StorageGRIDを使用する

StorageGRID software

NetApp
July 23, 2026

目次

StorageGRID テナントとクライアントの使用

テナントアカウントを使用する

StorageGRIDテナントアカウントを使用する

テナントアカウントを使用すると、Simple Storage Service（S3）REST API を使用して StorageGRID システムにオブジェクトを保存および取得できます。

テナントアカウントとは何ですか？

各テナントアカウントには、独自のフェデレーショングループまたはローカルグループ、ユーザー、S3バケット、およびオブジェクトがあります。

テナントアカウントを使用すると、保管されているオブジェクトを異なるエンティティごとに分類できます。例えば、複数のテナントアカウントは、以下のいずれかのユースケースに使用できます。

- 企業での使用例： StorageGRID システムが企業内で使用される場合、グリッドのオブジェクトストレージは組織内の異なる部門ごとに分離される可能性があります。例えば、マーケティング部門、カスタマーサポート部門、人事部門など、各部門ごとにテナントアカウントが存在する可能性があります。

> S3クライアントプロトコルを使用する場合、S3バケットとバケットポリシーを使用して、企業内の部門間でオブジェクトを分離することもできます。テナントアカウントを個別に作成する必要はありません。詳細については、"S3バケットとバケットポリシー"の実装手順を参照してください。

- サービスプロバイダーのユースケース： StorageGRID システムがサービスプロバイダーによって使用されている場合、グリッドのオブジェクトストレージは、ストレージをリースしている異なるエンティティによって分離される可能性があります。たとえば、Company A、Company B、Company C などのテナントアカウントが存在する場合があります。

テナントアカウントの作成方法

テナントアカウントは、"StorageGRID Grid Managerを使用するグリッド管理者"によって作成されます。テナントアカウントを作成する際、グリッド管理者は以下を指定します。

- テナント名、クライアントタイプ（S3）、オプションのストレージクォータなどの基本情報。
- テナントアカウントの権限。例えば、テナントアカウントが S3 プラットフォームサービスを利用できるかどうか、独自の ID ソースを設定できるかどうか、S3 Select を使用できるかどうか、またはグリッドフェデレーション接続を使用できるかどうかなど。
- テナントの初期ルートアクセスは、StorageGRID システムがローカルグループとユーザー、IDフェデレーション、またはシングル サインオン（SSO）を使用するかどうかによって異なります。

さらに、S3テナントアカウントが規制要件を遵守する必要がある場合、グリッド管理者はStorageGRIDシステムのS3オブジェクトロック設定を有効にすることができます。S3オブジェクトロックが有効になっている場合、すべてのS3テナントアカウントは準拠したバケットを作成および管理できます。

"S3テナントアカウントが作成されました"の後、テナントマネージャーにアクセスして、次のようなタスクを実行できます：

- IDフェデレーションを設定します（IDソースがグリッドと共有されている場合を除く）
- グループとユーザーを管理する
- アカウントクローンとクロスグリッドレプリケーションにはグリッドフェデレーションを使用する
- S3アクセスキーの管理
- S3バケットの作成と管理
- S3プラットフォームサービスを使用する
- S3 Selectを使用する
- ストレージ使用量を監視する

テナントマネージャーでS3バケットを作成および管理することはできますが、オブジェクトを取り込んで管理するには"S3クライアント"または"S3コンソール"を使用する必要があります。

サインインとサインアウトの方法

StorageGRID テナントマネージャーにサインインする

テナントマネージャーにアクセスするには、テナントのURLを"対応ウェブブラウザ"のアドレスバーに入力します。

開始する前に

- ログイン情報をお持ちです。
- テナントマネージャーにアクセスするためのURLは、グリッド管理者から提供されています。URLは、以下の例のいずれかのようになります。

```
https://FQDN_or_Admin_Node_IP/

https://FQDN_or_Admin_Node_IP:port/

https://FQDN_or_Admin_Node_IP/?accountId=20-digit-account-id

https://FQDN_or_Admin_Node_IP:port/?accountId=20-digit-account-id
```

  URLには必ず、完全修飾ドメイン名（FQDN）、管理ノードのIPアドレス、または管理ノードのHAグループの仮想IPアドレスが含まれます。これには、ポート番号、20桁のテナントアカウントID、またはその両方が含まれる場合もあります。

- URLにテナントの20桁のアカウントIDが含まれていない場合は、このアカウントIDを持っています。
- あなたは"対応ウェブブラウザ"を使用しています。
- お使いのウェブブラウザでCookieが有効になっています。
- あなたは、"特定のアクセス権限" を持つユーザーグループに属しています。

手順

1. "対応ウェブブラウザ"を起動します。
2. ブラウザのアドレスバーに、Tenant Manager にアクセスするための URL を入力します。
3. セキュリティ警告が表示された場合は、ブラウザのインストールウィザードを使用して証明書をインストールしてください。
4. テナントマネージャーにサインインします。

   表示されるサインイン画面は、入力したURLと、StorageGRID にシングル サインオン（SSO）が設定されているかどうかによって異なります。

   SSOを使用していません

   StorageGRID が SSO を使用していない場合、次のいずれかの画面が表示されます。

   - グリッドマネージャーのサインインページ。「テナントサインイン」リンクを選択してください。
   - テナントマネージャのサインインページ。「アカウント」フィールドはすでに入力されている場合があります。
     i. テナントの20桁のアカウントIDが表示されない場合は、最近使用したアカウントの一覧にテナントアカウント名が表示されている場合はそれを選択するか、アカウントIDを入力してください。
     ii. ユーザー名とパスワードを入力してください。
     iii. 「サインイン」を選択します。

     テナントマネージャーのダッシュボードが表示されます。

     iv. 他の人から初期パスワードを受け取った場合は、ユーザー名 > *パスワードの変更*を選択してアカウントを保護してください。

   SSOの使用

   StorageGRID が SSO を使用している場合、次のいずれかの画面が表示されます。

   - 組織の SSO ページ。

     標準の SSO 認証情報を入力し、*サインイン*を選択してください。

   - テナントマネージャー SSO サインインページ。
     i. テナントの20桁のアカウントIDが表示されない場合は、最近使用したアカウントの一覧にテナントアカウント名が表示されている場合はそれを選択するか、アカウントIDを入力してください。
     ii. 「サインイン」を選択します。
     iii. 組織のSSOサインインページで、通常のSSO認証情報を使用してサインインしてください。

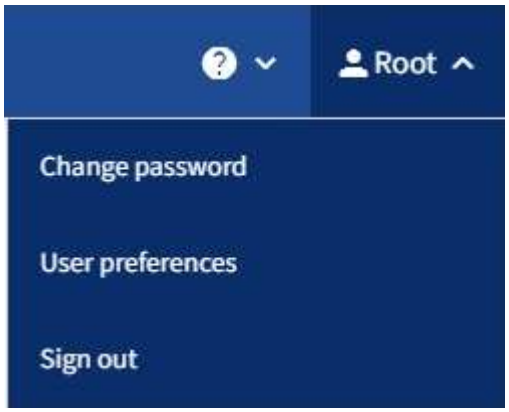
     テナントマネージャーのダッシュボードが表示されます。

StorageGRID Tenant Managerからサインアウトします

テナントマネージャーでの作業が完了したら、不正なユーザーが StorageGRID システムにアクセスできないように、必ずサインアウトしてください。ブラウザの Cookie 設定によっては、ブラウザを閉じてもシステムからサインアウトされない場合があります。

手順

1. ユーザーインターフェースの右上隅にあるユーザー名のドロップダウンメニューを探してください。



2. ユーザー名を選択し、次に*サインアウト*を選択してください。

   - SSOが使用されていない場合：

     管理者ノードからサインアウトしました。Tenant Manager のサインインページが表示されます。

     複数の管理ノードにサインインした場合は、各ノードからサインアウトする必要があります。

   - SSOが有効になっている場合：

     アクセスしていたすべての管理ノードからサインアウトされました。StorageGRID のサインインページが表示されます。最近使用したアカウント*ドロップダウンには、先ほどアクセスしたテナントアカウントの名前がデフォルトで表示され、テナントの*アカウント **ID** も表示されます。

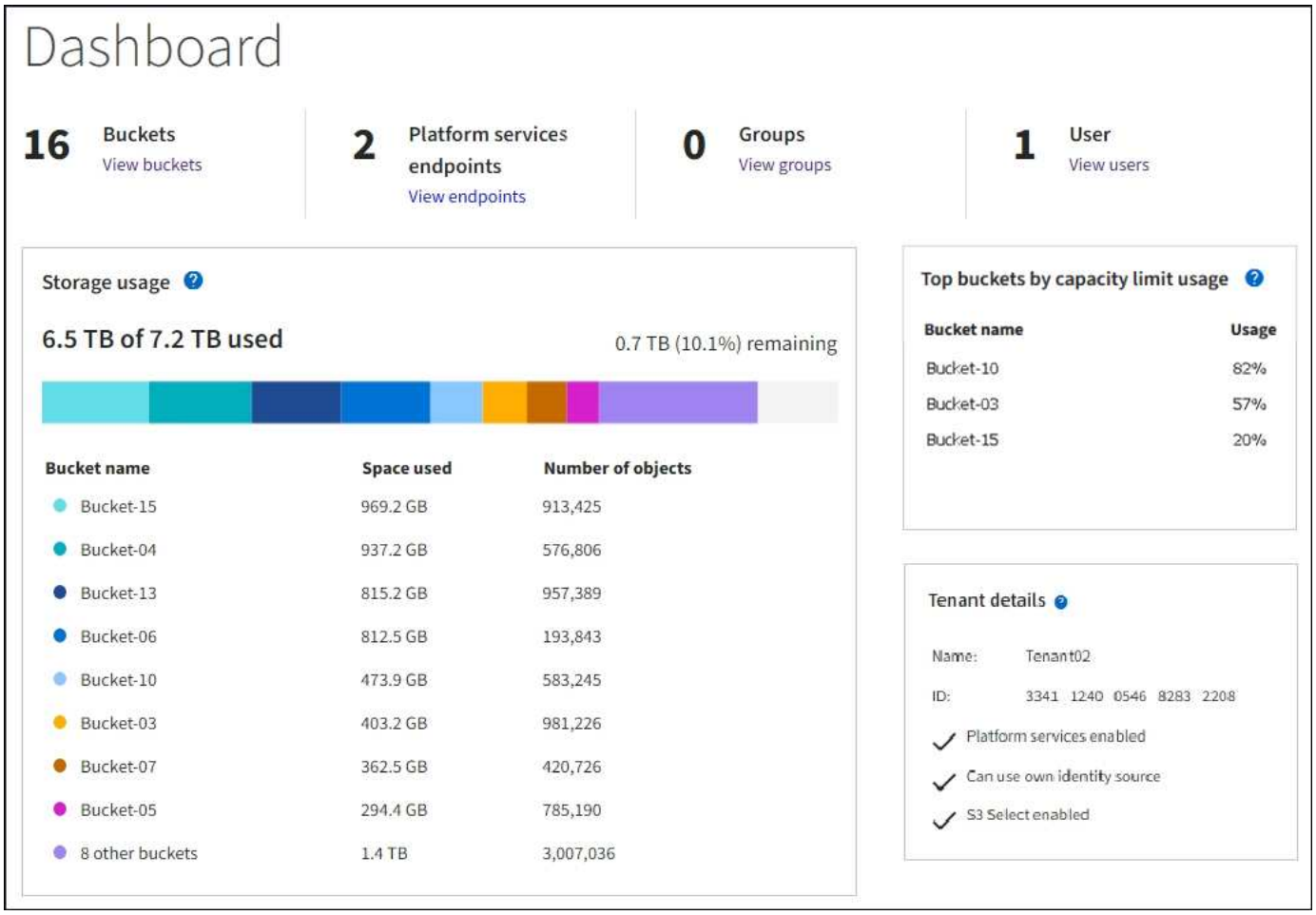
     SSO が有効になっていて、Grid Manager にもサインインしている場合は、SSO からサインアウトするために Grid Manager からもサインアウトする必要があります。

StorageGRIDのテナントマネージャーダッシュボード

テナントマネージャーのダッシュボードには、テナントアカウントの設定概要と、テナントのS3バケット内のオブジェクトが使用している容量が表示されます。テナントに割り当て量がある場合、ダッシュボードには割り当て量のどれだけが使用済みで、どれだけが残っているかが表示されます。テナントアカウントに関連するエラーが発生した場合、そのエラーはダッシュボードに表示されます。

このテナントに属するすべてのオブジェクトの論理サイズには、未完了および進行中のマルチパートアップロードが含まれます。このサイズには、ILMポリシーのために使用される追加の物理スペースは含まれていません。使用スペースの値は概算値です。これらの推定値は、データ取り込みのタイミング、ネットワーク接続性、およびノードの状態によって影響を受けます。

オブジェクトがアップロードされると、ダッシュボードは次の例のようになります。



テナントアカウント情報

ダッシュボードの上部には、設定済みのバケットまたはコンテナの数、グループ数、およびユーザー数が表示されます。設定されているプラットフォームサービスエンドポイントの数も表示されます。詳細を表示するには、リンクを選択してください。

お持ちの"テナント管理権限"と設定したオプションに応じて、ダッシュボードの残りの部分には、ガイドライン、ストレージ使用量、オブジェクト情報、テナントの詳細など、さまざまな組み合わせが表示されます。

ストレージとクォータの使用状況

ストレージ使用状況パネルには、以下の情報が含まれています：

- テナントのオブジェクトデータの量。

  この値はアップロードされたオブジェクトデータの総量を示しており、それらのオブジェクトのコピーと

そのメタデータを保存するために使用される容量を表すものではありません。

- クォータが設定されている場合、オブジェクトデータに使用可能な総容量、および残りの容量とその割合が表示されます。クォータは、取り込むことができるオブジェクトデータの量を制限します。

> 割り当て使用量は内部推定値に基づいており、場合によっては超過する可能性があります。例えば、StorageGRID はテナントがオブジェクトのアップロードを開始した際にクォータをチェックし、テナントがクォータを超過している場合は新しい取り込みを拒否します。ただし、StorageGRID はクォータを超過したかどうかを判断する際に、現在のアップロードのサイズは考慮されません。オブジェクトが削除された場合、クォータ使用量が再計算されるまで、テナントは一時的に新しいオブジェクトをアップロードできなくなる可能性があります。割り当て使用量の計算には 10 分以上かかる場合があります。

- 最も大きなバケットまたはコンテナの相対的なサイズを表す棒グラフ。

  グラフ上の任意のセグメントにカーソルを合わせると、そのバケットまたはコンテナが占める合計容量を確認できます。

- 棒グラフに対応するため、オブジェクトデータの総量と各バケットまたはコンテナ内のオブジェクト数を含む、最大のバケットまたはコンテナのリスト。

Bucket name	Space used	Number of objects
Bucket-02	944.7 GB	7,575
Bucket-09	899.6 GB	589,677
Bucket-15	889.6 GB	623,542
Bucket-06	846.4 GB	648,619
Bucket-07	730.8 GB	808,655
Bucket-04	700.8 GB	420,493
Bucket-11	663.5 GB	993,729
Bucket-03	656.9 GB	379,329
9 other buckets	2.3 TB	5,171,588

テナントに9個を超えるバケットまたはコンテナがある場合、それ以外のすべてのバケットまたはコンテナは、リストの一番下に1つの項目としてまとめられます。

> テナントマネージャーに表示されるストレージ値の単位を変更するには、テナントマネージャーの右上にあるユーザードロップダウンを選択し、*User preferences*を選択します。

割り当て量使用状況アラート

グリッドマネージャーでクォータ使用状況アラートが有効になっている場合、クォータが不足または超過した際に、テナントマネージャーに以下のアラートが表示されます：

- テナントのクォータの90%以上が使用された場合、**Tenant quota usage high** アラートがトリガーされます。

  グリッド管理者に割り当て量を増やすよう依頼することを検討してください。

- 割り当て量を超過すると、新しいオブジェクトをアップロードできないことを知らせる通知が表示されます。

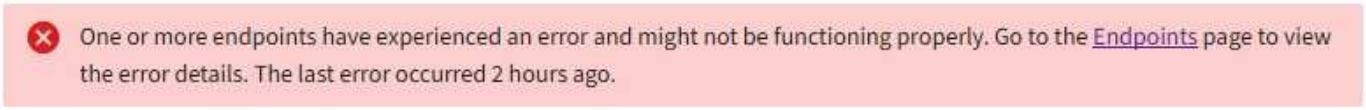
容量制限の使用状況

バケットに容量制限を設定している場合、テナントマネージャーのダッシュボードには、容量制限の使用状況に基づいて上位にランク付けされたバケットのリストが表示されます。

バケットに制限が設定されていない場合、その容量は無制限です。ただし、テナントアカウントにストレージ容量の上限が設定されており、その上限に達した場合、バケットの残りの容量制限に関係なく、それ以上オブジェクトを取り込むことはできません。

エンドポイントエラー

Grid Managerを使用してプラットフォームサービスで使用するエンドポイントを1つ以上設定した場合、過去7日以内にエンドポイントエラーが発生すると、テナントマネージャのダッシュボードにアラートが表示されます。



"プラットフォームサービスエンドポイントエラー"の詳細を確認するには、*エンドポイント*を選択してエンドポイントページを表示します。

テナント管理API

StorageGRIDテナント管理APIについて学ぶ

テナントマネージャーのユーザーインターフェースの代わりに、Tenant Management REST API を使用してシステム管理タスクを実行できます。たとえば、API を使用して操作を自動化したり、ユーザーなどの複数のエンティティをより迅速に作成したりすることができます。

テナント管理API：

- Swagger オープンソース API プラットフォームを使用します。Swagger は、開発者だけでなく非開発者も API とやり取りできる直感的なユーザーインターフェースを提供します。Swagger のユーザーインターフェースは、各 API 操作に関する詳細な情報とドキュメントを提供します。
- "無停止アップグレードをサポートするためのバージョン管理"を使用します。

テナント管理APIのSwaggerドキュメントにアクセスするには：

1. テナントマネージャーにサインインします。
2. テナントマネージャーの上部にあるヘルプアイコンを選択し、*APIドキュメント*を選択します。

APIの処理

テナント管理APIは、利用可能なAPI操作を以下のセクションに分類しています：

- アカウント：現在のテナントアカウントに対する操作。ストレージ使用状況情報の取得などが含まれます。
- **auth**：ユーザーセッション認証を実行するための操作。

  テナント管理APIは、ベアラートークン認証方式をサポートしています。テナントログインには、ユーザー名、パスワード、および accountId を認証リクエストの JSON ボディ（つまり、 `POST /api/v3/authorize`）に指定します。ユーザー認証が成功すると、セキュリティトークンが返されます。このトークンは、以降のAPIリクエストのヘッダーに必ず含める必要があります（「Authorization: Bearer token」）。

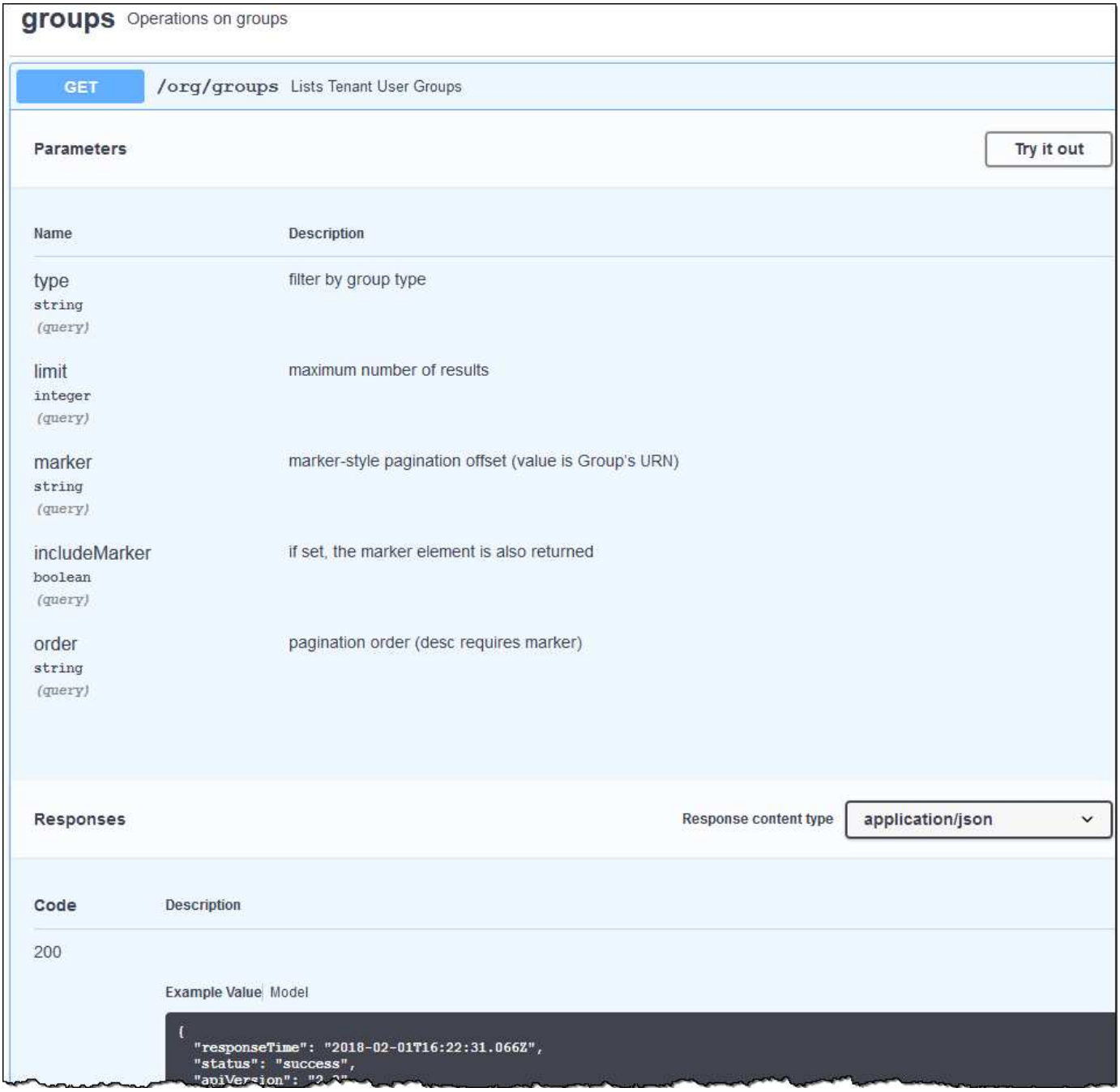
  認証セキュリティの改善に関する情報については、"クロスサイトリクエストフォージェリ攻撃から保護する"を参照してください。

  > StorageGRID システムでシングル サインオン（SSO）が有効になっている場合は、認証のために異なる手順を実行する必要があります。"グリッド管理APIの使用方法に関する手順"を参照してください。

- **config**: 製品リリースおよびTenant Management APIのバージョンに関連する操作。製品のリリースバージョンと、そのリリースでサポートされているAPIのメジャーバージョンを一覧表示できます。
- コンテナ：S3バケットに対する操作。
- **deactivated-features**：無効化された可能性のある機能を表示するための操作。
- エンドポイント：エンドポイントを管理するための操作。エンドポイントを使用すると、S3 バケットが外部サービスを使用して StorageGRID CloudMirror レプリケーション、通知、または検索統合を行うことができます。
- **grid-federation-connections**：グリッドフェデレーション接続およびクロスグリッドレプリケーションに関する操作。
- グループ：ローカルテナントグループを管理し、外部IDソースからフェデレーションテナントグループを取得するための操作。
- **identity-source**： 外部IDソースを設定し、フェデレーションされたグループとユーザー情報を手動で同期するための操作。
- **ilm**: 情報ライフサイクル管理（ILM）設定に対する操作。
- **regions**：StorageGRID システムに設定されているリージョンを確認するための操作。
- **s3**: テナントユーザーのS3アクセスキーを管理する操作。
- **s3-object-lock**：規制遵守をサポートするために使用される、グローバルな S3 Object Lock 設定に対する操作。
- ユーザー：テナントユーザーを表示および管理するための操作。

操作の詳細

各API操作を展開すると、そのHTTPアクション、エンドポイントURL、必須またはオプションのパラメータのリスト、リクエストボディの例（必要な場合）、および考えられるレスポンスが表示されます。



API リクエストを発行する

APIドキュメントのウェブページを使用して実行するAPI操作はすべて、ライブ操作となります。設定データやその他のデータを誤って作成、更新、または削除しないように注意してください。

手順

1. リクエストの詳細を表示するには、HTTPアクションを選択してください。

2. リクエストにグループIDやユーザーIDなどの追加パラメータが必要かどうかを判断します。次に、これらの値を取得します。必要な情報を取得するには、まず別のAPIリクエストを発行する必要がある場合があります。
3. サンプルリクエストボディを修正する必要があるかどうかを判断してください。必要な場合は、*モデル*を選択して各フィールドの要件を確認できます。
4. 「Try it out」を選択します。
5. 必要なパラメータを指定するか、必要に応じてリクエスト本文を変更してください。
6. 「実行」を選択します。
7. リクエストが成功したかどうかを判断するには、レスポンスコードを確認してください。

StorageGRID テナント管理 **API** のバージョン管理

テナント管理APIは、バージョン管理を利用して、システムへの影響を最小限に抑えたアップグレードをサポートします。

例えば、このリクエスト URL は API のバージョン 4 を指定しています。

```
https://hostname_or_ip_address/api/v4/authorize
```

APIのメジャーバージョンは、古いバージョンと互換性のない変更が行われた場合、更新されます。APIのマイナーバージョンは、古いバージョンと互換性のある変更が行われた場合、更新されます。互換性のある変更には、新しいエンドポイントまたは新しいプロパティの追加が含まれます。

以下の例は、行われた変更の種類に基づいてAPIバージョンがどのように更新されるかを示しています。

APIに対する変更の種類	旧バージョン	新バージョン
旧バージョンとの互換性あり	2.1	2.2
旧バージョンとの互換性はありません	2.1	3.0

StorageGRID ソフトウェアを初めてインストールする場合、最新バージョンの API のみが有効になります。ただし、StorageGRID の新しい機能リリースにアップグレードした場合でも、少なくとも 1 つの StorageGRID 機能リリースの間は、古い API バージョンに引き続きアクセスできます。

> 対応バージョンを設定できます。詳細については、"グリッド管理API" の Swagger API ドキュメントの **config** セクションを参照してください。すべての API クライアントを新しいバージョンに更新した後、古いバージョンのサポートを無効にする必要があります。

古いリクエストは、以下の方法で非推奨としてマークされます。

- レスポンスヘッダーは「Deprecated: true」です。
- JSON レスポンスボディには「deprecated」：true が含まれています
- 非推奨の警告がnms.logに追加されます。例：

```
Received call to deprecated v2 API at POST "/api/v2/authorize"
```

現在のリリースでサポートされている**API**バージョンを確認する

`GET /versions`API リクエストを使用して、サポートされている API のメジャーバージョンのリストを返します。このリクエストは、Swagger API ドキュメントの *config* セクションに記載されています。

```
GET https://{{IP-Address}}/api/versions
{
  "responseTime": "2023-06-27T22:13:50.750Z",
  "status": "success",
  "apiVersion": "4.0",
  "data": [
    2,
    3,
    4
  ]
}
```

リクエストの**API**バージョンを指定します

パスパラメータ（(/api/v4）またはヘッダー（(Api-Version: 4）を使用して API バージョンを指定できます。両方の値を指定した場合、ヘッダーの値がパスの値を上書きします。

```
curl https://[IP-Address]/api/v4/grid/accounts

curl -H "Api-Version: 4" https://[IP-Address]/api/grid/accounts
```

StorageGRIDでクロスサイトリクエストフォージェリから保護する

CSRFトークンを使用してCookieを使用する認証を強化することで、StorageGRIDに対するクロスサイトリクエストフォージェリ（CSRF）攻撃からの保護に役立てることができます。Grid ManagerとTenant Managerはこのセキュリティ機能を自動的に有効にします。その他のAPIクライアントは、サインイン時に有効にするかどうかを選択できます。

別のサイトへのリクエスト（HTTPフォームPOSTなど）をトリガーできる攻撃者は、ログインしているユーザーのCookieを使用して特定のリクエストを実行させることができます。

StorageGRID は、CSRF トークンを使用することで、CSRF 攻撃に対する保護に役立ちます。有効にした場合、特定のクッキーの内容は、特定のヘッダーまたは特定の POST ボディパラメータの内容と一致する必要

があります。

この機能を有効にするには、認証中に `csrfToken`パラメータを `true`に設定します。デフォルトは `false`です。

```
curl -X POST --header "Content-Type: application/json" --header "Accept:
application/json" -d "{
  \"username\": \"MyUserName\",
  \"password\": \"MyPassword\",
  \"cookie\": true,
  \"csrfToken\": true
}" "https://example.com/api/v3/authorize"
```

true の場合、 `GridCsrfToken` クッキーには Grid Manager へのサインイン用のランダムな値が設定され、 `AccountCsrfToken` クッキーには Tenant Manager へのサインイン用のランダムな値が設定されます。

Cookieが存在する場合、システムの状態を変更できるすべてのリクエスト（POST、PUT、PATCH、DELETE）には、以下のいずれかを含める必要があります。

- `X-Csrf-Token`ヘッダー。ヘッダーの値は、CSRFトークンクッキーの値に設定されます。
- フォームエンコードされたボディを受け入れるエンドポイントの場合：A `csrfToken`フォームエンコードされたリクエストボディパラメータ。

CSRF保護を設定するには、"グリッド管理API"または"テナント管理API"を使用します。

> CSRFトークンCookieが設定されているリクエストは、CSRF攻撃に対する追加の保護として、JSONリクエストボディを想定するすべてのリクエストに対して「Content-Type: application/json」ヘッダーを強制的に適用します。

グリッドフェデレーション接続を使用する

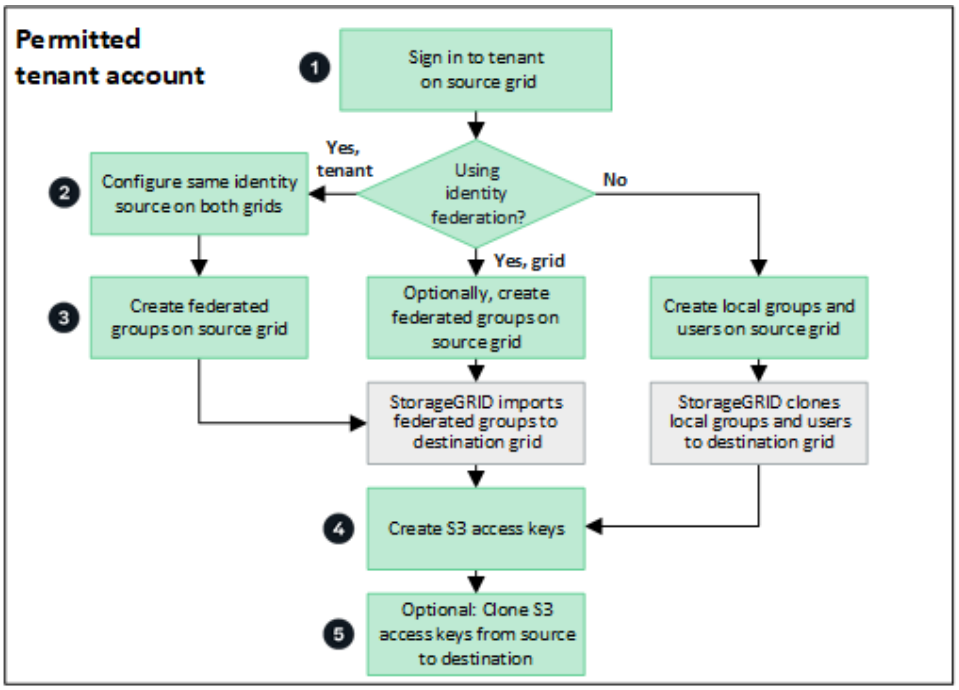
StorageGRID でテナントグループとユーザーをクローニング

テナントがグリッドフェデレーション接続を使用するように作成または編集された場合、そのテナントは1つのStorageGRIDシステム（ソーステナント）から別のStorageGRIDシステム（レプリカテナント）にレプリケートされます。テナントの複製が完了すると、ソーステナントに追加されたグループとユーザーは、レプリカテナントにクローニングされます。

テナントが最初に作成されたStorageGRIDシステムは、テナントの_ソースグリッド_です。テナントが複製されるStorageGRIDシステムは、テナントの_宛先グリッド_です。両方のテナントアカウントは、アカウントID、名前、説明、ストレージクォータ、および割り当てられた権限が同じですが、宛先テナントには初期状態でルートユーザーのパスワードがありません。詳細については、"アカウントクローンとは何ですか？"および"許可されたテナントを管理する"を参照してください。

テナントアカウント情報の複製は、バケットオブジェクトの"グリッド間複製"に必要です。両方のグリッドで同じテナントグループとユーザーを使用することで、どちらのグリッドでも対応するバケットとオブジェクトにアクセスできるようになります。

テナントアカウントに「グリッドフェデレーション接続を使用する」権限が付与されている場合は、ワークフロー図を参照して、グループ、ユーザー、およびS3アクセスキーを複製するために実行する手順を確認してください。



ワークフローの主な手順は以下のとおりです。

1 テナントにサインイン

ソースグリッド（テナントが最初に作成されたグリッド）でテナントアカウントにサインインします。

2 オプションで、IDフェデレーションを設定します

テナントアカウントに、フェデレーショングループとユーザーを使用するための「独自のIDソースを使用する」権限が付与されている場合は、送信元テナントアカウントと宛先テナントアカウントの両方に、同じIDソース（同じ設定）を設定してください。フェデレーションされたグループとユーザーは、両方のグリッドが同じIDソースを使用していない限り、複製できません。手順については、"IDフェデレーションを使用する"を参照してください。

3 グループとユーザーを作成する

グループやユーザーを作成する際は、必ずテナントのソースグリッドから始めてください。新しいグループを追加すると、StorageGRID は自動的に宛先グリッドに複製します。

- ID フェデレーションが StorageGRID システム全体またはテナントアカウントに対して設定されている場合、"新しいテナントグループを作成する" ID ソースからフェデレーテッドグループをインポートすることで実現できます。
- IDフェデレーションを使用していない場合は、"新しいローカルグループを作成する"、次に"ローカルユー

ザーを作成する"。

4 **S3**アクセスキーを作成する

"独自のアクセスキーを作成する"または"別のユーザーのアクセスキーを作成する"を使用して、ソースグリッドまたは宛先グリッドのいずれかにアクセスし、そのグリッド上のバケットにアクセスできます。

5 必要に応じて、**S3**アクセスキーを複製します

両方のグリッドで同じアクセスキーを使用してバケットにアクセスする必要がある場合は、ソースグリッドでアクセスキーを作成し、Tenant Manager APIを使用して手動で宛先グリッドに複製してください。手順については、"APIを使用してS3アクセスキーを複製する"を参照してください。

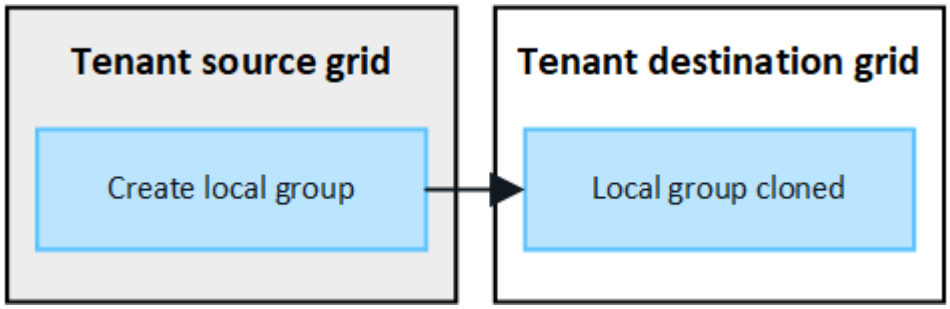
グループ、ユーザー、および**S3**アクセスキーはどのように複製されますか？

このセクションでは、テナントのソースグリッドと宛先グリッド間でグループ、ユーザー、およびS3アクセスキーがどのように複製されるかについて説明します。

ソースグリッド上で作成されたローカルグループはクローンされます

テナントアカウントが作成され、宛先グリッドにレプリケートされた後、StorageGRID はテナントのソースグリッドに追加したローカルグループをテナントの宛先グリッドに自動的にクローニングします。

元のグループとそのクローンは、同じアクセスモード、グループ権限、およびS3グループポリシーを持っています。手順については、"S3テナント用のグループを作成する"を参照してください。

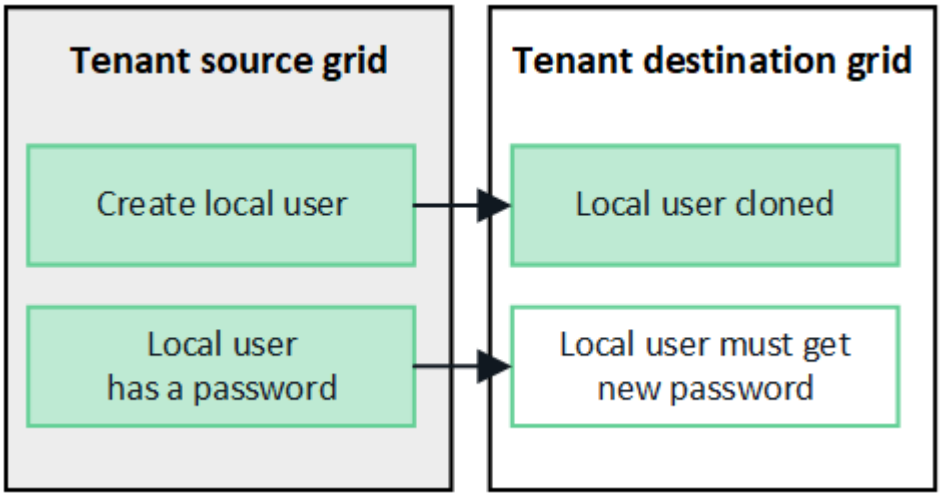


ソースグリッドでローカルグループを作成する際に選択したユーザーは、グループが宛先グリッドに複製される際には含まれません。そのため、グループを作成する際にユーザーを選択しないでください。代わりに、ユーザーを作成する際にグループを選択してください。

ソースグリッドで作成されたローカルユーザーはクローンされます

ソースグリッドで新しいローカルユーザーを作成すると、StorageGRIDはそのユーザーを自動的に宛先グリッドに複製します。元のユーザーとそのクローンは、どちらも同じフルネーム、ユーザー名、および「アクセス拒否」設定を持っています。両ユーザーは同じグループに所属しています。手順については、"ユーザの管理"を参照してください。

セキュリティ上の理由から、ローカルユーザーのパスワードは宛先グリッドに複製されません。ローカルユーザーが宛先グリッド上のテナントマネージャーにアクセスする必要がある場合、テナントアカウントのルートユーザーが、宛先グリッド上でそのユーザーのパスワードを追加する必要があります。手順については、"ユーザの管理"を参照してください。

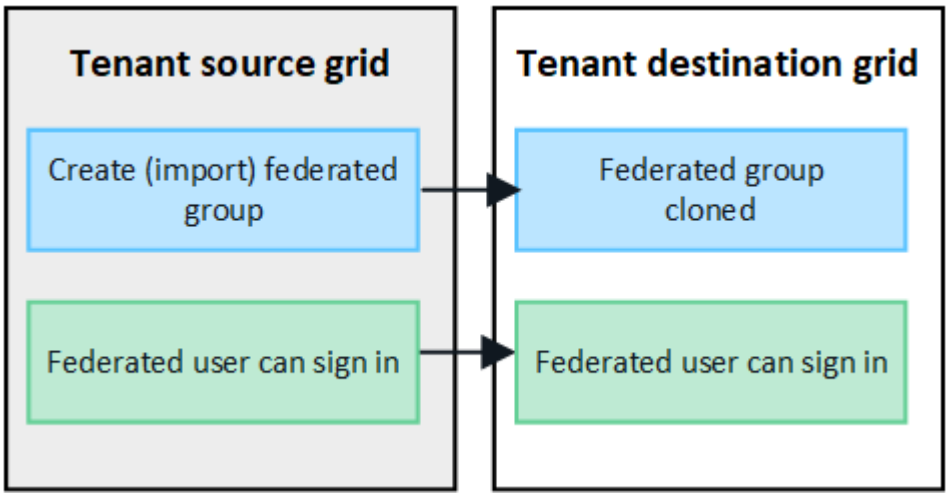


ソースグリッド上で作成されたフェデレーテッドグループはクローンされます

"シングル サインオン"および"アイデンティティフェデレーション"を使用したアカウントクローンの要件が満たされている場合、ソースグリッド上のテナント用に作成（インポート）したフェデレーショングループは、宛先グリッド上のテナントに自動的に複製されます。

両グループとも同じアクセスモード、グループ権限、およびS3グループポリシーを持っています。

ソーステナント用にフェデレーショングループが作成され、宛先テナントに複製されると、フェデレーションユーザーはどちらのグリッドからでもテナントにサインインできるようになります。

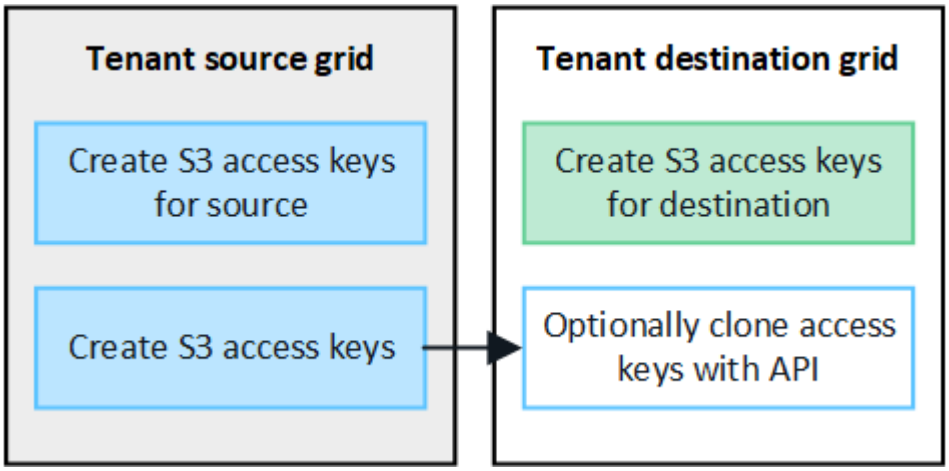


S3アクセスキーは手動でクローンできます

StorageGRID は、グリッドごとに異なるキーを使用することでセキュリティが向上するため、S3 アクセスキーを自動的に複製しません。

2つのグリッドのアクセスキーを管理するには、以下のいずれかの方法を実行できます。

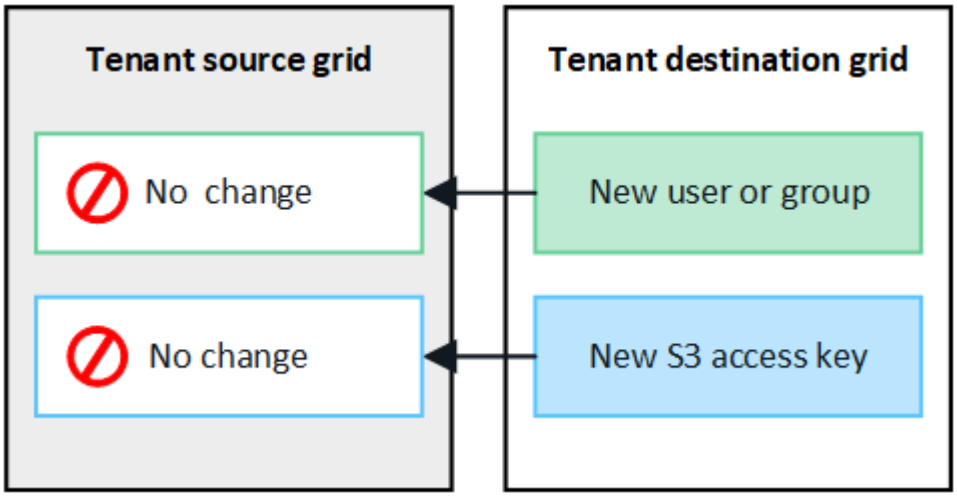
- 各グリッドで同じキーを使用する必要がない場合は、各グリッドで"独自のアクセスキーを作成する"または"別のユーザーのアクセスキーを作成する"することができます。
- 両方のグリッドで同じキーを使用する必要がある場合は、ソースグリッドでキーを作成し、Tenant Manager APIを使用して手動で"キーを複製する"宛先グリッドにインポートできます。



フェデレーションユーザーのS3アクセスキーをクローンすると、ユーザーとS3アクセスキーの両方がクローン先のテナントにクローンされます。

宛先グリッドに追加されたグループとユーザーは複製されません

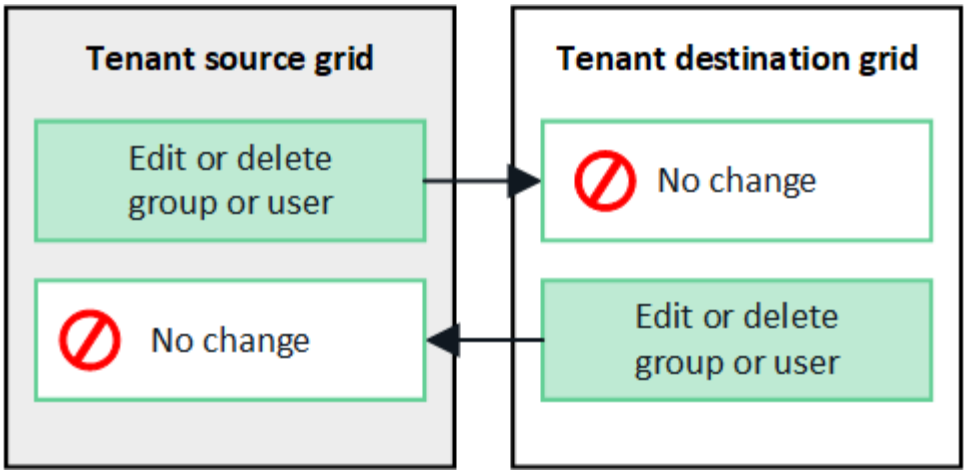
クローン作成は、テナントのソースグリッドからテナントの宛先グリッドへのみ行われます。テナントの宛先グリッドでグループやユーザーを作成またはインポートした場合、StorageGRID はこれらのアイテムをテナントのソースグリッドに複製しません。



編集または削除されたグループ、ユーザー、およびアクセスキーは複製されません

クローン作成は、新しいグループとユーザーを作成した場合にのみ発生します。

いずれかのグリッドでグループ、ユーザー、またはアクセスキーを編集または削除した場合、その変更はもう一方のグリッドには反映されません。



Tenant Management APIを使用して**StorageGRID**グリッド間で**S3**アクセスキーをクローンする

StorageGRIDテナントアカウントに*Use grid federation connection*権限がある場合は、Tenant Management APIを使用して、ソースグリッドのテナントから宛先グリッドのテナントにS3アクセスキーを手動でクローンできます。

開始する前に

- テナントアカウントには「グリッドフェデレーション接続を使用する」権限が付与されています。
- グリッド連携接続の*接続ステータス*は*接続済み*です。
- テナントのソースグリッドでテナントマネージャーに"対応ウェブブラウザ"を使用してサインインしています。
- あなたは、"S3認証情報またはルートアクセス権限を管理する"を持つユーザーグループに属しています。
- ローカルユーザーのアクセスキーを複製する場合、そのユーザーは既に両方のグリッドに存在しています。

> フェデレーションユーザーのS3アクセスキーを複製すると、ユーザーとS3アクセスキーの両方が宛先テナントに追加されます。

自分のアクセスキーを複製する

両方のグリッドで同じバケットにアクセスする必要がある場合は、独自のアクセスキーを複製できます。

手順

1. ソースグリッドのテナントマネージャーを使用して、"独自のアクセスキーを作成する"、 `.csv`ファイルをダウンロードします。
2. テナントマネージャーの上部にあるヘルプアイコンを選択し、*APIドキュメント*を選択します。
3. **s3** セクションで、次のエンドポイントを選択します。

```
POST /org/users/current-user/replicate-s3-access-key
```

4. 「Try it out」を選択します。
5. 本文*テキストボックスで、**accessKey** と **secretAccessKey** のサンプルエントリを、ダウンロードした **.csv** ファイルの値に置き換えます。

   各文字列を囲む二重引用符は必ず残してください。

6. キーの有効期限が切れる場合は、**expires** の例のエントリを、ISO 8601 日付時刻形式の文字列として有効期限の日時（例： `2024-02-28T22:46:33-08:00`）に置き換えてください。キーの有効期限が切れない場合は、**expires** エントリの値として **null** を入力してください（または、**Expires** の行とその前のカンマを削除してください）。
7. 「実行」を選択します。
8. サーバー応答コードが **204** であることを確認してください。これは、キーが宛先グリッドに正常に複製されたことを示しています。

他のユーザーのアクセスキーを複製する

他のユーザーが両方のグリッド上の同じバケットにアクセスする必要がある場合は、そのユーザーのアクセスキーを複製することができます。

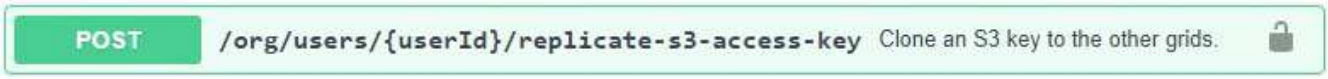
手順

1. ソースグリッドのテナントマネージャーを使用して、"他のユーザーのS3アクセスキーを作成する"、`.csv`ファイルをダウンロードします。
2. テナントマネージャーの上部にあるヘルプアイコンを選択し、*APIドキュメント*を選択します。
3. ユーザーIDを取得します。他のユーザーのアクセスキーを複製するには、この値が必要になります。
   a. **users** セクションから、次のエンドポイントを選択してください。

      `GET /org/users`

   b. 「Try it out」を選択します。
   c. ユーザー検索時に使用したいパラメータを指定してください。
   d. 「実行」を選択します。
   e. クローンしたいキーを持つユーザーを見つけて、**id** フィールドの番号をコピーしてください。
4. **s3** セクションで、次のエンドポイントを選択します。

   `POST /org/users/{userId}/replicate-s3-access-key`



5. 「Try it out」を選択します。
6. **userId** テキストボックスに、コピーしたユーザーIDを貼り付けてください。
7. 本文*テキストボックスで、*アクセスキーの例*と*シークレットアクセスキー*のサンプルエントリを、該当ユーザーの.csv*ファイルの値に置き換えてください。

   文字列を囲む二重引用符は必ず残してください。

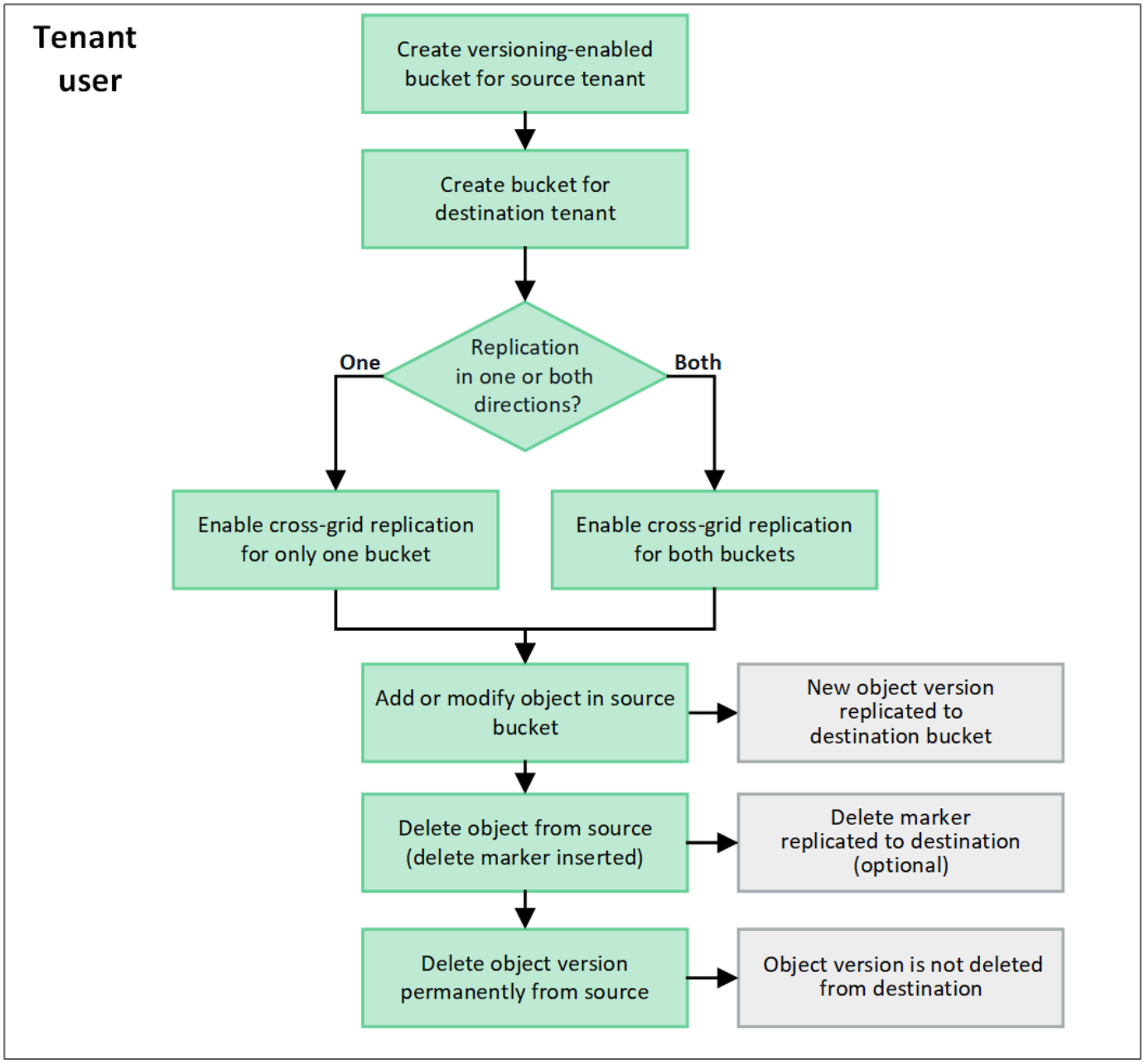
8. キーの有効期限が切れる場合は、**expires** の例のエントリを、ISO 8601 日付時刻形式の文字列として有効期限の日時に置き換えてください（例： `2023-02-28T22:46:33-08:00`）。キーの有効期限が切れない場合は、**expires** エントリの値として **null** を入力してください（または **Expires** の行とその前のカンマを削除してください）。
9. 「実行」を選択します。
10. サーバー応答コードが **204** であることを確認してください。これは、キーが宛先グリッドに正常に複製されたことを示しています。

StorageGRID でクロスグリッドレプリケーションを管理する

テナントアカウントの作成時に「グリッドフェデレーション接続を使用する」権限が割り当てられている場合、クロスグリッドレプリケーションを使用して、テナントのソースグリッド上のバケットとテナントの宛先グリッド上のバケット間でオブジェクトを自動的にレプリケートできます。クロスグリッドレプリケーションは、一方向または両方向で実行できます。

クロスグリッドレプリケーションのワークフロー

ワークフロー図は、2つのグリッド上のバケット間でクロスグリッドレプリケーションを構成するために実行する手順をまとめたものです。これらの手順については、図の下でさらに詳しく説明します。



クロスグリッドレプリケーションの設定

クロスグリッドレプリケーションを使用するには、まず各グリッド上の対応するテナントアカウントにサインインし、2つのバケットを作成する必要があります。その後、どちらか一方または両方のバケットでクロスグリッドレプリケーションを有効にできます。

開始する前に

- クロスグリッドレプリケーションの要件を確認しました。"クロスグリッドレプリケーションとは何ですか？"を参照してください。
- "対応ウェブブラウザ"を使用しています。
- テナントアカウントには「グリッドフェデレーション接続を使用する」権限が付与されており、両方のグリッドに同一のテナントアカウントが存在します。"グリッドフェデレーション接続の許可テナントを管理する"を参照してください。
- サインインしているテナントユーザーはすでに両方のグリッドに存在し、"ルートアクセス権限" を持つユ

ーザーグループに属しています。

- テナントの宛先グリッドにローカルユーザーとしてサインインする場合、テナントアカウントのルートユーザーが、そのグリッド上のユーザーアカウントにパスワードを設定しています。

2つのバケットを作成する

まず、各グリッド上の対応するテナントアカウントにサインインし、各グリッド上にバケットを作成します。

手順

1. グリッドフェデレーション接続のいずれかのグリッドから開始して、新しいバケットを作成します。
   a. 両方のグリッドに存在するテナントユーザーの認証情報を使用して、テナントアカウントにサインインしてください。

      テナントの宛先グリッドにローカルユーザーとしてサインインできない場合は、テナントアカウントのルートユーザーがあなたのユーザーアカウントにパスワードを設定していることを確認してください。

   b. 手順に従って"S3バケットを作成する"。

      バケット名とリージョンは、グリッドごとに異なる場合があります。

   c. 「オブジェクト設定の管理」タブで、「オブジェクトのバージョン管理を有効にする」を選択します。
   d. StorageGRID システムで S3 オブジェクトロックが有効になっている場合は、"S3オブジェクトロックを使用したクロスグリッドレプリケーション"を参照してください。
   e. 「バケットの作成」を選択します。
   f. **Finish** を選択します。
2. グリッドフェデレーション接続内の別のグリッドに、同じテナントアカウント用のバケットを作成するには、これらの手順を繰り返します。

   必要に応じて、各バケットは異なるリージョンを使用できます。

グリッド間レプリケーションを有効にする

どちらのバケットにもオブジェクトを追加する前に、これらの手順を実行する必要があります。

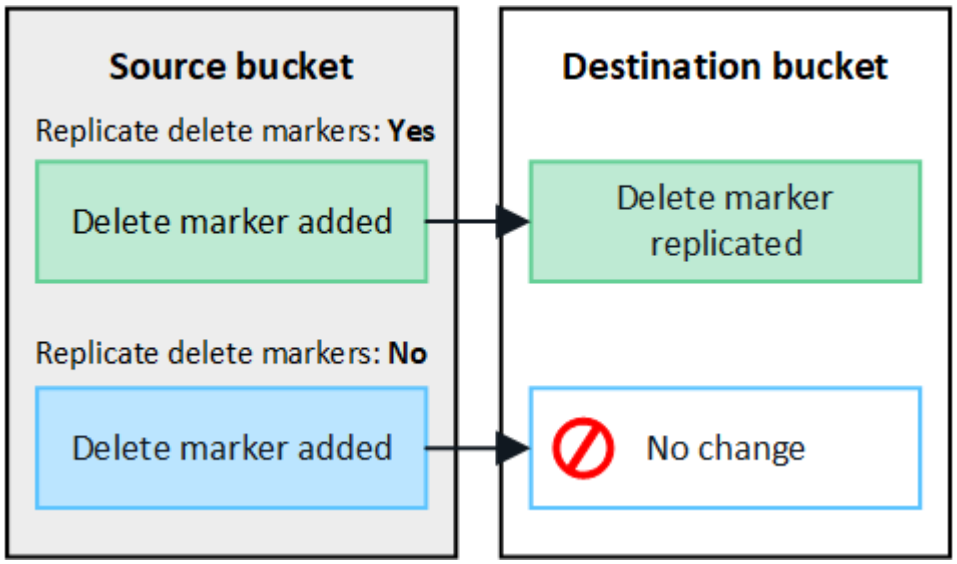
手順

1. 複製したいオブジェクトを含むグリッドから開始して、"一方向のクロスグリッド複製"を有効にします：
   a. バケットのテナントアカウントにサインインしてください。
   b. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3)** > **Buckets** を選択してください。
   c. バケットの詳細ページにアクセスするには、表からバケット名を選択してください。
   d. 「クロスグリッドレプリケーション」タブを選択します。
   e. **Enable** を選択し、要件リストを確認してください。

f. すべての要件を満たしている場合は、使用するグリッド連携接続を選択してください。

g. 必要に応じて、*削除マーカーの複製*の設定を変更して、S3クライアントがバージョンIDを含まない削除リクエストをソースグリッドに発行した場合に、宛先グリッドで何が起こるかを決定します。

- はい（デフォルト）：削除マーカーがソースバケットに追加され、宛先バケットに複製されます。
- いいえ: 削除マーカーはソースバケットに追加されますが、宛先バケットには複製されません。



削除リクエストにバージョンIDが含まれている場合、そのオブジェクトバージョンはソースバケットから完全に削除されます。StorageGRID はバージョンIDを含む削除リクエストをレプリケートしないため、同じオブジェクトバージョンはデスティネーションから削除されません。

詳細については、"クロスグリッドレプリケーションとは何ですか？"を参照してください。

a. 必要に応じて、監査メッセージの量を管理するために、*クロスグリッドレプリケーション*監査カテゴリの設定を変更します。

- エラー（デフォルト）：監査出力には、失敗したクロスグリッドレプリケーション要求のみが含まれます。
- **Normal**: すべてのクロスグリッドレプリケーション要求が含まれるため、監査出力の量が大幅に増加します。

b. 選択内容を確認してください。両方のバケットが空でない限り、これらの設定を変更することはできません。

c. 「Enable and test」を選択します。

しばらくすると、成功メッセージが表示されます。このバケットに追加されたオブジェクトは、自動的に他のグリッドにも複製されるようになりました。バケットの詳細ページでは、*クロスグリッドレプリケーション*が有効になっている機能として表示されます。

2. 必要に応じて、もう一方のグリッドの対応するバケットに移動して"グリッド間レプリケーションを双方向で有効にする"。

グリッド間のレプリケーションをテストする

バケットに対してクロスグリッドレプリケーションが有効になっている場合は、接続とクロスグリッドレプリケーションが正しく機能していること、およびソースバケットと宛先バケットがすべての要件を満たしていること（たとえば、バージョン管理が有効になっていること）を確認する必要がある場合があります。

開始する前に

- "対応ウェブブラウザ"を使用しています。
- あなたは、"ルートアクセス権限"に属するユーザーグループのメンバーです。

手順

1. バケットのテナントアカウントにサインインしてください。
2. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3) > Buckets** を選択してください。
3. バケットの詳細ページにアクセスするには、表からバケット名を選択してください。
4. 「クロスグリッドレプリケーション」タブを選択します。
5. 「接続テスト」を選択します。

   接続が正常であれば、成功を示すバナーが表示されます。そうでない場合は、エラーメッセージが表示されます。そのメッセージを使用して、お客様とグリッド管理者が問題を解決できます。詳細については、"グリッドフェデレーションのエラーをトラブルシューティングする"を参照してください。

6. クロスグリッドレプリケーションが双方向で行われるように構成されている場合は、もう一方のグリッド上の対応するバケットに移動し、*接続テスト*を選択して、クロスグリッドレプリケーションが逆方向にも機能していることを確認してください。

グリッド間レプリケーションを無効にする

オブジェクトを他のグリッドにコピーする必要がなくなった場合は、クロスグリッドレプリケーションを完全に停止できます。

グリッド間レプリケーションを無効にする前に、以下の点に注意してください。

- グリッド間レプリケーションを無効にしても、既にグリッド間でコピーされたオブジェクトは削除されません。例えば、Grid 1 の `my-bucket`にあるオブジェクトが Grid 2 の `my-bucket`にコピーされている場合、そのバケットのグリッド間レプリケーションを無効にしても削除されません。これらのオブジェクトを削除するには、手動で削除する必要があります。
- 各バケットでクロスグリッドレプリケーションが有効になっている場合（つまり、レプリケーションが双方向で行われる場合）、いずれか一方または両方のバケットのクロスグリッドレプリケーションを無効にすることができます。たとえば、Grid 1 上の `my-bucket`から Grid 2 上の `my-bucket`へのオブジェクトのレプリケーションを無効にしながら、Grid 2 上の `my-bucket`から Grid 1 上の `my-bucket`へのオブジェクトのレプリケーションは継続することができます。
- テナントのグリッドフェデレーション接続の使用権限を削除するには、まずグリッド間レプリケーションを無効にする必要があります。"許可されたテナントを管理する"を参照してください。
- オブジェクトを含むバケットのクロスグリッドレプリケーションを無効にした場合、ソースバケットと宛先バケットの両方からすべてのオブジェクトを削除しない限り、クロスグリッドレプリケーションを再度有効にすることはできません。

両方のバケットが空でない限り、レプリケーションを再度有効にすることはできません。

開始する前に

- "対応ウェブブラウザ"を使用しています。
- あなたは、"ルートアクセス権限"に属するユーザーグループのメンバーです。

手順

1. 複製したくないオブジェクトを含むグリッドから始めて、バケットのクロスグリッド複製を停止します。
   a. バケットのテナントアカウントにサインインしてください。
   b. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3)** > **Buckets** を選択してください。
   c. バケットの詳細ページにアクセスするには、表からバケット名を選択してください。
   d. 「クロスグリッドレプリケーション」タブを選択します。
   e. 「レプリケーションを無効にする」を選択します。
   f. このバケットのクロスグリッドレプリケーションを無効にしてよい場合は、テキストボックスに **Yes** と入力し、**Disable** を選択してください。

      しばらくすると、成功メッセージが表示されます。このバケットに追加された新しいオブジェクトは、他のグリッドに自動的に複製されなくなります。*クロスグリッドレプリケーション*は、バケットページで有効な機能として表示されなくなります。
2. クロスグリッドレプリケーションが双方向で実行されるように設定されている場合、もう一方のグリッド上の対応するバケットに移動し、逆方向のクロスグリッドレプリケーションを停止します。

StorageGRID でグリッドフェデレーション接続を表示する

テナントアカウントに「グリッドフェデレーション接続を使用する」権限が付与されている場合、許可されている接続を表示できます。

開始する前に

- テナントアカウントには「グリッドフェデレーション接続を使用する」権限が付与されています。
- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"ルートアクセス権限"に属するユーザーグループのメンバーです。

手順

1. **STORAGE (S3)** > **Grid federation connections** を選択します。

   グリッド連携接続ページが表示され、以下の情報をまとめた表が含まれています：

列	説明
接続名	このテナントが使用許可を得ているグリッド連携接続。

列	説明
クロスグリッドレプリケーションを備えたバケット	各グリッドフェデレーション接続について、クロスグリッドレプリケーションが有効になっているテナントバケット。これらのバケットに追加されたオブジェクトは、接続先の他のグリッドにも複製されます。
最後のエラー	各グリッド連携接続について、データが他のグリッドに複製されている際に発生した最新のエラー（もしあれば）。最後のエラーをクリアするを参照してください。

2. 必要に応じて、バケット名を選択して"バケットの詳細を表示"。

最後のエラーをクリアする

最後のエラー 列にエラーが表示される理由は、以下のいずれかです。

- ソースオブジェクトのバージョンが見つかりませんでした。
- ソースバケットが見つかりませんでした。
- 宛先バケットが削除されました。
- 宛先バケットは別のアカウントによって再作成されました。
- 宛先バケットではバージョン管理が一時停止されています。
- 宛先バケットは同じアカウントによって再作成されましたが、現在はバージョン管理されていません。

この列には、最後に発生したクロスグリッド複製エラーのみが表示されます。それ以前に発生した可能性のあるエラーは表示されません。

手順

1. *最後のエラー*列にメッセージが表示された場合は、メッセージ本文を確認してください。

   例えば、このエラーは、クロスグリッドレプリケーションの宛先バケットが無効な状態であったことを示しています。これは、バージョン管理が一時停止されたか、S3 Object Lockが有効になっていたことが原因である可能性があります。

2. 推奨された措置をすべて実行してください。例えば、クロスグリッドレプリケーションの宛先バケットで

バージョン管理が一時停止されていた場合は、そのバケットのバージョン管理を再度有効にします。

3. 表から接続を選択してください。
4. 「エラーをクリア」を選択します。
5. 「はい」を選択すると、メッセージが消去され、システムのステータスが更新されます。
6. 5～6分待ってから、バケットに新しいオブジェクトを取り込んでください。エラーメッセージが再表示されないことを確認してください。

> エラーメッセージが確実に消去されるようにするには、メッセージ内のタイムスタンプから少なくとも5分経過してから新しいオブジェクトを取り込んでください。

7. バケットエラーが原因でオブジェクトの複製に失敗したかどうかを確認するには、"失敗したレプリケーション操作を特定して再試行する"を参照してください。

グループとユーザーを管理する

StorageGRIDでIDフェデレーションを使用する

IDフェデレーションを利用することで、テナントグループとユーザーの設定が迅速になり、テナントユーザーは使い慣れた認証情報を使用してテナントアカウントにサインインできるようになります。

テナントマネージャーのIDフェデレーションを構成する

テナントグループとユーザーをActive Directory、Microsoft Entra ID、OpenLDAP、Oracle Directory Serverなどの別のシステムで管理したい場合は、テナントマネージャーのIDフェデレーションを設定できます。

開始する前に

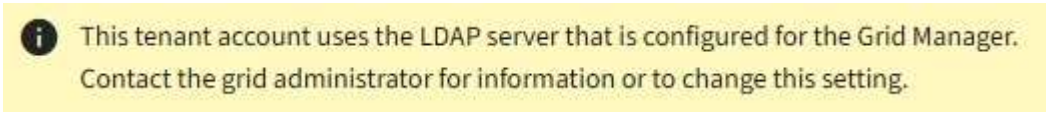
- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"ルートアクセス権限"に属するユーザーグループのメンバーです。
- お客様は、IDプロバイダーとしてActive Directory、Microsoft Entra ID、OpenLDAP、またはOracle Directory Serverを使用しています。

> リストに記載されていないLDAP v3サービスを利用したい場合は、テクニカルサポートにお問い合わせください。

- OpenLDAPを使用する場合は、OpenLDAPサーバを設定する必要があります。OpenLDAPサーバの設定に関するガイドラインを参照してください。
- LDAPサーバとの通信にTransport Layer Security（TLS）を使用する予定がある場合、アイデンティティプロバイダーはTLS 1.2または1.3を使用している必要があります。"送信TLS接続でサポートされている暗号スイート"を参照してください。

タスク概要

テナント向けにIDフェデレーションサービスを構成できるかどうかは、テナントアカウントの設定方法によって異なります。テナントによっては、Grid Manager用に構成されたIDフェデレーションサービスを共有している場合があります。IDフェデレーションページにアクセスした際にこのメッセージが表示される場合、このテナントに対して個別のフェデレーションIDソースを設定することはできません。

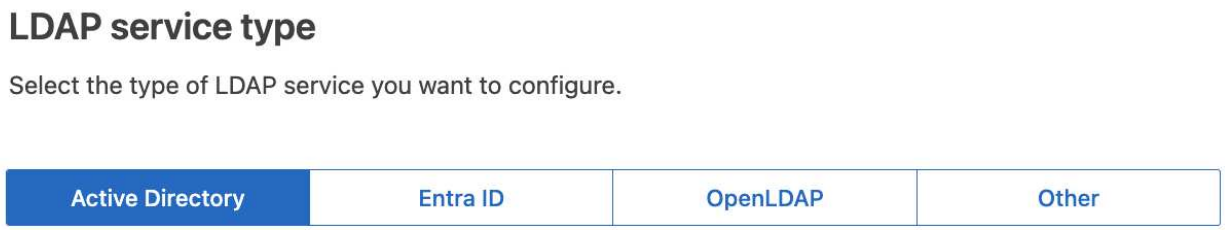


設定を入力

アイデンティティフェデレーションを設定する際には、StorageGRID が LDAPサービスに接続するために必要な値を指定します。

手順

1. アクセス管理 ＞ **ID**フェデレーション を選択します。
2. *ID連携を有効にする*を選択します。
3. LDAPサービスタイプのセクションで、設定するLDAPサービスのタイプを選択します。



「その他」を選択して、Oracle Directory Serverを使用するLDAPサーバの値を設定します。

4. 「その他」を選択した場合は、LDAP属性セクションの各項目を入力してください。そうでない場合は、次のステップに進んでください。
   - ユーザー固有名：LDAPユーザーの固有識別子を含む属性の名前。この属性は `sAMAccountName`（Active Directory 用）および `uid`（OpenLDAP用）と同等です。Oracle Directory Server を設定する場合は、 `uid`を入力してください。
   - ユーザー**UUID**：LDAPユーザーの永続的な一意の識別子を含む属性の名前。この属性は、Active Directory の場合は `objectGUID`、OpenLDAP の場合は `entryUUID`と同等です。Oracle Directory Server を設定する場合は、 `nsuniqueid`を入力してください。指定した属性に対する各ユーザーの値は、16バイト形式または文字列形式の32桁の16進数である必要があり、ハイフンは無視されます。
   - グループ固有名：LDAPグループの固有識別子を含む属性の名前。この属性は、Active Directory の場合は `sAMAccountName`、OpenLDAP の場合は `cn`と同等です。Oracle Directory Server を設定する場合は、 `cn`を入力してください。
   - グループ**UUID**：LDAPグループの永続的な一意の識別子を含む属性の名前。この属性は `objectGUID`（Active Directory 用）および `entryUUID`（OpenLDAP用）と同等です。Oracle Directory Server を構成する場合は、 `nsuniqueid`を入力してください。各グループの指定属性の値は、16バイト形式または文字列形式の32桁の16進数でなければならず、ハイフンは無視されます。
5. すべての LDAP サービスタイプについて、「LDAPサーバの設定」セクションに必要な LDAPサーバとネットワーク接続情報を入力してください。
   - ホスト名：LDAPサーバの完全修飾ドメイン名（FQDN）またはIPアドレス。
   - ポート：LDAPサーバへの接続に使用するポート。

> STARTTLSのデフォルトポートは389番、LDAPSのデフォルトポートは636番です。ただし、ファイアウォールが正しく設定されていれば、どのポートでも使用できます。

- ユーザー名：LDAPサーバに接続するユーザーの識別名（DN）の完全パス。

  Active Directoryの場合、下位レベルのログオン名またはユーザープリンシパル名を指定することもできます。

  指定されたユーザーは、グループとユーザーの一覧表示、および以下の属性へのアクセス権限を持っている必要があります：

  - `sAMAccountName` または `uid`
  - `objectGUID`、 `entryUUID`、または `nsuniqueid`
  - `cn`
  - `memberOf` または `isMemberOf`
  - **Active Directory**： `objectSid`、 `primaryGroupID`、 `userAccountControl`、および `userPrincipalName`
  - **Entra ID**： `accountEnabled`および `userPrincipalName`

- パスワード：ユーザー名に関連付けられたパスワード。

> 今後パスワードを変更する場合は、このページで更新する必要があります。

- グループベース**DN**：グループを検索するLDAPサブツリーの識別名（DN）の完全パス。下記の Active Directory の例では、識別名がベース DN （DC=storagegrid,DC=example,DC=com） に対して相対的なすべてのグループをフェデレーション グループとして使用できます。

> *グループ固有名*の値は、所属する*グループベースDN*内で一意である必要があります。

- ユーザーベース**DN**：ユーザーを検索するLDAPサブツリーの識別名（DN）の完全パス。

> *ユーザー固有名*の値は、所属する*ユーザーベース DN*内で一意である必要があります。

- バインドユーザー名フォーマット（オプション）：パターンを自動的に判別できない場合に StorageGRID が使用するデフォルトのユーザー名パターンです。

  *バインドユーザー名形式*を指定することを推奨します。これにより、StorageGRIDがサービスアカウントとバインドできない場合でも、ユーザーがサインインできるようになります。

  以下のいずれかのパターンを入力してください：

  - **UserPrincipalName**パターン（**AD** および **Entra ID**）： `[USERNAME]@*example*.com`
  - 下位レベルのログオン名パターン（**AD**および**Entra ID**）： `*example*\[USERNAME]`
  - 識別名パターン： `CN=[USERNAME],CN=Users,DC=*example*,DC=com`

[USERNAME] を記載どおりに正確に含めてください。

6. トランスポート層セキュリティ（TLS）セクションで、セキュリティ設定を選択します。
   - **STARTTLS**を使用する：STARTTLSを使用してLDAPサーバとの通信を保護します。これはActive Directory、OpenLDAP、またはその他のシステムでは推奨されるオプションですが、Microsoft Entra IDではサポートされていません。
   - **LDAPS**を使用する：LDAPS（LDAP over SSL）オプションはTLSを使用してLDAPサーバへの接続を確立します。Microsoft Entra IDの場合は、このオプションを選択する必要があります。
   - **TLS**を使用しない：StorageGRIDシステムとLDAPサーバ間のネットワーク トラフィックはセキュリティで保護されません。このオプションは、Microsoft Entra IDではサポートされていません。

   > Active Directory サーバーがLDAP署名を強制している場合、「TLSを使用しない」オプションはサポートされていません。STARTTLSまたはLDAPSを使用する必要があります。

7. STARTTLSまたはLDAPSを選択した場合は、接続を保護するために使用する証明書を選択してください。
   - オペレーティングシステムの**CA**証明書を使用する：オペレーティングシステムにインストールされているデフォルトのGrid CA証明書を使用して接続を保護します。
   - カスタム**CA**証明書を使用する：カスタムセキュリティ証明書を使用します。

     この設定を選択した場合は、カスタムセキュリティ証明書をコピーして、CA証明書のテキストボックスに貼り付けてください。

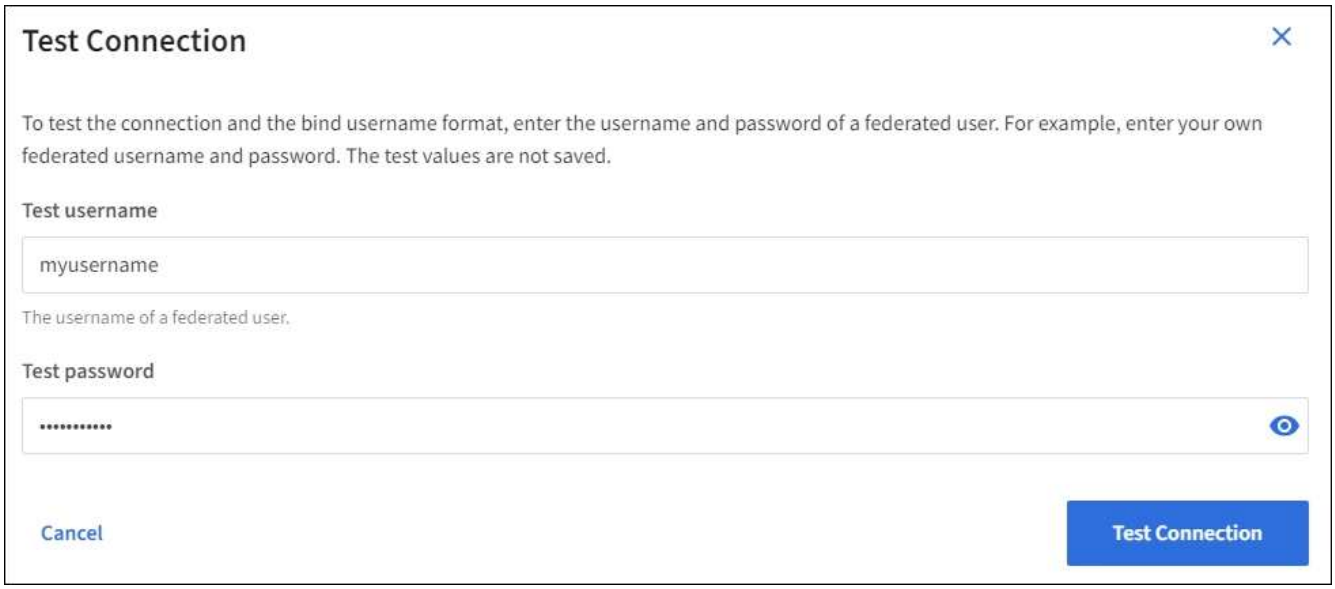
接続をテストして設定を保存する

すべての値を入力した後、設定を保存する前に接続テストを行う必要があります。StorageGRID は、LDAPサーバの接続設定と、指定した場合はバインドユーザー名の形式を確認します。

手順

1. 「接続テスト」を選択します。
2. バインドユーザー名の形式を指定しなかった場合：
   - 接続設定が有効な場合、「接続テスト成功」というメッセージが表示されます。設定を保存するには、*保存*を選択してください。
   - 接続設定が無効な場合、「テスト接続を確立できませんでした」というメッセージが表示されます。「閉じる」を選択します。その後、問題をすべて解決し、再度接続テストを行ってください。
3. バインドユーザー名の形式を指定した場合は、有効なフェデレーションユーザーのユーザー名とパスワードを入力してください。

   例えば、ご自身のユーザー名とパスワードを入力してください。ユーザー名には、@ や / などの特殊文字を含めないでください。



- 接続設定が有効な場合、「接続テスト成功」というメッセージが表示されます。設定を保存するには、*保存*を選択してください。
- 接続設定、バインドユーザー名の形式、またはテストユーザー名とパスワードが無効な場合、エラーメッセージが表示されます。問題を解決し、再度接続テストを行ってください。

IDソースとの強制同期

StorageGRID システムは、IDソースから連携グループとユーザーを定期的に同期します。ユーザー権限をできるだけ早く有効化または制限したい場合は、同期を強制的に開始することができます。

手順

1. ID連携ページに移動します。
2. ページ上部の「Sync server」を選択してください。

   同期処理には、お使いの環境によっては時間がかかる場合があります。

「IDフェデレーション同期失敗」アラートは、IDソースからのフェデレーショングループとユーザーの同期に問題が発生した場合にトリガーされます。

IDフェデレーションを無効にする

グループおよびユーザーのID連携を、一時的または永続的に無効にすることができます。ID連携が無効になっている場合、StorageGRIDとIDソース間の通信は行われません。ただし、設定した内容はすべて保持されるため、将来的にID連携を簡単に再度有効にすることができます。

タスク概要

IDフェデレーションを無効にする前に、以下の点に注意してください。

- フェデレーションユーザーはサインインできません。
- 現在サインインしているフェデレーションユーザーは、セッションが期限切れになるまでStorageGRIDシステムへのアクセスを維持できますが、セッションが期限切れになるとサインインできなくなります。
- StorageGRID システムとIDソース間の同期は行われず、同期されていないアカウントに関するアラートも

発生しません。

- 「ID フェデレーションを有効にする」チェックボックスは、シングル サインオン（SSO）ステータスが「有効」または「サンドボックスモード」の場合は無効になります。ID フェデレーションを無効にするには、まずシングル サインオンページの SSO ステータスを「無効」に設定する必要があります。参照してください"シングル サインオンを無効にする"。

手順

1. ID連携ページに移動します。
2. **ID**連携を有効にする チェックボックスのチェックを外します。

OpenLDAPサーバの設定に関するガイドライン

IDフェデレーションにOpenLDAPサーバーを使用する場合は、OpenLDAPサーバー上で特定の設定を構成する必要があります。

> Active Directory または Microsoft Entra ID 以外の ID ソースの場合、StorageGRID は外部から無効化されたユーザーによる S3 へのアクセスを自動的にブロックしません。S3 へのアクセスをブロックするには、ユーザーの S3 キーをすべて削除するか、ユーザーをすべてのグループから削除してください。

memberof および **refint** オーバーレイ

memberof および refint オーバーレイを有効にする必要があります。詳細については、http://www.openldap.org/doc/admin24/index.html["OpenLDAP ドキュメント：バージョン 2.4 管理者ガイド"^] のリバースグループメンバーシップのメンテナンスに関する手順を参照してください。

インデックス作成

以下の OpenLDAP 属性を、指定されたインデックスキーワードで設定する必要があります：

- `olcDbIndex: objectClass eq`
- `olcDbIndex: uid eq,pres,sub`
- `olcDbIndex: cn eq,pres,sub`
- `olcDbIndex: entryUUID eq`

さらに、最適なパフォーマンスを得るために、ユーザー名に関するヘルプに記載されているフィールドにインデックスが設定されていることを確認してください。

逆グループメンバーシップの維持に関する情報については、http://www.openldap.org/doc/admin24/index.html["OpenLDAP ドキュメント：バージョン 2.4 管理者ガイド"^]を参照してください。

テナントグループを管理する

StorageGRIDでS3テナントのグループを作成する

S3ユーザーグループの権限は、フェデレーショングループをインポートするか、ローカルグループを作成することで管理できます。

開始する前に

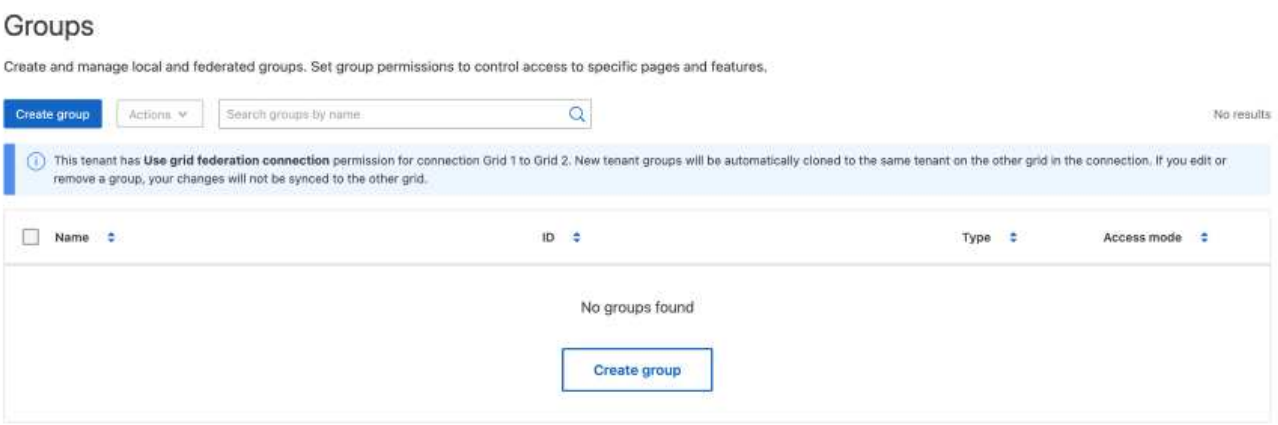
- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"ルートアクセス権限"に属するユーザーグループのメンバーです。
- フェデレーテッドグループをインポートする場合は、"設定されたIDフェデレーション"、フェデレーテッドグループがすでに設定済みのIDソースに存在している必要があります。
- テナントアカウントに グリッド フェデレーション接続の使用 権限があり、"テナントグループとユーザーのクローン作成"のワークフローと考慮事項を確認済みで、テナントのソースグリッドにサインインしています。

グループ作成ウィザードにアクセスする

まず最初に、グループ作成ウィザードにアクセスしてください。

手順

1. アクセス管理 > グループ を選択します。
2. テナントアカウントに「グリッドフェデレーション接続を使用する」権限が付与されている場合は、青いバナーが表示されることを確認してください。このバナーは、このグリッドで作成された新しいグループが、接続先の別のグリッド上の同じテナントに複製されることを示しています。このバナーが表示されない場合は、テナントの宛先グリッドにサインインしている可能性があります。



3. 「グループの作成」を選択します。

グループの種類を選択してください

ローカルグループを作成するか、フェデレーショングループをインポートすることができます。

手順

1. ローカルグループを作成するには、*ローカルグループ*タブを選択し、以前に設定したIDソースからグループをインポートするには、*フェデレーショングループ*タブを選択します。

   StorageGRID システムでシングル サインオン（SSO）が有効になっている場合、ローカルグループに属するユーザーはテナントマネージャーにサインインできませんが、グループの権限に基づいてクライアントアプリケーションを使用してテナントのリソースを管理することは可能です。

2. グループ名を入力してください。

- ローカルグループ：表示名と一意の名前の両方を入力してください。表示名は後で編集できます。

> テナントアカウントに **Use grid federation connection** 権限が付与されている場合、宛先グリッド上に同じ **Unique name** が既に存在すると、クローン作成エラーが発生します。

- フェデレーテッドグループ：一意の名前を入力してください。Active Directory の場合、一意の名前は `sAMAccountName`属性に関連付けられた名前です。OpenLDAP の場合、一意の名前は `uid`属性に関連付けられた名前です。

3. *続行*を選択します。

グループ権限の管理

グループ権限は、テナントマネージャーおよびテナント管理APIにおいて、ユーザーが実行できるタスクを制御します。

手順

1. *アクセスモード*については、以下のいずれかを選択してください。
   - 読み書き可能（デフォルト）：ユーザーはテナントマネージャーにサインインして、テナント構成を管理できます。
   - 読み取り専用：ユーザーは設定と機能のみを表示できます。Tenant Manager またはテナント管理 API で変更を加えたり、操作を実行したりすることはできません。ローカルの読み取り専用ユーザーは、自分のパスワードを変更できます。

> ユーザーが複数のグループに所属していて、いずれかのグループが読み取り専用に設定されている場合、ユーザーは選択されたすべての設定と機能に対して読み取り専用アクセス権を持つことになります。

2. このグループに対して、1つ以上の権限を選択してください。

   "テナント管理権限"を参照してください。

3. *続行*を選択します。

S3グループポリシーを設定する

グループポリシーによって、ユーザーが持つS3アクセス権限が決まります。

手順

1. このグループに適用するポリシーを選択してください。

グループポリシー	説明
S3へのアクセスなし	デフォルト。このグループのユーザーは、バケットポリシーによって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。

グループポリシー	説明
読み取り専用アクセス	このグループのユーザーは、S3リソースへの読み取り専用アクセス権を持ちます。例えば、このグループのユーザーは、オブジェクトの一覧表示や、オブジェクトのデータ、メタデータ、タグの読み取りを行うことができます。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は、バケットを含むS3リソースへの完全なアクセス権を持ちます。このオプションを選択すると、フルアクセス権限を持つ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。
ランサムウェア対策	このポリシー例は、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操作を実行できますが、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。 *すべてのバケットを管理する*権限を持つ Tenant Manager ユーザーは、このグループポリシーを上書きできます。「すべてのバケットを管理する」権限は信頼できるユーザーのみに制限し、利用可能な場合は Multi-Factor Authentication （MFA）を使用してください。
Custom	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。

2. **Custom** を選択した場合は、グループポリシーを入力してください。各グループポリシーのサイズ制限は 5,120 バイトです。有効な JSON 形式の文字列を入力してください。

   グループポリシーに関する詳細情報（言語構文や例を含む）については、"グループポリシーの例"を参照してください。

3. ローカルグループを作成する場合は、「続行」を選択します。フェデレーテッドグループを作成する場合は、「グループの作成」を選択し、「完了」を選択します。

ユーザーを追加する（ローカルグループのみ）

ユーザーを追加せずにグループを保存することもできますし、既存のローカルユーザーを任意で追加することもできます。

> テナントアカウントに「グリッドフェデレーション接続を使用する」権限が付与されている場合、ソースグリッドでローカルグループを作成する際に選択したユーザーは、グループが宛先グリッドに複製される際に含まれません。そのため、グループを作成する際にユーザーを選択しないでください。代わりに、ユーザーを作成する際にグループを選択してください。

手順

1. 必要に応じて、このグループに1人以上のローカルユーザーを選択してください。
2. 「グループの作成」と「完了」を選択します。

作成したグループがグループ一覧に表示されます。

テナントアカウントに「グリッドフェデレーション接続を使用する」権限があり、かつテナントのソースグリッド上にいる場合、新しいグループはテナントの宛先グリッドに複製されます。グループの詳細ページの概要セクションの「クローン作成ステータス」に「成功」と表示されます。

StorageGRID テナント管理権限

テナントグループを作成する前に、そのグループにどのような権限を割り当てるかを検討してください。テナント管理権限は、ユーザーがテナントマネージャーまたはテナント管理APIを使用して実行できるタスクを決定します。ユーザーは1つまたは複数のグループに所属できます。ユーザーが複数のグループに所属している場合、権限は累積されます。

テナントマネージャーにサインインしたり、テナント管理APIを使用したりするには、ユーザーは少なくとも1つの権限を持つグループに所属している必要があります。サインインできるすべてのユーザーは、以下のタスクを実行できます。

- ダッシュボードを表示する
- （ローカルユーザーの場合）各自でパスワードを変更する

すべての権限において、グループのアクセスモード設定によって、ユーザーが設定を変更したり操作を実行したりできるかどうか、または関連する設定や機能を表示することしかできないかどうかが決まります。

ユーザーが複数のグループに所属していて、いずれかのグループが読み取り専用に設定されている場合、ユーザーは選択されたすべての設定と機能に対して読み取り専用アクセス権を持つことになります。

グループには以下の権限を割り当てることができます。

アクセス許可	説明	詳細
ルート アクセス	テナントマネージャーおよびテナント管理APIへのフルアクセスを提供します。	
S3認証情報を自分で管理する	ユーザーが自身のS3アクセスキーを作成および削除できるようにします。	この権限を持たないユーザーには、**STORAGE (S3)** ＞ **My S3 access keys** メニューオプションが表示されません。
すべてのバケットを表示	ユーザーがすべてのバケットとバケット構成を表示できるようにします。	「すべてのバケットを表示」または「すべてのバケットを管理」の権限を持たないユーザーには、**Buckets** メニューオプションが表示されません。 この権限は、「すべてのバケットを管理する」権限に置き換えられます。これは、S3 クライアントまたは S3 Console で使用される S3 バケットまたはグループポリシーには影響しません。

アクセス許可	説明	詳細
すべてのバケットを管理する	ユーザーがTenant ManagerおよびTenant Management APIを使用して、S3バケットの作成と削除、およびテナントアカウント内のすべてのS3バケットの設定を管理できるようにします。これは、S3バケットまたはグループポリシーに関係なく適用されます。	「すべてのバケットを表示」または「すべてのバケットを管理」の権限を持たないユーザーには、**Buckets** メニューオプションが表示されません。 この権限は、「すべてのバケットを表示する」権限に優先します。これは、S3 クライアントまたは S3 Console で使用される S3 バケットまたはグループポリシーには影響しません。
エンドポイントを管理する	ユーザーは、Tenant Manager またはテナント管理 API を使用して、StorageGRID プラットフォームサービスの宛先として使用されるプラットフォームサービスエンドポイントを作成または編集できます。	この権限を持たないユーザーには、**Platform services endpoints** メニューオプションが表示されません。
S3コンソールタブを使用する	「すべてのバケットを表示」または「すべてのバケットを管理」の権限と組み合わせることで、ユーザーはバケットの詳細ページの S3 コンソールタブからオブジェクトを表示および管理できるようになります。	

StorageGRIDテナントグループを管理

必要に応じてテナントグループを管理し、グループの表示、編集、複製などを行うことができます。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"ルートアクセス権限"に属するユーザーグループのメンバーです。

グループの表示または編集

各グループの基本情報と詳細情報を表示および編集できます。

手順

1. アクセス管理 > グループ を選択します。
2. グループページに記載されている情報を確認してください。このページには、このテナントアカウントのすべてのローカルグループおよびフェデレーテッドグループの基本情報が一覧表示されています。

   テナントアカウントに「グリッドフェデレーション接続の使用」権限があり、テナントのソースグリッド上のグループを表示している場合：

   - バナーメッセージには、グループを編集または削除した場合、変更内容は他のグリッドに同期されないことが示されています。

- 必要に応じて、宛先グリッド上のテナントにグループが複製されなかった場合は、バナーメッセージが表示されます。失敗した操作をグループクローンを再試行できます。

3. グループ名を変更する場合：
   a. 該当するグループのチェックボックスを選択してください。
   b. **Actions** > **Edit group name** を選択します。
   c. 新しい名前を入力してください。
   d. 「変更を保存」を選択します。
4. 詳細を表示したり、追加の編集を行ったりする場合は、以下のいずれかの操作を行ってください。
   - グループ名を選択します。
   - グループのチェックボックスを選択し、**Actions** > **View group details** を選択します。
5. 概要セクションを確認してください。各グループについて以下の情報が表示されています。
   - 表示名
   - 一意の名前
   - Type
   - アクセス モード
   - 権限
   - S3ポリシー
   - このグループのユーザー数
   - テナントアカウントに*グリッドフェデレーション接続の使用*権限があり、テナントのソースグリッドでグループを表示している場合の追加フィールド：
     - クローン作成状況（成功*または*失敗）
     - このグループを編集または削除した場合、変更内容は他のグリッドに同期されないことを示す青いバナーが表示されます。
6. 必要に応じてグループ設定を編集してください。入力内容の詳細については、"S3テナント用のグループを作成する"を参照してください。
   a. 概要セクションで、名前または編集アイコン を選択して表示名を変更します。
   b. 「グループ権限」タブで権限を更新し、「変更を保存」を選択します。
   c. 「グループポリシー」タブで必要な変更を行い、「変更を保存」を選択します。

      必要に応じて、別のS3グループポリシーを選択するか、カスタムポリシーのJSON文字列を入力してください。

7. グループに既存のローカルユーザーを1人以上追加するには：
   a. 「ユーザー」タブを選択します。

b. 「ユーザーを追加」を選択します。

c. 追加したい既存ユーザーを選択し、*ユーザーの追加*を選択します。

画面右上に成功メッセージが表示されます。

8. ローカルユーザーをグループから削除するには：

a. 「ユーザー」タブを選択します。

b. 「ユーザーを削除」を選択します。

c. 削除するユーザーを選択し、*ユーザーの削除*を選択します。

画面右上に成功メッセージが表示されます。

9. 変更した各セクションについて、*変更を保存*を選択したことを確認してください。

重複グループ

既存のグループを複製することで、より迅速に新しいグループを作成できます。

テナントアカウントに*グリッドフェデレーション接続を使用する*権限が付与されている場合、テナントのソースグリッドからグループを複製すると、複製されたグループはテナントの宛先グリッドにクローンされます。

手順

1. アクセス管理 > グループ を選択します。
2. 複製したいグループのチェックボックスを選択してください。
3. アクション > *グループの複製*を選択します。
4. 入力内容の詳細については、"S3テナント用のグループを作成する"を参照してください。
5. 「グループの作成」を選択します。

グループクローンを再試行

失敗したクローンを再試行するには：

1. グループ名の下に「（クローン作成に失敗しました）」と表示されているグループをそれぞれ選択してください。
2. **Actions** > **Clone groups** を選択します。
3. クローン作成対象グループごとに、詳細ページからクローン作成操作のステータスを確認できます。

詳細については、"テナントグループとユーザーを複製する"を参照してください。

1つ以上のグループを削除する

1つまたは複数のグループを削除できます。削除されたグループにのみ所属しているユーザーは、テナントマネージャーにサインインしたり、テナントアカウントを使用したりすることができなくなります。

> テナントアカウントに グリッドフェデレーション接続の使用 権限があり、グループを削除しても、StorageGRID は他のグリッド上の対応するグループを削除しません。この情報を同期させる必要がある場合は、両方のグリッドから同じグループを削除する必要があります。

手順

1. アクセス管理 > グループ を選択します。
2. 削除する各グループのチェックボックスを選択してください。
3. **Actions** > **Delete group** または **Actions** > **Delete groups** を選択します。

   確認ダイアログボックスが表示されます。
4. 「グループを削除」または「グループを削除」を選択します。

AssumeRole の設定

開始する前に

AssumeRole を設定するには、管理者である必要があります。

タスク概要

AssumeRole を設定するには、引き受ける対象グループが存在しない場合は、そのグループを作成します。グループの S3 ポリシーを編集して、このグループを引き受ける際に許可されるアクションを指定します。グループの S3 信頼ポリシーを編集して、AssumeRole API を使用してグループを引き受けることを許可する信頼済みユーザーを指定します。

このグループを引き受けることで作成された一時的なセキュリティ認証情報は、有効期間が限定されています。セッション時間は15分から12時間までで、デフォルトのセッション時間は1時間です。グループのS3信頼ポリシーからユーザーを削除すると、そのユーザーはこのグループを引き受けることができなくなります。

手順

1. アクセス管理 > グループ を選択します。
2. グループ名をクリックしてください。
3. 「S3 trust policy」タブを選択します。
4. S3 信頼ポリシーを追加し、AssumeRole を実行できるユーザーのリストを含めます。
5. 変更を保存 を選択します。
6. 「S3グループポリシー」タブを選択します。
7. このグループのS3信頼ポリシーに追加された信頼済みユーザーに対して、必要なS3アクションのみを指定するようにS3ポリシーを編集します。
8. 変更を保存 を選択します。

AssumeRole S3 信頼ポリシーの例

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "sts:AssumeRole",
            "Principal": {
                "AWS": [
                    "urn:sgws:identity::1234567890:user/user1",
                    "arn:aws:iam::1234567890:user/user2"
                ]
            }
        }
    ]
}
```

設定が完了すると、S3 トラスト ポリシーにリストされているユーザーは AssumeRole を実行して認証情報を受け取ることができます。最終的なアクセス許可は、グループポリシー、バケットポリシー、およびセッションポリシーによって決定されます。詳細については、"アクセスポリシーを使用する" を参照してください。

StorageGRIDテナントユーザーを管理

ローカルユーザーを作成し、ローカルグループに割り当てることで、これらのユーザーがアクセスできる機能を決定できます。連携ユーザーをインポートすることもできます。テナントマネージャーには、「root」という名前の事前定義されたローカルユーザーが1人含まれています。ローカルユーザーの追加や削除は可能ですが、ルートユーザーを削除することはできません。

> StorageGRID システムでシングル サインオン（SSO）が有効になっている場合、ローカルユーザーはテナントマネージャーやテナント管理 API にサインインできませんが、グループ権限に基づいてクライアントアプリケーションを使用してテナントのリソースにアクセスすることは可能です。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"ルートアクセス権限"に属するユーザーグループのメンバーです。
- テナントアカウントに グリッド フェデレーション接続の使用 権限があり、"テナントグループとユーザーのクローン作成"のワークフローと考慮事項を確認済みで、テナントのソースグリッドにサインインしています。

ローカルユーザーを作成する

ローカルユーザーを作成し、そのユーザーを1つまたは複数のローカルグループに割り当てることで、アクセス権限を制御できます。

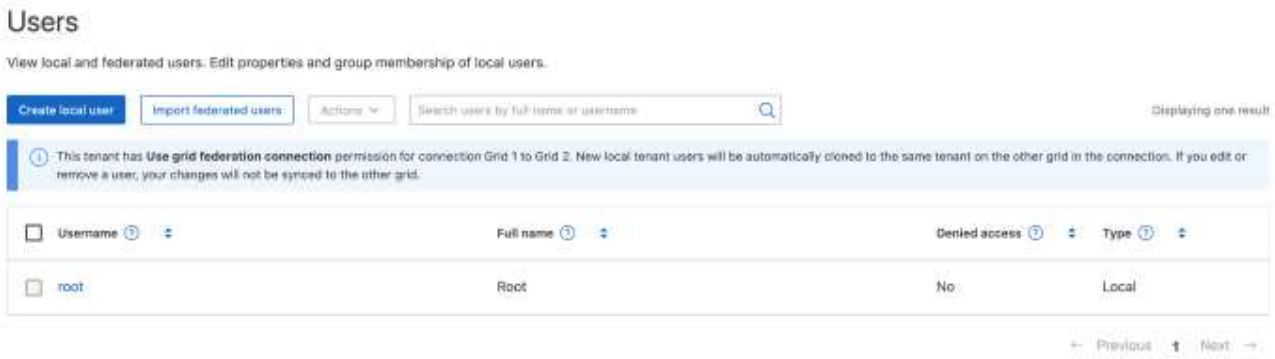
いずれのグループにも属していないS3ユーザーには、管理権限やS3グループポリシーは適用されません。これらのユーザーには、バケットポリシーを通じてS3バケットへのアクセス権が付与されている可能性があります。

ユーザー作成ウィザードにアクセスする

手順

1. アクセス管理 > ユーザー を選択します。

   テナントアカウントに **Use grid federation connection** 権限が付与されている場合、青色のバナーが表示され、これがテナントのソースグリッドであることを示します。このグリッド上で作成したローカルユーザーは、接続先のもう一方のグリッドにも複製されます。



2. 「ユーザーの作成」を選択します。

認証情報を入力

手順

1. 「ユーザー認証情報の入力」ステップでは、以下の項目を入力してください。

フィールド	説明
フルネーム	このユーザーのフルネーム。例えば、人名の場合は姓と名、アプリケーション名など。
Username	このユーザーがログイン時に使用する名前です。ユーザー名は一意である必要があり、変更することはできません。 注: テナントアカウントに グリッド フェデレーション接続の使用 権限がある場合、宛先グリッド上のテナントに同じ ユーザー名 が既に存在すると、クローン作成エラーが発生します。
パスワードとパスワードの確認	ユーザーが最初にログインする際に使用するパスワード。

フィールド	説明
アクセスを拒否する	このユーザーが1つ以上のグループに所属している場合でも、テナントアカウントへのサインインを禁止するには、*はい*を選択してください。 例えば、**Yes** を選択すると、ユーザーのサインイン機能を一時的に停止できます。

2. *続行*を選択します。

グループに割り当てる

手順

1. ユーザーを1つ以上のローカルグループに割り当てて、実行可能なタスクを決定します。

   ユーザーをグループに割り当てるのは任意です。ご希望であれば、グループの作成時または編集時にユーザーを選択することもできます。

   いずれのグループにも所属していないユーザーには、管理権限は付与されません。権限は累積されます。ユーザーは、所属するすべてのグループに対して、すべての権限を持つことになります。参照してください "テナント管理権限"。

2. 「ユーザーの作成」を選択します。

   テナントアカウントに「グリッドフェデレーション接続を使用する」権限があり、かつテナントのソースグリッド上にいる場合、新しいローカルユーザーがテナントの宛先グリッドに複製されます。ユーザーの詳細ページの概要セクションの「クローン作成ステータス」に「成功」と表示されます。

3. 「完了」を選択して、ユーザーページに戻ります。

ローカルユーザーの表示または編集

手順

1. アクセス管理 > ユーザー を選択します。
2. ユーザーページに記載されている情報を確認してください。このページには、このテナントアカウントのすべてのローカルユーザーおよびフェデレーションユーザーに関する基本情報が記載されています。

   テナントアカウントに*グリッドフェデレーション接続の使用*権限があり、テナントのソースグリッドでユーザーを表示している場合：

   - バナーメッセージには、ユーザーを編集または削除した場合、変更内容は他のグリッドに同期されないことが示されています。
   - 必要に応じて、宛先グリッド上でユーザーがテナントにクローンされなかった場合は、バナーメッセージが表示されます。失敗したユーザークローンを再試行することができます。

3. ユーザーのフルネームを変更する場合は、次の手順に従ってください。
   a. ユーザーのチェックボックスを選択します。
   b. アクション > *氏名を編集*を選択します。
   c. 新しい名前を入力してください。

d. 「変更を保存」を選択します。

4. 詳細を表示したり、追加の編集を行ったりする場合は、以下のいずれかの操作を行ってください。
   - ユーザー名を選択します。
   - 該当ユーザーのチェックボックスを選択し、**Actions** > **View user details** を選択します。
5. 概要セクションを確認してください。各ユーザーについて、以下の情報が表示されます。
   - フルネーム
   - Username
   - ユーザータイプ
   - アクセス拒否
   - アクセス モード
   - グループ メンバーシップ
   - テナントアカウントに*グリッドフェデレーション接続の使用*権限があり、テナントのソースグリッドでユーザーを表示している場合の追加フィールド：
     - クローン作成状況（成功*または*失敗）
     - このユーザーを編集した場合、変更内容は他のグリッドに同期されないことを示す青いバナーが表示されます。
6. 必要に応じてユーザー設定を編集します。入力内容の詳細については、ローカルユーザーを作成するを参照してください。
   a. 概要セクションで、名前または編集アイコン を選択してフルネームを変更します。

      ユーザー名は変更できません。

   b. 「パスワード」タブでユーザーのパスワードを変更し、「変更を保存」を選択します。
   c. アクセス タブで、ユーザーのサインインを許可する場合は いいえ を、サインインを禁止する場合は はい を選択します。次に、変更を保存 を選択します。
   d. 「アクセスキー」タブで「キーの作成」を選択し、"別のユーザーのS3アクセスキーを作成する"の指示に従ってください。
   e. 「グループ」タブで、「グループの編集」を選択すると、ユーザーをグループに追加したり、グループから削除したりできます。次に、「変更を保存」を選択します。
7. 変更した各セクションについて、*変更を保存*を選択したことを確認してください。

フェデレーションユーザーをインポート

1人または複数人の連携ユーザー（最大100人まで）を、ユーザーページに直接インポートできます。

手順

1. アクセス管理 > ユーザー を選択します。
2. 「フェデレーションユーザーのインポート」を選択します。
3. 1人以上のフェデレーションユーザーのUUIDまたはユーザー名を入力してください。

   複数のエントリがある場合は、各UUIDまたはユーザー名を新しい行に追加してください。

4. 「インポート」を選択します。

   1人以上のユーザーについて「ユーザー」フィールドへのインポートが失敗した場合は、以下の手順を実行してください。

   a. 「インポートされていないユーザー」を展開し、「ユーザーのコピー」を選択します。

   b. 前へ を選択し、コピーしたユーザーを フェデレーションユーザーのインポート ダイアログボックスに貼り付けて、インポートを再試行してください。

   「フェデレーションユーザーのインポート」ダイアログボックスを閉じると、正常にインポートされたユーザーのフェデレーションユーザー情報が「ユーザー」ページに表示されます。

ローカルユーザーの複製

ローカルユーザーを複製することで、より迅速に新しいユーザーを作成できます。

> テナントアカウントに*グリッドフェデレーション接続を使用する*権限が付与されている場合、テナントのソースグリッドからユーザーを複製すると、複製されたユーザーはテナントの宛先グリッドにクローンされます。

手順

1. アクセス管理 ＞ ユーザー を選択します。
2. 複製したいユーザーのチェックボックスを選択してください。
3. アクション > ユーザーの複製 を選択します。
4. 入力内容の詳細については、ローカルユーザーを作成するを参照してください。
5. 「ユーザーの作成」を選択します。

ユーザークローンを再試行

失敗したクローンを再試行するには：

1. ユーザー名の下に「*(Cloning failed)*」と表示されているユーザーをそれぞれ選択してください。
2. アクション > ユーザーの複製 を選択します。
3. クローン作成中の各ユーザーの詳細ページから、クローン作成操作のステータスを確認できます。

詳細については、"テナントグループとユーザーを複製する"を参照してください。

1人以上のローカルユーザーを削除する

StorageGRID テナントアカウントへのアクセスが不要になった1人以上のローカルユーザーを完全に削除できます。

> テナントアカウントに グリッドフェデレーション接続の使用 権限があり、ローカルユーザーを削除した場合、StorageGRID は他のグリッド上の対応するユーザーを削除しません。この情報を同期させる必要がある場合は、両方のグリッドから同じユーザーを削除する必要があります。

フェデレーションユーザーを削除するには、フェデレーションIDソースを使用する必要があります。

手順

1. アクセス管理 > ユーザー を選択します。
2. 削除する各ユーザーのチェックボックスを選択してください。
3. **Actions** > **Delete user** または **Actions** > **Delete users** を選択します。

   確認ダイアログボックスが表示されます。

4. 「ユーザーを削除」または「ユーザーを削除」を選択します。

S3アクセスキーの管理

StorageGRIDでS3アクセスキーを管理

S3テナントアカウントの各ユーザーは、StorageGRIDシステムでオブジェクトを保存および取得するためのアクセスキーを持っている必要があります。アクセスキーは、アクセスキーIDとシークレットアクセスキーで構成されます。

S3アクセスキーは次のように管理できます：

- *S3認証情報の管理*権限を持つユーザーは、自身のS3アクセスキーを作成または削除できます。
- *ルートアクセス*権限を持つユーザーは、S3ルートアカウントおよび他のすべてのユーザーのアクセスキーを管理できます。ルートアクセスキーは、バケットポリシーによって明示的に無効化されていない限り、テナントに対してすべてのバケットとオブジェクトへの完全なアクセス権を提供します。

StorageGRID は、Signature Version 2 および Signature Version 4 の認証をサポートしています。バケットポリシーで明示的に有効にしない限り、アカウント間のアクセスは許可されません。

StorageGRID で独自の S3 アクセスキーを作成

S3テナントを使用しており、適切な権限を持っている場合は、独自のS3アクセスキーを作成できます。バケットやオブジェクトにアクセスするには、アクセスキーが必要です。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"S3認証情報またはルートアクセス権限を管理する"を持つユーザーグループに属しています。

タスク概要

テナントアカウントのバケットを作成および管理するために、1つまたは複数のS3アクセスキーを作成できます。新しいアクセスキーを作成したら、アプリケーションに新しいアクセスキーIDとシークレットアクセスキーを追加して更新してください。セキュリティ上の理由から、必要以上のキーは作成せず、使用していないキーは削除してください。キーが1つしかなく、その有効期限が近づいている場合は、古いキーの有効期限が切れる前に新しいキーを作成し、その後古いキーを削除してください。

各キーには、特定の有効期限を設定することも、有効期限を設定しないことも可能です。有効期限については、以下のガイドラインに従ってください。

- アクセスを一定期間に制限するために、キーの有効期限を設定してください。有効期限を短く設定することで、アクセスキーIDと秘密アクセスキーが誤って漏洩した場合のリスクを軽減できます。有効期限切れのキーは自動的に削除されます。
- 環境におけるセキュリティリスクが低く、定期的に新しいキーを作成する必要がない場合は、キーの有効期限を設定する必要はありません。後で新しいキーを作成する場合は、古いキーを手動で削除してください。

> テナントマネージャーに表示されているアカウントのアクセスキーIDとシークレットアクセスキーを使用すると、アカウントに属するS3バケットとオブジェクトにアクセスできます。そのため、アクセスキーはパスワードと同様に厳重に保護してください。アクセスキーは定期的に変更し、使用していないキーはアカウントから削除し、他のユーザーと共有しないでください。

手順

1. **STORAGE (S3)** ＞ マイアクセスキー を選択します。

   「マイアクセスキー」ページが表示され、既存のアクセスキーが一覧表示されます。

2. 「キーの作成」を選択します。
3. 次のいずれかを実行します。
   - 「有効期限を設定しない」を選択すると、有効期限のないキーを作成できます。（デフォルト）
   - 「有効期限を設定」を選択し、有効期限の日時を設定します。

     > 有効期限は、現在の日付から最長で5年間とすることができます。有効期限は、現在時刻から最短で1分後とすることができます。

4. 「アクセスキーの作成」を選択します。

   アクセスキーのダウンロードダイアログボックスが表示され、アクセスキーIDとシークレットアクセスキーが表示されます。

5. アクセスキーIDとシークレットアクセスキーを安全な場所にコピーするか、**Download .csv** を選択して、アクセスキーIDとシークレットアクセスキーを含むスプレッドシートファイルを保存してください。

   > この情報をコピーまたはダウンロードするまで、このダイアログボックスを閉じないでください。ダイアログボックスを閉じた後は、キーをコピーまたはダウンロードすることはできません。

6. **Finish** を選択します。

   新しいキーは「マイアクセスキー」ページに表示されます。

7. テナントアカウントに「グリッドフェデレーション接続の使用」権限が付与されている場合は、必要に応じてテナント管理APIを使用して、ソースグリッド上のテナントから宛先グリッド上のテナントにS3アクセスキーを手動で複製できます。"APIを使用してS3アクセスキーを複製する"を参照してください。

StorageGRIDでS3アクセスキーを表示する

S3 テナントを使用していて、"適切な許可"がある場合は、S3 アクセスキーの一覧を表示できます。リストを有効期限順に並べ替えることができるので、どのキーが間もなく期限切れになるかを確認できます。必要に応じて、使用しなくなったキーを"新しいキーを作成する"または"削除キー"できます。

> テナントマネージャーに表示されているアカウントのアクセスキーIDとシークレットアクセスキーを使用すると、アカウントに属するS3バケットとオブジェクトにアクセスできます。そのため、アクセスキーはパスワードと同様に厳重に保護してください。アクセスキーは定期的に変更し、使用していないキーはアカウントから削除し、他のユーザーと共有しないでください。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、S3認証情報を管理する権限を持つユーザーグループに属しています"アクセス権"。

手順

1. **STORAGE (S3)** ＞ マイアクセスキー を選択します。
2. 「マイアクセスキー」ページから、既存のアクセスキーを「有効期限」または「アクセスキー ID」で並べ替えます。
3. 必要に応じて、新しいキーを作成するか、使用しなくなったキーを削除してください。

   既存のキーの有効期限が切れる前に新しいキーを作成すれば、アカウント内のオブジェクトへのアクセス権を一時的に失うことなく、新しいキーの使用を開始できます。

   有効期限切れのキーは自動的に削除されます。

StorageGRID で自分の S3 アクセスキーを削除する

S3テナントを使用しており、適切な権限を持っている場合は、自身のS3アクセスキーを削除できます。アクセスキーが削除されると、そのキーを使用してテナントアカウント内のオブジェクトやバケットにアクセスすることはできなくなります。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは"S3認証情報のアクセス許可を管理する"があります。

> テナントマネージャーに表示されているアカウントのアクセスキーIDとシークレットアクセスキーを使用すると、アカウントに属するS3バケットとオブジェクトにアクセスできます。そのため、アクセスキーはパスワードと同様に厳重に保護してください。アクセスキーは定期的に変更し、使用していないキーはアカウントから削除し、他のユーザーと共有しないでください。

手順

1. **STORAGE (S3)** ＞ マイアクセスキー を選択します。

2. 「マイアクセスキー」ページで、削除する各アクセスキーのチェックボックスを選択します。
3. 「Delete キー」を選択します。
4. 確認ダイアログボックスから「Delete key」を選択します。

   ページの右上隅に確認メッセージが表示されます。

StorageGRID で別のユーザーのS3アクセスキーを作成します

S3テナントを使用しており、適切な権限を持っている場合は、バケットやオブジェクトへのアクセスが必要なアプリケーションなど、他のユーザー向けにS3アクセスキーを作成できます。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"ルートアクセス権限"に属するユーザーグループのメンバーです。

タスク概要

他のユーザー向けに1つまたは複数のS3アクセスキーを作成することで、それらのユーザーが自身のテナントアカウント用のバケットを作成および管理できるようになります。新しいアクセスキーを作成したら、アプリケーションを新しいアクセスキーIDとシークレットアクセスキーで更新してください。セキュリティ上の理由から、ユーザーが必要とする以上のキーは作成せず、使用されていないキーは削除してください。キーが1つしかなく、その有効期限が近づいている場合は、古いキーの有効期限が切れる前に新しいキーを作成し、その後古いキーを削除してください。

各キーには、特定の有効期限を設定することも、有効期限を設定しないことも可能です。有効期限については、以下のガイドラインに従ってください。

- ユーザーのアクセスを一定期間に制限するために、キーの有効期限を設定してください。有効期限を短く設定することで、アクセスキーIDと秘密アクセスキーが誤って漏洩した場合のリスクを軽減できます。有効期限切れのキーは自動的に削除されます。
- 環境におけるセキュリティリスクが低く、定期的に新しいキーを作成する必要がない場合は、キーの有効期限を設定する必要はありません。後で新しいキーを作成する場合は、古いキーを手動で削除してください。

> ユーザーに属するS3バケットとオブジェクトには、テナントマネージャーに表示されるそのユーザーのアクセスキーIDとシークレットアクセスキーを使用してアクセスできます。そのため、アクセスキーはパスワードと同様に厳重に保護してください。アクセスキーは定期的に変更し、使用していないキーはアカウントから削除し、他のユーザーと共有しないでください。

手順

1. アクセス管理 > ユーザー を選択します。
2. 管理する S3 アクセスキーのユーザーを選択します。

   ユーザー詳細ページが表示されます。

3. 「アクセスキー」を選択し、次に「キーの作成」を選択します。
4. 次のいずれかを実行します。

- *有効期限を設定しない*を選択すると、有効期限のないキーが作成されます。（デフォルト）
- 「有効期限を設定」を選択し、有効期限の日時を設定します。

> 有効期限は、現在の日付から最長で5年間とすることができます。有効期限は、現在時刻から最短で1分後とすることができます。

5. 「アクセスキーの作成」を選択します。

   アクセスキーのダウンロードダイアログボックスが表示され、アクセスキーIDとシークレットアクセスキーが表示されます。

6. アクセスキーIDとシークレットアクセスキーを安全な場所にコピーするか、**Download .csv** を選択して、アクセスキーIDとシークレットアクセスキーを含むスプレッドシートファイルを保存してください。

> この情報をコピーまたはダウンロードするまで、このダイアログボックスを閉じないでください。ダイアログボックスを閉じた後は、キーをコピーまたはダウンロードすることはできません。

7. **Finish** を選択します。

   新しいキーは、ユーザー詳細ページの「アクセスキー」タブに表示されます。

8. テナントアカウントに「グリッドフェデレーション接続の使用」権限が付与されている場合は、必要に応じてテナント管理APIを使用して、ソースグリッド上のテナントから宛先グリッド上のテナントにS3アクセスキーを手動で複製できます。"APIを使用してS3アクセスキーを複製する"を参照してください。

StorageGRID で他のユーザーの **S3** アクセスキーを表示する

S3テナントを使用しており、適切な権限を持っている場合は、他のユーザーのS3アクセスキーを表示できます。有効期限順にリストを並べ替えることで、どのキーが間もなく期限切れになるかを確認できます。必要に応じて、新しいキーを作成したり、不要になったキーを削除したりできます。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは"ルートアクセス権限"があります。

> ユーザーに属するS3バケットとオブジェクトには、テナントマネージャーに表示されるそのユーザーのアクセスキーIDとシークレットアクセスキーを使用してアクセスできます。そのため、アクセスキーはパスワードと同様に厳重に保護してください。アクセスキーは定期的に変更し、使用していないキーはアカウントから削除し、他のユーザーと共有しないでください。

手順

1. アクセス管理 > ユーザー を選択します。
2. 「ユーザー」ページから、S3アクセスキーを表示するユーザーを選択します。
3. ユーザー詳細ページから「アクセスキー」を選択します。
4. キーを「有効期限」または「アクセスキー ID」で並べ替えます。

5. 必要に応じて新しいキーを作成し、使用されなくなったキーは手動で削除してください。

   既存のキーの有効期限が切れる前に新しいキーを作成すれば、ユーザーはアカウント内のオブジェクトへのアクセス権を一時的に失うことなく、新しいキーの使用を開始できます。

   有効期限切れのキーは自動的に削除されます。

関連情報

- "別のユーザーのS3アクセスキーを作成する"
- "他のユーザーのS3アクセスキーを削除する"

StorageGRID で他のユーザーの S3 アクセスキーを削除する

S3テナントを使用しており、適切な権限を持っている場合は、他のユーザーのS3アクセスキーを削除できます。アクセスキーが削除されると、そのキーを使用してテナントアカウント内のオブジェクトやバケットにアクセスすることはできなくなります。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは"ルートアクセス権限"があります。

> ユーザーに属するS3バケットとオブジェクトには、テナントマネージャーに表示されるそのユーザーのアクセスキーIDとシークレットアクセスキーを使用してアクセスできます。そのため、アクセスキーはパスワードと同様に厳重に保護してください。アクセスキーは定期的に変更し、使用していないキーはアカウントから削除し、他のユーザーと共有しないでください。

手順

1. アクセス管理 > ユーザー を選択します。
2. 「ユーザー」ページから、S3アクセスキーを管理するユーザーを選択します。
3. ユーザー詳細ページから「アクセスキー」を選択し、削除する各アクセスキーのチェックボックスを選択します。
4. **Actions** > *選択したキーを削除*を選択します。
5. 確認ダイアログボックスから「Delete key」を選択します。

   ページの右上隅に確認メッセージが表示されます。

S3バケットを管理する

StorageGRID S3バケットを作成する

テナントマネージャーを使用して、オブジェクトデータ用のS3バケットを作成できます。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。

- あなたは、Root access またはすべてのバケットを管理する権限を持つユーザーグループに属しています "アクセス権"。これらの権限は、グループポリシーまたはバケットポリシーの権限設定を上書きします。

> バケットまたはオブジェクトの S3 オブジェクト ロック プロパティを設定または変更する権限は、"バケットポリシーまたはグループポリシー"によって付与できます。

- バケットに対して S3 オブジェクトロックを有効にする予定がある場合、グリッド管理者が StorageGRID システムのグローバル S3 オブジェクトロック設定を有効にしており、S3 オブジェクトロックのバケットとオブジェクトの要件を確認済みであること。
- 各テナントが5,000個のバケットを持つ場合、グリッド内の各ストレージノードには最低でも64 GBのRAMが必要です。

> 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（"ブランチバケット"を含む）。

ウィザードにアクセスする

手順

1. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3) > Buckets** を選択してください。
2. 「バケットの作成」を選択します。

詳細を入力

手順

1. バケットの詳細を入力してください。

フィールド	説明
バケット名	これらのルールに準拠したバケット名： • 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だけでなく）。 • DNS規格に準拠している必要があります。 • 3文字以上63文字以内である必要があります。 • 各ラベルは小文字の英字または数字で始まり、小文字の英字または数字で終わる必要があり、使用できる文字は小文字の英字、数字、およびハイフンのみです。 • 仮想ホスト型スタイルのリクエストには、ピリオドを含めてはいけません。ピリオドは、サーバーのワイルドカード証明書の検証に問題を引き起こします。 詳細については、 "Amazon Web Services (AWS) のバケット命名規則に関するドキュメント"を参照してください。 注：バケットを作成した後は、バケット名を変更することはできません。

フィールド	説明
リージョン	バケットのリージョン。 StorageGRID管理者が利用可能なリージョンを管理します。バケットのリージョンは、オブジェクトに適用されるデータ保護ポリシーに影響する場合があります。デフォルトでは、すべてのバケットは `us-east-1`リージョンに作成されます。デフォルトリージョンが `us-east-1`以外のリージョンに設定されている場合、ドロップダウンリストでは最初にそのリージョンが選択されます。 注：バケット作成後にリージョンを変更することはできません。

2. *続行*を選択します。

設定の管理

手順

1. 必要に応じて、バケットのオブジェクトバージョン管理を有効にします。

   このバケットに各オブジェクトのすべてのバージョンを保存する場合は、オブジェクトのバージョン管理を有効にしてください。必要に応じて、オブジェクトの以前のバージョンを取得できます。

   オブジェクトのバージョン管理を有効にする必要があるのは、以下の場合です。

   - このバケットは、グリッド間レプリケーションに使用されます。
   - このバケットから"ブランチバケット"を作成します。

2. グローバルな S3 オブジェクトロック設定が有効になっている場合、必要に応じてバケットの S3 オブジェクトロックを有効にして、write-once-read-many （WORM）モデルを使用してオブジェクトを保存します。

   S3オブジェクトロックは、例えば特定の規制要件を満たすために、オブジェクトを一定期間保持する必要がある場合にのみ、バケットに対して有効にしてください。S3オブジェクトロックは、一定期間または無期限にオブジェクトの削除や上書きを防止する永続的な設定です。

   > バケットに対してS3オブジェクトロック設定を有効にすると、無効にすることはできません。適切な権限を持つユーザーであれば誰でも、変更できないオブジェクトをこのバケットに追加できます。これらのオブジェクトやバケット自体を削除できない場合があります。

   バケットに対してS3オブジェクトロックを有効にすると、バケットのバージョン管理が自動的に有効になります。

3. 「S3オブジェクトロックを有効にする」を選択した場合は、必要に応じてこのバケットの「デフォルト保持」を有効にしてください。

   > グリッド管理者から"S3オブジェクトロックの特定の機能を使用する"の権限を付与してもらう必要があります。

   *デフォルト保持*が有効になっている場合、バケットに追加された新しいオブジェクトは、削除または上

書きされないように自動的に保護されます。*デフォルト保持*設定は、独自の保持期間が設定されているオブジェクトには適用されません。

a. *デフォルト保持*が有効になっている場合は、バケットの*デフォルト保持モード*を指定してください。

デフォルトの保持モード	説明
ガバナンス	• `s3:BypassGovernanceRetention`権限を持つユーザーは、 `x-amz-bypass-governance-retention: true`リクエストヘッダーを使用して保持設定をバイパスできます。 • これらのユーザーは、オブジェクトのバージョンが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。
コンプライアンス	• オブジェクトは、保持期限が到来するまで削除できません。 • オブジェクトの保持期限は延長できますが、短縮することはできません。 • オブジェクトの保持期限日は、その日付が到来するまで削除することはできません。 注：グリッド管理者がコンプライアンスモードの使用を許可している必要があります。

b. *デフォルト保持*が有効になっている場合は、バケットの*デフォルト保持期間*を指定してください。

*デフォルトの保持期間*は、このバケットに追加された新しいオブジェクトが、取り込まれた時点からどのくらいの期間保持されるかを示します。グリッド管理者が設定したテナントの最大保持期間以下の値を指定してください。

_最大_保持期限は、グリッド管理者がテナントを作成する際に設定されます。この値は1日から100年までの範囲で設定可能です。_デフォルト_の保持期限を設定する場合、最大保持期限に設定された値を超えることはできません。必要に応じて、グリッド管理者に最大保持期限の延長または短縮を依頼してください。

4. 必要に応じて、*容量制限を有効にする*を選択し、値を入力して、容量の単位を選択します。

容量制限とは、このバケット内のオブジェクトに利用可能な最大容量のことです。この値は論理的な量（オブジェクトのサイズ）を表しており、物理的な量（ディスク上のサイズ）を表すものではありません。

制限が設定されていない場合、このバケットの容量は無制限です。詳細については、"容量制限の使用状況"を参照してください。

5. 必要に応じて、*オブジェクト数制限を有効にする*を選択します。

オブジェクト数の制限とは、このバケットに格納できるオブジェクトの最大数です。この値は論理的な量（オブジェクト数）を表します。制限が設定されていない場合、オブジェクトの数は無制限です。

6. 「バケットの作成」を選択します。

バケットが作成され、バケットページのテーブルに追加されます。

7. 必要に応じて、*バケットの詳細ページへ移動*を選択して"バケットの詳細を表示"追加の設定を行います。

また、必要に応じて"ブランチバケットを作成する"することもできます。

StorageGRID で S3 バケットの詳細を表示する

テナントアカウントでバケットを確認できます。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"ルートアクセス、すべてのバケットの管理、またはすべてのバケットの表示権限" を持つユーザーグループに属しています。これらの権限は、グループポリシーまたはバケットポリシーの権限設定を上書きします。

手順

1. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3)** > **Buckets** を選択してください。

   「バケット」ページが表示されます。

2. 各バケットの概要テーブルを確認してください。

   必要に応じて、任意の列で情報を並べ替えたり、リストを前後に移動したりできます。

   > ⓘ 表示されているオブジェクト数、使用容量、および使用状況の値は推定値です。これらの推定値は、データ取り込みのタイミング、ネットワーク接続性、およびノードの状態によって影響を受けます。バケットでバージョン管理が有効になっている場合、削除されたオブジェクトのバージョンもオブジェクト数に含まれます。

   Name
   バケットの固有名。変更することはできません。

   有効な機能
   バケットで有効になっている機能の一覧。

   S3 Object Lock
   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。

   この列は、グリッドに対してS3オブジェクトロックが有効になっている場合にのみ表示されます。この列には、従来の準拠バケットに関する情報も表示されます。

   リージョン
   バケットのリージョンは変更できません。この列はデフォルトでは非表示になっています。

   オブジェクト数
   このバケット内のオブジェクトの数。バケットでバージョン管理が有効になっている場合、この値には最新ではないオブジェクトバージョンも含まれます。

オブジェクトが追加または削除された場合、この値はすぐに更新されない場合があります。

使用済みスペース
: バケット内のすべてのオブジェクトの論理サイズ。論理サイズには、複製されたコピーや消去符号化されたコピー、またはオブジェクトのメタデータに必要な実際の容量は含まれません。

  この値の更新には最大10分かかる場合があります。

用途
: バケットの容量制限が設定されている場合、その制限に対する使用率。

  使用量の値は社内推定値に基づいており、場合によっては超過する可能性があります。例えば、StorageGRID はテナントがオブジェクトのアップロードを開始した際に容量制限（設定されている場合）を確認し、テナントが容量制限を超えている場合は、このバケットへの新規取り込みを拒否します。ただし、StorageGRID は容量制限を超過したかどうかを判断する際に、現在のアップロードのサイズは考慮されません。オブジェクトが削除された場合、容量制限の使用状況が再計算されるまで、テナントはこのバケットに新しいオブジェクトをアップロードできなくなる場合があります。計算には10分以上かかる場合があります。

  この値は、オブジェクトとそのメタデータを格納するために必要な物理的なサイズではなく、論理的なサイズを示しています。

Capacity
: 設定されている場合、バケットの容量制限。

作成日
: バケットが作成された日時。この列はデフォルトでは非表示になっています。

3. 特定のバケットの詳細を表示するには、表からバケット名を選択してください。
   a. ウェブページ上部の概要情報を参照して、リージョンやオブジェクト数など、バケットの詳細を確認してください。
   b. 容量制限の使用状況とオブジェクト数制限の使用状況を示すバーを表示します。使用率が100%またはそれに近い場合は、制限値を上げるか、一部のオブジェクトを削除することを検討してください。
   c. 必要に応じて、*バケット内のオブジェクトを削除*と*バケットを削除*を選択してください。

      これらのオプションを選択する際に表示される警告に十分注意してください。詳細については、以下を参照してください。

      - "バケット内のすべてのオブジェクトを削除する"
      - "バケットを削除する"（バケットは空である必要があります）

   d. 必要に応じて、各タブでバケットの設定を表示または変更してください。
      - **S3**コンソール：バケット内のオブジェクトを表示します。詳細については、"S3コンソールを使用する"を参照してください。
      - バケットオプション：オプション設定を表示または変更します。S3 Object Lockなどの一部の設定は、バケット作成後に変更できません。
        - "バケットの一貫性を管理する"

        - "最終アクセス時刻の更新"
        - "容量制限"
        - "オブジェクト数の制限"
        - "オブジェクトのバージョン管理"
        - "S3 Object Lock"
        - "デフォルトのバケット保持期間"
        - "クロスグリッドレプリケーションを管理する"（テナントに許可されている場合）
    - プラットフォームサービス："プラットフォームサービスを管理する"（テナントに許可されている場合）
    - バケットへのアクセス：オプション設定を表示または変更します。特定のアクセス権限が必要です。
        - バケットおよびバケット内のオブジェクトが他のドメインのWebアプリケーションからアクセスできるように、"バケットとオブジェクトのCORS"を設定します。
        - "ユーザーアクセスを制御する"S3バケットとそのバケット内のオブジェクトに対して。
    - ブランチ: バケットのブランチバケットの一覧を表示します。"新しいブランチバケットを作成するか、ブランチバケットを管理する"。

StorageGRIDブランチバケットについて学ぶ

ブランチバケットを使用すると、バケット内のオブジェクトを特定の時点の状態で取得できます。

既存のバケットからブランチバケットを作成します。ブランチバケットを作成すると、そのブランチバケットの作成元となった元のバケットは_ベースバケット_と呼ばれます。さらに、別のブランチバケットからブランチバケットを作成することもできます。

ブランチバケットは保護されたデータへのアクセスを提供しますが、バックアップとしては機能しません。データの保護を継続するには、ベースバケットで以下の機能を使用してください：

- "S3 Object Lock"
- "クロスグリッド複製"基本バケット用
- "バケットポリシー"バージョン管理されたバケットで古いオブジェクトバージョンをクリーンアップする

ブランチバケットの以下の特徴に注意してください。

- ブランチバケット内のオブジェクトには、"S3コンソールでオブジェクトをダウンロードする"を使用してアクセスできます。
- クライアントがブランチバケット内のオブジェクトにアクセスすると、ベースバケットのポリシーではなく、ブランチバケットの"アクセスポリシー"によって、アクセスを許可するか拒否するかが決定されます。
- ベースバケットで作成されたオブジェクトは、"ILMルール"がベースバケットにどのように適用されるかに基づいて評価されます。ブランチバケット内で作成されたオブジェクトは、ILMルールがそのブランチバケットにどのように適用されるかに基づいて評価されます。
- ブランチバケットでは、クロスグリッドレプリケーションはサポートされていません。

- ブランチバケットではプラットフォームサービスはサポートされていません。

ブランチバケットの使用例

- ブランチバケットを使用すると、破損が発生する前の時点からブランチバケットを作成し、アプリケーションを破損オブジェクトを含むベースバケットではなく、そのブランチバケットに向けることで、破損オブジェクトを削除できます。
- バージョン管理されたバケットにデータを保存しています。時間 *T* 以降に、意図しない多数のオブジェクトが取り込まれるという偶発的な脆弱性が発生しました。「Before」時間値 *T* に対応するブランチバケットを作成し、クライアント操作をそのブランチバケットにリダイレクトすることができます。これにより、Before 時間 *T* より前に取り込まれたオブジェクトのみがクライアントに公開されます。

ブランチバケット内のオブジェクトに対する操作

- ブランチバケットに対するPUTオブジェクト操作は、そのブランチ内にオブジェクトを作成します。
- ブランチバケットに対するGETオブジェクト操作は、ブランチからオブジェクトを取得します。オブジェクトがブランチバケットに存在しない場合、ベースバケットからオブジェクトが取得されます。
- ブランチバケットからのオブジェクト削除は、次のように行われます。

処理	ターゲット	結果	ベースバケット内のオブジェクトの可視性	ブランチバケット内のオブジェクトの可視性
バージョンIDなしで削除	ベースバケット	削除マーカーはベースバケットに対してのみ作成されます	HEAD/GETはオブジェクトが存在しないことを返しますが、特定のバージョンには引き続きアクセスできます	HEAD/GETはオブジェクトが存在することを返し、特定のバージョンには引き続きアクセスできます 削除マーカーはブランチバケットの `beforeTime` 後に作成されます。
バージョンIDで削除	ベースバケット	ベースバケットとブランチバケットの両方で、特定のオブジェクトバージョンが削除されます。	HEAD/GETリクエストで「オブジェクトバージョンが存在しません」というエラーが返されます	HEAD/GETリクエストで「オブジェクトバージョンが存在しません」というエラーが返されます
バージョンIDなしで削除	ブランチバケット	削除マーカーはブランチバケットに対してのみ作成されます	HEAD/GETはオブジェクトを返します（ベースバケットオブジェクトは影響を受けません）	HEAD/GETリクエストで「オブジェクトが存在しません」というエラーが返されます
バージョンIDで削除	ブランチバケット	特定のオブジェクトバージョンは、ブランチバケットに対してのみ削除されます。	HEAD/GETは特定のオブジェクトバージョンを返します（ベースバケットオブジェクトは影響を受けません）	HEAD/GETリクエストで「オブジェクトバージョンが存在しません」というエラーが返されます

また、"S3 バージョン管理されたオブジェクトはどのように削除されるのか"を参照してください。

テナントマネージャーを使用して、ブランチバケットの作成と詳細の表示を行います。

開始する前に

- "対応ウェブブラウザ"を使用してTenant Managerにサインインしました。
- あなたはルートアクセスまたは"すべてのバケットの権限を管理する"を持つユーザーグループに属しています。これらの権限は、グループポリシーまたはバケットポリシーの権限設定を上書きします。
- ブランチを作成するベースバケットには"バージョン管理が有効になっています"があります。
- あなたはベースバケットの所有者です。

タスク概要

ブランチバケットについては、以下の情報にご注意ください。

- バケットまたはオブジェクトの S3 オブジェクトロックプロパティを設定する権限は、"バケットポリシーまたはグループポリシー"によって付与できます。
- ベースバケットでバージョン管理を一時停止すると、ベースバケットの内容はブランチバケットに表示されなくなります。

> ブランチバケットを設定して作成した後は、設定を変更することはできません。

ブランチバケットを作成する

手順

1. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3)** > **Buckets** を選択してください。
2. ブランチを作成する元のバケット（「ベースバケット」）を選択してください。
3. バケットの詳細ページで、「Branches」>「Create branch bucket」を選択します。

   ベースバケットでバージョン管理が有効になっていない場合、「ブランチバケットを作成」ボタンは無効になります。

詳細を入力

手順

1. ブランチバケットの詳細を入力してください。

フィールド	説明
ブランチバケット名	これらのルールに準拠するブランチバケットの名前： • 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だけでなく）。 • DNS規格に準拠している必要があります。 • 3文字以上63文字以内である必要があります。 • 各ラベルは小文字の英字または数字で始まり、小文字の英字または数字で終わる必要があり、使用できる文字は小文字の英字、数字、およびハイフンのみです。 • 仮想ホスト型スタイルのリクエストには、ピリオドを含めてはいけません。ピリオドは、サーバーのワイルドカード証明書の検証に問題を引き起こします。 詳細については、 "Amazon Web Services (AWS) のバケット命名規則に関するドキュメント"を参照してください。 注：ブランチバケットを作成した後は、名前を変更できません。
リージョン（ブランチバケットでは変更できません）	ブランチバケットのリージョン。 ブランチバケットのリージョンはベースバケットのリージョンと一致する必要があるため、このフィールドはブランチバケットでは無効になっています。
事前に	ベースバケットで作成されたオブジェクトバージョンがブランチバケットからアクセス可能になる期限。ブランチバケットを使用すると、Before タイムより前に作成されたオブジェクトバージョンにアクセスできます。 「以前」とは、既に経過した日付と時刻を指します。未来の日付は指定できません。
ブランチバケットタイプ	• 読み書き：ブランチバケット内のオブジェクトまたはオブジェクトバージョンを追加または削除できます。 • 読み取り専用：ブランチバケット内のオブジェクトは変更できません。 注: ブランチバケットが空の場合に限り、ブランチバケットのタイプを読み取り専用に設定できます。既存のブランチバケットのタイプが読み書き可能に設定されていて、まだ書き込みを行っていない場合は、タイプを読み取り専用に変更できます。

2. *続行*を選択します。

オブジェクト設定の管理（オプション）

ブランチバケットのオブジェクト設定は、ベースバケット内のオブジェクトバージョンには影響しません。

手順

1. グローバルなS3オブジェクトロック設定が有効になっている場合は、必要に応じてブランチバケットのS3オブジェクトロックを有効にします。S3オブジェクトロックを有効にするには、ブランチバケットが読み書き可能なバケットである必要があります。

   ブランチバケットに対してS3オブジェクトロックを有効にするのは、たとえば特定の規制要件を満たすために、オブジェクトを一定期間保持する必要がある場合に限ってください。S3オブジェクトロックは、一定期間または無期限にオブジェクトの削除や上書きを防止する永続的な設定です。

   > バケットに対してS3オブジェクトロック設定を有効にすると、無効にすることはできません。適切な権限を持つユーザーであれば誰でも、変更できないオブジェクトをブランチバケットに追加できます。これらのオブジェクトやブランチバケット自体を削除できない場合があります。

2. 「S3 Object Lock を有効にする」を選択した場合は、必要に応じてブランチバケットの「デフォルトの保持期間」を有効にしてください。

   > グリッド管理者から"S3オブジェクトロックの特定の機能を使用する"の権限を付与してもらう必要があります。

   *デフォルト保持*が有効になっている場合、ブランチバケットに追加された新しいオブジェクトは、削除または上書きされないように自動的に保護されます。*デフォルト保持*の設定は、独自の保持期間が設定されているオブジェクトには適用されません。

   a. *デフォルト保持*が有効になっている場合は、ブランチバケットの*デフォルト保持モード*を指定してください。

デフォルトの保持モード	説明
ガバナンス	• `s3:BypassGovernanceRetention`権限を持つユーザーは、`x-amz-bypass-governance-retention: true`リクエストヘッダーを使用して保持設定をバイパスできます。 • これらのユーザーは、オブジェクトのバージョンが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。
コンプライアンス	• オブジェクトは、保持期限が到来するまで削除できません。 • オブジェクトの保持期限は延長できますが、短縮することはできません。 • オブジェクトの保持期限日は、その日付が到来するまで削除することはできません。 注：グリッド管理者がコンプライアンスモードの使用を許可している必要があります。

   b. *デフォルト保持*が有効になっている場合は、ブランチバケットの*デフォルト保持期間*を指定してください。

*デフォルトの保持期間*は、ブランチバケットに追加された新しいオブジェクトを、取り込まれた時点からどのくらいの期間保持するかを示します。グリッド管理者が設定したテナントの最大保持期間以下の値を指定してください。

_最大_保持期限は、グリッド管理者がテナントを作成する際に設定されます。この値は1日から100年までの範囲で設定可能です。_デフォルト_の保持期限を設定する場合、最大保持期限に設定された値を超えることはできません。必要に応じて、グリッド管理者に最大保持期限の延長または短縮を依頼してください。

3. 必要に応じて、*容量制限を有効にする*を選択します。

   容量制限とは、ブランチバケットで使用可能な最大容量のことです。この値は論理的な量（オブジェクトのサイズ）を表しており、物理的な量（ディスク上のサイズ）を表すものではありません。

   制限が設定されていない場合、ブランチバケットの容量は無制限です。詳細については、"容量制限の使用状況"を参照してください。

   > この設定は、ブランチバケットに直接取り込まれたオブジェクトにのみ適用され、ベースバケットからブランチバケットを通して見えるオブジェクトには適用されません。

4. 必要に応じて、*オブジェクト数の制限を有効にする*を選択します。

   オブジェクト数の制限とは、ブランチバケットに格納できるオブジェクトの最大数です。この値は論理的な量（オブジェクト数）を表します。制限が設定されていない場合、オブジェクトの数は無制限です。

   > この設定は、ブランチバケットに直接取り込まれたオブジェクトにのみ適用され、ベースバケットからブランチバケットを通して見えるオブジェクトには適用されません。

5. 「バケットの作成」を選択します。

   ブランチバケットが作成され、バケットページのテーブルに追加されます。

6. オプションで「バケットの詳細ページへ移動」を選択すると、"ブランチバケットの詳細を表示"追加の設定を行うことができます。

   バケットの詳細ページでは、読み取り専用バケットの場合、オブジェクトの変更に関連する一部の設定オプションが無効になっています。

StorageGRID バケットに **ILM** ポリシータグを適用する

オブジェクトストレージの要件に基づいて、バケットに適用する ILM ポリシータグを選択してください。

ILMポリシーは、オブジェクトデータの保存場所と、一定期間後に削除するかどうかを制御します。グリッド管理者は、複数のアクティブなポリシーを使用する場合、ILMポリシーを作成し、それらをILMポリシータグに割り当てます。

> バケットのポリシータグを頻繁に再割り当てすることは避けてください。そうしないと、パフォーマンスの問題が発生する可能性があります。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"ルートアクセス、すべてのバケットの管理、またはすべてのバケットの表示権限" を持つユーザーグループに属しています。これらの権限は、グループポリシーまたはバケットポリシーの権限設定を上書きします。

手順

1. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3)** > **Buckets** を選択してください。

   「バケット」ページが表示されます。必要に応じて、任意の列で情報を並べ替えたり、リストを前後に移動したりできます。

2. ILMポリシータグを割り当てるバケットの名前を選択します。

   既にタグが割り当てられているバケットに対しても、ILMポリシーのタグ割り当てを変更することができます。

   表示されているオブジェクト数と使用容量の値は推定値です。これらの推定値は、データ取り込みのタイミング、ネットワーク接続性、およびノードの状態によって影響を受けます。バケットでバージョン管理が有効になっている場合、削除されたオブジェクトのバージョンもオブジェクト数に含まれます。

3. バケットオプションタブで、ILMポリシータグのアコーディオンを展開します。このアコーディオンは、グリッド管理者がカスタムポリシータグの使用を有効にしている場合にのみ表示されます。
4. 各ポリシータグの説明を読んで、どのタグをバケットに適用すべきかを判断してください。

   バケットのILMポリシータグを変更すると、そのバケット内のすべてのオブジェクトに対してILMの再評価が実行されます。新しいポリシーでオブジェクトの保持期間が限定されている場合、古いオブジェクトは削除されます。

5. バケットに割り当てるタグのラジオボタンを選択します。
6. 「変更を保存」を選択します。キー `NTAP-SG-ILM-BUCKET-TAG`とILMポリシータグ名の値を持つ新しいS3バケットタグがバケットに設定されます。

   S3アプリケーションが誤って新しいバケットタグを上書きしたり削除したりしないようにしてください。バケットに新しいTagSetを適用する際にこのタグを省略した場合、バケット内のオブジェクトはデフォルトのILMポリシーに基づいて評価されるようになります。

   ILMポリシータグの設定および変更は、ILMポリシータグが検証されるテナントマネージャーまたはテナントマネージャーAPIのみを使用して行ってください。S3 PutBucketTagging APIまたはS3 DeleteBucketTagging APIを使用して `NTAP-SG-ILM-BUCKET-TAG` ILMポリシータグを変更しないでください。

   バケットに割り当てられたポリシータグを変更すると、新しいILMポリシーを使用してオブジェクトが再評価される間、一時的にパフォーマンスに影響が出ます。

StorageGRIDバケットポリシーを管理

S3バケットとそのバケット内のオブジェクトに対するユーザーアクセスを制御できます。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"ルートアクセス権限"を持つユーザーグループに属しています。「すべてのバケットを表示」および「すべてのバケットを管理」の権限では、表示のみが許可されます。
- 必要な数のストレージノードとサイトが利用可能であることを確認しました。いずれかのサイト内で2つ以上のストレージノードが利用できない場合、またはサイト自体が利用できない場合、これらの設定を変更できない可能性があります。

手順

1. 「Buckets」を選択し、管理するバケットを選択します。
2. バケットの詳細ページで、「Bucket access」＞「Bucket policy」を選択します。
3. 次のいずれかを実行します。
    - **Enable policy** チェックボックスを選択して、バケットポリシーを入力します。次に、有効な JSON 形式の文字列を入力します。

      各バケットポリシーには、20,480バイトのサイズ制限があります。
    - 文字列を編集することで、既存のポリシーを変更します。
    - ポリシーを有効にする の選択を解除すると、ポリシーが無効になります。

バケットポリシーに関する詳細情報（言語構文や例を含む）については、"バケットポリシーの例"を参照してください。

StorageGRIDバケットの整合性を管理

整合性値を使用すると、バケット設定変更の可用性を指定したり、バケット内のオブジェクトの可用性と、異なるストレージノードやサイト間でのオブジェクトの整合性とのバランスを取ったりすることができます。クライアントアプリケーションが運用上のニーズを満たすことができるように、整合性値をデフォルト値とは異なる値に変更できます。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"すべてのバケットを管理する、またはルートアクセス権限"を持つユーザーグループに属しています。これらの権限は、グループポリシーまたはバケットポリシーの権限設定を上書きします。

バケットの整合性に関するガイドライン

バケットの整合性は、そのS3バケット内のオブジェクトに影響を与えるクライアントアプリケーションの整合性を判断するために使用されます。一般的に、バケットには **Read-after-new-write** 整合性を使用することをお勧めします。

バケットの整合性の変更

Read-after-new-write の一貫性がクライアントアプリケーションの要件を満たさない場合は、バケットの一貫性を設定するか、 `Consistency-Control`ヘッダーを使用することで一貫性を変更できます。 `Consistency-Control`ヘッダーはバケットの一貫性を上書きします。

> バケットの整合性を変更した場合、変更後に取り込まれたオブジェクトのみが、改訂された設定を満たすことが保証されます。

手順

1. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3)** > **Buckets** を選択してください。
2. 表からバケット名を選択してください。

   バケットの詳細ページが表示されます。
3. 「バケットオプション」タブから、**アコーディオンを選択します。
4. このバケット内のオブジェクトに対して実行される操作の一貫性を選択してください。
   - すべて: 最高レベルの一貫性を提供します。すべてのノードがデータを即座に受信するか、リクエストは失敗します。
   - **Strong-global**：すべてのサイトにおけるすべてのクライアントリクエストに対して、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リクエストに対して、書き込み後の読み取りの一貫性を保証します。
   - **Read-after-new-write**（デフォルト）：新規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、オブジェクト更新に対しては結果整合性を提供します。高可用性とデータ保護を保証します。ほとんどの場合に推奨されます。
   - 利用可能：新規オブジェクトとオブジェクト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、または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。
5. 変更を保存 を選択します。

バケットの設定を変更するとどうなりますか？

バケットには、バケット自体の動作やバケット内のオブジェクトの動作に影響を与える複数の設定があります。

以下のバケット設定では、デフォルトで*強力な*一貫性が使用されます。いずれかのサイト内で2つ以上のストレージノードが利用できない場合、またはサイト自体が利用できない場合、これらの設定への変更が反映されない可能性があります。

- "バックグラウンドでの空バケットの削除"
- "最終アクセス時間"
- "バケットのライフサイクル"
- "バケットポリシー"

- "バケットタグ付け"
- "バケットのバージョン管理"
- "S3 Object Lock"
- "バケット暗号化"

> バケットのバージョン管理、S3 Object Lock、およびバケット暗号化の整合性値は、強整合性でない値に設定することはできません。

以下のバケット設定では、強力な一貫性は使用されず、変更に対する可用性が高くなります。これらの設定を変更しても、反映されるまでには時間がかかる場合があります。

- "プラットフォームサービス構成：通知、レプリケーション、または検索の統合"
- "バケットとオブジェクトの StorageGRID CORS を設定する"
- バケットの整合性を変更する

> バケット設定を変更する際に使用されるデフォルトの整合性がクライアントアプリケーションの要件を満たさない場合は、 `Consistency-Control`ヘッダーを"S3 REST API"に使用するか、 `reducedConsistency`または `force`オプションを"テナント管理API"で使用することで整合性を変更できます。

StorageGRIDで最終アクセス時刻の更新を有効または無効にする

グリッド管理者が StorageGRID システムの情報ライフサイクル管理（ILM）ルールを作成する際、オブジェクトの最終アクセス時刻を使用して、そのオブジェクトを別のストレージ場所に移動するかどうかを判断するようにオプションで指定できます。S3 テナントを使用している場合は、S3 バケット内のオブジェクトの最終アクセス時刻の更新を有効にすることで、これらのルールを活用できます。

これらの手順は、*最終アクセス時刻*オプションを高度なフィルタまたは参照時刻として使用するILMルールを少なくとも1つ含むStorageGRIDシステムにのみ適用されます。StorageGRIDシステムにそのようなルールが含まれていない場合は、これらの手順を無視してください。詳細については、"ILMルールで最終アクセス時間を使用する"を参照してください。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"すべてのバケットを管理する、またはルートアクセス権限"を持つユーザーグループに属しています。これらの権限は、グループポリシーまたはバケットポリシーの権限設定を上書きします。

タスク概要

*最終アクセス時刻*は、ILMルールの*参照時刻*配置指示で使用できるオプションの1つです。ルールの参照時刻を最終アクセス時刻に設定すると、グリッド管理者は、オブジェクトが最後に取得（読み取りまたは表示）された時刻に基づいて、オブジェクトを特定のストレージの場所に配置するよう指定できます。

例えば、最近閲覧したオブジェクトがより高速なストレージに保持されるようにするには、グリッド管理者は次のようなILMルールを作成できます：

- 過去1か月以内に取得されたオブジェクトは、ローカルストレージノード上に保持されるべきです。

- 過去 1 か月以内に取得されなかったオブジェクトは、オフサイトの場所に移動する必要があります。

デフォルトでは、最終アクセス時刻の更新は無効になっています。StorageGRID システムに 最終アクセス時刻 オプションを使用する ILM ルールが含まれており、このオプションをこのバケット内のオブジェクトに適用する場合は、そのルールで指定されている S3 バケットの最終アクセス時刻の更新を有効にする必要があります。

> オブジェクトが取得されたときに最終アクセス時刻を更新すると、特に小さいオブジェクトの場合、StorageGRID のパフォーマンスが低下する可能性があります。

最終アクセス時刻の更新によりパフォーマンスに影響が生じます。これは、オブジェクトを取得するたびに、StorageGRID が以下の追加手順を実行する必要があるためです。

- オブジェクトを新しいタイムスタンプで更新する
- オブジェクトをILMキューに追加して、現在のILMルールとポリシーに基づいて再評価できるようにします。

この表は、最終アクセス時刻が無効または有効になっている場合に、バケット内のすべてのオブジェクトに適用される動作をまとめたものです。

リクエストの種類	最終アクセス時刻が無効になっている場合の動作（デフォルト）		最終アクセス時刻が有効になっている場合の動作	
	最終アクセス時刻は更新されましたか？	オブジェクトがILM評価キューに追加されましたか？	最終アクセス時刻は更新されましたか？	オブジェクトがILM評価キューに追加されましたか？
HEAD操作が発行された際に、オブジェクトのメタデータを取得する要求	いいえ	いいえ	いいえ	いいえ
オブジェクト、そのアクセス制御リスト、またはそのメタデータを取得するリクエスト	いいえ	いいえ	はい	はい
オブジェクトのメタデータを更新するリクエスト	はい	はい	はい	はい
オブジェクトまたはオブジェクトバージョンの一覧表示リクエスト	いいえ	いいえ	いいえ	いいえ

あるバケットから別のバケットへオブジェクトをコピーするリクエスト	• いいえ、ソースコピーについては • はい、宛先コピーの場合	• いいえ、ソースコピーについては • はい、宛先コピーの場合	• はい、ソースコピーの場合 • はい、宛先コピーの場合	• はい、ソースコピーの場合 • はい、宛先コピーの場合
複数パートのアップロードを完了するようリクエストします	はい、組み立てられたオブジェクトの場合	はい、組み立てられたオブジェクトの場合	はい、組み立てられたオブジェクトの場合	はい、組み立てられたオブジェクトの場合

手順

1. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3)** > **Buckets** を選択してください。
2. 表からバケット名を選択してください。

   バケットの詳細ページが表示されます。
3. *バケットオプション*タブから、*最終アクセス時刻の更新*アコーディオンを選択します。
4. 最終アクセス時刻の更新を有効または無効にします。
5. 変更を保存 を選択します。

StorageGRID の **S3** バケットのオブジェクトバージョニングを変更する

S3テナントを使用している場合は、S3バケットのバージョン管理状態を変更できます。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"すべてのバケットを管理する、またはルートアクセス権限"を持つユーザーグループに属しています。これらの権限は、グループポリシーまたはバケットポリシーの権限設定を上書きします。
- 必要な数のストレージノードとサイトが利用可能であることを確認しました。いずれかのサイト内で2つ以上のストレージノードが利用できない場合、またはサイト自体が利用できない場合、これらの設定を変更できない可能性があります。

タスク概要

バケットのオブジェクトバージョン管理を有効化または一時停止できます。バケットのバージョン管理を有効にすると、バージョン管理されていない状態に戻すことはできません。ただし、バケットのバージョン管理を一時停止することは可能です。

- 無効：バージョン管理は一度も有効になっていません
- 有効：バージョン管理が有効になっています
- 一時停止：バージョン管理は以前は有効でしたが、現在は一時停止されています

詳細については、次を参照してください。

- "オブジェクトのバージョン管理"

- "S3バージョン管理対象オブジェクトに関するILMルールとポリシー（例4）"
- "オブジェクトが削除される方法"

手順

1. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3)** > **Buckets** を選択してください。
2. 表からバケット名を選択してください。

   バケットの詳細ページが表示されます。

3. 「バケットオプション」タブから、「オブジェクトバージョン管理」のアコーディオンを選択します。
4. このバケット内のオブジェクトのバージョン管理状態を選択してください。

   クロスグリッドレプリケーションに使用されるバケットでは、オブジェクトのバージョン管理を有効にしておく必要があります。S3オブジェクトロックまたはレガシーコンプライアンスが有効になっている場合、*オブジェクトバージョニング*オプションは無効になります。

オプション	説明
バージョン管理を有効にする	このバケットに各オブジェクトのすべてのバージョンを保存する場合は、オブジェクトのバージョン管理を有効にしてください。必要に応じて、オブジェクトの以前のバージョンを取得できます。 バケット内に既に存在するオブジェクトは、ユーザーによって変更されるとバージョン管理されます。
バージョン管理を一時停止する	新しいオブジェクトバージョンの作成を不要とする場合は、オブジェクトのバージョン管理を一時停止してください。既存のオブジェクトバージョンは引き続き取得できます。

5. 変更を保存 を選択します。

S3オブジェクトロックを使用して**StorageGRID**でオブジェクトを保持します

バケットとオブジェクトが保持に関する規制要件を遵守する必要がある場合は、S3 Object Lockを使用できます。

S3 Object Lockの特定の機能を使用するには、グリッド管理者の許可が必要です。

S3オブジェクトロックとは何ですか？

StorageGRID S3 Object Lock機能は、Amazon Simple Storage Service（Amazon S3）のS3 Object Lockと同等のオブジェクト保護ソリューションです。

グローバル S3 オブジェクトロック設定が StorageGRID システムで有効になっている場合、S3 テナントアカウントは S3 オブジェクトロックを有効にするかどうかにかかわらず、バケットを作成できます。バケットで S3 オブジェクトロックが有効になっている場合、バケットのバージョン管理が必要となり、自動的に有効になります。

S3オブジェクトロックが設定されていないバケットには、保持設定が指定されていないオブジェクトのみを含めることができます。取り込まれたオブジェクトには、保持設定は適用されません。

S3オブジェクトロックが有効になっているバケットには、S3クライアントアプリケーションによって指定された保持設定のあるオブジェクトと、保持設定のないオブジェクトの両方を含めることができます。取り込まれたオブジェクトの中には、保持設定が指定されているものがあります。

S3 Object Lockとデフォルトの保持期間が設定されているバケットには、保持設定が指定されたアップロード済みオブジェクトと、保持設定のない新規オブジェクトが存在する可能性があります。新規オブジェクトはデフォルト設定を使用します。これは、オブジェクトレベルで保持設定が構成されていないためです。

デフォルトの保持設定が有効になっている場合、新しく取り込まれたすべてのオブジェクトには、実質的に保持設定が適用されます。オブジェクト保持設定が適用されていない既存のオブジェクトは影響を受けません。

保持モード

StorageGRID S3オブジェクトロック機能は、オブジェクトに対して異なるレベルの保護を適用するための2つの保持モードをサポートしています。これらのモードは、Amazon S3の保持モードと同等です。

- コンプライアンスモードの場合：
  - オブジェクトは、保持期限が到来するまで削除できません。
  - オブジェクトの保持期限は延長できますが、短縮することはできません。
  - オブジェクトの保持期限日は、その日付が到来するまで削除することはできません。
- ガバナンスモードでは：
  - 特別な権限を持つユーザーは、リクエストにバイパスヘッダーを使用することで、特定の保持設定を変更できます。
  - これらのユーザーは、オブジェクトのバージョンが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。

オブジェクトバージョンの保持設定

S3オブジェクトロックを有効にしてバケットを作成する場合、ユーザーはS3クライアントアプリケーションを使用して、バケットに追加される各オブジェクトに対して、以下の保持設定を任意で指定できます。

- 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の保持期限が将来の日付の場合、オブジェクトは取得できますが、削除することはできません。
- 法的保留：オブジェクトの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即座にロックされます。例えば、調査や法的紛争に関連するオブジェクトに対して、法的保留を適用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なく、明示的に解除されるまで有効です。法的保留は、保管期限とは無関係です。

> ⓘ オブジェクトが法的保留状態にある場合、その保持モードに関係なく、誰もそのオブジェクトを削除することはできません。

オブジェクト設定の詳細については、"S3 REST APIを使用してS3オブジェクトロックを設定する"を参照してください。

バケットのデフォルトの保持設定

S3 Object Lockを有効にしてバケットを作成する場合、ユーザーはオプションでバケットの以下のデフォルト設定を指定できます。

- デフォルトの保持モード：コンプライアンスまたはガバナンスのいずれか。
- デフォルトの保持期間：このバケットに追加された新しいオブジェクトバージョンを、追加された日からどのくらいの期間保持するか。

デフォルトのバケット設定は、独自の保持設定を持たない新規オブジェクトにのみ適用されます。これらのデフォルト設定を追加または変更しても、既存のバケットオブジェクトには影響しません。

"S3バケットを作成する"および"S3オブジェクトロックのデフォルト保持期間を更新する"を参照してください。

S3オブジェクトロックタスク

グリッド管理者とテナントユーザー向けの以下のリストには、S3 Object Lock機能を使用するための概要タスクが含まれています。

グリッド管理者

- StorageGRID システム全体のグローバル S3 オブジェクトロック設定を有効にします。
- 情報ライフサイクル管理（ILM）ポリシーが_準拠_していること、つまり、"S3オブジェクトロックが有効になっているバケットの要件"を満たしていることを確認してください。
- 必要に応じて、テナントが保持モードとして Compliance を使用できるようにします。それ以外の場合は、Governance モードのみが許可されます。
- 必要に応じて、テナントの最大保持期間を設定します。

テナントユーザー

- S3 Object Lockを使用するバケットとオブジェクトに関する考慮事項を確認してください。
- 必要に応じて、グリッド管理者に連絡して、グローバル S3 オブジェクトロック設定を有効にし、権限を設定してください。
- S3オブジェクトロックを有効にしたバケットを作成します。
- 必要に応じて、バケットのデフォルトの保持設定を構成します。
  - デフォルトの保持モード：グリッド管理者が許可した場合、Governance または Compliance。
  - デフォルトの保持期間：グリッド管理者が設定した最大保持期間以下である必要があります。
- S3クライアントアプリケーションを使用してオブジェクトを追加し、必要に応じてオブジェクト固有の保持期間を設定します。
  - 保持モード。グリッド管理者が許可する場合、ガバナンスまたはコンプライアンス。
  - 保持期限日：グリッド管理者が設定した最大保持期間以下である必要があります。

S3オブジェクトロックが有効になっているバケットの要件

- 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、テナントマネージャー、テナント管理 API、または S3 REST API を使用して、S3 オブジェクトロックを有効にしたバケットを作成できます。

- S3オブジェクトロックを使用する場合は、バケットを作成する際にS3オブジェクトロックを有効にする必要があります。既存のバケットに対してS3オブジェクトロックを有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が有効になっている場合、StorageGRID はそのバケットのバージョン管理を自動的に有効にします。S3 オブジェクトロックを無効にしたり、バケットのバージョン管理を一時停止したりすることはできません。
- オプションとして、テナントマネージャー、テナント管理API、またはS3 REST APIを使用して、各バケットのデフォルトの保持モードと保持期間を指定できます。バケットのデフォルトの保持設定は、独自の保持設定を持たない、バケットに追加された新しいオブジェクトにのみ適用されます。オブジェクトのバージョンごとに、アップロード時に保持モードと保持期限を指定することで、これらのデフォルト設定を上書きできます。
- S3 Object Lockが有効になっているバケットでは、バケットのライフサイクル設定がサポートされています。
- CloudMirror レプリケーションは、S3 オブジェクトロックが有効になっているバケットではサポートされていません。

S3オブジェクトロックが有効になっているバケット内のオブジェクトの要件

- オブジェクトのバージョンを保護するには、バケットのデフォルトの保持設定を指定するか、オブジェクトのバージョンごとに保持設定を指定できます。オブジェクトレベルの保持設定は、S3クライアントアプリケーションまたはS3 REST APIを使用して指定できます。
- 保持設定は、個々のオブジェクトバージョンに適用されます。オブジェクトのバージョンには、保持期限と法的保留設定の両方を設定できる場合もあれば、どちらか一方のみを設定できる場合、あるいはどちらも設定できない場合もあります。オブジェクトに対して保持期限または法的保留設定を指定すると、リクエストで指定されたバージョンのみが保護されます。オブジェクトの新しいバージョンを作成できますが、以前のバージョンのオブジェクトはロックされたままです。

S3オブジェクトロックが有効になっているバケット内のオブジェクトのライフサイクル

S3 Object Lockが有効になっているバケットに保存された各オブジェクトは、以下の段階を経ます：

1. オブジェクトの取り込み

   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バージョンが追加されると、保持設定は次のように適用されます：

   - オブジェクトに対して保持設定が指定されている場合、オブジェクトレベルの設定が適用されます。デフォルトのバケット設定はすべて無視されます。
   - オブジェクトに対して保持設定が指定されていない場合、デフォルトのバケット設定が存在すれば、それが適用されます。
   - オブジェクトまたはバケットに対して保持設定が指定されていない場合、そのオブジェクトはS3 Object Lockによって保護されません。

   保持設定が適用されている場合、オブジェクトと S3 ユーザー定義メタデータの両方が保護されます。

2. オブジェクトの保持と削除

   保護対象オブジェクトの複数のコピーが、指定された保存期間の間、StorageGRID によって保存されます。オブジェクトのコピーの正確な数と種類、および保存場所は、有効な ILM ポリシーの準拠ルールによって決定されます。保護対象オブジェクトが保持期限日前に削除できるかどうかは、その保持モードによって異なります。

- オブジェクトが法的保留状態にある場合、その保持モードに関係なく、誰もそのオブジェクトを削除することはできません。

従来の準拠バケットは引き続き管理できますか？

S3オブジェクトロック機能は、以前のStorageGRIDバージョンで利用可能だったコンプライアンス機能に取って代わるものです。以前のバージョンのStorageGRIDを使用して準拠バケットを作成した場合、これらのバケットの設定は引き続き管理できますが、新しい準拠バケットを作成することはできなくなります。手順については、https://kb.netapp.com/Advice_and_Troubleshooting/Hybrid_Cloud_Infrastructure/StorageGRID/How_to_manage_legacy_Compliant_buckets_in_StorageGRID_11.5["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"^]を参照してください。

StorageGRIDでS3オブジェクトロックのデフォルトの保持期間を更新

バケット作成時にS3オブジェクトロックを有効にした場合、バケットを編集してデフォルトの保持設定を変更できます。デフォルトの保持を有効（または無効）にしたり、デフォルトの保持モードと保持期間を設定したりできます。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"すべてのバケットを管理する、またはルートアクセス権限"を持つユーザーグループに属しています。これらの権限は、グループポリシーまたはバケットポリシーの権限設定を上書きします。
- S3 オブジェクトロックはStorageGRIDシステムに対してグローバルに有効化されており、バケットの作成時にS3 オブジェクトロックを有効にしました。"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"を参照してください。

手順

1. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3)** > **Buckets** を選択してください。
2. 表からバケット名を選択してください。

   バケットの詳細ページが表示されます。

3. 「Bucket options」タブから、「S3 Object Lock」のアコーディオンを選択します。
4. 必要に応じて、このバケットの*デフォルトの保持期間*を有効または無効にします。

   この設定の変更は、既にバケット内に存在するオブジェクト、または独自の保持期間が設定されているオブジェクトには適用されません。

5. *デフォルト保持*が有効になっている場合は、バケットの*デフォルト保持モード*を指定してください。

デフォルトの保持モード	説明
ガバナンス	• `s3:BypassGovernanceRetention`権限を持つユーザーは、 `x-amz-bypass-governance-retention: true`リクエストヘッダーを使用して保持設定をバイパスできます。 • これらのユーザーは、オブジェクトのバージョンが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。
コンプライアンス	• オブジェクトは、保持期限が到来するまで削除できません。 • オブジェクトの保持期限は延長できますが、短縮することはできません。 • オブジェクトの保持期限日は、その日付が到来するまで削除することはできません。 注：グリッド管理者がコンプライアンスモードの使用を許可している必要があります。

6. *デフォルト保持*が有効になっている場合は、バケットの*デフォルト保持期間*を指定してください。

   *デフォルトの保持期間*は、このバケットに追加された新しいオブジェクトが、取り込まれた時点からどのくらいの期間保持されるかを示します。グリッド管理者が設定したテナントの最大保持期間以下の値を指定してください。

   _最大_保持期限は、グリッド管理者がテナントを作成する際に設定されます。この値は1日から100年までの範囲で設定可能です。_デフォルト_の保持期限を設定する場合、最大保持期限に設定された値を超えることはできません。必要に応じて、グリッド管理者に最大保持期限の延長または短縮を依頼してください。

7. 変更を保存 を選択します。

StorageGRIDで**S3**バケットの**CORS**を設定する

S3バケットとそのバケット内のオブジェクトを他のドメインのWebアプリケーションからアクセス可能にしたい場合は、S3バケットに対してクロスオリジンリソース共有（CORS）を設定できます。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- GET CORS 設定リクエストの場合、あなたは以下の権限を持つユーザーグループに属しています。"すべてのバケットを管理する、またはすべてのバケットを表示する権限"これらの権限は、グループポリシーまたはバケットポリシーの権限設定を上書きします。
- PUT CORS 設定リクエストの場合、あなたは"すべてのバケットの権限を管理する"を持つユーザーグループに属しています。この権限は、グループポリシーまたはバケットポリシーの権限設定を上書きします。
- "ルートアクセス権限"は、すべてのCORS設定リクエストへのアクセスを提供します。

タスク概要

クロスオリジンリソース共有（CORS）は、あるドメインのクライアントWebアプリケーションが別のドメイン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、グラフィックを保存するために `Images`という名前のS3バケットを使用するとします。 `Images`バケットにCORSを設定することで、そのバケット内の画像をウェブサイト `http://www.example.com`に表示できるようになります。

バケットの**CORS**を有効にする

手順

1. テキストエディタを使用して、必要なXMLを作成します。この例は、S3バケットでCORSを有効にするために使用されるXMLを示しています。具体的には：
   - 任意のドメインがバケットにGETリクエストを送信できるようにします
   - `http://www.example.com`ドメインにGET、POST、およびDELETEリクエストの送信のみを許可します
   - すべてのリクエストヘッダーが許可されます

   ```
   <CORSConfiguration
       xmlns="http://s3.amazonaws.com/doc/2020-10-22/">
       <CORSRule>
           <AllowedOrigin>*</AllowedOrigin>
           <AllowedMethod>GET</AllowedMethod>
           <AllowedHeader>*</AllowedHeader>
       </CORSRule>
       <CORSRule>
           <AllowedOrigin>http://www.example.com</AllowedOrigin>
           <AllowedMethod>GET</AllowedMethod>
           <AllowedMethod>POST</AllowedMethod>
           <AllowedMethod>DELETE</AllowedMethod>
           <AllowedHeader>*</AllowedHeader>
       </CORSRule>
   </CORSConfiguration>
   ```

   CORS設定XMLの詳細については、 "Amazon Web Services (AWS) ドキュメント：Amazon Simple Storage Service ユーザーガイド"を参照してください。

2. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3)** > **Buckets** を選択してください。
3. 表からバケット名を選択してください。

   バケットの詳細ページが表示されます。

4. 「Bucket access」タブから、「Cross-Origin Resource Sharing（CORS）」のアコーディオンを選択します。
5. 「CORSを有効にする」チェックボックスを選択します。
6. CORS設定XMLをテキストボックスに貼り付けてください。

7. 変更を保存 を選択します。

CORS設定を変更する

手順

1. テキストボックス内のCORS設定XMLを更新するか、*クリア*を選択して最初からやり直してください。
2. 変更を保存 を選択します。

CORS設定を無効にする

手順

1. **Enable CORS** チェックボックスをオフにします。
2. 変更を保存 を選択します。

関連情報

"管理インターフェイス用に StorageGRID CORS を設定する"

StorageGRID S3バケットからオブジェクトを削除する

テナントマネージャーを使用すると、1つまたは複数のバケット内のオブジェクトを削除できます。

考慮事項と要件

これらの手順を実行する前に、以下の点に注意してください。

- バケット内のオブジェクトを削除すると、StorageGRID は選択した各バケット内のすべてのオブジェクトとすべてのオブジェクトバージョンを、StorageGRID システム内のすべてのノードとサイトから完全に削除します。StorageGRID は、関連するオブジェクトメタデータもすべて削除します。この情報は復元できません。
- バケット内のすべてのオブジェクトを削除するには、オブジェクトの数、オブジェクトのコピー数、および同時実行操作の数に応じて、数分、数日、あるいは数週間かかる場合があります。
- バケットに"S3オブジェクトロックが有効になっています"がある場合、*オブジェクトの削除：読み取り専用*状態が_年間_続く可能性があります。

  S3 Object Lock を使用するバケットは、すべてのオブジェクトの保持期限が到達し、法的ホールドがすべて解除されるまで、*オブジェクトの削除中：読み取り専用*の状態が維持されます。

- オブジェクトが削除されている間、バケットの状態は*Deleting objects: read-only*となります。この状態では、バケットに新しいオブジェクトを追加することはできません。
- すべてのオブジェクトが削除されると、バケットは読み取り専用状態のままになります。以下のいずれかを実行できます。
  - バケットを書き込みモードに戻し、新しいオブジェクトに再利用する
  - バケットを削除します
  - バケットの名前を将来の使用のために確保するには、バケットを読み取り専用モードにしておきます。

- バケットでオブジェクトのバージョニングが有効になっている場合、StorageGRID 11.8以降で作成された削除マーカーは、バケット内のオブジェクトの削除操作を使用して削除できます。
- バケットでオブジェクトのバージョニングが有効になっている場合、オブジェクトの削除操作では、StorageGRID 11.7以前で作成された削除マーカーは削除されません。バケット内のオブジェクトの削除については、"S3 バージョン管理されたオブジェクトはどのように削除されるのか"を参照してください。
- "グリッド間複製"を使用する場合は、以下の点にご注意ください。
  - このオプションを使用しても、他のグリッド上のバケットからオブジェクトは削除されません。
  - ソースバケットでこのオプションを選択した場合、別のグリッド上の宛先バケットにオブジェクトを追加すると、「クロスグリッドレプリケーション失敗」アラートがトリガーされます。他のグリッドのバケットに誰もオブジェクトを追加しないことを保証できない場合は、すべてのバケットオブジェクトを削除する前に、そのバケットに対して"グリッド間レプリケーションを無効にする"を実行してください。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"ルートアクセス権限" を持つユーザーグループに属しています。この権限は、グループポリシーまたはバケットポリシーの権限設定を上書きします。

手順

1. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3)** > **Buckets** を選択してください。

   「バケット」ページが表示され、既存のすべての S3 バケットが表示されます。

2. 特定のバケットの **Actions** メニューまたは詳細ページを使用してください。

   [操作]メニュー

   a. オブジェクトを削除するバケットごとにチェックボックスを選択してください。

   b. **Actions** > **Delete objects in bucket** を選択します。

   詳細ページ

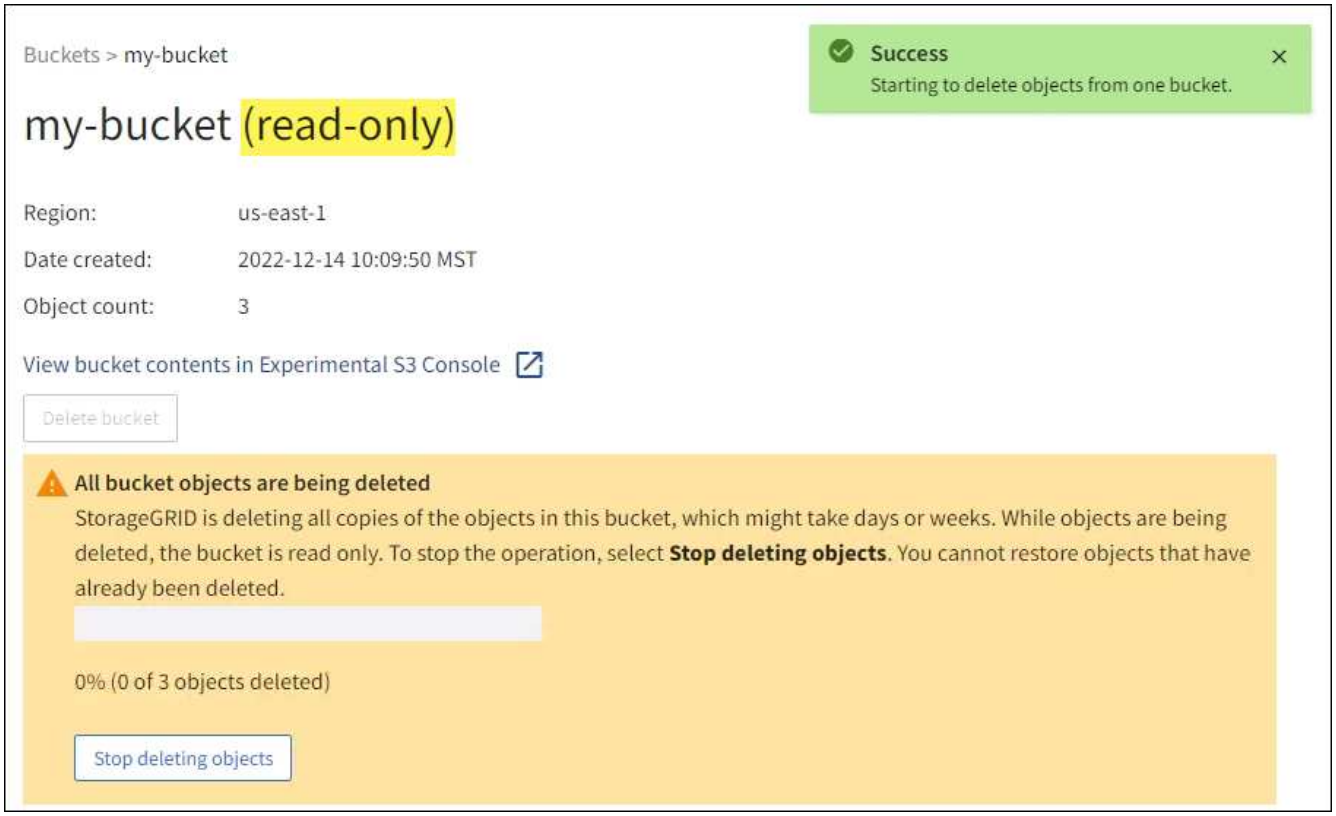
   a. 詳細を表示するには、バケット名を選択してください。

   b. 「バケット内のオブジェクトを削除」を選択します。

3. 確認ダイアログボックスが表示されたら、内容を確認し、**Yes** と入力して **OK** を選択してください。
4. 削除処理が開始されるまでお待ちください。

   数分後：

   - バケットの詳細ページに黄色のステータスバナーが表示されます。進行状況バーは、削除されたオブジェクトの割合を示しています。
   - バケット詳細ページでは、バケット名の後に **(read-only)** と表示されます。
   - バケットページでは、バケット名の横に*(オブジェクトの削除：読み取り専用)*と表示されます。

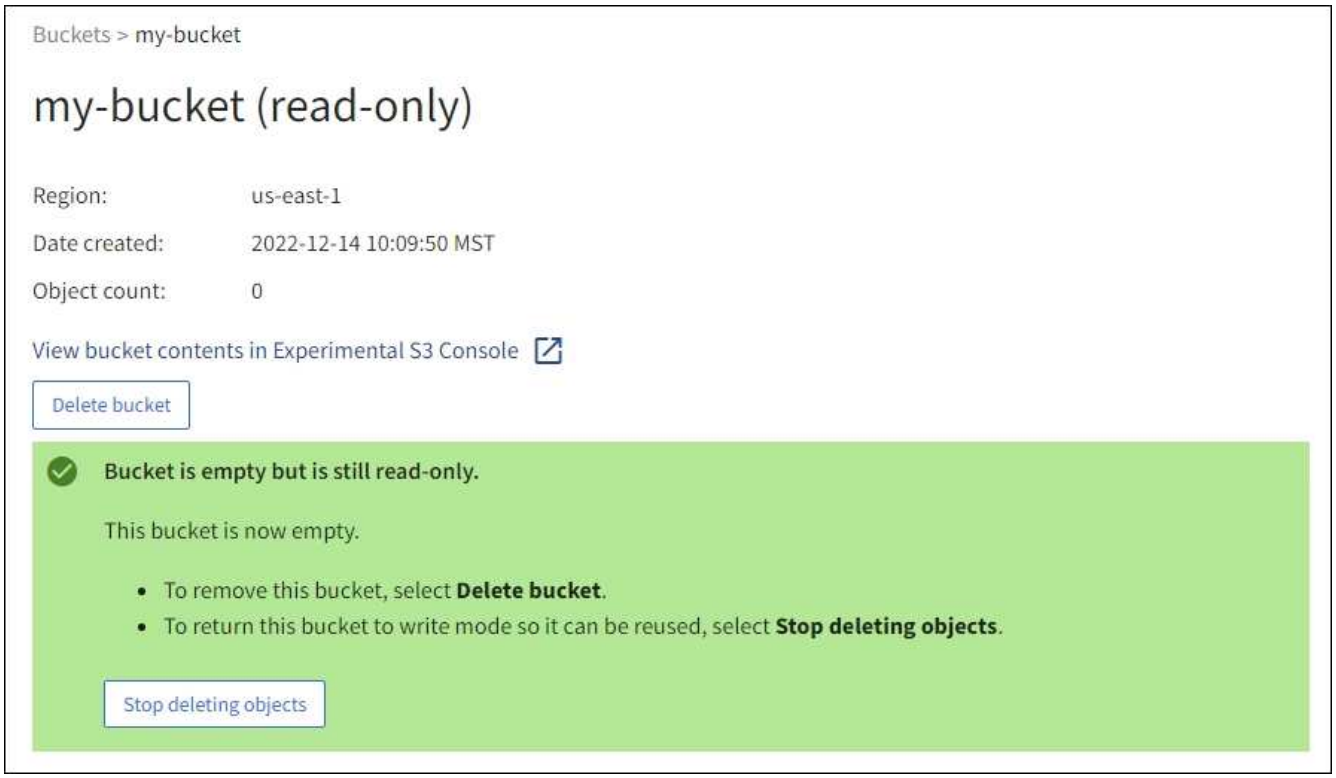


5. 操作の実行中に必要に応じて、オブジェクトの削除を停止 を選択して処理を停止してください。次に、必要に応じて バケット内のオブジェクトを削除 を選択して処理を再開します。

   「オブジェクトの削除を停止」を選択すると、バケットは書き込みモードに戻りますが、削除されたオブジェクトにアクセスしたり復元したりすることはできません。

6. 操作が完了するまでお待ちください。

   バケットが空になると、ステータスバナーは更新されますが、バケットは読み取り専用のままです。



7. 次のいずれかを実行します。
   - バケットを読み取り専用モードのままにするには、ページを閉じてください。例えば、空のバケットを読み取り専用モードで保持しておけば、バケット名を将来の使用のために確保しておくことができます。
   - バケットを削除します。単一のバケットを削除するには「バケットを削除」を選択し、複数のバケットを削除するには「バケット」ページに戻って「アクション」>「バケットを削除」を選択します。

     すべてのオブジェクトを削除した後でもバージョン管理されたバケットを削除できない場合、削除マーカーが残る可能性があります。バケットを削除するには、残っている削除マーカーをすべて削除する必要があります。

   - バケットを書き込みモードに戻し、必要に応じて新しいオブジェクトに再利用します。単一のバケットに対して「オブジェクトの削除を停止」を選択するか、バケットページに戻って「アクション」>「オブジェクトの削除を停止」を選択して複数のバケットに対して操作することができます。

StorageGRID S3 バケットを削除する

テナントマネージャーを使用すると、空になった S3 バケットを 1 つまたは複数削除できます。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"すべてのバケットを管理する、またはルートアクセス権限"を持つユーザーグループに属しています。これらの権限は、グループポリシーまたはバケットポリシーの権限設定を上書きします。
- 削除しようとしているバケットは空です。削除したいバケットが空でない場合、"バケットからオブジェクトを削除する"。

タスク概要

これらの手順では、テナントマネージャーを使用してS3バケットを削除する方法について説明します。S3バケットは、"テナント管理API"または"S3 REST API"を使用して削除することもできます。

S3バケットにオブジェクト、最新ではないオブジェクトバージョン、または削除マーカーが含まれている場合、そのバケットを削除することはできません。S3バージョン管理されたオブジェクトの削除方法については、"オブジェクトが削除される方法"を参照してください。

手順

1. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3)** > **Buckets** を選択してください。

   「バケット」ページが表示され、既存のすべての S3 バケットが表示されます。

2. 特定のバケットの **Actions** メニューまたは詳細ページを使用してください。

   [操作]メニュー

   a. 削除する各バケットのチェックボックスを選択します。

   b. **Actions** > **Delete buckets** を選択します。

   詳細ページ

   a. 詳細を表示するには、バケット名を選択してください。

   b. 「バケットを削除」を選択します。

3. 確認ダイアログボックスが表示されたら、**Yes** を選択してください。

   StorageGRID は各バケットが空であることを確認し、その後各バケットを削除します。この操作には数分かかる場合があります。

   バケットが空でない場合、エラーメッセージが表示されます。バケットを削除する前に、"バケット内のすべてのオブジェクトと削除マーカーを削除します"必要があります。

StorageGRIDで**S3**コンソールを使用する

S3コンソールを使用すると、S3バケット内のオブジェクトを表示および管理できます。

S3 Consoleでは以下のことが可能です：

- オブジェクトのアップロード、ダウンロード、名前変更、コピー、移動、削除
- オブジェクトのバージョンを表示、元に戻す、ダウンロード、削除する
- 接頭辞でオブジェクトを検索
- オブジェクトタグを管理する
- オブジェクトのメタデータを表示する
- フォルダの表示、作成、名前変更、コピー、移動、削除

S3 Consoleは、最も一般的なケースにおいて、ユーザーエクスペリエンスを向上させます。これは、あらゆる状況においてCLIやAPI操作を置き換えることを目的としたものではありません。

S3 Console を使用した結果、操作に時間がかかりすぎる場合（たとえば、数分または数時間）、以下を検討してください。

- 選択オブジェクトの数を減らす
- 非グラフィカルな方法（APIまたはCLI）を使用してデータにアクセスする

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- オブジェクトを管理するには、ルートアクセス権限を持つユーザーグループに属している必要があります。または、S3コンソールタブの使用権限と、すべてのバケットの表示権限またはすべてのバケットの管理権限のいずれかを持つユーザーグループに属している必要があります。参照してください"テナント管理権限"。
- ユーザーに対してS3グループまたはバケットポリシーが設定されています。"バケットおよびグループのアクセスポリシーを使用する"を参照してください。
- ユーザーのアクセスキーIDとシークレットアクセスキーを把握しています。オプションとして、この情報を含む `.csv`ファイルを用意します。"アクセスキーの作成手順"を参照してください。

手順

1. ストレージ＞バケット＞*バケット名*を選択します。
2. S3コンソールタブを選択します。
3. アクセスキーIDとシークレットアクセスキーをそれぞれのフィールドに貼り付けてください。それ以外の場合は、「アクセスキーをアップロード」を選択し、 `.csv`ファイルを選択してください。
4. 「サインイン」を選択します。
5. バケットオブジェクトのテーブルが表示されます。必要に応じてオブジェクトを管理できます。

その他の情報

- 接頭辞による検索：接頭辞検索機能は、現在のフォルダーに関連する特定の単語で始まるオブジェクトのみを検索します。検索対象には、他の箇所にその単語が含まれているオブジェクトは含まれません。このルールは、フォルダ内のオブジェクトにも適用されます。例えば、 `folder1/folder2/somefile-`を検索すると、 `folder1/folder2/`フォルダー内にあり、 `somefile-`という単語で始まるオブジェクトが返されます。
- ドラッグ＆ドロップ：コンピューターのファイルマネージャーから S3 Console にファイルをドラッグ＆ドロップできます。ただし、フォルダをアップロードすることはできません。
- フォルダに対する操作：フォルダを移動、コピー、または名前変更すると、フォルダ内のすべてのオブジェクトが一度に1つずつ更新されるため、時間がかかる場合があります。
- バケットのバージョン管理が無効になっている場合の完全削除：バージョン管理が無効になっているバケット内のオブジェクトを上書きまたは削除すると、その操作は永続的になります。参照してください "バケットのオブジェクトバージョン管理を変更する"。

S3プラットフォームサービスを管理する

S3プラットフォームサービス

StorageGRIDのプラットフォームサービスについて学ぶ

プラットフォームサービスを導入する前に、これらのサービスの利用に関する概要と留意事項を確認してください。

S3に関する情報については、"S3 REST APIを使用する"を参照してください。

プラットフォームサービスの概要

StorageGRID プラットフォームサービスは、イベント通知や S3 オブジェクトおよびオブジェクトメタデータのコピーを外部の宛先に送信できるようにすることで、ハイブリッドクラウド戦略の実装に役立ちます。

プラットフォームサービスのターゲットロケーションは通常、StorageGRID デプロイメントの外部にあるため、プラットフォームサービスを利用することで、外部ストレージリソース、通知サービス、およびデータに対する検索または分析サービスを活用する際のパワーと柔軟性を享受できます。

単一のS3バケットに対して、プラットフォームサービスの任意の組み合わせを設定できます。たとえば、StorageGRID S3バケットに"CloudMirrorサービス"と"通知"の両方を設定することで、特定のオブジェクトをAmazon Simple Storage Service（S3）にミラーリングしながら、AWSの費用を追跡するのに役立つサードパーティの監視アプリケーションに各オブジェクトの通知を送信できます。

> プラットフォームサービスの使用は、StorageGRID管理者がGrid ManagerまたはGrid Management APIを使用して、各テナントアカウントで有効にする必要があります。

プラットフォームサービスの構成方法

プラットフォームサービスは、"Tenant Manager"または"テナント管理API"を使用して設定した外部エンドポイントと通信します。各エンドポイントは、StorageGRID S3バケット、Amazon Web Servicesバケット、Amazon SNSトピック、Webhookエンドポイント、またはローカル、AWS、もしくはその他の場所でホストされているElasticsearchクラスターなどの外部宛先を表します。

外部エンドポイントを作成した後、バケットにXML構成を追加することで、バケットのプラットフォームサービスを有効にできます。XML設定では、バケットが操作する対象となるオブジェクト、バケットが実行すべきアクション、およびバケットがサービスに使用するエンドポイントが識別されます。

設定するプラットフォームサービスごとに、個別のXML設定を追加する必要があります。例：

- キーが `/images`で始まるすべてのオブジェクトを Amazon S3 バケットにレプリケートする場合は、ソースバケットにレプリケーション設定を追加する必要があります。
- これらのオブジェクトがバケットに保存されたときに通知も送信したい場合は、通知設定を追加する必要があります。
- これらのオブジェクトのメタデータをインデックス化する場合は、検索統合を実装するために使用されるメタデータ通知設定を追加する必要があります。

設定 XML の形式は、StorageGRID プラットフォームサービスの実装に使用される S3 REST API によって規定されます。

プラットフォームサービス	S3 REST API	参照先
CloudMirror レプリケーション	• GetBucketReplication • PutBucketReplication	• "CloudMirror レプリケーション" • "バケットに対する操作"
通知	• GetBucketNotificationConfiguration • PutBucketNotificationConfiguration	• "通知" • "バケットに対する操作"
検索統合	• バケットメタデータ通知設定を取得します • PUT Bucketメタデータ通知設定	• "検索統合" • "StorageGRID カスタムオペレーション"

プラットフォームサービスを利用する際の考慮事項

考慮事項	詳細
宛先エンドポイントの監視	各宛先エンドポイントの可用性を監視する必要があります。宛先エンドポイントへの接続が長時間失われ、大量のリクエストがバックログに存在する場合、StorageGRID への追加のクライアントリクエスト（PUT リクエストなど）は失敗します。エンドポイントにアクセス可能になったら、これらの失敗したリクエストを再試行する必要があります。
宛先エンドポイントのスロットリング	StorageGRID ソフトウェアは、S3 バケットへの受信リクエストの送信速度が、宛先エンドポイントがリクエストを受信できる速度を超えた場合、受信リクエストを制限する可能性があります。スロットリングは、宛先エンドポイントへの送信を待機しているリクエストのバックログが存在する場合にのみ発生します。 唯一目に見える影響は、受信する S3 リクエストの実行に時間がかかるようになることだけです。パフォーマンスが著しく低下し始めた場合は、取り込み速度を下げるか、より容量の大きいエンドポイントを使用してください。リクエストのバックログが増大し続けると、クライアントの S3 操作（PUT リクエストなど）は最終的に失敗します。 CloudMirror のリクエストは、検索統合やイベント通知のリクエストよりもデータ転送量が多いため、宛先エンドポイントのパフォーマンスの影響を受けやすい傾向があります。

考慮事項	詳細
順序保証	StorageGRID は、サイト内のオブジェクトに対する操作の順序を保証します。オブジェクトに対するすべての操作が同じサイト内で行われる限り、最終的なオブジェクト状態（レプリケーション用）は常に StorageGRID の状態と等しくなります。 StorageGRID は、StorageGRID サイト間で操作が行われる際に、リクエストを順序付けるために最善を尽くします。例えば、最初にサイト A にオブジェクトを書き込み、その後同じオブジェクトをサイト B で上書きした場合、CloudMirror によって宛先バケットにレプリケートされた最終的なオブジェクトが新しいオブジェクトであるとは限りません。
ILM駆動型オブジェクト削除	AWS CRR および Amazon Simple Notification Service の削除動作に合わせるため、StorageGRID ILM ルールによってソースバケット内のオブジェクトが削除された場合、CloudMirror およびイベント通知リクエストは送信されません。たとえば、ILM ルールによって 14 日後にオブジェクトが削除された場合、CloudMirror またはイベント通知リクエストは送信されません。 一方、検索統合リクエストは、ILM によってオブジェクトが削除された場合に送信されます。
Kafkaエンドポイントの使用	Kafkaエンドポイントでは、相互TLSはサポートされていません。そのため、Kafkaブローカーの設定で `ssl.client.auth`を `required`に設定している場合、Kafkaエンドポイントの設定に問題が発生する可能性があります。 Kafkaエンドポイントの認証には、以下の認証タイプが使用されます。これらの認証タイプは、Amazon SNSなどの他のエンドポイントの認証に使用されるものとは異なり、ユーザー名とパスワードの認証情報が必要です。 • SASL/PLAIN • SASL/SCRAM-SHA-256 • SASL/SCRAM-SHA-512 注: 設定されたストレージプロキシ設定は、Kafkaプラットフォームサービスのエンドポイントには適用されません。

CloudMirror レプリケーションサービスを使用する際の考慮事項

考慮事項	詳細
レプリケーションステータス	StorageGRID は `x-amz-replication-status`ヘッダーをサポートしていません。

考慮事項	詳細
オブジェクトサイズ	CloudMirror レプリケーションサービスによって宛先バケットにレプリケートできるオブジェクトの最大サイズは 5 TiB であり、これはサポートされる最大オブジェクトサイズと同じです。 注：単一のPutObject操作の最大推奨サイズは 5 GiB（5,368,709,120 バイト）です。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てください。
バケットのバージョン管理とバージョンID	StorageGRID のソース S3 バケットでバージョン管理が有効になっている場合は、宛先バケットでもバージョン管理を有効にする必要があります。 バージョン管理を使用する場合、S3プロトコルの制限により、宛先バケット内のオブジェクトバージョンの順序はベストエフォートであり、CloudMirror サービスによって保証されるものではないことに注意してください。 注: StorageGRID のソースバケットのバージョン ID は、宛先バケットのバージョン ID とは関係ありません。
オブジェクトバージョンのタグ付け	CloudMirrorサービスは、S3プロトコルの制限により、バージョンIDを指定するPutObjectTaggingまたはDeleteObjectTaggingリクエストを複製しません。ソースと宛先のバージョンIDは関連付けられていないため、特定のバージョンIDへのタグ更新が確実に複製されることを保証する方法はありません。 対照的に、CloudMirror サービスは、バージョン ID を指定しない PutObjectTagging リクエストまたは DeleteObjectTagging リクエストを複製します。これらのリクエストは、最新のキー（バケットがバージョン管理されている場合は最新バージョン）のタグを更新します。タグ付きの通常のデータ取り込み（タグ付けの更新を除く）も複製されます。
マルチパートアップロードと `ETag`値	マルチパートアップロードを使用してアップロードされたオブジェクトをミラーリングする場合、CloudMirrorサービスではパーツが保存されません。その結果、ミラーリングされたオブジェクトの `ETag`値は、元のオブジェクトの `ETag`値とは異なります。
SSE-C（顧客提供の鍵を使用したサーバー側暗号化）で暗号化されたオブジェクト	CloudMirrorサービスは、SSE-C で暗号化されたオブジェクトをサポートしていません。CloudMirrorレプリケーションのソースバケットにオブジェクトを取り込もうとした際に、リクエストに SSE-C リクエストヘッダーが含まれている場合、操作は失敗します。
S3オブジェクトロックが有効になっているバケット	S3オブジェクトロックが有効になっているソースバケットまたは宛先バケットでは、レプリケーションはサポートされていません。

StorageGRID CloudMirror レプリケーションサービスについて学ぶ

S3バケットに追加された指定オブジェクトをStorageGRIDが1つ以上の外部宛先バケットにレプリケートするようにする場合は、S3バケットのCloudMirrorレプリケーションを有効にすることができます。

例えば、CloudMirror レプリケーションを使用して特定の顧客レコードを Amazon S3 にミラーリングし、AWS サービスを活用してデータの分析を実行することができます。

CloudMirror レプリケーションは、ソースバケットで S3 オブジェクトロックが有効になっている場合はサポートされません。

CloudMirror と **ILM**

CloudMirrorレプリケーションは、グリッドのアクティブなILMポリシーとは独立して動作します。CloudMirrorサービスは、オブジェクトがソースバケットに保存されると同時に複製し、可能な限り速やかに宛先バケットに配信します。複製されたオブジェクトの配信は、オブジェクトの取り込みが成功したときに開始されます。

CloudMirrorおよびクロスグリッドレプリケーション

CloudMirrorレプリケーションには、クロスグリッドレプリケーション機能と重要な類似点と相違点があります。"グリッド間レプリケーションとCloudMirrorレプリケーションの比較"を参照してください。

CloudMirrorおよび**S3**バケット

CloudMirrorレプリケーションは通常、外部のS3バケットを宛先として使用するように構成されます。ただし、別の StorageGRID デプロイメントまたは任意のS3互換サービスを使用するようにレプリケーションを構成することもできます。

既存のバケット

既存のバケットに対して CloudMirror レプリケーションを有効にすると、そのバケットに追加された新しいオブジェクトのみがレプリケートされます。バケット内に既に存在するオブジェクトはレプリケートされません。既存のオブジェクトのレプリケーションを強制するには、オブジェクトのコピーを実行して、既存のオブジェクトのメタデータを更新します。

CloudMirror レプリケーションを使用してオブジェクトを Amazon S3 の宛先にコピーする場合は、Amazon S3 が各 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を 2 KB に制限していることに注意してください。オブジェクトに 2 KB を超えるユーザー定義メタデータが含まれている場合、そのオブジェクトはレプリケートされません。

複数の宛先バケット

単一のバケット内のオブジェクトを複数の宛先バケットに複製するには、レプリケーション設定XMLで各ルールの宛先を指定します。オブジェクトを複数のバケットに同時に複製することはできません。

バージョン管理されたバケットまたはバージョン管理されていないバケット

バージョン管理されたバケットまたはバージョン管理されていないバケットで CloudMirror レプリケーションを設定できます。宛先バケットは、バージョン管理されている場合も、バージョン管理されていない場合も構いません。バージョン管理されたバケットとバージョン管理されていないバケットは、任意の組み合わせで使用できます。例えば、バージョン管理されていないソースバケットの宛先としてバージョン管理されたバケットを指定したり、その逆も可能です。バージョン管理されていないバケット間でもレプリケーションを行うことができます。

削除、複製ループ、およびイベント

削除動作

Amazon S3 サービスのクロスリージョンレプリケーション（CRR）の削除動作と同じです。ソースバケット内のオブジェクトを削除しても、宛先バケット内の複製されたオブジェクトは削除されません。ソースバケットと宛先バケットの両方がバージョン管理されている場合、削除マーカーは複製されます。宛先バケットがバージョン管理されていない場合、ソースバケット内のオブジェクトを削除しても、削除マーカーは宛先バケットに複製されず、宛先オブジェクトも削除されません。

複製ループからの保護

オブジェクトが宛先バケットにレプリケートされると、StorageGRID はそれらを「レプリカ」としてマークします。宛先の StorageGRID バケットは、レプリカとしてマークされたオブジェクトを再度レプリケートしないため、意図しないレプリケーションループを防ぐことができます。このレプリカのマーキングは StorageGRID の内部処理であり、Amazon S3 バケットを宛先として使用する際に AWS CRR を活用することを妨げるものではありません。

レプリカをマークするために使用されるカスタムヘッダーは `x-ntap-sg-replica`です。このマーキングにより、連鎖的なミラーの発生を防ぎます。StorageGRID は、2つのグリッド間の双方向 CloudMirror をサポートしています。

宛先バケット内のイベント

宛先バケット内のイベントの一意性および順序は保証されません。配信の成功を保証するために行われる操作の結果として、ソースオブジェクトの同一のコピーが複数、宛先に配信される可能性があります。まれに、同じオブジェクトが 2 つ以上の異なる StorageGRID サイトから同時に更新される場合、宛先バケットでの操作の順序が、ソースバケットでのイベントの順序と一致しないことがあります。

S3バケットの**StorageGRID**イベント通知について学ぶ

S3バケットのイベント通知を有効にすると、StorageGRID が指定されたイベントに関する通知を、宛先の Kafka クラスター、Webhook エンドポイント、または Amazon Simple Notification Service に送信します。

例えば、バケットに追加された各オブジェクトについて、管理者にアラートを送信するように設定できます。ここでいうオブジェクトとは、重大なシステムイベントに関連付けられたログファイルを表します。

イベント通知は、通知設定で指定されたソースバケットで作成され、宛先に配信されます。オブジェクトに関連付けられたイベントが成功した場合、そのイベントに関する通知が作成され、配信待ちのキューに追加されます。

通知の一意性および順序は保証されません。配信の成功を保証するために講じられた措置の結果、イベントに関する通知が複数回宛先に配信される場合があります。また、配信は非同期であるため、特に異なる StorageGRID サイトから発生する操作の場合、宛先での通知の時間順序がソースバケット上のイベントの順序と一致することは保証されません。Amazon S3 のドキュメントに記載されているように、イベントメッセージの `sequencer`キーを使用して、特定のオブジェクトのイベントの順序を確認できます。

StorageGRID のイベント通知は、いくつかの制限はありますが、Amazon S3 API に準拠しています。

- 以下のイベントタイプがサポートされています：
  - s3:ObjectCreated:
  - s3:ObjectCreated:Put
  - s3:ObjectCreated:Post

- s3:ObjectCreated:Copy
- s3:ObjectCreated:CompleteMultipartUpload
- s3:ObjectRemoved:
- s3:ObjectRemoved:Delete
- s3:ObjectRemoved:DeleteMarkerCreated
- s3:ObjectRestore:Post

- StorageGRID から送信されるイベント通知は標準の JSON 形式を使用しますが、一部のキーは含まれず、他のキーには特定の値が使用されます。詳細は次の表を参照してください。

キー名	**StorageGRID** の価値
eventSource	`sgws:s3`
awsRegion	含まれていません
x-amz-id-2	含まれていません
arn	`urn:sgws:s3:::bucket_name`

StorageGRID検索統合サービスについて学ぶ

オブジェクトのメタデータに対して外部の検索およびデータ分析サービスを利用したい場合は、S3バケットの検索統合を有効にすることができます。

検索統合サービスは、オブジェクトが作成または削除されたとき、あるいはメタデータやタグが更新されたときに、S3オブジェクトのメタデータを自動的かつ非同期的に宛先エンドポイントに送信するカスタムのStorageGRIDサービスです。その後、宛先サービスが提供する高度な検索、データ分析、視覚化、または機械学習ツールを使用して、オブジェクトデータを検索・分析し、そこから知見を得ることができます。

例えば、バケットを設定して、S3オブジェクトのメタデータをリモートのElasticsearchサービスに送信するようにすることができます。その後、Elasticsearchを使用してバケット全体にわたる検索を実行したり、オブジェクトのメタデータに存在するパターンの高度な分析を実行したりすることができます。

ElasticsearchとのS3 Object Lockが有効になっているバケットで構成できますが、オブジェクトのS3 Object Lockメタデータ（Retain Until DateやLegal Hold statusなど）はElasticsearchに送信されるメタデータには含まれません。

> 検索統合サービスはオブジェクトのメタデータを宛先に送信するため、その構成XMLは「*metadata* 通知構成XML」と呼ばれます。この設定XMLは、イベント 通知を有効にするために使用される「通知設定XML」とは異なります。

検索統合と**S3**バケット

検索統合サービスは、バージョン管理されているバケットでも、バージョン管理されていないバケットでも有効にできます。検索統合は、メタデータ通知設定XMLを、操作対象となるオブジェクトとオブジェクトメタデータの送信先を指定するバケットに関連付けることによって構成されます。

メタデータ通知は、バケット名、オブジェクト名、およびバージョンID（存在する場合）を名前とするJSONドキュメントの形式で生成されます。各メタデータ通知には、オブジェクトのすべてのタグとユーザーメタデータに加えて、オブジェクトに関する標準的なシステムメタデータセットが含まれています。

> タグとユーザーメタデータについては、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。Elasticsearchが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設定するには、Elasticsearchの動的フィールドマッピングおよび日付形式のマッピングに関する手順に従ってください。検索統合サービスを設定する前に、インデックス上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。ドキュメントがインデックス化された後は、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。

検索通知

メタデータ通知は、以下の場合に生成され、配信待ちのキューに追加されます。

- オブジェクトが作成されます。
- オブジェクトが削除された場合（グリッドのILMポリシーの動作の結果としてオブジェクトが削除される場合を含む）。
- オブジェクトのメタデータまたはタグが追加、更新、または削除されます。更新時には、変更された値だけでなく、メタデータとタグの完全なセットが常に送信されます。

メタデータ通知設定XMLをバケットに追加すると、新しく作成したオブジェクト、およびデータ、ユーザーメタデータ、またはタグを更新して変更したオブジェクトに対して通知が送信されます。ただし、既にバケット内に存在するオブジェクトについては通知は送信されません。バケット内のすべてのオブジェクトのオブジェクトメタデータが宛先に確実に送信されるようにするには、次のいずれかの操作を行う必要があります：

- バケットを作成した後、オブジェクトを追加する前に、検索統合サービスを直ちに設定してください。
- バケット内に既に存在するすべてのオブジェクトに対して、メタデータ通知メッセージが宛先に送信されるような操作を実行します。

検索統合サービスと**Elasticsearch**

StorageGRID 検索統合サービスは、宛先として Elasticsearch クラスタをサポートしています。他のプラットフォームサービスと同様に、宛先はサービスの設定 XML で使用される URN を持つエンドポイントで指定されます。 "NetApp Interoperability Matrix Tool"を使用して、サポートされている Elasticsearch のバージョンを確認してください。

プラットフォームサービスのエンドポイントを管理する

StorageGRID でプラットフォームサービスエンドポイントを設定する

バケットのプラットフォームサービスを設定する前に、プラットフォームサービスの宛先となるエンドポイントを少なくとも1つ設定する必要があります。

プラットフォームサービスへのアクセスは、StorageGRID管理者によってテナントごとに有効化されます。プラットフォームサービスエンドポイントを作成または使用するには、ストレージノードが外部エンドポイントリソースにアクセスできるようにネットワークが構成されているグリッドにおいて、エンドポイントの管理権限またはルートアクセス権限を持つテナントユーザーである必要があります。単一テナントの場合、最大500個のプラットフォームサービスエンドポイントを設定できます。詳細については、StorageGRID管理者にお問い合わせください。

プラットフォームサービスのエンドポイントとは何ですか？

プラットフォームサービスエンドポイントは、StorageGRID が外部宛先にアクセスするために必要な情報を指定します。

たとえば、StorageGRID バケットから Amazon S3 バケットにオブジェクトをレプリケートする場合は、StorageGRID が Amazon 上の宛先バケットにアクセスするために必要な情報とクレデンシャルを含むプラットフォームサービスエンドポイントを作成します。

プラットフォームサービスの種類ごとに専用のエンドポイントが必要となるため、使用する予定のプラットフォームサービスごとに少なくとも1つのエンドポイントを設定する必要があります。プラットフォームサービスのエンドポイントを定義した後、そのエンドポイントのURNを、サービスを有効にするために使用する構成XMLの宛先として使用します。

複数のソースバケットに対して、同じエンドポイントを宛先として使用できます。例えば、複数のソースバケットからオブジェクトのメタデータを同じ検索統合エンドポイントに送信するように設定することで、複数のバケットにまたがる検索を実行できます。ソースバケットを複数のエンドポイントをターゲットとして使用するように構成することもできます。これにより、オブジェクトの作成に関する通知を1つの Amazon Simple Notification Service （Amazon SNS）トピックに送信し、オブジェクトの削除に関する通知を別の Amazon SNS トピックに送信するといったことが可能になります。

CloudMirror レプリケーションのエンドポイント

StorageGRID は、S3 バケットを表すレプリケーションエンドポイントをサポートします。これらのバケットは、Amazon Web Services、同じまたはリモートの StorageGRID デプロイメント、あるいは別のサービスでホストされる場合があります。

通知のエンドポイント

StorageGRIDは、Amazon SNS、Kafka、およびWebhookエンドポイントをサポートしています。Simple Queue Service (SQS) および AWS Lambda エンドポイントはサポートされていません。

Kafkaエンドポイントでは、相互TLSはサポートされていません。そのため、Kafkaブローカーの設定で`ssl.client.auth`を `required`に設定している場合、Kafkaエンドポイントの設定に問題が発生する可能性があります。

検索統合サービスのエンドポイント

StorageGRID は、Elasticsearch クラスタを表す検索統合エンドポイントをサポートします。これらの Elasticsearch クラスターは、ローカルデータセンターに設置することも、AWS クラウドやその他の場所にホストすることも可能です。

検索統合エンドポイントは、特定のElasticsearchインデックスとタイプを参照します。StorageGRIDでエンドポイントを作成する前に、Elasticsearchでインデックスを作成する必要があります。そうでなければ、エンドポイントの作成は失敗します。エンドポイントを作成する前に型を作成する必要はありません。StorageGRIDは、エンドポイントにオブジェクトメタデータを送信する際に、必要に応じて型を作成します。

関連情報

"StorageGRID の管理"

プラットフォームサービスのエンドポイントを作成する際には、一意のリソース名（URN）を指定する必要があります。プラットフォームサービスの構成XMLを作成する際に、URNを使用してエンドポイントを参照します。各エンドポイントのURNは一意である必要があります。

StorageGRID は、プラットフォームサービスのエンドポイントを作成する際に、その妥当性を検証します。プラットフォームサービスのエンドポイントを作成する前に、エンドポイントで指定されたリソースが存在し、かつアクセス可能であることを確認してください。

URN要素

プラットフォームサービスエンドポイントのURNは、次のように `arn:aws`または `urn:mysite`のいずれかで始まる必要があります：

- サービスがAmazon Web Services (AWS) でホストされている場合は、 `arn:aws`
- サービスが Google Cloud Platform (GCP) でホストされている場合は、 `arn:aws`
- サービスがローカルでホストされている場合は、 `urn:mysite`

例えば、StorageGRID でホストされている CloudMirror エンドポイントの URN を指定する場合、URN は次のように始まる場合があります `urn:sgws`。

URNの次の要素は、プラットフォームサービスのタイプを次のように指定します：

サービス	**Type**
CloudMirror レプリケーション	`s3`
通知	`sns`、 `kafka`、または `webhook`
検索統合	`es`

たとえば、StorageGRID でホストされている CloudMirror エンドポイントの URN の指定を続けるには、 `s3` を追加して `urn:sgws:s3` を取得します。

ほとんどのエンドポイントでは、URN の最後の要素は、宛先 URI の特定のターゲット リソースを識別します。たとえば、 `sns-topic-name`。

ウェブフックのエンドポイントの場合、ターゲットリソースは宛先URIそのものです。

サービス	特定のリソース
CloudMirror レプリケーション	`bucket-name`

サービス	特定のリソース
通知	`sns-topic-name` または `kafka-topic-name` 注: Webhookエンドポイントの場合、エンドポイントのURNが一意である限り、URNの最後の要素は任意の文字列にすることができます。
検索統合	`domain-name/index-name/type-name` 注: Elasticsearch クラスターがインデックスを自動的に作成するように構成されていない場合は、エンドポイントを作成する前にインデックスを手動で作成する必要があります。

AWSおよび**GCP**でホストされているサービスの**URN**

AWSおよびGCPエンティティの場合、完全なURNは有効なAWS ARNです。例：

- CloudMirror レプリケーション：

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

- 通知：

```
arn:aws:sns:region:account-id:topic-name
```

- 検索統合：

```
arn:aws:es:region:account-id:domain/domain-name/index-name/type-name
```

AWS検索統合エンドポイントの場合、 `domain-name`にはリテラル文字列 `domain/`を含める必要があります（ここに示すとおり）。

ローカルホスト型サービスの**URN**

クラウドサービスではなくローカルホスト型サービスを使用する場合、URN の 3 番目と最後の位置に必須要素が含まれている限り、有効かつ一意の URN を生成する方法であれば、URN を任意の方法で指定できます。オプションで示されている要素は空白のままにしておくことも、リソースを識別し、URN を一意にするために役立つ方法で指定することもできます。例えば：

- CloudMirror レプリケーション：

```
urn:mysite:s3:optional:optional:bucket-name
```

CloudMirror StorageGRID でホストされているエンドポイントの場合、次で始まる有効な URN を指定できます `urn:sgws`：

```
urn:sgws:s3:optional:optional:bucket-name
```

- 通知：

  Amazon Simple Notification Service のエンドポイントを指定します：

  ```
  urn:mysite:sns:optional:optional:sns-topic-name
  ```

  Kafkaエンドポイントを指定：

  ```
  urn:mysite:kafka:optional:optional:kafka-topic-name
  ```

  ウェブフックのエンドポイントを指定します：

  ```
  urn:mysite:webhook:optional:optional:webhook-name
  ```

- 検索統合：

  ```
  urn:mysite:es:optional:optional:domain-name/index-name/type-name
  ```

  ローカルでホストされている検索統合エンドポイントの場合、 `domain-name`要素は、エンドポイントのURNが一意である限り、任意の文字列を指定できます。

StorageGRIDプラットフォームサービスエンドポイントを作成する

プラットフォームサービスを有効にするには、まず適切なタイプのエンドポイントを少なくとも1つ作成する必要があります。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- プラットフォームサービスは、StorageGRID管理者によってテナントアカウントで有効化されました。
- あなたは、"エンドポイントの管理またはルートアクセス権限" を持つユーザーグループに属しています。
- プラットフォームサービスエンドポイントによって参照されるリソースが作成されました。
  - CloudMirrorレプリケーション：S3バケット
  - イベント通知：Amazon Simple Notification Service (Amazon SNS) トピック、Kafka トピック、または Webhook エンドポイント

- 検索通知：Elasticsearch インデックス（デスティネーション クラスタがインデックスを自動的に作成するように設定されていない場合）。

- デスティネーション リソースに関する情報があります。
  - URI（Uniform Resource Identifier）のホスト名とポート番号

    StorageGRID システムでホストされているバケットを CloudMirror レプリケーションのエンドポイントとして使用する予定がある場合は、グリッド管理者に連絡して、入力する必要のある値を確認してください。

  - 固有リソース名（URN）

    "プラットフォームサービスエンドポイントのURNを指定します。"

  - 認証情報（必要な場合）：

    検索統合エンドポイント

    検索統合エンドポイントには、以下の認証情報を使用できます。

    - アクセスキー：アクセスキーIDとシークレットアクセスキー
    - 基本的なHTTP：ユーザー名とパスワード

    CloudMirrorレプリケーションエンドポイント

    CloudMirror レプリケーションエンドポイントには、次の認証情報を使用できます：

    - アクセスキー：アクセスキーIDとシークレットアクセスキー
    - CAP（C2S Access Portal）：一時的な認証情報URL、サーバー証明書とクライアント証明書、クライアントキー、およびオプションのクライアント秘密鍵パスフレーズ。

    Amazon SNSエンドポイント

    Amazon SNS エンドポイントの場合、以下の認証情報を使用できます：

    - アクセスキー：アクセスキーIDとシークレットアクセスキー

    Kafkaエンドポイント

    Kafkaエンドポイントの場合、以下の認証情報を使用できます：

    - SASL/PLAIN: ユーザー名とパスワード
    - SASL/SCRAM-SHA-256：ユーザー名とパスワード
    - SASL/SCRAM-SHA-512: ユーザー名とパスワード

  - セキュリティ証明書（証明書の検証が必要な場合）

- Elasticsearchのセキュリティ機能が有効になっている場合、接続テストのためのモニタークラスタ権限と、ドキュメント更新のための書き込みインデックス権限、またはインデックス権限と削除インデックス権限の両方が必要です。

手順

1. **STORAGE (S3)** > **Platform services endpoints** を選択します。Platform services endpoints ページが表示されます。
2. 「エンドポイントの作成」を選択します。
3. エンドポイントとその目的を簡潔に説明する表示名を入力してください。

   エンドポイントがサポートするプラットフォームサービスの種類は、エンドポイントページにエンドポイント名が表示される際に、エンドポイント名の横に表示されるため、その情報を名前に含める必要はありません。

4. **URI** フィールドには、エンドポイントの固有リソース識別子（URI）を指定します。

   以下のいずれかの形式を使用してください。

   ```
   https://host:port
   http://host:port
   ```

   ポートを指定しない場合、以下のデフォルトポートが使用されます：

   - HTTPS URIにはポート443、HTTP URIにはポート80 （ほとんどのエンドポイント）
   - HTTPSおよびHTTP URI用のポート9092（Kafkaエンドポイントのみ）

   例えば、StorageGRID でホストされているバケットの URI は次のようになります：

   ```
   https://s3.example.com:10443
   ```

   この例では、 `s3.example.com`は StorageGRID 高可用性（HA）グループの仮想 IP（VIP）の DNS エントリを表し、 `10443`はロードバランサーエンドポイントで定義されたポートを表します。

   > ⓘ 可能な限り、単一障害点を回避するために、負荷分散ノードのHAグループに接続してください。

   同様に、AWSでホストされているバケットのURIは次のようになります：

   ```
   https://s3-aws-region.amazonaws.com
   ```

   > ⓘ エンドポイントが CloudMirror レプリケーションサービスに使用される場合は、URI にバケット名を含めないでください。バケット名は **URN** フィールドに含めます。

5. エンドポイントの固有リソース名（URN）を入力してください。

   > ⓘ エンドポイントが作成された後は、エンドポイントのURNを変更することはできません。

6. *続行*を選択します。

7. *認証タイプ*の値を選択してください。

ウェブフックエンドポイントの認証が必要な場合は、ステップ9 で Mutual Transport Layer Security (mTLS) を設定してください。

検索統合エンドポイント

検索統合エンドポイントの認証情報を入力またはアップロードしてください。

入力する認証情報には、デスティネーション リソースへの書き込み権限が必要です。

認証タイプ	説明	**Credentials**
匿名	デスティネーションへの匿名アクセスを提供します。セキュリティ機能が無効になっているエンドポイントでのみ機能します。	認証なし。
アクセス キー	AWSスタイルの認証情報を使用して、接続先との接続を認証します。	• アクセス キー ID • シークレット アクセス キー
基本的なHTTP	ユーザー名とパスワードを使用して、接続先への接続を認証します。	• Username • Password

CloudMirrorレプリケーションエンドポイント

CloudMirror レプリケーションエンドポイントの認証情報を入力またはアップロードします。

入力する認証情報には、デスティネーション リソースへの書き込み権限が必要です。

認証タイプ	説明	**Credentials**
匿名	デスティネーションへの匿名アクセスを提供します。セキュリティ機能が無効になっているエンドポイントでのみ機能します。	認証なし。
アクセス キー	AWSスタイルの認証情報を使用して、接続先との接続を認証します。	• アクセス キー ID • シークレット アクセス キー
CAP（C2S Access Portal）	証明書と鍵を使用して、デスティネーションへの接続を認証します。	• 一時的な認証情報 URL • サーバーCA証明書（PEMファイルアップロード） • クライアント証明書（PEMファイルアップロード） • クライアント秘密鍵（PEMファイルアップロード、OpenSSL暗号化形式、または暗号化されていない秘密鍵形式） • クライアント秘密鍵のパスフレーズ（オプション）

Amazon SNSエンドポイント

Amazon SNSエンドポイントの認証情報を入力またはアップロードしてください。

入力する認証情報には、デスティネーション リソースへの書き込み権限が必要です。

認証タイプ	説明	**Credentials**
匿名	デスティネーションへの匿名アクセスを提供します。セキュリティ機能が無効になっているエンドポイントでのみ機能します。	認証なし。
アクセス キー	AWSスタイルの認証情報を使用して、接続先との接続を認証します。	• アクセス キー ID • シークレット アクセス キー

Kafkaエンドポイント

Kafkaエンドポイントの認証情報を入力またはアップロードしてください。

入力する認証情報には、デスティネーション リソースへの書き込み権限が必要です。

認証タイプ	説明	**Credentials**
匿名	デスティネーションへの匿名アクセスを提供します。セキュリティ機能が無効になっているエンドポイントでのみ機能します。	認証なし。
SASL/PLAIN	平文のユーザー名とパスワードを使用して、デスティネーションへの接続を認証します。	• Username • Password
SASL/SCRAM-SHA-256	チャレンジレスポンスプロトコルとSHA-256ハッシュを使用して、ユーザー名とパスワードによりデスティネーションへの接続を認証します。	• Username • Password
SASL/SCRAM-SHA-512	チャレンジレスポンスプロトコルとSHA-512ハッシュを使用して、ユーザー名とパスワードによるデスティネーションへの接続認証を行います。	• Username • Password

ユーザー名とパスワードが Kafka クラスタから取得した委任トークンから派生している場合は、「委任トークン認証を使用する」を選択してください。

8. *続行*を選択します。
9. エンドポイントへのTLS接続の検証方法を選択するには、*証明書の検証*のラジオボタンを選択します。

ほとんどのエンドポイント

検索統合、CloudMirror レプリケーション、Amazon SNS、または Kafka エンドポイントの TLS 接続を確認します。

証明書検証の種類	説明
TLS	エンドポイントリソースへのTLS接続に使用するサーバー証明書を検証します。
無効	証明書の検証は無効になっています。このオプションは安全ではありません。
カスタムCA証明書を使用する	カスタムCA証明書は、エンドポイントに接続する際にサーバーの身元を確認するために使用されます。
オペレーティングシステムのCA証明書を使用する	接続を保護するには、オペレーティングシステムにインストールされているデフォルトのGrid CA証明書を使用してください。

Webhookエンドポイントのみ

ウェブフックエンドポイントのTLS接続を確認してください。

証明書検証の種類	説明
TLS	エンドポイントリソースへのTLS接続に使用するサーバー証明書を検証します。
mTLS	エンドポイントリソースへの相互TLS接続におけるクライアント証明書とサーバー証明書を検証します。
無効	証明書の検証は無効になっています。このオプションは安全ではありません。
カスタムCA証明書を使用する	カスタムCA証明書は、エンドポイントに接続する際にサーバーの身元を確認するために使用されます。

*mTLS*を選択すると、これらのオプションが利用可能になります。

証明書検証の種類	説明
サーバー証明書を検証しない	サーバー証明書の検証を無効にするため、サーバーの身元は検証されません。このオプションは安全ではありません。
クライアント証明書	クライアント証明書は、エンドポイントに接続する際にクライアントの身元を確認するために使用されます。

証明書検証の種類	説明
クライアントの秘密鍵	クライアント証明書の秘密鍵。暗号化する場合は、従来の形式である PKCS #1 を使用する必要があります（PKCS #8 形式はサポートされていません）。
クライアント秘密鍵パスフレーズ	クライアントの秘密鍵を復号するためのパスフレーズ。秘密鍵が暗号化されていない場合は、この欄を空欄にしてください。

10. 「Test and create endpoint」を選択します。
    - 指定された認証情報を使用してエンドポイントにアクセスできた場合、成功メッセージが表示されます。エンドポイントへの接続は、各サイトの1つのノードから検証されます。
    - エンドポイントの検証に失敗すると、エラーメッセージが表示されます。エラーを修正するためにエンドポイントを変更する必要がある場合は、*エンドポイントの詳細に戻る*を選択して情報を更新してください。次に、*エンドポイントのテストと作成*を選択します。

> テナントアカウントでプラットフォームサービスが有効になっていない場合、エンドポイントの作成は失敗します。StorageGRID 管理者にお問い合わせください。

エンドポイントを設定したら、そのURNを使用してプラットフォームサービスを設定できます。

関連情報

- "プラットフォームサービスエンドポイントのURNを指定します。"
- "CloudMirror レプリケーションの設定"
- "イベント通知の設定"
- "検索統合サービスの設定"

StorageGRIDプラットフォームサービスエンドポイントの接続をテストします

プラットフォームサービスへの接続が変更された場合は、エンドポイントへの接続をテストして、宛先リソースが存在すること、および指定した認証情報を使用してアクセスできることを検証できます。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"エンドポイントの管理またはルートアクセス権限" を持つユーザーグループに属しています。

タスク概要

StorageGRIDは、認証情報に適切な権限があるかどうかを検証しません。

手順

1. **STORAGE (S3)** > **Platform services endpoints** を選択します。

   プラットフォームサービスエンドポイントページが表示され、既に設定済みのプラットフォームサービス

エンドポイントの一覧が表示されます。

2. 接続テストを行うエンドポイントを選択してください。

   エンドポイントの詳細ページが表示されます。

3. 「接続テスト」を選択します。
   - 指定された認証情報を使用してエンドポイントにアクセスできた場合、成功メッセージが表示されます。エンドポイントへの接続は、各サイトの1つのノードから検証されます。
   - エンドポイントの検証に失敗すると、エラーメッセージが表示されます。エラーを修正するためにエンドポイントを変更する必要がある場合は、**Configuration** を選択して情報を更新してください。次に、**Test and save changes** を選択します。

StorageGRIDでプラットフォームサービスエンドポイントを編集する

プラットフォームサービスエンドポイントの設定を編集することで、その名前、URI、その他の詳細を変更できます。例えば、期限切れの認証情報を更新したり、フェイルオーバーのためにURIをバックアップのElasticsearchインデックスを指すように変更したりする必要がある場合があります。プラットフォームサービスのエンドポイントのURNは変更できません。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"エンドポイントの管理またはルートアクセス権限" を持つユーザーグループに属しています。

手順

1. **STORAGE (S3)** > **Platform services endpoints** を選択します。

   プラットフォームサービスエンドポイントページが表示され、既に設定済みのプラットフォームサービスエンドポイントの一覧が表示されます。

2. 編集するエンドポイントを選択します。

   エンドポイントの詳細ページが表示されます。

3. 「Configuration」を選択します。
4. 必要に応じて、エンドポイントの設定を変更してください。

   エンドポイントが作成された後は、エンドポイントのURNを変更することはできません。

   a. エンドポイントの表示名を変更するには、編集アイコン を選択します。
   b. 必要に応じてURIを変更してください。
   c. 必要に応じて、認証タイプを変更してください。
      - アクセスキー認証の場合、必要に応じて*S3キーの編集*を選択し、新しいアクセスキーIDとシークレットアクセスキーを貼り付けてキーを変更してください。変更を取り消す必要がある場合は、*S3キーの編集を元に戻す*を選択してください。

- CAP（C2S Access Portal）認証の場合、必要に応じて一時的な認証情報URLまたはオプションのクライアント秘密鍵パスフレーズを変更し、新しい証明書ファイルと鍵ファイルをアップロードしてください。

> クライアントの秘密鍵は、OpenSSLで暗号化された形式、または暗号化されていない秘密鍵形式のいずれかである必要があります。

d. 必要に応じて、証明書の検証方法を変更してください。

5. 「テストして変更を保存」を選択します。
   - 指定された認証情報を使用してエンドポイントにアクセスできた場合、成功メッセージが表示されます。各サイトの1つのノードからエンドポイントへの接続が検証されます。
   - エンドポイントの検証に失敗すると、エラーメッセージが表示されます。エラーを修正するためにエンドポイントを変更し、「テストして変更を保存」を選択します。

StorageGRIDプラットフォームサービスエンドポイントを削除する

関連付けられたプラットフォームサービスを使用しなくなった場合は、エンドポイントを削除できます。

開始する前に

- テナントマネージャーに "対応ウェブブラウザ" を使用してサインインしています。
- あなたは、"エンドポイントの管理またはルートアクセス権限" を持つユーザーグループに属しています。

手順

1. **STORAGE (S3)** ＞ **Platform services endpoints** を選択します。

   プラットフォームサービスエンドポイントページが表示され、既に設定済みのプラットフォームサービスエンドポイントの一覧が表示されます。

2. 削除する各エンドポイントのチェックボックスを選択します。

> 使用中のプラットフォームサービスエンドポイントを削除すると、そのエンドポイントを使用しているすべてのバケットで、関連付けられたプラットフォームサービスが無効になります。未完了のリクエストはすべて破棄されます。削除されたURNを参照しないようにバケット設定を変更するまで、新しいリクエストは引き続き生成されます。StorageGRIDは、これらのリクエストを回復不能なエラーとして報告します。

3. **Actions** > **Delete endpoint** を選択します。

   確認メッセージが表示されます。

4. 「エンドポイントを削除」を選択します。

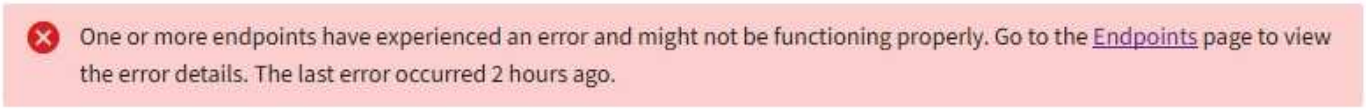
StorageGRIDプラットフォームサービスエンドポイントエラーのトラブルシューティング

StorageGRID がプラットフォームサービスのエンドポイントとの通信を試みたときにエラーが発生した場合、ダッシュボードにメッセージが表示されます。プラットフォームサービスのエンドポイントページでは、「最終エラー」列にエラーが発生してからの経

過時間が表示されます。エンドポイントの認証情報に関連付けられた権限が正しくない場合、エラーは表示されません。

エラーが発生したかどうかを判断する

過去7日以内にプラットフォームサービスのエンドポイントでエラーが発生した場合、テナントマネージャーのダッシュボードに警告メッセージが表示されます。エラーの詳細については、プラットフォームサービスのエンドポイントページをご覧ください。



ダッシュボードに表示されるのと同じエラーが、プラットフォームサービスエンドポイントページの上部にも表示されます。より詳細なエラーメッセージを表示するには：

手順

1. エンドポイントの一覧から、エラーが発生しているエンドポイントを選択してください。
2. エンドポイントの詳細ページで、「接続」を選択します。このタブには、エンドポイントに関する最新のエラーのみが表示され、エラーが発生した日時も示されます。赤いXアイコンを含むエラーは、過去7日以内に発生したものです。

エラーがまだ発生しているかどうかを確認してください

エラーが解決された後でも、*最後のエラー*列にエラーが表示され続ける場合があります。エラーが現在発生しているかどうかを確認する場合、または解決済みのエラーをテーブルから強制的に削除する場合は、以下の手順に従ってください：

手順

1. エンドポイントを選択します。

   エンドポイントの詳細ページが表示されます。

2. 「接続」＞「接続テスト」を選択します。

「接続テスト」を選択すると、StorageGRID はプラットフォームサービスのエンドポイントが存在し、現在の認証情報でアクセスできることを検証します。エンドポイントへの接続は、各サイトの1つのノードから検証されます。

エンドポイントのエラーを解決する

エンドポイントの詳細ページに表示される「最後のエラー」メッセージは、エラーの原因を特定するのに役立ちます。エラーによっては、問題を解決するためにエンドポイントを編集する必要がある場合があります。例えば、StorageGRID が宛先の S3 バケットにアクセスできない場合（アクセス権限が正しくないか、アクセスキーの有効期限が切れている場合）に、CloudMirroring エラーが発生することがあります。メッセージは「エンドポイント認証情報または宛先アクセスを更新する必要があります」で、詳細は「AccessDenied」または「InvalidAccessKeyId」です。

エラーを解決するためにエンドポイントを編集する必要がある場合は、「テストして変更を保存」を選択すると、StorageGRID が更新されたエンドポイントを検証し、現在の認証情報でアクセスできることを確認しま

す。エンドポイントへの接続は、各サイトの1つのノードから検証されます。

手順

1. エンドポイントを選択します。
2. エンドポイントの詳細ページで、**Configuration** を選択します。
3. 必要に応じてエンドポイントの設定を編集します。
4. 「接続」＞「接続テスト」を選択します。

エンドポイントの認証情報に権限が不足しています

StorageGRID がプラットフォームサービスのエンドポイントを検証する際、エンドポイントの認証情報を使用して宛先リソースに接続できることを確認し、基本的な権限チェックを実行します。ただし、StorageGRID は特定のプラットフォームサービス操作に必要なすべての権限を検証するわけではありません。そのため、プラットフォームサービスの使用時にエラー（「403 Forbidden」など）が発生した場合は、エンドポイントの認証情報に関連付けられている権限を確認してください。

関連情報

- **StorageGRID** の管理 › プラットフォームサービスのトラブルシューティング
- "プラットフォームサービスエンドポイントを作成する"
- "プラットフォームサービスエンドポイントへの接続テスト"
- "プラットフォームサービスエンドポイントを編集"

StorageGRIDで**CloudMirror**レプリケーションを設定する

バケットの CloudMirror レプリケーションを有効にするには、有効なバケットレプリケーション設定 XML を作成して適用します。

開始する前に

- プラットフォームサービスは、StorageGRID管理者によってテナントアカウントで有効化されました。
- レプリケーション元として機能するバケットは既に作成済みです。
- CloudMirror レプリケーションの宛先として使用するエンドポイントがすでに存在しており、その URN も取得済みです。
- あなたは、"すべてのバケットを管理する、またはルートアクセス権限" を持つユーザーグループに属しています。これらの権限は、テナントマネージャーを使用してバケットを設定する際に、グループポリシーまたはバケットポリシーの権限設定を上書きします。

タスク概要

CloudMirrorレプリケーションは、エンドポイントで指定された宛先バケットに、ソースバケットからオブジェクトをコピーします。

バケットレプリケーションとその設定方法に関する一般的な情報については、 "Amazon Simple Storage Service (S3) ドキュメント：オブジェクトのレプリケーション"を参照してください。StorageGRID が GetBucketReplication、DeleteBucketReplication、および PutBucketReplication を実装する方法については、"バケットに対する操作"を参照してください。

CloudMirrorレプリケーションには、クロスグリッドレプリケーション機能と重要な類似点と相違点があります。詳細については、"グリッド間レプリケーションとCloudMirrorレプリケーションの比較"を参照してください。

CloudMirror レプリケーションを設定する際は、以下の要件と特性に注意してください：

- 有効なバケットレプリケーション構成XMLを作成して適用する場合、各宛先に対してS3バケットエンドポイントのURNを使用する必要があります。
- S3オブジェクトロックが有効になっているソースバケットまたは宛先バケットでは、レプリケーションはサポートされていません。
- オブジェクトを含むバケットで CloudMirror レプリケーションを有効にすると、バケットに追加された新しいオブジェクトは複製されますが、バケット内の既存のオブジェクトは複製されません。レプリケーションをトリガーするには、既存のオブジェクトを更新する必要があります。
- レプリケーション構成 XML でストレージクラスを指定すると、StorageGRID は宛先 S3 エンドポイントに対して操作を実行する際に、そのクラスを使用します。宛先エンドポイントも、指定されたストレージクラスをサポートしている必要があります。宛先システムベンダーが提供する推奨事項には必ず従ってください。

手順

1. ソースバケットのレプリケーションを有効にする：
   - S3レプリケーションAPIで指定されているとおり、テキストエディタを使用して、レプリケーションを有効にするために必要なレプリケーション構成XMLを作成します。
   - XMLを設定する際：
     - StorageGRID はレプリケーション構成の V1 のみをサポートしていることに注意してください。つまり、StorageGRID はルールに対して `Filter`要素の使用をサポートしておらず、オブジェクトバージョンの削除については V1 の規約に従います。詳細については、レプリケーション設定に関する Amazon のドキュメントを参照してください。
     - 宛先として、S3バケットエンドポイントのURNを使用してください。
     - オプションで `<StorageClass>`要素を追加し、以下のいずれかを指定します。
       - `STANDARD`：デフォルトのストレージクラス。オブジェクトをアップロードする際にストレージクラスを指定しない場合、 `STANDARD`ストレージクラスが使用されます。
       - `STANDARD_IA`：（Standard - infrequent access.）アクセス頻度は低いものの、必要なときに迅速なアクセスが求められるデータには、このストレージクラスを使用してください。
       - `REDUCED_REDUNDANCY`：このストレージクラスは、 `STANDARD`ストレージクラスよりも冗長性を低くして保存できる、重要度の低い再現可能なデータに使用します。
     - 設定XMLで `Role`を指定した場合、無視されます。この値はStorageGRIDでは使用されません。

```
<ReplicationConfiguration>
    <Role></Role>
    <Rule>
        <Status>Enabled</Status>
        <Prefix>2020</Prefix>
        <Destination>
           <Bucket>urn:sgws:s3:::2017-records</Bucket>
           <StorageClass>STANDARD</StorageClass>
         </Destination>
     </Rule>
</ReplicationConfiguration>
```

2. ダッシュボードから「バケットの表示」を選択するか、**STORAGE (S3)** > **Buckets** を選択してください。
3. ソースバケットの名前を選択してください。

   バケットの詳細ページが表示されます。

4. プラットフォームサービス ＞ レプリケーション を選択します。
5. 「レプリケーションを有効にする」チェックボックスを選択します。
6. レプリケーション構成のXMLをテキストボックスに貼り付け、*変更を保存*を選択します。

   プラットフォームサービスは、Grid ManagerまたはGrid Management APIを使用するStorageGRID管理者が、テナントアカウントごとに有効化する必要があります。設定XMLを保存する際にエラーが発生した場合は、StorageGRID管理者にお問い合わせください。

7. レプリケーションが正しく構成されていることを確認してください。
   a. レプリケーション構成で指定されているレプリケーション要件を満たすオブジェクトをソースバケットに追加します。

      先に示した例では、「2020」という接頭辞に一致するオブジェクトが複製されます。

   b. オブジェクトが宛先バケットに複製されたことを確認してください。

      小さなオブジェクトの場合、レプリケーションは迅速に行われます。

関連情報

"プラットフォームサービスエンドポイントを作成する"

StorageGRID でイベント通知を設定

バケットの通知を有効にするには、通知設定XMLを作成し、テナントマネージャーを使用してそのXMLをバケットに適用します。

開始する前に

- プラットフォームサービスは、StorageGRID管理者によってテナントアカウントで有効化されました。
- 通知の送信元となるバケットは既に作成済みです。
- イベント通知の送信先として使用しようとしているエンドポイントは既に存在しており、そのURNも取得済みです。
- あなたは、"すべてのバケットを管理する、またはルートアクセス権限" を持つユーザーグループに属しています。これらの権限は、テナントマネージャーを使用してバケットを設定する際に、グループポリシーまたはバケットポリシーの権限設定を上書きします。

タスク概要

イベント通知を設定するには、通知設定XMLをソースバケットに関連付けます。通知設定XMLは、バケット通知の設定に関するS3の規則に従い、宛先となるAmazon SNSトピック、Kafkaトピック、またはWebhookエンドポイントは、エンドポイントのURNとして指定されます。

イベント通知とその設定方法に関する一般的な情報については、 "Amazonのドキュメント"を参照してください。StorageGRID が S3 バケット通知設定 API を実装する方法については、"S3クライアントアプリケーションを実装するための手順"を参照してください。

バケットのイベント通知を設定する際には、以下の要件と特性に注意してください。

- 有効な通知設定XMLを作成して適用する場合、各宛先に対してイベント通知エンドポイントのURNを使用する必要があります。
- S3オブジェクトロックが有効になっているバケットでイベント通知を設定することは可能ですが、オブジェクトのS3オブジェクトロックメタデータ（Retain Until DateやLegal Hold statusなど）は通知メッセージには含まれません。
- イベント通知を設定すると、ソースバケット内のオブジェクトに対して指定されたイベントが発生するたびに、通知が生成され、宛先として指定された Amazon SNS トピック、Kafka トピック、または Webhook エンドポイントに送信されます。
- オブジェクトを含むバケットに対してイベント通知を有効にした場合、通知は通知設定が保存された後に実行されたアクションに対してのみ送信されます。

手順

1. ソースバケットの通知を有効にする：
    - S3通知APIで指定されているとおり、イベント通知を有効にするために必要な通知設定XMLをテキストエディタを使用して作成します。
    - XMLを設定する際は、イベント通知エンドポイントのURNを宛先トピックとして使用してください。

```
<NotificationConfiguration>
  <TopicConfiguration>
     <Id>Image-created</Id>
     <Filter>
        <S3Key>
          <FilterRule>
            <Name>prefix</Name>
            <Value>images/</Value>
          </FilterRule>
        </S3Key>
    </Filter>
    <Topic>arn:aws:sns:us-east-1:050340950352:sgws-topic</Topic>
    <Event>s3:ObjectCreated:*</Event>
   </TopicConfiguration>
 </NotificationConfiguration>
```

2. テナントマネージャーで、**STORAGE (S3)** > *バケット*を選択します。
3. ソースバケットの名前を選択してください。

   バケットの詳細ページが表示されます。

4. **Platform services** > **Event notifications** を選択します。
5. 「イベント通知を有効にする」チェックボックスを選択します。
6. 通知設定のXMLをテキストボックスに貼り付け、*変更を保存*を選択してください。

   プラットフォームサービスは、Grid ManagerまたはGrid Management APIを使用するStorageGRID管理者が、テナントアカウントごとに有効化する必要があります。設定XMLを保存する際にエラーが発生した場合は、StorageGRID管理者にお問い合わせください。

7. イベント通知が正しく設定されていることを確認してください：

   a. 構成XMLで設定されている通知トリガーの要件を満たす、ソースバケット内のオブジェクトに対してアクションを実行します。

      この例では、 `images/`プレフィックスを使用してオブジェクトが作成されるたびに、イベント通知が送信されます。

   b. 通知が宛先のAmazon SNSトピック、Kafkaトピック、またはWebhookエンドポイントに配信されたことを確認してください。

      例えば、宛先トピックがAmazon SNSでホストされている場合、通知が配信されたときにメールを送信するようにサービスを設定できます。

```
{
   "Records":[
      {
         "eventVersion":"2.0",
         "eventSource":"sgws:s3",
         "eventTime":"2017-08-08T23:52:38Z",
         "eventName":"ObjectCreated:Put",
         "userIdentity":{
            "principalId":"11111111111111111"
         },
         "requestParameters":{
            "sourceIPAddress":"193.51.100.20"
         },
         "responseElements":{
            "x-amz-request-id":"122047343"
         },
         "s3":{
            "s3SchemaVersion":"1.0",
            "configurationId":"Image-created",
            "bucket":{
               "name":"test1",
               "ownerIdentity":{
                  "principalId":"11111111111111111"
               },
               "arn":"arn:sgws:s3:::test1"
            },
            "object":{
               "key":"images/cat.jpg",
               "size":0,
               "eTag":"d41d8cd98f00b204e9800998ecf8427e",
               "sequencer":"14D90402421461C7"
            }
         }
      }
   ]
}
```

+
宛先トピックで通知が受信された場合、StorageGRID 通知用のソースバケットの設定が正常に完了しています。

関連情報

- "バケットに関する通知を理解する"
- "S3 REST APIを使用する"

- "プラットフォームサービスエンドポイントを作成する"

StorageGRID で検索統合サービスを設定します

バケットの検索統合を有効にするには、検索統合XMLを作成し、テナントマネージャーを使用してそのXMLをバケットに適用します。

開始する前に

- プラットフォームサービスは、StorageGRID管理者によってテナントアカウントで有効化されました。
- 既にS3バケットを作成済みで、そのバケットの内容をインデックス化したいと考えています。
- 検索統合サービスの宛先として使用しようとしているエンドポイントは既に存在しており、そのURNも取得済みです。
- あなたは、"すべてのバケットを管理する、またはルートアクセス権限" を持つユーザーグループに属しています。これらの権限は、テナントマネージャーを使用してバケットを設定する際に、グループポリシーまたはバケットポリシーの権限設定を上書きします。

タスク概要

ソースバケットの検索統合サービスを設定した後、オブジェクトを作成したり、オブジェクトのメタデータやタグを更新したりすると、オブジェクトのメタデータが宛先エンドポイントに送信されます。

オブジェクトが既に含まれているバケットに対して検索統合サービスを有効にした場合、既存のオブジェクトに関するメタデータ通知は自動的に送信されません。これらの既存オブジェクトを更新して、メタデータが宛先検索インデックスに追加されるようにしてください。

手順

1. バケットの検索統合を有効にする：
    - 検索統合を有効にするために必要なメタデータ通知XMLを作成するには、テキストエディタを使用します。
    - XMLを設定する際は、検索統合エンドポイントのURNを宛先として使用してください。

      オブジェクトは、オブジェクト名のプレフィックスに基づいてフィルタリングできます。たとえば、プレフィックス `images`を持つオブジェクトのメタデータを1つの宛先に送信し、プレフィックス `videos`を持つオブジェクトのメタデータを別の宛先に送信できます。プレフィックスが重複している設定は無効であり、送信時に拒否されます。たとえば、プレフィックス `test`を持つオブジェクトに対するルールと、プレフィックス `test2`を持つオブジェクトに対する2つ目のルールを含む設定は許可されていません。

      必要に応じて、メタデータ構成XMLの例を参照してください。

```
<MetadataNotificationConfiguration>
 <Rule>
    <Status>Enabled</Status>
    <Prefix></Prefix>
    <Destination>
       <Urn>/Urn>
       </Destination>
 </Rule>
</MetadataNotificationConfiguration>
```

メタデータ通知設定XMLの要素：

Name	説明	必須
MetadataNotificationConfiguration	メタデータ通知の対象となる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を格納するタグ。 1つ以上のルール要素を含みます。	はい
Rule	メタデータを特定のインデックスに追加すべきオブジェクトを識別するルールのコンテナタグ。 接頭辞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	ルールの一意の識別子。 ルール要素に含まれています。	いいえ
ステータス	ステータスは「有効」または「無効」のいずれかになります。無効化されたルールに対しては、何も措置は講じられません。 ルール要素に含まれています。	はい
Prefix	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。 すべてのオブジェクトに一致させるには、空のプレフィックスを指定します。 ルール要素に含まれています。	はい
デスティネーション	ルールの適用先を示すコンテナタグ。 ルール要素に含まれています。	はい

Name	説明	必須
Urn	オブジェクトのメタデータが送信される宛先のURN。以下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： • `es`3番目の要素でなければなりません。 • 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 `domain-name/myindex/mytype`です。 エンドポイントは、Tenant ManagerまたはTenant Management APIを使用して設定します。形式は次のとおりです： • `arn:aws:es:region:account-ID:domain/mydomain/myindex/mytype` • `urn:mysite:es:::mydomain/myindex/mytype` エンドポイントは、設定XMLを送信する前に設定しておく必要があります。設定されていない場合、設定は404エラーで失敗します。 URNはDestination要素に含まれています。	はい

2. テナントマネージャーで、**STORAGE (S3)** > *Buckets*を選択します。
3. ソースバケットの名前を選択してください。

   バケットの詳細ページが表示されます。

4. 「プラットフォームサービス」 > 「検索統合」を選択します。
5. 検索統合を有効にする チェックボックスを選択します。
6. メタデータ通知の設定をテキストボックスに貼り付け、*変更を保存*を選択します。

   プラットフォームサービスは、Grid Managerまたは管理APIを使用して、StorageGRID管理者がテナントアカウントごとに有効化する必要があります。設定XMLを保存する際にエラーが発生した場合は、StorageGRID管理者にお問い合わせください。

7. 検索統合サービスが正しく設定されていることを確認してください。
   a. 構成XMLで指定されているメタデータ通知をトリガーするための要件を満たすオブジェクトをソースバケットに追加します。

      先に示した例では、バケットに追加されたすべてのオブジェクトがメタデータ通知をトリガーします。

   b. エンドポイントで指定された検索インデックスに、オブジェクトのメタデータとタグを含む JSON ドキュメントが追加されたことを確認してください。

終了後の操作

必要に応じて、以下のいずれかの方法を使用して、バケットの検索統合を無効にすることができます。

- **STORAGE (S3)** ＞ **Buckets** を選択し、**Enable search integration** チェックボックスをオフにします。
- S3 APIを直接使用する場合は、DELETE Bucket metadata notificationリクエストを使用してください。S3クライアントアプリケーションの実装手順を参照してください。

例：すべてのオブジェクトに適用されるメタデータ通知設定

この例では、すべてのオブジェクトのメタデータが同じ宛先に送信されます。

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

例：**2**つのルールを使用したメタデータ通知の設定

この例では、プレフィックス `/images`に一致するオブジェクトのオブジェクトメタデータは1つの宛先に送信され、プレフィックス `/videos`に一致するオブジェクトのオブジェクトメタデータは2番目の宛先に送信されます。

```
<MetadataNotificationConfiguration>
    <Rule>
        <ID>Images-rule</ID>
        <Status>Enabled</Status>
        <Prefix>/images</Prefix>
        <Destination>
           <Urn>arn:aws:es:us-east-1:3333333:domain/es-
domain/graphics/imagetype</Urn>
        </Destination>
    </Rule>
    <Rule>
        <ID>Videos-rule</ID>
        <Status>Enabled</Status>
        <Prefix>/videos</Prefix>
        <Destination>
           <Urn>arn:aws:es:us-west-1:22222222:domain/es-
domain/graphics/videotype</Urn>
        </Destination>
    </Rule>
</MetadataNotificationConfiguration>
```

メタデータ通知フォーマット

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。

この例は、キー `SGWS/Tagging.txt`を持つオブジェクトが `test`という名前のバケットに作成された場合に生成される可能性のある JSON の例を示しています。 `test`バケットはバージョン管理されていないため、`versionId`タグは空です。

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

JSONドキュメントに含まれるフィールド

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

バケットとオブジェクトの情報

`bucket`：バケットの名前

`key`：Object key name

`versionID`：バージョン管理されたバケット内のオブジェクトのバージョン

`region`：バケットリージョン（例： `us-east-1`）

システムメタデータ

`size`：HTTPクライアントから見えるオブジェクトサイズ（バイト単位）

`md5`：オブジェクトハッシュ

ユーザーメタデータ

`metadata`：オブジェクトに関するすべてのユーザーメタデータ（キーと値のペア形式）

`key:value`

Tags

`tags`：オブジェクトに定義されているすべてのオブジェクトタグ（キーと値のペア）

`key:value`

Elasticsearchで結果を表示する方法

タグとユーザーメタデータについては、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。Elasticsearchが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設定するには、Elasticsearchの動的フィールドマッピングおよび日付形式のマッピングに関する手順に従ってください。検索統合サービスを設定する前に、インデックス上で動的フィールドマッピングを有効にしてください。ドキュメントがインデックス化された後は、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。

S3 REST APIを使用する

StorageGRID S3 REST API でサポートされているバージョンとアップデート

StorageGRID は、Simple Storage Service （S3） API をサポートしており、一連の Representational State Transfer （REST） Web サービスとして実装されています。

S3 REST APIのサポートにより、S3 Webサービス向け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オブジェクトストレージに接続できます。クライアントアプリケーションで現在使用しているS3 REST API呼び出しへの変更は最小限で済みます。

サポート対象のバージョン

StorageGRID は、以下の特定のバージョンの STS、S3、および HTTP をサポートしています。

項目	**version**
STS AssumeRole API仕様	"Amazon Web Services (AWS) ドキュメント：Amazon Assumerole API リファレンス" AssumeRole の詳細については、"AssumeRole の設定" を参照してください。
S3 API仕様	"AWSドキュメント：Amazon Simple Storage Service API リファレンス"
HTTP	1.1 HTTPに関する詳細については、HTTP/1.1（RFCs 7230-35）を参照してください。 "IETF RFC 2616：ハイパーテキスト転送プロトコル（HTTP/1.1）" 注記：StorageGRID は HTTP/1.1 パイプライン処理をサポートしていません。

S3 REST APIサポートのアップデート

リリース	**Comments**
12.0	• 管理インターフェイスのクロスオリジンリソース共有（CORS）のサポートを追加しました。これにより、別のドメインが管理APIを使用してStorageGRID内のデータにアクセスできるようになります。"詳細情報"。 • STS AssumeRole およびセッションポリシーのサポートを追加しました。"セッションポリシーの例" を参照してください。テナントグループで AssumeRole を設定できます。

リリース	Comments
11.9	• 以下のリクエストおよびサポートされているヘッダーに対して、事前計算済みの SHA-256 チェックサム値のサポートを追加しました。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。 ◦ CompleteMultipartUpload： `x-amz-checksum-sha256` ◦ CreateMultipartUpload： `x-amz-checksum-algorithm` ◦ GetObject： `x-amz-checksum-mode` ◦ HeadObject： `x-amz-checksum-mode` ◦ ListParts ◦ PutObject： `x-amz-checksum-sha256` ◦ UploadPart： `x-amz-checksum-sha256` • グリッド管理者がテナントレベルのデータ保持およびコンプライアンス設定を制御できる機能を追加しました。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。 ◦ バケットのデフォルトの保持モードとオブジェクトの保持モード：グリッド管理者が許可している場合、Governance または Compliance。 ◦ 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終了日：グリッド管理者が設定した最大保持期間以下である必要があります。 • サポートの改善 `aws-chunked`コンテンツのエンコードとストリーミング `x-amz-content-sha256`値。制限事項： ◦ 存在する場合、 `chunk-signature`はオプションであり、検証は行われません ◦ 存在する場合、 `x-amz-trailer`コンテンツは無視されます
11.8	S3 操作の名前を、 "Amazon Web Services (AWS) ドキュメント：Amazon Simple Storage Service API リファレンス"で使用されている名前と一致するように更新しました。
11.7	• "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"を追加しました。 • S3オブジェクトロックでガバナンスモードを使用するためのサポートを追加しました。 • GET Object および HEAD Object リクエストに対する StorageGRID 固有の `x-ntap-sg-cgr-replication-status`レスポンスヘッダーのサポートを追加しました。このヘッダーは、クロスグリッドレプリケーションにおけるオブジェクトのレプリケーション状態を示します。 • SelectObjectContent リクエストが Parquet オブジェクトをサポートするようになりました。

リリース	Comments
11.6	• GET Object および HEAD Object リクエストにおける `partNumber`リクエストパラメータの使用のサポートを追加しました。 • S3オブジェクトロックにおいて、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を追加しました。 • `s3:object-lock-remaining-retention-days`ポリシー条件キーのサポートが追加され、オブジェクトの許容保持期間の範囲を設定できるようになりました。 • 単一の PUT Object 操作における_推奨_最大サイズを 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てください。
11.5	• バケット暗号化の管理機能を追加しました。 • S3オブジェクトロックのサポートを追加し、従来のコンプライアンスリクエストを廃止しました。 • バージョン管理されたバケットで複数のオブジェクトを削除する機能のサポートを追加しました。 • The `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。
11.4	• DELETE Bucket tagging、GET Bucket tagging、PUT Bucket tagging のサポートを追加しました。コスト配分タグはサポートされていません。 • StorageGRID 11.4 で作成されたバケットについては、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なくなりました。 • `s3:ObjectRestore:Post`イベントタイプのバケット通知のサポートが追加されました。 • AWSのマルチパートパーツの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以上5 GiB以下である必要があります。最後のパートは5 MiB未満でも構いません。 • TLS 1.3のサポートを追加しました。
11.3	• 顧客提供の鍵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）のサポートを追加しました。 • 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）と `x-amz-expiration`レスポンスヘッダーのサポートが追加されました。 • PUT Object、PUT Object - Copy、およびMultipart Uploadを更新し、取り込み時に同期配置を使用するILMルールの影響について説明を追加しました。 • TLS 1.1暗号はサポートされなくなりました。

リリース	Comments
11.2	クラウドストレージプールで使用するための、POST Object restoreのサポートを追加しました。グループポリシーおよびバケットポリシーにおいて、ARN、ポリシー条件キー、ポリシー変数に AWS 構文を使用できるようにサポートを追加しました。StorageGRID 構文を使用する既存のグループポリシー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を含め、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クライアント接続のためのHTTP、およびバケットのコンプライアンス設定のサポートを追加しました。
11.0	プラットフォームサービスの構成に対するサポートを追加しました（CloudMirrorレプリケーション、通知、およびElasticsearch検索統合）。また、バケットに対するオブジェクトタグ付け位置制約と、Available整合性のサポートも追加しました。
10.4	ILM スキャンのバージョン管理への変更、エンドポイントドメイン名ページの更新、ポリシー内の条件と変数、ポリシーの例、および PutOverwriteObject 権限に対するサポートを追加しました。
10.3	バージョン管理機能のサポートを追加しました。
10.2	グループおよびバケットのアクセスポリシー、ならびにマルチパートコピー（Upload Part - Copy）のサポートを追加しました。
10.1	マルチパートアップロード、仮想ホスト型リクエスト、およびv4認証のサポートを追加しました。
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 *Simple Storage Service API Reference* のバージョンは 2006-03-01 です。

StorageGRID で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説明します。

このページには、StorageGRID でサポートされている S3 オペレーションのみが含まれています。

各操作に関する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てください。

一般的な**URI**クエリパラメータとリクエストヘッダー

特に明記されていない限り、以下の一般的な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）

特に明記されていない限り、以下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 Reference：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、このリクエストに対してすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加 URI クエリパラメータもサポートしています：

- `uploadId`

要求の本文

None

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、このリクエストに対してすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加 URI クエリパラメータもサポートしています：

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

リクエストボディの**XML**タグ

StorageGRIDは以下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、このリクエストに対してすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加ヘッダーもサポートしています：

- `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>`

要求の本文

None

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、このリクエストに対してすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加ヘッダーもサポートしています：

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

要求の本文

StorageGRID は、実装時点で 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ています。

StorageGRID ドキュメント

"バケットに対する操作"

"CreateMultipartUpload"

URIクエリパラメータとリクエストヘッダー

StorageGRID は、このリクエストに対してすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加ヘッダーもサポートしています：

- `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>`

要求の本文

None

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 はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

StorageGRID ドキュメント

"バケットに対する操作"

"DeleteBucketCors"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"DeleteBucketEncryption"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"DeleteBucketLifecycle"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

- "バケットに対する操作"
- "S3ライフサイクル構成を作成する"

"DeleteBucketPolicy"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"DeleteBucketReplication"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"DeleteBucketTagging"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"DeleteObject"

URIクエリパラメータとリクエストヘッダー

StorageGRID は、このリクエストに対するすべての共通パラメータとヘッダーに加え、以下の追加リクエストヘッダーをサポートしています：

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

要求の本文

None

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、このリクエストに対するすべての共通パラメータとヘッダーに加え、以下の追加リクエストヘッダーをサポートしています：

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

要求の本文

StorageGRID は、実装時点で 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

タをサポートしています。

StorageGRID ドキュメント
"オブジェクトに対する操作"

"DeleteObjectTagging"

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文
None

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 はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文
None

StorageGRID ドキュメント
"バケットに対する操作"

"GetBucketCors"

URIクエリパラメータとリクエストヘッダー
StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文
None

StorageGRID ドキュメント
"バケットに対する操作"

"GetBucketEncryption"

URIクエリパラメータとリクエストヘッダー
StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文
None

StorageGRID ドキュメント
"バケットに対する操作"

"GetBucketLifecycleConfiguration"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

- "バケットに対する操作"
- "S3ライフサイクル構成を作成する"

"GetBucketLocation"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"GetBucketNotificationConfiguration"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"GetBucketPolicy"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"GetBucketReplication"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"GetBucketTagging"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"GetBucketVersioning"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"GetObject"

URIクエリパラメータとリクエストヘッダー

StorageGRID は、このリクエストに対してすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加 URI クエリパラメータもサポートしています：

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

追加のリクエストヘッダーは次のとおりです：

- `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`

要求の本文
None

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 はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文
None

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 はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文
None

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 はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文
None

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 はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

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 はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

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 はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"HeadObject"

URIクエリパラメータとリクエストヘッダー

StorageGRID は、このリクエストに対してすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加ヘッダーもサポートしています：

- `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`

要求の本文

None

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 はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

None

StorageGRID ドキュメント

サービス上での操作 › **ListBuckets**

"ListMultipartUploads"

URIクエリパラメータとリクエストヘッダー

StorageGRID は、このリクエストに対するすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加パラメータもサポートしています：

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

要求の本文

None

StorageGRID ドキュメント

"ListMultipartUploads"

"ListObjects"

URIクエリパラメータとリクエストヘッダー

StorageGRID は、このリクエストに対するすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加パラメータもサポートしています：

- `delimiter`
- `encoding-type`

- marker
- max-keys
- prefix

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"ListObjectsV2を使用したチャンク アップロード署名要求がサポートされるようになりました。**"**

URIクエリパラメータとリクエストヘッダー

StorageGRID は、このリクエストに対するすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加パラメータもサポートしています：

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

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"ListObjectVersions"

URIクエリパラメータとリクエストヘッダー

StorageGRID は、このリクエストに対するすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加パラメータもサポートしています：

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

要求の本文

None

StorageGRID ドキュメント

"バケットに対する操作"

"ListParts"

URIクエリパラメータとリクエストヘッダー

StorageGRID は、このリクエストに対するすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加パラメータもサポートしています：

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

要求の本文

None

StorageGRID ドキュメント

"ListMultipartUploads"

"PutBucketCors"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

StorageGRID は、実装時点で 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ています。

StorageGRID ドキュメント

"バケットに対する操作"

"PutBucketEncryption"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

リクエストボディの**XML**タグ

StorageGRIDは以下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

"バケットに対する操作"

"PutBucketLifecycleConfiguration"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

リクエストボディの**XML**タグ

StorageGRIDは以下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

- "バケットに対する操作"
- "S3ライフサイクル構成を作成する"

"PutBucketNotificationConfiguration"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

リクエストボディの**XML**タグ

StorageGRIDは以下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

"バケットに対する操作"

"PutBucketPolicy"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

サポートされている JSON ボディ フィールドの詳細については、"バケットおよびグループのアクセスポリシーを使用する"を参照してください。

"PutBucketReplication"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

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

"バケットに対する操作"

"PutBucketTagging"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

StorageGRID は、実装時点で 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ています。

StorageGRID ドキュメント

"バケットに対する操作"

"PutBucketVersioning"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

リクエストボディのパラメータ

StorageGRID は、以下のリクエストボディパラメータをサポートしています：

- `VersioningConfiguration`
- `Status`

StorageGRID ドキュメント

"バケットに対する操作"

"PutObject"

URIクエリパラメータとリクエストヘッダー

StorageGRID は、このリクエストに対してすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加ヘッダーもサポートしています：

- `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>`

要求の本文

- 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 はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

StorageGRID は、実装時点で 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ています。

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 はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

StorageGRID は、実装時点で 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ています。

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、このリクエストに対するすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加ヘッダーもサポートしています：

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

要求の本文

StorageGRID は、実装時点で 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ています。

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 はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

StorageGRID は、実装時点で 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ています。

StorageGRID ドキュメント

"オブジェクトに対する操作"

"RestoreObject"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

サポートされている本文フィールドの詳細については、"RestoreObject"を参照してください。

"SelectObjectContent"

URIクエリパラメータとリクエストヘッダー

StorageGRID はこのリクエストに対して、すべての共通パラメータとヘッダーをサポートしています。

要求の本文

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"
- "SelectObjectContent"

"UploadPart"

URIクエリパラメータとリクエストヘッダー

StorageGRID は、このリクエストに対してすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加 URI クエリパラメータもサポートしています：

- `partNumber`
- `uploadId`

追加のリクエストヘッダーは次のとおりです：

- `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`

要求の本文

- 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、このリクエストに対してすべての共通パラメータとヘッダーをサポートしており、さらに以下の追加 URI クエリパラメータもサポートしています：

- `partNumber`
- `uploadId`

追加のリクエストヘッダーは次のとおりです：

- `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`

要求の本文

None

StorageGRID ドキュメント

"UploadPartCopy"

StorageGRID S3 REST API 構成をテストします

Amazon Web Services コマンドラインインターフェイス （AWS CLI） を使用すると、システムへの接続をテストしたり、オブジェクトの読み書きが可能かどうかを確認したりできます。

開始する前に

- AWS CLI を "aws.amazon.com/cli"からダウンロードしてインストールしました。
- オプションとして、"ロードバランサーエンドポイントを作成しました"。それ以外の場合は、接続先の Storage Node の IP アドレスと使用するポート番号がわかっている必要があります。"クライアント接続用のIPアドレスとポート"を参照してください。
- "S3テナントアカウントを作成しました"があります。
- 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"を参照してください。

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 の設定を構成します：
   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を入力してください。
   c. 作成したアカウントの秘密アクセスキーを入力してください。
   d. 使用するデフォルトのリージョンを入力してください。例えば、 `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 を選択してください。
2. バケットを作成

   この例では、ロードバランサーのエンドポイントをIPアドレス10.96.101.17とポート10443を使用するように構成済みであ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た場合、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値であるEtagが返されます。

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 による S3 REST API の実装方法

StorageGRID S3 REST APIが競合するクライアント要求を解決する方法

2つのクライアントが同じキーに書き込むなど、クライアントのリクエストが競合する場合は、「最新のリクエストが優先される」方式で解決されます。

「最新優先」評価のタイミングは、S3クライアントが操作を開始したときではなく、StorageGRID システムが特定のリクエストを完了したときを基準とします。

StorageGRID S3 REST APIが可用性と一貫性のバランスをとる方法

一貫性とは、オブジェクトの可用性と、異なるストレージノードやサイト間でのオブジェクトの一貫性とのバランスを取るものです。アプリケーションの要件に応じて、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について、書き込み後の読み取りの一貫性を保証します。PUT が正常に完了した後に行われる GET リクエストは、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、最終的に整合性が保たれます。

オブジェクト操作を異なる整合性で実行する場合は、次の操作を実行できます。

- 各バケットの整合性を指定します。
- 各API操作の一貫性を指定します。
- 以下のいずれかの操作を実行して、グリッド全体のデフォルトの整合性を変更します。
  - Grid Manager で、**Configuration** > **System** > **Storage settings** > **Default bucket consistency** に移動します。
  - 。

グリッド全体の整合性に関する変更は、設定変更後に作成されたバケットにのみ適用されます。変更の詳細を確認するには、 `/var/local/log`にある監査ログを参照してください（**consistencyLevel** を検索）。

一貫性値

一貫性は、StorageGRIDがオブジェクトの追跡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ます。一貫性は、クライアントリクエスト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操作の一貫性は、以下のいずれかの値に設定できます：

- すべて：すべてのノードがオブジェクトメタデータを即座に受信します。受信しない場合、リクエストは失敗します。
- **Strong-global**: すべてのサイトにおけるすべてのクライアントリクエストに対して、書き込み後の読み取りの一貫性を保証します。Quorumのセマンティクスが設定されている場合、以下の動作が適用されます：
  - グリッドに3つ以上のサイトがある場合、クライアント要求に対するサイト障害の耐性を確保します。2サイトグリッドにはサイト障害の耐性はありません。
  - 1つのサイトがダウンしている場合、以下のS3操作は成功しません：
    - DeleteBucketEncryption
    - PutBucketBranch
    - PutBucketEncryption
    - PutBucketVersioning
    - PutObjectLegalHold
    - PutObjectLockConfiguration
    - PutObjectRetention

  必要であれば、 "StorageGRID のクォーラムセマンティクスをstrong-global整合性に設定する"。
- **Strong-site**: オブジェクトのメタデータは、サイト内の他のノードに即座に配信されます。サイト内のすべてのクライアントリクエストに対して、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：新規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結果整合性を提供します。高可用性とデータ保護を保証します。ほとんどの場合に推奨されます。
- 利用可能：新規オブジェクトとオブジェクト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、または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」の整合性を使用する

HEAD または GET 操作で「Read-after-new-write」整合性を使用する場合、StorageGRID は以下の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、次の整合性値で検索を繰り返し、strong-global の動作と同等の整合性に達する

まで続けます。

HEAD または GET 操作で「新規書き込み後の読み取り」整合性が使用されている場合でも、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 の動作と同等の整合性に達します。この整合性を維持するには、各サイトにオブジェクトメタデータの複数のコピーが必要となるため、同じサイトにある 2 つ以上のストレージノードが利用できない場合、500 Internal Server Error が多数発生する可能性があります。

Amazon S3 と同様の整合性保証が不要な場合は、整合性を「Available」に設定することで、HEAD および GET 操作でこれらのエラーが発生するのを防ぐことができます。HEAD または GET 操作で「Available」整合性を使用する場合、StorageGRID は最終的な整合性のみを提供します。失敗した操作をより高い整合性で再試行しないため、オブジェクトメタデータの複数のコピーを用意する必要はありません。

API操作の整合性の指定

個々のAPI操作の一貫性を設定するには、その操作で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 と GetObject の両方の操作に同じ整合性を使用する必要があります。

バケットの整合性を指定する

バケットの整合性を設定するには、StorageGRID "PUT Bucketの一貫性" リクエストを使用します。または、Tenant Manager から "バケットの整合性を変更する" することもできます。

バケットの整合性を設定する際には、以下の点に注意してください。

- バケットの整合性を設定することで、バケット内のオブジェクトまたはバケット構成に対して実行されるS3操作に使用される整合性が決まります。これは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。
- 一般的に、バケットはデフォルトの整合性設定である「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てください。または、クライアント側で各 API リクエストの整合性を指定するように設定することもできます。バケットレベルでの整合性の設定は、最終手段としてのみ行ってください。

一貫性とILMルールがデータ保護に与える影響

一貫性の選択とILMルールは、どちらもオブジェクトの保護方法に影響を与えます。これらの設定は相互に影響し合う可能性があります。

例えば、オブジェクトを保存する際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を与え、ILMルールで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を与えます。StorageGRID が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データの両方へのアク

セスを必要とするため、一貫性と取り込み動作に対して適切なレベルの保護を選択することで、より優れた初期データ保護とより予測可能なシステム応答を実現できます。

次の"取り込みオプション"は、ILMルールで使用できます：

Dual commit

StorageGRID はオブジェクトの暫定コピーを即座に作成し、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。

厳格

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

バランス

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。そ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ルールと次の整合性を持つ3サイトグリッドがあるとします。

- **ILM**ルール：オブジェクトのコピーを3つ作成します。1つはローカルサイトに、残りの2つは各リモートサイトにそれぞれ1つずつ作成します。厳密な取り込み動作を使用します。
- 一貫性：Strong-global（オブジェクトのメタデータは複数のサイトに即座に配信されます）。

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 はオブジェクトのコピーを3つすべて作成し、メタデータを複数のサイトに配布してから、クライアントに成功を返します。

取り込み成功メッセージの時点で、オブジェクトは損失から完全に保護されます。例えば、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。オブジェクトは他のサイトから完全に取得可能です。

代わりに同じILMルールとstrong-siteの整合性を使用した場合、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、オブジェクトの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、オブジェクトの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。

整合性とILMルールの相互関係は複雑になる場合があります。サポートが必要な場合は、NetAppにお問い合わせください。

StorageGRIDでのオブジェクトバージョン管理の使用方法

各オブジェクトの複数のバージョンを保持したい場合は、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、オブジェクトの誤削除を防ぎ、オブジェクトの以前のバージョンを取得および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、各オブジェクトにつき最大 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バケットのライフサイクル構成と組み合わせることができます。各バケットでバージョン管理を明示的に有効にする必要があります。バケットで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バージョン ID が割り当てられます。このバージョン ID は StorageGRID システムによって生成されます。

多要素認証（MFA）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ポリシーに基づいて再評価します。ILMポリシーに加えた変更は、既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、これに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になっているバケット内の 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、参照時間として「非現在の時間」を使用する ILM ルールを作成できます（"ILMルール作成ウィザードのステップ1"の「このルールを古いオブジェクトバージョンのみに適用しますか？」という質問に対して はい を選択します）。オブジェクトが更新されると、その以前のバージョンは最新ではなくなります。「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がストレージに与える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場合、元のバージョンのオブジェクトの非現在時刻は、マルチパートアップロードが完了した時刻ではなく、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時刻を反映します。限られたケースでは、元のバージョンの非現在時刻が、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場合があります。

関連情報

- "S3 バージョン管理されたオブジェクトはどのように削除されるのか"
- "S3バージョン管理対象オブジェクトに関するILMルールとポリシー（例4）"。

S3 REST APIを使用してStorageGRID S3 Object Lockを設定する

グローバル S3 オブジェクトロック設定が StorageGRID システムで有効になっている場合、S3 オブジェクトロックを有効にしたバケットを作成できます。各バケットのデフォルト保持期間を指定したり、オブジェクトのバージョンごとに保持設定を指定したりできます。

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、各バケットを作成する際にオプションで S3 オブジェクトロックを有効にすることができます。

S3オブジェクトロックは、バケット作成時にのみ有効にできる永続的な設定です。バケット作成後に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。

バケットの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。"S3バケットを作成する"を参照してください。
- CreateBucket リクエストと `x-amz-bucket-object-lock-enabled`リクエストヘッダーを使用してバケットを作成します。"バケットに対する操作"を参照してください。

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あり、これはバケット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。"オブジェクトのバージョン管理"を参照してください。

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、オプションでバケットのデフォルト保持を有効にし、デフォルト保持モードとデフォルト保持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、保持期限が到来するまで削除できません。
  - オブジェクトの保持期限は延長できますが、短縮することはできません。
  - オブジェクトの保持期限日は、その日付が到来するまで削除することはできません。
- ガバナンスモードでは：
  - `s3:BypassGovernanceRetention`権限を持つユーザーは、 `x-amz-bypass-governance-retention: true`リクエストヘッダーを使用して保持設定をバイパスできます。
  - これらのユーザーは、オブジェクトのバージョンが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、年または日単位で指定できる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。"S3バケットを作成する"および"S3オブジェクトロックのデフォルト保持期間を更新する"を参照してください。
- バケットに対して PutObjectLockConfiguration リクエストを発行し、デフォルト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。以前に設定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、 `x-amz-object-lock-mode`および `x-amz-object-lock-retain-until-date`が指定されていない場合、デフォルトの保持モードが適用されます。 `x-amz-

object-lock-retain-until-date`が指定されていない場合、デフォルトの保持期間を使用して保持期限日が計算されます。

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変更されず、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。

この操作を完了するには、 `s3:PutBucketObjectLockConfiguration`権限が必要です。またはアカウントのrootである必要があります。

> `Content-MD5`リクエストヘッダーは、PUTリクエストで指定する必要があります。

リクエスト例

この例では、バケットに対して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、また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確認してください。"S3バケットを表示する"を参照してください。
- GetObjectLockConfiguration リクエストを発行します。

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でき、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、 `x-amz-object-lock-mode`が指定されていない場合、デフォルトの保持モードが適用されます。デフォルトの保持期間は、 `x-amz-object-lock-retain-until-date`が指定されていない場合に保持期限日を計算するために使用されます。

この操作を完了するには、 `s3:GetBucketObjectLockConfiguration`権限が必要です。またはアカウントの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
```

回答例

```
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>
```

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適用されているオブジェクトと適用されていないオブジェクトが混在して格納される可能性があります。

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、以下の設定を指定できます。

- 保持モード：COMPLIANCE または GOVERNANCE のいずれか。
- 保持期限：オブジェクトのバージョンを 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、保持期限が未来の日付の場合、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。保持期限は延長できますが、短縮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、保持期限設定をバイパスできます。オブジェクトのバージョンは、保持期間が経過する前に削除することができます。また、保存期限を延長、短縮、あるいは削除することも可能です。
- 法的保留：オブジェクトの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即座にロックされます。例えば、調査や法的紛争に関連するオブジェクトに対して、法的保留を適用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なく、明示的に解除されるまで有効です。

  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独立しています。オブジェクトのバージョンが法的保留状態にある場合、誰もそのバージョンを削除することはできません。

バケットにオブジェクトバージョンを追加する際に S3 Object Lock 設定を指定するには、"PutObject"、"CopyObject"、または "CreateMultipartUpload" リクエストを発行します。

以下のものを使用できます。

- `x-amz-object-lock-mode`これは、COMPLIANCE または GOVERNANCE のいずれかになります（大文字と小文字を区別します）。

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

- `x-amz-object-lock-retain-until-date`
  - 保持期限の値は、次の形式である必要があります `2020-08-10T21:46:00Z`。小数秒も許容されますが、保持されるのは小数点以下3桁（ミリ秒単位の精度）のみです。その他のISO 8601形式は許可されていません。
  - 保持期限日は未来の日付でなければなりません。
- `x-amz-object-lock-legal-hold`

  法的保留がオン（大文字小文字を区別）の場合、対象物は法的保留下に置かれます。法的保留がオフの場合、法的保留は行われません。その他の値では、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を使用する場合は、以下の制限事項に注意してください：

- `Content-MD5`リクエストヘッダーは、 `x-amz-object-lock-*`リクエストヘッダーがPutObjectリクエストに存在する場合に必須です。 `Content-MD5`はCopyObjectまたはCreateMultipartUploadには必要ありません。
- バケットで S3 Object Lock が有効になっておらず、 `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、AWSの動作に合わせるために `x-amz-storage-class: REDUCED_REDUNDANCY`の使用をサポートしています。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を取り込む場合、StorageGRIDは常にデュアルコミット取り込みを実行します。
- その後のGETまたはHeadObjectバージョンの応答には、設定されており、かつリクエスト送信者が適切な `s3:Get*`権限を持っている場合、ヘッダー `x-amz-object-lock-mode`、 `x-amz-object-lock-retain-until-date`、および `x-amz-object-lock-legal-hold`が含まれます。

  `s3:object-lock-remaining-retention-days`ポリシー条件キーを使用して、オブジェクトの最小および最大許容保持期間を制限できます。

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ONの場合、対象物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のいずれかになります（大文字と小文字を区別します）。
  - 保持期限の値は、次の形式である必要があります `2020-08-10T21:46:00Z`。小数秒も許容されますが、保持されるのは小数点以下3桁（ミリ秒単位の精度）のみです。その他のISO 8601形式は許可されていません。
  - オブジェクトのバージョンに保持期限が設定されている場合、その期限を延長することしかできません。新しい値は将来の日付である必要があります。

ガバナンスモードの使用方法

ユーザーは `s3:BypassGovernanceRetention`権限によって、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をバイパスすることができます。DELETE または PutObjectRetention 操作には、 `x-amz-bypass-governance-retention:true`リクエストヘッダーを含める必要があります。これらのユーザーは、以下の追加操作を実行できます：

- DeleteObject または DeleteObjects 操作を実行して、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します。

  法的拘束を受けている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。

  コンプライアンスモードからガバナンスモードへの変更は一切許可されていません。

- PutObjectRetention 操作を実行して、オブジェクトバージョンの保持期間を延長、短縮、または削除します。

関連情報

- "S3 Object Lock 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"

StorageGRIDのS3ライフサイクル設定について学ぶ

S3ライフサイクル構成を作成して、特定のオブジェクトが StorageGRID システムから削除されるタイミングを制御できます。

このセクションの簡単な例では、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削除（期限切れ）されるタイミングをどのように制御できるかを示しています。このセクションの例は、あくまで説明のためのものです。S3ライフサイクル設定の作成に関する詳細については、 "Amazon Simple Storage Service ユーザーガイド：オブジェクトライフサイクル管理"を参照してください。StorageGRID は Expiration アクションのみをサポートしており、Transition アクションはサポートしていません。

ライフサイクル構成とは

ライフサイクル設定とは、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のことです。各ルールは、どのオブジェクトが影響を受けるか、そしてそれらのオブジェクトがいつ期限切れになるか（特定の日付、または一定日数後）を指定します。

StorageGRID はライフサイクル構成において、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、以下のXML要素を含めることができます。

- 有効期限：オブジェクトが取り込まれた時点から、指定された日付に達したとき、または指定された日数が経過したときに、オブジェクトを削除します。
- NoncurrentVersionExpiration：オブジェクトが非現行状態になった時点から指定された日数が経過したら、その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ポリシーのいずれか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、バケットライフサイクルフィルタに一致するオブジェクトについては、ライフサイクルの有効期限切れアクションがILMポリシーを上書きします。バケットライフサイクルフィルタ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ライフサイクルフィルタに一致し、かつ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、オブジェクトのバージョンは永久に保持されるとみなされます。"S3バケットのライフサイクルとILMポリシーの優先順位の例"を参照してください。

その結果、ILMルール内の配置指示がオブジェクトに依然として適用されている場合でも、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。あるいは、オブジェクトに対するILM配置指示の有効期限が切れた後でも、オブジェクトがグリッド上に保持される場合があります。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を通してどのように機能するか"を参照してください。

> バケットライフサイクル設定は、S3 Object Lock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、ライフサイクル設定を管理するために以下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ステップとして、1つ以上のルールを含むJSONファイルを作成します。例えば、このJSONファイルには以下の3つのルールが含まれています。

1. ルール1は、プレフィックス `category1/` に一致し、 `key2`の値が `tag2`であるオブジェクトにのみ適用されます。 `Expiration`パラメータは、フィルターに一致するオブジェクトが2020年8月22日午前0時に期限切れになることを指定します。
2. ルール2は、プレフィックス `category2/`に一致するオブジェクトにのみ適用されます。 `Expiration`パラメータは、フィルタ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付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、プレフィックス `category3/`に一致するオブジェクトにのみ適用されます。 `Expiration`パラメータは、一致するオブジェクトの非現行バージョンが非現行になってから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"
        }
    ]
}
```

ライフサイクル構成ファイルを作成したら、PutBucketLifecycleConfiguration リクエストを発行してバケットに適用します。

このリクエストは、サンプルファイル内のライフサイクル構成を、 `testbucket`という名前のバケット内のオブジェクトに適用します。

```
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：

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

正常に応答すると、適用したライフサイクル構成が表示されます。

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ることを確認します。

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は、PutObject 、HeadObject、またはGetObjectリクエストの発行時に判断できます。ルールが適用される場合、応答にはオブジェクトの有効期限が切れるタイミングと一致した有効期限ルールを示す `Expiration`パラメータが含まれます。

> バケットライフサイクルはILMを上書きするため、 `expiry-date`表示されているのは、オブジェクトが実際に削除される日付です。詳細については、"オブジェクト保持期間の決定方法"を参照してください。

たとえば、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`バケットに配置します。

```
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）に期限切れとなり、ライフサイクル構成のルール2に一致したことを示しています。

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

例えば、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、およびRule 2に一致したことが示されています。

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

バージョン管理が有効になっているバケットの場合、 `x-amz-expiration`レスポンスヘッダーは、オブジェクトの最新バージョンにのみ適用されます。

StorageGRIDでS3 REST APIを実装する際の推奨事項

StorageGRID で使用するための S3 REST API を実装する際には、以下の推奨事項に従ってください。

存在しないオブジェクトへの**HEAD**に関する推奨事項

アプリケーションが、オブジェクトが実際には存在しないと想定さ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チェックする場合は、「Available」"一貫性"を使用する必要があります。例えば、アプリケーションがPUTする前にHEADで場所を確認する場合は、「Available」の整合性を使用する必要があります。

それ以外の場合、HEAD操作でオブジェクトが見つからない場合、同じサイトにある2つ以上のストレージノードが利用できない場合、またはリモートサイトにアクセスできない場合は、多数の500 Internal Server Errorが発生する可能性があります。

各バケットの「利用可能」整合性は、"PUT Bucketの一貫性"リクエストを使用して設定するか、個々のAPI操

作のリクエストヘッダーで整合性を指定することができます。

オブジェクトキーに関する推奨事項

オブジェクトキー名については、バケットが最初に作成された時期に基づいて、以下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文字にランダムな値を使用しないでください。これは、以前のAWSのキープレフィックスに関する推奨事項とは対照的です。代わりに、 `image`のような非ランダムで非一意のプレフィックスを使用してください。
- 以前のAWSの推奨事項に従って、キーのプレフィックスにランダムで一意の文字を使用する場合は、オブジェクトキーのプレフィックスにディレクトリ名を追加してください。つまり、この形式を使用してください：

  ```
  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、オブジェクトの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。

> ただし、短時間後にすべてのオブジェクトを継続的に削除するS3ワークロードは例外です。このユースケースにおけるパフォーマンスへの影響を最小限に抑えるため、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の値で変更してください。例えば、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/yyyymmddhhmmss-random_UUID.jpg`

「範囲読み取り」に関する推奨事項

"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、返すバイト範囲を指定する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」操作は非効率的です。これは、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です。非常に大きなオブジェクトから少数のバイト範囲を要求するGetObject操作は特に非効率的です。たとえば、50 GBの圧縮オブジェクトから10 MBの範囲を読み取ることは非効率的です。

圧縮されたオブジェクトから範囲を読み取る場合、クライアントからのリクエスト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が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長くしてください。

Amazon S3 REST APIのサポート

StorageGRID における **S3 REST API** のサポートされている操作と制限

StorageGRID システムは、Simple Storage Service API（API Version 2006-03-01）を実装しており、ほとんどの操作をサポートしていますが、一部制限があります。S3 REST API クライアントアプリケーションを統合する際には、実装の詳細を理解しておく必要があります。

StorageGRIDシステムは、仮想ホスト形式のリクエストとパス形式のリクエストの両方をサポートしています。

日付処理

StorageGRID の S3 REST API の実装は、有効な HTTP 日付形式のみをサポートします。

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。日付の時刻部分は、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協定世界時（UTC）形式（+0000 を指定する必要があります）で指定できます。リクエストに `x-amz-date`ヘッダーを含めると、Date リクエストヘッダーで指定された値よりも優先されます。AWS Signature Version 4 を使用する場合、日付ヘッダーはサポートされていないため、署名付きリクエストには `x-amz-date`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、 "Amazon Simple Storage Service API Reference：共通リクエストヘッダー"で定義される一般的なリクエストヘッダーをサポートしていますが、例外が1つあります。

リクエストヘッダー	導入
許可	AWS Signature Version 2 の完全サポート AWS Signature Version 4 をサポートします。ただし、以下の例外があります。 • 実際のペイロードのチェックサム値を `x-amz-content-sha256`に指定すると、その値は検証なしで受け入れられます。これは、値 `UNSIGNED-PAYLOAD`がヘッダーに指定された場合と同様です。`x-amz-content-sha256`ヘッダー値にストリーミング（例：STREAMING-AWS4-HMAC-SHA256-PAYLOAD）を意味する `aws-chunked`値を指定した場合、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。

一般的なレスポンスヘッダー

StorageGRID システムは、*Simple Storage Service API Reference* で定義されているすべての一般的なレスポンスヘッダーをサポートしていますが、例外が1つあります。

レスポンスヘッダー	導入
x-amz-id-2	未使用

StorageGRID S3 REST APIによるリクエストの認証方法

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

S3 APIは、S3 APIリクエストの認証において、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。

認証済みリクエストは、アクセスキーIDと秘密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 `Authorization`ヘッダーとクエリパラメータの使用という2つの認証方法をサポートしています。

HTTP Authorization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アクセスリクエストを除き、すべてのS3 API操作で使用されます。 `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すると、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。署名付き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、サードパーティに対してリソースへのアクセスを制限できます。

StorageGRID でサポートされているサービス操作

StorageGRID システムは、サービスに対して以下の操作をサポートしています。

処理	導入
ListBuckets （以前は GET Service と呼ばれていました）	すべての Amazon S3 REST API の動作を実装済み。予告なく変更される場合があります。
ストレージ使用量を取得する	StorageGRID "ストレージ使用量を取得する" リクエストは、アカウントが使用しているストレージの総量と、その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容量を示します。これは、パスが / でカスタムクエリパラメータ (`?x-ntap-sg-usage`) が追加されたサービスに対する操作です。
OPTIONS /	クライアントアプリケーションは、S3 認証クレデンシャルを提供せずにストレージノードの S3 ポートへ `OPTIONS /`リクエストを発行して、ストレージノードが利用可能かどうかを確認できます。このリクエストは、監視目的、または外部ロードバランサーがストレージノードのダウンを検知できるようにするために使用できます。

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個のバケットを設定できます。

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最低64 GBのRAMが必要です。

バケット名の制限は AWS US Standard リージョンの制限に従いますが、S3 仮想ホスト型リクエストをサポートするために、さらに DNS 命名規則に基づいて制限する必要があります。

詳細については、次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、および制約"
- "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）操作は StorageGRID "一貫性値"をサポートしています。

個々のバケットについて、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。"GET Bucketの最終アクセス時間"を参照してください。

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認証情報を提供する必要があります。

処理	導入
CreateBucket	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。 • バケット名は、以下の規則に従う必要があります： ◦ 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だけでなく）。 ◦ DNS規格に準拠している必要があります。 ◦ 3文字以上63文字以内である必要があります。 ◦ 1つまたは複数のラベルの連続で、隣接するラベルはピリオドで区切られます。各ラベルは小文字の英字または数字で始まり、小文字の英字または数字で終わる必要があり、使用できる文字は小文字の英字、数字、およびハイフンのみです。 ◦ テキスト形式のIPアドレスのように見えてはなりません。 ◦ 仮想ホスト型リクエストではピリオドを使用しないでください。ピリオドは、サーバーのワイルドカード証明書の検証に問題を引き起こします。 • デフォルトでは、バケットは `us-east-1`リージョンに作成されます。ただし、リクエスト本文の `LocationConstraint`リクエスト要素を使用して、別のリージョンを指定することができます。 `LocationConstraint`要素を使用する場合は、Grid Manager または Grid Management 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。使用すべきリージョン名がわからない場合は、システム管理者にお問い合わせください。 注：CreateBucketリクエストで StorageGRID に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。 • `x-amz-bucket-object-lock-enabled`リクエストヘッダーを含めることで、S3 Object Lock を有効にしたバケットを作成できます。"S3 REST APIを使用してS3オブジェクトロックを設定する"を参照してください。 バケットを作成する際には、S3 Object Lockを有効にする必要があります。バケット作成後にS3 Object Lockを追加または無効にすることはできません。S3 Object Lockではバケットのバージョン管理が必要ですが、これはバケットを作成する際に自動的に有効になります。
DeleteBucket	バケットを削除します。
DeleteBucketCors	バケットのCORS設定を削除します。
DeleteBucketEncryption	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、バケットに新しく追加されたオブジェクトは暗号化されません。
DeleteBucketLifecycle	バケットからライフサイクル設定を削除します。"S3ライフサイクル構成を作成する"を参照してください。

処理	導入
DeleteBucketPolicy	バケットに紐付けられているポリシーを削除します。
DeleteBucketReplication	バケットに紐付けられているレプリケーション構成を削除します。
DeleteBucketTagging	`tagging`サブリソースを使用して、バケットからすべてのタグを削除します。 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、値が割り当てられた `NTAP-SG-ILM-BUCKET-TAG`バケットタグが存在します。 `NTAP-SG-ILM-BUCKET-TAG`バケットタグが存在する場合は、DeleteBucketTaggingリクエストを発行しないでください。代わりに、`NTAP-SG-ILM-BUCKET-TAG`タグとその割り当てられた値のみを指定してPutBucketTaggingリクエストを発行し、バケットから他のすべてのタグを削除してください。 `NTAP-SG-ILM-BUCKET-TAG`バケットタグを変更または削除しないでください。
GetBucketAcl	肯定的な応答と、バケット所有者のID、DisplayName、および権限を返します。これは、所有者がバケットへの完全なアクセス権を持っていることを示します。
GetBucketCors	バケットの `cors`設定を返します。
GetBucketEncryption	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 lifecycle 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。"S3ライフサイクル構成を作成する"を参照してください。
GetBucketLocation	`LocationConstraint`要素を使用してCreateBucketリクエストで設定されたリージョンを返します。バケットのリージョンが `us-east-1`の場合、リージョンには空の文字列が返されます。
GetBucketNotificationConfiguration （以前はGET Bucket notificationという名称でした）	バケットに添付されている通知設定を返します。
GetBucketPolicy	バケットに添付されているポリシーを返します。

処理	導入
GetBucketReplication	バケットに添付されているレプリケーション構成を返します。
GetBucketTagging	`tagging`サブリソースを使用して、バケットのすべてのタグを返します。 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、値が割り当てられた `NTAP-SG-ILM-BUCKET-TAG`バケットタグが存在します。このタグを変更または削除しないでください。
GetBucketVersioning	この実装では、 `versioning`サブリソースを使用してバケットのバージョン管理状態を返します。 • *blank*: バージョン管理が有効になっていません（バケットは「バージョン管理なし」です） • 有効：バージョン管理が有効になっています • 一時停止：バージョン管理は以前は有効でしたが、現在は一時停止されています
GetObjectLockConfiguration	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が設定されている場合は、それらを返します。 "S3 REST APIを使用してS3オブジェクトロックを設定する"を参照してください。
HeadBucket	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。 この操作は以下を返します： • `x-ntap-sg-bucket-id`：UUID形式のバケットのUUID。 • `x-ntap-sg-trace-id`：関連するリクエストの一意のトレースID。

処理	導入
ListObjectsおよびListObjectsV2 （以前はGET Bucketという名称でした）	バケット内のオブジェクトの一部または全部（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を使用して取り込まれた場合でも、2つの値のいずれかを持つことができます： • `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格納されていることを示します。 • `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。 バケットに同じプレフィックスを持つ削除済みキーが多数含まれている場合、レスポンスにはキーを含まない `CommonPrefixes`が含まれる場合があります。 HeadObject および ListObject リクエストの場合、StorageGRID は以下の例に示すように異なる精度で LastModified タイムスタンプを返しますが、AWS は同じ精度でタイムスタンプを返します。 • StorageGRID HeadObject: 「LastModified」: 「2024-09-26T16:43:24+00:00」 • StorageGRID ListObject: 「LastModified」: 「2024-09-26T16:43:24.931000+00:00」 • AWS HeadObject："LastModified"："2023-10-17T00:19:54+00:00" • AWS ListObject："LastModified"："2023-10-17T00:19:54+00:00"
ListObjectVersions （以前は GET Bucket Object versions と呼ばれていました）	バケットへの READ アクセスがある場合、この操作を `versions`サブリソースとともに使用すると、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が一覧表示されます。
PutBucketCors	バケットのCORS設定を行い、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します。クロスオリジンリソース共有（CORS）は、あるドメインのクライアントWebアプリケーションが別のドメイン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。例えば、グラフィックを保存するために `images`という名前のS3バケットを使用するとします。 `images`バケットのCORS設定を行うことで、そのバケット内の画像をウェブサイト `http://www.example.com`に表示させることができます。

処理	導入
PutBucketEncryption	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が有効になっている場合、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 が管理するキーを使用したサーバー側暗号化をサポートします。サーバー側の暗号化設定ルールを指定する場合は、 `SSEAlgorithm`パラメータを `AES256`に設定し、 `KMSMasterKeyID`パラメータは使用しないでください。 オブジェクトアップロード要求で既に暗号化が指定されている場合（つまり、要求に `x-amz-server-side-encryption-*`リクエストヘッダーが含まれている場合）、バケットのデフォルトの暗号化設定は無視されます。
PutBucketLifecycleConfiguration （以前は PUT Bucket lifecycle と呼ばれていました）	バケットの新しいライフサイクル構成を作成するか、既存のライフサイクル構成を置き換えます。StorageGRIDはライフサイクル構成において、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、以下のXML要素を含めることができます： • 有効期限（日数、日付、ExpiredObjectDeleteMarker） • NoncurrentVersionExpiration（NewerNoncurrentVersions、NoncurrentDays） • フィルター（プレフィックス、タグ） • ステータス • ID StorageGRID は以下のアクションをサポートしていません： • AbortIncompleteMultipartUpload • 移行 "S3ライフサイクル構成を作成する"を参照してください。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、"ILMがオブジェクトのライフサイクル全体を通してどのように機能するか"を参照してください。 注：バケットライフサイクル構成は、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ライフサイクル構成はサポートされていません。

処理	導入

PutBucketNotificationConfiguration
 	
（以前はPUT Bucket notificationと呼ばれていました）
	リクエスト本文に含まれる通知設定XMLを使用して、バケットの通知を設定します。以下の実装の詳細についてご留意ください：
	
	• StorageGRID は、Amazon Simple Notification Service (Amazon SNS) トピック、Kafka トピック、または Webhook エンドポイントを宛先としてサポートします。Simple Queue Service (SQS) または AWS Lambda のエンドポイントはサポートされていません。
	• 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、Tenant Manager または Tenant Management API を使用して作成できます。
	
	通知設定を成功させるには、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、 `400 Bad Request`エラーがコード `InvalidArgument`とともに返されます。
	
	• 以下のイベントタイプについては、通知を設定できません。これらのイベントタイプはサポートされていません。
	◦ `s3:ReducedRedundancyLostObject`
	◦ `s3:ObjectRestore:Completed`
	• StorageGRID から送信されるイベント通知は標準の JSON 形式を使用しますが、一部のキーは含まれず、その他のキーには特定の値が使用されます。詳細は以下のリストを参照してください。
	◦ **eventSource**
	
	`sgws:s3`
	
	◦ **awsRegion**
	
	含まれていません
	
	◦ **x-amz-id-2**
	
	含まれていません
	
	◦ **arn**
	
	`urn:sgws:s3:::bucket_name`

PutBucketPolicy	バケットに適用されるポリシーを設定します。"バケットおよびグループのアクセスポリシーを使用する"を参照してください。

処理	導入
PutBucketReplication	リクエスト本文で提供されるレプリケーション設定XMLを使用して、バケットの"StorageGRID CloudMirror レプリケーション"を設定します。CloudMirrorレプリケーションについては、以下の実装の詳細に注意してください： • StorageGRID はレプリケーション設定の V1 のみをサポートしています。つまり、StorageGRID はルールに対して `Filter`要素の使用をサポートしておらず、オブジェクトバージョンの削除については V1 の規約に従います。詳細については、 "Amazon Simple Storage Service ユーザーガイド：レプリケーション設定"を参照してください。 • 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とバージョン管理されていないバケットの両方で設定できます。 • 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レプリケートできます。 • 宛先バケットは、Tenant Manager またはTenant Management 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。参照してください "CloudMirror レプリケーションの設定"。 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、リクエストは `400 Bad Request`として失敗します。エラーメッセージには次のように記載されています： `Unable to save the replication policy. The specified endpoint URN does not exist: URN.` • 設定 XML で `Role`を指定する必要はありません。この値は StorageGRID では使用されず、送信されても無視されます。 • 構成 XML からストレージクラスを省略すると、StorageGRID はデフォルトで `STANDARD`ストレージクラスを使用します。 • ソースバケットからオブジェクトを削除した場合、またはソースバケット自体を削除した場合、リージョン間レプリケーションの動作は次のとおりです： ◦ 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、通知もされません。 ◦ オブジェクトまたはバケットが複製された後に削除した場合、StorageGRID はリージョン間レプリケーションの V1 に対する標準の Amazon S3 削除動作に従います。

処理	導入
PutBucketTagging	`tagging`サブリソースを使用して、バケットのタグセットを追加または更新します。バケットタグを追加する際は、以下の制限事項にご注意ください。 • StorageGRID と Amazon S3 はどちらも、各バケットにつき最大 50 個のタグをサポートします。 • 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。タグキーは最大 128 文字の Unicode 文字で構成できます。 • タグの値は、最大256文字のUnicode文字を使用できます。 • キーと値は大文字と小文字を区別します。 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、値が割り当てられた `NTAP-SG-ILM-BUCKET-TAG`バケットタグが存在します。割り当てられた値を持つ `NTAP-SG-ILM-BUCKET-TAG`バケットタグが、すべての PutBucketTagging リクエストに含まれていることを確認してください。このタグを変更または削除しないでください。 注：この操作を行うと、バケットに既に存在するタグはすべて上書きされます。セットから既存のタグが省略された場合、それらのタグはバケットから削除されます。
PutBucketVersioning	`versioning`サブリソースを使用して、既存のバケットのバージョン管理状態を設定します。バージョン管理の状態は、以下のいずれかの値で設定できます。 • 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。 • 一時停止: 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 `null`が付与されます。
PutObjectLockConfiguration	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。 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変更されず、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。 詳細については、"S3 REST APIを使用してS3オブジェクトロックを設定する"を参照してください。

StorageGRIDがオブジェクトの**S3 REST API**操作を実装する方法

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。

以下の条件は、すべてのオブジェクト操作に適用されます。

- StorageGRID "一貫性値" は、以下の操作を除き、オブジェクトに対するすべての操作でサポートされています。
  - GetObjectAcl
  - `OPTIONS /`
  - PutObjectLegalHold
  - PutObjectRetention
  - SelectObjectContent
- 2つのクライアントが同じキーに書き込むなど、クライアントのリクエストが競合する場合は、「最新の要求が優先される」方式で解決されます。「最新の要求が優先される」評価のタイミングは、S3クライアントが操作を開始したときではなく、StorageGRID システムが特定のリクエストを完了したときに基づきます。
- 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、バケットの所有者が所有します。

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	DeleteObjectリクエストを処理する際、StorageGRIDはオブジェクトのすべてのコピーを、保存されているすべての場所から直ちに削除しようとします。成功した場合、StorageGRIDはクライアントに即座に応答を返します。すべてのコピーを30秒以内に削除できない場合（たとえば、場所が一時的に利用できない場合）、StorageGRIDは削除対象のコピーをキューに追加し、その後クライアントに成功を通知します。 制限事項 • 多要素認証（MFA）とレスポンスヘッダー `x-amz-mfa`はサポートされていません。 • `If-None`および `If-None-Match`ヘッダーは受け入れられますが、機能しません。 バージョン管理 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る必要があり、 `versionId`サブリソースを使用する必要があります。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。 `versionId`が削除マーカーに対応する場合、レスポンスヘッダー `x-amz-delete-marker`は `true`に設定されて返されます。 • バージョン管理が有効になっているバケットで `versionId`サブリソースを指定せずにオブジェクトを削除すると、削除マーカーが生成されます。削除マーカーの `versionId`は `x-amz-version-id`レスポンスヘッダーを使用して返され、 `x-amz-delete-marker`レスポンスヘッダーは `true`に設定されて返されます。 • バージョン管理が停止されているバケットで `versionId`サブリソースを指定せずにオブジェクトを削除すると、既に存在する「null」バージョンまたは「null」削除マーカーが完全に削除され、新しい「null」削除マーカーが生成されます。 `x-amz-delete-marker`レスポンスヘッダーは `true`に設定されて返されます。 注：場合によっては、オブジェクトに複数の削除マーカーが存在する可能性があります。 GOVERNANCE モードでオブジェクトのバージョンを削除する方法については、"S3 REST APIを使用してS3オブジェクトロックを設定する"を参照してください。
DeleteObjects （以前は「複数オブジェクトの削除」という名称でした）	多要素認証（MFA）とレスポンスヘッダー `x-amz-mfa`はサポートされていません。 同じリクエストメッセージ内で複数のオブジェクトを削除できます。 GOVERNANCE モードでオブジェクトのバージョンを削除する方法については、"S3 REST APIを使用してS3オブジェクトロックを設定する"を参照してください。

処理	導入
DeleteObjectTagging	`tagging`サブリソースを使用して、オブジェクトからすべてのタグを削除します。 バージョン管理 `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を削除します。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、`x-amz-delete-marker`レスポンスヘッダーは`true`に設定されます。
GetObject	"GetObject"
GetObjectAcl	アカウントに必要なアクセス認証情報が提供された場合、操作は肯定応答とID、DisplayName、およびオブジェクト所有者のPermissionを返し、所有者がオブジェクトへの完全な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	`tagging`サブリソースを使用して、オブジェクトのすべてのタグを返します。 バージョン管理 `versionId`クエリパラメータがリクエストで指定されていない場合、この操作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を返します。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、`x-amz-delete-marker`レスポンスヘッダーは`true`に設定されます。
HeadObject	"HeadObject"
RestoreObject	"RestoreObject"
PutObject	"PutObject"
CopyObject （以前は PUT Object - Copy という名称でした）	"CopyObject"

処理	導入
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"
PutObjectTagging	`tagging`サブリソースを使用して、既存のオブジェクトに一連のタグを追加します。 オブジェクトタグの制限 新規オブジェクトをアップロードする際にタグを追加することも、既存のオブジェクトにタグを追加することもできます。StorageGRID と Amazon S3 はどちらも、各オブジェクトにつき最大 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。タグキーは最大 128 Unicode 文字、タグ値は最大 256 Unicode 文字で構成できます。キーと値は大文字と小文字を区別します。 タグの更新と取り込み動作 PutObjectTagging を使用してオブジェクトのタグを更新する場合、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」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 が再評価された際に反映されます。 これは、ILMルールで取り込み動作に「Strict」オプションを使用している場合、必要なオブジェクトの配置ができない場合（たとえば、新たに必要となった場所が利用できない場合など）は、何もアクションが実行されないことを意味します。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、現在の配置を維持します。 競合の解決 2つのクライアントが同じキーに書き込むなど、クライアントのリクエストが競合する場合は、「最新の要求が優先される」方式で解決されます。「最新の要求が優先される」評価のタイミングは、S3クライアントが操作を開始したときではなく、StorageGRID システムが特定のリクエストを完了したときに基づきます。 バージョン管理 `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を追加します。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、 `x-amz-delete-marker`レスポンスヘッダーが `true`に設定されます。
SelectObjectContent	"SelectObjectContent"

StorageGRID は、"SelectObjectContent コマンド"に対して以下の Amazon S3 Select 句、データ型、および演算子をサポートしています。

記載されていない項目はサポート対象外です。

構文については、"SelectObjectContent"を参照してください。S3 Selectの詳細については、 "S3 Select のAWSドキュメント"を参照してください。

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。"S3 Selectを使用する際の考慮事項と要件" を参照してください。

条項

- SELECTリスト
- FROM句
- WHERE句
- LIMIT句

データ型

- ブール値
- 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
- NOT
- または

比較演算子

- <
- >
- ⇐
- >=
- =

- =
- <>
- !=
- 間
- で

パターンマッチング演算子

- LIKE
- _
- %

ユニタリ演算子

- IS NULL
- IS NOT NULL

数学演算子

- +
- -
- *
- /
- %

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

集計関数

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

条件関数

- ケース
- COALESCE
- 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

- DATE_ADD
- DATE_DIFF
- EXTRACT
- TO_STRING
- TO_TIMESTAMP
- UTCNOW

文字列関数

- CHAR_LENGTH、CHARACTER_LENGTH
- LOWER
- SUBSTRING
- TRIM
- アッパー

StorageGRIDがサーバー側の暗号化を使用する方法

サーバー側暗号化により、保存されている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際にデータを暗号化し、オブジェクトにアクセスする際にデータを復号化します。

サーバー側暗号化を使用する場合、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に S3 リクエストを発行すると、StorageGRID はオブジェクトを固有のキーで暗号化します。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 は保存されているキーを使用してオブジェクトを復号します。
- **SSE-C**（顧客提供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にS3リクエストを発行する場合、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際には、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した場合、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。

> ご提供いただいた暗号化キーは一切保存されません。暗号化キーを紛失した場合、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 が管理する一意の暗号化キーで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リクエストヘッダーは、以下のオブジェクト操作でサポートされています。

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

SSE-Cを使用する

管理する一意のキーで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てください。ヘッダー値は `AES256`である必要があります。
`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 ビットで、Base64 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エンコードされた128ビットである必要があります。

SSE-Cリクエストヘッダーは、以下のオブジェクト操作でサポートされています：

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

顧客提供キーを使用したサーバー側暗号化（**SSE-C**）の利用に関する考慮事項

SSE-Cを使用する前に、以下の点にご注意ください。

- https を使用する必要があります。

StorageGRID は、SSE-C を使用している場合、HTTP 経由で行われたすべてのリクエストを拒否します。セキュリティ上の理由から、誤って HTTP を使用して送信したキーはすべて漏洩している可能性があるとみなしてください。そのキーは破棄し、必要に応じてローテーションしてください。

- レスポンスに含まれる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。各オブジェクトに対して提供する暗号化キーを追跡する責任はお客様にあります。
- バケットでバージョン管理が有効になっている場合、各オブジェクトバージョンには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の追跡はお客様の責任となります。
- 暗号化キーをクライアント側で管理するため、鍵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。

ご提供いただいた暗号化キーは一切保存されません。暗号化キーを紛失した場合、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。取り込み処理は失敗します。

関連情報

"Amazon S3ユーザーガイド：カスタマー提供のキーを使用したサーバー側の暗号化（SSE-C）の使用"

S3 CopyObjectリクエストを使用して**StorageGRID**でオブジェクトのコピーを作成する

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 を実行することと同じです。

競合を解決する

2つのクライアントが同じキーに書き込むなど、クライアントのリクエストが競合する場合は、「最新の要求が優先される」方式で解決されます。「最新の要求が優先される」評価のタイミングは、S3クライアントが操作を開始したときではなく、StorageGRID システムが特定のリクエストを完了したときに基づきます。

オブジェクトサイズ

単一のPutObjectオペレーションの推奨最大サイズは 5 GiB（5,368,709,120 バイト）です。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"を使用してください。

単一の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と、「S3 PUT Object size too large」アラートがトリガーされます。StorageGRID 11.7または11.8を新規インストールした場合、この場合は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の今後のリリースでは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内の**UTF-8**文字

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

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、キー名または値の解釈された値に印刷不可能な文字が含まれている場合、 `x-amz-missing-meta` ヘッダーを返しません。

サポートされているリクエストヘッダー

以下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-`続いて、ユーザー定義のメタデータを含む名前と値のペア
- `x-amz-metadata-directive`：デフォルト値は `COPY`で、オブジェクトとその関連メタデータをコピーできます。

  オブジェクトをコピーする際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するかを `REPLACE`指定できます。

- `x-amz-storage-class`
- `x-amz-tagging-directive`：デフォルト値は `COPY`で、オブジェクトとすべてのタグをコピーできます。

  オブジェクトをコピーする際に既存のタグを上書きするか、タグを更新するかを `REPLACE`で指定できます。

- 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、オブジェクトのバージョンモードと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"を参照してください。

- 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`

    [サーバー側暗号化のためのリクエストヘッダー]を参照してください。

サポートされていないリクエストヘッダー

以下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`
- `If-Match`

  `If-Match header`は受け入れられるが、機能しない。

- `If-None-Match`

  `If-None-Match header`は受け入れられるが、機能しない。

- `x-amz-checksum-algorithm`

  オブジェクトをコピーする場合、ソースオブジェクトにチェックサムがあっても、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、オブジェクトリクエストで `x-amz-checksum-algorithm`を使用しようとするかどうかに関わらず適用されます。

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

ストレージクラスのオプション

`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link:../ilm/data-protection-options-for-ingest.html["取り込みオプション"]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。

- STANDARD

  （デフォルト）ILMルールで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、デュアルコミット取り込み操作を指定します。

- REDUCED_REDUNDANCY

  ILMルールで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。レガシー準拠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を満たすため、常に二重コミットによるデータ取り込みを実行します。

x-amz-copy-source を **CopyObject** で使用する

`x-amz-copy-source`ヘッダーで指定されたソースバケットおよびキーが、コピー先のバケットおよびキーと異なる場合、ソースオブジェクトデータのコピーがコピー先に書き込まれます。

送信元と宛先が一致し、`x-amz-metadata-directive`ヘッダーが`REPLACE`として指定されている場合、オブジェクトのメタデータはリクエストで指定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その場で変更したりすることはできません。`x-amz-server-side-encryption`ヘッダーまたは`x-amz-server-side-encryption-customer-algorithm`ヘッダーを指定した場合、StorageGRID はリクエストを拒否し、`XNotImplemented`を返します。
- 一致するILMルールで指定されている「取り込み動作」オプションは使用されません。アップデートによって発生す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が再評価された際に反映されます。

  これは、ILMルールで取り込み動作に「Strict」オプションを使用している場合、必要なオブジェクトの配置ができない場合（たとえば、新たに必要となった場所が利用できない場合など）は、何もアクションが実行されないことを意味します。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、現在の配置を維持します。

サーバー側暗号化のためのリクエストヘッダー

"サーバー側暗号化を使用する"を実行する場合、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キー（SSE-C）を使用して暗号化されている場合、オブジェクトを復号化してコピーできるように、CopyObjectリクエストに次の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`：`AES256`を指定します。
  - `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を作成した際に指定した MD5 ダイジェストを指定してください。
- 対象オブジェクト（コピー）を、ユーザーが提供および管理する固有のキーで暗号化する場合は、次の 3 つのヘッダーを含めてください。
  - `x-amz-server-side-encryption-customer-algorithm`：`AES256`を指定します。
  - `x-amz-server-side-encryption-customer-key`：対象オブジェクトに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。

> ご提供いただいた暗号化キーは一切保存されません。暗号化キーを紛失した場合、対応するオブジェクトも失われます。オブジェクトデータを保護するために顧客提供のキーを使用する前に、"サーバー側暗号化を使用する"に関する考慮事項を確認してください。

- ターゲットオブジェクト（コピー）を StorageGRID で管理される一意のキーで暗号化する場合（SSE）は、このヘッダーを CopyObject リクエストに含めてください：
  - `x-amz-server-side-encryption`

> `server-side-encryption`オブジェクトの値は更新できません。代わりに、`server-side-encryption`の新しい値を持つコピーを `x-amz-metadata-directive`：`REPLACE`を使用して作成してください。

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`ヘッダーを使用してオブジェクトの最新バージョンをコピーできます。オブジェクトの特定のバージョンをコピーするには、`versionId`サブリソースを使用してコピーするバージョンを明示的に指定する必要があります。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に返されます。対象バケットのバージョン管理が中断されている場合、`x-amz-version-id`は「null」値を返します。

StorageGRID のバケットから **GetObject** リクエストでオブジェクトを取得する

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

`partNumber`リクエストパラメータを使用して、マルチパートオブジェク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できます。 `x-amz-mp-parts-count`レスポンス要素は、オブジェクトがいくつのパートで構成されているかを示します。

`partNumber`は、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` に設定できますが、 `x-amz-mp-parts-count`レスポンス要素は、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の場合にのみ返されます。

ユーザーメタデータ内の**UTF-8**文字

StorageGRID は、ユーザー定義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、キー名または値に印刷不可能な文字が含まれている場合、 `x-amz-missing-meta` ヘッダーを返しません。

サポートされているリクエストヘッダー

以下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`

  `Range`ヘッダーは、`GetObject`の `x-amz-checksum-mode`ではサポートされていません。リクエストに `Range`を含め、 `x-amz-checksum-mode`が有効になっている場合、`StorageGRID` はレスポンスに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

以下のリクエストヘッダーはサポートされておらず、 `XNotImplemented`を返します：

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

バージョン管理

`versionId`サブリソースが指定されていない場合、この操作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を取得します。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、「見つかりません」ステータスが返され、 `x-amz-delete-marker`レスポンスヘッダーは `true`に設定されます。

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

オブジェクトが指定した固有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てください。

- `x-amz-server-side-encryption-customer-algorithm`： `AES256`を指定します。

- `x-amz-server-side-encryption-customer-key`：オブジェクトの暗号化キーを指定してください。
- `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。

> ご提供いただいた暗号化キーは一切保存されません。暗号化キーを紛失した場合、対応するオブジェクトも失われます。オブジェクトデータの保護に顧客提供のキーを使用する前に、"サーバー側暗号化を使用する"の考慮事項を確認してください。

クラウドストレージプール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"に保存されている場合、GetObjectリクエストの動作はオブジェクトの状態によって異なります。詳細については、"HeadObject"を参照してください。

> オブジェクトがクラウドストレージプールに保存され、かつそのオブジェクトのコピーがグリッド上にも1つ以上存在する場合、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、まず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	**GetObject** 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オブジェクト、またはイレイジャー コーディングを使用して保存され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。
クラウドストレージプール内のオブジェクトだが、まだ取得不可能な状態に移行していない	`200 OK` オブジェクトのコピーが取得されます。
オブジェクトが復元不可能な状態に遷移しました	`403 Forbidden, InvalidObjectState` "RestoreObject" リクエストを使用して、オブジェクトを取得可能な状態に復元します。
復元不可能な状態から復元処理中のオブジェクト	`403 Forbidden, InvalidObjectState` RestoreObject リクエストが完了するまで待ちます。
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オブジェクトまたはセグメントオブジェクト

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はオブジェクトの一部またはセグメントのサブセットをサンプリングすることにより、そのオブジェクトがクラウドストレージプールで利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、オブジェクトの一部がすでに取得不可能な状態に移行している場合や、オブジェクトの一部がまだ復元されていない場合に、誤って `200 OK`を返すことがあります。

このような場合：

- GetObjectリクエストは一部のデータを返す可能性がありますが、転送の途中で停止する場合があります。
- その後のGetObjectリクエストは `403 Forbidden`を返す可能性があります。

GetObjectおよびクロスグリッド複製

"グリッドフェデレーション"を使用していて、"グリッド間複製"が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リクエストを発行することで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が含まれ、以下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	• 完了：複製は成功しました。 • 保留中：オブジェクトはまだレプリケートされていません。 • **FAILURE**: レプリケーションが永続的な障害で失敗しました。ユーザーはエラーを解決する必要があります。
デスティネーション	レプリカ：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 `x-amz-replication-status`ヘッダーをサポートしていません。

StorageGRID の S3 HeadObject リクエストを使用してオブジェクトからメタデータを取得する

S3 HeadObject リクエストを使用して、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。オブジェクトがクラウドストレージ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移行状態を確認できます。

HeadObject とマルチパートオブジェクト

`partNumber`リクエストパラメータを使用して、マルチパートオブジェクトまたはセグメント化されたオブジェクトの特定部分のメタデータを取得できます。 `x-amz-mp-parts-count`レスポンス要素は、オブジェクトのパート数を示します。

`partNumber`は、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` に設定できますが、 `x-amz-mp-parts-count`レスポンス要素は、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の場合にのみ返されます。

ユーザーメタデータ内のUTF-8文字

StorageGRID は、ユーザー定義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、キー名または値に印刷不可能な文字が含まれている場合、 `x-amz-missing-meta` ヘッダーを返しません。

サポートされているリクエストヘッダー

以下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`

> `partNumber`パラメータと `Range`ヘッダーは、 `x-amz-checksum-mode`では `HeadObject` に対してサポートされていません。リクエストに `x-amz-checksum-mode`を有効にして含める場合、`StorageGRID` はレスポンスに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

以下のリクエストヘッダーはサポートされておらず、 `XNotImplemented`を返します：

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

バージョン管理

> `versionId`サブリソースが指定されていない場合、この操作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を取得します。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、「見つかりません」ステータスが返され、 `x-amz-delete-marker`レスポンスヘッダーは `true`に設定されます。

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

オブジェクトが指定した固有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てください。

- `x-amz-server-side-encryption-customer-algorithm`： `AES256`を指定します。
- `x-amz-server-side-encryption-customer-key`：オブジェクトの暗号化キーを指定してください。
- `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。

> ご提供いただいた暗号化キーは一切保存されません。暗号化キーを紛失した場合、対応するオブジェクトも失われます。オブジェクトデータの保護に顧客提供のキーを使用する前に、"サーバー側暗号化を使用する"の考慮事項を確認してください。

HeadObject クラウドストレージプールオブジェクトに対する応答

オブジェクトが"クラウド ストレージ プール"に格納されている場合、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、オブジェクトがクラウドストレージプールに移動される際、オプションで復元不可能な状態に移行される際、および復元される際のオブジェクトの状態に関する情報が含まれます。

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オブジェクト、またはイレイジャー コーディングを使用して保存されているオブジェクト	`200 OK`（特別なレスポンスヘッダーは返されません。）

クラウドストレージプール内のオブジェクトだが、まだ取得不可能な状態に移行していない	`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`の値は遠い未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、グリッド上には少なくとも1つのコピーが存在します。	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"`
	
	`expiry-date`の値は、遠い将来のある時点に設定されています。
	
	注: 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）、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ための"RestoreObject" リクエストを発行する必要があります。

オブジェクトの状態	**HeadObject** への応答

オブジェクトが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`

復元不可能な状態から復元処理中のオブジェクト	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="true"`

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`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2018 00:00:00 GMT"`
	
	`expiry-date`は、クラウドストレージプール内のオブジェクトがいつ復元不可能な状態に戻されるかを示します。

クラウドストレージプール内のマルチパートオブジェクトまたはセグメントオブジェクト

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はオブジェクトの一部またはセグメントのサブセットをサンプリングすることにより、そのオブジェクトがクラウドストレージプールで利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可能性があります `x-amz-restore: ongoing-request="false"`オブジェクトの一部がすでに取得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およびクロスグリッド複製

使用している場合は"グリッドフェデレーション"、"グリッド間複製"が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リクエストを発行することで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が含まれ、以下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	• 完了：複製は成功しました。 • 保留中：オブジェクトはまだレプリケートされていません。 • **FAILURE**: レプリケーションが永続的な障害で失敗しました。ユーザーはエラーを解決する必要があります。

Grid	レプリケーションステータス
デスティネーション	レプリカ：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 `x-amz-replication-status`ヘッダーをサポートしていません。

S3 PutObjectリクエストを使用して**StorageGRID**のバケットにオブジェクトを追加する

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。

競合を解決する

2つのクライアントが同じキーに書き込むなど、クライアントのリクエストが競合する場合は、「最新の要求が優先される」方式で解決されます。「最新の要求が優先される」評価のタイミングは、S3クライアントが操作を開始したときではなく、StorageGRID システムが特定のリクエストを完了したときに基づきます。

オブジェクトサイズ

単一のPutObjectオペレーションの推奨最大サイズは 5 GiB（5,368,709,120 バイト）です。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"を使用してください。

単一の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と、「S3 PUT Object size too large」アラートがトリガーされます。StorageGRID 11.7または11.8を新規インストールした場合、この場合は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の今後のリリースでは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サイズ

Amazon S3 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を 2 KB に制限します。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 エンコーディングにおける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内の**UTF-8**文字

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

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：

- PutObject、CopyObject、GetObject、およびHeadObjectリクエストは、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に成功します。
- StorageGRID は、キー名または値の解釈された値に印刷不可能な文字が含まれている場合、 `x-amz-missing-meta` ヘッダーを返しません。

オブジェクトタグの制限

新規オブジェクトをアップロードする際にタグを追加することも、既存のオブジェクトにタグを追加することもできます。StorageGRID と Amazon S3 はどちらも、各オブジェクトにつき最大 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。タグキーは最大 128 Unicode 文字、タグ値は最大 256 Unicode 文字で構成できます。キーと値は大文字と小文字を区別します。

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を含め、すべてのオブジェクトはバケット所有者アカウントが所有します。

サポートされているリクエストヘッダー

以下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`

  に対して `aws-chunked`を指定した場合`Content-Encoding`、StorageGRID は以下の項目を検証しません：

  - StorageGRID は、 `chunk-signature`をチャンクデータと照合して検証しません。
  - StorageGRID は、 `x-amz-decoded-content-length` に指定した値をオブジェクトに対して検証しません。
- `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、 `aws-chunked`ペイロード署名も使用されている場合にサポートされます。

- `x-amz-checksum-sha256`
- `x-amz-meta-`、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てください。

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

  ILM ルールの参照時間として ユーザー定義の作成時間 オプションを使用する場合は、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 `creation-time`を使用する必要があります。例：

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

`creation-time`の値は、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、参照時間に「ユーザー定義の作成時間」と「バランス型」または「厳密型」の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ルールを作成する際にエラーが返されます。

- `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、オブジェクトのバージョンモードと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"を参照してください。
- 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ためのリクエストヘッダー]を参照してください。

サポートされていないリクエストヘッダー

以下のリクエストヘッダーはサポートされていません：

- `If-Match`

  `If-Match header`は受け入れられるが、機能しない。
- `If-None-Match`

  `If-None-Match header`は受け入れられるが、機能しない。
- `x-amz-acl`

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

> `x-amz-website-redirect-location`ヘッダーは `XNotImplemented`を返します。

ストレージクラスのオプション

> `x-amz-storage-class`リクエストヘッダーはサポートされています。 `x-amz-storage-class`に送信された値は、取り込み中に `StorageGRID` がオブジェクトデータを保護する方法に影響しますが、`StorageGRID` システムにオブジェクトの永続的なコピーがいくつ保存されるか（`ILM` によって決定）には影響しません。

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する場合、 `x-amz-storage-class`ヘッダーは効果がありません。

以下の値を `x-amz-storage-class`に使用できます：

- `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、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を評価する際、StorageGRIDはこれらの最初の暫定コピーがルールに定められた配置指示を満たしているかどうかを判定します。満たしていない場合は、別の場所に新しいオブジェクトのコピーを作成する必要があり、最初の暫定コピーを削除する必要が生じる場合があります。
  - バランス：ILM ルールでバランスオプションが指定されており、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上に2つの暫定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座に作成できる場合（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ルールの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（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、一時的なコピーを 1 つ作成します。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。 `REDUCED_REDUNDANCY`オプションは、オブジェクトに一致する ILM ルールが単一の複製コピーを作成する場合に最適です。この場合、 `REDUCED_REDUNDANCY`を使用することで、取り込み操作ごとに不要なオブジェクトコピーの作成と削除をなくすことができます。

`REDUCED_REDUNDANCY`オプションの使用は、その他の状況では推奨されません。`REDUCED_REDUNDANCY`取り込み時のオブジェクトデータ損失のリスクを高めます。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。

一定期間にわたって複製コピーが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コピーが1つしか存在しない場合、ストレージノードが故障したり重大なエラーが発生したりすると、そのオブジェクトは失われます。アップグレードなどのメンテナンス手順中は、一時的にオブジェクトへのアクセスができなくなる場合もあります。

指定する `REDUCED_REDUNDANCY`は、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、アクティブな ILM ポリシーによってオブジェクトが評価されるときに作成されるオブジェクトのコピーの数には影響せず、StorageGRID システムでより低いレベルの冗長性でデータが保存されることもありません。

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。レガシー準拠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を満たすため、常に二重コミットによるデータ取り込みを実行します。

サーバー側暗号化のためのリクエストヘッダー

以下のリクエストヘッダーを使用すると、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 が管理する一意のキーで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`

`x-amz-server-side-encryption`ヘッダーが`PutObject`リクエストに含まれていない場合、グリッド全体の`link:../admin/changing-network-options-object-encryption.html["保存されたオブジェクトの暗号化設定"]`は`PutObject`レスポンスから省略されます。

- **SSE-C**：提供および管理する固有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 つのヘッダーすべてを使用してください。
  - `x-amz-server-side-encryption-customer-algorithm`： `AES256`を指定します。
  - `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てください。
  - `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。

ご提供いただいた暗号化キーは一切保存されません。暗号化キーを紛失した場合、対応するオブジェクトも失われます。オブジェクトデータを保護するために顧客提供のキーを使用する前に、"サーバー側暗号化を使用する"に関する考慮事項を確認してください。

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

バケットでバージョン管理が有効になっている場合、一意の `versionId`は、保存されるオブジェクトのバージョンに対して自動的に生成されます。この `versionId`は、 `x-amz-version-id`レスポンスヘッダーを使用してレスポンスにも返されます。

バージョン管理が中断された場合、オブジェクトのバージョンはnull `versionId`で保存され、nullバージョンが既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

`Authorization`ヘッダーを使用してリクエストを認証する場合、StorageGRID は次の点で AWS と異なります：

- StorageGRID は、 `host` ヘッダーを `CanonicalHeaders` 内に含める必要はありません。
- StorageGRID は、 `Content-Type` が `CanonicalHeaders` 内に含まれることを必要としません。
- StorageGRID は、 `x-amz-*` ヘッダーを `CanonicalHeaders` 内に含める必要はありません。

一般的なベストプラクティスとして、これらのヘッダーは常に `CanonicalHeaders`に含めて検証されるようにしてください。ただし、これらのヘッダーを除外しても、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：ペイロードを単一のチャンクで転送する（AWS Signature Version 4）"を参照してください。

関連情報

- "ILMを使用してオブジェクトを管理する"
- "Amazon Simple Storage Service API リファレンス：PutObject"

S3 RestoreObjectリクエストを使用して**StorageGRID**でオブジェクトを復元する

S3 RestoreObject リクエストを使用して、Cloud Storage Pool に保存されているオブジェクトをリストアできます。

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 リクエストのみをサポートしています。 `SELECT` タイプの復元はサポートしていません。Select リクエストは `XNotImplemented`を返します。

バージョン管理

オプションで、 `versionId`を指定して、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バージョンを復元します。 `versionId`を指定しない場合、オブジェクトの最新バージョンが復元されます

クラウドストレージプールオブジェクトに対する **RestoreObject** の動作

オブジェクトが"クラウド ストレージ プール"に保存されている場合、RestoreObjectリクエストは、オブジェクトの状態に基づいて以下の動作をします。詳細については、"HeadObject"を参照してください。

> オブジェクトがクラウドストレージプールに保存され、かつそのオブジェクトのコピーがグリッド上にも1つ以上存在する場合、RestoreObjectリクエストを発行してオブジェクトをリストアする必要はありません。代わりに、GetObjectリクエストを使用して、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
StorageGRID に取り込まれたオブジェクトで、ILM による評価がまだ行われていないか、クラウドストレージプールに存在しないオブジェクト	`403 Forbidden, InvalidObjectState`
クラウドストレージプール内のオブジェクトだが、まだ取得不可能な状態に移行していない	`200 OK`変更は行われません。 注: オブジェクトが取得不可能な状態に移行する前は、その `expiry-date`を変更することはできません。
オブジェクトが復元不可能な状態に遷移しました	`202 Accepted`リクエスト本文で指定された日数だけ、オブジェクトの復元可能なコピーをクラウドストレージプールに復元します。この期間が終了すると、オブジェクトは復元不可能な状態に戻ります。 オプションで、 `Tier`リクエスト要素を使用して、復元ジョブの完了にかかる時間を決定します（`Expedited`、 `Standard`、または `Bulk`）。 `Tier`を指定しない場合、 `Standard`ティアが使用されます。 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、 `Expedited`階層を使用してオブジェクトを復元することはできません。以下のエラーが返されます `403 Forbidden`、 `InvalidTier：Retrieval option is not supported by this storage class`。
復元不可能な状態から復元処理中のオブジェクト	`409 Conflict, RestoreAlreadyInProgress`

オブジェクトの状態	**RestoreObject** 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` 注： オブジェクトが取得可能な状態に復元されている場合、 `Days`に新しい値を指定してRestoreObjectリクエストを再発行することで、その `expiry-date`を変更できます。復元日は、リクエストの時間を基準に更新されます。

S3 SelectObjectContentリクエストを使用して**StorageGRID**でオブジェクトデータをフィルタリングする

S3 SelectObjectContent リクエストを使用して、単純なSQL文に基づいて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 Reference：SelectObjectContent"をご覧ください。

開始する前に

- テナントアカウントにはS3 Select権限が付与されています。
- クエリ対象のオブジェクトに対する `s3:GetObject`権限が必要です。
- クエリ対象のオブジェクトは、以下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アーカイブに圧縮して使用することもできます。
  - **Parquet**。Parquetオブジェクトに関する追加要件：
    - S3 Selectは、GZIPまたはSnappyを使用したカラム型圧縮のみをサポートしています。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形式の出力に対応していません。出力形式はCSVまたはJSONのいずれかを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ているデータ型を使用する必要があります。
    - INTERVAL、JSON、LIST、TIME、またはUUIDの論理型は使用できません。
- SQL式の最大長は256 KBです。
- 入力または結果に含まれるレコードの最大長は 1 MiB です。

CSVリクエスト構文例

```
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。ファイル内の州以外の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
```

クエリ対象ファイルの最初の数行、 `SUB-EST2020_ALL.csv`は次のようになっています：

```
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、 `changes.csv`は次のようになります：

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

AWS-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

StorageGRIDでサポートされているマルチパートアップロード操作

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。

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

- 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 を超えないようにしてください。そのバケットに対する ListMultipartUploads クエリが不完全な結果を返す可能性があるためです。
- StorageGRIDはAWSのマルチパートパーツのサイズ制限を適用します。S3クライアントは、以下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トは、5 MiB (5,242,880 バイト) から 5 GiB (5,368,709,120 バイト) の範囲内である必要があります。
  - 最後の部分は5 MiB（5,242,880バイト）未満でも構いません。
  - 一般的に、パートのサイズはできるだけ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 のパートサイズを使用します。各パートは固有のオブジェクトとみなされるため、大きなパート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、代わりにマルチパートではないアップロードの使用を検討してください。
- 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、またILMルールがBalancedまたはStrictを使用している場合、マルチパートアップロードが完了した際にオブジェクト全体として評価されます"取り込みオプション"。これがオブジェクトやパーツの配置にどのような影響を与えるかを認識しておく必要があります。

- S3マルチパートアップロードの進行中にILMが変更された場合、マルチパートアップロードが完了した時点で、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トはすべてILMによる再評価のためにキューに入れられ、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。これは、オブジェクト全体に対するILM要件を満たさない場所に、オブジェクトの一部が保存される可能性があることを意味します。例えば、ルールで10GB以上のオブジェクトはすべてDC1に保存され、それより小さいオブジェクトはすべてDC2に保存されるように指定されている場合、10個のパートからなるマルチパートアップロードの各1GBパートは、取り込み時にDC2に保存されます。ただし、ILMをオブジェクト全体として評価する場合、オブジェクトのすべてのパーツが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 "一貫性値"をサポートしています。
- 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、"オブジェクト分割しきい値（1 GiB）"は適用されません。
- 必要に応じて、"サーバー側暗号化"をマルチパートアップロードで使用できます。SSE（StorageGRID管理キーを使用したサーバー側暗号化）を使用するには、 `x-amz-server-side-encryption`リクエストヘッダーをCreateMultipartUploadリクエストにのみ含めます。SSE-C（顧客提供キーによるサーバー側暗号化）を使用するには、同じ3つの暗号化キーリクエストヘッダーをCreateMultipartUploadリクエストおよびその後の各UploadPartリクエストに指定します。
- マルチパートアップロードオブジェクトは、取り込みがベースバケットの Before タイムスタンプより前に開始された場合、アップロードが完了したタイミングに関係なく、"ブランチバケット"に含まれます。

処理	導入
AbortMultipartUpload	すべての Amazon S3 REST API の動作を実装済み。予告なく変更される場合があります。
CompleteMultipartUpload	"CompleteMultipartUpload"を参照してください。
CreateMultipartUpload （以前は「マルチパートアップロードの開始」という名称でした）	"CreateMultipartUpload"を参照してください。
ListMultipartUploads	"ListMultipartUploads"を参照してください。
ListParts	すべての Amazon S3 REST API の動作を実装済み。予告なく変更される場合があります。
UploadPart	"UploadPart"を参照してください。
UploadPartCopy	"UploadPartCopy"を参照してください。

StorageGRID S3 REST APIによるマルチパートアップロードの完了方法

CompleteMultipartUpload オペレーションは、以前にアップロードされた各パート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、CompleteMultipartUpload の `partNumber`リクエストパラメータにおいて、昇順での連続しない値をサポートします。パラメータは任意の値から開始できます。

競合を解決する

2つのクライアントが同じキーに書き込むなど、クライアントのリクエストが競合する場合は、「最新の要求が優先される」方式で解決されます。「最新の要求が優先される」評価のタイミングは、S3クライアントが操作を開始したときではなく、StorageGRID システムが特定のリクエストを完了したときに基づきます。

サポートされているリクエストヘッダー

以下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`

  `x-amz-storage-class`ヘッダーは、一致する ILM ルールがlink:../ilm/data-protection-options-for-ingest.html["デュアルコミットまたはバランス型取り込みオプション"]を指定している場合に StorageGRID が作成するオブジェクトコピーの数に影響します。

- `STANDARD`

  （デフォルト）ILMルールで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、デュアルコミット取り込み操作を指定します。

- `REDUCED_REDUNDANCY`

  ILMルールで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。レガシー準拠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を満たすため、常に二重コミットによるデータ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、その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。

返される `ETag`値はデータのMD5合計ではなく、マルチパートオブジェクトに対するAmazon S3 APIの `ETag`値の実装に従います。

サポートされていないリクエストヘッダー

以下のリクエストヘッダーはサポートされていません：

- `If-Match`

  `If-Match header`は受け入れられるが、機能しない。

- `If-None-Match`

  `If-None-Match header`は受け入れられるが、機能しない。

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

バージョン管理

この操作により、複数パートのアップロードが完了します。バケットで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。

バケットでバージョン管理が有効になっている場合、一意の `versionId`は、保存されるオブジェクトのバージョンに対して自動的に生成されます。この `versionId`は、 `x-amz-version-id`レスポンスヘッダーを使用してレスポンスにも返されます。

バージョン管理が中断された場合、オブジェクトのバージョンはnull `versionId`で保存され、nullバージョンが既に存在する場合は上書きされます。

> バケットで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に対して同時に複数のマルチパートアップロードが完了した場合でも、マルチパートアップロードが完了すると必ず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た後、同じオブジェクトキーに対して別のマルチパートアップロードが先に開始され、完了する可能性があります。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た場合でも、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、レプリケーションの失敗または通知をトリガーすることができます。テナントは、意図しない変更を避けるために、既存の値を再送信することができます。

"プラットフォームサービスのトラブルシューティング"を参照してください。

StorageGRID での S3 CreateMultipartUpload リクエストヘッダーとオプション

CreateMultipartUpload（以前は「マルチパートアップロードの開始」と呼ばれていました）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。

`x-amz-storage-class`リクエストヘッダーはサポートされています。 `x-amz-storage-class`に送信された値は、取り込み中に StorageGRID がオブジェクトデータを保護する方法に影響しますが、StorageGRID システムにオブジェクトの永続的なコピーがいくつ保存されるか（ILM によって決定）には影響しません。

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

以下の値を `x-amz-storage-class`に使用できます：

- 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、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を評価する際、StorageGRIDはこれらの最初の暫定コピーがルールに定められた配置指示を満たしているかどうかを判定します。満たしていない場合は、別の場所に新しいオブジェクトのコピーを作成する必要があり、最初の暫定コピーを削除する必要が生じる場合があります。
  - バランス：ILM ルールでバランスオプションが指定されており、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上に2つの暫定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座に作成できる場合（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際に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、一時的なコピーを 1 つ作成します。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。 `REDUCED_REDUNDANCY`オプションは、オブジェクトに一致する ILM ルールが単一の複製コピーを作成する場合に最適です。この場合、 `REDUCED_REDUNDANCY`を使用することで、取り込み操作ごとに不要なオブジェクトコピーの作成と削除をなくすことができます。

    `REDUCED_REDUNDANCY`オプションの使用は、その他の状況では推奨されません。 `REDUCED_REDUNDANCY`取り込み時のオブジェクトデータ損失のリスクを高めます。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。

    一定期間にわたって複製コピーが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コピーが1つしか存在しない場合、ストレージノードが故障したり重大なエラーが発生したりすると、そのオブジェクトは失われます。アップグレードなどのメンテナンス手順中は、一時的にオブジェクトへのアクセスができなくなる場合もあります。

指定する `REDUCED_REDUNDANCY`は、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、アクティブな ILM ポリシーによってオブジェクトが評価されるときに作成されるオブジェクトのコピーの数には影響せず、StorageGRID システムでより低いレベルの冗長性でデータが保存されることもありません。

> 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。レガシー準拠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を満たすため、常に二重コミットによるデータ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

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

- `x-amz-meta-`続いて、ユーザー定義のメタデータを含む名前と値のペア

  ユーザー定義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てください。

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

  ILM ルールの参照時間として ユーザー定義の作成時間 オプションを使用する場合は、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 `creation-time`を使用する必要があります。例：

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

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

  > `creation-time`をユーザー定義メタデータとして追加することは、レガシー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には許可されません。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ため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、"PutObject"を参照してください。

サーバー側暗号化のためのリクエストヘッダー

サーバー側暗号化を使用してマルチパートオブジェクトを暗号化するには、以下のリクエストヘッダーを使用できます。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 が管理する一意のキーで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てください。このヘッダーは、UploadPart リクエストには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固有のキーでオブジェクトを暗号化する場合は、CreateMultipartUploadリクエスト（およびその後の各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`：`AES256`を指定します。
  - `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てください。
  - `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。

ご提供いただいた暗号化キーは一切保存されません。暗号化キーを紛失した場合、対応するオブジェクトも失われます。オブジェクトデータを保護するために顧客提供のキーを使用する前に、"サーバー側暗号化を使用する"に関する考慮事項を確認してください。

サポートされていないリクエストヘッダー

以下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`

`x-amz-website-redirect-location`ヘッダーは `XNotImplemented`を返します。

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、パートのアップロード、アップロードされたパートの組み立て、およびアップロードの完了という、それぞれ独立した操作で構成されます。オブジェクトは、CompleteMultipartUpload操作が実行されると作成されます（該当する場合はバージョン管理されます）。

S3 ListMultipartUploads 操作と StorageGRID でサポートされているパラメータ

ListMultipartUploads オペレーションは、バケットに対して進行中のマルチパートアップロードの一覧を表示します。

以下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、パートのアップロード、アップロードされたパートの組み立て、およびアップロードの完了という、それぞれ独立した操作で構成されます。オブジェクトは、CompleteMultipartUpload操作が実行されると作成されます（該当する場合はバージョン管理されます）。

StorageGRID の S3 UploadPart 処理でサポートされているリクエストヘッダーとサポートされていないリクエストヘッダー

UploadPart オペレーションは、オブジェクトのマルチパートアップロードにおけるパート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

以下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ため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リクエストに以下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`： `AES256`を指定します。
- `x-amz-server-side-encryption-customer-key`：CreateMultipartUpload リクエストで指定したものと同じ暗号化キーを指定してください。
- `x-amz-server-side-encryption-customer-key-MD5`：CreateMultipartUpload リクエストで指定したものと同じ MD5 ダイジェストを指定します。

> ご提供いただいた暗号化キーは一切保存されません。暗号化キーを紛失した場合、対応するオブジェクトも失われます。オブジェクトデータの保護に顧客提供のキーを使用する前に、"サーバー側暗号化を使用する"の考慮事項を確認してください。

CreateMultipartUploadリクエスト時に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

以下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`

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、パートのアップロード、アップロードされたパートの組み立て、およびアップロードの完了という、それぞれ独立した操作で構成されます。オブジェクトは、CompleteMultipartUpload操作が実行されると作成されます（該当する場合はバージョン管理されます）。

S3 UploadPartCopy 操作を使用して StorageGRID にオブジェクトの一部をアップロードする

UploadPartCopy オペレーションは、既存のオブジェクトをデータソースとしてデータをコピーすることにより、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 の動作で実装されています。予告なく変更される場合があります。

このリクエストは、StorageGRIDシステム内の `x-amz-copy-source-range`で指定されたオブジェクトデータを読み書きします。

以下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ため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リクエストに以下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`：`AES256`を指定します。
- `x-amz-server-side-encryption-customer-key`：CreateMultipartUpload リクエストで指定したものと同じ暗号化キーを指定してください。
- `x-amz-server-side-encryption-customer-key-MD5`：CreateMultipartUpload リクエストで指定したものと同じ 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キー（SSE-C）を使用して暗号化されている場合、オブジェクトを復号化してコピーできるように、UploadPartCopy リクエストに次の 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`：`AES256`を指定します。
- `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を作成した際に指定した MD5 ダイジェストを指定してください。

> ご提供いただいた暗号化キーは一切保存されません。暗号化キーを紛失した場合、対応するオブジェクトも失われます。オブジェクトデータの保護に顧客提供のキーを使用する前に、"サーバー側暗号化を使用する"の考慮事項を確認してください。

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、パートのアップロード、アップロードされたパートの組み立て、およびアップロードの完了という、それぞれ独立した操作で構成されます。オブジェクトは、CompleteMultipartUpload操作が実行されると作成されます（該当する場合はバージョン管理されます）。

StorageGRID S3 REST API エラー応答

StorageGRID システムは、該当するすべての標準 S3 REST API エラーレスポンスをサポートしています。さらに、StorageGRID の実装ではいくつかのカスタムレスポンスが追加されています。

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
AccessDenied	403 Forbidden
BadDigest	400 Bad Request
BucketAlreadyExists	409 Conflict

Name	HTTPのステータス
BucketNotEmpty	409 Conflict
IncompleteBody	400 Bad Request
InternalError	500 内部サーバーエラー
InvalidAccessKeyId	403 Forbidden
InvalidArgument	400 Bad Request
InvalidBucketName	400 Bad Request
InvalidBucketState	409 Conflict
InvalidDigest	400 Bad Request
InvalidEncryptionAlgorithmError	400 Bad Request
InvalidPart	400 Bad Request
InvalidPartOrder	400 Bad Request
InvalidRange	416 Requested Range Not Satisfiable
InvalidRequest	400 Bad Request
InvalidStorageClass	400 Bad Request
InvalidTag	400 Bad Request
無効なURI	400 Bad Request
KeyTooLong	400 Bad Request
不正なXML	400 Bad Request
MetadataTooLarge	400 Bad Request
MethodNotAllowed	405 Method Not Allowed
MissingContentLength	411 Length Required
MissingRequestBodyError	400 Bad Request

Name	HTTPのステータス
MissingSecurityHeader	400 Bad Request
NoSuchBucket	404 Not Found
NoSuchKey	404 Not Found
NoSuchUpload	404 Not Found
NotImplemented	501 未実装
NoSuchBucketPolicy	404 Not Found
ObjectLockConfigurationNotFoundError	404 Not Found
PreconditionFailed	412 Precondition Failed
RequestTimeTooSkewed	403 Forbidden
ServiceUnavailable	503 Service Unavailable
SignatureDoesNotMatch	403 Forbidden
TooManyBuckets	400 Bad Request
UserKeyMustBeSpecified	400 Bad Request

StorageGRID カスタムエラーコード

Name	説明	HTTPのステータス
XBucketLifecycleNotAllowed	従来の準拠バケットでは、バケットのライフサイクル構成は許可されていません。	400 Bad Request
XBucketPolicyParseException	受信したバケットポリシーのJSONを解析できませんでした。	400 Bad Request
XComplianceConflict	旧式のコンプライアンス設定のため、操作が拒否されました。	403 Forbidden
XComplianceReducedRedundancyForbidden	レガシー準拠バケットでは冗長性の削減は許可されていません	400 Bad Request

Name	説明	HTTPのステータス
XMaxBucketPolicyLengthExceeded	お客様のポリシーは、許可されている最大バケットポリシーの長さを超えています。	400 Bad Request
XMissingInternalRequestHeader	内部リクエストのヘッダーが欠落しています。	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
XNotAcceptable	リクエストには、満たすことのできなかった1つ以上のAcceptヘッダーが含まれています。	406 Not Acceptable
XNotImplemented	ご提示いただいたご要望は、実装されていない機能に関するものです。	501 未実装

StorageGRID カスタムオペレーション

StorageGRID でサポートされているカスタムバケット操作

StorageGRIDシステムは、S3 REST APIに追加される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の一覧です。

処理	説明
"GET Bucket consistency"	特定のバケットに適用されている整合性を返します。
"PUT Bucketの一貫性"	特定のバケットに適用される整合性を設定します。
"GET Bucketの最終アクセス時間"	特定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返します。
"PUT Bucketの最終アクセス時刻"	特定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。
"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ているメタデータ通知設定XMLを削除します。
"バケットメタデータ通知設定を取得します"	特定のバケットに関連付けられたメタデータ通知設定XMLを返します。
"PUT Bucket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。

処理	説明
"ストレージ使用量を取得する"	アカウントで使用されているストレージの総量と、そのアカウントに関連付けられている各バケットごとのストレージ量を表示します。
"非推奨：CreateBucket（コンプライアンス設定付き）"	非推奨となり、サポート対象外です。コンプライアンスを有効にした状態で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 Bucket compliance"	非推奨ですがサポートされています：既存のレガシー準拠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 Bucket compliance"	非推奨ですがサポートされています：既存のレガシー準拠バケットのコンプライアンス設定を変更できます。

StorageGRIDで**GET Bucket**整合性リクエストを使用してバケットの整合性レベルを取得する

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確認できます。

デフォルトの整合性設定では、新しく作成されたオブジェクトに対して書き込み後の読み取りを保証するようになっています。

この操作を完了するには、s3:GetBucketConsistency 権限が必要です。またはアカウントのrootである必要があります。

リクエスト例

```
GET /bucket?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>`は次のいずれかの値を返します。

一貫性	説明
all	すべてのノードがデータを即座に受信しなければ、リクエストは失敗します。
strong-global	すべてのサイトにおけるすべてのクライアントリクエストに対して、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リクエストに対して、書き込み後の読み取りの一貫性を保証します。

一貫性	説明
新規書き込み後の読み取り	（デフォルト）新規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、オブジェクト更新に対しては結果整合性を提供します。高可用性とデータ保護を保証します。ほとんどの場合に推奨されます。
使用可能	新規オブジェクトとオブジェクト更新の両方に対して、最終的な整合性を提供します。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、または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。

回答例

```
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>
```

関連情報

"一貫性"

PUT Bucket整合性リクエストを使用して**StorageGRID**で整合性レベルを指定する

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。

デフォルトの整合性設定では、新しく作成されたオブジェクトに対して書き込み後の読み取りを保証するようになっています。

開始する前に

この操作を完了するには、s3:PutBucketConsistency 権限が必要です。またはアカウントのrootである必要があります。

要求

`x-ntap-sg-consistency`パラメータには、以下のいずれかの値を含める必要があります：

一貫性	説明
all	すべてのノードがデータを即座に受信しなければ、リクエストは失敗します。
strong-global	すべてのサイトにおけるすべてのクライアントリクエストに対して、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リクエストに対して、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）新規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、オブジェクト更新に対しては結果整合性を提供します。高可用性とデータ保護を保証します。ほとんどの場合に推奨されます。
使用可能	新規オブジェクトとオブジェクト更新の両方に対して、最終的な整合性を提供します。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、または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」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てください。または、クライアント側で各APIリクエストの一貫性を指定するように設定することもできます。最終手段としてのみ、バケットレベルでの一貫性設定を行ってください。

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"

GET Bucket last access timeリクエストを使用して**StorageGRID**でバケットの最終アクセス時間ステータスを取得する

GET Bucket last access time リクエストを使用すると、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判断できます。

この操作を完了するには、s3:GetBucketLastAccessTime 権限が必要です。またはアカウントのrootである必要があります。

リクエスト例

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

回答例

この例は、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>
```

StorageGRID で **PUT Bucket** 最終アクセス時刻リクエストを使用して最終アクセス時刻の更新を有効または無効にする

PUT Bucket last access time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、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となっています。

この操作を完了するには、バケットに対する s3:PutBucketLastAccessTime 権限、またはアカウントのルート権限が必要です。

> StorageGRID バージョン 10.3 以降、すべての新規バケットにおいて、最終アクセス時刻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合わせたい場合は、以前のバケットそれぞれについて、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のバケットの詳細ページから、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。参照してください "最終アクセス時刻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刻の更新が無効になっている場合、そのバケットに対する操作には以下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刻が更新されません。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。

- CopyObject および PutObjectTagging のメタデータのみを更新するリクエストは、最終アクセス時刻も更新します。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刻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刻は更新されません。コピーされたオブジェクトは、ソースバケットのILM評価キューには追加されません。ただし、コピー先については、CopyObjectリクエストは常に最終アクセス時刻を更新します。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 リクエストは最終アクセス時刻を更新します。完成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刻を有効にします。

```
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刻を無効にします。

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

StorageGRIDのDELETE Bucketメタデータ通知設定リクエストで検索統合サービスを無効にします

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

この操作を完了するには、バケットに対する s3:DeleteBucketMetadataNotification 権限、またはアカウントのルート権限が必要です。

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
```

GET Bucket メタデータ通知設定リクエストを使用すると、個々のバケットの検索統合を設定するために使用される設定 XML を取得できます。

この操作を完了するには、s3:GetBucketMetadataNotification 権限が必要です。またはアカウントのrootである必要があります。

リクエスト例

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

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

応答

レスポンスボディには、バケットのメタデータ通知設定が含まれます。メタデータ通知の設定により、検索統合のためにバケットをどのように構成するかを決定できます。つまり、どのオブジェクトがインデックス化されているか、およびそのオブジェクトの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確認できます。

```
<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>
```

各メタデータ通知設定には、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
MetadataNotificationConfiguration	メタデータ通知の対象となる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を格納するタグ。 1つ以上のルール要素を含みます。	はい
Rule	メタデータを特定のインデックスに追加すべきオブジェクトを識別するルールのコンテナタグ。 接頭辞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	ルールの一意の識別子。 ルール要素に含まれています。	いいえ
ステータス	ステータスは「有効」または「無効」のいずれかになります。無効化されたルールに対しては、何も措置は講じられません。 ルール要素に含まれています。	はい
Prefix	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。 すべてのオブジェクトに一致させるには、空のプレフィックスを指定します。 ルール要素に含まれています。	はい
デスティネーション	ルールの適用先を示すコンテナタグ。 ルール要素に含まれています。	はい

Name	説明	必須
Urn	オブジェクトのメタデータが送信される宛先のURN。以下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： • `es`3番目の要素でなければなりません。 • 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`domain-name/myindex/mytype`です。 エンドポイントは、Tenant ManagerまたはTenant Management APIを使用して設定します。形式は次のとおりです： • `arn:aws:es:_region:account-ID_:domain/mydomain/myindex/mytype` • `urn:mysite:es:::mydomain/myindex/mytype` エンドポイントは、設定XMLを送信する前に設定しておく必要があります。設定されていない場合、設定は404エラーで失敗します。 URN は Destination 要素に含まれています。	はい

回答例

`<MetadataNotificationConfiguration></MetadataNotificationConfiguration>`タグの間に含まれるXMLは、バケットに対して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ています。この例では、オブジェクトのメタデータが、 `records` という名前のAWSドメインでホストされている、 `current`という名前のElasticsearchインデックスと `2017`という名前のタイプに送信されています。

```
HTTP/1.1 200 OK
Date: Thu, 20 Jul 2017 18:24:05 GMT
Connection: KEEP-ALIVE
Server: StorageGRID/11.0.0
x-amz-request-id: 3832973499
Content-Length: 264
Content-Type: application/xml

<MetadataNotificationConfiguration>
    <Rule>
        <ID>Rule-1</ID>
        <Status>Enabled</Status>
        <Prefix>2017</Prefix>
        <Destination>
           <Urn>arn:aws:es:us-east-
1:3333333:domain/records/current/2017</Urn>
        </Destination>
    </Rule>
</MetadataNotificationConfiguration>
```

関連情報

"テナントアカウントを使用する"

PUT Bucketメタデータ通知設定リクエストを使用して**StorageGRID**で検索統合サービスを有効にします

PUTバケットメタデータ通知設定リクエストを使用すると、個々のバケットに対して検索統合サービスを有効にできます。リクエスト本文で指定するメタデータ通知設定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、またはアカウントのルート権限が必要です。

要求

リクエストの本文には、メタデータ通知の設定を含める必要があります。各メタデータ通知設定には、1つ以上のルールが含まれます。各ルールは、適用対象のオブジェクトと、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に基づいてフィルタリングできます。たとえば、プレフィックスが `/images`のオブジェクトのメタデータを1つの宛先に送信し、プレフィックスが `/videos`のオブジェクトを別の宛先に送信することができます。

プレフィックスが重複している設定は無効であり、送信時に拒否されます。例えば、プレフィックス `test`を持つオブジェクトに対する1つのルールと、プレフィックス `test2`を持つオブジェクトに対する2つ目のルールを含む設定は許可されません。

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

信される時点でエンドポイントが存在していない場合、リクエストは `400 Bad Request`として失敗します。エラーメッセージには次のように記載されています： `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>
```

この表は、メタデータ通知設定XMLの要素について説明しています。

Name	説明	必須
MetadataNotificationConfiguration	メタデータ通知の対象となる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を格納するタグ。 1つ以上のルール要素を含みます。	はい
Rule	メタデータを特定のインデックスに追加すべきオブジェクトを識別するルールのコンテナタグ。 接頭辞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	ルールの一意の識別子。 ルール要素に含まれています。	いいえ
ステータス	ステータスは「有効」または「無効」のいずれかになります。無効化されたルールに対しては、何も措置は講じられません。 ルール要素に含まれています。	はい

Name	説明	必須
Prefix	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。 すべてのオブジェクトに一致させるには、空のプレフィックスを指定します。 ルール要素に含まれています。	はい
デスティネーション	ルールの適用先を示すコンテナタグ。 ルール要素に含まれています。	はい
Urn	オブジェクトのメタデータが送信される宛先のURN。以下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： • `es`3番目の要素でなければなりません。 • 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`domain-name/myindex/mytype`です。 エンドポイントは、Tenant ManagerまたはTenant Management APIを使用して設定します。形式は次のとおりです： • `arn:aws:es:region:account-ID:domain/mydomain/myindex/mytype` • `urn:mysite:es:::mydomain/myindex/mytype` エンドポイントは、設定XMLを送信する前に設定しておく必要があります。設定されていない場合、設定は404エラーで失敗します。 URN は Destination 要素に含まれています。	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 `/images`に一致するオブジェクトのオブジェクトメタデータは1つの宛先に送信され、プレフィックス `/videos`に一致するオブジェクトのオブジェクトメタデータは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-1:3333333:domain/es-
domain/graphics/imagetype</Urn>
        </Destination>
    </Rule>
    <Rule>
        <ID>Videos-rule</ID>
        <Status>Enabled</Status>
        <Prefix>/videos</Prefix>
        <Destination>
           <Urn>arn:aws:es:us-west-1:22222222:domain/es-
domain/graphics/videotype</Urn>
        </Destination>
    </Rule>
</MetadataNotificationConfiguration>
```

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。

この例は、キー `SGWS/Tagging.txt`を持つオブジェクトが `test`という名前のバケットに作成された場合に生成される可能性のある JSON の例を示しています。 `test`バケットはバージョン管理されていないため、 `versionId`タグは空です。

```
{
  "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場合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一覧表示されています。

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

Type	商品名	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	versionID	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	リージョン	バケットリージョン（例： `us-east-1`）
システムメタデータ	サイズ	HTTPクライアントから見え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を使用したチャンク アップロード署名要求がサポートされるようになりました。	オブジェクトハッシュ

Type	商品名	説明
ユーザーメタデータ	メタデータ *key:value*	オブジェクトに関するすべてのユーザーメタデータ（キーと値のペアとして）
Tags	タグ *key:value*	オブジェクトに対して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については、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。Elasticsearchが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設定するには、Elasticsearchの動的フィールドマッピングおよび日付形式のマッピングに関する手順に従ってください。検索統合サービスを設定する前に、インデックス上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。ドキュメントがインデックス化された後は、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"

StorageGRIDで**GET Storage Usage**リクエストを使用してアカウントとバケットのストレージ使用量を取得

GET Storage Usageリクエストは、アカウントが使用しているストレージの総量と、そのアカウントに関連付けられている各バケットごとのストレージ使用量を示します。

アカウントとそのバケットが使用するストレージ容量は、 `x-ntap-sg-usage`クエリパラメータを使用した修正済みのListBucketsリクエストで取得できます。バケットストレージの使用状況は、システムによって処理される PUT および DELETE リクエストとは別に追跡されます。リクエストの処理状況によっては、特にシステム負荷が高い場合、使用状況の値が期待値と一致するまでに多少の遅延が生じ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整合性を使用して使用状況情報を取得しようとします。強力なグローバル整合性が達成できない場合、StorageGRID はサイトの強力な整合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 権限が必要です。またはアカウントのroot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
```

回答例

この例は、2つのバケットに4つのオブジェクトと12バイトのデータを持つ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ています。

```
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

保存されるすべてのオブジェクトバージョンは、応答内の `ObjectCount`および `DataBytes`の値に反映されます。削除マーカーは `ObjectCount`の合計には追加されません。

関連情報

"一貫性"

レガシーコンプライアンス向けのバケットリクエストは非推奨です

レガシー **StorageGRID** コンプライアンス向けの非推奨のバケットリクエスト

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非推奨になりました

以前のバージョンの StorageGRID で利用可能だった StorageGRID コンプライアンス機能は非推奨となり、S3 Object Lock 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た場合、グローバル S3 オブジェクトロック設定は StorageGRID 11.6 で有効になります。コンプライアンスが有効になっている新しいバケットを作成することはできなくなりましたが、必要に応じて、StorageGRID S3 REST API を使用して既存のレガシー準拠バケットを管理できます。

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

廃止されたコンプライアンス要求：

- "非推奨 - コンプライアンスのための PUT バケットリクエストの変更"

  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することができました。

- "非推奨 - GET Bucket コンプライアンス"

  GET Bucket コンプライアンス要求は非推奨になりました。ただし、このリクエストを使用して、既存のレガシー準拠バケットに対して現在適用されているコンプライアンス設定を確認することは引き続き可能です。

- "非推奨 - PUT Bucket 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、このリクエストを使用して、既存のレガシー準拠バケットのコンプライアンス設定を変更することは引き続き可能です。例えば、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することができます。

StorageGRIDのSGCompliance要素は廃止されました

SGCompliance XML要素は非推奨です。以前は、準拠バケットを作成するための CreateBucket 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ができました。

以前のバージョンの StorageGRID で利用可能だった StorageGRID コンプライアンス機能は非推奨となり、S3 Object Lock に置き換えられました。詳細については、以下をご覧ください。

- "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"

コンプライアンスが有効になっている状態では、新しいバケットを作成できなくなりました。新しい準拠バケットを作成するためにコンプライアンス向けの CreateBucket リクエストの変更を使用しようとすると、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.
```

GET Bucket コンプライアンス要求は非推奨になりました。ただし、このリクエストを使用して、既存のレガシー準拠バケットに対して現在適用されているコンプライアンス設定を確認することは引き続き可能です。

以前のバージョンの StorageGRID で利用可能だった StorageGRID コンプライアンス機能は非推奨となり、S3 Object Lock に置き換えられました。詳細については、以下をご覧ください。

- "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 権限が必要です。またはアカウントのrootである必要があります。

リクエスト例

このリクエスト例では、 `mybucket`という名前のバケットのコンプライアンス設定を確認できます。

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

回答例

レスポンスXMLでは、 `<SGCompliance>`バケットに適用されているコンプライアンス設定を一覧表示します。この応答例は、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）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ホールドはありません。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
RetentionPeriodMinutes	このバケットに追加されたオブジェクトの保持期間（分単位）。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から開始されます。
LegalHold	• True：このバケットは現在、法的保留状態にあります。このバケット内のオブジェクトは、保持期間が過ぎていても、法的保留が解除されるまで削除できません。 • False：このバケットは現在、法的保留下にはありません。このバケット内のオブジェクトは、保持期間が経過すると削除できます。
AutoDelete	• 正しい：このバケット内のオブジェクトは、バケットが法的保留下にない限り、保持期間が満了すると自動的に削除されます。 • False: このバケット内のオブジェクトは、保持期間が満了しても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、手動で削除してください。

エラー応答

バケットが準拠するように作成されていない場合、応答の HTTP ステータスコードは `404 Not Found`となり、S3 エラーコードは `XNoSuchBucketCompliance`です。

StorageGRID で廃止された PUT Bucket compliance 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、このリクエストを使用して、既存のレガシー準拠バケットのコンプライアンス設定を変更することは引き続き可能です。例えば、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することができます。

以前のバージョンの StorageGRID で利用可能だった StorageGRID コンプライアンス機能は非推奨となり、S3 Object Lock に置き換えられました。詳細については、以下をご覧ください。

- "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 権限が必要です。またはアカウントのroot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際には、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、 `mybucket`という名前のバケットのコンプライアンス設定を変更します。この例では、 `mybucket`内のオブジェクトがグリッドに取り込まれた時点から、保持期間は1年間ではなく2年間（1,051,200分）になります。このバケットにはリーガルホールドはありません。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
RetentionPeriodMinutes	このバケットに追加されたオブジェクトの保持期間（分単位）。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から開始されます。 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、その値を小さくすることはできません。大きくすることしかできません。
LegalHold	• True：このバケットは現在、法的保留状態にあります。このバケット内のオブジェクトは、保持期間が過ぎていても、法的保留が解除されるまで削除できません。 • False：このバケットは現在、法的保留下にはありません。このバケット内のオブジェクトは、保持期間が経過すると削除できます。
AutoDelete	• 正しい：このバケット内のオブジェクトは、バケットが法的保留下にない限り、保持期間が満了すると自動的に削除されます。 • False: このバケット内のオブジェクトは、保持期間が満了しても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、手動で削除してください。

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 **Strong-global** 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が、変更されたコンプライアンス設定に対してライトアフターリード整合性を持つことを保証します。

StorageGRID が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 **Strong-global** 整合性を達成できない場合、レスポンスの HTTP ステータスコードは `503 Service Unavailable.`

この応答を受け取った場合は、必要なストレージサービスが可能な限り速やかに利用可能になるよう、グリッ

ド管理者に連絡してください。グリッド管理者が各サイトのストレージノードを十分に利用可能にできない場合、テクニカルサポートから、**Strong-site** の整合性を強制することで失敗したリクエストを再試行するよう指示される場合があります。

> テクニカルサポートから指示された場合、およびこのレベルを使用することによる潜在的な影響を理解している場合を除き、PUT バケットコンプライアンスに対して **Strong-site** の整合性を強制しないでください。

一貫性が*ストロングサイト*に低下すると、StorageGRID は、更新されたコンプライアンス設定について、サイト内のクライアント要求に対してのみ書き込み後読み取りの一貫性を保証します。これはつまり、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 システムはこのバケットに対して一時的に複数の矛盾した設定を持つ可能性があるということです。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。例えば、バケットを法的保留下に置く際に、より低い整合性を強制的に適用した場合、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、法的保留オフ）が引き続き有効になる可能性があります。その結果、法的保留中と思われるオブジェクトは、保持期間が満了すると、ユーザーまたは AutoDelete（有効な場合）によって削除される可能性があります。

Strong-site の一貫性の使用を強制するには、PUT Bucket コンプライアンス要求を再発行し、以下のように `Consistency-Control` HTTP リクエストヘッダーを含めます：

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

エラー応答

- バケットが準拠するように作成されていない場合、応答の HTTP ステータスコードは `404 Not Found`です。
- リクエストの `RetentionPeriodMinutes`がバケットの現在の保持期間より短い場合、HTTP ステータスコードは `400 Bad Request`です。

関連情報

"非推奨: PUT Bucket リクエストのコンプライアンス変更"

アクセスポリシーの管理

StorageGRID が **AWS** ポリシーを使用して **S3** バケットとオブジェクトへのアクセスを制御する方法

StorageGRID は Amazon Web Services (AWS) の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ています。S3 API のアクセスポリシーは JSON 形式で記述されます。

アクセスポリシーの概要

StorageGRID は、3種類のアクセスポリシーをサポートしています。

- *バケットポリシー*は、GetBucketPolicy、PutBucketPolicy、およびDeleteBucketPolicy S3 API操作、またはテナントマネージャーもしくはテナント管理APIを使用して管理されます。バケットポリシーはバケッ

トに紐付けられており、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つのバケットと、場合によっては複数のグループにのみ適用されます。

- *グループポリシー*は、Tenant ManagerまたはTenant Management APIを使用して設定します。グループポリシーはアカウント内のグループに関連付けられており、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設定されます。グループポリシーは、1つのグループと、場合によっては複数のバケットにのみ適用されます。
- セッションポリシー は、AssumeRole リクエストの一部として含まれます。セッションポリシーは特定のセッションにのみ適用され、グループポリシーおよびバケットポリシーによって付与される権限に加えて、ユーザーが持つ権限をさらに詳細に定義します。

> グループポリシー、バケットポリシー、セッションポリシーの間には、優先順位の違いはありません。

StorageGRID バケットポリシーとグループポリシーは、Amazon が定義した特定の文法に従います。各ポリシーの中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以下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）
- 影響
- Principal/NotPrincipal
- リソース/NotResource
- アクション/NotAction
- 条件（任意）

ポリシーステートメントは、権限を指定するために次の構造を使用して作成されます：<Effect> を付与して、<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
Sid	Sid要素は省略可能です。Sidは、ユーザー向けの説明としてのみ使用されるものです。StorageGRIDシステムによって保存されますが、解釈はされません。
影響	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設定します。サポートされているAction要素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）する操作を指定する必要があります。

要素	説明
Principal/NotPrincipal	ユーザー、グループ、アカウントに対して、特定のリソースへのアクセス権限や特定のアクションの実行権限を付与できます。リクエストにS3署名が含まれていない場合、ワイルドカード文字（*）をプリンシパルとして指定することで、匿名アクセスが許可されます。デフォルトでは、アカウントのルートユーザーのみが、そのアカウントが所有するリソースにアクセスできます。 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が暗黙のPrincipal要素となります。
リソース/NotResource	リソース要素は、バケットとオブジェクトを識別します。Amazon Resource Name （ARN）を使用してリソースを識別することで、バケットやオブジェクトに対するアクセス許可を許可または拒否できます。
アクション/NotAction	権限を構成する2つの要素は、アクション要素とエフェクト要素です。グループがリソースを要求すると、そのリソースへのアクセスが許可されるか、拒否されるかのいずれかになります。明示的に権限を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することで、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要素は省略可能です。条件を使用すると、ポリシーを適用するタイミング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。例えば、このActionはs3:GetObject、s3:PutObject、s3:DeleteObjectなどの権限に一致します。

```
s3:*Object
```

Resource 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文字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の設定を除き、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 Principal の値として設定します。

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

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

次の例では、ステートメントでEffect、Principal、Action、およびResource要素を使用しています。この例は、Effectの「Allow」を使用して、Principalである管理グループ `federated-group/admin`と財務グループ `federated-group/finance`に対して、指定した名前のバケットに対するAction `s3:ListBucket`の実行権限 `mybucket`、およびそのバケット内のすべてのオブジェクトに対するAction `s3:GetObject`の実行権限を付与する、完全なバケットポリシー記述を示しています。

```
{
  "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。

ポリシーの整合性

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整合性が保たれます。グループポリシーが整合性を保つようになると、ポリシーのキャッシュ処理のため、変更が反映されるまでにさらに15分ほどかかる場合があります。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整合性が保たれます。

必要に応じて、バケットポリシーの更新における整合性保証を変更できます。たとえば、サイト停止中にバケットポリシーの変更を有効にする場合などです。

この場合、PutBucketPolicyリクエストに `Consistency-Control`ヘッダーを設定するか、PUT Bucket 整合性リクエストを使用することができます。バケットポリシーが整合性を保つようになると、ポリシーのキャッシュのため、変更が反映されるまでにさらに8秒かかる場合があります。

> 一時的な状況に対処するために整合性を別の値に設定した場合は、作業が完了したら必ずバケットレベルの設定を元の値に戻してください。そうでない場合、今後のすべてのバケット要求は変更された設定を使用します。

セッションポリシーとは何ですか？

セッションポリシーとは、ユーザーがグループを代理で利用する場合など、特定のセッション中に利用可能な権限を一時的に制限するアクセスポリシーのことです。セッションポリシーでは、許可できる権限は一部のみであり、追加の権限を付与することはできません。グループ自体は、より広範な権限を持っている可能性があります。

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、以下の構文を使用します。

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

- IDリソースARN（ユーザーおよびグループ）を指定するには、以下の構文を使用します：

```
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：

- オブジェクトキー内の0個以上の文字に一致させるには、アスタリスク（*）をワイルドカードとして使用できます。
- 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。パーセント表記はサポートされていません。

  "RFC 2141 URN構文"

  PutBucketPolicy 操作の HTTP リクエストボディは、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、アクセス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、Resource要素が必要です。ポリシーでは、リソースは要素 `Resource` で表されます。または、除外の場合は `NotResource`で表されます。
- S3リソースARNを使用してリソースを指定します。例：

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

- オブジェクトキーの中に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、グループポリシー作成時にまだ存在しないバケットを指定できます。

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または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。グループポリシーのポリシーステートメントには、グループ自体がプリンシパルであると解釈されるため、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」、または除外の場合は「NotPrincipal」で示されます。
- 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、テナントアカウントID 27233906934684427525を使用します。このアカウントには、アカウントのルートとアカウント内のすべてのユーザーが含まれます。

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

- アカウントのルートのみを指定できます。

```
"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（「Managers」）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、ユーザー名の代わりにユーザーUUIDを使用できます。

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

例えば、Alex が組織を離れ、ユーザー名 `Alex`が削除されたとします。新しい Alex が組織に加わり、同じ `Alex`ユーザー名が割り当てられた場合、新しいユーザーが意図せず元のユーザーに付与された権限を継承してしまう可能性があります。

- プリンシパル値には、バケットポリシー作成時にまだ存在しないグループ名またはユーザー名を指定できます。

ポリシーで権限を指定する

ポリシーにおいて、Action要素はリソースに対するアクセス許可を許可または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、それは要素「Action」、または除外の場合は「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。

この表には、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一覧表示されています。

Amazon S3は現在、PutBucketReplicationアクションとDeleteBucketReplicationアクションの両方にs3:PutReplicationConfiguration権限を使用します。StorageGRIDは各アクションに個別の権限を使用します。これは、元のAmazon S3の仕様と一致しています。

削除は、put操作を使用して既存の値を上書きする際に実行されます。

バケットに適用される権限

権限	**S3 REST API** オペレーション	**StorageGRID** のカスタム
s3：CreateBucket	CreateBucket	○ 注：グループポリシーでのみ使用してください。
s3:DeleteBucket	DeleteBucket	
s3:DeleteBucketMetadataNotification	バケットメタデータ通知設定の削除	はい
s3:DeleteBucketPolicy	DeleteBucketPolicy	
s3:DeleteReplicationConfiguration	DeleteBucketReplication	はい、PUTとDELETEには別々の権限があります
s3:GetBucketAcl	GetBucketAcl	

権限	**S3 REST API** オペレーション	**StorageGRID** のカスタム
s3:GetBucketCompliance	GET Bucket compliance（非推奨）	はい
s3:GetBucketConsistency	GET Bucket consistency	はい
s3:GetBucketCORS	GetBucketCors	
s3:GetEncryptionConfiguration	GetBucketEncryption	
s3:GetBucketLastAccessTime	GET Bucketの最終アクセス時間	はい
s3:GetBucketLocation	GetBucketLocation	
s3:GetBucketMetadataNotification	バケットメタデータ通知設定を取得します	はい
s3:GetBucketNotification	GetBucketNotificationConfiguration	
s3:GetBucketObjectLockConfiguration	GetObjectLockConfiguration	
s3:GetBucketPolicy	GetBucketPolicy	
s3:GetBucketTagging	GetBucketTagging	
s3:GetBucketVersioning	GetBucketVersioning	
s3:GetLifecycleConfiguration	GetBucketLifecycleConfiguration	
s3:GetReplicationConfiguration	GetBucketReplication	
s3:ListAllMyBuckets	• ListBuckets • ストレージ使用量を取得する	はい、GET Storage Usageの場合です。 注：グループポリシーでのみ使用してください。
s3：ListBucket	• ListObjects • HeadBucket • RestoreObject	
s3:ListBucketMultipartUploads	• ListMultipartUploads • RestoreObject	

権限	**S3 REST API** オペレーション	**StorageGRID** のカスタム
s3:ListBucketVersions	GET Bucket versions	
s3:PutBucketCompliance	PUT Bucket compliance（非推奨）	はい
s3:PutBucketConsistency	PUT Bucketの一貫性	はい
s3:PutBucketCORS	• DeleteBucketCors† • PutBucketCors	
s3:PutEncryptionConfiguration	• DeleteBucketEncryption • PutBucketEncryption	
s3:PutBucketLastAccessTime	PUT Bucketの最終アクセス時刻	はい
s3:PutBucketMetadataNotification	PUT Bucketメタデータ通知設定	はい
s3:PutBucketNotification	PutBucketNotificationConfiguration	
s3:PutBucketObjectLockConfiguration	• CreateBucket と `x-amz-bucket-object-lock-enabled: true`リクエストヘッダー（s3:CreateBucket 権限も必要） • PutObjectLockConfiguration	
s3：PutBucketPolicy	PutBucketPolicy	
s3：PutBucketTagging	• DeleteBucketTagging† • PutBucketTagging	
s3:PutBucketVersioning	PutBucketVersioning	
s3:PutLifecycleConfiguration	• DeleteBucketLifecycle† • PutBucketLifecycleConfiguration	
s3:PutReplicationConfiguration	PutBucketReplication	はい、PUTとDELETEには別々の権限があります

オブジェクトに適用される権限

権限	**S3 REST API** オペレーション	**StorageGRID** のカスタム
s3:AbortMultipartUpload	• AbortMultipartUpload • RestoreObject	
s3:BypassGovernanceRetention	• DeleteObject • DeleteObjects • PutObjectRetention	
s3:DeleteObject	• DeleteObject • DeleteObjects • RestoreObject	
s3:DeleteObjectTagging	DeleteObjectTagging	
s3:DeleteObjectVersionTagging	DeleteObjectTagging（オブジェクトの特定バージョン）	
s3:DeleteObjectVersion	DeleteObject（オブジェクトの特定バージョン）	
s3:GetObject	• GetObject • HeadObject • RestoreObject • SelectObjectContent	
s3:GetObjectAcl	GetObjectAcl	
s3:GetObjectLegalHold	GetObjectLegalHold	
s3:GetObjectRetention	GetObjectRetention	
s3：GetObjectTagging	GetObjectTagging	
s3:GetObjectVersionTagging	GetObjectTagging（オブジェクトの特定バージョン）	
s3:GetObjectVersion	GetObject（オブジェクトの特定バージョン）	
s3:ListMultipartUploadParts	ListParts、RestoreObject	

権限	**S3 REST API** オペレーション	**StorageGRID** のカスタム
s3:PutObject	• PutObject • CopyObject • RestoreObject • CreateMultipartUpload • CompleteMultipartUpload • UploadPart • UploadPartCopy	
s3:PutObjectLegalHold	PutObjectLegalHold	
s3:PutObjectRetention	PutObjectRetention	
s3:PutObjectTagging	PutObjectTagging	
s3:PutObjectVersionTagging	PutObjectTagging（オブジェクトの特定バージョン）	
s3:PutOverwriteObject	• PutObject • CopyObject • PutObjectTagging • DeleteObjectTagging • CompleteMultipartUpload	はい
s3:RestoreObject	RestoreObject	

PutOverwriteObject 権限を使用する

s3:PutOverwriteObject 権限は、オブジェクトの作成または更新を行う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どうかが決まります。

この権限の設定可能な項目は以下のとおりです：

- 許可：クライアントはオブジェクトを上書きできます。これは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。「拒否」に設定すると、PutOverwriteObject 権限の仕組みは以下のとおりです。
  - 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 バージョン管理が有効になっている場合、拒否設定により、PutObjectTagging または DeleteObjectTagging 操作がオブジェクトとその非現行バージョンの TagSet を変更できなくなります。

  - 既存のオブジェクトが見つからない場合、この権限は無効です。

- この権限が存在しない場合、効果は「許可」が設定されている場合と同じです。

> ⓘ 現在のS3ポリシーで上書きが許可されており、PutOverwriteObject権限が「拒否」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することができません。さらに、クライアントの変更を防止する* 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権限の設定より優先されます。

ポリシーで条件を指定する

条件とは、ポリシーがいつ有効になるかを定めるものです。条件は、演算子とキーと値のペアで構成されます。

条件式は、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 条件は SourceIp 条件キーを使用しています。

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

サポートされている条件演算子

条件演算子は以下のように分類されます：

- 文字列
- 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
StringEquals	キーと文字列値を完全一致（大文字小文字を区別）に基づいて比較します。
StringNotEquals	否定一致（大文字小文字を区別）に基づいて、キーと文字列値を比較します。
StringEqualsIgnoreCase	キーと文字列値を完全一致に基づいて比較します（大文字小文字は区別しません）。
StringNotEqualsIgnoreCase	否定一致に基づいてキーと文字列値を比較します（大文字と小文字は区別しません）。
StringLike	キーと文字列値を完全一致（大文字小文字を区別）に基づいて比較します。ワイルドカード文字として「*」と「?」を含めることができます。
StringNotLike	否定一致（大文字小文字を区別）に基づいて、キーと文字列値を比較します。ワイルドカード文字として * と？を含めることができます。
NumericEquals	キーと数値を完全一致に基づいて比較します。
NumericNotEquals	否定一致に基づいて、キーと数値を比較します。
NumericGreaterThan	キーと数値を「より大きい」という条件に基づいて比較します。
NumericGreaterThanEquals	キーと数値を「以上」の一致に基づいて比較します。
NumericLessThan	キーと数値を「より小さい」条件に基づいて比較します。
NumericLessThanEquals	キーと数値を「以下」のマッチングに基づいて比較します。
ブール値	キーをブール値と「true or false」の一致に基づいて比較します。
IpAddress	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。
NotIpAddress	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。
IfExists	Null条件を除く、任意の条件演算子に追加して、その条件キーが存在しないかどうかを確認します。条件キーが存在しない場合は TRUE を返します。

サポートされている条件キー

条件キー	操作	説明
aws:SourceIp	IPオペレーター	リクエストが送信されたIPアドレスと比較します。バケット操作またはオブジェクト操作に使用できます。 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ーサービス経由で送信された場合、これはロードバランサーサービスの上流の IP アドレスと比較されます。 注: サードパーティ製の非透過型ロードバランサーを使用する場合、これはそのロードバランサーのIPアドレスと比較されます。 `X-Forwarded-For`ヘッダーは有効性を確認できないため、無視されます。
aws:username	リソース/アイデンティティ	リクエストが送信された送信者のユーザー名と比較します。バケット操作またはオブジェクト操作に使用できます。
s3:区切り文字	s3:ListBucket および 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されます。
s3:ExistingObjectTag/<tag-key>	s3:DeleteObjectTagging s3:DeleteObjectVersionTagging s3:GetObject s3:GetObjectAcl 3：GetObjectTagging s3:GetObjectVersion s3:GetObjectVersionAcl s3:GetObjectVersionTagging s3:PutObjectAcl s3:PutObjectTagging s3:PutObjectVersionAcl s3:PutObjectVersionTagging	既存のオブジェクトに特定のタグキーと値が存在する必要があります。

条件キー	操作	説明
s3:max-keys	s3:ListBucket および 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:object-lock-mode	s3:PutObject	`object-lock-mode`から展開された値と比較します。これは、PutObject、CopyObject、およびCreateMultipartUploadリクエストのリクエストヘッダーから展開されます。
s3:object-lock-mode	s3:PutObjectRetention	PutObjectRetentionリクエストのXML本文から展開された `object-lock-mode`と比較します。
s3:object-lock-remaining-retention-days	s3:PutObject	`x-amz-object-lock-retain-until-date`リクエストヘッダーで指定された保持期限、またはバケットのデフォルト保持期間から計算された保持期限と比較して、これらの値が次のリクエストの許容範囲内にあることを確認します： • PutObject • CopyObject • CreateMultipartUpload
s3:object-lock-remaining-retention-days	s3:PutObjectRetention	PutObjectRetention リクエストで指定された保持期限日と比較し、許容範囲内であることを確認します。
s3:prefix	s3:ListBucket および 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/<tag-key>	s3:PutObject s3:PutObjectTagging s3:PutObjectVersionTagging	オブジェクト要求にタグ付けが含まれる場合、特定のタグキーと値が必要になります。

条件キー	操作	説明
s3:x-amz-server-side-encryption-customer-algorithm	s3:PutObject	`sse-customer-algorithm`または`copy-source-sse-customer-algorithm`と比較します。これらは、PutObject、CopyObject、CreateMultipartUpload、UploadPart、UploadPartCopy、およびCompleteMultipartUploadリクエストのリクエストヘッダーから展開されます。

ポリシーで変数を指定する

ポリシー内で変数を使用すると、ポリシー情報が利用可能な場合に、その情報を入力できます。ポリシー変数は `Resource`要素と、 `Condition`要素内の文字列比較で使用できます。

この例では、変数 `${aws:username}`は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}`は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}	SourceIp キーを指定された変数として使用します。
${aws:username}	指定された変数としてユーザー名キーを使用します。
${s3:prefix}	サービス固有のプレフィックスキーを、指定された変数として使用します。
${s3:max-keys}	指定された変数として、サービス固有のmax-keysキーを使用します。
${*}	特殊文字。この文字をリテラルの「*」文字として使用します。
${?}	特殊文字。文字をリテラルの「?」文字として使用します。

変数	説明
`${$}`	特殊文字。文字をリテラルの $ 文字として使用します。

特別な取り扱いが必要なポリシーを作成する

ポリシーによっては、セキュリティ上危険であったり、継続的な運用に危険を及ぼすような権限を付与してしまう場合があります。例えば、アカウントのルートユーザーを締め出してしまうといった場合です。StorageGRID S3 REST API の実装は、ポリシー検証時には Amazon よりも制限が少ないですが、ポリシー評価時には Amazon と同程度に厳格です。

ポリシーの説明	ポリシータイプ	**Amazon** の動作	**StorageGRID**の動作
ルートアカウントに対するすべての権限を自分自身に拒否する	バケット	有効かつ強制適用されますが、root 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に対する権限を保持します。	同じ
ユーザー/グループに対する自身の権限をすべて拒否する	グループ	有効かつ施行済み	同じ
外部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、ポリシーで許可されている場合、すべての S3 バケットポリシー操作の権限は 405 Method Not Allowed 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すべての権限を許可する	バケット	有効ですが、ポリシーで許可されている場合、すべての S3 バケットポリシー操作の権限は 405 Method Not Allowed エラーを返します	同じ
すべてのアクションに対して全員に権限を付与する	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 root とユーザーに対して 405 Method Not Allowed エラーを返します。	同じ
すべてのアクションに対する全員の権限を拒否する	バケット	有効かつ強制適用されますが、root 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に対する権限を保持します。	同じ

ポリシーの説明	ポリシータイプ	**Amazon** の動作	**StorageGRID**の動作
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
プリンシパルはローカルグループです	バケット	無効なプリンシパル	有効
このポリシーは、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。オブジェクトの所有権は作成者アカウントにあり、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。オブジェクトの所有権は、バケット所有者のアカウントに帰属します。バケットポリシーが適用されます。

Write-once-read-many（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バケットを設定することで、新しいオブジェクトの作成を許可し、既存コンテンツの上書きや削除を防止できます。ここで説明する方法のいずれかを使用してください。

上書きが常に拒否されるようにするには、次の方法があります。

- グリッドマネージャから、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、クライアントによる変更を防止する チェックボックスを選択します。
- 以下のルールと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 の DENY 操作を追加します。
  - S3ポリシーに PutObject の 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日後にコピーをゼロにする」などのルールが存在する場合、ILM によるオブジェクトの削除を防ぐことはできません。

> これらのルールやポリシーをすべて適用したとしても、同時書き込みを防ぐことはできません（状況Aを参照）。これらは、完了した連続上書きを防ぐ対策を講じています（状況Bを参照）。

状況**A**：同時書き込み（対策なし）

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

状況**B**：連続した上書き処理（対策済み）

```
/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"
- "セッションポリシーの例"
- "ILMを使用して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"

StorageGRID S3 REST API セッションポリシーの例

次の例を使用して、StorageGRID セッションポリシーを作成します。

例：オブジェクトの取得を許可するセッションポリシーを設定する

この例では、セッションのプリンシパルはbucket1からオブジェクトを取得することのみが許可されています。その他のアクションはすべて暗黙的に拒否されますが、"s3:PutOverwriteObject"権限の使用などのStorageGRID固有のアクションは除きます。セッションポリシーは、AssumeRole APIの呼び出し時にJSONファイルとして提供できます。

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

StorageGRID S3 REST API 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作成します。

バケットポリシーは、そのポリシーが適用されるバケットへのアクセス権限を指定します。S3 PutBucketPolicy API を使用して、以下のいずれかのツールからバケットポリシーを設定します：

- "Tenant Manager"。
- AWS CLI でこのコマンドを使用します（"バケットに対する操作" を参照）：

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

例：バケットへの読み取り専用アクセスを全員に許可する

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操作を実行することができます。その他のすべての操作は拒否されます。アカウントのルート以外にはバケットへの書き込み権限がないため、このポリシーはあまり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権限を、別のアカウントの全員には読み取り専用アクセス権限を付与する

この例では、指定された1つのアカウントのすべてのユーザーにはバケットへのフルアクセスが許可されますが、別の指定されたアカウントのすべてのユーザーには、バケットの一覧表示と、`shared/`オブジェクトキープレフィックスで始まるバケット内のオブジェクトに対するGetObject操作の実行のみが許可されます。

StorageGRID では、非所有者アカウント（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が所有します。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が、指定されたアカウントのグループ `Marketing`に属するユーザーのみがフルアクセス権限を持ちます。

```
{
  "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：クライアントがIPアドレス範囲内にいる場合、バケットへの読み取りおよび書き込みアクセスを全員に許可する

この例では、匿名ユーザーを含むすべてのユーザーが、指定されたIP範囲（54.240.143.0～54.240.143.255、ただし54.240.143.188を除く）からのリクエストであれば、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。その他の操作はすべて拒否され、IPアドレス範囲外からのすべてのリクエストも拒否されます。

```
{
  "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にバケットへのフルアクセスを許可する

この例では、フェデレーションユーザーの Alex には、 `examplebucket`バケットとそのオブジェクトへのフルアクセスが許可されています。'root' を含むその他のすべてのユーザーは、すべての操作を明示的に拒否されます。ただし、'root' は Put/Get/DeleteBucketPolicy の権限を拒否されることはありません。

```
{
  "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 Effect 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 オブジェクトタグが誰にも上書きまたは削除されないことが保証されます。

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

StorageGRID S3 REST API グループポリシーの例

このセクションの例を使用して、グループ向けの StorageGRID アクセスポリシーを作成します。

グループポリシーは、そのポリシーが適用されるグループのアクセス権限を指定します。ポリシー内に`Principal`要素はありません。これは暗黙的であるためです。グループポリシーは、テナントマネージャーまたはAPIを使用して設定します。

例：**Tenant Manager** 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際に、グループポリシーを選択して、そのグループのメンバーが持つS3アクセス権限を決定できます。"S3テナント用のグループを作成する"を参照してください。

- **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、バケットポリシーによって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は、S3リソースへの読み取り専用アクセス権限を持ちます。例えば、このグループのユーザーは、オブジェクトの一覧表示や、オブジェクトのデータ、メタデータ、タグの読み取りを行うことができます。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は、バケットを含む S3 リソースへのフルアクセス権限を持ちます。このオプションを選択すると、フルアクセス権限を持つ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。
- ランサムウェア対策：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操作を実行できますが、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。

  「すべてのバケットを管理する」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。「すべてのバケットを管理する」権限は信頼できるユーザーのみに制限し、利用可能な場合は多要素認証（MFA）を使用してください。

- 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は、テナントアカウントが所有するすべてのバケットへの完全なアクセスが許可されます。

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

例：グループに対してすべての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、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。例えば、このグループのユーザーは、オブジェクトの一覧表示や、オブジェクトのデータ、メタデータ、タグの読み取りを行うことができます。

```
{
  "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「フォルダー」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自分の特定のフォルダ（キープレフィックス）のみを一覧表示およびアクセスすることが許可されています。これらのフォルダのプライバシー設定を決定する際には、他のグループポリシーおよびバケットポリシーによるアクセス権限も考慮する必要があります。

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

StorageGRID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。監査ログにある S3 固有の監査メッセージを確認することで、バケットやオブジェクトの操作に関する詳細情報を取得できます。

監査ログに記録されるバケット操作

- CreateBucket
- DeleteBucket
- DeleteBucketTagging
- DeleteObjects
- GetBucketTagging
- HeadBucket
- ListObjects
- ListObjectVersions
- PUT Bucket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
- RestoreObject
- SelectObject
- UploadPart（ILMルールがバランス型または厳密型の取り込みを使用する場合）
- UploadPartCopy（ILMルールがバランス型または厳密型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"
- "クライアントが監査メッセージを書き込む"
- "クライアント読み取り監査メッセージ"

Swift REST APIの使用（サポート終了）

StorageGRID での Swift REST API サポート

Swift APIのサポートはサポート終了（EOL）となり、今後のリリースで削除される予定です。

このバージョンのドキュメントサイトからは、Swiftに関する詳細情報が削除されました。参照してください "StorageGRID 11.8: Swift REST APIの使用"。

著作権に関する情報

商標に関する情報