Understanding the Tenant Management API

StorageGRID 11.5

NetApp
January 04, 2024

Table of Contents

Understanding the Tenant Management API

You can perform system management tasks using the Tenant Management REST API instead of the Tenant Manager user interface. For example, you might want to use the API to automate operations or to create multiple entities, such as users, more quickly.

The Tenant Management API uses the Swagger open source API platform. Swagger provides an intuitive user interface that allows developers and non-developers to interact with the API. The Swagger user interface provides complete details and documentation for each API operation.

To access the Swagger documentation for the Tenant Management API:

Steps

1. Sign in to the Tenant Manager.
2. Select **Help** > **API Documentation** from the Tenant Manager header.

API operations

The Tenant Management API organizes the available API operations into the following sections:

- **account** — Operations on the current tenant account, including getting storage usage information.
- **auth** — Operations to perform user session authentication.

  The Tenant Management API supports the Bearer Token Authentication Scheme. For a tenant login, you provide a username, password, and accountId in the JSON body of the authentication request (that is, `POST /api/v3/authorize`). If the user is successfully authenticated, a security token is returned. This token must be provided in the header of subsequent API requests ("Authorization: Bearer token").

  See “Protecting against Cross-Site Request Forgery” for information on improving authentication security.

  > If single sign-on (SSO) is enabled for the StorageGRID system, you must perform different steps to authenticate. See “Authenticating in to the API if single sign-on is enabled” in the instructions for administering StorageGRID.

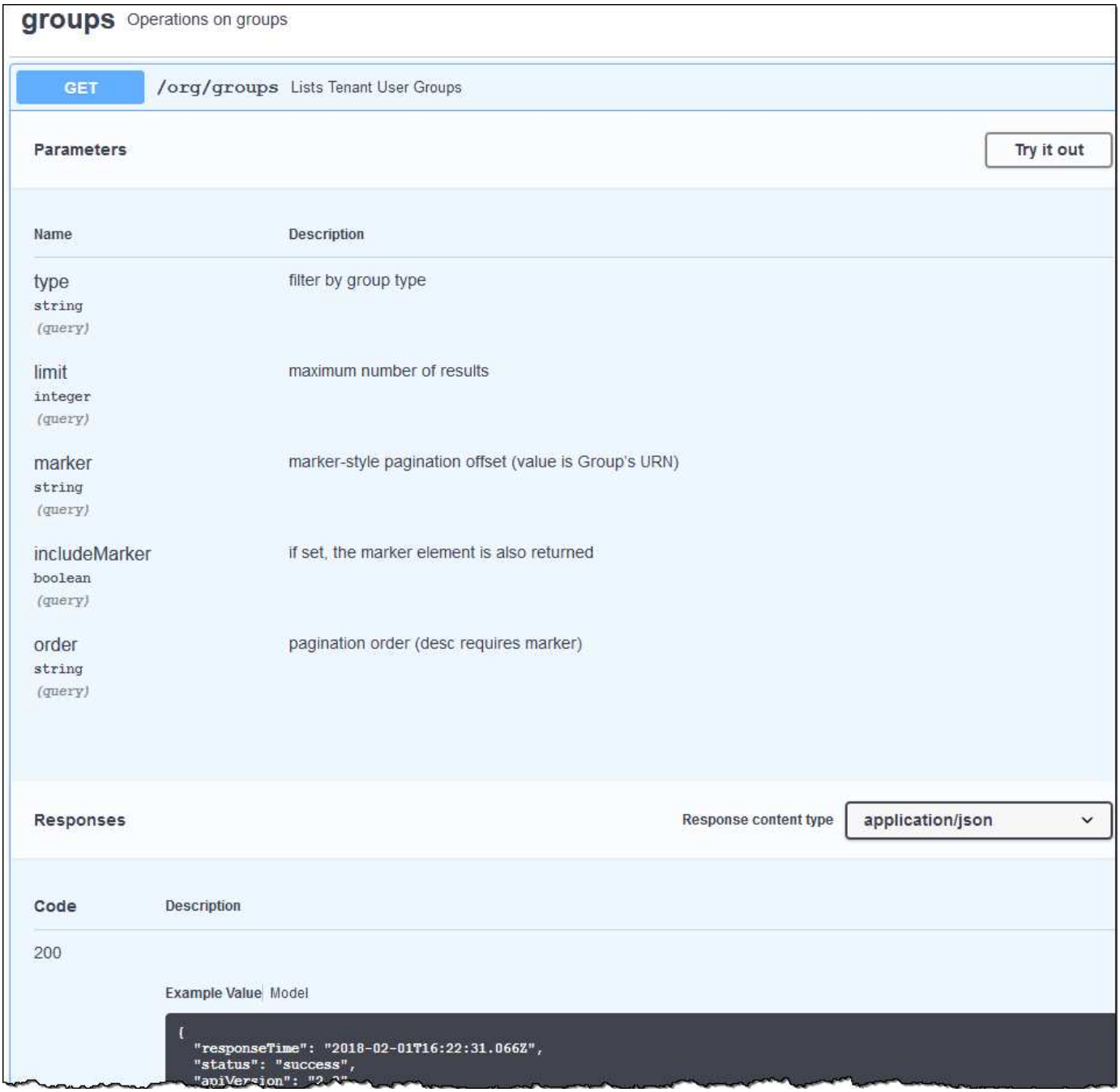
- **config** — Operations related to the product release and versions of the Tenant Management API. You can list the product release version and the major versions of the API supported by that release.
- **containers** — Operations on S3 buckets or Swift containers, as follows:

Protocol	Permission allows
S3	• Creating compliant and non-compliant buckets • Modifying legacy compliance settings • Setting the consistency control for operations performed on objects • Creating, updating, and deleting a bucket's CORS configuration • Enabling and disabling last access time updates for objects • Managing the configuration settings for platform services, including CloudMirror replication, notifications, and search integration (metadata-notification) • Deleting empty buckets
Swift	Setting the consistency level used for containers

- **deactivated-features** — Operations to view features that might have been deactivated.
- **endpoints** — Operations to manage an endpoint. Endpoints allow an S3 bucket to use an external service for StorageGRID CloudMirror replication, notifications, or search integration.
- **groups** — Operations to manage local tenant groups and to retrieve federated tenant groups from an external identity source.
- **identity-source** — Operations to configure an external identity source and to manually synchronize federated group and user information.
- **regions** — Operations to determine which regions have been configured for the StorageGRID system.
- **s3** — Operations to manage S3 access keys for tenant users.
- **s3-object-lock** — Operations to determine how global S3 Object Lock (compliance) is configured for the StorageGRID system.
- **users** — Operations to view and manage tenant users.

Operation details

When you expand each API operation, you can see its HTTP action, endpoint URL, a list of any required or optional parameters, an example of the request body (when required), and the possible responses.



Issuing API requests

Any API operations you perform using the API Docs webpage are live operations. Be careful not to create, update, or delete configuration data or other data by mistake.

Steps

1. Click the HTTP action to see the request details.
2. Determine if the request requires additional parameters, such as a group or user ID. Then, obtain these values. You might need to issue a different API request first to get the information you need.
3. Determine if you need to modify the example request body. If so, you can click **Model** to learn the requirements for each field.

4. Click **Try it out**.
5. Provide any required parameters, or modify the request body as required.
6. Click **Execute**.
7. Review the response code to determine if the request was successful.

Related information

Protecting against Cross-Site Request Forgery (CSRF)

Administer StorageGRID

Tenant Management API versioning

The Tenant Management API uses versioning to support non-disruptive upgrades.

For example, this Request URL specifies version 3 of the API.

```
https://hostname_or_ip_address/api/v3/authorize
```

The major version of the Tenant Management API is bumped when changes are made that are ***not compatible*** with older versions. The minor version of the Tenant Management API is bumped when changes are made that ***are compatible*** with older versions. Compatible changes include the addition of new endpoints or new properties. The following example illustrates how the API version is bumped based on the type of changes made.

Type of change to API	Old version	New version
Compatible with older versions	2.1	2.2
Not compatible with older versions	2.1	3.0

When StorageGRID software is installed for the first time, only the most recent version of the Tenant Management API is enabled. However, when StorageGRID is upgraded to a new feature release, you continue to have access to the older API version for at least one StorageGRID feature release.

Outdated requests are marked as deprecated in the following ways:

- The response header is "Deprecated: true"
- The JSON response body includes "deprecated": true

Determining which API versions are supported in the current release

Use the following API request to return a list of the supported API major versions:

```
GET https://{{IP-Address}}/api/versions
{
  "responseTime": "2019-01-10T20:41:00.845Z",
  "status": "success",
  "apiVersion": "3.0",
  "data": [
    2,
    3
  ]
}
```

Specifying an API version for a request

You can specify the API version using a path parameter (`/api/v3`) or a header (`Api-Version: 3`). If you provide both values, the header value overrides the path value.

```
curl https://[IP-Address]/api/v3/grid/accounts

curl -H "Api-Version: 3" https://[IP-Address]/api/grid/accounts
```

Protecting against Cross-Site Request Forgery (CSRF)

You can help protect against Cross-Site Request Forgery (CSRF) attacks against StorageGRID by using CSRF tokens to enhance authentication that uses cookies. The Grid Manager and Tenant Manager automatically enable this security feature; other API clients can choose whether to enable it when they sign in.

An attacker that can trigger a request to a different site (such as with an HTTP form POST) can cause certain requests to be made using the signed-in user's cookies.

StorageGRID helps protect against CSRF attacks by using CSRF tokens. When enabled, the contents of a specific cookie must match the contents of either a specific header or a specific POST body parameter.

To enable the feature, set the `csrfToken` parameter to `true` during authentication. The default is `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"
```

When true, a `GridCsrfToken` cookie is set with a random value for sign-ins to the Grid Manager, and the

`AccountCsrfToken` cookie is set with a random value for sign-ins to the Tenant Manager.

If the cookie is present, all requests that can modify the state of the system (POST, PUT, PATCH, DELETE) must include one of the following:

- The `X-Csrf-Token` header, with the value of the header set to the value of the CSRF token cookie.
- For endpoints that accept a form-encoded body: A `csrfToken` form-encoded request body parameter.

See the online API documentation for additional examples and details.

> Requests that have a CSRF token cookie set will also enforce the `"Content-Type: application/json"` header for any request that expects a JSON request body as an additional protection against CSRF attacks.

Copyright information

Trademark information