REST implementation details

Cloud Manager Automation

NetApp
October 07, 2021

Table of Contents

REST implementation details

While REST establishes a common set of technologies and best practices, the details of each API can vary based on the design choices of the development team. You should be aware of the details and operational characteristics of the Cloud Manager REST API before using it with a live deployment.

Basic concepts

Representational State Transfer (REST) is a style for creating distributed web applications. When applied to the design of a web services API, it establishes a set of technologies and best practices for exposing server-based resources and managing their states. The Cloud Manager REST API uses mainstream protocols and standards to provide a flexible foundation for deploying and administering Cloud Volumes ONTAP instances and the associated resources.

Overview of the API resources

The REST style of application development begins by identifying the set of server-based resources. The Cloud Manager resources are broadly categorized as follows:

- Administrative resources

  Used to set up and configure Cloud Manager

- Auditing resources

  Used to view details about Cloud Manager activities and operations

- Authentication resources

  Used to authenticate to Cloud Manager so you can make API calls

- Working environment resources

  Used to deploy and manage working environments, including: single Cloud Volumes ONTAP systems, Cloud Volumes ONTAP HA configurations, and ONTAP clusters

REST endpoints

The REST resources are accessed through endpoints based on the URL path. Each endpoint provides access to one of the following:

- Resource instance
- Collection of resource instances

See API reference for more information.

Types of input parameters

The Cloud Manager REST API uses several types of parameters in the HTTP request.

Type	Description
Path parameter	Identifiers or names for resource instances that are included in the URL path.
Query parameter	One or more key-value pairs at the end of the URL which qualify and extend the base call.
Request header	Key-value pairs in the request which carry additional information available to the server.
Body parameter	Data which is optionally included with a request and formatted using JSON.

Security

The Cloud Manager REST API provides robust security based on token authentication and authorization.

In addition to token authorization, all connections are protected using the Transport Layer Security (TLS) protocol.

All NetApp Cloud Central Services, including Cloud Manager, use OAuth 2.0 for authorization. OAuth 2.0 is an open standard implemented by several authorization providers including **Auth0**. Connecting and communicating with a secure REST endpoint is a two-step process:

1. Acquire a JWT (JSON Web Token) access token from the trusted OAuth 2.0 token endpoint
2. Make a REST API call to the target endpoint with the access token in the `Authorization: Bearer` request header

Authorization can be performed in a federated or non-federated environment. The type of authorization environment you have determines which token and procedure to use.

Authorization

You must use a valid token to access the API based on the authorization mode.

Users with federated authorization need to create a token using Create a user token with federated authentication. Non-federated user can optionally use this type of token.

Users with non-federated authorization need to create a token using Create a user token with nonfederated authentication.

HTTP details

The Cloud Manager REST API is based on the HTTP protocol as well as JSON for content exchange. This section describes the details of how HTTP is used.

Request

HTTP methods

The HTTP methods supported by the Cloud Manager REST API are shown in the following table. Not all the HTTP methods are available at each of the REST endpoints. See the API reference documentation for more information.

HTTP method	Description
GET	Retrieves object properties for a resource instance or collection of resources.
POST	Creates a new resource instance based on the supplied input values.
PUT	Updates an existing resource instance based on the supplied input values.
DELETE	Deletes an existing resource instance.

Request headers

The common request headers are presented below.

Request header	Description
Authorization	This header contains a bearer token used to access the server.
x-agent-id	The agent identifier is based on the client ID and is used to identify the user agent.
Content-Type	This representation header is used to indicate the original media type of the resource.
Accept	The server automatically returns content in JSON format if Accept header is not specified.

Response

HTTP status codes

The HTTP status codes used by the Cloud Manager REST API are described below.

Status code	Reason Phrase	Description
200	OK	The request was completed successfully.
202	Accepted	The request was accepted and is currently in process. Cloud Manager returns this code when the API call operates asynchronously. For example, the `/vsa/working-environments` call returns with 202 but the Cloud Volumes ONTAP instance launches up to 25 minutes later.
204	No Content	The operation was completed successfully and the server did not send a response message.
400	Bad Request	The request input is not recognized or is inappropriate. An error response explains the reason.
401	Unauthorized	The user has not authenticated.
403	Forbidden	This operation is not allowed for the current authenticated user.
409	Conflict	The operation failed because another operation is already in progress.

Status code	Reason Phrase	Description
420	---	Cloud Manager has not been set up. You must set up Cloud Manager using the API call /occm/setup/init
5xx	---	An unexpected error occurred within the Cloud Manager server which has prevented it from fulfilling the request.

Additional considerations

There are several additional characteristics of the Cloud Manager REST API affecting its operation and use. You should be aware of these considerations before issuing an API call.

Public identifiers

All Cloud Manager resources (for example, working environments) are assigned a public ID. Whenever a resource is created or returned, the public ID is displayed in the response. You must specify a resource's public ID when performing operations on the resource. For example, you must specify the public ID for a working environment when you create a volume.

Asynchronous processing

All HTTP request methods except GET are processed asynchronously. If needed, you can check the status of an active task based on the `request id` returned in the original HTTP response. Each task has a status value as shown in the following table.

Status	Description
1	The asynchronous task completed successfully.
0	The background task is still running and has not completed.
-1	The asynchronous task completed but failed.

For more information about how to retrieve the status of a background task for an asynchronous request see Get active task.