエストでもこのヘッダーを指定しないでください。

- x-amz-server-side-encryption

- **SSE-C**: 提供および管理する一意のキーを使用してオブジェクトを暗号化する場合は、CreateMultipartUpload リクエスト (および後続の各 UploadPart リクエスト) でこれら 3 つのヘッダーをすべて使用します。
  - x-amz-server-side-encryption-customer-algorithm: 特定 AES256。
  - x-amz-server-side-encryption-customer-key: 新しいオブジェクトの暗号化キーを指定します。
  - x-amz-server-side-encryption-customer-key-MD5: 新しいオブジェクトの暗号化キーの MD5 ダイジェストを指定します。



提供された暗号化キーは保存されません。暗号化キーを紛失すると、対応するオブジェクトも失われます。顧客提供のキーを使用してオブジェクトデータを保護する前に、以下の考慮事項を確認してください。"[サーバー側暗号化を使用する](#)"。

サポートされていないリクエストヘッダー

次のリクエスト ヘッダーはサポートされていません。

- x-amz-website-redirect-location

その x-amz-website-redirect-location `ヘッダーリターン` XNotImplemented。

バージョン管理

マルチパートアップロードは、アップロードの開始、アップロードの一覧表示、パーツのアップロード、アップロードされたパーツの組み立て、アップロードの完了という個別の操作で構成されます。

CompleteMultipartUpload 操作が実行されると、オブジェクトが作成され (該当する場合はバージョン管理も行われます)。

リストマルチパートアップロード

ListMultipartUploads オペレーションは、バケットの進行中のマルチパートアップロードを一覧表示します。

次のリクエスト パラメータがサポートされています。

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

## バージョン管理

マルチパートアップロードは、アップロードの開始、アップロードの一覧表示、パーツのアップロード、アップロードされたパーツの組み立て、アップロードの完了という個別の操作で構成されます。CompleteMultipartUpload 操作が実行されると、オブジェクトが作成され (該当する場合はバージョン管理も行われます)。

## UploadPart

UploadPart 操作は、オブジェクトのマルチパートアップロードの一部をアップロードします。

サポートされているリクエストヘッダー

次のリクエスト ヘッダーがサポートされています。

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

サーバー側暗号化のリクエストヘッダー

CreateMultipartUpload リクエストに SSE-C 暗号化を指定した場合は、各 UploadPart リクエストに次のリクエスト ヘッダーも含める必要があります。

- x-amz-server-side-encryption-customer-algorithm: 特定 AES256。
- x-amz-server-side-encryption-customer-key: CreateMultipartUpload リクエストで指定したのと同じ暗号化キーを指定します。
- x-amz-server-side-encryption-customer-key-MD5: CreateMultipartUpload リクエストで指定したのと同じ MD5 ダイジェストを指定します。



提供された暗号化キーは保存されません。暗号化キーを紛失すると、対応するオブジェクトも失われます。顧客提供のキーを使用してオブジェクトデータを保護する前に、以下の考慮事項を確認してください。"[サーバー側の暗号化を使用する](#)"。

CreateMultipartUpload リクエスト中に SHA-256 チェックサムを指定した場合は、各 UploadPart リクエストに次のリクエスト ヘッダーも含める必要があります。

- x-amz-checksum-sha256: この部分の SHA-256 チェックサムを指定します。

サポートされていないリクエストヘッダー

次のリクエスト ヘッダーはサポートされていません。

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

マルチパートアップロードは、アップロードの開始、アップロードの一覧表示、パーツのアップロード、アップロードされたパーツの組み立て、アップロードの完了という個別の操作で構成されます。CompleteMultipartUpload 操作が実行されると、オブジェクトが作成され (該当する場合はバージョン管理も行われます)。

## UploadPartCopy

UploadPartCopy 操作は、データ ソースとして既存のオブジェクトからデータをコピーして、オブジェクトの一部をアップロードします。

UploadPartCopy 操作は、すべての Amazon S3 REST API 動作で実装されます。予告なく変更されることがあります。

このリクエストは、指定されたオブジェクトデータの読み取りと書き込みを行います。x-amz-copy-source-range StorageGRIDシステム内。

次のリクエスト ヘッダーがサポートされています。

- 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

サーバー側暗号化のリクエストヘッダー

CreateMultipartUpload リクエストに SSE-C 暗号化を指定した場合は、各 UploadPartCopy リクエストに次のリクエスト ヘッダーも含める必要があります。

- x-amz-server-side-encryption-customer-algorithm: 特定 AES256。
- x-amz-server-side-encryption-customer-key: CreateMultipartUpload リクエストで指定したのと同じ暗号化キーを指定します。
- x-amz-server-side-encryption-customer-key-MD5: CreateMultipartUpload リクエストで指定したのと同じ MD5 ダイジェストを指定します。

ソース オブジェクトが顧客提供のキー (SSE-C) を使用して暗号化されている場合は、オブジェクトを復号化してコピーできるように、UploadPartCopy 要求に次の 3 つのヘッダーを含める必要があります。

- x-amz-copy-source-server-side-encryption-customer-algorithm: 特定 AES256。
- x-amz-copy-source-server-side-encryption-customer-key: ソース オブジェクトの作成時に指定した暗号化キーを指定します。
- x-amz-copy-source-server-side-encryption-customer-key-MD5: ソース オブジェクトの作成時に指定した MD5 ダイジェストを指定します。



提供された暗号化キーは保存されません。暗号化キーを紛失すると、対応するオブジェクトも失われます。顧客提供のキーを使用してオブジェクトデータを保護する前に、以下の考慮事項を確認してください。"[サーバー側の暗号化を使用する](#)"。

## バージョン管理

マルチパートアップロードは、アップロードの開始、アップロードの一覧表示、パーツのアップロード、アップロードされたパーツの組み立て、アップロードの完了という個別の操作で構成されます。

CompleteMultipartUpload 操作が実行されると、オブジェクトが作成され (該当する場合はバージョン管理も行われます)。

## エラー応答

StorageGRIDシステムは、適用されるすべての標準 S3 REST API エラー応答をサポートします。さらに、StorageGRID実装ではいくつかのカスタム応答が追加されます。

### サポートされているS3 APIエラーコード

Name	HTTPのステータス
アクセスが拒否されました	403 Forbidden
バッドダイジェスト	400 Bad Request
バケットが既に存在する	409 紛争
バケットが空でない	409 紛争
不完全なボディ	400 Bad Request
内部エラー	500 内部サーバーエラー
無効なアクセスキーID	403 Forbidden
無効な引数	400 Bad Request
無効なバケット名	400 Bad Request
無効なバケット状態	409 紛争
無効なダイジェスト	400 Bad Request
無効な暗号化アルゴリズムエラー	400 Bad Request
無効な部分	400 Bad Request
無効な部品順序	400 Bad Request
無効な範囲	416 要求された範囲は満たされません

Name	HTTPのステータス
無効なリクエスト	400 Bad Request
無効なストレージクラス	400 Bad Request
無効なタグ	400 Bad Request
無効なURI	400 Bad Request
キーが長すぎる	400 Bad Request
不正なXML	400 Bad Request
メタデータが大きすぎる	400 Bad Request
メソッドが許可されていません	405 メソッドは許可されていません
コンテンツの長さが不足しています	411 長さが必要です
リクエストボディエラーが見つかりません	400 Bad Request
セキュリティヘッダーがありません	400 Bad Request
NoSuchBucket	404 Not Found
NoSuchKey	404 Not Found
アップロード不可	404 Not Found
実装されていません	501 実装されていません
NoSuchBucketPolicy	404 Not Found
オブジェクトロック構成が見つかりませんエラー	404 Not Found
前提条件失敗	412 前提条件が失敗しました
リクエスト時間が偏りすぎている	403 Forbidden
サービスは利用できません	503 サービスは利用できません
署名が一致しない	403 Forbidden

Name	HTTPのステータス
バケットが多すぎる	400 Bad Request
ユーザーキーを指定する必要があります	400 Bad Request

### StorageGRIDカスタムエラーコード

Name	説明	HTTPのステータス
XBucketライフサイクルが許可されていません	レガシーコンプライアンスバケットではバケットライフサイクル設定は許可されません	400 Bad Request
XBucketPolicyParseException	受信したバケット ポリシー JSON を解析できませんでした。	400 Bad Request
Xコンプライアンス競合	従来のコンプライアンス設定のため、操作は拒否されました。	403 Forbidden
XComplianceReducedRedundancyForbidden	従来の準拠バケットでは冗長性の低下は許可されません	400 Bad Request
XMaxBucketPolicyLength を超えました	ポリシーがバケット ポリシーの最大許容長を超えています。	400 Bad Request
XMissingInternalRequestHeader	内部リクエストのヘッダーがありません。	400 Bad Request
XNoSuchBucketCompliance	指定されたバケットでは、従来のコンプライアンスが有効になっていません。	404 Not Found
X受け入れられない	リクエストには、満たすことができなかった1つ以上の Accept ヘッダーが含まれています。	406 受け入れられません
X未実装	提供されたリクエストには、実装されていない機能が含まれています。	501 実装されていません

## StorageGRIDカスタム操作

### StorageGRIDカスタム操作

StorageGRIDシステムは、S3 REST API に追加されたカスタム操作をサポートします。

次の表は、StorageGRIDでサポートされているカスタム操作を示しています。

処理	説明
"GETバケットの一貫性"	特定のバケットに適用されている一貫性を返します。
"PUT バケットの一貫性"	特定のバケットに適用される一貫性を設定します。
"GET バケットの最終アクセス時刻"	特定のバケットの最終アクセス時間の更新が有効になっているか無効になっているかを返します。
"バケットの最終アクセス時刻をPUTする"	特定のバケットの最終アクセス時間の更新を有効または無効にすることができます。
"バケットメタデータ通知設定の削除"	特定のバケットに関連付けられたメタデータ通知構成 XML を削除します。
"バケットメタデータ通知設定の取得"	特定のバケットに関連付けられたメタデータ通知構成 XML を返します。
"PUT バケットメタデータ通知設定"	バケットのメタデータ通知サービスを設定します。
"ストレージ使用量の取得"	アカウントと、アカウントに関連付けられている各バケットで使用されているストレージの合計量を示します。
"非推奨: コンプライアンス設定による CreateBucket"	非推奨およびサポートされていません: コンプライアンスを有効にして新しいバケットを作成することはできなくなりました。
"非推奨: GET バケットコンプライアンス"	非推奨ですがサポートされています: 既存のレガシー コンプライアンスバケットに対して現在有効なコンプライアンス設定を返します。
"非推奨: PUT バケットのコンプライアンス"	非推奨ですがサポートされています: 既存のレガシー コンプライアンスバケットのコンプライアンス設定を変更できます。

## GETバケットの一貫性

GET Bucket 整合性リクエストを使用すると、特定のバケットに適用されている整合性を判別できます。

デフォルトの一貫性は、新しく作成されたオブジェクトの書き込み後の読み取りを保証するように設定されています。

この操作を完了するには、s3:GetBucketConsistency 権限を持っているか、アカウント ルートである必要があります。

### リクエスト例

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

## 応答

レスポンスXMLでは、`<Consistency>`次のいずれかの値を返します。

一貫性	説明
全て	すべてのノードがデータを直ちに受信します。そうでない場合、要求は失敗します。
strong-global	すべてのサイトにわたるすべてのクライアント要求に対して、書き込み後の読み取りの一貫性を保証します。
strong-site	サイト内のすべてのクライアント要求に対して、書き込み後の読み取り一貫性を保証します。
新規書き込み後の読み取り	(デフォルト) 新しいオブジェクトに対しては書き込み後の読み取り一貫性を提供し、オブジェクトの更新に対しては最終的な一貫性を提供します。高可用性とデータ保護の保証を提供します。ほとんどの場合に推奨されます。
使用可能	新しいオブジェクトとオブジェクトの更新の両方に対して最終的な一貫性を提供します。S3 バケットの場合は、必要な場合にのみ使用してください (たとえば、めったに読み取られないログ値を含むバケットの場合や、存在しないキーに対する HEAD または GET 操作の場合など)。S3 FabricPoolバケットではサポートされていません。

## 応答例

```
HTTP/1.1 200 OK
Date: Fri, 18 Sep 2020 01:02:18 GMT
Connection: CLOSE
Server: StorageGRID/11.5.0
x-amz-request-id: 12345
Content-Length: 127
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<Consistency xmlns="http://s3.storagegrid.com/doc/2015-02-01/">read-after-
new-write</Consistency>
```

## PUT バケットの一貫性

PUT Bucket 整合性リクエストを使用すると、バケットに対して実行される操作に適用する整合性を指定できます。

デフォルトの一貫性は、新しく作成されたオブジェクトの書き込み後の読み取りを保証するように設定されています。

### 開始する前に

この操作を完了するには、s3:PutBucketConsistency 権限を持っているか、アカウント ルートである必要があります。

### 要求

その `x-ntap-sg-consistency` パラメータには次のいずれかの値が含まれている必要があります。

一貫性	説明
全て	すべてのノードがデータを直ちに受信します。そうでない場合、要求は失敗します。
strong-global	すべてのサイトにわたるすべてのクライアント要求に対して、書き込み後の読み取りの一貫性を保証します。
strong-site	サイト内のすべてのクライアント要求に対して、書き込み後の読み取り一貫性を保証します。
新規書き込み後の読み取り	(デフォルト) 新しいオブジェクトに対しては書き込み後の読み取り一貫性を提供し、オブジェクトの更新に対しては最終的な一貫性を提供します。高可用性とデータ保護の保証を提供します。ほとんどの場合に推奨されます。
使用可能	新しいオブジェクトとオブジェクトの更新の両方に対して最終的な一貫性を提供します。S3 バケットの場合は、必要な場合にのみ使用してください (たとえば、めったに読み取られないログ値を含むバケットの場合や、存在しないキーに対する HEAD または GET 操作の場合など)。S3 FabricPoolバケットではサポートされていません。

注: 一般的には、「新規書き込み後の読み取り」の一貫性を使用する必要があります。リクエストが正しく機能しない場合は、可能であればアプリケーション クライアントの動作を変更します。または、各 API リクエストの一貫性を指定するようにクライアントを構成します。最後の手段としてのみ、バケット レベルで一貫性を設定します。

### リクエスト例

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

## 関連情報

["一貫性の値"](#)

## GET バケットの最終アクセス時刻

GET Bucket 最終アクセス時刻リクエストを使用すると、個々のバケットの最終アクセス時刻の更新が有効になっているか無効になっているかを判断できます。

この操作を完了するには、s3:GetBucketLastAccessTime 権限を持っているか、アカウントルートである必要があります。

## リクエスト例

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

## 応答例

この例では、バケットの最終アクセス時刻の更新が有効になっていることを示しています。

```
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>
```

## バケットの最終アクセス時刻をPUTする

PUT Bucket 最終アクセス時刻リクエストを使用すると、個々のバケットの最終アクセス時刻の更新を有効または無効にすることができます。最終アクセス時刻の更新を無効に

するとパフォーマンスが向上し、これはバージョン 10.3.0 以降で作成されたすべてのバケットのデフォルト設定になります。

この操作を完了するには、バケットに対する `s3:PutBucketLastAccessTime` 権限を持っているか、アカウントルートである必要があります。



StorageGRIDバージョン 10.3 以降では、すべての新しいバケットの最終アクセス時間の更新はデフォルトで無効になっています。以前のバージョンのStorageGRIDを使用して作成されたバケットがあり、新しいデフォルトの動作に一致させたい場合は、以前のバケットごとに最終アクセス時刻の更新を明示的に無効にする必要があります。PUT Bucket last access time リクエストを使用するか、テナント マネージャーのバケットの詳細ページから、最終アクセス時間の更新を有効または無効にすることができます。見る"[最終アクセス時間の更新を有効または無効にする](#)"。

バケットの最終アクセス時間の更新が無効になっている場合、バケットに対する操作には次の動作が適用されます。

- `GetObject`、`GetObjectAcl`、`GetObjectTagging`、および `HeadObject` リクエストでは、最終アクセス時間が更新されません。オブジェクトは、情報ライフサイクル管理 (ILM) 評価のキューに追加されません。
- メタデータのみを更新する `CopyObject` および `PutObjectTagging` リクエストでは、最終アクセス時刻も更新されます。オブジェクトは ILM 評価のキューに追加されます。
- ソース バケットの最終アクセス時間の更新が無効になっている場合、`CopyObject` リクエストではソースバケットの最終アクセス時間が更新されません。コピーされたオブジェクトは、ソースバケットの ILM 評価のキューに追加されません。ただし、宛先の場合、`CopyObject` 要求によって最終アクセス時刻が常に更新されます。オブジェクトのコピーは、ILM 評価のキューに追加されます。
- `CompleteMultipartUpload` は最終アクセス時間の更新を要求します。完了したオブジェクトは、ILM 評価のキューに追加されます。

#### リクエスト例

この例では、バケットの最終アクセス時間を有効にします。

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

この例では、バケットの最終アクセス時間を無効にします。

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

## バケットメタデータ通知設定の削除

DELETE Bucket メタデータ通知構成リクエストを使用すると、構成 XML を削除して、個々のバケットの検索統合サービスを無効にすることができます。

この操作を完了するには、バケットに対する s3:DeleteBucketMetadataNotification 権限を持っているか、アカウント ルートである必要があります。

### リクエスト例

この例では、バケットの検索統合サービスを無効にする方法を示します。

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

## バケットメタデータ通知設定の取得

GET バケット メタデータ通知構成リクエストを使用すると、個々のバケットの検索統合を構成するために使用される構成 XML を取得できます。

この操作を完了するには、s3:GetBucketMetadataNotification 権限を持っているか、アカウント ルートである必要があります。

### リクエスト例

このリクエストは、バケットのメタデータ通知設定を取得します。 bucket 。

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

### 応答

レスポンス本文には、バケットのメタデータ通知構成が含まれます。メタデータ通知構成では、検索統合用にバケットを構成する方法を決定できます。つまり、どのオブジェクトがインデックスされているか、またそのオブジェクト メタデータがどのエンドポイントに送信されているかを判別できます。

```

<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>

```

各メタデータ通知構成には、1つ以上のルールが含まれます。各ルールは、適用されるオブジェクトと、StorageGRIDがオブジェクトメタデータを送信する宛先を指定します。宛先は、StorageGRIDエンドポイントのURNを使用して指定する必要があります。

Name	説明	必須
メタデータ通知構成	メタデータ通知のオブジェクトと宛先を指定するために使用されるルールのコンテナ タグ。  1つ以上の Rule 要素が含まれます。	はい
Rule	指定されたインデックスにメタデータを追加するオブジェクトを識別するルールのコンテナ タグ。  プレフィックスが重複するルールは拒否されます。  MetadataNotificationConfiguration 要素に含まれます。	はい
ID	ルールの一意的識別子。  ルール要素に含まれます。	いいえ
ステータス	ステータスは「有効」または「無効」になります。無効にされているルールに対してはアクションは実行されません。  ルール要素に含まれます。	はい

Name	説明	必須
接頭辞	<p>プレフィックスに一致するオブジェクトはルールの影響を受け、そのメタデータは指定された宛先に送信されます。</p> <p>すべてのオブジェクトを一致させるには、空のプレフィックスを指定します。</p> <p>ルール要素に含まれます。</p>	はい
デスティネーション	<p>ルールの宛先のコンテナ タグ。</p> <p>ルール要素に含まれます。</p>	はい
壺	<p>オブジェクト メタデータが送信される宛先の URN。次のプロパティを持つStorageGRIDエンドポイントの URN である必要があります。</p> <ul style="list-style-type: none"> <li>• `es` 3 番目の要素である必要があります。</li> <li>• URNは、メタデータが格納されているインデックスとタイプで終わる必要があります。形式は次のようになります。 domain-name/myindex/mytype 。</li> </ul> <p>エンドポイントは、テナント マネージャーまたはテナント管理 API を使用して構成されます。それらは次の形式をとります:</p> <ul style="list-style-type: none"> <li>• arn:aws:es:_region:account-ID_:domain/mydomain/myindex/mytype</li> <li>• urn:mysite:es:::mydomain/myindex/mytype</li> </ul> <p>構成 XML を送信する前にエンドポイントを構成する必要があります。そうしないと、404 エラーが発生して構成が失敗します。</p> <p>Urn は Destination 要素に含まれます。</p>	はい

## 応答例

XMLは、`<MetadataNotificationConfiguration></MetadataNotificationConfiguration>` tags は、バケットの検索統合エンドポイントとの統合がどのように構成されているかを示します。この例では、オブジェクトメタデータはElasticsearchインデックスに送信されています。`current`と名付けられたタイプ `2017`AWSドメインでホストされている `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-
1:33333333:domain/records/current/2017</Urn>
    </Destination>
  </Rule>
</MetadataNotificationConfiguration>
```

## 関連情報

["テナントアカウントを使用する"](#)

## PUT バケットメタデータ通知設定

PUT バケットメタデータ通知構成リクエストを使用すると、個々のバケットの検索統合サービスを有効にすることができます。リクエスト本文に指定するメタデータ通知構成XMLは、メタデータが宛先検索インデックスに送信されるオブジェクトを指定します。

この操作を完了するには、バケットに対する `s3:PutBucketMetadataNotification` 権限を持っているか、アカウントルートである必要があります。

## 要求

リクエストには、リクエスト本文にメタデータ通知構成が含まれている必要があります。各メタデータ通知構成には、1つ以上のルールが含まれます。各ルールは、適用されるオブジェクトと、StorageGRIDがオブジェクトメタデータを送信する宛先を指定します。

オブジェクトは、オブジェクト名のプレフィックスでフィルタリングできます。たとえば、プレフィックスを持つオブジェクトのメタデータを送信できます。 `/images` 1つの宛先に、接頭辞を持つオブジェクト `/videos` 別の宛先に。

プレフィックスが重複する構成は無効であり、送信時に拒否されます。たとえば、接頭辞を持つオブジェクトに対する1つのルールを含む構成では、``test`` 接頭辞を持つオブジェクトに対する2番目のルール ``test2`` 許可されません。

宛先は、StorageGRIDエンドポイントのURNを使用して指定する必要があります。メタデータ通知設定が送

信されるときにエンドポイントが存在している必要があります。存在しない場合、リクエストは失敗します。400 Bad Request エラーメッセージは次のようになります。`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>
```

この表では、メタデータ通知構成 XML の要素について説明します。

Name	説明	必須
メタデータ通知構成	メタデータ通知のオブジェクトと宛先を指定するために使用されるルールのコンテナ タグ。  1 つ以上の Rule 要素が含まれます。	はい
Rule	指定されたインデックスにメタデータを追加するオブジェクトを識別するルールのコンテナ タグ。  プレフィックスが重複するルールは拒否されます。  MetadataNotificationConfiguration 要素に含まれません。	はい
ID	ルールの一意的識別子。  ルール要素に含まれます。	いいえ
ステータス	ステータスは「有効」または「無効」になります。無効にされているルールに対してはアクションは実行されません。  ルール要素に含まれます。	はい

Name	説明	必須
接頭辞	<p>プレフィックスに一致するオブジェクトはルールの影響を受け、そのメタデータは指定された宛先に送信されます。</p> <p>すべてのオブジェクトを一致させるには、空のプレフィックスを指定します。</p> <p>ルール要素に含まれます。</p>	はい
デスティネーション	<p>ルールの宛先のコンテナ タグ。</p> <p>ルール要素に含まれます。</p>	はい
壺	<p>オブジェクト メタデータが送信される宛先の URN。次のプロパティを持つStorageGRIDエンドポイントの URN である必要があります。</p> <ul style="list-style-type: none"> <li>• `es` 3 番目の要素である必要があります。</li> <li>• URNは、メタデータが格納されているインデックスとタイプで終わる必要があります。形式は次のようになります。 domain-name/myindex/mytype 。</li> </ul> <p>エンドポイントは、テナント マネージャーまたはテナント管理 API を使用して構成されます。それらは次の形式をとります:</p> <ul style="list-style-type: none"> <li>• arn:aws:es:region:account-ID:domain/mydomain/myindex/mytype</li> <li>• urn:mysite:es:::mydomain/myindex/mytype</li> </ul> <p>構成 XML を送信する前にエンドポイントを構成する必要があります。そうしないと、404 エラーが発生して構成が失敗します。</p> <p>Urn は Destination 要素に含まれます。</p>	はい

### リクエスト例

この例では、バケットの検索統合を有効にする方法を示します。この例では、すべてのオブジェクトのオブジェクト メタデータが同じ宛先に送信されます。

```
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>
```

この例では、プレフィックスに一致するオブジェクトのオブジェクトメタデータ `images` 一つの宛先に送信される一方、プレフィックスに一致するオブジェクトのオブジェクトメタデータは `videos` 2 番目の宛先に送信されます。

```
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-1:33333333: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>
```

### 検索統合サービスによって生成されたJSON

バケットの検索統合サービスを有効にすると、オブジェクトのメタデータまたはタグが追加、更新、または削除されるたびに、JSON ドキュメントが生成され、宛先エンドポイントに送信されます。

この例では、キーを持つオブジェクトが生成された場合に生成されるJSONの例を示します。SGWS/Tagging.txt`バケットに作成されます `test。その `test`バケットはバージョン管理されていないため、`versionId`タグが空です。

```

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

```

### メタデータ通知に含まれるオブジェクトメタデータ

この表には、検索統合が有効になっているときに宛先エンドポイントに送信される JSON ドキュメントに含まれるすべてのフィールドがリストされています。

ドキュメント名には、バケット名、オブジェクト名、バージョン ID (存在する場合) が含まれます。

タイプ	アイテム名	説明
バケットとオブジェクトの情報	バケット	バケットの名前
バケットとオブジェクトの情報	キー	オブジェクトキー名
バケットとオブジェクトの情報	バージョンID	オブジェクトバージョン (バージョン管理されたバケット内のオブジェクトの場合)
バケットとオブジェクトの情報	地域	バケット領域の例 us-east-1
システムメタデータ	サイズ	HTTPクライアントに表示されるオブジェクトサイズ (バイト単位)
システムメタデータ	md5	オブジェクトハッシュ
ユーザーメタデータ	メタデータ <i>key:value</i>	オブジェクトのすべてのユーザーメタデータ (キーと値のペア)

タイプ	アイテム名	説明
タグ	タグ <i>key:value</i>	オブジェクトに定義されたすべてのオブジェクトタグ（キーと値のペア）



タグとユーザー メタデータの場合、StorageGRID は日付と数値を文字列または S3 イベント通知として Elasticsearch に渡します。これらの文字列を日付または数値として解釈するように Elasticsearch を構成するには、動的フィールド マッピングと日付形式のマッピングに関する Elasticsearch の指示に従います。検索統合サービスを構成する前に、インデックスで動的フィールド マッピングを有効にする必要があります。ドキュメントのインデックスが作成された後は、インデックス内のドキュメントのフィールド タイプを編集することはできません。

#### 関連情報

["テナントアカウントを使用する"](#)

## GET ストレージ使用量リクエスト

GET ストレージ使用量リクエストは、アカウントで使用されているストレージの合計量と、アカウントに関連付けられている各バケットのストレージ量を通知します。

アカウントとそのバケットで使用されているストレージの量は、ListBuckets リクエストを修正することで取得できます。`x-ntap-sg-usage` クエリパラメータ。バケット ストレージの使用量は、システムによって処理される PUT および DELETE 要求とは別に追跡されます。特にシステムの負荷が高い場合、リクエストの処理に基づいて使用量の値が予想値と一致するまでに多少の遅延が発生する可能性があります。

デフォルトでは、StorageGRID は強力なグローバル一貫性を使用して使用状況情報を取得しようとします。強力なグローバル一貫性を実現できない場合、StorageGRID は強力なサイト一貫性で使用状況情報を取得しようとします。

この操作を完了するには、s3:ListAllMyBuckets 権限を持っているか、アカウント ルートである必要があります。

#### リクエスト例

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

#### 応答例

この例では、2つのバケットに4つのオブジェクトと12バイトのデータがあるアカウントを示しています。各バケットには2つのオブジェクトと6バイトのデータが含まれます。

```
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>
```

## バージョン管理

保存されたすべてのオブジェクトバージョンは、`ObjectCount` そして `DataBytes` 応答内の値。削除マークは追加されません `ObjectCount` 合計。

## 関連情報

["一貫性の値"](#)

## 従来のコンプライアンスに対するバケット リクエストのサポート終了

従来のコンプライアンスに対するバケット リクエストのサポート終了

従来のコンプライアンス機能を使用して作成されたバケットを管理するには、StorageGRID S3 REST API を使用する必要がある場合があります。

コンプライアンス機能は廃止されました

以前のStorageGRIDバージョンで利用可能だったStorageGRIDコンプライアンス機能は廃止され、S3 オブジェクト ロックに置き換えられました。

以前にグローバルコンプライアンス設定を有効にしていた場合、StorageGRID 11.6 ではグローバル S3 オブジェクトロック設定が有効になっています。コンプライアンスを有効にした新しいバケットを作成することはできなくなりました。ただし、必要に応じて、StorageGRID S3 REST API を使用して、既存のレガシーコンプライアンスバケットを管理することができます。

- ["S3 REST API を使用して S3 オブジェクトロックを設定する"](#)
- ["ILMでオブジェクトを管理する"](#)
- ["NetAppナレッジベース: StorageGRID 11.5 でレガシー準拠バケットを管理する方法"](#)

非推奨のコンプライアンス リクエスト:

- ["非推奨 - コンプライアンスのためのPUT Bucketリクエストの変更"](#)

SGCompliance XML 要素は非推奨です。以前は、PUT Bucket リクエストのオプションの XML リクエスト本文にこのStorageGRIDカスタム要素を含めることで、準拠バケットを作成できました。

- ["非推奨 - GET バケットコンプライアンス"](#)

GET Bucket コンプライアンス リクエストは非推奨です。ただし、このリクエストを引き続き使用して、既存のレガシーコンプライアンスバケットに現在有効なコンプライアンス設定を確認することはできません。

- ["非推奨 - PUT バケットのコンプライアンス"](#)

PUT Bucket コンプライアンス リクエストは非推奨です。ただし、このリクエストを引き続き使用して、既存のレガシーコンプライアンスバケットのコンプライアンス設定を変更することはできます。たとえば、既存のバケットを法的保留状態にしたり、保持期間を延長したりできます。

非推奨: コンプライアンスのための **CreateBucket** リクエストの変更

SGCompliance XML 要素は非推奨です。以前は、このStorageGRIDカスタム要素をCreateBucket リクエストのオプションの XML リクエスト本文に含めることで、準拠バケットを作成できました。



以前のStorageGRIDバージョンで利用可能だったStorageGRIDコンプライアンス機能は廃止され、S3 オブジェクトロックに置き換えられました。詳細については、以下を参照してください。

- ["S3 REST API を使用して S3 オブジェクトロックを設定する"](#)
- ["NetAppナレッジベース: StorageGRID 11.5 でレガシー準拠バケットを管理する方法"](#)

コンプライアンスを有効にした新しいバケットを作成することはできなくなりました。コンプライアンスのためのCreateBucket リクエストの変更を使用して新しいコンプライアンスバケットを作成しようとすると、次のエラーメッセージが返されます。

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

## 非推奨: GET バケットコンプライアンスリクエスト

GET Bucket コンプライアンス リクエストは非推奨です。ただし、このリクエストを引き続き使用して、既存のレガシー コンプライアンス バケットに現在有効なコンプライアンス設定を確認することはできます。



以前のStorageGRIDバージョンで利用可能だったStorageGRIDコンプライアンス機能は廃止され、S3 オブジェクト ロックに置き換えられました。詳細については、以下を参照してください。

- ["S3 REST API を使用して S3 オブジェクトロックを設定する"](#)
- ["NetAppナレッジベース: StorageGRID 11.5 でレガシー準拠バケットを管理する方法"](#)

この操作を完了するには、s3:GetBucketCompliance 権限を持っているか、アカウント ルートである必要があります。

### リクエスト例

この例のリクエストでは、バケットのコンプライアンス設定を確認できます。 mybucket 。

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

### 応答例

レスポンスXMLでは、`<SGCompliance>`バケットに有効なコンプライアンス設定を一覧表示します。このサンプル応答は、オブジェクトがグリッドに取り込まれた時点から 1 年間 (525,600 分) 各オブジェクトが保持されるバケットのコンプライアンス設定を示しています。現在、このバケットには法的保留はありません。各オブジェクトは 1 年後に自動的に削除されます。

```
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	説明
保持期間 (分)	このバケットに追加されたオブジェクトの保持期間の長さ (分)。保持期間は、オブジェクトがグリッドに取り込まれた時点で開始されます。
法的保留	<ul style="list-style-type: none"> <li>• True: このバケットは現在、法的保留中です。このバケット内のオブジェクトは、保持期間が終了していても、法的保留が解除されるまで削除できません。</li> <li>• False: このバケットは現在、法的保留の対象ではありません。このバケット内のオブジェクトは、保持期間が終了すると削除できます。</li> </ul>
自動削除	<ul style="list-style-type: none"> <li>• True: バケットが法的保留中でない限り、このバケット内のオブジェクトは、保持期間が終了すると自動的に削除されます。</li> <li>• False: 保持期間が終了しても、このバケット内のオブジェクトは自動的に削除されません。これらのオブジェクトを削除する必要がある場合は、手動で削除する必要があります。</li> </ul>

#### エラー応答

バケットが準拠して作成されていない場合、レスポンスのHTTPステータスコードは次のようになります。  
404 Not Found S3エラーコードは XNoSuchBucketCompliance。

#### 非推奨: PUT バケットコンプライアンスリクエスト

PUT Bucket コンプライアンス リクエストは非推奨です。ただし、このリクエストを引き続き使用して、既存のレガシー コンプライアンス バケットのコンプライアンス設定を変更することはできます。たとえば、既存のバケットを法的保留状態にしたり、保持期間を延長したりできます。



以前のStorageGRIDバージョンで利用可能だったStorageGRIDコンプライアンス機能は廃止され、S3 オブジェクト ロックに置き換えられました。詳細については、以下を参照してください。

- ["S3 REST API を使用して S3 オブジェクトロックを設定する"](#)
- ["NetAppナレッジベース: StorageGRID 11.5 でレガシー準拠バケットを管理する方法"](#)

この操作を完了するには、s3:PutBucketCompliance 権限を持っているか、アカウント ルートである必要があります。

PUT Bucket コンプライアンス リクエストを発行するときは、コンプライアンス設定のすべてのフィールドに値を指定する必要があります。

#### リクエスト例

この例のリクエストは、バケットのコンプライアンス設定を変更します。mybucket。この例では、`mybucket` オブジェクトがグリッドに取り込まれた時点から、1年間ではなく2年間(1,051,200分)保持されるようになりました。このバケットには法的拘束力はありません。各オブジェクトは2年後に自動的に削除

されます。

```
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	説明
保持期間 (分)	<p>このバケットに追加されたオブジェクトの保持期間の長さ (分)。保持期間は、オブジェクトがグリッドに取り込まれた時点で開始されます。</p> <p>重要 RetentionPeriodMinutes に新しい値を指定する場合は、バケットの現在の保持期間以上の値を指定する必要があります。バケットの保持期間が設定されると、その値を減らすことはできず、増やすことしかできません。</p>
法的保留	<ul style="list-style-type: none"><li>• True: このバケットは現在、法的保留中です。このバケット内のオブジェクトは、保持期間が終了していても、法的保留が解除されるまで削除できません。</li><li>• False: このバケットは現在、法的保留の対象ではありません。このバケット内のオブジェクトは、保持期間が終了すると削除できます。</li></ul>
自動削除	<ul style="list-style-type: none"><li>• True: バケットが法的保留中でない限り、このバケット内のオブジェクトは、保持期間が終了すると自動的に削除されます。</li><li>• False: 保持期間が終了しても、このバケット内のオブジェクトは自動的に削除されません。これらのオブジェクトを削除する必要がある場合は、手動で削除する必要があります。</li></ul>

#### コンプライアンス設定の一貫性

PUT Bucket コンプライアンス リクエストを使用して S3 バケットのコンプライアンス設定を更新すると、StorageGRID はグリッド全体でバケットのメタデータを更新しようとします。デフォルトでは、StorageGRIDは強力なグローバル 整合性を使用して、バケット メタデータを含むすべてのデータ センター サイトとすべてのストレージ ノードで、変更されたコンプライアンス設定に対する書き込み後の読み取り整合性が確保されることを保証します。

データセンターサイトまたはサイト内の複数のストレージノードが利用できないためにStorageGRIDが\*強力なグローバル\*一貫性を実現できない場合、応答のHTTPステータスコードは次のようになります。 503

Service Unavailable.

この応答を受け取った場合は、グリッド管理者に連絡して、必要なストレージ サービスができるだけ早く利用できるようにする必要があります。グリッド管理者が各サイトで十分なストレージ ノードを用意できない場合、テクニカル サポートは、\*強力なサイト\*の一貫性を強制して失敗した要求を再試行するように指示することがあります。



テクニカル サポートから指示された場合、またはこのレベルを使用した場合の潜在的な結果を理解していない限り、PUT バケット コンプライアンスに対して **Strong-site** の一貫性を強制しないでください。

一貫性が **Strong-site** に低下すると、StorageGRID は、更新されたコンプライアンス設定がサイト内のクライアント要求に対してのみ書き込み後の読み取り一貫性を持つことを保証します。つまり、すべてのサイトとストレージ ノードが利用可能になるまで、StorageGRIDシステムではこのバケットに対して一時的に複数の矛盾した設定が存在する可能性があります。設定に一貫性がないと、予期しない望ましくない動作が発生する可能性があります。たとえば、バケットをリーガルホールドの対象とし、一貫性を低く強制した場合、一部のデータセンター サイトではバケットの以前のコンプライアンス設定（つまり、リーガルホールドの解除）が引き続き有効になる可能性があります。その結果、法的保留中であると思われるオブジェクトは、保持期間の期限が切れると、ユーザーまたは AutoDelete（有効になっている場合）によって削除される可能性があります。

\*Strong-site\*一貫性の使用を強制するには、PUT Bucketコンプライアンスリクエストを再発行し、Consistency-Control HTTP リクエスト ヘッダーは次のとおりです。

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

#### エラー応答

- バケットが準備して作成されていない場合、レスポンスのHTTPステータスコードは次のようになります。404 Not Found。
- もし RetentionPeriodMinutes`リクエスト内の値がバケットの現在の保存期間より短い場合、HTTPステータスコードは `400 Bad Request。

#### 関連情報

["非推奨: コンプライアンスのための PUT Bucket リクエストの変更"](#)

## バケットとグループのアクセスポリシー

### バケットとグループのアクセスポリシーを使用する

StorageGRID は、Amazon Web Services (AWS) ポリシー言語を使用して、S3 テナントがバケットおよびバケット内のオブジェクトへのアクセスを制御できるようにします。StorageGRIDシステムは、S3 REST API ポリシー言語のサブセットを実装します。S3 API のアクセス ポリシーは JSON で記述されます。

## アクセスポリシーの概要

StorageGRIDでサポートされているアクセス ポリシーには 2 種類あります。

- バケット ポリシー。GetBucketPolicy、PutBucketPolicy、DeleteBucketPolicy S3 API 操作、または Tenant Manager または Tenant Management API を使用して管理されます。バケット ポリシーはバケットに添付され、バケット所有者アカウントまたはその他のアカウントのユーザーによるバケットおよびバケット内のオブジェクトへのアクセスを制御するように構成されます。バケット ポリシーは 1 つのバケットにのみ適用され、複数のグループに適用される場合があります。
- グループ ポリシー。これは、Tenant Manager または Tenant Management API を使用して設定されます。グループ ポリシーはアカウント内のグループに添付され、そのグループがそのアカウントが所有する特定のリソースにアクセスできるように構成されます。グループ ポリシーは 1 つのグループにのみ適用され、複数のバケットに適用される場合もあります。



グループ ポリシーとバケット ポリシーの優先順位に違いはありません。

StorageGRIDバケットおよびグループ ポリシーは、Amazon によって定義された特定の文法に従います。各ポリシー内にはポリシー ステートメントの配列があり、各ステートメントには次の要素が含まれています。

- ステートメントID (Sid) (オプション)
- 効果
- プリンシパル/非プリンシパル
- リソース/非リソース
- アクション/非アクション
- 条件 (オプション)

ポリシー ステートメントは、次の構造を使用して作成され、アクセス許可を指定します: <Condition> が適用される場合に、<Principal> が <Resource> に対して <Action> を実行することを許可/拒否するには、<Effect> を付与します。

各ポリシー要素は特定の機能に使用されます。

要素	説明
シド	Sid 要素はオプションです。Sid はユーザーへの説明としてのみ使用されます。これはStorageGRIDシステムによって保存されますが、解釈されません。
効果	Effect 要素を使用して、指定された操作が許可されるか拒否されるかを決定します。サポートされているアクション要素のキーワードを使用して、バケットまたはオブジェクトに対して許可 (または拒否) する操作を識別する必要があります。

要素	説明
プリンシパル/非プリンシパル	<p>ユーザー、グループ、アカウントが特定のリソースにアクセスし、特定のアクションを実行できるようにすることができます。リクエストに S3 署名が含まれていない場合は、プリンシパルとしてワイルドカード文字 (*) を指定することにより匿名アクセスが許可されます。デフォルトでは、アカウント ルートのみがアカウントが所有するリソースにアクセスできます。</p> <p>バケット ポリシーでは、Principal 要素のみを指定する必要があります。グループ ポリシーの場合、ポリシーが関連付けられているグループは暗黙的な Principal 要素です。</p>
リソース/非リソース	リソース要素はバケットとオブジェクトを識別します。リソースを識別する Amazon リソースネーム (ARN) を使用して、バケットとオブジェクトへのアクセス許可を許可または拒否できます。
アクション/非アクション	アクション要素と効果要素は、権限の 2 つのコンポーネントです。グループがリソースを要求すると、リソースへのアクセスが許可または拒否されます。権限を明示的に割り当てない限りアクセスは拒否されますが、明示的な拒否を使用して、別のポリシーによって付与された権限を上書きすることができます。
条件	Condition 要素はオプションです。条件を使用すると、ポリシーをいつ適用するかを決定する式を作成できます。

Action 要素では、ワイルドカード文字 (\*) を使用して、すべての操作または操作のサブセットを指定できます。たとえば、このアクションは、s3:GetObject、s3:PutObject、s3:DeleteObject などの権限と一致します。

```
s3:*Object
```

リソース要素では、ワイルドカード文字 (\*) と (?) を使用できます。アスタリスク (\*) は 0 個以上の文字に一致しますが、疑問符 (?) は任意の 1 文字に一致します。

Principal 要素では、すべてのユーザーに権限を付与する匿名アクセスを設定する場合を除き、ワイルドカード文字はサポートされていません。たとえば、ワイルドカード (\*) をプリンシパル値として設定します。

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

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

次の例では、ステートメントは Effect、Principal、Action、および Resource 要素を使用しています。この例では、効果「許可」を使用してプリンシパル、管理者グループに権限を付与する完全なバケットポリシーステートメントを示しています。federated-group/admin`財務グループ`federated-group/finance、アクションを実行する権限`s3:ListBucket`バケットの名前`mybucket`そしてアクション`s3:GetObject`そのバケット内のすべてのオブジェクトに適用されます。

```

{
  "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/*"
      ]
    }
  ]
}

```

バケット ポリシーのサイズ制限は 20,480 バイト、グループ ポリシーのサイズ制限は 5,120 バイトです。

## 政策の一貫性

デフォルトでは、グループ ポリシーに加えた更新はすべて最終的に一貫性を持ちます。グループ ポリシーが整合すると、ポリシーのキャッシュにより、変更が有効になるまでにさらに 15 分かかることがあります。デフォルトでは、バケット ポリシーに加えた更新はすべて強力な整合性を持ちます。

必要に応じて、バケット ポリシー更新の一貫性保証を変更できます。たとえば、サイトが停止している間もバケット ポリシーの変更を利用できるようにしたい場合があります。

この場合、Consistency-Control PutBucketPolicy リクエストのヘッダーを追加するか、PUT Bucket 一貫性リクエストを使用することもできます。バケット ポリシーが整合すると、ポリシーのキャッシュにより、変更が有効になるまでにさらに 8 秒かかることがあります。



一時的な状況を回避するために一貫性を別の値に設定する場合は、完了したらバケット レベルの設定を元の値に戻してください。それ以外の場合、今後のすべてのバケット リクエストでは変更された設定が使用されます。

ポリシーステートメントで **ARN** を使用する

ポリシーステートメントでは、ARN は Principal 要素と Resource 要素で使用されます。

- S3 リソース ARN を指定するには、次の構文を使用します。

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

- 次の構文を使用して、アイデンティティ リソース ARN (ユーザーとグループ) を指定します。

```
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
```

#### その他の考慮事項:

- アスタリスク (\*) をワイルドカードとして使用して、オブジェクト キー内の 0 個以上の文字を一致させることができます。
- オブジェクト キーで指定できる国際文字は、JSON UTF-8 または JSON \u エスケープ シーケンスを使用してエンコードする必要があります。パーセントエンコーディングはサポートされていません。

#### "RFC 2141 URN構文"

PutBucketPolicy 操作の HTTP リクエスト本体は、charset=UTF-8 でエンコードする必要があります。

#### ポリシーでリソースを指定する

ポリシー ステートメントでは、Resource 要素を使用して、権限を許可または拒否するバケットまたはオブジェクトを指定できます。

- 各ポリシー ステートメントには、リソース要素が必要です。ポリシーでは、リソースは要素で表されます Resource、または、`NotResource` 除外のため。
- S3 リソース ARN を使用してリソースを指定します。例えば：

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

- オブジェクト キー内でポリシー変数を使用することもできます。例えば：

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

- リソース値には、グループ ポリシーの作成時にまだ存在しないバケットを指定できます。

#### ポリシーでプリンシパルを指定する

Principal 要素を使用して、ポリシー ステートメントによってリソースへのアクセスが許可/拒否されるユーザー、グループ、またはテナント アカウントを識別します。

- バケット ポリシー内の各ポリシー ステートメントには、Principal 要素を含める必要があります。グループ ポリシー内のポリシー ステートメントでは、グループがプリンシパルであると認識されるため、Principal 要素は必要ありません。
- ポリシーでは、プリンシパルは要素「Principal」で示され、除外する場合は「NotPrincipal」で示されません。
- アカウントベースの ID は、ID または ARN を使用して指定する必要があります。

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

- この例では、アカウント ルートとアカウント内のすべてのユーザーを含むテナント アカウント ID 27233906934684427525 を使用します。

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

- アカウント ルートのみを指定することもできます。

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

- 特定のフェデレーション ユーザー (「Alex」) を指定できます。

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

- 特定のフェデレーション グループ (「マネージャー」) を指定できます。

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

- 匿名プリンシパルを指定できます。

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

- 曖昧さを避けるために、ユーザー名の代わりにユーザーの UUID を使用できます。

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

例えば、アレックスが組織を離れ、ユーザー名が `Alex` 削除されます。新しいアレックスが組織に参加

し、同じ役割を担う場合 `Alex` ユーザー名を変更すると、新しいユーザーが元のユーザーに付与された権限を意図せず継承してしまう可能性があります。

- プリンシパル値には、バケット ポリシーの作成時にまだ存在しないグループ/ユーザー名を指定できません。

### ポリシーで権限を指定する

ポリシーでは、アクション要素はリソースへのアクセス許可を許可/拒否するために使用されます。ポリシーで指定できる一連の権限があり、これらは要素「Action」で示され、除外の場合は「NotAction」で示されます。これらの各要素は、特定の S3 REST API 操作にマッピングされます。

表には、バケットに適用される権限とオブジェクトに適用される権限がリストされています。



Amazon S3 は、PutBucketReplication アクションと DeleteBucketReplication アクションの両方に s3:PutReplicationConfiguration 権限を使用するようになりました。StorageGRID は、元の Amazon S3 仕様に一致する、アクションごとに個別の権限を使用します。



既存の値を上書きするために put が使用されると、削除が実行されます。

### バケットに適用される権限

権限	S3 REST API 操作	StorageGRID向けカスタム
s3:バケットの作成	CreateBucket	○ 注: グループ ポリシーでのみ使用します。
s3:バケットの削除	DeleteBucket	
s3>DeleteBucketMetadata通知	バケットメタデータ通知設定の削除	はい
s3>DeleteBucketポリシー	DeleteBucketPolicy	
s3:レプリケーション構成の削除	バケットレプリケーションの削除	はい、PUT とDELETEに別々の権限が必要です
s3:GetBucketAcl	GetBucketAcl	
s3:GetBucketCompliance	GET バケットコンプライアンス (非推奨)	はい
s3:GetBucketConsistency	GETバケットの一貫性	はい

権限	S3 REST API 操作	StorageGRID向けカスタム
s3:GetBucketCORS	GetBucketCors	
s3:暗号化設定の取得	バケット暗号化を取得	
s3:GetBucketLastAccessTime	GET バケットの最終アクセス時刻	はい
s3:GetBucketLocation	GetBucketLocation	
s3:GetBucketMetadataNotification	バケットメタデータ通知設定の取得	はい
s3:GetBucketNotification	GetBucketNotificationConfiguration	
s3:GetBucketObjectLockConfiguration	GetObjectLockConfiguration	
s3:GetBucketPolicy	GetBucketPolicy	
s3:GetBucketTagging	GetBucketTagging	
s3:GetBucketVersioning	GetBucketVersioning	
s3:GetLifecycleConfiguration	GetBucketLifecycleConfiguration	
s3:GetReplicationConfiguration	GetBucketReplication	
s3:すべてのバケットをリスト	<ul style="list-style-type: none"> <li>ListBuckets</li> <li>ストレージ使用量の取得</li> </ul>	<p>はい、GET ストレージ使用量の場合です。</p> <p>注: グループポリシーでのみ使用しません。</p>
s3:リストバケット	<ul style="list-style-type: none"> <li>ListObjects</li> <li>HeadBucket</li> <li>オブジェクトの復元</li> </ul>	
s3:リストバケットマルチパートアップロード	<ul style="list-style-type: none"> <li>リストマルチパートアップロード</li> <li>オブジェクトの復元</li> </ul>	
s3:バケットバージョンのリスト	GETバケットのバージョン	

権限	S3 REST API 操作	StorageGRID向けカスタム
s3:PutBucketコンプライアンス	PUT バケットコンプライアンス (非推奨)	はい
s3:PutBucketConsistency	PUT バケットの一貫性	はい
s3:PutBucketCORS	<ul style="list-style-type: none"> <li>バケットCorsの削除+</li> <li>PutBucketCors</li> </ul>	
s3:PutEncryptionConfiguration	<ul style="list-style-type: none"> <li>バケット暗号化の削除</li> <li>PutBucketEncryption</li> </ul>	
s3:PutBucket最終アクセス時間	バケットの最終アクセス時刻をPUTする	はい
s3:PutBucketMetadataNotification	PUT バケットメタデータ通知設定	はい
s3:PutBucket通知	PutBucketNotificationConfiguration	
s3:PutBucketObjectLockConfiguration	<ul style="list-style-type: none"> <li>CreateBucketで `x-amz-bucket-object-lock-enabled: true` リクエストヘッダー (s3:CreateBucket権限も必要)</li> <li>PutObjectLockConfiguration</li> </ul>	
s3:PutBucketポリシー	PutBucketPolicy	
s3:PutBucketTagging	<ul style="list-style-type: none"> <li>バケットタグの削除+</li> <li>PutBucketTagging</li> </ul>	
s3:PutBucketバージョン管理	PutBucketVersioning	
s3:PutLifecycleConfiguration	<ul style="list-style-type: none"> <li>バケットライフサイクルの削除+</li> <li>PutBucketLifecycleConfiguration</li> </ul>	
s3:PutReplicationConfiguration	PutBucketレプリケーション	はい、PUTとDELETEに別々の権限が必要です

オブジェクトに適用される権限

権限	S3 REST API 操作	StorageGRID向けカスタム
s3:マルチパートアップロードの中止	<ul style="list-style-type: none"> <li>AbortMultipartUpload</li> <li>オブジェクトの復元</li> </ul>	
s3:バイパスガバナンス保持	<ul style="list-style-type: none"> <li>DeleteObject</li> <li>DeleteObjects</li> <li>PutObjectRetention</li> </ul>	
s3:オブジェクトの削除	<ul style="list-style-type: none"> <li>DeleteObject</li> <li>DeleteObjects</li> <li>オブジェクトの復元</li> </ul>	
s3:オブジェクトのタグ付けを削除	DeleteObjectTagging	
s3:オブジェクトバージョンタグ付けの削除	DeleteObjectTagging (オブジェクトの特定のバージョン)	
s3:オブジェクトバージョンの削除	DeleteObject (オブジェクトの特定のバージョン)	
s3:GetObject	<ul style="list-style-type: none"> <li>GetObject</li> <li>HeadObject</li> <li>オブジェクトの復元</li> <li>オブジェクトコンテンツの選択</li> </ul>	
s3:GetObjectAcl	GetObjectAcl	
s3:GetObjectLegalHold	GetObjectLegalHold	
s3:GetObjectRetention	GetObjectRetention	
s3:GetObjectTagging	GetObjectTagging	
s3:GetObjectVersionTagging	GetObjectTagging (オブジェクトの特定のバージョン)	
s3:GetObjectVersion	GetObject (オブジェクトの特定のバージョン)	
s3:ListMultipartUploadParts	リストパーツ、復元オブジェクト	

権限	S3 REST API 操作	StorageGRID向けカスタム
s3:PutObject	<ul style="list-style-type: none"> <li>PutObject</li> <li>CopyObject</li> <li>オブジェクトの復元</li> <li>CreateMultipartUpload</li> <li>CompleteMultipartUpload</li> <li>UploadPart</li> <li>UploadPartCopy</li> </ul>	
s3:PutObjectLegalHold	PutObjectLegalHold	
s3:PutObjectRetention	PutObjectRetention	
s3:オブジェクトのタグ付け	PutObjectTagging	
s3:PutObjectVersionTagging	PutObjectTagging (オブジェクトの特定のバージョン)	
s3:上書きオブジェクト	<ul style="list-style-type: none"> <li>PutObject</li> <li>CopyObject</li> <li>PutObjectTagging</li> <li>DeleteObjectTagging</li> <li>CompleteMultipartUpload</li> </ul>	はい
s3:オブジェクトの復元	オブジェクトの復元	

### PutOverwriteObject権限を使用する

s3:PutOverwriteObject 権限は、オブジェクトを作成または更新する操作に適用されるカスタムStorageGRID権限です。この権限の設定により、クライアントがオブジェクトのデータ、ユーザー定義のメタデータ、または S3 オブジェクトのタグ付けを上書きできるかが決まります。

この権限で可能な設定は次のとおりです:

- 許可: クライアントはオブジェクトを上書きできます。これがデフォルト設定です。
- 拒否: クライアントはオブジェクトを上書きできません。Deny に設定すると、PutOverwriteObject 権限は次のように動作します。
  - 同じパスに既存のオブジェクトが見つかった場合:
    - オブジェクトのデータ、ユーザー定義のメタデータ、または S3 オブジェクトのタグ付けは上書きできません。
    - 進行中の取り込み操作はすべてキャンセルされ、エラーが返されます。

- S3 のバージョン管理が有効になっている場合、Deny 設定により、PutObjectTagging または DeleteObjectTagging 操作によってオブジェクトとその非現在のバージョンの TagSet が変更されることが防止されます。

◦ 既存のオブジェクトが見つからない場合、この権限は効果がありません。

- この権限が存在しない場合は、Allow が設定されている場合と同じ効果になります。



現在の S3 ポリシーで上書きが許可されていて、PutOverwriteObject 権限が Deny に設定されている場合、クライアントはオブジェクトのデータ、ユーザー定義のメタデータ、またはオブジェクトのタグ付けを上書きできません。さらに、クライアントによる変更を禁止するチェックボックスが選択されている場合 ([構成] > [セキュリティ設定] > [ネットワークとオブジェクト])、その設定は PutOverwriteObject 権限の設定よりも優先されます。

## ポリシーで条件を指定する

条件は、ポリシーがいつ有効になるかを定義します。条件は演算子とキーと値のペアで構成されます。

条件では、評価にキーと値のペアを使用します。Condition 要素には複数の条件を含めることができ、各条件には複数のキーと値のペアを含めることができます。条件ブロックは次の形式を使用します。

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

次の例では、IpAddress 条件で SourceIp 条件キーが使用されています。

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

## サポートされている条件演算子

条件演算子は次のように分類されます。

- 弦
- 数字
- ブーリアン
- IPアドレス
- ヌルチェック

条件演算子	説明
文字列に等しい	完全一致 (大文字と小文字を区別) に基づいてキーと文字列値を比較します。
文字列が等しくない	否定一致 (大文字と小文字を区別) に基づいてキーと文字列値を比較します。
文字列が等しい場合、大文字と小文字を区別しない	完全一致 (大文字と小文字は区別されません) に基づいてキーと文字列値を比較します。
文字列が等しくない場合大文字と小文字を区別しない	否定一致 (大文字と小文字を区別しない) に基づいてキーと文字列値を比較します。
文字列のような	完全一致 (大文字と小文字を区別) に基づいてキーと文字列値を比較します。* および? ワイルドカード文字を含めることができます。
文字列が似ていない	否定一致 (大文字と小文字を区別) に基づいてキーと文字列値を比較します。* および? ワイルドカード文字を含めることができます。
数値等しい	完全一致に基づいてキーと数値を比較します。
数値NotEquals	否定一致に基づいてキーと数値を比較します。
数値より大きい	「より大きい」一致に基づいてキーと数値を比較します。
数値より大きい	「以上」の一致に基づいてキーと数値を比較します。
数値の小数点以下	「より小さい」一致に基づいてキーを数値と比較します。
数値の「未満」	「以下」の一致に基づいてキーと数値を比較します。
ブール	「true または false」の一致に基づいてキーをブール値と比較します。
IPアドレス	キーを IP アドレスまたは IP アドレスの範囲と比較します。
IPアドレスなし	否定一致に基づいて、キーを IP アドレスまたは IP アドレスの範囲と比較します。
Null	現在のリクエスト コンテキストに条件キーが存在するかどうかを確認します。

サポートされている条件キー

条件キー	アクション	説明
aws:ソースIP	IPオペレーター	<p>リクエストの送信元の IP アドレスと比較します。バケットまたはオブジェクトの操作に使用できます。</p> <p>注: S3 リクエストが管理ノードおよびゲートウェイノードのロード バランサ サービスを通じて送信された場合、これはロード バランサ サービスのアップストリームの IP アドレスと比較されます。</p> <p>注: サードパーティの非透過ロードバランサが使用されている場合、これはそのロードバランサの IP アドレスと比較されます。どれでも `X-Forwarded-For` ヘッダーの有効性を確認できないため、無視されます。</p>
aws:ユーザー名	リソース/アイデンティティ	リクエストの送信元のユーザー名と比較します。バケットまたはオブジェクトの操作に使用できます。
s3:区切り文字	s3:ListBucketと s3:ListBucketVersions 権限	ListObjects または ListObjectVersions リクエストで指定された区切り文字パラメータと比較します。

条件キー	アクション	説明
s3:ExistingObjectTag/<タグキー>	s3:オブジェクトのタグ付けを削除 s3:オブジェクトバージョンタグ付けの削除 s3:GetObject s3:GetObjectAcl 3:オブジェクトのタグ付けを取得する s3:GetObjectVersion s3:GetObjectVersionAcl s3:GetObjectVersionTagging s3:PutObjectAcl s3:オブジェクトのタグ付け s3:PutObjectVersionAcl s3:PutObjectVersionTagging	既存のオブジェクトに特定のタグ キーと値があることが必要になります。
s3:最大キー数	s3:ListBucketと s3:ListBucketVersions 権限	ListObjects または ListObjectVersions リクエストで指定された max-keys パラメータと比較します。
s3:オブジェクトロックの残り保持日数	s3:PutObject	指定された保持期限日と比較します `x-amz-object-lock-retain-until-date` リクエスト ヘッダーまたはバケットのデフォルトの保持期間から計算され、これらの値が次のリクエストの許容範囲内であることを確認します。 <ul style="list-style-type: none"> <li>• PutObject</li> <li>• CopyObject</li> <li>• CreateMultipartUpload</li> </ul>
s3:オブジェクトロックの残り保持日数	s3:PutObjectRetention	PutObjectRetention リクエストで指定された retain-until-date と比較して、それが許容範囲内であることを確認します。

条件キー	アクション	説明
s3:プレフィックス	s3:ListBucketと s3:ListBucketVersions 権限	ListObjects または ListObjectVersions リクエストで指定されたプレフィックス パラメータと比較します。
s3:RequestObjectTag/<タグキー>	s3:PutObject s3:オブジェクトのタグ付け s3:PutObjectVersionTagging	オブジェクト リクエストにタグ付けが含まれている場合は、特定のタグ キーと値が必要になります。

### ポリシーで変数を指定する

利用可能な場合にポリシー情報を入力するには、ポリシー内の変数を使用できます。ポリシー変数は、`Resource` 要素と文字列の比較では `Condition` 要素。

この例では、変数 `\${aws:username}` リソース要素の一部です:

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

この例では、変数 `\${aws:username}` 条件ブロック内の条件値の一部です。

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

変数	説明
\${aws:SourceIp}	提供された変数として SourceIp キーを使用します。
\${aws:username}	提供された変数としてユーザー名キーを使用します。
\${s3:prefix}	提供された変数としてサービス固有のプレフィックス キーを使用します。
\${s3:max-keys}	提供された変数としてサービス固有の max-keys キーを使用します。
\${*}	特殊文字。文字をリテラル * 文字として使用します。

変数	説明
<code>\${?}</code>	特殊文字。文字をリテラル ? 文字として使用します。
<code>\${\$}</code>	特殊文字。文字をリテラル \$ 文字として使用します。

### 特別な処理を必要とするポリシーを作成する

場合によっては、ポリシーによって、アカウントのルート ユーザーをロックアウトするなど、セキュリティ上危険な権限や継続的な操作にとって危険な権限が付与されることがあります。StorageGRID S3 REST API 実装は、ポリシー検証時には Amazon よりも制限が少なくなりますが、ポリシー評価時には同様に厳格になります。

ポリシーの説明	ポリシータイプ	アマゾンの行動	StorageGRIDの動作
ルートアカウントへのすべての権限を拒否する	バケット	有効かつ強制されていますが、ルートユーザーアカウントはすべての S3 バケットポリシー操作の権限を保持します。	同じ
ユーザー/グループへのすべての権限を拒否する	グループ	有効かつ強制	同じ
外国のアカウントグループに任意の権限を許可する	バケット	無効なプリンシパル	有効ですが、すべての S3 バケットポリシー操作の権限は、ポリシーで許可されている場合、405 メソッドが許可されていないエラーを返します。
外部アカウントのルートまたはユーザーに任意の権限を許可する	バケット	有効ですが、すべての S3 バケットポリシー操作の権限は、ポリシーで許可されている場合、405 メソッドが許可されていないエラーを返します。	同じ
すべてのアクションに対する権限を全員に許可する	バケット	有効ですが、すべての S3 バケットポリシー操作の権限は、外部アカウントのルートとユーザーに対して 405 Method Not Allowed エラーを返します。	同じ

ポリシーの説明	ポリシータイプ	アマゾンの行動	StorageGRIDの動作
すべてのアクションに対する全員の権限を拒否	バケット	有効かつ強制されていますが、ルートユーザーアカウントはすべての S3 バケットポリシー操作の権限を保持します。	同じ
プリンシパルは存在しないユーザーまたはグループです	バケット	無効なプリンシパル	有効
リソースは存在しないS3バケットです	グループ	有効	同じ
校長は地元のグループです	バケット	無効なプリンシパル	有効
ポリシーは、所有者以外のアカウント (匿名アカウントを含む) にオブジェクトを配置する権限を付与します。	バケット	有効。オブジェクトは作成者アカウントによって所有され、バケットポリシーは適用されません。作成者アカウントは、オブジェクトACLを使用してオブジェクトへのアクセス権限を付与する必要があります。	有効。オブジェクトはバケット所有者アカウントによって所有されます。バケットポリシーが適用されます。

#### 一度書き込み、何度も読み取り可能な (WORM) 保護

データ、ユーザー定義のオブジェクトメタデータ、および S3 オブジェクトのタグ付けを保護するために、Write Once Read Many (WORM) バケットを作成できます。WORM バケットを構成して、新しいオブジェクトの作成を許可し、既存のコンテンツの上書きや削除を防止します。ここで説明するいずれかのアプローチを使用します。

上書きが常に拒否されるようにするには、次の操作を実行します。

- グリッド マネージャーから、[構成] > [セキュリティ] > [セキュリティ設定] > [ネットワークとオブジェクト] に移動し、[クライアントの変更を禁止する] チェックボックスをオンにします。
- 次のルールと S3 ポリシーを適用します。
  - S3 ポリシーに PutOverwriteObject DENY 操作を追加します。
  - S3 ポリシーに DeleteObject DENY 操作を追加します。
  - S3 ポリシーに PutObject ALLOW 操作を追加します。



S3 ポリシーで DeleteObject を DENY に設定しても、「30 日後にコピーがゼロ」などのルールが存在する場合は ILM によるオブジェクトの削除を防ぐことはできません。



これらすべてのルールとポリシーを適用しても、同時書き込みを防ぐことはできません (状況 A を参照)。連続した完了上書きは防止します (状況 B を参照)。

状況A: 同時書き込み（防御されていない）

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

状況B: 連続した完了上書き（防止対象）

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

#### 関連情報

- ["StorageGRID ILMルールによるオブジェクトの管理方法"](#)
- ["バケットポリシーの例"](#)
- ["グループポリシーの例"](#)
- ["ILMでオブジェクトを管理する"](#)
- ["テナントアカウントを使用する"](#)

#### バケットポリシーの例

このセクションの例を使用して、バケットのStorageGRIDアクセス ポリシーを構築します。

バケット ポリシーは、ポリシーが添付されているバケットのアクセス権限を指定します。バケットポリシーは、次のいずれかのツールを通じて S3 PutBucketPolicy API を使用して設定します。

- ["Tenant Manager"](#)。
- このコマンドを使用するAWS CLI（参照["バケットの操作"](#)）:

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

例: すべてのユーザーにバケットへの読み取り専用アクセスを許可する

この例では、匿名ユーザーを含むすべてのユーザーがバケット内のオブジェクトを一覧表示し、バケット内のすべてのオブジェクトに対して GetObject 操作を実行できます。その他の操作はすべて拒否されます。アカウント ルート以外のユーザーにはバケットへの書き込み権限がないため、このポリシーは特に役に立たない可能性があることに注意してください。

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

例: あるアカウントの全員にバケットへのフルアクセスを許可し、別のアカウントの全員に読み取り専用アクセスを許可する

この例では、指定された1つのアカウントの全員にバケットへのフルアクセスが許可されていますが、指定された別のアカウントの全員にはバケットの一覧表示と、バケット内の先頭のオブジェクトに対するGetObject操作の実行のみが許可されています。`shared/`オブジェクト キーのプレフィックス。



StorageGRIDでは、所有者以外のアカウント (匿名アカウントを含む) によって作成されたオブジェクトは、バケット所有者アカウントによって所有されます。バケット ポリシーはこれらのオブジェクトに適用されます。

```

{
  "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/*"
        }
      }
    }
  ]
}

```

例: バケットへの読み取り専用アクセスを全員に許可し、指定されたグループにはフルアクセスを許可する

この例では、匿名ユーザーを含むすべてのユーザーがバケットの一覧を表示し、バケット内のすべてのオブジェクトに対してGetObject操作を実行できますが、グループに属するユーザーのみが `Marketing` 指定されたアカウントのユーザーにはフルアクセスが許可されます。

```

{
  "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/*"
      ]
    }
  ]
}

```

例: IP範囲内のクライアントにバケットへの読み取りと書き込みアクセスを許可する

この例では、リクエストが指定された IP 範囲 (54.240.143.0 ~ 54.240.143.255、54.240.143.188 を除く) から送信された場合、匿名ユーザーを含むすべてのユーザーがバケットを一覧表示し、バケット内のすべてのオブジェクトに対して任意のオブジェクト操作を実行できます。その他のすべての操作は拒否され、IP 範囲外のすべての要求は拒否されます。

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

例: 指定されたフェデレーションユーザーのみがバケットにフルアクセスできるようにする

この例では、フェデレーションユーザーAlexは、examplebucket`バケットとそのオブジェクト。`root`を含む他のすべてのユーザーは、すべての操作が明示的に拒否されます。ただし、`root`にはPut/Get/DeleteBucketPolicy へのアクセス許可が拒否されることはないことに注意してください。

```

{
  "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/*"
      ]
    }
  ]
}

```

#### 例: PutOverwriteObject 権限

この例では、Deny PutOverwriteObject および DeleteObject の効果により、オブジェクトのデータ、ユーザー定義のメタデータ、および S3 オブジェクトのタグ付けを誰も上書きまたは削除できないようになります。

```

{
  "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/*"
    }
  ]
}

```

## グループポリシーの例

このセクションの例を使用して、グループのStorageGRIDアクセス ポリシーを構築します。

グループ ポリシーは、ポリシーが関連付けられているグループのアクセス権限を指定します。ありません `Principal` 暗黙的であるため、ポリシーの要素ではありません。グループ ポリシーは、テナント マネージャー または API を使用して構成されます。

例: テナント マネージャーを使用してグループ ポリシーを設定する

テナント マネージャーでグループを追加または編集するときに、グループ ポリシーを選択して、このグループのメンバーに付与する S3 アクセス権を決定できます。見る["S3テナントのグループを作成する"](#)。

- **S3 アクセスなし:** デフォルト オプション。このグループのユーザーは、バケットポリシーでアクセスが許可されない限り、S3 リソースにアクセスできません。このオプションを選択すると、デフォルトではルートユーザーのみが S3 リソースにアクセスできるようになります。
- **読み取り専用アクセス:** このグループのユーザーには、S3 リソースへの読み取り専用アクセス権があります。たとえば、このグループのユーザーはオブジェクトを一覧表示したり、オブジェクトのデータ、メタデータ、タグを読み取ったりできます。このオプションを選択すると、読み取り専用グループ ポリシーの JSON 文字列がテキスト ボックスに表示されます。この文字列は編集できません。
- **フルアクセス:** このグループのユーザーには、バケットを含む S3 リソースへのフルアクセス権があります。このオプションを選択すると、フルアクセス グループ ポリシーの JSON 文字列がテキスト ボックスに表示されます。この文字列は編集できません。
- **ランサムウェア軽減:** このサンプル ポリシーは、このテナントのすべてのバケットに適用されます。このグループのユーザーは一般的なアクションを実行できますが、オブジェクトのバージョン管理が有効になっているバケットからオブジェクトを完全に削除することはできません。

すべてのバケットの管理権限を持つテナント マネージャー ユーザーは、このグループ ポリシーを上書きできます。すべてのバケットの管理権限を信頼できるユーザーに制限し、可能な場合は多要素認証 (MFA) を使用します。

- **カスタム:** グループ内のユーザーには、テキスト ボックスで指定した権限が付与されます。

例: グループにすべてのバケットへのフルアクセスを許可する

この例では、バケット ポリシーによって明示的に拒否されない限り、グループのすべてのメンバーに、テナント アカウントが所有するすべてのバケットへのフル アクセスが許可されます。

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

例: すべてのバケットへのグループ読み取り専用アクセスを許可する

この例では、バケット ポリシーによって明示的に拒否されない限り、グループのすべてのメンバーは S3 リソースへの読み取り専用アクセス権を持ちます。たとえば、このグループのユーザーはオブジェクトを一覧表示したり、オブジェクトのデータ、メタデータ、タグを読み取ったりできます。

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

例: グループメンバーにバケット内の自分の「フォルダ」のみへのフルアクセスを許可する

この例では、グループのメンバーは、指定されたバケット内の特定のフォルダー (キー プレフィックス) の一覧表示とアクセスのみが許可されます。これらのフォルダーのプライバシーを決定する際には、他のグループポリシーとバケット ポリシーからのアクセス権限を考慮する必要があることに注意してください。

```

{
  "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}/*"
    }
  ]
}

```

## 監査ログで追跡されるS3操作

監査メッセージはStorageGRIDサービスによって生成され、テキスト ログ ファイルに保存されます。監査ログで S3 固有の監査メッセージを確認して、バケットおよびオブジェクトの操作に関する詳細を取得できます。

### 監査ログで追跡されるバケット操作

- CreateBucket
- DeleteBucket
- バケットタグの削除
- DeleteObjects
- GetBucketTagging
- HeadBucket
- ListObjects
- ListObjectVersions
- PUTバケットコンプライアンス
- PutBucketTagging
- PutBucketVersioning

## 監査ログで追跡されるオブジェクト操作

- CompleteMultipartUpload
- CopyObject
- DeleteObject
- GetObject
- HeadObject
- PutObject
- オブジェクトの復元
- オブジェクトを選択
- UploadPart (ILM ルールが Balanced または Strict の取り込みを使用する場合)
- UploadPartCopy (ILM ルールが Balanced または Strict の取り込みを使用する場合)

### 関連情報

- ["アクセス監査ログファイル"](#)
- ["クライアント書き込み監査メッセージ"](#)
- ["クライアント読み取り監査メッセージ"](#)

## 著作権に関する情報

Copyright © 2025 NetApp, Inc. All Rights Reserved. Printed in the U.S.このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

ネットアップの著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、ネットアップによって「現状のまま」提供されています。ネットアップは明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。ネットアップは、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

ネットアップは、ここに記載されているすべての製品に対する変更を随時、予告なく行う権利を保有します。ネットアップによる明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、ネットアップは責任を負いません。この製品の使用または購入は、ネットアップの特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許によって保護されている場合があります。

権利の制限について：政府による使用、複製、開示は、DFARS 252.227-7013（2014年2月）およびFAR 5252.227-19（2007年12月）のRights in Technical Data -Noncommercial Items（技術データ - 非商用品目に関する諸権利）条項の(b)(3)項、に規定された制限が適用されます。

本書に含まれるデータは商用製品および/または商用サービス（FAR 2.101の定義に基づく）に関係し、データの所有権はNetApp, Inc.にあります。本契約に基づき提供されるすべてのネットアップの技術データおよびコンピュータソフトウェアは、商用目的であり、私費のみで開発されたものです。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用权を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc.の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用权については、DFARS 252.227-7015(b)項（2014年2月）で定められた権利のみが認められます。

## 商標に関する情報

NetApp、NetAppのロゴ、<http://www.netapp.com/TM>に記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。