



Red Hat Satellite 6.2

API Guide

Reference documentation for using Satellite's Representational State
Transfer (REST) API

Edition 1.0

Red Hat Satellite 6.2 API Guide

Reference documentation for using Satellite's Representational State Transfer (REST) API
Edition 1.0

Red Hat Satellite Documentation Team
satellite-doc-list@redhat.com

Legal Notice

Copyright © 2017 Red Hat, Inc.

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](#). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

The Red Hat Satellite 6.2 Representational State Transfer (REST) API guide explains the concepts behind a REST API and provides example usage for various types of requests. This provides a basis for administrators and developers to write custom scripts and integrate Red Hat Satellite with third-party applications.

Table of Contents

CHAPTER 1. INTRODUCTION TO RED HAT SATELLITE API	3
1.1. THE RED HAT SATELLITE API	3
1.2. REPRESENTATIONAL STATE TRANSFER	3
1.3. SATELLITE API COMPARED TO HAMMER CLI TOOL	4
CHAPTER 2. API REFERENCE	5
2.1. UNDERSTANDING THE API SYNTAX	5
Using the GET HTTP Verb	5
Using the POST HTTP Verb	6
Using the PUT HTTP Verb	8
Using the DELETE HTTP Verb	9
Relating API Error Messages to the API Reference	9
2.2. UNDERSTANDING THE JSON RESPONSE FORMAT	9
JSON Response Format for Collections	10
JSON Response Format for Single Objects	11
CHAPTER 3. AUTHENTICATING API CALLS	12
3.1. USING SSL AUTHENTICATION	12
3.2. USING HTTP AUTHENTICATION	13
3.3. USING OAUTH AUTHENTICATION	14
3.3.1. Configuring OAuth	14
3.3.2. Making an OAuth Request	14
CHAPTER 4. GETTING STARTED WITH THE RED HAT SATELLITE API	16
4.1. API EXAMPLES USING CURL	16
4.1.1. Performing Simple Queries	16
4.1.2. Creating and Modifying Resources	19
4.1.2.1. Uploading Content to the Satellite Server	21
4.1.3. Overriding Smart Class Parameters	24
4.1.3.1. Modifying a Smart Class Parameter Using an External File	25
4.1.4. Applying Errata to a Host or Host Collection	29
4.2. API EXAMPLES USING RUBY	29
4.2.1. Creating Objects Using Ruby	30
4.2.2. Using Apipie Bindings	31
4.3. API EXAMPLES USING PYTHON	33
4.3.1. Creating Objects Using Python	33
4.3.2. Running Queries Using Python	35
4.4. USING EXTENDED SEARCHES	37
4.5. USING SEARCHES WITH PAGINATION CONTROL	37
4.6. WORKING WITH LIFE CYCLE ENVIRONMENTS	38
Updating a Life Cycle Environment	41
Deleting a Life Cycle Environment	41
APPENDIX A. API RESPONSE CODES	42
APPENDIX B. API PERMISSIONS MATRIX	43

CHAPTER 1. INTRODUCTION TO RED HAT SATELLITE API

Red Hat Satellite is a system management solution that allows organizations to deploy, configure, and maintain their systems across physical, virtual, and cloud environments. It allows for provisioning, remote management and monitoring of multiple Red Hat Enterprise Linux deployments with a single, centralized tool. *Red Hat Satellite Server* synchronizes the content from Red Hat Customer Portal, and provides functionality including fine-grained life cycle management, user and group role-based access control, integrated subscription management, as well as advanced GUI, CLI, or API access.

1.1. THE RED HAT SATELLITE API

Red Hat Satellite provides a *Representational State Transfer (REST)* API. The API provides software developers and system administrators with control over their Red Hat Satellite environment outside of the standard web interface. The REST API is useful for developers and administrators who aim to integrate the functionality of Red Hat Satellite with custom scripts or external applications that access the API over HTTP.



NOTE

The full API reference is available on your Satellite Server at <https://satellite6.example.com/apidoc/v2.html> (replace *satellite6.example.com* with the host name of your Satellite Server). Be aware that even though versions 1 and 2 of the Satellite 6 API are available, Red Hat only supports version 2.

The benefits of the REST API are:

- *Broad client support*: Any programming language, framework, or system with support for HTTP protocol can use the API;
- *Self descriptive*: Client applications require minimal knowledge of the Red Hat Satellite infrastructure as many details are discovered at runtime;
- *Resource based model*: The resource based REST model provides a natural way to manage a virtualization platform.

This provides developers and administrators the ability to:

- integrate with enterprise IT systems;
- integrate with third-party applications;
- perform automated maintenance or error checking tasks;
- automate repetitive tasks with scripts.

1.2. REPRESENTATIONAL STATE TRANSFER

Representational State Transfer (REST) is a software design architecture that focuses on resources for a specific service and their representations. A resource representation is a key abstraction of information that corresponds to one specific managed element on a server. A client sends a request to a server element located at a Uniform Resource

Identifier (URI) and performs operations with standard HTTP methods, such as **GET**, **POST**, **PUT**, and **DELETE**. This provides a stateless communication between the client and server where each request acts independent of any other request and contains all necessary information to complete the request.

1.3. SATELLITE API COMPARED TO HAMMER CLI TOOL

For many tasks, both Hammer and Satellite API are equally applicable. Hammer can be used as a human-friendly interface to Satellite API, for example to test responses to API calls before applying them in a script (use the **-d** option to inspect API calls issued by Hammer, for example **hammer -d organization list**). Changes in the API are automatically reflected in Hammer, while scripts using the API directly have to be updated manually.

In the background, each Hammer command first establishes a binding to the API, then sends a request. This can have performance implications when executing a large number of Hammer commands in sequence. In contrast, a script communicating directly with the API establishes the binding only once. See the [Hammer CLI Guide](#) for more information.

CHAPTER 2. API REFERENCE

The full API reference is available on your Satellite Server at <https://satellite6.example.com/apidoc/v2.html> (replace *satellite6.example.com* with the host name of your Satellite Server). Be aware that even though versions 1 and 2 of the Satellite 6 API are available, Red Hat only supports version 2.

2.1. UNDERSTANDING THE API SYNTAX

The built-in API reference shows the API route, or path, preceded by an HTTP verb:

```
HTTP_VERB API_ROUTE
```

The HTTP verbs used by the API are GET, POST, PUT, and DELETE. See the HOSTS section of the API reference document at <http://satellite6.example.com/apidoc/v2/hosts.html> for some examples. If you are already familiar with API syntax and the `curl` command you can skip this section.

To work with the API, construct a command using the API route from the reference document and the command syntax from the documentation for the command. For example, the `curl` manual page show the following basic syntax:

```
curl [options] [URL...]
```

The options used in this guide include: `-X`, `--request command`, where *command* is an HTTP verb.

Using the GET HTTP Verb

The GET HTTP verb is used to get data from the API about an existing entry or resource.

Combining an example from the API HOSTS section such as `GET /api/hosts` with the `curl` syntax results in:

```
curl -X GET https://satellite6.example.com/api/hosts
```

Satellite only supports HTTPS for connecting to the API, and some form of authentication is required.

For a usable example, we must add at least a user name with the `-u` option, and the `-k` option to skip SSL peer certificate verification checks:

```
$ curl -X GET -k -u sat_username https://satellite6.example.com/api/hosts
Enter host password for user 'sat_username':
{
  "total": 2,
  "subtotal": 2,
  "page": 1,
  "per_page": 20,
  "search": null,
  "sort": {
    "by": null,
    "order": null
  },
  "results":
  output truncated
```

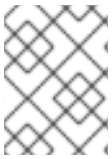
■

The above response from the API indicates that there are two results in total, two results are being returned below, this is the first page of the results, and the maximum results per page is set to 20. This is explained more verbosely in [Section 2.2, “Understanding the JSON Response Format”](#).

Some examples in the API reference include terms preceded by a colon in the form **:parameter**. For example:

```
GET /api/hosts/:id
```

These are API route parameters and must be replaced by an appropriate value. Route parameters start with a colon and end with **id**.



NOTE

In Satellite 6, version 2 of the API is the default. Therefore it is not necessary to use **v2** in the URL for API calls.

Using the POST HTTP Verb

The POST HTTP verb is used to submit data to the API to create a new entry or resource. The data must be in JSON format, and can be included inline using the **-d, --data** option followed by the quoted JSON formatted data enclosed in braces **{}**. Alternatively, the unquoted JSON formatted data can be enclosed in a file, and specified using the **curl** command's **@** option. For example, **-d @file.json**.

The advantages of using external files for JSON formatted data include less problems with quoting and escaping, being able to use your favorite editor with syntax checkers to help you find and avoid mistakes, and external tools to check the validity of JSON data or to reformat it. For example, the **yajl** package contains the **json_verify** tool and the **json_reformat** tool.

Using the **json_verify** tool, you can check the validity of a JSON file as follows:

```
$ json_verify < test_file.json
```

The unstructured JSON data returned by an API call can be piped through the python module **json.tool**:

```
curl API_call | python -m json.tool
```

Alternately, use the **json_reformat** tool:

```
curl API_call | json_reformat
```

The output format is explained in [Section 2.2, “Understanding the JSON Response Format”](#).

The API Reference includes the following in the **Activation keys** section:

```
POST /katello/api/activation_keys
```

This is one possible format for a **POST /katello/api/activation_keys** command:

```
curl -X POST -k -u sat_username:sat_password \
```

```
-d @file_of_json-formatted_data \
https://satellite6.example.com/katello/api/activation_keys
```

To see how the POST HTTP verb works, create a test file, for example, **activation-key.json**, with contents as follows:

```
{"organization_id":1, "name":"TestKey", "description":"Just for testing"}
```

The following example will create a new Activation key by applying the data in the file just created:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" -X POST \
-u sat_username:sat_password -k \
-d @activation-key.json \
https://satellite6.example.com/katello/api/activation_keys | json_reformat
{
  "id": 2,
  "name": "TestKey",
  "description": "Just for testing",
  "unlimited_hosts": true,
  "auto_attach": true,
  "content_view_id": null,
  "environment_id": null,
  "usage_count": 0,
  "user_id": 3,
  "max_hosts": null,
  "release_version": null,
  "service_level": null,
  "content_overrides": [

  ],
  "organization": {
    "name": "Default Organization",
    "label": "Default_Organization",
    "id": 1
  },
  "created_at": "2017-02-16 12:37:47 UTC",
  "updated_at": "2017-02-16 12:37:48 UTC",
  "content_view": null,
  "environment": null,
  "products": null,
  "host_collections": [

  ],
  "permissions": {
    "view_activation_keys": true,
    "edit_activation_keys": true,
    "destroy_activation_keys": true
  }
}
```

To view this entry in the web UI, navigate to **Content** → **Activation keys**. Remember to reload the page after any changes.

Using the PUT HTTP Verb

The PUT HTTP verb is used to submit data to the API to update an existing entry or resource. Similarly to the POST API call, the data must be in JSON format, and can be included inline using the **-d**, **--data** option followed by the quoted JSON formatted data enclosed in braces **{}**. Alternatively, the unquoted JSON formatted data can be enclosed in a file, and specified using the **curl** command's **@** option. For example, **-d @file.json**.

To change an existing value or append to an existing resource use the PUT HTTP verb. The API reference has the following entry for updating an Activation key:

```
PUT /katello/api/activation_keys/:id
```

To update an existing Activation key, use a command in the following format:

```
curl -X PUT -k -u sat_username:sat_password \  
-d @file_of_json-formatted_data \  
https://satellite6.example.com/katello/api/activation_keys/:id
```

Replace *:id* with the ID of the Activation key to be updated. Using the PUT command multiple times with the same values will **not** create multiple entries.

For example, the test Activation key created in the previous example can be updated by editing the file created previously as follows:

```
{"organization_id":1, "name":"TestKey", "description":"Just for testing", "max_hosts":"10" }
```

Use a command as follows to apply the changes in the JSON file:

```
$ curl -H "Accept:application/json,version=2" \  
-H "Content-Type:application/json" -X PUT \  
-u sat_username:sat_password -k \  
-d @activation-key.json \  
https://satellite6.example.com/katello/api/activation_keys/2  
{  
  "id": 2,  
  "name": "TestKey",  
  "description": "Just for testing",  
  "unlimited_hosts": false,  
  "auto_attach": true,  
  "content_view_id": null,  
  "environment_id": null,  
  "usage_count": 0,  
  "user_id": 3,  
  "max_hosts": 10,  
  "release_version": null,  
  "service_level": null,  
  "content_overrides": [  
  
  ],  
  "organization": {  
    "name": "Default Organization",  
    "label": "Default_Organization",  
    "id": 1  
  },  
}
```

```

    "created_at": "2017-02-16 12:37:47 UTC",
    "updated_at": "2017-02-16 12:46:17 UTC",
    "content_view": null,
    "environment": null,
    "products": null,
    "host_collections": [

    ],
    "permissions": {
      "view_activation_keys": true,
      "edit_activation_keys": true,
      "destroy_activation_keys": true
    }
  }
}

```

Using the DELETE HTTP Verb

To delete a resource, use the DELETE verb with an API route that includes the ID of the resource to be deleted.

To delete an existing Activation key, use a command in the following format:

```

curl -X DELETE -k -u sat_username:sat_password \
https://satellite6.example.com/katello/api/activation_keys/:id

```

Replace *:id* with the ID of the Activation key to be deleted. For example:

```

$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" -X DELETE \
-u admin:RedHat1! -k \
https://satellite6.example.com/katello/api/activation_keys/2 |
json_reformat
output omitted
  "started_at": "2017-02-16 12:58:17 UTC",
  "ended_at": "2017-02-16 12:58:18 UTC",
  "state": "stopped",
  "result": "success",
  "progress": 1.0,
  "input": {
    "activation_key": {
      "id": 2,
      "name": "TestKey"
    }
  }
output truncated

```

Relating API Error Messages to the API Reference

The API uses a RAILS format to indicate an error:

```

Nested_Resource.Attribute_Name

```

This translates to the following format used in the API reference:

```

Resource[Nested_Resource_attributes][Attribute_Name_id]

```

2.2. UNDERSTANDING THE JSON RESPONSE FORMAT

Calls to the API using GET will return results in JSON format. Passing the output through the Python `json.tool` module gives a more human readable format.

JSON Response Format for Collections

Collections are a list of objects such as hosts and domains. The format for a collection JSON response consists of a metadata fields section followed by a results section. Below is an example of the format for a collection JSON response for a list of domains when using the API route `GET /api/domains`. The output was piped through `json.tool` to make the results section easier to read.

```
$ curl -X GET -k -u admin:password
https://satellite6.example.com/api/domains | python -m json.tool
{
  "total": 3,
  "subtotal": 3,
  "page": 1,
  "per_page": 20,
  "search": null,
  "sort": {
    "by": null,
    "order": null
  },
  "results": [
    {
      "id": 23,
      "name": "qa.lab.example.com",
      "fullname": "QA",
      "dns_id": 10,
      "created_at": "2013-08-13T09:02:31Z",
      "updated_at": "2013-08-13T09:02:31Z"
    },
    {
      "id": 25,
      "name": "sat.lab.example.com",
      "fullname": "SATLAB",
      "dns_id": 8,
      "created_at": "2013-08-13T08:32:48Z",
      "updated_at": "2013-08-14T07:04:03Z"
    },
    {
      "id": 32,
      "name": "hr.lab.example.com",
      "fullname": "HR",
      "dns_id": 8,
      "created_at": "2013-08-16T08:32:48Z",
      "updated_at": "2013-08-16T07:04:03Z"
    }
  ]
}
```

The response metadata fields are described below:

- **total** — The total number of objects without any search parameters.
- **subtotal** — The number of objects returned with the given search parameters (if there is no search, then subtotal is equal to total).

- **page** — The page number.
- **per_page** — The maximum number of objects returned per page.
- **limit** — The specified number of objects to return in a collection response.
- **offset** — The number of objects skipped before returning a collection.
- **search** — The search string based on **scoped_scoped** syntax.
- **sort**
 - **by** — The field that the collection is sorted by.
 - **order** — The sort order, either ASC for ascending or DESC for descending.
- **results** — The collection of objects.

JSON Response Format for Single Objects

Single-object JSON responses are used to show a single object. The object's unique identifier, **:id** or **:name**, is required in the GET request. Note that **:name** cannot always be used as a unique identifier, but **:id** can always be used. The format for a single-object JSON response consists of only the object's attributes.

Below is an example of the format for a single-object JSON response when using the API route **GET /api/domains/23** or **GET /api/domains/qa.lab.example.com**.

```
$ curl -X GET -k -u admin:password
https://satellite6.example.com/api/domains/23 | python -m json.tool
{
  "id": 23,
  "name": "qa.lab.example.com",
  "fullname": "QA",
  "dns_id": 10,
  "created_at": "2013-08-13T09:02:31Z",
  "updated_at": "2013-08-13T09:02:31Z"
}
```

CHAPTER 3. AUTHENTICATING API CALLS

Interaction with the Satellite API requires authentication. You can download the Satellite Server CA certificate to your local host for use in each API request to provide SSL authentication. Each API request also requires a valid user name and password. Each of these is discussed in the following sections.

3.1. USING SSL AUTHENTICATION

Red Hat Satellite uses HTTPS, which provides a degree of encryption and identity verification when communicating with a Red Hat Satellite Server. Non-SSL communications are not supported by Satellite 6.

Each Red Hat Satellite Server uses a self signed certificate. This certificate acts as both the server certificate to verify the encryption key and the certificate authority (CA) to trust the identity of the Satellite Server. The following steps show how to set up a SSL authentication for the Satellite Server (in this case *satellite6.example.com*):

1. Obtain a certificate from the Satellite Server with which you want to communicate (*satellite6.example.com*) using one of the following options:

- a. To obtain a certificate using SSH, run the following command:

```
$ scp root@satellite6.example.com:/var/www/html/pub/katello-server-ca.crt ./
```

- b. If you execute the command directly on the Satellite Server, obtain a certificate from the locally available copy by running the following command:

```
$ cp /var/www/html/pub/katello-server-ca.crt ./
```

- c. To obtain a certificate using HTTP, run the following command:

```
$ curl -O http://satellite6.example.com/pub/katello-server-ca.crt
```



WARNING

Retrieving a certificate using an unencrypted HTTP connection might present a security risk.

2. Use the certificate on your client as a certificate authority to verify the identity of the Satellite Server:

```
$ curl -X GET -u sat_username:sat_password \  
-H "Accept:application/json" --cacert katello-server-ca.crt \  
https://satellite6.example.com/katello/api/organizations
```

GET is the default action and therefore **-X GET** attribute can be omitted here.

3. Create a Network Security Services (NSS) database to store the certificate:

```
$ certutil -N -d sql:$HOME/.pki/nssdb
Enter a password which will be used to encrypt your keys.
The password should be at least 8 characters long,
and should contain at least one non-alphabetic character.

Enter new password:
Re-enter password:
```

If the NSS database already exists you will be prompted for the password as follows:

```
Enter Password or Pin for "NSS Certificate DB":
```

4. Use the following command to permanently include the certificate in the NSS database:

```
$ certutil -d sql:$HOME/.pki/nssdb -A -t TC -n "Red Hat Satellite" \
-i katello-server-ca.crt
```

This imports the certificate into the NSS database, which means you can omit the `--cacert` option for each request. You can test this as follows:

```
$ curl -X GET -u sat_username:sat_password
https://satellite6.example.com/api/v2/hosts
{
  "total": 2,
  ...,
  "results": [
    ...
  ]
}
output omitted
```

3.2. USING HTTP AUTHENTICATION

All requests to the Satellite API require a suitable user name and password. The API uses HTTP Basic Authentication ^[1] to encode these credentials, which are then added to the Authorization header. If a request does not include an appropriate Authorization header, the API returns a 401 Authorization Required error.

IMPORTANT

Basic authentication involves potentially sensitive information, such as passwords, sent as plain text. The REST API requires HTTPS for transport level encryption of plain text requests.

Some base64 libraries break encoded credentials into multiple lines and terminate each line with a newline character. This invalidates the header and causes a faulty request. The authorization header requires that the encoded credentials be on a single line within the header.

3.3. USING OAUTH AUTHENTICATION

As an alternative to basic authentication, limited OAuth 1.0 authentication is supported in the API (this is sometimes referred to as 1-legged OAuth in version 1.0a of the protocol).

3.3.1. Configuring OAuth

OAuth is enabled by default in Satellite 6.2. The configuration settings are stored in the `/etc/foreman/settings.yaml` configuration file and can be viewed in the web UI by going to **Administer** → **Settings** → **Auth**. The **OAuth consumer key** is the token to be used by all OAuth clients. Certain values in the web UI cannot be changed. These values can only be changed by running the `satellite-installer` script again with the new options. Note that any manual changes to the file will be lost when upgrading. Enter the following command to see all the OAuth related installer options:

```
# satellite-installer --full-help | grep oauth
```

If you want all API requests made using OAuth to be authorized as the built-in anonymous API administrator account, then keep **OAuth map users** set to **false** in the `/etc/foreman/settings.yaml` file. If you want to specify the user under which the request is made, change this configuration option to **true**. This allows clients to send a FOREMAN-USER header with the login of an existing Foreman user.



IMPORTANT

The header is not signed in an OAuth request, it can therefore be forged. Anyone with valid consumer key can impersonate any Foreman user.

3.3.2. Making an OAuth Request

Usually an OAuth client library is used to generate the request. An example OAuth request using `curl` is shown here to aid in understanding how it works.

Example 3.1. OAuth Request Example Using curl

```
$ curl 'https://satellite6.example.com/api/architectures' \
-H 'Content-Type: application/json' \
-H 'Accept: application/json,version=2' \
-H 'FOREMAN-USER: User1' \
-H 'Authorization: OAuth
oauth_version="1.0",oauth_consumer_key="secretkey",oauth_signature_metho
d="hmac-
sha1",oauth_timestamp=1321473112,oauth_signature=I18hR8/ogj/XVu0qMPB9qNj
Sy6E='
```

In the example above architectures are listed using OAuth for authentication. The request is attempted as a user with login `User1`. If mapping is enabled in the Foreman settings, the result will only include architectures that user `User1` can see. Note that the signature was constructed manually, this should change with any **oauth_timestamp** change. Also, the signature reflects every parameter, HTTP method, and URI change. Therefore it is recommend to use an OAuth client library to construct all OAuth parameters.

[1] Basic Authentication is described in [RFC 2617 HTTP Authentication: Basic and Digest Access Authentication](#).

CHAPTER 4. GETTING STARTED WITH THE RED HAT SATELLITE API

This chapter provides a range of examples of how to use the Red Hat Satellite API to perform different tasks. These examples focus on the Satellite Server, using HTTPS on port 443. You can also access the API via the Satellite Capsule, but you need to use port 8443 or the API calls will fail.

You can address these different port requirements within the script itself. For example, in Ruby, you can specify the Satellite and Capsule URLs as follows:

```
url = 'https://satellite6.example.com/api/v2/'
capsule_url = 'https://capsule.example.com:8443/api/v2/'
katello_url = 'https://satellite6.example.com/katello/api/v2/'
```

If the host is subscribed to a Satellite Server or Capsule Server, you can determine the correct port required to access the API from the `/etc/rhsm/rhsm.conf` file, in the `port` entry of the `[server]` section. You can use these values to fully automate your scripts, removing any need to verify which ports to use.

4.1. API EXAMPLES USING CURL

This section describes how to use `curl` to perform various tasks using the Satellite API.

4.1.1. Performing Simple Queries

The following examples describe how to use `curl` to search for information about your Satellite deployment. These examples include both the actual command and some sample output, and example values for user names and passwords. Expect different results for each deployment. These examples also use the `python -m json.tool` command to format the output.



NOTE

Red Hat Satellite requires the use of HTTPS, and by default a certificate for host identification. If you have not added the Satellite Server certificate as described in [Section 3.1, “Using SSL Authentication”](#), then you can use the `-k` (insecure) option to bypass certificate checks.

For user authentication, you can use the form `-u username:password` or, if you do not include the password, the command prompts you to enter it. Red Hat recommends that you do not include the password as part of the command, because it then becomes part of your shell history and might present a security risk. These examples include the password only for the sake of simplicity.

Be aware that if you use the `-s` (silent) option with `curl` that you will not see a progress meter or any error messages.

Retrieving a List of Resources

The following is a basic query that returns a list of resources. Such requests return a list of data wrapped in metadata, while other request types only return the actual object.

■

```
$ curl -X GET -s -k -u sat_username:sat_password
https://satellite6.example.com/api/v2/hosts | python -m json.tool

{
    "total" => 2,
    "subtotal" => 2,
    "page" => 1,
    "per_page" => 1000,
    "search" => nil,
    "sort" => {
        "by" => nil,
        "order" => nil
    },
    "results" => [
        ...
    ]
}
```

Example 4.1. Listing Users

```
$ curl -X GET -s -k -u sat_username:sat_password
https://satellite6.example.com/api/users
{
    "total": 1,
    "subtotal": 1,
    "page": 1,
    "per_page": 20,
    "search": null,
    "sort": {
        "by": null,
        "order": null
    },
    "results":
    [{"firstname": "Admin", "lastname": "User", "mail": "root@example.com", "admin": true, "auth_source_id": 1, "auth_source_name": "Internal", "timezone": null, "locale": null, "last_login_on": "2017-02-08 23:25:51 UTC", "created_at": "2017-01-09 12:10:02 UTC", "updated_at": "2017-02-08 23:25:51 UTC", "id": 3, "login": "admin", "default_location": null, "locations": [], "default_organization": {"id": 1, "name": "Default Organization", "title": "Default Organization", "description": null}, "organizations": []}]
}
```

Running a Generic Host Query

The following query returns information for the host **satellite6.example.com**:

```
$ curl -X GET -s -k -u sat_username:sat_password
https://satellite6.example.com/api/v2/hosts/satellite6.example.com |
python -m json.tool
{
    "all_puppetclasses": [],
    "architecture_id": 1,
    "architecture_name": "x86_64",
```

```
    "build": false,
    "capabilities": [
      "build"
    ],
    "certname": "satellite6.example.com",
    "comment": null,
    "compute_profile_id": null,
    ...
  }
```

Searching for Facts for a Specific Host

The following query returns all facts for the host `satellite6.example.com`:

```
$ curl -X GET -s -k -u sat_username:sat_password
https://satellite6.example.com/api/v2/hosts/satellite6.example.com/facts |
python -m json.tool
{
  ...
  "results": {
    "satellite6.example.com": {
      "augeasversion": "1.0.0",
      "bios_release_date": "01/01/2007",
      "bios_version": "0.5.1",
      "blockdevice_sr0_size": "1073741312",
      "facterversion": "1.7.6",
      ...
    }
  }
}
```

Searching all Hosts for Matching Patterns

The following query returns all hosts that match the pattern "example":

```
$ curl -X GET -s -k -u sat_username:sat_password
https://satellite6.example.com/api/v2/hosts?search=example | python -m
json.tool
{
  ...
  "results": [
    {
      "name": "satellite6.example.com",
      ...
    }
  ],
  "search": "example",
  ...
}
```

Searching for all Hosts in a Specific Environment

The following query returns all hosts in the "production" environment:

```
$ curl -X GET -s -k -u sat_username:sat_password
https://satellite6.example.com/api/v2/hosts?search=environment=production
| python -m json.tool
{
```

```

...
"results": [
  {
    "environment_name": "production",
    "name": "satellite6.example.com",
    ...
  }
],
"search": "environment=production",
...
}

```

Searching for all Hosts with a Specific Fact Value

The following query returns all hosts with a model name "RHEV Hypervisor":

```

$ curl -X GET -s -k -u sat_username:sat_password
https://satellite6.example.com/api/v2/hosts?
search=model=\"RHEV+Hypervisor\" | python -m json.tool
{
  ...
  "results": [
    {
      "model_id": 1,
      "model_name": "RHEV Hypervisor",
      "name": "satellite6.example.com",
      ...
    }
  ],
  "search": "model=\"RHEV Hypervisor\"",
  ...
}

```

4.1.2. Creating and Modifying Resources

You can use the Satellite API to manipulate resources on the Satellite Server. These API calls require that you pass various parameters beyond the simple user name, password, and URI that you want to query. For example, to upload content to your Satellite Server, or to modify Satellite resources, you need to include extra information in the header when you construct your request.

You can specify the version of the API either in the header, as described in the following examples, or as part of the URL. For example, `https://satellite6.example.com/api/v2/architectures` is the equivalent of using **Accept: version=2** in the request header. The URL specification takes precedence.

The following is the basic syntax for a POST request:

```

$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" -X POST \
-u username:password -k \
-d json-formatted-data https://satellite6.example.com

```

For example, to create a new architecture, you can use the following example request:

```
$ curl -H "Accept:application/json,version=2" \
      -H "Content-Type:application/json" -X POST -u
sat_username:sat_password \
      -k -d "{\"architecture\":{\"name\":\"i686\"}}" \
      https://satellite6.example.com/api/architectures
```

This returns output similar to the following:

```
{"name":"i686","id":3,"created_at":"2015-10-
29T13:21:09Z","updated_at":"2015-10-29T13:21:09Z","operatingsystems":
[],"images":[]}
```

You can use the following command to verify that the architecture was created:

```
$ curl -X GET -u sat_username:sat_password -k
https://satellite6.example.com/api/v2/architectures | python -m json.tool
{
  "page": 1,
  "per_page": 20,
  "results": [
    {
      "created_at": "2015-04-02T05:29:46Z",
      "id": 2,
      "name": "i386",
      "updated_at": "2015-04-02T05:29:46Z"
    },
    {
      "created_at": "2015-04-02T05:29:46Z",
      "id": 1,
      "name": "x86_64",
      "updated_at": "2015-04-02T05:29:46Z"
    },
    {
      "created_at": "2015-11-04T19:40:15Z",
      "id": 3,
      "name": "i686",
      "updated_at": "2015-11-04T19:40:15Z"
    }
  ],
  "search": null,
  "sort": {
    "by": null,
    "order": null
  },
  "subtotal": 3,
  "total": 3
}
```

You can also use **hammer** on the Satellite Server to verify the results:

```
$ hammer -u sat_username -p sat_password architecture list
---|-----
ID | NAME
---|-----
2  | i386
```



```
1 | x86_64
3 | i686
---|-----
```

Example 4.2. Creating a New User

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" -X POST \
-u sat_username:sat_password -k \
-d "{\"firstname\":\"Test\",\"lastname\":\"API-
User\",\"mail\":\"test@example.com\",\"login\":\"test_api\",\"password\"
: \"123456\",\"auth_source_id\":1}" \
https://satellite6.example.com/api/users
```

4.1.2.1. Uploading Content to the Satellite Server

This section describes how to use `curl` with the Satellite 6 API to upload and import large files to your Satellite Server. This process involves four steps:

1. Create an upload request.
2. Upload the content.
3. Import the content.
4. Delete the upload request.

The maximum file size that you can upload is about 30 MB. To upload larger content, refer to [Example 4.3, “Uploading Content Larger than 30 MB”](#).

Procedure 4.1. Uploading Content to the Satellite Server

1. Create the upload request. Ensure you modify the example parameters to suit your deployment:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" \
-X POST \
-u sat_username:sat_password -k -d "{}" \

https://satellite6.example.com/katello/api/repositories/3/content_uploads
```

This command returns the `upload_id` similar to the following:

```
{"upload_id":"0be156b1-f373-4cad-89d0-
924f8f4491d2","_href":"/pulp/api/v2/content/uploads/0be156b1-f373-
4cad-89d0-924f8f4491d2/"}
```

2. Upload your content. Ensure you use the correct MIME type when you upload data. The "application/json" MIME type is used for the majority of requests to Satellite 6. Combine the `upload_id`, MIME type, and other parameters to upload content:

■

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:multipart/form-data" \
-X PUT \
-u sat_username:sat_password \
-k --data-urlencode
"content@/home/sat6user/rpmbuild/RPMS/noarch/python-scripttest-
1.1.1-1.fc21.noarch.rpm" \
--data-urlencode offset=0 \

https://satellite6.example.com/katello/api/repositories/3/content_uploads/0be156b1-f373-4cad-89d0-924f8f4491d2
```

3. After you have uploaded the content to the Satellite Server, you need to import it into the appropriate repository. Until you complete this step, the Satellite Server will not be aware of the new content:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" \
-X PUT \
-u sat_username:sat_password \
-k -d "{\"upload_ids\":[\"0be156b1-f373-4cad-89d0-924f8f4491d2\"]}" \

https://satellite6.example.com/katello/api/repositories/3/import_uploads
```

4. After you have successfully uploaded and imported your content, you can delete the upload request. This frees any temporary disk space that was used during the upload:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" \
-X DELETE -d "{}" \
-u sat_username:sat_password \
-k

https://satellite6.example.com/katello/api/repositories/3/content_uploads/0be156b1-f373-4cad-89d0-924f8f4491d2
```

Example 4.3. Uploading Content Larger than 30 MB

The following example demonstrates in full how to split a large file into chunks, create an upload request, upload the individual files, import them to Satellite, and then delete the upload request. Note that this example uses sample content, host names, user names, and file names.

1. Download the sample module:

```
$ wget https://forgeapi.puppetlabs.com/v3/files/theforeman-foreman-5.0.1.tar.gz?_ga=1.267255502.1792403825.1430297670 -O theforeman-foreman-5.0.1.tar.gz
```

Split the module into 50,000 byte chunks:

```
$ split --bytes 50000 --numeric-suffixes --suffix-length=1
theforeman-foreman-5.0.1.tar.gz foreman_module.
```

View the resulting files:

```
$ ls -la theforeman-foreman-5.0.1.tar.gz foreman_module.*
-rw-r--r--. 1 root root 50000 Nov  4 04:42 foreman_module.0
-rw-r--r--. 1 root root 32928 Nov  4 04:42 foreman_module.1
-rw-r--r--. 1 root root 82928 Nov  4 04:41 theforeman-foreman-
5.0.1.tar.gz
```

2. Create a new upload request (this is the equivalent of `cat` on the Satellite Server).

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" \
-X POST \
-u sat_username:sat_password -k -d "{}" \
https://ibm-
vm01.example.com/katello/api/repositories/2/content_uploads
```

The above command returns an upload ID:

```
{"upload_id":"9585528f-07ad-4bb1-9c80-
ccece249b2b7","_href":"/pulp/api/v2/content/uploads/9585528f-07ad-
4bb1-9c80-ccece249b2b7/"}
```

3. Upload the file chunks that you created in Step 1. Notice the use of the ***offset*** parameter in this example and how it relates to the file size:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:multipart/form-data" \
-X PUT \
-u sat_username:sat_password \
-k --data-urlencode "content@foreman_module.0" \
--data-urlencode offset=0 \
https://ibm-
vm01.example.com/katello/api/repositories/2/content_uploads/958552
8f-07ad-4bb1-9c80-ccece249b2b7
```

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:multipart/form-data" \
-X PUT \
-u sat_username:sat_password \
-k --data-urlencode "content@foreman_module.1" \
--data-urlencode offset=50000 \
https://ibm-
vm01.example.com/katello/api/repositories/2/content_uploads/958552
8f-07ad-4bb1-9c80-ccece249b2b7
```

4. Import the complete upload to the repository:

```
$ curl -H "Accept:application/json,version=2" \
```

```
-H "Content-Type:application/json" \
-X PUT \
-u sat_username:sat_password \
-k -d "{\"upload_ids\":[\"9585528f-07ad-4bb1-9c80-
ccece249b2b7\"]}" \
https://ibm-
vm01.example.com/katello/api/repositories/2/import_uploads
```

5. Delete the upload request:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" \
-X DELETE -d "{}" \
-u sat_username:sat_password \
-k https://ibm-
vm01.example.com/katello/api/repositories/2/content_uploads/958552
8f-07ad-4bb1-9c80-ccece249b2b7
```

6. Log in to the Satellite Server to check if the file was transferred correctly:

```
$ ls -la /var/lib/pulp/content/puppet_module/foreman-foreman-
5.0.1.tar.gz
-rw-r--r--. 1 apache apache 82928 Nov  4 04:55
/var/lib/pulp/content/puppet_module/foreman-foreman-
5.0.1.tar.gz
```

Compare the files:

```
$ cmp /var/lib/pulp/content/puppet_module/foreman-foreman-
5.0.1.tar.gz theforeman-foreman-5.0.1.tar.gz
```

```
$ echo $?
0
```

4.1.3. Overriding Smart Class Parameters

You can search for Smart Parameters using the API and supply a value to override a Smart Parameter in a Class. The full list of attributes that can be modified can be found in the built-in API reference at

https://satellite6.example.com/apidoc/v2/smart_class_parameters/update.html.

For example, to list all Smart Class Parameters, the API route is shown as **GET** `/api/smart_class_parameters`. Using **curl**, the command is as follows:

```
$ curl -X GET -s -k -u sat_username:sat_password \
https://satellite6.example.com/api/smart_class_parameters
```

If you know the Puppet class ID, for example 5, you can restrict the scope as follows:

```
$ curl -X GET -s -k -u sat_username:sat_password
https://satellite6.example.com/api/puppetclasses/5/smart_class_parameters
```

Both calls accept a search parameter. The full list of searchable fields is visible in the web UI in the search input box. Navigate to **Configure** → **Smart variables** and click in the search query box to reveal the list of fields.

Two particularly useful search parameters are **puppetclass_name** and **key**, which enables you to search for a specific parameter. For example, using the **-d, --data** option to pass URL encoded data:

```
$ curl -X GET -s -k -u sat_username:sat_password
https://satellite6.example.com/api/smart_class_parameters -d
'search=puppetclass_name = access_insights_client and key = authmethod'
```

Standard scoped-search syntax is supported.

Once you have found the ID of the parameter, you can proceed with listing the full details including current override values. For example, for an ID of 63, the API route is **GET /api/smart_class_parameters/63**. Using **curl**, the command would be:

```
$ curl -X GET -s -k -u sat_username:sat_password \
https://satellite6.example.com/api/smart_class_parameters/63
```

Now you can enable overriding of parameter values with a PUT call:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" -X PUT \
-s -k -u sat_username:sat_password \
-d '{"smart_class_parameter":{"override":true}}' \
https://satellite6.example.com/api/smart_class_parameters/63
```

Note that there is no way to create or delete the parameters manually. Users can only modify their attributes. Parameters get created and deleted only upon class import from a proxy.

When override is enabled, you can add custom override matchers:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" -X PUT \
-s -k -u sat_username:sat_password \
-d '{"smart_class_parameter":{"override_value":
{"match":"hostgroup=Test","value":"2.4.6"}}}' \
https://satellite6.example.com/api/smart_class_parameters/63
```

Details about all parameters of the API call are here:

https://satellite6.example.com/apidoc/v2/override_values.html.

To delete override values use a command as follows:

```
$ curl -X DELETE -s -u sat_username:sat_password \
https://satellite6.example.com/api/smart_class_parameters/63/override_valu
es/3
```

4.1.3.1. Modifying a Smart Class Parameter Using an External File

Using external files simplifies working with JSON data. Using an editor with syntax highlighting can help you avoid and locate mistakes.

Procedure 4.2. Modifying a Smart Class Parameter Using an External File

For this example we will use a MOTD Puppet manifest.

1. Search for the Puppet Class by name, in this case, **motd**:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" -X GET \
-u sat_user:sat_passwd -k \
"https://satellite6.example.com/api/smart_class_parameters?
search=puppetclass_name=motd" \
| python -m json.tool
{
  "page": 1,
  "per_page": 20,
  "results": [
    {
      "avoid_duplicates": false,
      "created_at": "2017-02-06 12:37:48 UTC",
      "default_value": "",
      "description": "",
      "hidden_value": "*****",
      "hidden_value?": false,
      "id": 3,
      "merge_default": false,
      "merge_overrides": false,
      "override": false,
      "override_value_order": "fqdn\nhostgroup\nnos\ndomain",
      "override_values_count": 0,
      "parameter": "content",
      "parameter_type": "string",
      "puppetclass_id": 3,
      "puppetclass_name": "motd",
      "required": false,
      "updated_at": "2017-02-07 13:08:42 UTC",
      "use_puppet_default": false,
      "validator_rule": null,
      "validator_type": ""
    },
    {
      "avoid_duplicates": false,
      "created_at": "2017-02-06 12:37:48 UTC",
      "default_value": true,
      "description": "",
      "hidden_value": "*****",
      "hidden_value?": false,
      "id": 1,
      "merge_default": false,
      "merge_overrides": false,
      "override": false,
      "override_value_order": "fqdn\nhostgroup\nnos\ndomain",
      "override_values_count": 0,"parameter": "dynamic_motd",
      "parameter_type": "boolean",
      "puppetclass_id": 3,
      "puppetclass_name": "motd",
      "required": false,
```

```

"updated_at": "2017-02-06 15:21:06 UTC",
"use_puppet_default": null,
"validator_rule": null,
"validator_type": null
},
{
"avoid_duplicates": false,
"created_at": "2017-02-06 12:37:48 UTC",
"default_value": "",
"description": "",
"hidden_value": "*****",
"hidden_value?": false,
"id": 2,
"merge_default": false,
"merge_overrides": false,
"override": false,
"override_value_order": "fqdn\nnhostgroup\nnos\ndomain",
"override_values_count": 0,
"parameter": "template",
"parameter_type": "string",
"puppetclass_id": 3,
"puppetclass_name": "motd",
"required": false,
"updated_at": "2017-02-06 15:21:06 UTC",
"use_puppet_default": null,
"validator_rule": null,
"validator_type": null
}
],
"search": "puppetclass_name=motd",
"sort": {
"by": null,
"order": null
},
"subtotal": 3,
"total": 66
}

```

Each Smart Class Parameter has an ID that is global for the same Satellite instance. The **content** parameter of the **motd** class has **id=3** in this Satellite Server. Do not confuse this with the Puppet Class ID which appears just before the Puppet Class name.

2. Use the parameter ID **3** to get the information specific to the **motd** parameter and redirect the output to a file, for example, **output_file.json**:

```

$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" -X GET \
-u sat_user:sat_passwd -k \
"https://satellite6.example.com/api/smart_class_parameters/3 \
| python -m json.tool > output_file.json

```

3. Copy the file created in the previous step to a new file for editing, for example, **changed_file.json**. Open the file in an editor and modify the desired values. In this example, we wish to change the content parameter of the **motd** module, which requires changing the **override** option from **false** to **true**:

```
{
  "avoid_duplicates": false,
  "created_at": "2017-02-06 12:37:48 UTC", # This line must be
  removed.
  "default_value": "", # A new value should be supplied here.
  "description": "",
  "hidden_value": "*****",
  "hidden_value?": false,
  "id": 3,
  "merge_default": false,
  "merge_overrides": false,
  "override": false, # The override value must be set to true.
  "override_value_order": "fqdn\ngroup\nnos\ndomain",
  "override_values": [], # This line must be removed.
  "override_values_count": 0,
  "parameter": "content",
  "parameter_type": "string",
  "puppetclass_id": 3,
  "puppetclass_name": "motd",
  "required": false,
  "updated_at": "2017-02-07 11:56:55 UTC", # This line must be
  removed.
  "use_puppet_default": false,
  "validator_rule": null,
  "validator_type": ""
}
```

4. After editing the file, verify that it looks as follows and then save the changes:

```
{
  "avoid_duplicates": false,
  "default_value": "No Unauthorized Access Allowed",
  "description": "",
  "hidden_value": "*****",
  "hidden_value?": false,
  "id": 3,
  "merge_default": false,
  "merge_overrides": false,
  "override": true,
  "override_value_order": "fqdn\ngroup\nnos\ndomain",
  "override_values_count": 0,
  "parameter": "content",
  "parameter_type": "string",
  "puppetclass_id": 3,
  "puppetclass_name": "motd",
  "required": false,
  "use_puppet_default": false,
  "validator_rule": null,
  "validator_type": ""
}
```

5. Use a PUT command as follows to apply the changes to Satellite Server:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" \
```



```
-X PUT -u $user:$passwd \  
-d @changed_file.json \  
-k "https://satellite6.example.com/api/smart_class_parameters/3"
```

4.1.4. Applying Errata to a Host or Host Collection

You can use `curl` with the PUT command to apply errata to a host, host group, or host collection. The following is the basic syntax of a PUT request:

```
$ curl -H "Accept:application/json,version=2" \  
      -H "Content-Type:application/json" -X PUT \  
      -u sat_username:sat_password -k \  
      -d json-formatted-data https://satellite6.example.com
```

You can browse the built in API doc (<https://satellite6.example.com/apidoc/v2.html>) to find a URL to use for applying Errata. You can use the Satellite web UI to help discover the format for the search query. Navigate to **Hosts** → **Host Collections** and select a host collection. Go to **Collection Actions** → **Errata Installation** and notice the search query box contents. For example, for a Host Collection called *my-collection* the search box contains `host_collection="my-collection"`. This will be used in the example below for Host Collections.

Example 4.4. Applying Errata to a Host

In this example the API URL for bulk actions, `/katello/api/hosts/bulk/install_content`, is used to show the format required for a simple search.

```
$ curl -H "Accept:application/json,version=2" \  
      -H "Content-Type:application/json" -X PUT \  
      -u sat_username:sat_password -k \  
      -d "{\"organization_id\":1,\"included\":{\"search\":\"my-  
host\"},\"content_type\":\"errata\", \"content\": [\"RHBA-2016:1981\"]}" \  
      https://satellite6.example.com/api/v2/hosts/bulk/install_content
```

Example 4.5. Applying Errata to a Host Collection

In this example, notice the level of escaping required to pass the search string `host_collection="my-collection"` as seen in the Satellite web UI.

```
$ curl -H "Accept:application/json,version=2" \  
      -H "Content-Type:application/json" -X PUT \  
      -u sat_username:sat_password -k \  
      -d "{\"organization_id\":1,\"included\":{\"  
search\":\"host_collection=\\\\"my-  
collection\\\\"\"},\"content_type\":\"errata\", \"content\": [\"RHBA-  
2016:1981\"]}" \  
      https://satellite6.example.com/api/v2/hosts/bulk/install_content
```

4.2. API EXAMPLES USING RUBY

The following examples describe how to perform various tasks using Ruby to communicate with the Satellite API.



IMPORTANT

These are example scripts and commands. Ensure you review these scripts carefully before use, and replace any variables, user names, passwords, and other information to suit your own deployment.

4.2.1. Creating Objects Using Ruby

The following script connects to the Red Hat Satellite 6 API and creates a new organization, and then creates three environments in the new organization. If the organization already exists, the script uses that organization. If any of the environments already exist in the organization, the script raises an error and quits.

```
#!/usr/bin/ruby

require 'rest-client'
require 'json'

url = 'https://satellite6.example.com/api/v2/'
katello_url = "#{url}/katello/api/v2/"

$username = 'admin'
$password = 'changeme'

org_name = "MyOrg"
environments = [ "Development", "Testing", "Production" ]

# Performs a GET using the passed URL location
def get_json(location)
  response = RestClient::Request.new(
    :method => :get,
    :url => location,
    :user => $username,
    :password => $password,
    :headers => { :accept => :json,
                  :content_type => :json }
  ).execute
  JSON.parse(response.to_str)
end

# Performs a POST and passes the data to the URL location
def post_json(location, json_data)
  response = RestClient::Request.new(
    :method => :post,
    :url => location,
    :user => $username,
    :password => $password,
    :headers => { :accept => :json,
                  :content_type => :json},
    :payload => json_data
  ).execute
  JSON.parse(response.to_str)
```

```

end

# Creates a hash with ids mapping to names for an array of records
def id_name_map(records)
  records.inject({}) do |map, record|
    map.update(record['id'] => record['name'])
  end
end

# Get list of existing organizations
orgs = get_json("#{katello_url}/organizations")
org_list = id_name_map(orgs['results'])

if !org_list.has_value?(org_name)
  # If our organization is not found, create it
  puts "Creating organization: \t#{org_name}"
  org_id = post_json("#{katello_url}/organizations",
JSON.generate({"name"=> org_name}))["id"]
else
  # Our organization exists, so let's grab it
  org_id = org_list.key(org_name)
  puts "Organization \"#{org_name}\" exists"
end

# Get list of organization's lifecycle environments
envs = get_json("#{katello_url}/organizations/#{org_id}/environments")
env_list = id_name_map(envs['results'])
prior_env_id = env_list.key("Library")

# Exit the script if at least one life cycle environment already exists
environments.each do |e|
  if env_list.has_value?(e)
    puts "ERROR: One of the Environments is not unique to organization"
    exit
  end
end

# Create life cycle environments
environments.each do |environment|
  puts "Creating environment: \t#{environment}"
  prior_env_id = post_json("#{katello_url}/organizations/#{
org_id}/environments", JSON.generate({"name" => environment,
"organization_id" => org_id, "prior_id" => prior_env_id}))["id"]
end

```

4.2.2. Using Apipie Bindings

Apipie bindings are the Ruby bindings for apipie documented APIs, they fetch and cache the API definition from Satellite and then generate API calls on demand. Using apipie bindings enables you to make simpler Ruby API queries. Apipie is usually pronounced "appy-pie", to rhyme with "happy" without the h.

The following example creates a new organization, and then creates three environments in the new organization. If the organization already exists, the script uses that organization. If any of the environments already exist in the organization, the script raises an error and quits.

```
#!/usr/bin/ruby

require 'apipie-bindings'

org_name = "MyOrg"
environments = [ "Development", "Testing", "Production" ]

# Create an instance of apipie bindings
@api = ApipieBindings::API.new({
  :uri => 'https://satellite6.example.com/',
  :username => 'admin',
  :password => 'changeme',
  :api_version => 2
})

# Performs an API call with default options
def call_api(resource_name, action_name, params = {})
  http_headers = {}
  apipie_options = { :skip_validation => true }
  @api.resource(resource_name).call(action_name, params, http_headers,
  apipie_options)
end

# Creates a hash with IDs mapping to names for an array of records
def id_name_map(records)
  records.inject({}) do |map, record|
    map.update(record['id'] => record['name'])
  end
end

# Get list of existing organizations
orgs = call_api(:organizations, :index)
org_list = id_name_map(orgs['results'])

if !org_list.has_value?(org_name)
  # If our organization is not found, create it
  puts "Creating organization: \t#{org_name}"
  org_id = call_api(:organizations, :create, {'organization' => { :name =>
  org_name }})['id']
else
  # Our organization exists, so let's grab it
  org_id = org_list.key(org_name)
  puts "Organization \"#{org_name}\" exists"
end

# Get list of organization's life cycle environments
envs = call_api(:lifecycle_environments, :index, {'organization_id' =>
  org_id})
env_list = id_name_map(envs['results'])
prior_env_id = env_list.key("Library")

# Exit the script if at least one life cycle environment already exists
environments.each do |e|
  if env_list.has_value?(e)
    puts "ERROR: One of the Environments is not unique to organization"
    exit
  end
end
```

```

    end
end

# Create life cycle environments
environments.each do |environment|
  puts "Creating environment: \t#{environment}"
  prior_env_id = call_api(:lifecycle_environments, :create, {"name" =>
environment, "organization_id" => org_id, "prior_id" => prior_env_id })
  ['id']
end

```

4.3. API EXAMPLES USING PYTHON

The following examples describe how to perform various tasks using Python to communicate with the Satellite API.



IMPORTANT

These are example scripts and commands. Ensure you review these scripts carefully before use, and replace any variables, user names, passwords, and other information to suit your own deployment.

The following scripts do not use SSL verification for interacting with the REST API and are provided here only as a demonstration.

4.3.1. Creating Objects Using Python

The following script connects to the Red Hat Satellite 6 API and creates a new organization, and then creates three environments in the new organization. If the organization already exists, the script uses that organization. If any of the environments already exist in the organization, the script raises an error and quits.

```

#!/usr/bin/python

import json
import sys

try:
    import requests
except ImportError:
    print "Please install the python-requests module."
    sys.exit(-1)

# URL to your Satellite 6 server
URL = "https://satellite6.example.com"
# URL for the API to your deployed Satellite 6 server
SAT_API = "%s/katello/api/v2/" % URL
# Katello-specific API
KATELLO_API = "%s/katello/api/" % URL
POST_HEADERS = {'content-type': 'application/json'}
# Default credentials to login to Satellite 6
USERNAME = "admin"
PASSWORD = "changeme"
# Ignore SSL for now

```

```
SSL_VERIFY = False

# Name of the organization to be either created or used
ORG_NAME = "MyOrg"
# Name for life cycle environments to be either created or used
ENVIRONMENTS = ["Development", "Testing", "Production"]

def get_json(location):
    """
    Performs a GET using the passed URL location
    """

    r = requests.get(location, auth=(USERNAME, PASSWORD),
verify=SSL_VERIFY)

    return r.json()

def post_json(location, json_data):
    """
    Performs a POST and passes the data to the URL location
    """

    result = requests.post(
        location,
        data=json_data,
        auth=(USERNAME, PASSWORD),
        verify=SSL_VERIFY,
        headers=POST_HEADERS)

    return result.json()

def main():
    """
    Main routine that creates or re-uses an organization and
    life cycle environments. If life cycle environments already
    exist, exit out.
    """

    # Check if our organization already exists
    org = get_json(SAT_API + "organizations/" + ORG_NAME)

    # If our organization is not found, create it
    if org.get('error', None):
        org_id = post_json(
            SAT_API + "organizations/",
            json.dumps({"name": ORG_NAME}))["id"]
        print "Creating organization: \t" + ORG_NAME
    else:
        # Our organization exists, so let's grab it
        org_id = org['id']
        print "Organization '%s' exists." % ORG_NAME

    # Now, let's fetch all available life cycle environments for this
```

```

org...
    envs = get_json(
        SAT_API + "organizations/" + str(org_id) + "/environments/")

    # ... and add them to a dictionary, with respective 'Prior'
environment
    prior_env_id = 0
    env_list = {}
    for env in envs['results']:
        env_list[env['id']] = env['name']
        prior_env_id = env['id'] if env['name'] == "Library" else
prior_env_id

    # Exit the script if at least one life cycle environment already
exists
    if all(environment in env_list.values() for environment in
ENVIRONMENTS):
        print "ERROR: One of the Environments is not unique to
organization"
        sys.exit(-1)

    # Create life cycle environments
    for environment in ENVIRONMENTS:
        new_env_id = post_json(
            SAT_API + "organizations/" + str(org_id) + "/environments/",
            json.dumps(
                {
                    "name": environment,
                    "organization_id": org_id,
                    "prior": prior_env_id}
            ))["id"]

        print "Creating environment: \t" + environment
        prior_env_id = new_env_id

if __name__ == "__main__":
    main()

```

4.3.2. Running Queries Using Python

You can create and run a Python script to achieve the same results as those described in [Section 4.1, “API Examples Using Curl”](#). The following example script describes this approach. First, create an executable file called **sat6api.py** and then add the following content:

```

#!/usr/bin/python
import json
import sys
try:
    import requests
except ImportError:
    print "Please install the python-requests module."
    sys.exit(-1)

```

```
SAT_API = 'https://satellite6.example.com/api/v2/'
USERNAME = "admin"
PASSWORD = "password"
SSL_VERIFY = False # Ignore SSL for now

def get_json(url):
    # Performs a GET using the passed URL location
    r = requests.get(url, auth=(USERNAME, PASSWORD), verify=SSL_VERIFY)
    return r.json()

def get_results(url):
    jsn = get_json(url)
    if jsn.get('error'):
        print "Error: " + jsn['error']['message']
    else:
        if jsn.get('results'):
            return jsn['results']
        elif 'results' not in jsn:
            return jsn
        else:
            print "No results found"
    return None

def display_all_results(url):
    results = get_results(url)
    if results:
        print json.dumps(results, indent=4, sort_keys=True)

def display_info_for_hosts(url):
    hosts = get_results(url)
    if hosts:
        for host in hosts:
            print "ID: %-10d Name: %-30s IP: %-20s OS: %-30s" %
(host['id'], host['name'], host['ip'], host['operatingsystem_name'])

def main():
    host = 'satellite6.example.com'
    print "Displaying all info for host %s ..." % host
    display_all_results(SAT_API + 'hosts/' + host)

    print "Displaying all facts for host %s ..." % host
    display_all_results(SAT_API + 'hosts/%s/facts' % host)

    host_pattern = 'example'
    print "Displaying basic info for hosts matching pattern '%s'..." %
host_pattern
    display_info_for_hosts(SAT_API + 'hosts?search=' + host_pattern)

    environment = 'production'
    print "Displaying basic info for hosts in environment %s..." %
environment
    display_info_for_hosts(SAT_API + 'hosts?search=environment=' +
environment)

    model = 'RHEV Hypervisor'
    print "Displaying basic info for hosts with model name %s..." % model
```



```

        display_info_for_hosts(SAT_API + 'hosts?search=model="' + model + '"')
if __name__ == "__main__":
    main()

```

You can then run `./sat6api.py` from the command line to display the results.

4.4. USING EXTENDED SEARCHES

You can use the web UI to determine other search terms that you can use to construct your queries. Satellite 6 supports scoped search and tab completion to make this task easier.

For example, to search for hosts by their operating system, navigate to **Hosts** → **All Hosts** and click inside the **Search** text box to display a list of search terms. One of the search terms for operating systems is **os_description**, which you can use in your API query as follows:

```

$ curl -s -k -u sat_username:sat_password
https://satellite6.example.com/api/v2/hosts?
search=os_description=\"RHEL+Server+6.6\" | python -m json.tool
{
  ...
  "results": [
    {
      "name": "satellite6.example.com",
      "operatingsystem_id": 1,
      "operatingsystem_name": "RHEL Server 6.6",
      ...
    }
  ],
  "search": "os_description=\"RHEL Server 6.6\"",
}

```

4.5. USING SEARCHES WITH PAGINATION CONTROL

You can use the **per_page** and **page** pagination parameters to limit the search results returned by an API search query. The **per_page** parameter specifies the amount per page and the **page** parameter specifies which page, as calculated by the **per_page** parameter, to return.

The default number of items to return is set to 1000 when you do not specify any pagination parameters, but the **per_page** value has a default of 20 which applies when you specify the **page** parameter.

Example 4.6. Listing Content Views

This example shows listing Content Views returning the third page of 10 results per page:

```

$ curl -X GET --user sat_username:sat_password \
"https://satellite6.example.com/katello/api/content_views?
per_page=10&page=3"

```

Example 4.7. Listing Activation Keys

This example shows listing Activation Keys for organization with ID 1 returning the second page of 30 keys per page:

```
$ curl -X GET --user sat_username:sat_password \
  "https://satellite6.example.com/katello/api/activation_keys?
  organization_id=1&per_page=30&page=2"
```

To get multiple pages of results you can use a *for loop* structure.

Example 4.8. Returning Multiple Pages

This example returns pages 1 to 3 of Content Views with 5 results per page:

```
$ for i in `seq 1 3`; do curl -X GET --user sat_username:sat_password \
  "https://satellite6.example.com/katello/api/content_views?
  per_page=5&page=$i"; done
```

4.6. WORKING WITH LIFE CYCLE ENVIRONMENTS

As explained in the [Life Cycle Environments](#) section of the *Server Administration Guide*, Application life cycles are divided into *life cycle environments*, which represent each stage of the application life cycle. Life cycle environments are linked to form an *environment path*. To create linked Life Cycle Environments using the API, make use of the `prior_id` parameter.

You can find the built-in API reference for Life Cycle Environments at https://satellite6.example.com/apidoc/v2/lifecycle_environments.html. The API routes include `/katello/api/environments` and `/katello/api/organizations/:organization_id/environments`.

You can list all the current Life Cycle Environments on your Satellite, for the default organization **1**, as follows:

```
$ curl -H "Accept:application/json,version=2" \
  -H "Content-Type:application/json" -X GET \
  -u sat_user:sat_password -k \
  https://satellite6.example.com/katello/api/organizations/1/environments |
  python -m json.tool
```

A newly installed Satellite would have an output with a section similar to the following:

```
output omitted
  "description": null,
  "id": 1,
  "label": "Library",
  "library": true,
  "name": "Library",
  "organization": {
    "id": 1,
```

```

        "label": "Default_Organization",
        "name": "Default Organization"
    },
    "permissions": {
        "destroy_lifecycle_environments": false,
        "edit_lifecycle_environments": true,
        "promote_or_remove_content_views_to_environments": true,
        "view_lifecycle_environments": true
    },
    "prior": null,
    "successor": null,
    output truncated

```

In the procedure below the default Library environment, with ID **1**, is used as the starting point for creating Life Cycle Environments.

Procedure 4.3. Creating Linked Life Cycle Environments

1. Choose an existing Life Cycle Environment that you want to use as a starting point. List the environment using its ID, in this case the environment with ID **1**:

```

$ curl -X GET -s -k -u sat_user:sat_password \
https://satellite6.example.com/katello/api/environments/1 | python -
m json.tool
output omitted
    "id": 1,
    "label": "Library",
output omitted
    "prior": null,
    "successor": null,
output truncated

```

2. Create a new Life Cycle Environment using the **prior** option set to **1**:
 - a. Create a JSON file, for example, **life-cycle.json**, with the following contents:

```

{"organization_id":1,"label":"api-dev","name":"API
Development","prior":1}

```
 - b. Enter a command as follows:

```

$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" -X POST \
-u sat_user:sat_password -k \
-d @life-cycle.json \
https://satellite6.example.com/katello/api/environments \
| python -m json.tool
output omitted
    "description": null,
    "id": 2,
    "label": "api-dev",
    "library": false,
    "name": "API Development",
    "organization": {
        "id": 1,
        "label": "Default_Organization",

```

```

        "name": "Default Organization"
    },
    "permissions": {
        "destroy_lifecycle_environments": true,
        "edit_lifecycle_environments": true,
        "promote_or_remove_content_views_to_environments": true,
        "view_lifecycle_environments": true
    },
    "prior": {
        "id": 1,
        "name": "Library"
    },
    },
output truncated

```

In the command output you can see the ID for this Life Cycle Environment is **2**, and the Life Cycle Environment prior to this one is **1**. This signifies that Life Cycle Environment **1** and **2** are linked. The Life Cycle Environment ID**2** is used when creating a successor to this environment.

3. Create another Life Cycle Environment, using the **prior** option set to **2**:
 - a. Edit the previously created **life-cycle.json**, updating the **label**, **name**, and **prior** values: **{"organization_id":1,"label":"api-qa","name":"API QA","prior":2}**
 - b. Enter a command as follows:

```

$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" -X POST \
-u sat_user:sat_password -k \
-d @life-cycle.json \
https://satellite6.example.com/katello/api/environments \
| python -m json.tool
output omitted
"description": null,
"id": 3,
"label": "api-qa",
"library": false,
"name": "API QA",
"organization": {
    "id": 1,
    "label": "Default Organization",
    "name": "Default Organization"
},
"permissions": {
    "destroy_lifecycle_environments": true,
    "edit_lifecycle_environments": true,
    "promote_or_remove_content_views_to_environments": true,
    "view_lifecycle_environments": true
},
"prior": {
    "id": 2,
    "name": "API Development"
},
"successor": null,
output truncated

```

In the command output you can see the ID for this Life Cycle Environment is **3**, and the Life Cycle Environment prior to this one is **2**. This signifies that Life Cycle Environment **2** and **3** are linked.

Updating a Life Cycle Environment

A Life Cycle Environment can be updated using a PUT command. For example:

```
$ curl -H "Accept:application/json,version=2" \
-H "Content-Type:application/json" -X POST \
-u sat_user:sat_password -k \
-d '{"description":"Quality Acceptance Testing"}' \
https://satellite6.example.com/katello/api/environments/3 \
| python -m json.tool
    output omitted
{"description": "Quality Acceptance Testing",
 "id": 3,
 "label": "api-qa",
 "library": false,
 "name": "API QA",
 "organization": {
   "id": 1,
   "label": "Default_Organization",
   "name": "Default Organization"
 },
 "permissions": {
   "destroy_lifecycle_environments": true,
   "edit_lifecycle_environments": true,
   "promote_or_remove_content_views_to_environments": true,
   "view_lifecycle_environments": true
 },
 "prior": {
   "id": 2,
   "name": "API Development"
 },
 }
    output truncated
```

Deleting a Life Cycle Environment

A Life Cycle Environment can be deleted provided it has no successor. Therefore, delete them in reverse order using a command in the following format:

```
curl -X DELETE -s -k -u sat_user:sat_password
https://satellite6.example.com/katello/api/environments/:id
```

APPENDIX A. API RESPONSE CODES

The Red Hat Satellite 6 API provides HTTP response status codes for API calls. The following codes are common for all resources in the Satellite API.

Table A.1. API Response Codes

Response	Explanation
200 OK	For a successful request action: show, index, update, or delete (GET, PUT, DELETE requests).
201 Created	For a successful create action (POST request).
301 Moved Permanently	Redirect when Satellite is restricted to use HTTPS and HTTP is attempted.
400 Bad Request	A required parameter is missing or the search query has invalid syntax.
401 Unauthorized	Failed to authorize the user (for example, incorrect credentials).
403 Forbidden	The user has insufficient permissions to perform the action or read the resource, or the action is unsupported in general.
404 Not Found	The record with the given ID does not exist. It can appear in show and delete actions when the requested record does not exist; or in create, update and delete actions when one of the associated records does not exist.
409 Conflict	Could not delete the record due to existing dependencies (for example, host groups with hosts).
415 Unsupported Media Type	The content type of the HTTP request is not JSON.
422 Unprocessable Entity	Failed to create an entity due to some validation errors. Applies to create or update actions only.
500 Internal Server Error	Unexpected internal server error.
503 Service Unavailable	The server is not running.

APPENDIX B. API PERMISSIONS MATRIX

The Red Hat Satellite 6 API supports numerous actions, many of which require specific permissions. The following table lists the API permission names, the actions associated with those permissions, and the associated resource type.

Table B.1. API Permissions Matrix

Permission Name	Actions	Resource Type
view_activation_keys	<ul style="list-style-type: none"> katello/activation_keys/all katello/activation_keys/index katello/activation_keys/auto_complete_search katello/api/v2/activation_keys/index katello/api/v2/activation_keys/show katello/api/v2/activation_keys/available_host_collections katello/api/v2/activation_keys/available_releases katello/api/v2/activation_keys/product_content 	Katello::ActivationKey
create_activation_keys	<ul style="list-style-type: none"> katello/api/v2/activation_keys/create katello/api/v2/activation_keys/copy 	Katello::ActivationKey
edit_activation_keys	<ul style="list-style-type: none"> katello/api/v2/activation_keys/update katello/api/v2/activation_keys/content_override katello/api/v2/activation_keys/add_subscriptions katello/api/v2/activation_keys/remove_subscriptions 	Katello::ActivationKey
destroy_activation_keys	<ul style="list-style-type: none"> katello/api/v2/activation_keys/destroy 	Katello::ActivationKey
logout	<ul style="list-style-type: none"> users/logout 	

Permission Name	Actions	Resource Type
view_architectures	<ul style="list-style-type: none">• architectures/index• architectures/show• architectures/auto_complete_search• api/v1/architectures/index• api/v1/architectures/show• api/v2/architectures/index• api/v2/architectures/show	
create_architectures	<ul style="list-style-type: none">• architectures/new• architectures/create• api/v1/architectures/create• api/v2/architectures/create	
edit_architectures	<ul style="list-style-type: none">• architectures/edit• architectures/update• api/v1/architectures/update• api/v2/architectures/update	
destroy_architectures	<ul style="list-style-type: none">• architectures/destroy• api/v1/architectures/destroy• api/v2/architectures/destroy	
view_audit_logs	<ul style="list-style-type: none">• audits/index• audits/show• audits/auto_complete_search• api/v1/audits/index• api/v1/audits/show• api/v2/audits/index• api/v2/audits/show	

Permission Name	Actions	Resource Type
view_authenticators	<ul style="list-style-type: none"> • auth_source_ldaps/index • auth_source_ldaps/show • api/v1/auth_source_ldaps/index • api/v1/auth_source_ldaps/show • api/v2/auth_source_ldaps/index • api/v2/auth_source_ldaps/show 	
create_authenticators	<ul style="list-style-type: none"> • auth_source_ldaps/new • auth_source_ldaps/create • api/v1/auth_source_ldaps/create • api/v2/auth_source_ldaps/create 	
edit_authenticators	<ul style="list-style-type: none"> • auth_source_ldaps/edit • auth_source_ldaps/update • api/v1/auth_source_ldaps/update • api/v2/auth_source_ldaps/update 	
destroy_authenticators	<ul style="list-style-type: none"> • auth_source_ldaps/destroy • api/v1/auth_source_ldaps/destroy • api/v2/auth_source_ldaps/destroy 	
view_bookmarks	<ul style="list-style-type: none"> • bookmarks/index • bookmarks/show • api/v1/bookmarks/index • api/v1/bookmarks/show • api/v2/bookmarks/index • api/v2/bookmarks/show 	

Permission Name	Actions	Resource Type
create_bookmarks	<ul style="list-style-type: none">bookmarks/newbookmarks/createapi/v1/bookmarks/newapi/v1/bookmarks/createapi/v2/bookmarks/newapi/v2/bookmarks/create	
edit_bookmarks	<ul style="list-style-type: none">bookmarks/editbookmarks/updateapi/v1/bookmarks/editapi/v1/bookmarks/updateapi/v2/bookmarks/editapi/v2/bookmarks/update	
destroy_bookmarks	<ul style="list-style-type: none">bookmarks/destroyapi/v1/bookmarks/destroyapi/v2/bookmarks/destroy	
download_bootdisk	<ul style="list-style-type: none">foreman_bootdisk/disks/genericforeman_bootdisk/disks/hostforeman_bootdisk/disks/full_hostforeman_bootdisk/disks/subnetforeman_bootdisk/disks/helpforeman_bootdisk/api/v2/disks/genericforeman_bootdisk/api/v2/disks/host	

Permission Name	Actions	Resource Type
manage_capsule_content	<ul style="list-style-type: none"> katello/api/v2/capsule_content/lifecycle_environments katello/api/v2/capsule_content/available_lifecycle_environments katello/api/v2/capsule_content/add_lifecycle_environment katello/api/v2/capsule_content/remove_lifecycle_environment katello/api/v2/capsule_content/sync katello/api/v2/capsule_content/sync_status katello/api/v2/capsule_content/cancel_sync 	SmartProxy
view_capsule_content	<ul style="list-style-type: none"> smart_proxies/pulp_storage smart_proxies/pulp_status smart_proxies/show_with_content 	SmartProxy
view_compute_profiles	<ul style="list-style-type: none"> compute_profiles/index compute_profiles/show compute_profiles/auto_complete_search api/v2/compute_profiles/index api/v2/compute_profiles/show 	
create_compute_profiles	<ul style="list-style-type: none"> compute_profiles/new compute_profiles/create api/v2/compute_profiles/create 	
edit_compute_profiles	<ul style="list-style-type: none"> compute_profiles/edit compute_profiles/update api/v2/compute_profiles/update 	

Permission Name	Actions	Resource Type
destroy_compute_profiles	<ul style="list-style-type: none"> • compute_profiles/destroy • api/v2/compute_profiles/destroy 	
view_compute_resources	<ul style="list-style-type: none"> • compute_resources/index • compute_resources/show • compute_resources/auto_complete_search • compute_resources/ping • compute_resources/available_images • api/v1/compute_resources/index • api/v1/compute_resources/show • api/v2/compute_resources/index • api/v2/compute_resources/show • api/v2/compute_resources/available_images • api/v2/compute_resources/available_clusters • api/v2/compute_resources/available_folders • api/v2/compute_resources/available flavors • api/v2/compute_resources/available_networks • api/v2/compute_resources/available_resource_pools • api/v2/compute_resources/available_security_groups • api/v2/compute_resources/available_storage_domains • api/v2/compute_resources/available_zones • api/v2/compute_resources/available_storage_pods 	

Permission Name	Actions	Resource Type
create_compute_resources	<ul style="list-style-type: none"> • compute_resources/new • compute_resources/create • compute_resources/test_connection • api/v1/compute_resources/create • api/v2/compute_resources/create 	
edit_compute_resources	<ul style="list-style-type: none"> • compute_resources/edit • compute_resources/update • compute_resources/test_connection • compute_attributes/new • compute_attributes/create • compute_attributes/edit • compute_attributes/update • api/v1/compute_resources/update • api/v2/compute_resources/update • api/v2/compute_attributes/create • api/v2/compute_attributes/update 	
destroy_compute_resources	<ul style="list-style-type: none"> • compute_resources/destroy • api/v1/compute_resources/destroy • api/v2/compute_resources/destroy 	
view_compute_resources_vms	<ul style="list-style-type: none"> • compute_resources_vms/index • compute_resources_vms/show 	
create_compute_resources_vms	<ul style="list-style-type: none"> • compute_resources_vms/new • compute_resources_vms/create 	

Permission Name	Actions	Resource Type
edit_compute_resources_vms	<ul style="list-style-type: none"> compute_resources_vms/edit compute_resources_vms/update 	
destroy_compute_resources_vms	<ul style="list-style-type: none"> compute_resources_vms/destroy 	
power_compute_resources_vms	<ul style="list-style-type: none"> compute_resources_vms/power compute_resources_vms/pause 	
console_compute_resources_vms	<ul style="list-style-type: none"> compute_resources_vms/console 	
view_config_groups	<ul style="list-style-type: none"> config_groups/index config_groups/auto_complete_search api/v2/config_groups/index api/v2/config_groups/show 	
create_config_groups	<ul style="list-style-type: none"> config_groups/new config_groups/create api/v2/config_groups/create 	
edit_config_groups	<ul style="list-style-type: none"> config_groups/edit config_groups/update api/v2/config_groups/update 	
destroy_config_groups	<ul style="list-style-type: none"> config_groups/destroy api/v2/config_groups/destroy 	

Permission Name	Actions	Resource Type
view_config_reports	<ul style="list-style-type: none"> • api/v1/reports/index • api/v1/reports/show • api/v1/reports/last • api/v2/reports/index • api/v2/reports/show • api/v2/reports/last • config_reports/index • config_reports/show • config_reports/auto_complete_search • api/v2/config_reports/index • api/v2/config_reports/show • api/v2/config_reports/last 	
destroy_config_reports	<ul style="list-style-type: none"> • config_reports/destroy • api/v1/reports/destroy • api/v2/reports/destroy • api/v2/config_reports/destroy 	
upload_config_reports	<ul style="list-style-type: none"> • api/v2/reports/create • api/v2/config_reports/create 	
view_containers	<ul style="list-style-type: none"> • containers/index • containers/show • api/v2/containers/index • api/v2/containers/show • api/v2/containers/logs 	Container
commit_containers	<ul style="list-style-type: none"> • containers/commit 	Container

Permission Name	Actions	Resource Type
create_containers	<ul style="list-style-type: none"> containers/steps/show containers/steps/update containers/new api/v2/containers/create api/v2/containers/power 	Container
destroy_containers	<ul style="list-style-type: none"> containers/destroy api/v2/containers/destroy 	Container
power_compute_resources_vms	<ul style="list-style-type: none"> containers/power api/v2/containers/create api/v2/containers/power 	ComputeResource
view_content_hosts	<ul style="list-style-type: none"> katello/content_hosts/auto_complete_search katello/api/v2/systems/index katello/api/v2/systems/show katello/api/v2/systems/errata katello/api/v2/systems/package_profile katello/api/v2/systems/product_content katello/api/v2/systems/report katello/api/v2/systems/releases katello/api/v2/systems/available_host_collections katello/api/v2/host_collections/systems 	Katello::System
create_content_hosts	<ul style="list-style-type: none"> katello/api/v2/systems/create katello/api/rhsm/candlepin_proxies/consumer_create katello/api/rhsm/candlepin_proxies/consumer_show 	Katello::System

Permission Name	Actions	Resource Type
edit_content_hosts	<ul style="list-style-type: none"> • katello/api/v2/systems/update • katello/api/v2/systems/content_override • katello/api/rhsm/candlepin_proxies/upload_package_profile • katello/api/rhsm/candlepin_proxies/regenerate_identity_certificates • katello/api/rhsm/candlepin_proxies/hypervisors_update 	Katello::System
destroy_content_hosts	<ul style="list-style-type: none"> • katello/api/v2/systems/destroy 	Katello::System
view_content_views	<ul style="list-style-type: none"> • katello/api/v2/content_views/index • katello/api/v2/content_views/show • katello/api/v2/content_views/available_puppet_modules • katello/api/v2/content_views/available_puppet_module_names • katello/api/v2/content_view_filters/index • katello/api/v2/content_view_filters/show • katello/api/v2/content_view_filter_rules/index • katello/api/v2/content_view_filter_rules/show • katello/api/v2/content_view_histories/index • katello/api/v2/content_view_puppet_modules/index • katello/api/v2/content_view_puppet_modules/show • katello/api/v2/content_view_versions/index • katello/api/v2/content_view_versions/show • katello/api/v2/package_groups/index • katello/api/v2/package_groups/show 	Katello::ContentView

Permission Name	Actions	Resource Type
	<ul style="list-style-type: none"> • katello/api/v2/errata/index • katello/api/v2/errata/show • katello/api/v2/puppet_modules/index • katello/api/v2/puppet_modules/show • katello/content_views/auto_complete • katello/content_views/auto_complete_s earch • katello/errata/short_details • katello/errata/auto_complete • katello/packages/details • katello/packages/auto_complete • katello/products/auto_complete • katello/repositories/auto_complete_libr ary • katello/content_search/index • katello/content_search/products • katello/content_search/repos • katello/content_search/packages • katello/content_search/errata • katello/content_search/puppet_module s • katello/content_search/packages_items • katello/content_search/errata_items • katello/content_search/puppet_module s_items • katello/content_search/view_packages • katello/content_search/view_puppet_m odules • katello/content_search/repo_packages • katello/content_search/repo_errata • katello/content_search/repo_puppet_m odules • katello/content_search/repo_compare_ errata • katello/content_search/repo_compare_ packages • katello/content_search/repo_compare_ puppet_modules 	

Permission Name	Actions	Resource Type
	<ul style="list-style-type: none"> • katello/content_search/view_compare_errata • katello/content_search/view_compare_packages • katello/content_search/view_compare_puppet_modules • katello/content_search/views 	
create_content_views	<ul style="list-style-type: none"> • katello/api/v2/content_views/create • katello/api/v2/content_views/copy 	Katello::ContentView
edit_content_views	<ul style="list-style-type: none"> • katello/api/v2/content_views/update • katello/api/v2/content_view_filters/create • katello/api/v2/content_view_filters/update • katello/api/v2/content_view_filters/destroy • katello/api/v2/content_view_filter_rules/create • katello/api/v2/content_view_filter_rules/update • katello/api/v2/content_view_filter_rules/destroy • katello/api/v2/content_view_puppet_modules/create • katello/api/v2/content_view_puppet_modules/update • katello/api/v2/content_view_puppet_modules/destroy 	Katello::ContentView
destroy_content_views	<ul style="list-style-type: none"> • katello/api/v2/content_views/destroy • katello/api/v2/content_views/remove • katello/api/v2/content_view_versions/destroy 	Katello::ContentView

Permission Name	Actions	Resource Type
publish_content_views	<ul style="list-style-type: none"> katello/api/v2/content_views/publish katello/api/v2/content_view_versions/incremental_update 	Katello::ContentView
promote_or_remove_content_views	<ul style="list-style-type: none"> katello/api/v2/content_view_versions/promote katello/api/v2/content_views/remove_from_environment katello/api/v2/content_views/remove 	Katello::ContentView
export_content_views	<ul style="list-style-type: none"> katello/api/v2/content_view_versions/export 	Katello::ContentView
access_dashboard	<ul style="list-style-type: none"> dashboard/index dashboard/save_positions dashboard/reset_default dashboard/create dashboard/destroy api/v1/dashboard/index api/v2/dashboard/index 	
view_discovered_hosts	<ul style="list-style-type: none"> discovered_hosts/index discovered_hosts/show discovered_hosts/auto_complete_search api/v2/discovered_hosts/index api/v2/discovered_hosts/show 	Host
submit_discovered_hosts	<ul style="list-style-type: none"> api/v2/discovered_hosts/facts api/v2/discovered_hosts/create 	Host

Permission Name	Actions	Resource Type
auto_provision_discovered_hosts	<ul style="list-style-type: none"> discovered_hosts/auto_provision discovered_hosts/auto_provision_all api/v2/discovered_hosts/auto_provision api/v2/discovered_hosts/auto_provision_all 	Host
provision_discovered_hosts	<ul style="list-style-type: none"> discovered_hosts/edit discovered_hosts/update api/v2/discovered_hosts/update 	Host
edit_discovered_hosts	<ul style="list-style-type: none"> discovered_hosts/update_multiple_location discovered_hosts/select_multiple_organization discovered_hosts/update_multiple_organization discovered_hosts/select_multiple_location discovered_hosts/refresh_facts discovered_hosts/reboot discovered_hosts/reboot_all api/v2/discovered_hosts/refresh_facts api/v2/discovered_hosts/reboot api/v2/discovered_hosts/reboot_all 	Host
destroy_discovered_hosts	<ul style="list-style-type: none"> discovered_hosts/destroy discovered_hosts/submit_multiple_destroy discovered_hosts/multiple_destroy api/v2/discovered_hosts/destroy 	Host

Permission Name	Actions	Resource Type
view_discovery_rules	<ul style="list-style-type: none"> discovery_rules/index discovery_rules/show discovery_rules/auto_complete_search api/v2/discovery_rules/index api/v2/discovery_rules/show 	DiscoveryRule
create_discovery_rules	<ul style="list-style-type: none"> discovery_rules/new discovery_rules/create api/v2/discovery_rules/create 	DiscoveryRule
edit_discovery_rules	<ul style="list-style-type: none"> discovery_rules/edit discovery_rules/update discovery_rules/enable discovery_rules/disable api/v2/discovery_rules/create api/v2/discovery_rules/update 	DiscoveryRule
execute_discovery_rules	<ul style="list-style-type: none"> discovery_rules/auto_provision discovery_rules/auto_provision_all api/v2/discovery_rules/auto_provision api/v2/discovery_rules/auto_provision_all 	DiscoveryRule
destroy_discovery_rules	<ul style="list-style-type: none"> discovery_rules/destroy api/v2/discovery_rules/destroy 	DiscoveryRule

Permission Name	Actions	Resource Type
view_domains	<ul style="list-style-type: none"> • domains/index • domains/show • domains/auto_complete_search • api/v1/domains/index • api/v1/domains/show • api/v2/domains/index • api/v2/domains/show • api/v2/parameters/index • api/v2/parameters/show 	
create_domains	<ul style="list-style-type: none"> • domains/new • domains/create • api/v1/domains/create • api/v2/domains/create 	
edit_domains	<ul style="list-style-type: none"> • domains/edit • domains/update • api/v1/domains/update • api/v2/domains/update • api/v2/parameters/create • api/v2/parameters/update • api/v2/parameters/destroy • api/v2/parameters/reset 	
destroy_domains	<ul style="list-style-type: none"> • domains/destroy • api/v1/domains/destroy • api/v2/domains/destroy 	

Permission Name	Actions	Resource Type
view_environments	<ul style="list-style-type: none">• environments/index• environments/show• environments/auto_complete_search• api/v1/environments/index• api/v1/environments/show• api/v2/environments/index• api/v2/environments/show	
create_environments	<ul style="list-style-type: none">• environments/new• environments/create• api/v1/environments/create• api/v2/environments/create	
edit_environments	<ul style="list-style-type: none">• environments/edit• environments/update• api/v1/environments/update• api/v2/environments/update	
destroy_environments	<ul style="list-style-type: none">• environments/destroy• api/v1/environments/destroy• api/v2/environments/destroy	

Permission Name	Actions	Resource Type
import_environments	<ul style="list-style-type: none"> environments/import_environments environments/obsolete_and_new api/v1/environments/import_puppetclasses api/v2/environments/import_puppetclasses api/v1/smart_proxies/import_puppetclasses api/v2/smart_proxies/import_puppetclasses 	
view_external_usergroups	<ul style="list-style-type: none"> external_usergroups/index external_usergroups/show api/v2/external_usergroups/index api/v2/external_usergroups/show 	
create_external_usergroups	<ul style="list-style-type: none"> external_usergroups/new external_usergroups/create api/v2/external_usergroups/new api/v2/external_usergroups/create 	
edit_external_usergroups	<ul style="list-style-type: none"> external_usergroups/edit external_usergroups/update external_usergroups/refresh api/v2/external_usergroups/update api/v2/external_usergroups/refresh 	
destroy_external_usergroups	<ul style="list-style-type: none"> external_usergroups/destroy api/v2/external_usergroups/destroy 	

Permission Name	Actions	Resource Type
view_external_variables	<ul style="list-style-type: none">• lookup_keys/index• lookup_keys/show• lookup_keys/auto_complete_search• puppetclass_lookup_keys/index• puppetclass_lookup_keys/show• puppetclass_lookup_keys/auto_complete_search• variable_lookup_keys/index• variable_lookup_keys/show• variable_lookup_keys/auto_complete_search• lookup_values/index• api/v1/lookup_keys/index• api/v1/lookup_keys/show• api/v2/smart_variables/index• api/v2/smart_variables/show• api/v2/smart_class_parameters/index• api/v2/smart_class_parameters/show• api/v2/override_values/index• api/v2/override_values/show	

Permission Name	Actions	Resource Type
create_external_variables	<ul style="list-style-type: none"> • lookup_keys/new • lookup_keys/create • puppetclass_lookup_keys/new • puppetclass_lookup_keys/create • variable_lookup_keys/new • variable_lookup_keys/create • lookup_values/create • api/v1/lookup_keys/create • api/v2/smart_variables/create • api/v2/smart_class_parameters/create • api/v2/override_values/create 	
edit_external_variables	<ul style="list-style-type: none"> • lookup_keys/edit • lookup_keys/update • puppetclass_lookup_keys/edit • puppetclass_lookup_keys/update • variable_lookup_keys/edit • variable_lookup_keys/update • lookup_values/create • lookup_values/update • lookup_values/destroy • api/v1/lookup_keys/update • api/v2/smart_variables/update • api/v2/smart_class_parameters/update • api/v2/override_values/create • api/v2/override_values/update • api/v2/override_values/destroy 	

Permission Name	Actions	Resource Type
destroy_external_variables	<ul style="list-style-type: none"> lookup_keys/destroy puppetclass_lookup_keys/destroy variable_lookup_keys/destroy lookup_values/destroy api/v1/lookup_keys/destroy api/v2/smart_variables/destroy api/v2/smart_class_parameters/destroy api/v2/override_values/create api/v2/override_values/update api/v2/override_values/destroy 	
view_facts	<ul style="list-style-type: none"> facts/index facts/show fact_values/index fact_values/show fact_values/auto_complete_search api/v1/fact_values/index api/v1/fact_values/show api/v2/fact_values/index api/v2/fact_values/show 	
upload_facts	<ul style="list-style-type: none"> api/v2/hosts/facts 	
view_filters	<ul style="list-style-type: none"> filters/index filters/auto_complete_search api/v2/filters/index api/v2/filters/show 	

Permission Name	Actions	Resource Type
create_filters	<ul style="list-style-type: none"> filters/new filters/create api/v2/filters/create 	
edit_filters	<ul style="list-style-type: none"> filters/edit filters/update permissions/index api/v2/filters/update api/v2/permissions/index api/v2/permissions/show api/v2/permissions/resource_types 	
destroy_filters	<ul style="list-style-type: none"> filters/destroy api/v2/filters/destroy 	
view_arf_reports	<ul style="list-style-type: none"> arf_reports/index arf_reports/show arf_reports/parse_html arf_reports/show_html arf_reports/parse_bzip arf_reports/auto_complete_search api/v2/compliance/arf_reports/index api/v2/compliance/arf_reports/show compliance_hosts/show 	
destroy_arf_reports	<ul style="list-style-type: none"> arf_reports/destroy arf_reports/delete_multiple arf_reports/submit_delete_multiple api/v2/compliance/arf_reports/destroy 	

Permission Name	Actions	Resource Type
create_arf_reports	<ul style="list-style-type: none"> • api/v2/compliance/arf_reports/create 	
view_policies	<ul style="list-style-type: none"> • policies/index • policies/show • policies/parse • policies/auto_complete_search • policy_dashboard/index • compliance_dashboard/index • api/v2/compliance/policies/index • api/v2/compliance/policies/show • api/v2/compliance/policies/content 	ForemanOpenscap::Policy
edit_policies	<ul style="list-style-type: none"> • policies/edit • policies/update • policies/scap_content_selected • api/v2/compliance/policies/update 	ForemanOpenscap::Policy
create_policies	<ul style="list-style-type: none"> • policies/new • policies/create • api/v2/compliance/policies/create 	ForemanOpenscap::Policy
destroy_policies	<ul style="list-style-type: none"> • policies/destroy • api/v2/compliance/policies/destroy 	ForemanOpenscap::Policy
assign_policies	<ul style="list-style-type: none"> • policies/select_multiple_hosts • policies/update_multiple_hosts • policies/disassociate_multiple_hosts • policies/remove_policy_from_multiple_hosts 	ForemanOpenscap::Policy

Permission Name	Actions	Resource Type
view_scap_contents	<ul style="list-style-type: none"> scap_contents/index scap_contents/show scap_contents/auto_complete_search api/v2/compliance/scap_contents/index api/v2/compliance/scap_contents/show 	ForemanOpenscap::ScapContent
view_scap_contents	<ul style="list-style-type: none"> scap_contents/index scap_contents/show scap_contents/auto_complete_search api/v2/compliance/scap_contents/index api/v2/compliance/scap_contents/show 	ForemanOpenscap::ScapContent
edit_scap_contents	<ul style="list-style-type: none"> scap_contents/edit scap_contents/update api/v2/compliance/scap_contents/update 	ForemanOpenscap::ScapContent
create_scap_contents	<ul style="list-style-type: none"> scap_contents/new scap_contents/create api/v2/compliance/scap_contents/create 	ForemanOpenscap::ScapContent
destroy_scap_contents	<ul style="list-style-type: none"> scap_contents/destroy api/v2/compliance/scap_contents/destroy 	ForemanOpenscap::ScapContent
edit_hosts	<ul style="list-style-type: none"> hosts/openscap_proxy_changed 	Host
edit_hostgroups	<ul style="list-style-type: none"> hostgroups/openscap_proxy_changed 	Host

Permission Name	Actions	Resource Type
view_job_templates	<ul style="list-style-type: none">• job_templates/index• job_templates/show• job_templates/revision• job_templates/auto_complete_search• job_templates/auto_complete_job_category• job_templates/preview• job_templates/export• api/v2/job_templates/index• api/v2/job_templates/show• api/v2/job_templates/revision• api/v2/job_templates/export• api/v2/template_inputs/index• api/v2/template_inputs/show• api/v2/foreign_input_sets/index• api/v2/foreign_input_sets/show	JobTemplate
create_job_templates	<ul style="list-style-type: none">• job_templates/new• job_templates/create• job_templates/clone_template• job_templates/import• api/v2/job_templates/create• api/v2/job_templates/clone• api/v2/job_templates/import	JobTemplate

Permission Name	Actions	Resource Type
edit_job_templates	<ul style="list-style-type: none"> • job_templates/edit • job_templates/update • api/v2/job_templates/update • api/v2/template_inputs/create • api/v2/template_inputs/update • api/v2/template_inputs/destroy • api/v2/foreign_input_sets/create • api/v2/foreign_input_sets/update • api/v2/foreign_input_sets/destroy 	
edit_job_templates	<ul style="list-style-type: none"> • job_templates/edit • job_templates/update • api/v2/job_templates/update • api/v2/template_inputs/create • api/v2/template_inputs/update • api/v2/template_inputs/destroy • api/v2/foreign_input_sets/create • api/v2/foreign_input_sets/update • api/v2/foreign_input_sets/destroy 	
edit_remote_execution_features	<ul style="list-style-type: none"> • remote_execution_features/index • remote_execution_features/show • remote_execution_features/update • api/v2/remote_execution_features/index • api/v2/remote_execution_features/show • api/v2/remote_execution_features/update 	RemoteExecutionFeature

Permission Name	Actions	Resource Type
destroy_job_templates	<ul style="list-style-type: none"> job_templates/destroy api/v2/job_templates/destroy 	JobTemplate
lock_job_templates	<ul style="list-style-type: none"> job_templates/lock job_templates/unlock 	JobTemplate
create_job_invocations	<ul style="list-style-type: none"> job_invocations/new job_invocations/create job_invocations/refresh job_invocations/rerun job_invocations/preview_hosts api/v2/job_invocations/create 	JobInvocation
view_job_invocations	<ul style="list-style-type: none"> job_invocations/index job_invocations/show template_invocations/show api/v2/job_invocations/index api/v2/job_invocations/show api/v2/job_invocations/output 	JobInvocation
execute_template_invocation		TemplateInvocation
filter_autocompletion_for_template_invocation	<ul style="list-style-type: none"> template_invocations/auto_complete_search job_invocations/show template_invocations/index 	TemplateInvocation

Permission Name	Actions	Resource Type
view_foreman_tasks	<ul style="list-style-type: none"> • foreman_tasks/tasks/auto_complete_search • foreman_tasks/tasks/sub_tasks • foreman_tasks/tasks/index • foreman_tasks/tasks/show • foreman_tasks/api/tasks/bulk_search • foreman_tasks/api/tasks/show • foreman_tasks/api/tasks/index • foreman_tasks/api/tasks/summary 	ForemanTasks::Task
edit_foreman_tasks	<ul style="list-style-type: none"> • foreman_tasks/tasks/resume • foreman_tasks/tasks/unlock • foreman_tasks/tasks/force_unlock • foreman_tasks/tasks/cancel_step • foreman_tasks/api/tasks/bulk_resume 	ForemanTasks::Task
create_recurring_logics		ForemanTasks::RecurringLogic
view_recurring_logics	<ul style="list-style-type: none"> • foreman_tasks/recurring_logics/index • foreman_tasks/recurring_logics/show • foreman_tasks/api/recurring_logics/index • foreman_tasks/api/recurring_logics/show 	ForemanTasks::RecurringLogic
edit_recurring_logics	<ul style="list-style-type: none"> • foreman_tasks/recurring_logics/cancel • foreman_tasks/api/recurring_logics/cancel 	ForemanTasks::RecurringLogic

Permission Name	Actions	Resource Type
view_globals	<ul style="list-style-type: none"> • common_parameters/index • common_parameters/show • common_parameters/auto_complete_search • api/v1/common_parameters/index • api/v1/common_parameters/show • api/v2/common_parameters/index • api/v2/common_parameters/show 	
create_globals	<ul style="list-style-type: none"> • common_parameters/new • common_parameters/create • api/v1/common_parameters/create • api/v2/common_parameters/create 	
edit_globals	<ul style="list-style-type: none"> • common_parameters/edit • common_parameters/update • api/v1/common_parameters/update • api/v2/common_parameters/update 	
destroy_globals	<ul style="list-style-type: none"> • common_parameters/destroy • api/v1/common_parameters/destroy • api/v2/common_parameters/destroy 	
view_gpg_keys	<ul style="list-style-type: none"> • katello/gpg_keys/all • katello/gpg_keys/index • katello/gpg_keys/auto_complete_search • katello/api/v2/gpg_keys/index • katello/api/v2/gpg_keys/show 	Katello::GpgKey

Permission Name	Actions	Resource Type
create_gpg_keys	<ul style="list-style-type: none"> katello/api/v2/gpg_keys/create 	Katello::GpgKey
edit_gpg_keys	<ul style="list-style-type: none"> katello/api/v2/gpg_keys/update katello/api/v2/gpg_keys/content 	Katello::GpgKey
destroy_gpg_keys	<ul style="list-style-type: none"> katello/api/v2/gpg_keys/destroy 	Katello::GpgKey
view_host_collections	<ul style="list-style-type: none"> katello/api/v2/host_collections/index katello/api/v2/host_collections/show katello/host_collections/auto_complete_search 	Katello::HostCollection
create_host_collections	<ul style="list-style-type: none"> katello/api/v2/host_collections/create katello/api/v2/host_collections/copy 	Katello::HostCollection
edit_host_collections	<ul style="list-style-type: none"> katello/api/v2/host_collections/update katello/api/v2/host_collections/add_systems katello/api/v2/host_collections/remove_systems 	Katello::HostCollection
destroy_host_collections	<ul style="list-style-type: none"> katello/api/v2/host_collections/destroy 	Katello::HostCollection
edit_classes	<ul style="list-style-type: none"> host_editing/edit_classes api/v2/host_classes/index api/v2/host_classes/create api/v2/host_classes/destroy 	
create_params	<ul style="list-style-type: none"> host_editing/create_params api/v2/parameters/create 	

Permission Name	Actions	Resource Type
edit_params	<ul style="list-style-type: none">• host_editing/edit_params• api/v2/parameters/update	
destroy_params	<ul style="list-style-type: none">• host_editing/destroy_params• api/v2/parameters/destroy• api/v2/parameters/reset	
view_hostgroups	<ul style="list-style-type: none">• hostgroups/index• hostgroups/show• hostgroups/auto_complete_search• api/v1/hostgroups/index• api/v1/hostgroups/show• api/v2/hostgroups/index• api/v2/hostgroups/show	

Permission Name	Actions	Resource Type
create_hostgroups	<ul style="list-style-type: none"> • hostgroups/new • hostgroups/create • hostgroups/clone • hostgroups/nest • hostgroups/process_hostgroup • hostgroups/architecture_selected • hostgroups/domain_selected • hostgroups/environment_selected • hostgroups/medium_selected • hostgroups/os_selected • hostgroups/use_image_selected • hostgroups/process_hostgroup • hostgroups/puppetclass_parameters • host/process_hostgroup • puppetclasses/parameters • api/v1/hostgroups/create • api/v1/hostgroups/clone • api/v2/hostgroups/create • api/v2/hostgroups/clone 	

Permission Name	Actions	Resource Type
edit_hostgroups	<ul style="list-style-type: none"> • hostgroups/edit • hostgroups/update • hostgroups/architecture_selected • hostgroups/process_hostgroup • hostgroups/architecture_selected • hostgroups/domain_selected • hostgroups/environment_selected • hostgroups/medium_selected • hostgroups/os_selected • hostgroups/use_image_selected • hostgroups/process_hostgroup • hostgroups/puppetclass_parameters • host/process_hostgroup • puppetclasses/parameters • api/v1/hostgroups/update • api/v2/hostgroups/update • api/v2/parameters/create • api/v2/parameters/update • api/v2/parameters/destroy • api/v2/parameters/reset • api/v2/hostgroup_classes/index • api/v2/hostgroup_classes/create • api/v2/hostgroup_classes/destroy 	
destroy_hostgroups	<ul style="list-style-type: none"> • hostgroups/destroy • api/v1/hostgroups/destroy • api/v2/hostgroups/destroy 	
view_hosts	<ul style="list-style-type: none"> • hosts/index • hosts/show • hosts/errors 	

Permission Name	Actions	Resource Type
	<ul style="list-style-type: none"> • hosts/active • hosts/out_of_sync • hosts/disabled • hosts/pending • hosts/vm • hosts/externalNodes • hosts/pxe_config • hosts/storeconfig_klasses • hosts/auto_complete_search • hosts/bmc • hosts/runtime • hosts/resources • hosts/templates • hosts/overview • hosts/nics • dashboard/OutOfSync • dashboard/errors • dashboard/active • unattended/host_template • unattended/hostgroup_template • api/v1/hosts/index • api/v1/hosts/show • api/v1/hosts/status • api/v2/hosts/index • api/v2/hosts/show • api/v2/hosts/status • api/v2/hosts/get_status • api/v2/hosts/vm_compute_attributes • api/v2/hosts/template • api/v2/interfaces/index • api/v2/interfaces/show • locations/mismatches • organizations/mismatches 	

Permission Name	Actions	Resource Type
	<ul style="list-style-type: none">• hosts/puppet_environment_for_content_view• katello/api/v2/host_autocomplete/auto_complete_search• katello/api/v2/host_errata/index• katello/api/v2/host_errata/show• katello/api/v2/host_errata/auto_complete_search• katello/api/v2/host_subscriptions/index• katello/api/v2/host_subscriptions/events• katello/api/v2/host_subscriptions/product_content• katello/api/v2/hosts_bulk_actions/installable_errata• katello/api/v2/hosts_bulk_actions/available_incremental_updates• katello/api/v2/host_packages/index	

Permission Name	Actions	Resource Type
create_hosts	<ul style="list-style-type: none"> • hosts/new • hosts/create • hosts/clone • hosts/architecture_selected • hosts/compute_resource_selected • hosts/domain_selected • hosts/environment_selected • hosts/hostgroup_or_environment_selected • hosts/medium_selected • hosts/os_selected • hosts/use_image_selected • hosts/process_hostgroup • hosts/process_taxonomy • hosts/current_parameters • hosts/puppetclass_parameters • hosts/template_used • hosts/interfaces • compute_resources/cluster_selected • compute_resources/template_selected • compute_resources/provider_selected • compute_resources/resource_pools • puppetclasses/parameters • subnets/freeip • interfaces/new • api/v1/hosts/create • api/v2/hosts/create • api/v2/interfaces/create • api/v2/tasks/index 	
edit_hosts	<ul style="list-style-type: none"> • hosts/edit • hosts/update 	

Permission Name	Actions	Resource Type
	<ul style="list-style-type: none"> • hosts/multiple_actions • hosts/reset_multiple • hosts/submit_multiple_enable • hosts/select_multiple_hostgroup • hosts/select_multiple_environment • hosts/submit_multiple_disable • hosts/multiple_parameters • hosts/multiple_disable • hosts/multiple_enable • hosts/update_multiple_environment • hosts/update_multiple_hostgroup • hosts/update_multiple_parameters • hosts/toggle_manage • hosts/select_multiple_organization • hosts/update_multiple_organization • hosts/disassociate • hosts/multiple_disassociate • hosts/update_multiple_disassociate • hosts/select_multiple_owner • hosts/update_multiple_owner • hosts/select_multiple_power_state • hosts/update_multiple_power_state • hosts/select_multiple_puppet_proxy • hosts/update_multiple_puppet_proxy • hosts/select_multiple_puppet_ca_proxy • hosts/update_multiple_puppet_ca_proxy • hosts/select_multiple_location • hosts/update_multiple_location • hosts/architecture_selected • hosts/compute_resource_selected • hosts/domain_selected • hosts/environment_selected 	

Permission Name	Actions	Resource Type
	<ul style="list-style-type: none"> • hosts/hostgroup_or_environment_selected • hosts/medium_selected • hosts/os_selected • hosts/use_image_selected • hosts/process_hostgroup • hosts/process_taxonomy • hosts/current_parameters • hosts/puppetclass_parameters • hosts/template_used • hosts/interfaces • compute_resources/associate • compute_resources/[:cluster_selected, :template_selected, :provider_selected, :resource_pools] • compute_resources_vms/associate • puppetclasses/parameters • subnets/freeip • interfaces/new • api/v1/hosts/update • api/v2/hosts/update • api/v2/hosts/disassociate • api/v2/interfaces/create • api/v2/interfaces/update • api/v2/interfaces/destroy • api/v2/compute_resources/associate • api/v2/hosts/host_collections • katello/api/v2/host_errata/apply • katello/api/v2/host_packages/install • katello/api/v2/host_packages/upgrade • katello/api/v2/host_packages/upgrade_all • katello/api/v2/host_packages/remove • katello/api/v2/host_subscriptions/auto_attach 	

Permission Name	Actions	Resource Type
	<ul style="list-style-type: none"> • katello/api/v2/host_subscriptions/add_subscriptions • katello/api/v2/host_subscriptions/remove_subscriptions • katello/api/v2/host_subscriptions/content_override • katello/api/v2/hosts_bulk_actions/bulk_add_host_collections • katello/api/v2/hosts_bulk_actions/bulk_remove_host_collections • katello/api/v2/hosts_bulk_actions/install_content • katello/api/v2/hosts_bulk_actions/update_content • katello/api/v2/hosts_bulk_actions/remove_content • katello/api/v2/hosts_bulk_actions/environment_content_view 	
destroy_hosts	<ul style="list-style-type: none"> • hosts/destroy • hosts/multiple_actions • hosts/reset_multiple • hosts/multiple_destroy • hosts/submit_multiple_destroy • api/v1/hosts/destroy • api/v2/hosts/destroy • api/v2/interfaces/destroy • katello/api/v2/hosts_bulk_actions/destroy_hosts 	

Permission Name	Actions	Resource Type
build_hosts	<ul style="list-style-type: none"> • hosts/setBuild • hosts/cancelBuild • hosts/multiple_build • hosts/submit_multiple_build • hosts/review_before_build • hosts/rebuild_config • hosts/submit_rebuild_config • tasks/show • api/v2/tasks/index • api/v2/hosts/rebuild_config 	
power_hosts	<ul style="list-style-type: none"> • hosts/power • api/v2/hosts/power 	
console_hosts	<ul style="list-style-type: none"> • hosts/console 	
ipmi_boot	<ul style="list-style-type: none"> • hosts/ipmi_boot • api/v2/hosts/boot 	
puppetrun_hosts	<ul style="list-style-type: none"> • hosts/puppetrun • hosts/multiple_puppetrun • hosts/update_multiple_puppetrun • api/v2/hosts/puppetrun 	
search_repository_image_search	<ul style="list-style-type: none"> • image_search/auto_complete_repository_name • image_search/auto_complete_image_tag • image_search/search_repository 	Docker/ImageSearch

Permission Name	Actions	Resource Type
view_images	<ul style="list-style-type: none"> images/index images/show images/auto_complete_search api/v1/images/index api/v1/images/show api/v2/images/index api/v2/images/show 	
create_images	<ul style="list-style-type: none"> images/new images/create api/v1/images/create api/v2/images/create 	
edit_images	<ul style="list-style-type: none"> images/edit images/update api/v1/images/update api/v2/images/update 	
destroy_images	<ul style="list-style-type: none"> images/destroy api/v1/images/destroy api/v2/images/destroy 	
view_lifecycle_environments	<ul style="list-style-type: none"> katello/api/v2/environments/index katello/api/v2/environments/show katello/api/v2/environments/paths katello/api/v2/environments/repositories katello/api/rhsm/candlepin_proxies/rhsm_index katello/environments/auto_complete_search 	Katello::KTEEnvironment

Permission Name	Actions	Resource Type
create_lifecycle_environments	<ul style="list-style-type: none"> katello/api/v2/environments/create 	Katello::KTEEnvironment
edit_lifecycle_environments	<ul style="list-style-type: none"> katello/api/v2/environments/update 	Katello::KTEEnvironment
destroy_lifecycle_environments	<ul style="list-style-type: none"> katello/api/v2/environments/destroy 	Katello::KTEEnvironment
promote_or_remove_content_views_to_environments		Katello::KTEEnvironment
view_locations	<ul style="list-style-type: none"> locations/index locations/show locations/auto_complete_search api/v1/locations/index api/v1/locations/show api/v2/locations/index api/v2/locations/show 	
create_locations	<ul style="list-style-type: none"> locations/new locations/create locations/clone_taxonomy locations/step2 locations/nest api/v1/locations/create api/v2/locations/create 	

Permission Name	Actions	Resource Type
edit_locations	<ul style="list-style-type: none"> • locations/edit • locations/update • locations/import_mismatches • locations/parent_taxonomy_selected • api/v1/locations/update • api/v2/locations/update 	
destroy_locations	<ul style="list-style-type: none"> • locations/destroy • api/v1/locations/destroy • api/v2/locations/destroy 	
assign_locations	<ul style="list-style-type: none"> • locations/assign_all_hosts • locations/assign_hosts • locations/assign_selected_hosts 	
view_mail_notifications	<ul style="list-style-type: none"> • mail_notifications/index • mail_notifications/auto_complete_search • mail_notifications/show • api/v2/mail_notifications/index • api/v2/mail_notifications/show 	
view_media	<ul style="list-style-type: none"> • media/index • media/show • media/auto_complete_search • api/v1/media/index • api/v1/media/show • api/v2/media/index • api/v2/media/show 	

Permission Name	Actions	Resource Type
create_media	<ul style="list-style-type: none"> • media/new • media/create • api/v1/media/create • api/v2/media/create 	
edit_media	<ul style="list-style-type: none"> • media/edit • media/update • api/v1/media/update • api/v2/media/update 	
destroy_media	<ul style="list-style-type: none"> • media/destroy • api/v1/media/destroy • api/v2/media/destroy 	
view_models	<ul style="list-style-type: none"> • models/index • models/show • models/auto_complete_search • api/v1/models/index • api/v1/models/show • api/v2/models/index • api/v2/models/show 	
create_models	<ul style="list-style-type: none"> • models/new • models/create • api/v1/models/create • api/v2/models/create 	

Permission Name	Actions	Resource Type
edit_models	<ul style="list-style-type: none"> models/edit models/update api/v1/models/update api/v2/models/update 	
destroy_models	<ul style="list-style-type: none"> models/destroy api/v1/models/destroy api/v2/models/destroy 	
view_operatingsystems	<ul style="list-style-type: none"> operatingsystems/index operatingsystems/show operatingsystems/bootfiles operatingsystems/auto_complete_search api/v1/operatingsystems/index api/v1/operatingsystems/show api/v1/operatingsystems/bootfiles api/v2/operatingsystems/index api/v2/operatingsystems/show api/v2/operatingsystems/bootfiles api/v2/os_default_templates/index api/v2/os_default_templates/show 	
create_operatingsystems	<ul style="list-style-type: none"> operatingsystems/new operatingsystems/create api/v1/operatingsystems/create api/v2/operatingsystems/create api/v2/os_default_templates/create 	

Permission Name	Actions	Resource Type
edit_operatingsystems	<ul style="list-style-type: none"> • operatingsystems/edit • operatingsystems/update • api/v1/operatingsystems/update • api/v2/operatingsystems/update • api/v2/parameters/create • api/v2/parameters/update • api/v2/parameters/destroy • api/v2/parameters/reset • api/v2/os_default_templates/create • api/v2/os_default_templates/update • api/v2/os_default_templates/destroy 	
destroy_operatingsystems	<ul style="list-style-type: none"> • operatingsystems/destroy • api/v1/operatingsystems/destroy • api/v2/operatingsystems/destroy • api/v2/os_default_templates/create 	
view_organizations	<ul style="list-style-type: none"> • organizations/index • organizations/show • organizations/auto_complete_search • api/v1/organizations/index • api/v1/organizations/show • api/v2/organizations/index • api/v2/organizations/show • katello/api/v2/organizations/index • katello/api/v2/organizations/show • katello/api/v2/organizations/redhat_provider • katello/api/v2/organizations/download_debug_certificate • katello/api/v2/tasks/index 	

Permission Name	Actions	Resource Type
create_organizations	<ul style="list-style-type: none"> • organizations/new • organizations/create • organizations/clone_taxonomy • organizations/step2 • organizations/nest • api/v1/organizations/create • api/v2/organizations/create • katello/api/v2/organizations/create 	
edit_organizations	<ul style="list-style-type: none"> • organizations/edit • organizations/update • organizations/import_mismatches • organizations/parent_taxonomy_selected • api/v1/organizations/update • api/v2/organizations/update • katello/api/v2/organizations/update • katello/api/v2/organizations/autoattach_subscriptions 	
destroy_organizations	<ul style="list-style-type: none"> • organizations/destroy • api/v1/organizations/destroy • api/v2/organizations/destroy • katello/api/v2/organizations/destroy 	
assign_organizations	<ul style="list-style-type: none"> • organizations/assign_all_hosts • organizations/assign_hosts • organizations/assign_selected_hosts 	

Permission Name	Actions	Resource Type
view_ptables	<ul style="list-style-type: none"> • ptables/index • ptables/show • ptables/auto_complete_search • ptables/revision • ptables/preview • api/v1/ptables/index • api/v1/ptables/show • api/v2/ptables/index • api/v2/ptables/show • api/v2/ptables/revision 	
create_ptables	<ul style="list-style-type: none"> • ptables/new • ptables/create • ptables/clone_template • api/v1/ptables/create • api/v2/ptables/create • api/v2/ptables/clone 	
edit_ptables	<ul style="list-style-type: none"> • ptables/edit • ptables/update • api/v1/ptables/update • api/v2/ptables/update 	
destroy_ptables	<ul style="list-style-type: none"> • ptables/destroy • api/v1/ptables/destroy • api/v2/ptables/destroy 	

Permission Name	Actions	Resource Type
lock_ptables	<ul style="list-style-type: none"> • ptables/lock • ptables/unlock • api/v2/ptables/lock • api/v2/ptables/unlock 	
view_plugins	<ul style="list-style-type: none"> • plugins/index • api/v2/plugins/index 	
view_products	<ul style="list-style-type: none"> • katello/products/auto_complete • katello/products/auto_complete_search • katello/api/v2/products/index • katello/api/v2/products/show • katello/api/v2/repositories/index • katello/api/v2/repositories/show • katello/api/v2/packages/index • katello/api/v2/packages/show • katello/api/v2/distributions/index • katello/api/v2/distributions/show • katello/api/v2/package_groups/index • katello/api/v2/package_groups/show • katello/api/v2/errata/index • katello/api/v2/errata/show • katello/api/v2/puppet_modules/index • katello/api/v2/puppet_modules/show • katello/errata/short_details • katello/errata/auto_complete • katello/packages/details • katello/packages/auto_complete • katello/puppet_modules/show • katello/repositories/auto_complete_library 	Katello::Product

Permission Name	Actions	Resource Type
	<ul style="list-style-type: none"> • katello/repositories/repository_types • katello/content_search/index • katello/content_search/products • katello/content_search/repos • katello/content_search/packages • katello/content_search/errata • katello/content_search/puppet_modules • katello/content_search/packages_items • katello/content_search/errata_items • katello/content_search/puppet_modules_items • katello/content_search/repo_packages • katello/content_search/repo_errata • katello/content_search/repo_puppet_modules • katello/content_search/repo_compare_errata • katello/content_search/repo_compare_packages • katello/content_search/repo_compare_puppet_modules 	
create_products	<ul style="list-style-type: none"> • katello/api/v2/products/create • katello/api/v2/repositories/create 	Katello::Product

Permission Name	Actions	Resource Type
edit_products	<ul style="list-style-type: none">katello/api/v2/products/updatekatello/api/v2/repositories/updatekatello/api/v2/repositories/remove_contentkatello/api/v2/repositories/import_uploadskatello/api/v2/repositories/upload_contentkatello/api/v2/products_bulk_actions/update_sync_planskatello/api/v2/content_uploads/createkatello/api/v2/content_uploads/updatekatello/api/v2/content_uploads/destroykatello/api/v2/organizations/repo_discoverkatello/api/v2/organizations/cancel_repo_discover	Katello::Product
destroy_products	<ul style="list-style-type: none">katello/api/v2/products/destroykatello/api/v2/repositories/destroykatello/api/v2/products_bulk_actions/destroy_productskatello/api/v2/repositories_bulk_actions/destroy_repositories	Katello::Product

Permission Name	Actions	Resource Type
sync_products	<ul style="list-style-type: none"> • katello/api/v2/products/sync • katello/api/v2/repositories/sync • katello/api/v2/products_bulk_actions/sync_products • katello/api/v2/repositories_bulk_actions/sync_repositories • katello/api/v2/sync/index • katello/api/v2/sync_plans/sync • katello/sync_management/index • katello/sync_management/sync_status • katello/sync_management/product_status • katello/sync_management/sync • katello/sync_management/destroy 	Katello::Product
export_products	<ul style="list-style-type: none"> • katello/api/v2/repositories/export 	Katello::Product

Permission Name	Actions	Resource Type
view_provisioning_templates	<ul style="list-style-type: none"> • provisioning_templates/index • provisioning_templates/show • provisioning_templates/revision • provisioning_templates/auto_complete_search • provisioning_templates/preview • api/v1/config_templates/index • api/v1/config_templates/show • api/v1/config_templates/revision • api/v2/config_templates/index • api/v2/config_templates/show • api/v2/config_templates/revision • api/v2/provisioning_templates/index • api/v2/provisioning_templates/show • api/v2/provisioning_templates/revision • api/v2/template_combinations/index • api/v2/template_combinations/show • api/v1/template_kinds/index • api/v2/template_kinds/index 	
create_provisioning_templates	<ul style="list-style-type: none"> • provisioning_templates/new • provisioning_templates/create • provisioning_templates/clone_template • api/v1/config_templates/create • api/v2/config_templates/create • api/v2/config_templates/clone • api/v2/provisioning_templates/create • api/v2/provisioning_templates/clone • api/v2/template_combinations/create 	

Permission Name	Actions	Resource Type
edit_provisioning_templates	<ul style="list-style-type: none"> provisioning_templates/edit provisioning_templates/update api/v1/config_templates/update api/v2/config_templates/update api/v2/provisioning_templates/update api/v2/template_combinations/update 	
destroy_provisioning_templates	<ul style="list-style-type: none"> provisioning_templates/destroy api/v1/config_templates/destroy api/v2/config_templates/destroy api/v2/provisioning_templates/destory api/v2/template_combinations/destory 	
deploy_provisioning_templates	<ul style="list-style-type: none"> provisioning_templates/build_pxe_default api/v1/config_templates/build_pxe_default api/v2/config_templates/build_pxe_default api/v2/provisioning_templates/build_pxe_default 	
lock_provisioning_templates	<ul style="list-style-type: none"> provisioning_templates/lock provisioning_templates/unlock api/v2/config_templates/lock api/v2/config_templates/unlock api/v2/provisioning_templates/lock api/v2/provisioning_templates/unlock 	
user_logout	<ul style="list-style-type: none"> users/logout 	

Permission Name	Actions	Resource Type
my_account	<ul style="list-style-type: none"> • users/edit • katello/api/v2/tasks/show 	
api_status	<ul style="list-style-type: none"> • api/v1/home/status • api/v2/home/status 	
view_puppetclasses	<ul style="list-style-type: none"> • puppetclasses/index • puppetclasses/show • puppetclasses/auto_complete_search • api/v1/puppetclasses/index • api/v1/puppetclasses/show • api/v2/puppetclasses/index • api/v2/puppetclasses/show • api/v1/lookup_keys/index • api/v1/lookup_keys/show • api/v2/smart_variables/index • api/v2/smart_variables/show • api/v2/smart_class_parameters/index • api/v2/smart_class_parameters/show 	
create_puppetclasses	<ul style="list-style-type: none"> • puppetclasses/new • puppetclasses/create • api/v1/puppetclasses/create • api/v2/puppetclasses/create 	

Permission Name	Actions	Resource Type
edit_puppetclasses	<ul style="list-style-type: none"> • puppetclasses/edit • puppetclasses/update • puppetclasses/override • api/v1/puppetclasses/update • api/v2/puppetclasses/update • api/v1/lookup_keys/create • api/v1/lookup_keys/update • api/v1/lookup_keys/destroy • api/v2/smart_variables/create • api/v2/smart_variables/update • api/v2/smart_variables/destroy • api/v2/smart_class_parameters/create • api/v2/smart_class_parameters/update • api/v2/smart_class_parameters/destroy 	
destroy_puppetclasses	<ul style="list-style-type: none"> • puppetclasses/destroy • api/v1/puppetclasses/destroy • api/v2/puppetclasses/destroy 	
import_puppetclasses	<ul style="list-style-type: none"> • puppetclasses/import_environments • puppetclasses/obsolete_and_new • api/v1/environments/import_puppetclasses • api/v2/environments/import_puppetclasses • api/v1/smart_proxies/import_puppetclasses • api/v2/smart_proxies/import_puppetclasses 	

Permission Name	Actions	Resource Type
view_realms	<ul style="list-style-type: none">• realms/index• realms/show• realms/auto_complete_search• api/v2/realms/index• api/v2/realms/show	
create_realms	<ul style="list-style-type: none">• realms/new• realms/create• api/v2/realms/create	
edit_realms	<ul style="list-style-type: none">• realms/edit• realms/update• api/v2/realms/update	
destroy_realms	<ul style="list-style-type: none">• realms/destroy• api/v2/realms/destroy	
view_search	<ul style="list-style-type: none">• redhat_access/search/index	
view_cases	<ul style="list-style-type: none">• redhat_access/cases/index• redhat_access/cases/create	
attachments	<ul style="list-style-type: none">• redhat_access/attachments/index• redhat_access/attachments/create	
configuration	<ul style="list-style-type: none">• redhat_access/configuration/index	
app_root	<ul style="list-style-type: none">• redhat_access/redhat_access/index	

Permission Name	Actions	Resource Type
view_log_viewer	<ul style="list-style-type: none"> redhat_access/logviewer/index 	
logs	<ul style="list-style-type: none"> redhat_access/logs/index 	
rh_telemetry_api	<ul style="list-style-type: none"> redhat_access/api/telemetry_api/proxy redhat_access/api/telemetry_api/connection_status 	
rh_telemetry_view	<ul style="list-style-type: none"> redhat_access/analytics_dashboard/index 	
rh_telemetry_configurations	<ul style="list-style-type: none"> redhat_access/telemetry_configurations/show redhat_access/telemetry_configurations/update 	
view_registries	<ul style="list-style-type: none"> registries/index registries/show api/v2/registries/index api/v2/registries/show 	DockerRegistry
view_registries	<ul style="list-style-type: none"> registries/index registries/show api/v2/registries/index api/v2/registries/show 	DockerRegistry

Permission Name	Actions	Resource Type
create_registries	<ul style="list-style-type: none">registries/newregistries/createregistries/updateregistries/editapi/v2/registries/createapi/v2/registries/update	DockerRegistry
destroy_registries	<ul style="list-style-type: none">registries/destroyapi/v2/registries/destroy	DockerRegistry
view_roles	<ul style="list-style-type: none">roles/indexroles/auto_complete_searchapi/v2/roles/indexapi/v2/roles/show	
create_roles	<ul style="list-style-type: none">roles/newroles/createroles/cloneapi/v2/roles/create	
edit_roles	<ul style="list-style-type: none">roles/editroles/updateapi/v2/roles/update	
destroy_roles	<ul style="list-style-type: none">roles/destroyapi/v2/roles/destroy	
access_settings	<ul style="list-style-type: none">home/settings	

Permission Name	Actions	Resource Type
view_smart_proxies	<ul style="list-style-type: none"> • smart_proxies/index • smart_proxies/ping • smart_proxies/auto_complete_search • smart_proxies/version • smart_proxies/show • smart_proxies/plugin_version • smart_proxies/tftp_server • smart_proxies/puppet_environments • smart_proxies/puppet_dashboard • smart_proxies/log_pane • smart_proxies/failed_modules • smart_proxies/errors_card • smart_proxies/modules_card • api/v1/smart_proxies/index • api/v1/smart_proxies/show • api/v2/smart_proxies/index • api/v2/smart_proxies/show • api/v2/smart_proxies/version • api/v2/smart_proxies/log 	
create_smart_proxies	<ul style="list-style-type: none"> • smart_proxies/new • smart_proxies/create • api/v1/smart_proxies/create • api/v2/smart_proxies/create 	

Permission Name	Actions	Resource Type
edit_smart_proxies	<ul style="list-style-type: none"> • smart_proxies/edit • smart_proxies/update • smart_proxies/refresh • smart_proxies/expire_logs • api/v1/smart_proxies/update • api/v1/smart_proxies/refresh • api/v2/smart_proxies/update • api/v2/smart_proxies/refresh 	
destroy_smart_proxies	<ul style="list-style-type: none"> • smart_proxies/destroy • api/v1/smart_proxies/destroy • api/v2/smart_proxies/destroy 	
view_smart_proxies_autosign	<ul style="list-style-type: none"> • autosign/index • autosign/show • autosign/counts • api/v1/autosign/index • api/v2/autosign/index 	
create_smart_proxies_autosign	<ul style="list-style-type: none"> • autosign/new • autosign/create 	
destroy_smart_proxies_autosign	<ul style="list-style-type: none"> • autosign/destroy 	
view_smart_proxies_puppetca	<ul style="list-style-type: none"> • puppetca/index • puppetca/counts • puppetca/expiry 	
edit_smart_proxies_puppetca	<ul style="list-style-type: none"> • puppetca/update 	

Permission Name	Actions	Resource Type
destroy_smart_proxies_puppetca	<ul style="list-style-type: none"> • puppetca/destroy 	
view_statistics	<ul style="list-style-type: none"> • statistics/index • api/v1/statistics/index • api/v2/statistics/index 	
view_subnets	<ul style="list-style-type: none"> • subnets/index • subnets/show • subnets/auto_complete_search • api/v1/subnets/index • api/v1/subnets/show • api/v2/subnets/index • api/v2/subnets/show 	
create_subnets	<ul style="list-style-type: none"> • subnets/new • subnets/create • api/v1/subnets/create • api/v2/subnets/create 	
edit_subnets	<ul style="list-style-type: none"> • subnets/edit • subnets/update • api/v1/subnets/update • api/v2/subnets/update 	
destroy_subnets	<ul style="list-style-type: none"> • subnets/destroy • api/v1/subnets/destroy • api/v2/subnets/destroy 	

Permission Name	Actions	Resource Type
import_subnets	<ul style="list-style-type: none"> • subnets/import • subnets/create_multiple 	
view_subscriptions	<ul style="list-style-type: none"> • katello/api/v2/subscriptions/index • katello/api/v2/subscriptions/show • katello/api/v2/subscriptions/available • katello/api/v2/subscriptions/manifest_history • katello/api/v2/subscriptions/auto_complete_search • katello/api/v2/repository_sets/index • katello/api/v2/repository_sets/show • katello/api/v2/repository_sets/available_repositories 	Organization
attach_subscriptions	<ul style="list-style-type: none"> • katello/api/v2/subscriptions/create 	Organization
unattach_subscriptions	<ul style="list-style-type: none"> • katello/api/v2/subscriptions/destroy 	Organization
import_manifest	<ul style="list-style-type: none"> • katello/products/available_repositories • katello/products/toggle_repository • katello/providers/redhat_provider • katello/providers/redhat_provider_tab • katello/api/v2/subscriptions/upload • katello/api/v2/subscriptions/refresh_manifest • katello/api/v2/repository_sets/enable • katello/api/v2/repository_sets/disable 	Organization
delete_manifest	<ul style="list-style-type: none"> • katello/api/v2/subscriptions/delete_manifest 	Organization

Permission Name	Actions	Resource Type
view_sync_plans	<ul style="list-style-type: none"> katello/sync_plans/all katello/sync_plans/index katello/sync_plans/auto_complete_search katello/api/v2/sync_plans/index katello/api/v2/sync_plans/show katello/api/v2/sync_plans/add_products katello/api/v2/sync_plans/remove_products katello/api/v2/sync_plans/available_products katello/api/v2/products/index 	Katello::SyncPlan
create_sync_plans	<ul style="list-style-type: none"> katello/api/v2/sync_plans/create 	Katello::SyncPlan
edit_sync_plans	<ul style="list-style-type: none"> katello/api/v2/sync_plans/update 	Katello::SyncPlan
destroy_sync_plans	<ul style="list-style-type: none"> katello/api/v2/sync_plans/destroy 	Katello::SyncPlan
view_tasks	<ul style="list-style-type: none"> trends/show 	
view_trends	<ul style="list-style-type: none"> trends/index trends/show 	
create_trends	<ul style="list-style-type: none"> trends/new trends/create 	
edit_trends	<ul style="list-style-type: none"> trends/edit trends/update 	

Permission Name	Actions	Resource Type
destroy_trends	<ul style="list-style-type: none"> trends/destroy 	
update_trends	<ul style="list-style-type: none"> trends/count 	
my_organizations	<ul style="list-style-type: none"> katello/api/rhsm/candlepin_proxies/list_owners 	
view_usergroups	<ul style="list-style-type: none"> usergroups/index usergroups/show usergroups/auto_complete_search api/v1/usergroups/index api/v1/usergroups/show api/v2/usergroups/index api/v2/usergroups/show 	
create_usergroups	<ul style="list-style-type: none"> usergroups/new usergroups/create api/v1/usergroups/create api/v2/usergroups/create 	
edit_usergroups	<ul style="list-style-type: none"> usergroups/edit usergroups/update api/v1/usergroups/update api/v2/usergroups/update 	
destroy_usergroups	<ul style="list-style-type: none"> usergroups/destroy api/v1/usergroups/destroy api/v2/usergroups/destroy 	

Permission Name	Actions	Resource Type
view_users	<ul style="list-style-type: none">• users/index• users/show• users/auto_complete_search• api/v1/users/index• api/v1/users/show• api/v2/users/index• api/v2/users/show	
create_users	<ul style="list-style-type: none">• users/new• users/create• users/auth_source_selected• api/v1/users/create• api/v2/users/create	
edit_users	<ul style="list-style-type: none">• users/edit• users/update• users/auth_source_selected• users/test_mail• api/v1/users/update• api/v2/users/update	
destroy_users	<ul style="list-style-type: none">• users/destroy• api/v1/users/destroy• api/v2/users/destroy	