



グリッドマネージャーを使い始める StorageGRID software

NetApp
December 03, 2025

目次

グリッドマネージャーを使い始める	1
Webブラウザの要件	1
グリッドマネージャーにSign in	1
最初の管理ノードでグリッド マネージャーにSign in	2
別の管理ノードにサインインする	6
グリッドマネージャーからサインアウトする	7
パスワードを変更する	7
StorageGRIDライセンス情報を表示する	8
StorageGRIDライセンス情報を更新する	9
APIを使用する	10
グリッド管理APIを使用する	10
グリッド管理API操作	13
グリッド管理 API のバージョン管理	14
クロスサイトリクエストフォージェリ (CSRF) から保護する	16
シングルサインオンが有効になっている場合はAPIを使用する	17
APIを使用して機能を無効にする	31

グリッドマネージャーを使い始める

Webブラウザの要件

サポートされている Web ブラウザを使用する必要があります。

Webブラウザ	サポートされる最小バージョン
Google Chrome	119
Microsoft Edge	119
Mozilla Firefox	119

ブラウザウィンドウを推奨幅に設定する必要があります。

ブラウザの幅	ピクセル
最小	1024
最適	1280

グリッドマネージャーにSign in

Grid Manager サインイン ページにアクセスするには、サポートされている Web ブラウザのアドレス バーに管理ノードの完全修飾ドメイン名 (FQDN) または IP アドレスを入力します。

各StorageGRIDシステムには、1つのプライマリ管理ノードと任意の数の非プライマリ管理ノードが含まれます。任意の管理ノードの Grid Manager にサインインして、StorageGRIDシステムを管理できます。ただし、一部のメンテナンス手順はプライマリ管理ノードからのみ実行できます。

HAグループに接続する

管理ノードが高可用性 (HA) グループに含まれている場合は、HA グループの仮想 IP アドレスまたは仮想 IP アドレスにマップされる完全修飾ドメイン名を使用して接続します。プライマリ管理ノードをグループのプライマリ インターフェイスとして選択する必要があります。これにより、プライマリ管理ノードが使用できない場合を除き、グリッド マネージャにアクセスするときにプライマリ管理ノードからアクセスできるようになります。見る["高可用性グループの管理"](#)。

SSO を使用

サインインの手順は、以下の場合、若干異なります。["シングルサインオン \(SSO\) が設定されています"](#)。

最初の管理ノードでグリッド マネージャーにSign in

開始する前に

- ログイン資格情報をお持ちです。
- 使用しています"[サポートされているウェブブラウザ](#)".
- Web ブラウザで Cookie が有効になっています。
- 少なくとも 1 つの権限を持つユーザー グループに属しています。
- グリッド マネージャーの URL は次のとおりです。

```
https://FQDN_or_Admin_Node_IP/
```

完全修飾ドメイン名、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 アドレスを使用できます。

HTTPS のデフォルト ポート (443) 以外のポートで Grid Manager にアクセスするには、URL にポート番号を含めます。

```
https://FQDN_or_Admin_Node_IP:port/
```



制限された Grid Manager ポートでは SSO は使用できません。ポート443を使用する必要があります。

手順

1. サポートされている Web ブラウザを起動します。
2. ブラウザのアドレスバーに、グリッド マネージャーの 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ます。見る"[セキュリティ証明書を管理する](#)"。
4. グリッド マネージャーにSign in。

表示されるサインイン画面は、StorageGRIDに対してシングル 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を使用していない

- a. グリッド マネージャーのユーザー名とパスワードを入力します。
- b. *サインイン*を選択します。



NetApp StorageGRID®

Grid Manager

Username

Password

Sign in

[Tenant sign in](#) | [NetApp support](#) | [NetApp.com](#)

SSOの使用

- StorageGRIDが SSO を使用しており、このブラウザで URL にアクセスするのは初めての場合:
 - i. *Sign in*を選択します。アカウント フィールドには 0 を残しておけます。

NetApp StorageGRID[®]

Sign in

Account

Sign in

[NetApp support](#) | [NetApp.com](#)

- ii. 組織の SSO サインイン ページで標準の SSO 資格情報を入力します。例えば：

Sign in with your organizational account

Sign in

- ° StorageGRIDが SSO を使用しており、以前に Grid Manager またはテナント アカウントにアクセスしたことがある場合：
 - i. **0** (グリッド マネージャーのアカウント ID) を入力するか、最近のアカウントのリストにグリッド マネージャー が表示されている場合はそれを選択します。

NetApp StorageGRID[®]

Sign in

Recent

Grid Manager ▼

Account

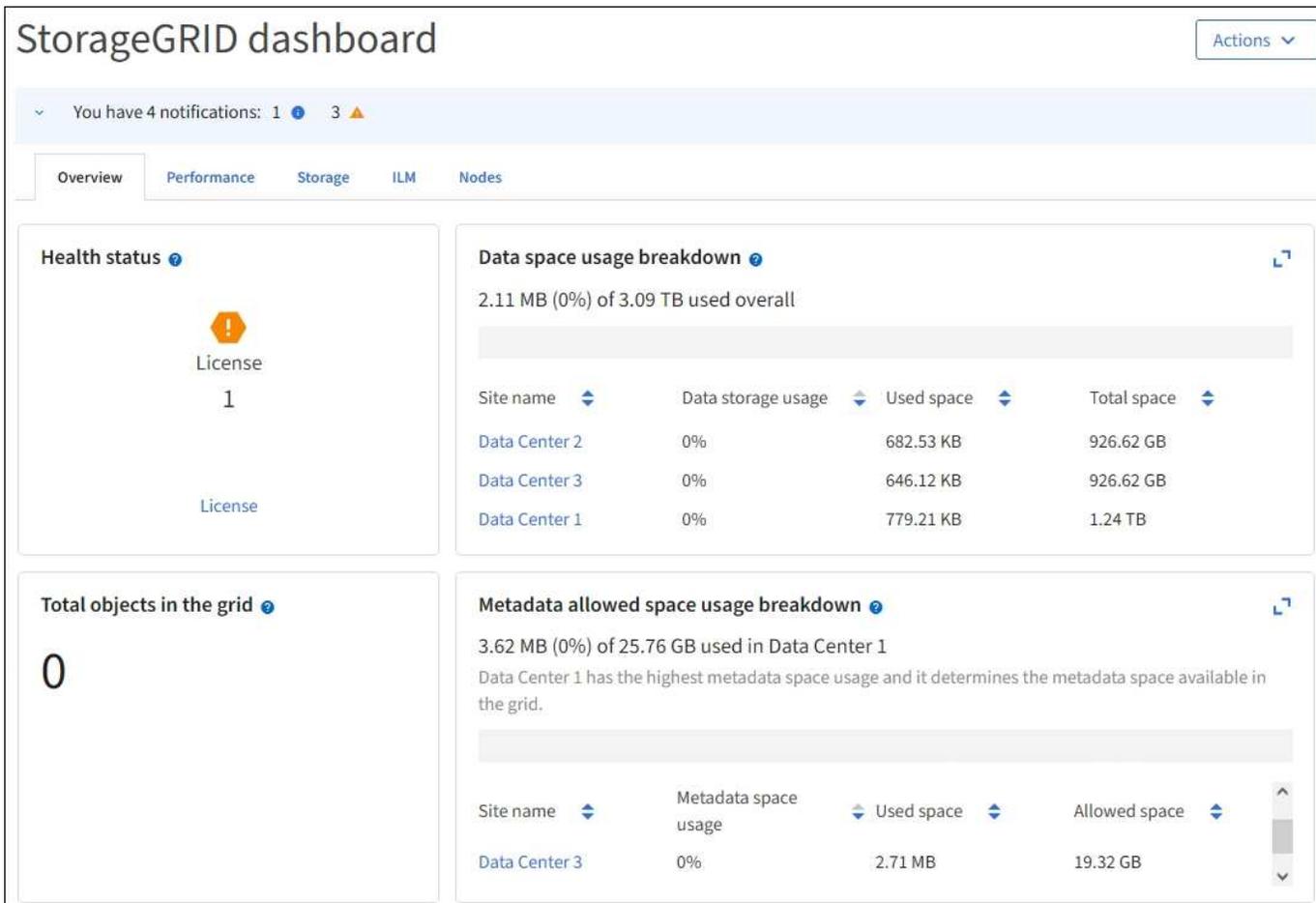
0

Sign in

[NetApp support](#) | [NetApp.com](#)

- ii. *Sign in*を選択します。
- iii. 組織の SSO サインイン ページで標準の SSO 資格情報を使用してSign in。

サインインすると、ダッシュボードを含むグリッド マネージャーのホームページが表示されます。提供される情報については、"[ダッシュボードの表示と管理](#)"。



別の管理ノードにサインインする

別の管理ノードにサインインするには、次の手順に従います。

SSOを使用していない

手順

1. ブラウザのアドレスバーに、他の管理ノードの完全修飾ドメイン名または IP アドレスを入力します。必要に応じてポート番号を含めます。
2. グリッド マネージャーのユーザー名とパスワードを入力します。
3. *サインイン*を選択します。

SSOの使用

StorageGRIDが SSO を使用しており、1つの管理ノードにサインインしている場合は、再度サインインしなくても他の管理ノードにアクセスできます。

手順

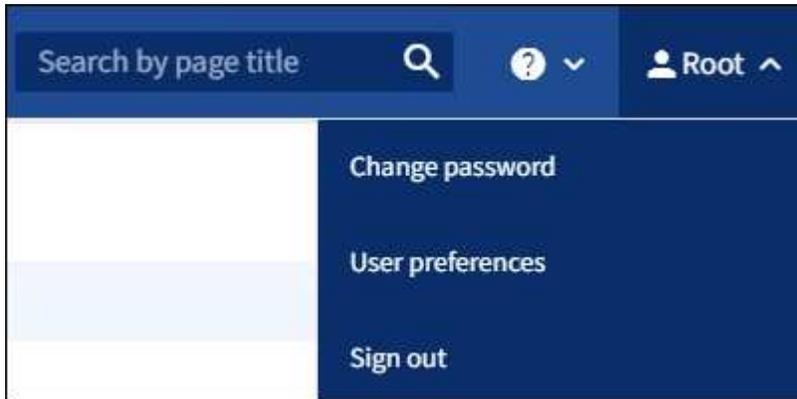
1. ブラウザのアドレスバーに、他の管理ノードの完全修飾ドメイン名または IP アドレスを入力します。
2. SSO セッションの有効期限が切れている場合は、資格情報を再度入力してください。

グリッドマネージャーからサインアウトする

Grid Manager での作業が完了したら、権限のないユーザーがStorageGRIDシステムにアクセスできないようにするためにサインアウトする必要があります。ブラウザの Cookie 設定によっては、ブラウザを閉じて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. 右上隅にあるユーザー名を選択します。



2. *サインアウト*を選択します。

オプション	説明
SSOは使用されていません	管理ノードからサインアウトしました。 Grid Manager のサインイン ページが表示されます。 注意: 複数の管理ノードにサインインしている場合は、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	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しました。StorageGRIDサインイン ページが表示されます。 Grid Manager は最近のアカウント ドロップダウンにデフォルトとしてリストされ、アカウント ID フィールドには 0 が表示されます。 注意: SSOが有効になっていて、テナントマネージャにもサインインしている場合は、" テナントアカウントからサインアウトする " に" SSOからサインアウト "。

パスワードを変更する

Grid Manager のローカル ユーザーの場合は、自分のパスワードを変更できます。

開始する前に

グリッドマネージャにサインインするには、"[サポートされているウェブブラウザ](#)"。

タスク概要

フェデレーション ユーザーとしてStorageGRIDにサインインしている場合、またはシングル サインオン (SSO) が有効になっている場合は、Grid Manager でパスワードを変更することはできません。代わりに、Active Directory や OpenLDAP などの外部 ID ソースでパスワードを変更する必要があります。

手順

1. グリッド マネージャー ヘッダーから、**your name** > パスワードの変更 を選択します。
2. 現在のパスワードを入力してください。
3. 新しいパスワードを入力してください。

パスワードは 8 文字以上 32 文字以下でなければなりません。パスワードでは大文字と小文字が区別されます。

4. 新しいパスワードをもう一度入力します。
5. *保存*を選択します。

StorageGRID ライセンス情報を表示する

必要に応じて、グリッドの最大ストレージ容量など、StorageGRIDシステムのライセンス情報を表示できます。

開始する前に

グリッドマネージャにサインインするには、["サポートされているウェブブラウザ"](#)。

タスク概要

このStorageGRIDシステムのソフトウェア ライセンスに問題がある場合は、ダッシュボードのヘルス ステータス カードにライセンス ステータス アイコンと ライセンス リンクが表示されます。数字はライセンス関連の問題の数を示します。



手順

1. 次のいずれかの方法でライセンス ページにアクセスします。
 - メンテナンス > システム > *ライセンス*を選択します。

- ダッシュボードのヘルス ステータス カードから、ライセンス ステータス アイコンまたは ライセンス リンクを選択します。

このリンクは、ライセンスに問題がある場合にのみ表示されます。

2. 現在のライセンスの読み取り専用の詳細を表示します。

- StorageGRIDシステム ID。これは、このStorageGRIDインストールの一意的識別番号です。
- ライセンスシリアル番号
- ライセンスの種類（永久*または*サブスクリプション）
- グリッドのライセンスされたストレージ容量
- サポートされているストレージ容量
- ライセンスの終了日。永久ライセンスの場合は*N/A*が表示されます。
- サポート終了日

この日付は現在のライセンス ファイルから読み取られますが、ライセンス ファイルの取得後にサポート サービス契約を延長または更新した場合は、日付が古くなっている可能性があります。この値を更新するには、"[StorageGRIDライセンス情報を更新する](#)"。Active IQを使用して実際の契約終了日を確認することもできます。

- ライセンステキストファイルの内容

StorageGRIDライセンス情報を更新する

ライセンスの条件が変更されるたびに、StorageGRIDシステムのライセンス情報を更新する必要があります。たとえば、グリッドに追加のストレージ容量を購入する場合は、ライセンス情報を更新する必要があります。

開始する前に

- StorageGRIDシステムに適用する新しいライセンス ファイルがあります。
- あなたが持っている"[特定のアクセス権限](#)"。
- プロビジョニング パスフレーズを持っています。

手順

1. メンテナンス > システム > *ライセンス*を選択します。
2. ライセンスの更新セクションで、[参照] を選択します。
3. 新しいライセンスファイルを見つけて選択します(.txt)。

新しいライセンス ファイルが検証され、表示されます。

4. プロビジョニング パスフレーズを入力します。
5. *保存*を選択します。

APIを使用する

グリッド管理APIを使用する

Grid Manager ユーザー インターフェイスの代わりに Grid 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。

トップレベルのリソース

グリッド管理 API は、次の最上位リソースを提供します。

- /grid: アクセスは Grid Manager ユーザーに制限され、構成されたグループ権限に基づいて行われます。
- /org: アクセスは、テナント アカウントのローカルまたはフェデレーション LDAP グループに属するユーザーに制限されます。詳細については、"[テナントアカウントを使用する](#)"。
- /private: アクセスは Grid Manager ユーザーに制限され、構成されたグループ権限に基づいて行われます。プライベート API は予告なく変更される場合があります。StorageGRID プライベート エンドポイントは、リクエストの API バージョンも無視します。

APIリクエストを発行する

グリッド管理 API は、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、開発者と非開発者が API を使用して StorageGRID でリアルタイム操作を実行できる直感的なユーザー インターフェイスを提供します。

Swagger ユーザー インターフェイスは、各 API 操作の完全な詳細とドキュメントを提供します。

開始する前に

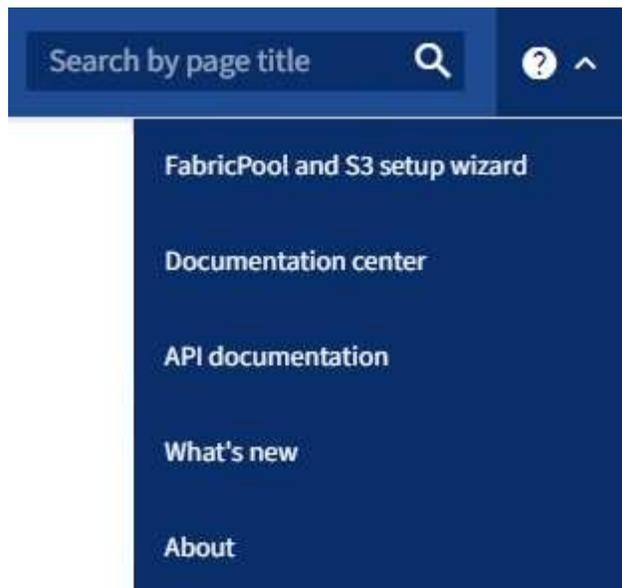
- グリッドマネージャにサインインするには、"[サポートされているウェブブラウザ](#)"。
- あなたが持っている "[特定のアクセス権限](#)"。



API ドキュメント Web ページを使用して実行するすべての API 操作はライブ操作です。誤って設定データやその他のデータを作成、更新、削除しないように注意してください。

手順

1. グリッド マネージャー ヘッダーからヘルプ アイコンを選択し、**API ドキュメント** を選択します。



2. プライベート API を使用して操作を実行するには、StorageGRID管理 API ページで **プライベート API** ドキュメントに移動を選択します。

プライベート API は予告なく変更される場合があります。StorageGRIDプライベート エンドポイントは、リクエストの API バージョンも無視します。

3. 希望する操作を選択します。

API 操作を展開すると、GET、PUT、UPDATE、DELETE などの利用可能な HTTP アクションが表示されます。

4. HTTP アクションを選択すると、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、リクエスト本文の例 (必要な場合)、および可能な応答などのリクエストの詳細が表示されます。

GET /grid/groups Lists Grid Administrator Groups

Parameters Try it out

Name	Description
type string (query)	filter by group type Available values : local, federated --
limit integer (query)	maximum number of results Default value : 25 25
marker string (query)	marker-style pagination offset (value is Group's URN) marker - marker-style pagination offset (value
includeMarker boolean (query)	if set, the marker element is also returned --
order string (query)	pagination order (desc requires marker) Available values : asc, desc --

Responses Response content type application/json

Code	Description
200	successfully retrieved Example Value Model

```
{
  "responseTime": "2021-03-29T14:22:19.673Z",
  "status": "success",
  "apiVersion": "3.3",
  "deprecated": false,
  "data": [
    {
      "displayName": "Developers",

```

- リクエストにグループ ID やユーザー ID などの追加の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、最初に別の API リクエストを発行する必要がある場合があります。
- サンプルのリクエスト本文を変更する必要があるかどうかを判断します。その場合は、「モデル」を選択して、各フィールドの要件を確認することができます。
- *試してみる*を選択します。
- 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ます。
- *実行*を選択します。
- 応答コードを確認して、リクエストが成功したかどうかを確認します。

グリッド管理API操作

グリッド管理 API は、利用可能な操作を次のセクションに分類します。



このリストには、パブリック API で利用可能な操作のみが含まれます。

- **アカウント**: 新しいアカウントの作成や特定のアカウントのストレージ使用量の取得など、ストレージ テナント アカウントを管理するための操作。
- **alert-history**: 解決されたアラートに対する操作。
- **alert-receivers**: アラート通知受信者 (電子メール) に対する操作。
- **alert-rules**: アラート ルールに対する操作。
- **alert-silences**: アラートサイレンスに関する操作。
- **アラート**: アラートに関する操作。
- **audit**: 監査構成を一覧表示および更新する操作。
- **auth**: ユーザーセッション認証を実行する操作。

グリッド管理 API は、ベアラー トークン認証スキームをサポートしています。サインインするには、認証リクエストのJSON本文にユーザー名とパスワードを入力します (つまり、POST /api/v3/authorize)。ユーザーが正常に認証されると、セキュリティ トークンが返されます。このトークンは、後続の API リクエストのヘッダー (「Authorization: Bearer token」) で提供する必要があります。トークンは16時間後に期限切れになります。



StorageGRIDシステムでシングル サインオンが有効になっている場合は、認証のために別の手順を実行する必要があります。「シングル サインオンが有効な場合の API への認証」を参照してください。

認証セキュリティの向上については、「クロスサイト リクエスト フォージェリの防止」を参照してください。

- **client-certificates**: 外部監視ツールを使用してStorageGRID に安全にアクセスできるようにクライアント証明書を構成する操作。
- **config**: Grid Management API の製品リリースとバージョンに関連する操作。製品リリース バージョンと、そのリリースでサポートされている Grid Management API のメジャー バージョンを一覧表示したり、API の非推奨バージョンを無効にしたりできます。
- **deactivated-features**: 非アクティブ化された可能性のある機能を表示する操作。
- **dns-servers**: 構成された外部 DNS サーバーを一覧表示および変更する操作。
- **drive-details**: 特定のストレージアプライアンス モデルのドライブに対する操作。
- **endpoint-domain-names**: S3 エンドポイントのドメイン名を一覧表示および変更する操作。
- **消去コーディング**: 消去コーディング プロファイルに対する操作。
- **展開**: 展開に関する操作 (プロシージャ レベル)。
- **expansion-nodes**: 拡張 (ノード レベル) に関する操作。
- **拡張サイト**: 拡張に関する操作 (サイトレベル)。

- **grid-networks**: グリッド ネットワーク リストを一覧表示および変更する操作。
- **grid-passwords**: グリッド パスワード管理の操作。
- **groups**: ローカル グリッド管理者グループを管理し、外部 LDAP サーバーからフェデレーション グリッド管理者グループを取得するための操作。
- **identity-source**: 外部 ID ソースを構成し、フェデレーション グループとユーザー情報を手動で同期する操作。
- **ilm**: 情報ライフサイクル管理 (ILM) に関する操作。
- **in-progress-procedures**: 現在進行中のメンテナンス手順を取得します。
- **license**: StorageGRIDライセンスを取得および更新する操作。
- **logs**: ログファイルを収集およびダウンロードするための操作。v
- **metrics**: 単一時点でのインスタント メトリック クエリや、一定範囲の時間にわたる範囲メトリック クエリなどのStorageGRIDメトリックに対する操作。グリッド管理 API は、バックエンド データ ソースとして Prometheus システム監視ツールを使用します。Prometheus クエリの構築については、Prometheus Web サイトを参照してください。



含まれる指標 *private* 名前に含まれる文字は内部使用のみを目的としています。これらのメトリックは、StorageGRIDリリース間で予告なく変更されることがあります。

- **node-details**: ノードの詳細に関する操作。
- **node-health**: ノードのヘルスステータスに関する操作。
- **node-storage-state**: ノードストレージステータスに関する操作。
- **ntp-servers**: 外部ネットワーク タイム プロトコル (NTP) サーバーを一覧表示または更新する操作。
- オブジェクト: オブジェクトおよびオブジェクト メタデータに対する操作。
- **recovery**: 回復手順のための操作。
- **recovery-package**: リカバリ パッケージをダウンロードする操作。
- **regions**: リージョンを表示および作成する操作。
- **s3-object-lock**: グローバル S3 オブジェクトロック設定に対する操作。
- **server-certificate**: Grid Manager サーバー証明書を表示および更新する操作。
- **snmp**: 現在の SNMP 構成に対する操作。
- **storage-watermarks**: ストレージ ノードのウォーターマーク。
- **traffic-classes**: トラフィック分類ポリシーの操作。
- **untrusted-client-network**: 信頼できないクライアント ネットワーク構成での操作。
- **users**: Grid Manager ユーザーを表示および管理する操作。

グリッド管理 API のバージョン管理

グリッド管理 API は、バージョン管理を使用して、中断のないアップグレードをサポートします。

たとえば、このリクエスト URL は API バージョン 4 を指定します。

https://hostname_or_ip_address/api/v4/authorize

古いバージョンと互換性のない変更が行われた場合には、API のメジャー バージョンが引き上げられます。古いバージョンと互換性のある変更が行われた場合に、API のマイナー バージョンが引き上げられます。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。

次の例は、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変更されるかを示しています。

APIの変更の種類	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
旧バージョンとは互換性がありません	2.1	3.0

StorageGRIDソフトウェアを初めてインストールすると、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すると、少なくとも 1 つのStorageGRID機能リリースについては引き続き古い API バージョンにアクセスできます。



サポートされるバージョンを設定できます。Swagger APIドキュメントの*config*セクションを参照してください。["グリッド管理API"](#)詳細についてはこちらをご覧ください。すべてのAPIクライアントを更新して新しいバージョンを使用するようにした後、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。

古くなったリクエストは、次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レスポンス本文に「deprecated」が含まれています: true
- 非推奨の警告が nms.log に追加されます。例えば：

```
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

使用 `GET /versions` サポートされている API メジャー バージョンのリストを返す API リクエスト。このリクエストは、Swagger API ドキュメントの **config** セクションにあります。

```
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**を指定する

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

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

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

クロスサイトリクエストフォージェリ (**CSRF**) 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) 攻撃から保護することができます。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。他の API クライアントはサインイン時にこの機能を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をトリガーできる攻撃者 (HTTP フォーム POST など) は、サインインしたユーザーの Cookie を使用して特定のリクエストを実行させる可能性があります。

StorageGRID は、CSRF トークンを使用して CSRF 攻撃から保護します。有効にすると、特定の Cookie の内容は、特定のヘッダーまたは特定の POST 本文パラメータのいずれか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、csrfToken`パラメータに `true` 認証中。デフォルトは `false`。

```
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"
```

真の場合、`GridCsrfToken`グリッドマネージャへのサインイン時にランダムな値でクッキーが設定され、`AccountCsrfToken`テナント マネージャーへのサインイン用に、Cookie にランダムな値が設定されます。

クッキーが存在する場合、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、次のいずれかが含まれている必要があります。

- その `X-Csrf-Token`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本文を受け入れるエンドポイントの場合: `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。

追加の例と詳細については、オンライン API ドキュメントを参照してください。



CSRF トークン クッキーが設定されているリクエストでは、CSRF 攻撃に対する追加の保護として、JSON リクエスト本文を期待するすべてのリクエストに「Content-Type: application/json」ヘッダーも適用されます。

シングルサインオンが有効になっている場合はAPIを使用する

シングル サインオンが有効になっている場合は **API** を使用する (**Active Directory**)

もしあなたが**シングル サインオン (SSO) を設定して有効にする**Active Directory を SSO プロバイダーとして使用する場合は、一連の API 要求を発行して、グリッド管理 API またはテナント管理 API に有効な認証トークンを取得する必要があります。

シングル サインオンが有効になっている場合は、**API** に**Sign in**。

これらの手順は、Active Directory を SSO ID プロバイダーとして使用している場合に適用されます。

開始する前に

- StorageGRIDユーザー グループに属するフェデレーション ユーザーの SSO ユーザー名とパスワードがわかっています。
- テナント管理 API にアクセスする場合は、テナント アカウント ID がわかっている必要があります。

タスク概要

認証トークンを取得するには、次のいずれかの例を使用できます。

- その `storagegrid-ssoauth.py` StorageGRIDインストール ファイル ディレクトリにある Python スクリプト (`./rpms` Red Hat Enterprise Linuxの場合、`./debs` UbuntuまたはDebianの場合、`./vsphere` VMware の場合)。

- curl リクエストのワークフローの例。

curl ワークフローは、実行速度が遅すぎるとタイムアウトする可能性があります。次のエラーが表示される場合があります: A valid SubjectConfirmation was not found on this Response。



サンプルの curl ワークフローでは、パスワードが他のユーザーから見られないように保護されません。

URL エンコードの問題がある場合は、次のエラーが表示されることがあります。Unsupported SAML version。

手順

1. 認証トークンを取得するには、次のいずれかの方法を選択します。
 - 使用 `storagegrid-ssoauth.py` Python スクリプト。ステップ 2 に進みます。
 - curl リクエストを使用します。ステップ 3 に進みます。
2. 使用したい場合は `storagegrid-ssoauth.py` スクリプトの場合は、スクリプトを Python インタープリターに渡してスクリプトを実行します。

プロンプトが表示されたら、次の引数の値を入力します。

- SSO 方式。ADFS または adfs を入力します。
- SSO ユーザ名
- StorageGRID がインストールされているドメイン
- StorageGRID のアドレス
- テナント管理 API にアクセスする場合のテナント アカウント ID。

```
python3 storagegrid-ssoauth.py
sso_method: adfs
saml_user: my-sso-username
saml_domain: my-domain
sg_address: storagegrid.example.com
tenant_account_id: 12345
Enter the user's SAML password:
*****
*****
StorageGRID Auth Token: 56eb07bf-21f6-40b7-afob-5c6cacfb25e7
```

StorageGRID 認証トークンが出力に提供されます。SSO が使用されていない場合に API を使用するのと同様に、他のリクエストにもトークンを使用できるようになりました。

3. curl リクエストを使用する場合は、次の手順に従います。
 - a. サインインに必要な変数を宣言します。

```
export SAMLUSER='my-sso-username'  
export SAMLPASSWORD='my-password'  
export SAMLDOMAIN='my-domain'  
export TENANTACCOUNTID='12345'  
export STORAGEGRID_ADDRESS='storagegrid.example.com'  
export AD_FS_ADDRESS='adfs.example.com'
```



グリッド管理APIにアクセスするには、0を次のように使用します。
TENANTACCOUNTID。

- b. 署名された認証URLを受け取るには、POSTリクエストを発行します。 /api/v3/authorize-saml、レスポンスから追加の JSON エンコーディングを削除します。

この例では、署名付き認証URLのPOSTリクエストを示しています。 TENANTACCOUNTID。結果は `python -m json.tool` JSON エンコーディングを削除します。

```
curl -X POST "https://$STORAGEGRID_ADDRESS/api/v3/authorize-saml" \  
  -H "accept: application/json" -H "Content-Type: application/json" \  
  --data "{\"accountId\": \"$TENANTACCOUNTID\"}" | python -m \  
  json.tool
```

この例の応答には、URL エンコードされた署名付き URL が含まれますが、追加の JSON エンコードレイヤーは含まれません。

```
{  
  "apiVersion": "3.0",  
  "data":  
  "https://adfs.example.com/adfs/ls/?SAMLRequest=fZHLbsIwEEV%2FJTuv7...  
  sSl%2BfQ33cvfwA%3D&RelayState=12345",  
  "responseTime": "2018-11-06T16:30:23.355Z",  
  "status": "success"  
}
```

- c. 保存する `SAMLRequest` 後続のコマンドで使用するために応答から取得します。

```
export SAMLREQUEST='fZHLbsIwEEV%2FJTuv7...sSl%2BfQ33cvfwA%3D'
```

- d. AD FS からクライアント要求 ID を含む完全な URL を取得します。

1つのオプションは、前の応答からの URL を使用してログイン フォームを要求することです。

```
curl "https://$AD_FS_ADDRESS/adfs/ls/?SAMLRequest=$SAMLREQUEST&RelayState=$TENANTACCOUNTID" | grep 'form method="post" id="loginForm"'
```

応答にはクライアント要求 ID が含まれます。

```
<form method="post" id="loginForm" autocomplete="off" novalidate="novalidate" onKeyPress="if (event && event.keyCode == 13) Login.submitLoginRequest();" action="/adfs/ls/?SAMLRequest=fZHRTomwFIZfhh...UJikvo77sXPw%3D%3D&RelayState=12345&client-request-id=00000000-0000-0000-ee02-0080000000de" >
```

e. 応答からクライアント要求 ID を保存します。

```
export SAMLREQUESTID='00000000-0000-0000-ee02-0080000000de'
```

f. 前の応答からのフォーム アクションに資格情報を送信します。

```
curl -X POST "https://$AD_FS_ADDRESS/adfs/ls/?SAMLRequest=$SAMLREQUEST&RelayState=$TENANTACCOUNTID&client-request-id=$SAMLREQUESTID" \ --data "UserName=$SAMLUSER@$SAMLDOMAIN&Password=$SAMPLPASSWORD&AuthMethod=FormsAuthentication" --include
```

AD FS は、ヘッダーに追加情報を含む 302 リダイレクトを返します。



SSO システムで多要素認証 (MFA) が有効になっている場合は、フォーム投稿に 2 番目のパスワードまたはその他の資格情報も含まれます。

```
HTTP/1.1 302 Found
Content-Length: 0
Content-Type: text/html; charset=utf-8
Location:
https://adfs.example.com/adfs/ls/?SAMLRequest=fZHRTomwFIZfhh...UJikvo77sXPw%3D%3D&RelayState=12345&client-request-id=00000000-0000-0000-ee02-0080000000de
Set-Cookie: MSISAuth=AAEAADAvsHpXk6ApV...pmP0aEiNtJvWY=; path=/adfs; HttpOnly; Secure
Date: Tue, 06 Nov 2018 16:55:05 GMT
```

g. 保存する `MSISAuth` 応答から Cookie を取得します。

ンを生成するためのリクエスト。

のために RelayState、テナント アカウント ID を使用するが、Grid Management API にサインインする場合は 0 を使用します。

```
curl -X POST "https://$STORAGEGRID_ADDRESS:443/api/saml-response" \  
  -H "accept: application/json" \  
  --data-urlencode "SAMLResponse=$SAMLResponse" \  
  --data-urlencode "RelayState=$TENANTACCOUNTID" \  
  | python -m json.tool
```

応答には認証トークンが含まれます。

```
{  
  "apiVersion": "3.0",  
  "data": "56eb07bf-21f6-40b7-af0b-5c6cacfb25e7",  
  "responseTime": "2018-11-07T21:32:53.486Z",  
  "status": "success"  
}
```

- a. レスポンス内の認証トークンを次のように保存します。MYTOKEN。

```
export MYTOKEN="56eb07bf-21f6-40b7-af0b-5c6cacfb25e7"
```

使用できるようになりました `MYTOKEN` その他のリクエストについては、SSO が使用されていない場合に API を使用する方法と同様です。

シングル サインオンが有効になっている場合は、**API** からサインアウトします。

シングル サインオン (SSO) が有効になっている場合は、グリッド管理 API またはテナント管理 API からサインアウトするための一連の API 要求を発行する必要があります。これらの手順は、Active Directory を SSO ID プロバイダーとして使用している場合に適用されます。

タスク概要

必要に応じて、組織のシングル ログアウト ページからログアウトすることで、StorageGRID API からサインアウトできます。または、有効なStorageGRIDベアラー トークンを必要とするStorageGRIDからシングル ログアウト (SLO) をトリガーすることもできます。

手順

1. 署名付きログアウト リクエストを生成するには、`cookie "sso=true"` を SLO API に渡します。

```
curl -k -X DELETE "https://$STORAGEGRID_ADDRESS/api/v3/authorize" \  
-H "accept: application/json" \  
-H "Authorization: Bearer $MYTOKEN" \  
--cookie "sso=true" \  
| python -m json.tool
```

ログアウト URL が返されます:

```
{  
  "apiVersion": "3.0",  
  "data":  
  "https://ads.example.com/ads/ls/?SAMLRequest=fZDNboMwEIRfhZ...HcQ%3D%3D",  
  "responseTime": "2018-11-20T22:20:30.839Z",  
  "status": "success"  
}
```

2. ログアウト URL を保存します。

```
export LOGOUT_REQUEST  
='https://ads.example.com/ads/ls/?SAMLRequest=fZDNboMwEIRfhZ...HcQ%3D%3D'
```

3. ログアウト URL にリクエストを送信して、SLO をトリガーし、StorageGRIDにリダイレクトします。

```
curl --include "$LOGOUT_REQUEST"
```

302 応答が返されます。リダイレクト場所は、API のみのログアウトには適用されません。

```
HTTP/1.1 302 Found  
Location: https://$STORAGEGRID_ADDRESS:443/api/saml-logout?SAMLResponse=fVLLasMwEPwVo7ss%...%23rsa-sha256  
Set-Cookie: MSISSignoutProtocol=U2FtbA==; expires=Tue, 20 Nov 2018 22:35:03 GMT; path=/ads; HttpOnly; Secure
```

4. StorageGRIDベアラー トークンを削除します。

StorageGRIDベアラー トークンの削除は、SSO がない場合と同じように機能します。`cookie "sso=true" が提供されない場合、ユーザーは SSO 状態に影響を与えずにStorageGRIDからログアウトされます。

```
curl -X DELETE "https://$STORAGEGRID_ADDRESS/api/v3/authorize" \  
-H "accept: application/json" \  
-H "Authorization: Bearer $MYTOKEN" \  
--include
```

あ `204 No Content`この応答は、ユーザーが現在サインアウトしていることを示します。

```
HTTP/1.1 204 No Content
```

シングルサインオンが有効になっている場合は **API** を使用する (**Azure**)

もしあなたが"[シングルサインオン \(SSO\) を設定して有効にする](#)"Azure を SSO プロバイダーとして使用する場合、2つのサンプルスクリプトを使用して、グリッド管理API またはテナント管理API に有効な認証トークンを取得できます。

Azure シングルサインオンが有効になっている場合は、**API** に**Sign in**。

これらの手順は、Azure を SSO ID プロバイダーとして使用している場合に適用されます。

開始する前に

- StorageGRIDユーザーグループに属するフェデレーションユーザーの SSO 電子メール アドレスとパスワードがわかっています。
- テナント管理 API にアクセスする場合は、テナント アカウント ID がわかっている必要があります。

タスク概要

認証トークンを取得するには、次のサンプルスクリプトを使用できます。

- その `storagegrid-ssoauth-azure.py` Pythonスクリプト
- その `storagegrid-ssoauth-azure.js` Node.jsスクリプト

両方のスクリプトはStorageGRIDインストールファイルディレクトリにあります。(./rpms Red Hat Enterprise Linuxの場合、 ./debs UbuntuまたはDebianの場合、 ./vsphere VMware の場合)。

Azureとの独自のAPI統合を作成するには、`storagegrid-ssoauth-azure.py`スクリプト。Python スクリプトは、StorageGRIDに2つのリクエストを直接送信します(最初に SAMLRequest を取得し、後で認証トークンを取得します)。また、Node.js スクリプトを呼び出して Azure と対話し、SSO 操作を実行します。

SSO 操作は一連の API リクエストを使用して実行できますが、実行するのは簡単ではありません。Puppeteer Node.js モジュールは、Azure SSO インターフェイスをスクレイピングするために使用されません。

URL エンコードの問題がある場合は、次のエラーが表示されることがあります。Unsupported SAML version。

手順

1. 次のように、必要な依存関係をインストールします。
 - a. Node.jsをインストールする（"<https://nodejs.org/en/download/>"）。
 - b. 必要な Node.js モジュール (puppeteer と jsdom) をインストールします。

```
npm install -g <module>
```

2. Python スクリプトを Python インタープリターに渡してスクリプトを実行します。

次に、Python スクリプトは対応する Node.js スクリプトを呼び出して、Azure SSO のやり取りを実行します。

3. プロンプトが表示されたら、次の引数の値を入力します (またはパラメータを使用して渡します)。
 - Azure へのサインインに使用する SSO 電子メール アドレス
 - StorageGRIDのアドレス
 - テナント管理 API にアクセスする場合のテナント アカウント ID
4. プロンプトが表示されたら、パスワードを入力し、要求された場合に Azure に MFA 認証を提供できるように準備します。

```
c:\Users\user\Documents\azure_sso>py storagegrid-azure-ssoauth.py --sso-email-address user@my-domain.com
--sg-address storagegrid.examp.e.com --tenant-account-id 0
Enter the user's SSO password:
*****

Watch for and approve a 2FA authorization request
*****
StorageGRID Auth Token: {'responseTime': '2021-10-04T21:30:48.807Z', 'status': 'success', 'apiVersion':
'3.4', 'data': '4807d93e-a3df-48f2-9680-906cd255979e'}
```



このスクリプトでは、MFA が Microsoft Authenticator を使用して実行されることを前提としています。他の形式の MFA (テキスト メッセージで受信したコードの入力など) をサポートするには、スクリプトを変更する必要がある場合があります。

StorageGRID認証トークンが出力に提供されます。SSO が使用されていない場合に API を使用するのと同様に、他のリクエストにもトークンを使用できるようになりました。

シングル サインオンが有効になっている場合は **API** を使用する (**PingFederate**)

もしあなたが"**シングル サインオン (SSO) を設定して有効にする**"PingFederate を SSO プロバイダーとして使用する場合は、一連の API 要求を発行して、Grid Management API または Tenant Management API に有効な認証トークンを取得する必要があります。

シングル サインオンが有効になっている場合は、**API** に**Sign in**。

これらの手順は、PingFederateをSSO IDプロバイダーとして使用している場合に適用されます。

開始する前に

- StorageGRIDユーザー グループに属するフェデレーション ユーザーの SSO ユーザー名とパスワードがわかっています。
- テナント管理 API にアクセスする場合は、テナント アカウント ID がわかっている必要があります。

タスク概要

認証トークンを取得するには、次のいずれかの例を使用できます。

- その `storagegrid-ssoauth.py`StorageGRID` インストール ファイル ディレクトリにある Python スクリプト (`./rpms`Red Hat Enterprise Linux` の場合、 `./debs`Ubuntu` または `Debian` の場合、 `./vsphere`VMware` の場合)。
- `curl` リクエストのワークフローの例。

`curl` ワークフローは、実行速度が遅すぎるとタイムアウトする可能性があります。次のエラーが表示される場合があります: `A valid SubjectConfirmation was not found on this Response.`



サンプルの `curl` ワークフローでは、パスワードが他のユーザーから見られないように保護されません。

URL エンコードの問題がある場合は、次のエラーが表示されることがあります。 `Unsupported SAML version.`

手順

1. 認証トークンを取得するには、次のいずれかの方法を選択します。
 - 使用 `storagegrid-ssoauth.py`Python` スクリプト。ステップ 2 に進みます。
 - `curl` リクエストを使用します。ステップ 3 に進みます。
2. 使用したい場合は `storagegrid-ssoauth.py`スクリプト` の場合は、スクリプトを Python インタープリターに渡してスクリプトを実行します。

プロンプトが表示されたら、次の引数の値を入力します。

- SSO 方式。「pingfederate」の任意のバリエーション (`PINGFEDERATE`、`pingfederate` など) を入力できます。
- SSO ユーザー名
- `StorageGRID` がインストールされているドメイン。このフィールドは `PingFederate` では使用されません。空白のままにすることも、任意の値を入力することもできます。
- `StorageGRID` のアドレス
- テナント管理 API にアクセスする場合のテナント アカウント ID。

```
python3 storagegrid-ssoauth.py
sso_method: pingfederate
saml_user: my-sso-username
saml_domain:
sg_address: storagegrid.example.com
tenant_account_id: 12345
Enter the user's SAML password:
*****
*****
StorageGRID Auth Token: 56eb07bf-21f6-40b7-afob-5c6cacfb25e7
```

StorageGRID認証トークンが出力に提供されます。SSO が使用されていない場合に API を使用するのと同様に、他のリクエストにもトークンを使用できるようになりました。

3. curl リクエストを使用する場合は、次の手順に従います。

a. サインインに必要な変数を宣言します。

```
export SAMLUSER='my-sso-username'  
export SAMLPASSWORD='my-password'  
export TENANTACCOUNTID='12345'  
export STORAGEGRID_ADDRESS='storagegrid.example.com'
```



グリッド管理APIにアクセスするには、0を次のように使用します。
TENANTACCOUNTID。

b. 署名された認証URLを受け取るには、POSTリクエストを発行します。/api/v3/authorize-saml、レスポンスから追加の JSON エンコーディングを削除します。

この例は、TENANTACCOUNTID の署名付き認証 URL に対する POST リクエストを示しています。結果は `python -m json.tool` に渡され、JSON エンコーディングが削除されます。

```
curl -X POST "https://$STORAGEGRID_ADDRESS/api/v3/authorize-saml" \  
  -H "accept: application/json" -H "Content-Type: application/json" \  
  --data "{\"accountId\": \"$TENANTACCOUNTID\"}" | python -m \  
  json.tool
```

この例の応答には、URL エンコードされた署名付き URL が含まれますが、追加の JSON エンコードレイヤーは含まれません。

```
{  
  "apiVersion": "3.0",  
  "data": "https://my-pf-baseurl/idp/SSO.saml2?...",  
  "responseTime": "2018-11-06T16:30:23.355Z",  
  "status": "success"  
}
```

c. 保存する `SAMLRequest` 後続のコマンドで使用するために応答から取得します。

```
export SAMLREQUEST="https://my-pf-baseurl/idp/SSO.saml2?..."
```

d. 応答と Cookie をエクスポートし、応答をエコーします。

```
RESPONSE=$(curl -c - "$SAMLREQUEST")
```

```
echo "$RESPONSE" | grep 'input type="hidden" name="pf.adapterId" id="pf.adapterId"'
```

- e. 'pf.adapterId' の値をエクスポートし、応答をエコーします。

```
export ADAPTER='myAdapter'
```

```
echo "$RESPONSE" | grep 'base'
```

- f. 'href' 値をエクスポートし (末尾のスラッシュ / を削除)、応答をエコーします。

```
export BASEURL='https://my-pf-baseurl'
```

```
echo "$RESPONSE" | grep 'form method="POST"'
```

- g. 「アクション」 値をエクスポートします。

```
export SSOPING='/idp/.../resumeSAML20/idp/SSO.ping'
```

- h. 資格情報とともに Cookie を送信します。

```
curl -b <(echo "$RESPONSE") -X POST "$BASEURL$SSOPING" \  
--data "pf.username=$SAMLUSER&pf.pass=  
$SAMLPASSWORD&pf.ok=clicked&pf.cancel=&pf.adapterId=$ADAPTER"  
--include
```

- i. 保存する `SAMLResponse` 隠しフィールドから:

```
export SAMLResponse='PHNhbwXwOlJlc3BvbnN...1scDpSZXNwb25zZT4='
```

- j. 保存した SAMLResponse、StorageGRIDを作成する/api/saml-responseStorageGRID認証トークン
を生成するためのリクエスト。

のために RelayState、テナント アカウント ID を使用するか、Grid Management API にサインイン
する場合は 0 を使用します。

```
curl -X POST "https://$STORAGEGRID_ADDRESS:443/api/saml-response" \  
-H "accept: application/json" \  
--data-urlencode "SAMLResponse=$SAMLResponse" \  
--data-urlencode "RelayState=$TENANTACCOUNTID" \  
| python -m json.tool
```

応答には認証トークンが含まれます。

```
{  
  "apiVersion": "3.0",  
  "data": "56eb07bf-21f6-40b7-af0b-5c6cacfb25e7",  
  "responseTime": "2018-11-07T21:32:53.486Z",  
  "status": "success"  
}
```

- a. レスポンス内の認証トークンを次のように保存します。MYTOKEN。

```
export MYTOKEN="56eb07bf-21f6-40b7-af0b-5c6cacfb25e7"
```

使用できるようになりました `MYTOKEN` その他のリクエストについては、SSO が使用されていない場合に API を使用する方法と同様です。

シングル サインオンが有効になっている場合は、**API** からサインアウトします。

シングル サインオン (SSO) が有効になっている場合は、グリッド管理 API またはテナント管理 API からサインアウトするための一連の API 要求を発行する必要があります。これらの手順は、PingFederateをSSO IDプロバイダーとして使用している場合に適用されます。

タスク概要

必要に応じて、組織のシングル ログアウト ページからログアウトすることで、StorageGRID API からサインアウトできます。または、有効なStorageGRIDベアラ トークンを必要とするStorageGRIDからシングル ログアウト (SLO) をトリガーすることもできます。

手順

1. 署名付きログアウト リクエストを生成するには、`cookie "sso=true"` を SLO API に渡します。

```
curl -k -X DELETE "https://$STORAGEGRID_ADDRESS/api/v3/authorize" \  
-H "accept: application/json" \  
-H "Authorization: Bearer $MYTOKEN" \  
--cookie "sso=true" \  
| python -m json.tool
```

ログアウト URL が返されます:

```
{
  "apiVersion": "3.0",
  "data": "https://my-ping-
url/idp/SLO.saml2?SAMLRequest=fZDNboMwEIRfhZ...HcQ%3D%3D",
  "responseTime": "2021-10-12T22:20:30.839Z",
  "status": "success"
}
```

2. ログアウト URL を保存します。

```
export LOGOUT_REQUEST='https://my-ping-
url/idp/SLO.saml2?SAMLRequest=fZDNboMwEIRfhZ...HcQ%3D%3D'
```

3. ログアウト URL にリクエストを送信して、SLO をトリガーし、StorageGRIDにリダイレクトします。

```
curl --include "$LOGOUT_REQUEST"
```

302 応答が返されます。リダイレクト場所は、API のみのログアウトには適用されません。

```
HTTP/1.1 302 Found
Location: https://$STORAGEGRID_ADDRESS:443/api/saml-
logout?SAMLResponse=fVLLasMwEPwVo7ss%...%23rsa-sha256
Set-Cookie: PF=QoKs...SgCC; Path=/; Secure; HttpOnly; SameSite=None
```

4. StorageGRIDベアラー トークンを削除します。

StorageGRIDベアラー トークンの削除は、SSO がない場合と同じように機能します。`cookie "sso=true" が提供されない場合、ユーザーは SSO 状態に影響を与えずにStorageGRIDからログアウトされます。

```
curl -X DELETE "https://$STORAGEGRID_ADDRESS/api/v3/authorize" \
-H "accept: application/json" \
-H "Authorization: Bearer $MYTOKEN" \
--include
```

あ `204 No Content`この応答は、ユーザーが現在サインアウトしていることを示します。

```
HTTP/1.1 204 No Content
```

APIを使用して機能を無効にする

Grid Management API を使用すると、StorageGRIDシステムの特定の機能を完全に非アクティブ化できます。機能が非アクティブ化されると、その機能に関連するタスクを実行する権限を誰にも割り当てることができなくなります。

タスク概要

非アクティブ化された機能システムを使用すると、StorageGRIDシステム内の特定の機能へのアクセスを禁止できます。機能を非アクティブ化することは、ルート ユーザーまたはルート アクセス 権限を持つ管理者グループに属するユーザーがその機能を使用できないようにする唯一の方法です。

この機能がどのように役立つかを理解するには、次のシナリオを検討してください。

A社は、テナント アカウントを作成してStorageGRIDシステムのストレージ容量をリースするサービス プロバイダーです。賃借人の物件のセキュリティを保護するために、A社は、アカウントが展開された後は自社の従業員がテナント アカウントにアクセスできないようにしたいと考えています。

A社は、グリッド管理 API の非アクティブ化機能システムを使用してこの目標を達成できます。A社は、グリッド マネージャー (UI と API の両方) の テナント ルート パスワードの変更 機能を完全に無効にすることで、管理者ユーザー (ルート ユーザーおよび ルート アクセス 権限を持つグループに属するユーザーを含む) が、テナント アカウントのルート ユーザーのパスワードを変更できないようにします。

手順

1. グリッド管理 API の Swagger ドキュメントにアクセスします。見る["グリッド管理APIを使用する"](#)。
2. 機能の非アクティブ化エンドポイントを見つけます。
3. テナント ルート パスワードの変更などの機能を無効にするには、次のような本文を API に送信します。

```
{ "grid": {"changeTenantRootPassword": true} }
```

リクエストが完了すると、テナント ルート パスワードの変更機能は無効になります。テナント ルート パスワードの変更 管理権限はユーザー インターフェースに表示されなくなり、テナントのルート パスワードを変更しようとするすべての API 要求は「403 禁止」で失敗します。

無効化された機能を再有効化

デフォルトでは、Grid Management API を使用して、非アクティブ化された機能を再アクティブ化できます。ただし、非アクティブ化された機能が再度アクティブ化されないようにしたい場合は、**activateFeatures** 機能自体を非アクティブ化することができます。



activateFeatures 機能は再アクティブ化できません。この機能を非アクティブ化することにした場合、非アクティブ化された他の機能を再アクティブ化することができなくなることに注意してください。失われた機能を復元するには、テクニカル サポートに連絡する必要があります。

手順

1. グリッド管理 API の Swagger ドキュメントにアクセスします。
2. 機能の非アクティブ化エンドポイントを見つけます。
3. すべての機能を再アクティブ化するには、次のように本文を API に送信します。

```
{ "grid": null }
```

この要求が完了すると、テナント ルート パスワードの変更機能を含むすべての機能が再アクティブ化されます。テナント ルート パスワードの変更 管理権限がユーザー インターフェイスに表示されるようになり、ユーザーがルート アクセスまたはテナント ルート パスワードの変更 管理権限を持っていると仮定すると、テナントのルート パスワードを変更しようとするすべてのAPI 要求が成功します。



前の例では、非アクティブ化されたすべての機能が再アクティブ化されます。非アクティブ化されていて非アクティブのままにしておく必要がある他の機能が非アクティブ化されている場合は、PUT リクエストでそれらを明示的に指定する必要があります。たとえば、テナント ルート パスワードの変更機能を再度アクティブ化し、storageAdmin 管理権限を引き続き非アクティブ化するには、次の PUT リクエストを送信します。+ { "grid": {"storageAdmin": true} }

著作権に関する情報

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

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

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

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

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

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

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

商標に関する情報

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