



Swift REST APIの使用（廃止）

StorageGRID 11.7

NetApp
April 12, 2024

目次

Swift REST APIの使用（廃止）	1
Swift REST APIの使用：概要	1
テナントアカウントと接続を設定する	4
Swift REST API でサポートされている処理	8
StorageGRID の Swift REST API 処理	21
REST API のセキュリティを設定する	25
監視と監査の処理	27

Swift REST APIの使用（廃止）

Swift REST APIの使用：概要

クライアントアプリケーションでは、OpenStack Swift API を使用して、StorageGRID システムを操作できます。



Swiftクライアントアプリケーションのサポートは廃止され、今後のリリースで削除される予定です。

StorageGRID でサポートしている Swift および HTTP のバージョンは次のとおりです。

項目	バージョン
Swift の仕様	2015 年 11 月時点の OpenStack Swift Object Storage API v1
HTTP	1.1 HTTP の詳細については、HTTP/1.1（RFC 7230~7235）を参照してください。 • 注：StorageGRID は、HTTP/1.1 パイプラインをサポートしません。

関連情報

["OpenStack：オブジェクトストレージ API"](#)

StorageGRID での Swift API サポートの履歴

StorageGRID システムでの Swift REST API のサポートに関する変更点に注意する必要があります。

リリース。	コメント
11.7	Swiftクライアントアプリケーションのサポートは廃止され、今後のリリースで削除される予定です。
11.6	編集上のいくつかの変更点。
11.5	弱い整合性制御を削除しました。代わりに、available 整合性レベルが使用されます。
11.4	TLS 1.3のサポートが追加されました。ILM と整合性設定の間の相互関係の概要が追加されました。

リリース。	コメント
11.3	PUT Object 処理が更新され、取り込み時に同期配置を使用する ILM ルールの影響（取り込み動作の Balanced オプションと Strict オプション）が記述されるようになりました。ロードバランサエンドポイントまたはハイアベイラビリティグループを使用するクライアント接続の概要が追加されました。TLS 1.1 暗号はサポートされなくなりました。
11.2	ドキュメントに対する編集上の変更がいくつかあります。
11.1	グリッドノードへの Swift クライアント接続での HTTP の使用のサポートが追加されました。整合性制御の定義が更新されました。
11.0	テナントアカウントにつき 1、000 個のコンテナのサポートが追加されました。
10.3	ドキュメントの管理に関する記述の更新と修正カスタムサーバ証明書の設定に関するセクションが削除されました。
10.2	StorageGRID システムで Swift API が初めてサポートされました。現在サポートされているバージョンは、OpenStack Swift Object Storage API v1 です。

StorageGRID での Swift REST API の実装

クライアントアプリケーションは、Swift REST API 呼び出しを使用してストレージノードやゲートウェイノードに接続し、コンテナの作成やオブジェクトの格納と読み出しを行うことができます。これを利用して、OpenStack Swift 向けに開発されたサービス指向アプリケーションを、StorageGRID システムで利用できるオンプレミスのオブジェクトストレージに接続することができます。

Swift オブジェクトの管理

StorageGRID システムに取り込まれた Swift オブジェクトは、システムのアクティブな ILM ポリシー内の情報ライフサイクル管理（ILM）ルールによって管理されます。。"[ILM ルール](#)" および "[ILM ポリシー](#)" StorageGRID でオブジェクトデータのコピーを作成および分散し、一定の期間にわたって管理する方法を決定します。たとえば、ILM ルールを特定の Swift コンテナ内のオブジェクトに適用し、複数のオブジェクトコピーを複数のデータセンターに一定期間保存するように指定できます。

グリッドの ILM ルールとポリシーが Swift テナントアカウントのオブジェクトに与える影響については、ネットアッププロフェッショナルサービスのコンサルタントまたは StorageGRID 管理者にお問い合わせください。

競合するクライアント要求です

同じキーに書き込む 2 つのクライアントなど、競合するクライアント要求は、「latest-wins」ベースで解決されます。「latest-wins」評価は、Swift クライアントが処理を開始するタイミングではなく、StorageGRID システムが特定の要求を完了したタイミングで行われます。

整合性の保証と制御

デフォルトでは、StorageGRID は、新規作成されたオブジェクトにはリードアフターライト整合性を、オブジェクトの更新と HEAD 処理には結果整合性を提供します。任意 "取得" 正常に完了しました "PUT" 新しく書き込まれたデータを読み取ることができます。既存のオブジェクトの上書き、メタデータの更新、および削除の整合性レベルは、結果整合性です。上書きは通常、数秒から数分で反映されますが、最大で 15 日かかることがあります。

StorageGRID では、コンテナごとに整合性を制御することもできます。整合性制御では、アプリケーションの必要に応じて、オブジェクトの可用性と異なるストレージノードおよびサイト間でのオブジェクトの整合性のバランスを調整できます。

Swift REST API を実装する際の推奨事項

StorageGRID で使用するために Swift REST API を実装する場合は、次の推奨事項を考慮してください。

存在しないオブジェクトに対する HEAD の推奨事項

オブジェクトが実際に存在するとは思わないパスにオブジェクトが存在するかどうかをアプリケーションが定期的にチェックする場合は、「使用可能」整合性制御を使用する必要があります。たとえば 'アプリケーションがそのロケーションに対して PUT 操作を実行する前に 'そのロケーションに対して HEAD 操作を実行する場合は 'Available 整合性制御を使用する必要があります

そうしないと、使用できないストレージノードがある場合に HEAD 処理でオブジェクトが見つからないと、「500 Internal Server Error」が大量に返される可能性があります。

を使用して、各コンテナに「使用可能」の整合性制御を設定できます "PUT コンテナセイコウセイヨウキユウ"。を使用して、各コンテナの整合性制御を「使用可能」に設定します "GET コンテナセイコウセイヨウキユウ"。

オブジェクト名の推奨事項

StorageGRID 11.4 以降で作成されたコンテナの場合、オブジェクト名がパフォーマンスのベストプラクティスに適合するように制限する必要はなくなりました。たとえば、オブジェクト名の最初の 4 文字にランダムな値を使用できるようになりました。

StorageGRID 11.4 よりも前のリリースで作成されたコンテナの場合は、オブジェクト名に関する次の推奨事項に進みます。

- オブジェクト名の最初の 4 文字に、ランダムな値を使用しないでください。これは、AWS が以前に推奨していた名前プレフィックスの推奨とは異なります。代わりに、などの一意ではないランダムなプレフィックスを使用してください image。
- 名前のプレフィックスにランダムな一意の文字を使用するように AWS の以前の推奨事項に従っている場合は、オブジェクト名の前にディレクトリ名を指定する必要があります。つまり、次の形式を使用します。

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

次の形式は使用しないでください。

mycontainer/f8e3-image3132.jpg

「範囲の読み取り」に関する推奨事項

状況に応じて **"格納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は、Swiftクライアントアプリケーションで、返されるバイト数の範囲を指定するGET Object処理を実行しないでください。StorageGRID は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圧縮解除する必要があるため、これらの "range read" 操作は非効率的です。非常に大きなオブジェクトから小さい範囲のバイト数を要求するGET Object 処理は特に効率が悪く、たとえば、50GB の圧縮オブジェクトから 10MB の範囲を読み取る処理は非常に非効率的です。

圧縮オブジェクトから範囲を読み取ると、クライアント要求がタイムアウトする可能性があります。



オブジェクトを圧縮する必要があり、クライアントアプリケーション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増やしてください。

テナントアカウントと接続を設定する

クライアントアプリケーションからの接続を受け入れるように StorageGRID を設定するには、テナントアカウントを 1 つ以上作成し、接続を設定する必要があります。

Swift テナントアカウントを作成および設定します

Swift API クライアントで StorageGRID に対してオブジェクトの格納や読み出しを行うには、Swift テナントアカウントが必要です。各テナントアカウントには、専用のアカウント ID、専用のグループとユーザ、および専用のコンテナとオブジェクトがあります。

Swift テナントアカウントは、StorageGRID のグリッド管理者がグリッドマネージャまたはグリッド管理 API を使用して作成します。

いつ **"Swiftテナントアカウントを作成する"**グリッド管理者は次の情報を指定します。

- **"テナントの表示名"** (テナントのアカウントIDは自動的に割り当てられ、変更できません)
- オプションで、を指定します **"テナントアカウントのストレージクォータ"**-テナントのオブジェクトに使用できる最大のギガバイト数、テラバイト数、またはペタバイト数。テナントのストレージクォータは、物理容量 (ディスクのサイズ) ではなく、論理容量 (オブジェクトのサイズ) を表します。
- 状況 **"シングルサインオン (SSO) "** StorageGRID システムでは使用されていない。テナントアカウントが独自のアイデンティティソースを使用するかグリッドのアイデンティティソースを共有するか、およびテナントのローカルrootユーザの初期パスワード。
- SSOが有効な場合は、テナントアカウントを設定するためのRoot Access権限が割り当てられているフェデレーテッドグループ。

Swiftテナントアカウントを作成すると、Root Access権限が割り当てられたユーザは、Tenant Managerにアクセスして次のタスクを実行できるようになります。

- アイデンティティフェデレーションの設定 (グリッドとアイデンティティソースを共有する場合を除く)、およびローカルグループとユーザの作成

- ストレージ使用状況を監視しています



Swiftユーザには、に対するRootアクセス権限が必要です ["Tenant Managerにアクセスします"](#)。ただし Root Access 権限では、Swift REST API に認証してコンテナを作成したりオブジェクトを取り込んだりすることはできません。Swift REST API に認証するには、Swift 管理者の権限が必要です。

クライアント接続の設定方法

グリッド管理者は、Swift クライアントがデータの格納と読み出しを行うために StorageGRID に接続する方法に関連する設定を行います。接続するために必要な具体的な情報は、選択した設定によって異なります。

クライアントアプリケーションは、管理ノードまたはゲートウェイノード上のロードバランササービスに接続するか、必要に応じて管理ノードまたはゲートウェイノードのハイアベイラビリティ (HA) グループの仮想IPアドレスに接続することで、オブジェクトの格納や読み出しを行うことができます。



ロードバランシングにStorageGRID を使用するすべてのアプリケーションは、ロードバランササービスを使用して接続する必要があります。

- 外部ロードバランサを使用するかどうかに関係なく、ストレージノードに追加されます

StorageGRID を設定する場合、グリッド管理者はグリッドマネージャまたはグリッド管理 API を使用して次の手順を実行できます。これらはすべてオプションです。

1. ロードバランササービスのエンドポイントを設定する。

ロードバランササービスを使用するようにエンドポイントを設定する必要があります。管理ノードまたはゲートウェイノード上のロードバランササービスは、クライアントアプリケーションからの受信ネットワーク接続を複数のストレージノードに分散します。ロードバランサエンドポイントを作成する際、StorageGRID 管理者は、ポート番号、エンドポイントで HTTP / HTTPS 接続を許可するかどうか、エンドポイントを使用するクライアントのタイプ (S3 または Swift)、HTTPS 接続に使用する証明書 (該当する場合) を指定します。Swiftはこれらをサポートしています ["エンドポイントタイプ"](#)。

2. 信頼されていないクライアントネットワークを設定する

StorageGRID 管理者がノードのクライアントネットワークを信頼されていないクライアントネットワークとして設定した場合、ノードはロードバランサエンドポイントとして明示的に設定されたポートでクライアントネットワークのインバウンド接続だけを受け入れます。

3. ハイアベイラビリティグループを設定する。

管理者が HA グループを作成すると、複数の管理ノードまたはゲートウェイノードのネットワークインターフェイスがアクティブ / バックアップ構成になります。クライアント接続は、HA グループの仮想 IP アドレスを使用して確立されます。

を参照してください ["HA グループの設定オプション"](#) を参照してください。

Summary : クライアント接続の IP アドレスとポート

クライアントアプリケーションは、グリッドノードの IP アドレスおよびそのノード上のサービスのポート番号を使用して StorageGRID に接続します。ハイアベイラビリティ (HA) グループが設定されている場合

は、HAグループの仮想IPアドレスを使用してクライアントアプリケーションを接続できます。

クライアント接続に必要な情報

次の表に、クライアントがStorageGRIDに接続できるさまざまな方法、および各接続タイプで使用されるIPアドレスとポートを示します。を参照してください "[クライアント接続用のIPアドレスとポート](#)" 詳細については、StorageGRID 管理者にお問い合わせください。

接続が確立される場所	クライアントが接続するサービス	IP アドレス	ポート
HA グループ	ロードバランサ	HA グループの仮想 IP アドレス	• ロードバランサエンドポイントのポート
管理ノード	ロードバランサ	管理ノードの IP アドレス	• ロードバランサエンドポイントのポート
ゲートウェイノード	ロードバランサ	ゲートウェイノードの IP アドレス	• ロードバランサエンドポイントのポート
ストレージノード	LDR	ストレージノードの IP アドレス	デフォルトの Swift ポート： • HTTPS : 18083 • HTTP : 18085

例

Swift クライアントをゲートウェイノードの HA グループのロードバランサエンドポイントに接続するには、次のように構造化された URL を使用します。

- `https://VIP-of-HA-group:LB-endpoint-port`

たとえば、HAグループの仮想IPアドレスが192.0.2.6で、Swiftロードバランサエンドポイントのポート番号が10444の場合、Swiftクライアントは次のURLを使用してStorageGRIDに接続できます。

- `https://192.0.2.6:10444`

クライアントがStorageGRIDへの接続に使用するIPアドレスにDNS名を設定できます。ローカルネットワーク管理者にお問い合わせください。

HTTPS 接続または HTTP 接続を使用するかどうかを決定します

ロードバランサエンドポイントを使用してクライアント接続を行う場合は、そのエンドポイントに指定されているプロトコル（HTTP または HTTPS）を使用して接続を確立する必要があります。ストレージノードへのクライアント接続にHTTPを使用するには、HTTPの使用を有効にする必要があります。

デフォルトでは、クライアントアプリケーションがストレージノードに接続する際に、すべての接続に暗号化されたHTTPSを使用する必要があります。必要に応じて、を選択してセキュアでないHTTP接続を有効にすることもできます "[ストレージノード接続用のHTTPを有効にします](#)" オプションを選択します。たとえば、非本

番環境でストレージノードへの接続をテストする際に、クライアントアプリケーションで HTTP を使用できません。



要求と応答が暗号化されずに送信されるため、本番環境のグリッドでHTTPを有効にする場合は注意が必要です。

[ストレージノード接続用のHTTPを有効にする]*オプションが選択されている場合、クライアントはHTTPSとは別のポートをHTTPに使用する必要があります。

Swift API 設定で接続をテストします

Swift の CLI を使用して、StorageGRID システムへの接続をテストし、システムに対するオブジェクトの読み取りと書き込みが可能であることを確認できます。

作業を開始する前に

- Swift のコマンドラインクライアント `python-swiftclient` をダウンロードしてインストールしておく必要があります。

"swiftStack : `python-swiftclient`"

- StorageGRID システムに Swift テナントアカウントが必要です。

このタスクについて

セキュリティを設定していない場合は、を追加する必要があります `--insecure` これらの各コマンドにフラグを設定します。

手順

1. StorageGRID Swift 環境の情報 URL を照会します。

```
swift
-U <Tenant_Account_ID:Account_User_Name>
-K <User_Password>
-A https://<FQDN | IP>:<Port>/info
capabilities
```

この手順で、Swift 環境が機能することをテストできます。オブジェクトを格納してアカウント設定をさらにテストするには、以降の手順を実行します。

2. オブジェクトをコンテナに配置します。

```
touch test_object
swift
-U <Tenant_Account_ID:Account_User_Name>
-K <User_Password>
-A https://<FQDN | IP>:<Port>/auth/v1.0
upload test_container test_object
--object-name test_object
```

3. コンテナを取得してオブジェクトを確認します。

```
swift
-U <Tenant_Account_ID:Account_User_Name>
-K <User_Password>
-A https://<FQDN | IP>:<Port>/auth/v1.0
list test_container
```

4. オブジェクトを削除します。

```
swift
-U <Tenant_Account_ID:Account_User_Name>
-K <User_Password>
-A https://<FQDN | IP>:<Port>/auth/v1.0
delete test_container test_object
```

5. コンテナを削除します。

```
swift
-U `<_Tenant_Account_ID:Account_User_Name_>`
-K `<_User_Password_>`
-A `https://<_FQDN_ | _IP_>:<_Port_>/auth/v1.0`
delete test_container
```

関連情報

["Swift テナントアカウントを作成および設定します"](#)

["REST API のセキュリティを設定する"](#)

Swift REST API でサポートされている処理

StorageGRID システムは、OpenStack Swift API のほとんどの処理をサポートしています。Swift REST API クライアントを StorageGRID に統合する前に、アカウント、コン

テナ、およびオブジェクトの処理の実装に関する詳細を確認します。

StorageGRID でサポートされている操作

次の Swift API 処理がサポートされています。

- "アカウントの処理"
- "コンテナの処理"
- "オブジェクトの処理"

すべての処理に共通の応答ヘッダー

StorageGRID システムでは、OpenStack Swift Object Storage API v1 の定義に従って、サポートされるすべての処理に共通のヘッダーが実装されます。

関連情報

["OpenStack : オブジェクトストレージ API"](#)

サポートされている Swift API エンドポイント

StorageGRID でサポートされている Swift API エンドポイントは、情報 URL、認証 URL、およびストレージ URL です。

情報 URL

StorageGRID Swift 実装の機能と制限事項については、Swift のベース URL に /info パスを付加して GET 要求を発行することで確認できます。

```
https://FQDN | Node IP:Swift Port/info/
```

要求の内容は次のとおりです。

- *FQDN* は完全修飾ドメイン名です。
- *Node IP* は、StorageGRID ネットワークのストレージノードまたはゲートウェイノードの IP アドレスです。
- *Swift Port* は、ストレージノードまたはゲートウェイノードの Swift API 接続に使用するポート番号です。

たとえば、次の情報 URL は、IP アドレスが 10.99.106.103 でポート 18083 を使用しているストレージノードから情報を要求します。

```
https://10.99.106.103:18083/info/
```

応答には、Swift 実装の機能が JSON ディクショナリとして含まれます。クライアントツールは、JSON 応答を解析して実装の機能を特定し、後続のストレージ処理で制約として使用できます。

StorageGRID 実装の Swift では、情報 URL への認証されていないアクセスが許可されます。

認証 URL

クライアントは、Swift 認証 URL を使用してテナントアカウントユーザとして認証できます。

```
https://FQDN | Node IP:Swift Port/auth/v1.0/
```

で、テナントアカウントID、ユーザ名、およびパスワードをパラメータとして指定する必要があります x-Auth-User および X-Auth-Key 次のように要求ヘッダー

```
X-Auth-User: Tenant_Account_ID:Username
```

```
X-Auth-Key: Password
```

要求ヘッダーは次のようになります。

- *Tenant_Account_ID* は、Swiftテナントの作成時にStorageGRID によって割り当てられたアカウントID です。Tenant Manager のサインインページで使用するテナントアカウント ID と同じです。
- *Username* は、Tenant Managerで作成されたテナントユーザの名前です。このユーザは、Swift 管理者権限を持つグループに属している必要があります。テナントのrootユーザをSwift REST APIを使用するように設定することはできません。

テナントアカウントに対してアイデンティティフェデレーションが有効になっている場合は、LDAP サーバからのフェデレーテッドユーザのユーザ名とパスワードを指定します。または、LDAP ユーザのドメイン名を指定します。例：

```
X-Auth-User: Tenant_Account_ID:Username@Domain_Name
```

- *Password* は、テナントユーザのパスワードです。ユーザパスワードは Tenant Manager で作成および管理します。

認証要求が成功すると、ストレージ URL と認証トークンが次のように返されます。

```
X-Storage-Url: https://FQDN | Node_IP:Swift_Port/v1/Tenant_Account_ID
```

```
X-Auth-Token: token
```

```
X-Storage-Token: token
```

デフォルトでは、トークンの有効期間は生成時刻から 24 時間です。

トークンは特定のテナントアカウントに対して生成されます。あるアカウントに対して有効なトークンで、別のアカウントにアクセスするユーザを許可することはできません。

ストレージ URL

クライアントアプリケーションは、ゲートウェイノードまたはストレージノードに対して、問題の Swift REST API 呼び出しを使用して、アカウント、コンテナ、オブジェクトのサポートされる処理を実行できます。ストレージ要求は、認証応答で返されたストレージ URL にアドレスが指定されます。要求には、認証要求から返された X-Auth-Token ヘッダーと値も含める必要があります。

```
https://FQDN | IP:Swift_Port/v1/Tenant_Account_ID
```

[/container] [/object]

X-Auth-Token: token

使用状況の統計が含まれるストレージ応答ヘッダーに、最近変更されたオブジェクトの正確な数が反映されない場合があります。このヘッダーに正確な数値が表示されるまでに数分かかることがあります。

使用状況の統計が含まれているアカウントおよびコンテナ処理の応答ヘッダーの例を次に示します。

- X-Account-Bytes-Used
- X-Account-Object-Count
- X-Container-Bytes-Used
- X-Container-Object-Count

関連情報

["テナントアカウントと接続を設定する"](#)

["アカウントの処理"](#)

["コンテナの処理"](#)

["オブジェクトの処理"](#)

アカウントの処理

アカウントに対して実行する Swift API 処理を次に示します。

GET アカウント

この処理は、アカウントに関連付けられているコンテナリストおよびアカウントの使用状況を示す統計を取得します。

次の要求パラメータが必要です。

- Account

次の要求ヘッダーが必要です。

- X-Auth-Token

次のサポートされている要求クエリパラメータはオプションです。

- Delimiter
- End_marker
- Format
- Limit
- Marker

- Prefix

実行が成功すると 'アカウントが見つかってコンテナがないかコンテナリストが空である場合' またはアカウントが見つかってコンテナリストが空でない場合には 'HTTP/1.1 204 No Content' の応答とともに '次のヘッダーが返され' コンテナリストが空でない場合は 'HTTP/1.1 200 OK' の応答が返されます

- Accept-Ranges
- Content-Length
- Content-Type
- Date
- X-Account-Bytes-Used
- X-Account-Container-Count
- X-Account-Object-Count
- X-Timestamp
- X-Trans-Id

HEAD アカウント

この処理は、Swift アカウントからアカウント情報と統計情報を取得します。

次の要求パラメータが必要です。

- Account

次の要求ヘッダーが必要です。

- X-Auth-Token

実行が成功すると、「HTTP/1.1 204 No Content」の応答とともに次のヘッダーが返されます。

- Accept-Ranges
- Content-Length
- Date
- X-Account-Bytes-Used
- X-Account-Container-Count
- X-Account-Object-Count
- X-Timestamp
- X-Trans-Id

関連情報

["監視と監査の処理"](#)

コンテナの処理

StorageGRID では、Swift アカウントあたり最大で 1、000 個のコンテナがサポートされます。コンテナに対して実行する Swift API 処理を次に示します。

コンテナを削除します

この処理は、StorageGRID システムの Swift アカウントから空のコンテナを削除します。

次の要求パラメータが必要です。

- Account
- Container

次の要求ヘッダーが必要です。

- X-Auth-Token

実行が成功すると、「HTTP/1.1 204 No Content」の応答とともに次のヘッダーが返されます。

- Content-Length
- Content-Type
- Date
- X-Trans-Id

GET コンテナ

この処理は、コンテナに関連付けられているオブジェクトリストを、StorageGRID システム内のコンテナの統計情報およびメタデータとともに読み出します。

次の要求パラメータが必要です。

- Account
- Container

次の要求ヘッダーが必要です。

- X-Auth-Token

次のサポートされている要求クエリパラメータはオプションです。

- Delimiter
- End_marker
- Format
- Limit
- Marker

- Path
- Prefix

実行が成功すると、「HTTP/1.1 200 Success」または「HTTP/1.1 204 No Content」の応答とともに次のヘッダーが返されます。

- Accept-Ranges
- Content-Length
- Content-Type
- Date
- X-Container-Bytes-Used
- X-Container-Object-Count
- X-Timestamp
- X-Trans-Id

HEAD コンテナ

この処理は、StorageGRID システムからコンテナの統計情報とメタデータを読み出します。

次の要求パラメータが必要です。

- Account
- Container

次の要求ヘッダーが必要です。

- X-Auth-Token

実行が成功すると、「HTTP/1.1 204 No Content」の応答とともに次のヘッダーが返されます。

- Accept-Ranges
- Content-Length
- Date
- X-Container-Bytes-Used
- X-Container-Object-Count
- X-Timestamp
- X-Trans-Id

PUT コンテナ

この処理は、StorageGRID システムのアカウントにコンテナを作成します。

次の要求パラメータが必要です。

- Account
- Container

次の要求ヘッダーが必要です。

- X-Auth-Token

実行が成功すると、「HTTP/1.1 201 Created」または「HTTP/1.1 202 Accepted」の応答（このアカウントにコンテナがすでに存在する場合）とともに次のヘッダーが返されます。

- Content-Length
- Date
- X-Timestamp
- X-Trans-Id

コンテナ名は StorageGRID ネームスペース内で一意である必要があります。このコンテナが別のアカウントの下に存在する場合は、ヘッダー「HTTP/1.1 409 Conflict」が返されます。

関連情報

["監視と監査の処理"](#)

オブジェクトの処理

オブジェクトに対して実行する Swift API 処理を次に示します。これらの処理はで追跡できます ["StorageGRID 監査ログ"](#)。

オブジェクトを削除します

この処理は、オブジェクトのコンテンツとメタデータを StorageGRID システムから削除します。

次の要求パラメータが必要です。

- Account
- Container
- Object

次の要求ヘッダーが必要です。

- X-Auth-Token

実行が成功すると、が指定された次の応答ヘッダーが返されます HTTP/1.1 204 No Content 対応：

- Content-Length
- Content-Type
- Date
- X-Trans-Id

StorageGRID は、DELETE Object 要求を処理する際に、オブジェクトのすべてのコピーをすべての格納場所からただちに削除しようとします。成功すると、StorageGRID はただちにクライアントに応答を返します。30秒以内にすべてのコピーを削除できない場合（場所が一時的に使用できない場合など）、StorageGRID は削除対象のコピーをキューに登録し、クライアントに成功を通知します。

詳細については、を参照してください "[オブジェクトの削除方法](#)"。

GET オブジェクト

この処理は、StorageGRID から、オブジェクトのコンテンツを読み出し、オブジェクトメタデータを取得します。

次の要求パラメータが必要です。

- Account
- Container
- Object

次の要求ヘッダーが必要です。

- X-Auth-Token

次の要求ヘッダーはオプションです。

- Accept-Encoding
- If-Match
- If-Modified-Since
- If-None-Match
- If-Unmodified-Since
- Range

実行が成功すると、が指定された次のヘッダーが返されます HTTP/1.1 200 OK 対応：

- Accept-Ranges
- Content-Disposition`の場合にのみ返されます `Content-Disposition`メタデータが設定されました
- Content-Encoding`の場合にのみ返されます `Content-Encoding`メタデータが設定されました
- Content-Length
- Content-Type
- Date
- ETag
- Last-Modified
- X-Timestamp

- X-Trans-Id

HEAD オブジェクト

この処理は、取り込まれたオブジェクトのメタデータとプロパティを StorageGRID システムから読み出します。

次の要求パラメータが必要です。

- Account
- Container
- Object

次の要求ヘッダーが必要です。

- X-Auth-Token

実行が成功すると、「HTTP/1.1 200 OK」の応答とともに次のヘッダーが返されます。

- Accept-Ranges
- Content-Disposition`の場合にのみ返されます `Content-Disposition`メタデータが設定されました
- Content-Encoding`の場合にのみ返されます `Content-Encoding`メタデータが設定されました
- Content-Length
- Content-Type
- Date
- ETag
- Last-Modified
- X-Timestamp
- X-Trans-Id

PUT オブジェクト

この処理は、StorageGRID システムで、データとメタデータを含む新しいオブジェクトを作成するか、データとメタデータを含む既存のオブジェクトを置換します。

StorageGRID では、サイズが 5TiB（5、497、558、138、880 バイト）までのオブジェクトがサポートされます。



同じキーに書き込む 2 つのクライアントなど、競合するクライアント要求は、「latest-wins」ベースで解決されます。「latest-wins」評価は、Swift クライアントが処理を開始するタイミングではなく、StorageGRID システムが特定の要求を完了したタイミングで行われます。

次の要求パラメータが必要です。

- Account

- Container
- Object

次の要求ヘッダーが必要です。

- X-Auth-Token

次の要求ヘッダーはオプションです。

- Content-Disposition
- Content-Encoding

chunkedは使用しないでください Content-Encoding 環境 オブジェクトがサイズに基づいてオブジェクトをフィルタリングし、取り込み時に同期配置を使用するILMルール（取り込み動作にBalancedオプションまたはStrictオプション）の場合。

- Transfer-Encoding

圧縮やチャックは使用しないでください Transfer-Encoding 環境 オブジェクトがサイズに基づいてオブジェクトをフィルタリングし、取り込み時に同期配置を使用するILMルール（取り込み動作にBalancedオプションまたはStrictオプション）の場合。

- Content-Length

ILMルールで、オブジェクトがサイズでフィルタリングされ、取り込み時に同期配置が使用される場合は、を指定する必要があります Content-Length。



のこれらのガイドラインに従わない場合 Content-Encoding、`Transfer-Encoding`および`Content-Length`ではStorageGRID、オブジェクトのサイズを確認してILMルールを適用する前に、オブジェクトを保存しておく必要があります。つまり、StorageGRIDで取り込み時にデフォルトでオブジェクトの中間コピーを作成する必要があります。つまり、StorageGRIDでの取り込み動作にはDual Commit オプションを使用する必要があります。

同期配置とILMルールの詳細については、を参照してください ["取り込みのデータ保護オプション"](#)。

- Content-Type
- ETag
- X-Object-Meta-<name\>（オブジェクト関連のメタデータ）

ILMルールの参照時間として* User defined creation time *オプションを使用する場合は、というユーザ定義のヘッダーに値を格納する必要があります X-Object-Meta-Creation-Time。例：

```
X-Object-Meta-Creation-Time: 1443399726
```

このフィールドの値は、1970年1月1日からの秒数となります。

- X-Storage-Class: reduced_redundancy

このヘッダーは、取り込まれたオブジェクトに一致する ILM ルールで取り込み動作に Dual Commit または Balanced が指定されている場合に StorageGRID で作成されるオブジェクトコピーの数に影響しません。

- * Dual commit * : ILM ルールの取り込み動作が Dual commit オプションに指定されている場合は、オブジェクトの取り込み時に StorageGRID が中間コピーを 1 つ作成します (シングルコミット)。
 - * Balanced * : ILMルールでBalancedオプションが指定されている場合、StorageGRID は、ルールで指定されたすべてのコピーをただちに作成できない場合にのみ中間コピーを1つ作成します。StorageGRID で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 ヘッダーは、オブジェクトに一致する ILM ルールで単一のレプリケートコピーが作成される場合に最も適しています。この場合は、を使用します reduced_redundancy 取り込み処理のたびに追加のオブジェクトコピーを不要に作成および削除する必要がなくなります。

を使用する reduced_redundancy 取り込み中にオブジェクトデータが失われるリスクが高まるため、他の状況ではヘッダーを使用することは推奨されません。たとえば、ILM 評価の前にコピーが 1 つだけ格納されていたストレージノードに障害が発生すると、データが失われる可能性があります。



レプリケートコピーを一定期間に 1 つだけ作成すると、データが永続的に失われるリスクがあります。オブジェクトのレプリケートコピーが 1 つしかない場合、ストレージノードに障害が発生したり、重大なエラーが発生すると、そのオブジェクトは失われます。また、アップグレードなどのメンテナンス作業中は、オブジェクトへのアクセスが一時的に失われます。

を指定することに注意してください reduced_redundancy オブジェクトの初回取り込み時に作成されるコピー数にのみ影響します。オブジェクトがアクティブな ILM ポリシーで評価される際に作成されるオブジェクトのコピー数には影響せず、StorageGRID システムでデータが格納されるときに冗長性レベルが低下することはありません。

実行が成功すると、「HTTP/1.1 201 Created」の応答とともに次のヘッダーが返されます。

- Content-Length
- Content-Type
- Date
- ETag
- Last-Modified
- X-Trans-Id

OPTIONS 要求

OPTIONS 要求は、個々の Swift サービスが使用可能かどうかを確認します。OPTIONS 要求は、URL で指定されたストレージノードまたはゲートウェイノードによって処理されます。

OPTIONS メソッド

たとえば、クライアントアプリケーションでは、Swift 認証クレデンシャルを入力することなく、ストレージノード上の Swift ポートに OPTIONS 要求を問題 で送信して、ストレージノードが使用可能かどうかを判別で

きます。この要求は、監視に使用できるほか、外部のロードバランサがストレージノードの停止を特定する目的でも使用できます。

情報（info） URL またはストレージ（storage） URL と併用する場合、OPTIONS メソッドは、HEAD、GET、OPTIONS、PUT など、指定された URL でサポートされる動詞のリストを返します。OPTIONS メソッドは認証URLでは使用できません。

次の要求パラメータが必要です。

- Account

次の要求パラメータはオプションです。

- Container
- Object

実行が成功すると、「HTTP/1.1 204 No Content」の応答とともに次のヘッダーが返されます。ストレージ URL への OPTIONS 要求には、ターゲットが存在する必要はありません。

- Allow（HEAD、GET、OPTIONSなど、指定されたURLでサポートされる動詞のリスト）およびPUT
- Content-Length
- Content-Type
- Date
- X-Trans-Id

関連情報

["サポートされている Swift API エンドポイント"](#)

Swift API 処理に対するエラー応答

エラー応答について理解しておくこと、処理をトラブルシューティングする際に役立ちます。

処理中にエラーが発生した場合に返される HTTP ステータスコードを次に示します。

Swift エラーの名前	HTTP ステータス
AccountNameTooLong、ContainerNameTooLong、HeaderTooBig、InvalidContainerName、InvalidRequest、InvalidURI、MetadataNameTooLong、MetadataValueTooBig、MissingSecurityHeader、ObjectNameTooLong、TooManyContainers、TooManyMetadataItems、TotalMetadataTooLarge	400 不正な要求です
アクセスが拒否されました	403 禁止

Swift エラーの名前	HTTP ステータス
ContainerNotEmpty、ContainerAlreadyExists です	409 競合
内部エラー	500 Internal Server Error (内部サーバエラー)
InvalidRange : 無効な範囲	416 リクエストされた範囲が適合しません
MethodNotAllowed のように入力します	405 メソッドは許可されていません
MissingContentLength (MissingContentLength)	411 長さが必要です
NOTFOUND	404 が見つかりません
実装なし	501 は実装されていません
PreconditionalFailed	412 事前条件が失敗しました
resourceNotFound です	404 が見つかりません
権限がありません	401 認証なし
UnprocessableEntity の場合	422 加工不能エンティティ

StorageGRID の Swift REST API 処理

StorageGRID システム固有の処理が Swift REST API に追加されています。

GET コンテナセイコウセイヨウキユウ

"**整合性制御**" オブジェクトの可用性と、異なるストレージノードおよびサイト間でのオブジェクトの整合性のバランスを確保します。GET コンテナ整合性要求では、特定のコンテナに適用されている整合性レベルを確認できます。

リクエスト

要求の HTTP ヘッダー	説明
X-Auth-Tokenの略	要求に使用するアカウントの Swift 認証トークンを指定します。
x-ntap-sg-consistencyの略	要求のタイプを指定します <code>true</code> = GETコンテナconsistency、および <code>false</code> =コンテナを取得します。
ホスト	要求の転送先のホスト名。

要求例

```
GET /v1/28544923908243208806/Swift container
X-Auth-Token: SGRD_3a877009a2d24cb1801587bfa9050f29
x-ntap-sg-consistency: true
Host: test.com
```

応答

応答の HTTP ヘッダー	説明
日付	応答の日時。
接続	サーバへの接続が開いているかどうか。
X-Trans-ID	要求の一意のトランザクション ID。
Content-Lengthの略	応答の本文の長さ。
x-ntap-sg-consistencyの略	<p>コンテナに適用されている整合性制御レベルです。次の値がサポートされています。</p> <ul style="list-style-type: none">• all * :すべてのノードが即座にデータを受け取り、受け取れない場合は要求が失敗します。• strong-global * :すべてのサイトにおけるすべてのクライアント要求について、リードアフターライト整合性が保証されます。• strong-site * : 1つのサイトにおけるすべてのクライアント要求について、リードアフターライト整合性が保証されます。• read-after-new-write * : (デフォルト) 新規オブジェクトにはリードアフターライト整合性が提供され、オブジェクトの更新には結果整合性が提供されます。高可用性が確保され、データ保護が保証されます。ほとんどの場合に推奨されます。• available * : 新しいオブジェクトとオブジェクトの更新の両方について、結果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ます (読み取り頻度の低いログ値を含むバケットや、存在しないキーに対するHEAD処理やGET処理など)。S3 FabricPool バケットではサポートされません。

応答例


```

HTTP/1.1 204 No Content
Date: Sat, 29 Nov 2015 01:02:18 GMT
Connection: CLOSE
X-Trans-Id: 1936575373
Content-Length: 0
x-ntap-sg-consistency: strong-site

```

PUT コンテナセイコウセイヨウキユウ

PUT コンテナ整合性要求では、コンテナに対して実行される処理に適用する整合性レベルを指定できます。デフォルトでは '新しいコンテナは' リードアフター・ア・ニュー・ライトの整合性レベルを使用して作成されます

リクエスト

要求の HTTP ヘッダー	説明
X-Auth-Tokenの略	要求に使用するアカウントの Swift 認証トークンです。
x-ntap-sg-consistencyの略	<p>コンテナに対する処理に適用される整合性制御レベルです。次の値がサポートされています。</p> <ul style="list-style-type: none"> • all * : すべてのノードが即座にデータを受け取り、受け取れない場合は要求が失敗します。 • strong-global * : すべてのサイトにおけるすべてのクライアント要求について、リードアフターライト整合性が保証されます。 • strong-site * : 1つのサイトにおけるすべてのクライアント要求について、リードアフターライト整合性が保証されます。 • read-after-new-write * : (デフォルト) 新規オブジェクトにはリードアフターライト整合性が提供され、オブジェクトの更新には結果整合性が提供されます。高可用性が確保され、データ保護が保証されます。ほとんどの場合に推奨されます。 • available * : 新しいオブジェクトとオブジェクトの更新の両方について、結果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ます（読み取り頻度の低いログ値を含むバケットや、存在しないキーに対するHEAD処理やGET処理など）。S3 FabricPool バケットではサポートされません。
Host	要求の転送先のホスト名。

整合性制御と ILM ルールの相互作用によるデータ保護への影響

あなたの選択の両方 **"整合性制御"** オブジェクトの保護方法にはILMルールが影響します。これらの設定は対話的に操作できます。

たとえば、オブジェクトの格納時に使用される整合性制御は、でのオブジェクトメタデータの初期配置に影響

します "取り込み動作" ILMルールに対して選択すると、オブジェクトコピーの初期配置に影響します。StorageGRID では、クライアント要求に対応するためにオブジェクトのメタデータとそのデータの両方にアクセスするため、整合性レベルと取り込み動作に一致する保護レベルを選択することで、より適切な初期データ保護と予測可能なシステム応答を実現できます。

整合性制御と ILM ルールの連動の例

次の ILM ルールと次の整合性レベル設定の 2 サイトグリッドがあるとします。

- * ILM ルール * : ローカルサイトとリモートサイトに 1 つずつ、2 つのオブジェクトコピーを作成します。Strict 取り込み動作が選択されています。
- * 整合性レベル * : "Strong-GLOBAL" (オブジェクトメタデータはすべてのサイトにただちに分散されます)

クライアントがオブジェクトをグリッドに格納すると、StorageGRID は両方のオブジェクトをコピーし、両方のサイトにメタデータを分散してからクライアントに成功を返します。

オブジェクトは、取り込みが成功したことを示すメッセージが表示された時点で損失から完全に保護されます。たとえば、取り込み直後にローカルサイトが失われた場合、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っています。オブジェクトを完全に読み出し可能にしている。

代わりに同じ ILM ルールと「strong-site」整合性レベルを使用する場合は、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あとで、オブジェクトメタデータがそこに分散される前に、クライアントに成功メッセージが送信され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が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ると、オブジェクトメタデータが失われます。オブジェクトを取得できません。

整合性レベルと ILM ルールの間関係は複雑になる可能性があります。サポートが必要な場合は、ネットアップにお問い合わせください。

要求例

```
PUT /v1/28544923908243208806/_Swift_container_  
X-Auth-Token: SGRD_3a877009a2d24cb1801587bfa9050f29  
x-ntap-sg-consistency: strong-site  
Host: test.com
```

応答

応答の HTTP ヘッダー	説明
Date	応答の日時。
Connection	サーバへの接続が開いているかどうか。
X-Trans-Id	要求の一意のトランザクション ID。
Content-Length	応答の本文の長さ。

```
HTTP/1.1 204 No Content
Date: Sat, 29 Nov 2015 01:02:18 GMT
Connection: CLOSE
X-Trans-Id: 1936575373
Content-Length: 0
```

REST API のセキュリティを設定する

REST API のセキュリティの実装を確認し、システムの保護方法について理解しておく必要があります。

StorageGRID がREST APIのセキュリティを提供する仕組み

StorageGRID システムで REST API のセキュリティ、認証、および許可がどのように実装されるかを理解しておく必要があります。

StorageGRID では、次のセキュリティ対策が使用されます。

- ロードバランサエンドポイントで HTTPS が設定されている場合は、ロードバランササービスとのクライアント通信に HTTPS が使用されます。

いつでも ["ロードバランサエンドポイントを設定します"](#) HTTP を有効にすることもできます。たとえば、非本番環境でのテストなどに HTTP を使用できます。

- StorageGRID は、ストレージノードとのクライアント通信にデフォルトで HTTPS を使用します。

必要に応じて、["これらの接続に対して HTTP を有効にします"](#)。たとえば、非本番環境でのテストなどに HTTP を使用できます。

- StorageGRID とクライアント間の通信は、TLS を使用して暗号化されます。
- ロードバランササービスとグリッド内のストレージノードの間の通信は、ロードバランサエンドポイントが HTTP と HTTPS どちらの接続を受け入れるように設定されているかに関係なく暗号化されます。
- REST API 処理を実行するには、クライアントが StorageGRID に HTTP 認証ヘッダーを提供する必要があります。

セキュリティ証明書とクライアントアプリケーション

クライアントは、ゲートウェイノードまたは管理ノード上のロードバランササービスに、ストレージノードに直接接続できます。

いずれの場合も、クライアントアプリケーションは、グリッド管理者がアップロードしたカスタムサーバ証明書または StorageGRID システムが生成した証明書を使用して、TLS 接続を確立できます。

- ロードバランササービスに接続する場合、クライアントアプリケーションは、接続に使用するロードバランサエンドポイント用に設定された証明書を使用します。各エンドポイントには独自の証明書があり、グリッド管理者がアップロードしたカスタムサーバ証明書か、グリッド管理者がエンドポイントの設定時に

StorageGRID で生成した証明書のいずれかです。

- クライアントアプリケーションは、ストレージノードに直接接続する場合、StorageGRID システムのインストール時にストレージノード用に生成されたシステム生成のサーバ証明書（システム認証局によって署名されたもの）を使用します。または、グリッド管理者がグリッド用に提供した単一のカスタムサーバ証明書。

TLS 接続の確立に使用する証明書に署名した認証局を信頼するよう、クライアントを設定する必要があります。

を参照してください ["ロードバランサエンドポイントを設定しています"](#) および ["単一のカスタムサーバ証明書を追加しています"](#) ストレージノードへの直接TLS接続の場合。

まとめ

次の表に、S3 および Swift の REST API におけるセキュリティの問題に対する実装を示します。

Security 問題 の略	REST API の実装
接続のセキュリティ	TLS
サーバ認証	システム CA によって署名された X.509 サーバ証明書、または管理者から提供されたカスタムサーバ証明書
クライアント認証	<ul style="list-style-type: none">• S3 : S3 アカウント（アクセスキー ID とシークレットアクセスキー）• Swift : Swift アカウント（ユーザ名とパスワード）
クライアント許可	<ul style="list-style-type: none">• S3 : バケットの所有権と適用可能なすべてのアクセス制御ポリシー• Swift : 管理者ロールのアクセス

TLS ライブラリのハッシュアルゴリズムと暗号化アルゴリズムがサポートされます

StorageGRID システムでは、クライアントアプリケーションが Transport Layer Security（TLS）セッションを確立する際に使用できる暗号スイートに制限があります。暗号を設定するには、[\[設定\]>\[セキュリティ設定\]](#)に移動し、[TLSおよびSSHポリシー*](#)を選択します。

サポートされる **TLS** のバージョン

StorageGRID では、TLS 1.2 と TLS 1.3 がサポートされています。



SSLv3 と TLS 1.1（またはそれ以前のバージョン）はサポートされなくなりました。

関連情報

["テナントアカウントと接続を設定する"](#)

監視と監査の処理

グリッド全体または特定のノードのトランザクションの傾向を確認することで、クライアント処理のワークロードと効率を監視できます。監査メッセージを使用して、クライアント処理とトランザクションを監視できます。

オブジェクトの取り込み速度と読み出し速度を監視する

オブジェクトの取り込み速度と読み出し速度、およびオブジェクト数、クエリ、検証関連の指標を監視できます。StorageGRID システムのオブジェクトに対してクライアントアプリケーションが試みた読み取り、書き込み、変更の各処理について、成功した回数と失敗した回数を表示できます。

手順

1. を使用して Grid Manager にサインインします ["サポートされている Web ブラウザ"](#)。
2. ダッシュボードで、[パフォーマンス]>* S3処理]または[パフォーマンス]> Swift処理*を選択します。

このセクションには、StorageGRID システムによって実行されたクライアント処理の回数に関する概要が表示されます。プロトコル速度は過去 2 分間の平均値です。

3. [* nodes (ノード)]を選択します
4. ノードのホームページ (導入レベル) で、* ロードバランサ * タブをクリックします。

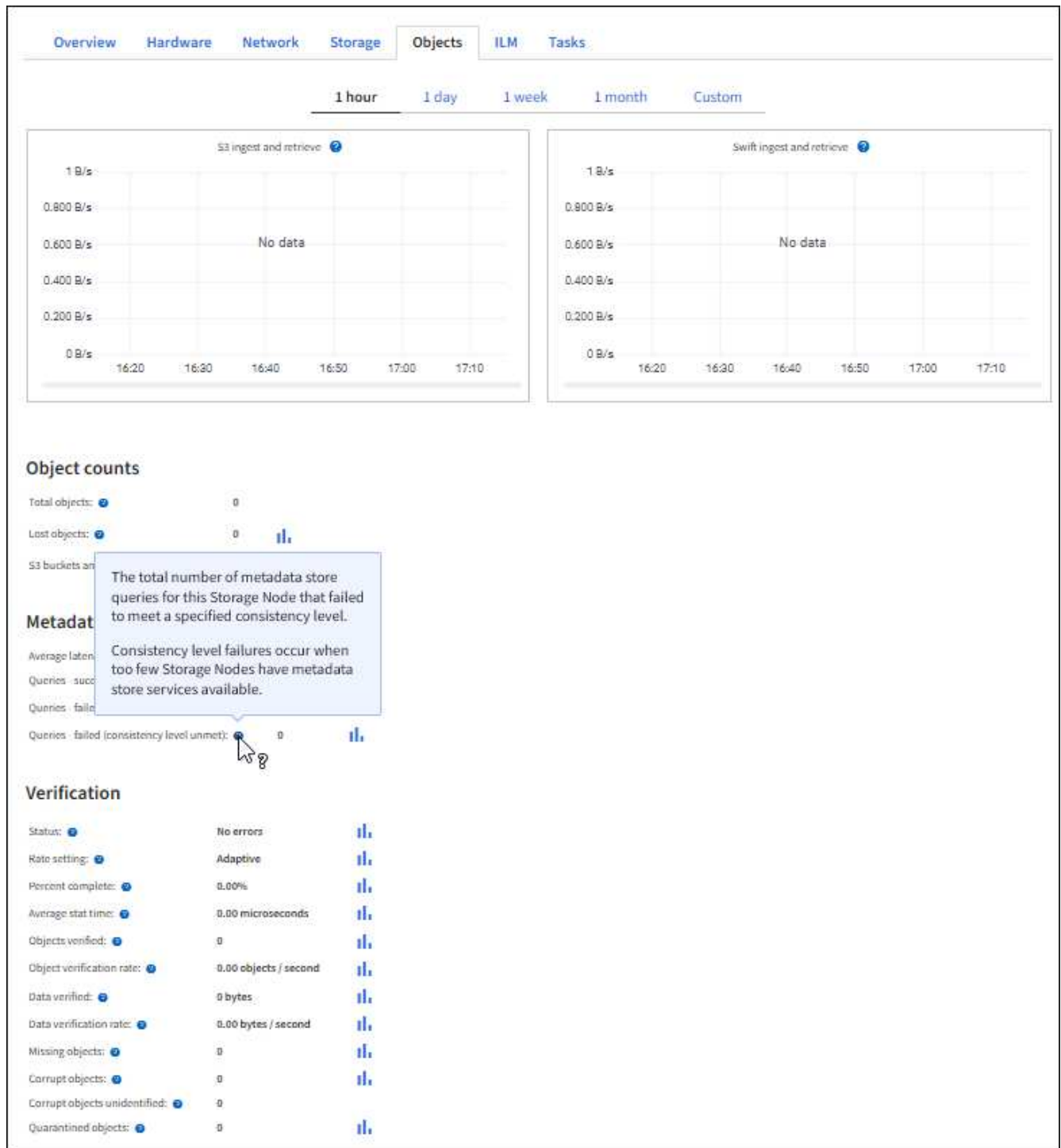
このグラフには、グリッド内でロードバランサエンドポイントに送信されるすべてのクライアントトラフィックの傾向が表示されます。時間、日、週、月、年単位の間隔を選択できます。または、カスタムの間隔を適用することもできます。

5. ノードのホームページ (導入レベル) で、* Objects * タブをクリックします。

グラフには、StorageGRID システム全体の取り込み速度と読み出し速度が、1 秒あたりのバイト数と合計バイト数で表示されます。時間、日、週、月、年単位の間隔を選択できます。または、カスタムの間隔を適用することもできます。

6. 特定のストレージノードに関する情報を表示するには、左側のリストからノードを選択し、* Objects * タブをクリックします。

グラフには、このストレージノードのオブジェクトの取り込み速度と読み出し速度が表示されます。このタブには、オブジェクト数、クエリ、検証関連の指標も表示されます。ラベルをクリックすると、これらの指標の定義を確認できます。



7. さらに詳細な情報が必要な場合は、次の手順に従います

- サポート * > * ツール * > * グリッドトポロジ * を選択します。
- [_site * >] > [Overview] > [Main*] を選択します。

API Operations セクションには、グリッド全体の概要情報が表示されます。

- 「*_ストレージノード_* > * LDR * > *_クライアントアプリケーション_* > * 概要 * > * Main *」を選択します

Operations セクションには、選択したストレージノードに関する概要情報が表示されます。

監査ログにアクセスして確認する

監査メッセージは StorageGRID サービスによって生成され、テキスト形式のログファイルに保存されます。監査ログの API 固有の監査メッセージにより、セキュリティ、運用、およびパフォーマンスについて、システムの健全性の評価に役立つ重要な監視データが提供されます。

作業を開始する前に

- 特定のアクセス権限が必要です。
- を用意しておく必要があります Passwords.txt ファイル。
- 管理ノードの IP アドレスを確認しておく必要があります。

このタスクについて

。"アクティブな監査ログファイル" という名前が付けられます `audit.log` をクリックし、を管理ノードに格納します。

1 日に 1 回、アクティブな audit.log ファイルが保存され、新しい audit.log ファイルが開始されます。保存されたファイルの名前は、保存された日時をの形式で示しています yyyy-mm-dd.txt。

1 日後、保存されたファイルは圧縮され、という形式で名前が変更されます `yyyy-mm-dd.txt.gz` 元の日付を保持します。

次の例は、アクティブな audit.log ファイル、前日のファイル (2018-04-15.txt)、および前日の圧縮されたファイルを示しています (2018-04-14.txt.gz)。

```
audit.log
2018-04-15.txt
2018-04-14.txt.gz
```

手順

1. 管理ノードにログインします。
 - a. 次のコマンドを入力します。 `ssh admin@primary_Admin_Node_IP`
 - b. に記載されているパスワードを入力します Passwords.txt ファイル。
 - c. 次のコマンドを入力して root に切り替えます。 `su -`
 - d. に記載されているパスワードを入力します Passwords.txt ファイル。

root としてログインすると、プロンプトがから変わります \$ 終了: #。

2. 監査ログファイルが保存されているディレクトリに移動します。 `cd /var/local/audit/export`
3. 必要に応じて、現在の監査ログファイルまたは保存された監査ログファイルを表示します。

監査ログで追跡される **Swift** 処理

ストレージで成功した DELETE、GET、HEAD、POST、PUT の各処理は、で追跡されます "StorageGRID 監査ログ"。障害はログに記録されず、info、auth、options 要求も記録されません。

次のSwift処理に関する情報が追跡されます。

アカウントの処理

- "GET アカウント"
- "HEAD アカウント"

コンテナの処理

- "コンテナを削除します"
- "GET コンテナ"
- "HEAD コンテナ"
- "PUT コンテナ"

オブジェクトの処理

- "オブジェクトを削除します"
- "GET オブジェクト"
- "HEAD オブジェクト"
- "PUT オブジェクト"

著作権に関する情報

Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S.このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

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

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

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

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

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

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

商標に関する情報

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