テナント管理API

StorageGRID software

NetApp
December 03, 2025

目次

テナント管理API

テナント管理APIを理解する

テナント マネージャー ユーザー インターフェイスの代わりに、テナント管理 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、開発者と非開発者が API を操作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 ユーザー インターフェースは、各 API 操作の完全な詳細とドキュメントを提供します。
- 用途"中断のないアップグレードをサポートするためのバージョン管理"。

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Sign in。
2. テナント マネージャーの上部から、ヘルプ アイコンを選択し、**API** ドキュメント を選択します。

APIの処理

テナント管理 API は、利用可能な API 操作を次のセクションに分類します。

- **account**: ストレージ使用状況情報の取得など、現在のテナント アカウントに対する操作。
- **auth**: ユーザーセッション認証を実行する操作。

  テナント管理 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。テナントログインの場合、認証リクエストのJSON本文にユーザー名、パスワード、アカウントIDを指定します（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。

  認証セキュリティの向上については、以下を参照してください。"クロスサイトリクエストフォージェリから保護する" 。

  > 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、認証のために別の手順を実行する必要があります。参照"グリッド管理APIの使用手順"。

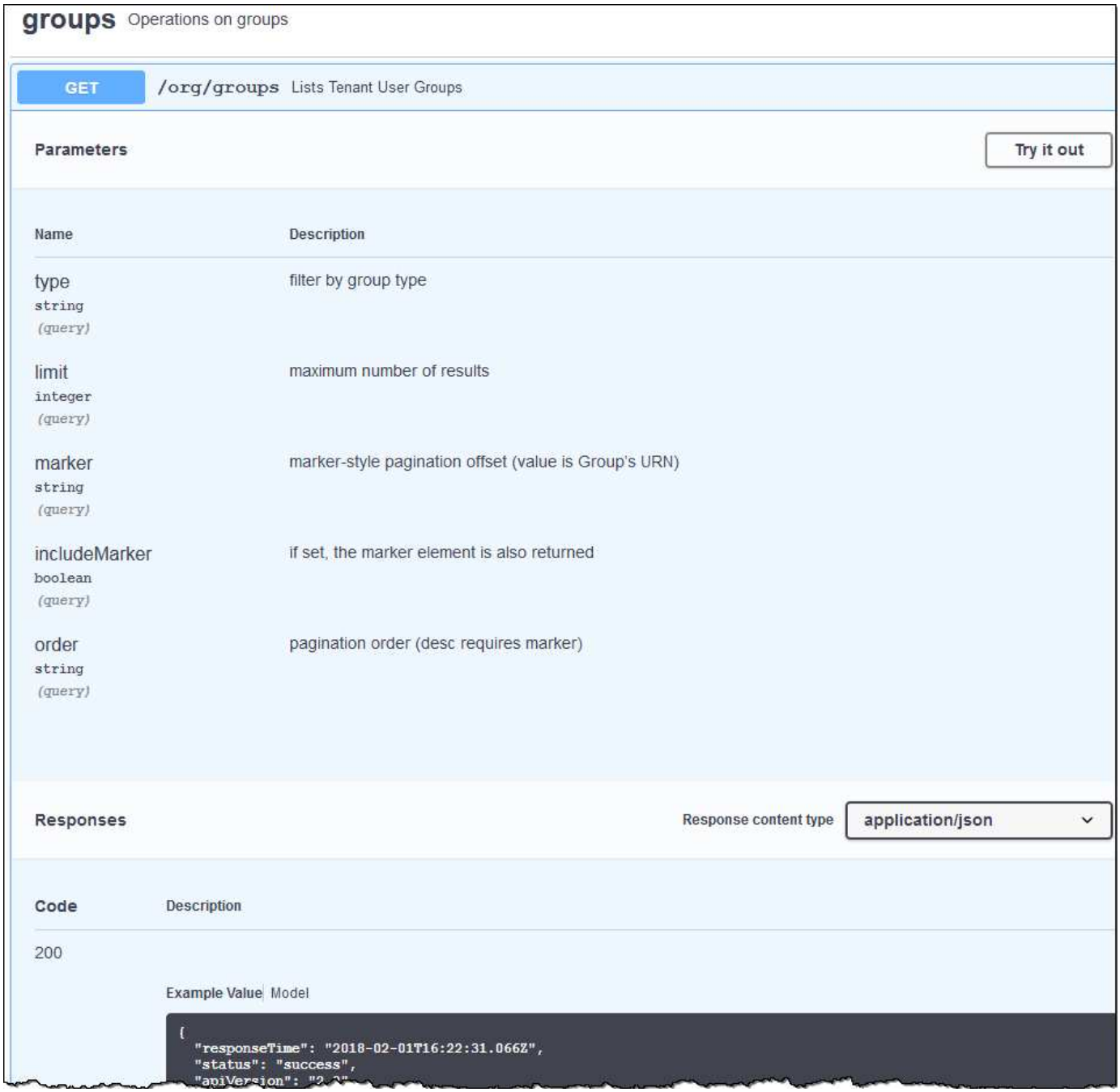
- **config**: テナント管理 API の製品リリースとバージョンに関連する操作。製品のリリース バージョンと、そのリリースでサポートされている API のメジャー バージョンを一覧表示できます。
- コンテナ: S3 バケットまたは Swift コンテナに対する操作。
- **deactivated-features**: 非アクティブ化された可能性のある機能を表示する操作。
- エンドポイント: エンドポイントを管理するための操作。エンドポイントにより、S3 バケットはStorageGRID CloudMirror レプリケーション、通知、または検索統合に外部サービスを使用できるよう

になります。

- **grid-federation-connections**: グリッド フェデレーション接続およびグリッド間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) 設定に関する操作。
- **regions**: StorageGRIDシステムに設定されているリージョンを判別する操作。
- **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**: 規制コンプライアンスをサポートするために使用される、グローバル S3 オブジェクトロック設定に対する操作。
- **users**: テナント ユーザーを表示および管理する操作。

操作の詳細

各 API 操作を展開すると、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、リクエスト本文の例 (必要な場合)、および可能な応答が表示されます。



APIリクエストを発行する

API ドキュメント 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. 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ます。
2. 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。
3. 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。

4. *試してみる*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ます。
6. *実行*を選択します。
7. 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。

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。

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"。

> 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。

著作権に関する情報

商標に関する情報