



Red Hat Satellite 6.4

Hammer Cheat Sheet

Hammer CLI Cheat Sheet for Red Hat Satellite 6.4.

Red Hat Satellite 6.4 Hammer Cheat Sheet

Hammer CLI Cheat Sheet for Red Hat Satellite 6.4.

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

Legal Notice

Copyright © 2018 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

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

This document contains Hammer CLI commands for Red Hat Satellite 6.4.

Table of Contents

1. INTRODUCTION	2
2. GENERAL INFORMATION	2
3. ORGANIZATIONS, LOCATIONS, AND REPOSITORIES	2
4. CONTENT LIFE CYCLES	3
5. PROVISIONING ENVIRONMENTS	5
6. ACTIVATION KEYS	6
7. USERS AND PERMISSIONS	6
8. ERRATA	7
9. DOCKER CONTAINERS	8
10. HOSTS	8
11. TASKS	10

1. INTRODUCTION

Hammer is a command-line tool provided with Red Hat Satellite 6. You can use Hammer to configure and manage a Red Hat Satellite Server by using either CLI commands or shell script automation. The following cheat sheet provides a condensed overview of essential Hammer commands.

For more information about Hammer, see the [Red Hat Hammer CLI Guide](#).

2. GENERAL INFORMATION

Table 1. General Information

Subcommand	Description and tasks
--help	Display hammer commands and options, append after a subcommand to get more information
org	<p>The setting is organization-specific, append --organization <i>org_name</i>, or set default organization with:</p> <pre>hammer defaults add \ --param-name <i>organization_id</i> \ --param-value <i>org_ID</i></pre>
loc	<p>The setting is location-specific, append --location <i>loc_name</i>, or set default location with:</p> <pre>hammer defaults add \ --param-name <i>location_id</i> \ --param-value <i>loc_ID</i></pre>

Note: This cheat sheet assumes saved credentials in `~/.hammer/cli_config.yml`. For more information, see [Authentication](#) in the [Red Hat Hammer CLI Guide](#).

3. ORGANIZATIONS, LOCATIONS, AND REPOSITORIES

Table 2. Organizations, Locations, and Repositories

Subcommand	Description and tasks
------------	-----------------------

Subcommand	Description and tasks
organization	<p>Create an organization:</p> <pre>hammer organization create \ --name <i>org_name</i></pre> <p>List organizations:</p> <pre>hammer organization list</pre>
location	See the options for organization
subscription org	<p>Upload a subscription manifest:</p> <pre>hammer subscription upload \ --file path</pre>
repository-set org	<p>Enable a repository:</p> <pre>hammer repository-set enable \ --product <i>prod_name</i> \ --basearch <i>base_arch</i> \ --releasever <i>rel_v</i> \ --name <i>repo_name</i></pre>
repository org	<p>Synchronize a repository:</p> <pre>hammer repository synchronize \ --product <i>prod_name</i> \ --name <i>repo_name</i></pre> <p>Create a custom repository:</p> <pre>hammer repository create \ --product <i>prod_name</i> \ --content-type <i>cont_type</i> \ --publish-via-http true \ --url <i>repo_url</i> \ --name <i>repo_name</i></pre> <p>Upload content to a custom repository:</p> <pre>hammer repository upload-content \ --product <i>prod_name</i> \ --id <i>repo_id</i> \ --path <i>path_to_dir</i></pre>

4. CONTENT LIFE CYCLES

Table 3. Content Life Cycles

Subcommand	Description and tasks
lifecycle-environment org	<p>Create a life cycle environment:</p> <pre>hammer lifecycle-environment create \ --name <i>env_name</i> --description <i>env_desc</i> --prior <i>prior_env_name</i></pre> <p>List life cycle environments:</p> <pre>hammer lifecycle-environment list</pre>
content-view org	<p>Create a content view:</p> <pre>hammer content-view create \ --name <i>cv_n</i> \ --repository-ids <i>repo_ID1,...</i> \ --description <i>cv_description</i></pre> <p>Add repositories to a content view:</p> <pre>hammer content-view add-repository \ --name <i>cv_n</i> \ --repository-id <i>repo_ID</i></pre> <p>Add Puppet modules to a content view:</p> <pre>hammer content-view puppet-module add \ --content-view <i>cv_n</i> \ --name <i>module_name</i></pre> <p>Publishing a content view:</p> <pre>hammer content-view publish \ --id <i>cv_ID</i></pre> <p>Promoting a content view:</p> <pre>hammer content-view version promote \ --content-view <i>cv_n</i> \ --to-lifecycle-environment <i>env_name</i></pre> <p>Incremental update of a content view:</p> <pre>hammer content-view version incremental-update \ --content-view-version-id <i>cv_ID</i> \ --packages <i>pkg_n1,...</i> \ --environment-ids <i>env_ID1,...</i></pre>

5. PROVISIONING ENVIRONMENTS

Table 4. Provisioning Environments

Subcommand	Description and tasks
domain	<p>Create a domain:</p> <pre>hammer domain create \ --name <i>domain_name</i></pre>
subnet org loc	<p>Add a subnet:</p> <pre>hammer subnet create \ --name <i>subnet_name</i> \ --organization-ids <i>org_ID1,...</i> \ --location-ids <i>loc_ID1,...</i> \ --domain-ids <i>dom_ID1,...</i> \ --boot-mode <i>boot_mode</i> \ --network <i>network_address</i> \ --mask <i>netmask</i> --ipam <i>ipam</i></pre>
compute-resource org loc	<p>Create a compute resource:</p> <pre>hammer compute-resource create \ --name <i>cr_name</i> \ --organization-ids <i>org_ID1,...</i> \ --location-ids <i>loc_ID1,...</i> \ --provider <i>provider_name</i></pre>
medium	<p>Add an installation medium:</p> <pre>hammer medium create \ --name <i>med_name</i> \ --path <i>path_to_medium</i></pre>
partition-table	<p>Add a partition table:</p> <pre>hammer partition-table create \ --name <i>tab_name</i> \ --path <i>path_to_file</i> \ --os-family <i>os_family</i></pre>
template	<p>Add a provisioning template:</p> <pre>hammer template create \ --name <i>tmp_name</i> \ --file <i>path_to_template</i></pre>

Subcommand	Description and tasks
os	<p>Add an operating system:</p> <pre>hammer os create \ --name <i>os_name</i> \ --version <i>version_num</i></pre>

6. ACTIVATION KEYS

Table 5. Activation Keys

Subcommand	Description and tasks
activation-key org	<p>Create an activation key:</p> <pre>hammer activation-key create \ --name <i>ak_name</i> \ --content-view <i>cv_n</i> \ --lifecycle-environment <i>lc_name</i></pre> <p>Add a subscription to the activation key:</p> <pre>hammer activation-key add-subscription \ --id <i>ak_ID</i> \ --subscription-id <i>sub_ID</i></pre>

7. USERS AND PERMISSIONS

Table 6. Users and Permissions

Subcommand	Description and tasks
user org	<p>Create a user:</p> <pre>hammer user create \ --login <i>user_name</i> \ --mail <i>user_mail</i> \ --auth-source-id 1 \ --organization-ids <i>org_ID1,org_ID2,...</i></pre> <p>Add a role to a user:</p> <pre>hammer user add-role \ --id <i>user_id</i> \ --role <i>role_name</i></pre>

Subcommand	Description and tasks
user-group	<p>Create a user group:</p> <pre>hammer user-group create \ --name <i>ug_name</i></pre> <p>Add a role to a user group:</p> <pre>hammer user-group add-role \ --id <i>ug_id</i> \ --role <i>role_name</i></pre>
role	<p>Create a role:</p> <pre>hammer role create \ --name <i>role_name</i></pre>
filter	<p>Create a filter and add it to a role:</p> <pre>hammer filter create \ --name <i>role_name</i> \ --permission-ids <i>perm_ID1, perm_ID2, ...</i></pre>

8. ERRATA

Table 7. Errata

Subcommand	Description and tasks
erratum	<p>List errata:</p> <pre>hammer erratum list</pre> <p>Find erratum by CVE:</p> <pre>hammer erratum list --cve CVE</pre> <p>Inspect erratum:</p> <pre>hammer erratum info --id <i>err_ID</i></pre>

Subcommand	Description and tasks
host	<p>List errata applicable to a host:</p> <pre>hammer host errata list \ --host <i>host_name</i></pre> <p>Apply errata to a host:</p> <pre>hammer host errata apply \ --host <i>host_name</i> \ --errata-ids <i>err_ID1, err_ID2, ...</i></pre>

9. DOCKER CONTAINERS

Table 8. Docker Containers

Subcommand	Description and tasks
docker	<p>Create a container:</p> <pre>hammer docker container create \ --name <i>container_name</i> \ --compute-resource-id <i>cr_ID</i> \ --repository-name <i>repo_name</i> \ --tag <i>tag</i> --command <i>command</i></pre> <p>Start a container:</p> <pre>hammer docker container start \ --id <i>cont_ID</i></pre>

10. HOSTS

Table 9. Hosts

Subcommand	Description and tasks
------------	-----------------------

Subcommand	Description and tasks
hostgroup org loc	<p>Create a host group:</p> <pre>hammer hostgroup create \ --name <i>hg_name</i> \ --environment <i>env_name</i> \ --architecture <i>arch_name</i> \ --domain <i>domain_name</i> \ --subnet <i>subnet_name</i> \ --puppet-proxy <i>proxy_name</i> \ --puppet-ca-proxy <i>ca-proxy_name</i> \ --operatingsystem <i>os_name</i> \ --partition-table <i>table_name</i> \ --medium <i>medium_name</i> \ --organization-ids <i>org_ID1</i>,... \ --location-ids <i>loc_ID1</i>,...</pre> <p>Add an activation key to a host group:</p> <pre>hammer hostgroup set-parameter \ --hostgroup "hg_name" \ --name "kt_activation_keys" \ --value <i>key_name</i></pre>
host org loc	<p>Create a host (inheriting parameters from a host group):</p> <pre>hammer host create \ --name <i>host_name</i> \ --hostgroup <i>hg_name</i> \ --interface="primary=true, \ mac=<i>mac_addr</i>, ip=<i>ip_addr</i>, \ provision=true" \ --organization-id <i>org_ID</i> \ --location-id <i>loc_ID</i> \ --ask-root-password yes</pre>
job-template	<p>Add a job template for remote execution:</p> <pre>hammer job-template create \ --file <i>path</i> \ --name <i>template_name</i> \ --provider-type SSH \ --job-category <i>category_name</i></pre>

Subcommand	Description and tasks
job-invocation	<div>Invoke a remote job:<pre>hammer job-invocation create \ --job-template <i>template_name</i> \ --inputs key1=<i>value</i>,... \ --search-query <i>query</i></pre></div> <div>Monitor the remote job:<pre>hammer job-invocation output \ --id <i>job_id</i> --host <i>host_name</i></pre></div>

11. TASKS

Table 10. Tasks

Subcommand	Description and tasks
task	<div>List all tasks:<pre>hammer task list</pre></div> <div>Monitor progress of a running task:<pre>hammer task progress \ --id <i>task_ID</i></pre></div>