



Red Hat Satellite 6.12

Administering Red Hat Satellite

A guide to administering Red Hat Satellite.

Red Hat Satellite 6.12 Administering Red Hat Satellite

A guide to administering Red Hat Satellite.

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

Legal Notice

Copyright © 2023 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, the Red Hat 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 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 guide provides instructions on how to configure and administer a Red Hat Satellite 6 Server. Before continuing with this workflow you must have successfully installed a Red Hat Satellite 6 Server and any required Capsule Servers.

Table of Contents

CHAPTER 1. ACCESSING RED HAT SATELLITE	6
1.1. IMPORTING THE KATELLO ROOT CA CERTIFICATE	6
1.2. LOGGING IN TO SATELLITE	6
1.3. NAVIGATION TABS IN THE SATELLITE WEB UI	7
1.4. CHANGING THE PASSWORD	7
1.5. RESETTING THE ADMINISTRATIVE USER PASSWORD	8
1.6. SETTING A CUSTOM MESSAGE ON THE LOGIN PAGE	9
CHAPTER 2. STARTING AND STOPPING RED HAT SATELLITE	10
CHAPTER 3. MIGRATING FROM INTERNAL SATELLITE DATABASES TO EXTERNAL DATABASES	11
3.1. POSTGRESQL AS AN EXTERNAL DATABASE CONSIDERATIONS	11
3.2. PREPARING A HOST FOR EXTERNAL DATABASES	12
3.3. INSTALLING POSTGRESQL	12
3.4. MIGRATING TO EXTERNAL DATABASES	14
CHAPTER 4. MANAGING SATELLITE WITH ANSIBLE COLLECTIONS	16
4.1. INSTALLING THE SATELLITE ANSIBLE MODULES	16
4.2. VIEWING THE SATELLITE ANSIBLE MODULES	16
CHAPTER 5. MANAGING ORGANIZATIONS	17
5.1. CREATING AN ORGANIZATION	17
5.2. SETTING THE ORGANIZATION CONTEXT	18
5.3. CREATING AN ORGANIZATION DEBUG CERTIFICATE	18
5.4. BROWSING REPOSITORY CONTENT USING AN ORGANIZATION DEBUG CERTIFICATE	19
5.5. DELETING AN ORGANIZATION	20
CHAPTER 6. MANAGING LOCATIONS	21
6.1. CREATING A LOCATION	21
6.2. CREATING MULTIPLE LOCATIONS	21
6.3. SETTING THE LOCATION CONTEXT	22
6.4. DELETING A LOCATION	22
CHAPTER 7. MANAGING USERS AND ROLES	24
7.1. MANAGING USERS	24
7.1.1. Creating a User	24
7.1.2. Assigning Roles to a User	25
7.1.3. Impersonating a Different User Account	26
7.1.4. Creating an API-Only User	26
7.2. MANAGING SSH KEYS	27
7.2.1. Managing SSH Keys for a User	27
7.3. CREATING AND MANAGING USER GROUPS	28
7.3.1. User Groups	28
7.3.2. Creating a User Group	28
7.3.3. Removing a User Group	29
7.4. CREATING AND MANAGING ROLES	29
7.4.1. Creating a Role	29
7.4.2. Cloning a Role	30
7.4.3. Adding Permissions to a Role	30
7.4.4. Viewing Permissions of a Role	31
7.4.5. Creating a Complete Permission Table	31
7.4.6. Removing a Role	32
7.5. PREDEFINED ROLES AVAILABLE IN SATELLITE	32

7.6. GRANULAR PERMISSION FILTERING	34
7.6.1. Creating a Granular Permission Filter	34
7.6.2. Examples of Using Granular Permission Filters	35
7.6.2.1. Applying Permissions for the Host Resource Type	35
7.6.2.2. Creating an Organization Specific Manager Role	35
7.6.3. Supported Operators for Granular Search	36
CHAPTER 8. CONFIGURING EMAIL NOTIFICATIONS	38
8.1. NOTIFICATION TYPES	38
8.2. CONFIGURING EMAIL NOTIFICATION PREFERENCES	38
8.3. TESTING EMAIL DELIVERY	39
8.4. TESTING EMAIL NOTIFICATIONS	39
8.5. CHANGING EMAIL NOTIFICATION SETTINGS FOR A HOST	40
CHAPTER 9. MANAGING SECURITY COMPLIANCE	41
9.1. SECURITY CONTENT AUTOMATION PROTOCOL	41
9.1.1. SCAP Content	41
9.1.2. XCCDF Profile	41
9.1.3. Listing Available XCCDF Profiles	41
9.2. INSTALLING THE OPENSAP PLUG-IN	41
9.3. CONFIGURING SCAP CONTENT	42
9.3.1. Importing OpenSCAP Puppet Modules	42
9.3.2. Loading the Default OpenSCAP Content	42
9.3.3. Extra SCAP Content	43
9.3.4. Uploading Extra SCAP Content	43
9.4. MANAGING COMPLIANCE POLICIES	43
9.4.1. Creating a Compliance Policy	44
9.4.2. Viewing a Compliance Policy	45
9.4.3. Editing a Compliance Policy	45
9.4.4. Deleting a Compliance Policy	45
9.5. CUSTOMIZING OPENSAP POLICIES	45
9.5.1. Uploading a Tailoring File	45
9.5.2. Assigning a Tailoring File to a Policy	46
9.6. CONFIGURING A HOST GROUP FOR OPENSAP	46
CHAPTER 10. RUNNING OPENSAP SCANS	48
10.1. CONFIGURING A HOST FOR OPENSAP	48
10.2. MONITORING COMPLIANCE	49
10.2.1. Compliance Policy Dashboard	49
10.2.2. Viewing the Compliance Policy Dashboard	49
10.2.3. Compliance Email Notifications	50
10.2.4. Compliance Reports	50
10.2.5. Examining Compliance Failures of Hosts	51
10.2.6. Searching Compliance Reports	52
10.2.7. Deleting a Compliance Report	53
10.2.8. Deleting Multiple Compliance Reports	53
10.3. SPECIFICATIONS SUPPORTED BY OPENSAP	53
CHAPTER 11. BACKING UP SATELLITE SERVER AND CAPSULE SERVER	55
11.1. ESTIMATING THE SIZE OF A BACKUP	55
11.2. PERFORMING A FULL BACKUP OF SATELLITE SERVER OR CAPSULE SERVER	56
11.3. PERFORMING A BACKUP WITHOUT PULP CONTENT	58
11.4. PERFORMING AN INCREMENTAL BACKUP	59
11.5. EXAMPLE OF A WEEKLY FULL BACKUP FOLLOWED BY DAILY INCREMENTAL BACKUPS	60

11.6. PERFORMING AN ONLINE BACKUP	60
11.7. PERFORMING A SNAPSHOT BACKUP	61
11.8. WHITE-LISTING AND SKIPPING STEPS WHEN PERFORMING BACKUPS	61
CHAPTER 12. RESTORING SATELLITE SERVER OR CAPSULE SERVER FROM A BACKUP	63
12.1. RESTORING FROM A FULL BACKUP	63
12.2. RESTORING FROM INCREMENTAL BACKUPS	64
12.3. BACKUP AND RESTORE CAPSULE SERVER USING A VIRTUAL MACHINE SNAPSHOT	64
12.3.1. Synchronizing an External Capsule	65
CHAPTER 13. RENAMING SATELLITE SERVER OR CAPSULE SERVER	66
13.1. RENAMING SATELLITE SERVER	66
13.2. RENAMING CAPSULE SERVER	67
CHAPTER 14. MAINTAINING SATELLITE SERVER	70
14.1. DELETING AUDIT RECORDS	70
14.2. ANONYMIZING AUDIT RECORDS	70
14.3. DELETING REPORT RECORDS	70
14.4. CONFIGURING THE CLEANING UNUSED TASKS FEATURE	70
14.5. DELETING TASK RECORDS	71
14.6. DELETING A TASK BY ID	71
14.7. RECOVERING FROM A FULL DISK	72
14.8. MANAGING PACKAGES ON THE BASE OPERATING SYSTEM OF SATELLITE SERVER OR CAPSULE SERVER	73
14.9. RECLAIMING POSTGRESQL SPACE	74
CHAPTER 15. LOGGING AND REPORTING PROBLEMS	75
15.1. ENABLING DEBUG LOGGING	75
15.2. INCREASING THE LOGGING LEVELS TO HELP WITH DEBUGGING	75
15.2.1. Increasing the Logging Level For Hammer	76
15.2.2. Increasing the Logging Level On Capsule	76
15.2.3. Increasing the Logging Level For Candlepin	76
15.2.4. Increasing the Logging Level On Satellite	77
15.2.5. Increasing the Logging Level For Qpid Dispatch Router	77
15.2.6. Increasing the Logging Level For Qpid Broker	78
15.2.7. Increasing the Logging Level For Redis	78
15.2.8. Increasing the Logging Level For Postgres	78
15.2.9. Increasing the Logging Level For Satellite Installer	79
15.2.10. Increasing the Logging Level For Pulp	79
15.2.11. Increasing the Logging Level For Puppet Agent	79
15.2.12. Increasing the Logging Level For Puppet Server	79
15.3. RETRIEVING THE STATUS OF SERVICES	80
15.4. RESTARTING SERVICES	80
15.5. ENABLING INDIVIDUAL LOGGERS	80
15.6. CONFIGURING LOGGING TO JOURNAL	81
15.7. LOG FILE DIRECTORIES PROVIDED BY SATELLITE	81
15.8. UTILITIES FOR COLLECTING LOG INFORMATION	82
CHAPTER 16. MONITORING RESOURCES	83
16.1. USING THE RED HAT SATELLITE CONTENT DASHBOARD	83
16.1.1. Managing Tasks	86
16.2. CONFIGURING RSS NOTIFICATIONS	87
16.3. MONITORING SATELLITE SERVER	87
16.4. MONITORING CAPSULE SERVER	87

16.4.1. Viewing General Capsule Information	87
16.4.2. Monitoring Services	88
16.4.3. Monitoring Puppet	88
CHAPTER 17. USING WEBHOOKS	89
17.1. MIGRATING TO WEBHOOKS	89
17.2. INSTALLING WEBHOOKS	90
17.3. CREATING A WEBHOOK TEMPLATE	90
17.4. CREATING A WEBHOOK	90
17.5. AVAILABLE WEBHOOK EVENTS	91
17.6. SHELLHOOKS	94
17.7. INSTALLING THE SHELLHOOKS PLUGIN	94
17.8. USING SHELLHOOK ARGUMENTS	95
CHAPTER 18. SEARCHING AND BOOKMARKING	96
18.1. BUILDING SEARCH QUERIES	96
18.1.1. Query Syntax	96
18.1.2. Query Operators	96
18.1.3. Query Values	97
18.2. USING FREE TEXT SEARCH	98
18.3. MANAGING BOOKMARKS	98
18.3.1. Creating Bookmarks	99
18.3.2. Deleting Bookmarks	99
APPENDIX A. ADMINISTRATION SETTINGS	100
A.1. GENERAL SETTINGS	100
A.2. SATELLITE TASK SETTINGS	101
A.3. TEMPLATE SYNC SETTINGS	102
A.4. DISCOVERED SETTINGS	103
A.5. BOOT DISK SETTINGS	105
A.6. RED HAT CLOUD SETTINGS	106
A.7. CONTENT SETTINGS	106
A.8. AUTHENTICATION SETTINGS	110
A.9. EMAIL SETTINGS	113
A.10. NOTIFICATIONS SETTINGS	114
A.11. PROVISIONING SETTINGS	114
A.12. FACTS SETTINGS	118
A.13. CONFIGURATION MANAGEMENT SETTINGS	119
A.14. REMOTE EXECUTION SETTINGS	119
A.15. ANSIBLE SETTINGS	121

CHAPTER 1. ACCESSING RED HAT SATELLITE

After Red Hat Satellite has been installed and configured, use the Satellite web UI interface to log in to Satellite for further configuration.

1.1. IMPORTING THE KATELLO ROOT CA CERTIFICATE

The first time you log in to Satellite, you might see a warning informing you that you are using the default self-signed certificate and you might not be able to connect this browser to Satellite until the root CA certificate is imported in the browser. Use the following procedure to locate the root CA certificate on Satellite and to import it into your browser.

To use the CLI instead of the Satellite web UI, see [CLI Procedure](#).

Prerequisite

- Your Red Hat Satellite is installed and configured.

Procedure

1. Identify the fully qualified domain name of your Satellite Server:

```
# hostname -f
```

2. Access the **pub** directory on your Satellite Server using a web browser pointed to the fully qualified domain name:

```
https://satellite.example.com/pub
```

3. When you access Satellite for the first time, an untrusted connection warning displays in your web browser. Accept the self-signed certificate and add the Satellite URL as a security exception to override the settings. This procedure might differ depending on the browser being used. Ensure that the Satellite URL is valid before you accept the security exception.
4. Select **katello-server-ca.crt**.
5. Import the certificate into your browser as a certificate authority and trust it to identify websites.

CLI Procedure

1. From the Satellite CLI, copy the **katello-server-ca.crt** file to the machine you use to access the Satellite web UI:

```
# scp /var/www/html/pub/katello-server-ca.crt username@hostname:remotefile
```

2. In the browser, import the **katello-server-ca.crt** certificate as a certificate authority and trust it to identify websites.

1.2. LOGGING IN TO SATELLITE

Use the web user interface to log in to Satellite for further configuration.

Prerequisite

- Ensure that the Katello root CA certificate is installed in your browser. For more information, see [Section 1.1, “Importing the Katello Root CA Certificate”](#).

Procedure


1. Access Satellite Server using a web browser pointed to the fully qualified domain name:

`https://satellite.example.com/`

2. Enter the user name and password created during the configuration process. If a user was not created during the configuration process, the default user name is *admin*. If you have problems logging in, you can reset the password. For more information, see [Section 1.5, “Resetting the Administrative User Password”](#).

1.3. NAVIGATION TABS IN THE SATELLITE WEB UI

Use the navigation tabs to browse the Satellite web UI.

Navigation Tabs	Description
Any Context	Clicking this tab changes the organization and location. If no organization or location is selected, the default organization is <i>Any Organization</i> and the default location is <i>Any Location</i> . Use this tab to change to different values.
Monitor	Provides summary dashboards and reports.
Content	Provides content management tools. This includes Content Views, Activation Keys, and Life Cycle Environments.
Hosts	Provides host inventory and provisioning configuration tools.
Configure	Provides general configuration tools and data including Host Groups and Puppet data.
Infrastructure	Provides tools on configuring how Satellite interacts with the environment.
User Name	Provides user administration where users can edit their personal information.
	Provides event notifications to keep administrators informed of important environment changes.
Administer	Provides advanced configuration for settings such as Users and RBAC, as well as general settings.

1.4. CHANGING THE PASSWORD

These steps show how to change your password.

Procedure

1. Click your user name at the top right corner.
2. Select **My Account** from the menu.
3. In the **Current Password** field, enter the current password.
4. In the **Password** field, enter a new password.
5. In the **Verify** field, enter the new password again.
6. Click the **Submit** button to save your new password.

1.5. RESETTING THE ADMINISTRATIVE USER PASSWORD

Use the following procedures to reset the administrative password to randomly generated characters or to set a new administrative password.

To Reset the Administrative User Password

1. Log in to the base operating system where Satellite Server is installed.
2. Enter the following command to reset the password:

```
# foreman-rake permissions:reset  
Reset to user: admin, password: qwJxBptxb7Gfcjj5
```

3. Use this password to reset the password in the Satellite web UI.
4. Edit the `~/.hammer/cli.modules.d/foreman.yml` file on Satellite Server to add the new password:

```
# vi ~/.hammer/cli.modules.d/foreman.yml
```

Unless you update the `~/.hammer/cli.modules.d/foreman.yml` file, you cannot use the new password with Hammer CLI.

To Set a New Administrative User Password

1. Log in to the base operating system where Satellite Server is installed.
2. To set the password, enter the following command:

```
# foreman-rake permissions:reset password=new_password
```

3. Edit the `~/.hammer/cli.modules.d/foreman.yml` file on Satellite Server to add the new password:

```
# vi ~/.hammer/cli.modules.d/foreman.yml
```

Unless you update the `~/.hammer/cli.modules.d/foreman.yml` file, you cannot use the new password with Hammer CLI.

1.6. SETTING A CUSTOM MESSAGE ON THE LOGIN PAGE

Procedure

1. In the Satellite web UI, navigate to **Administer** > **Settings**, and click the **General** tab.
2. Click the edit button next to **Login page footer text**, and enter the desired text to be displayed on the login page. For example, this text may be a warning message required by your company.
3. Click **Save**.
4. Log out of the Satellite web UI and verify that the custom text is now displayed on the login page below the Satellite version number.

CHAPTER 2. STARTING AND STOPPING RED HAT SATELLITE

Satellite provides the **satellite-maintain service** command to manage Satellite services from the command line. This is useful when creating a backup of Satellite. For more information on creating backups, see [Chapter 11, Backing Up Satellite Server and Capsule Server](#).

After installing Satellite with the **satellite-installer** command, all Satellite services are started and enabled automatically. View the list of these services by executing:

```
# satellite-maintain service list
```

To see the status of running services, execute:

```
# satellite-maintain service status
```

To stop Satellite services, execute:

```
# satellite-maintain service stop
```

To start Satellite services, execute:

```
# satellite-maintain service start
```

To restart Satellite services, execute:

```
# satellite-maintain service restart
```

CHAPTER 3. MIGRATING FROM INTERNAL SATELLITE DATABASES TO EXTERNAL DATABASES

When you install Red Hat Satellite, the **satellite-installer** command installs PostgreSQL databases on the same server as Satellite. If you are using the default internal databases but want to start using external databases to help with the server load, you can migrate your internal databases to external databases.

To confirm whether your Satellite Server has internal or external databases, you can query the status of your databases:

For PostgreSQL, enter the following command:

```
# satellite-maintain service status --only postgresql
```

Red Hat does not provide support or tools for external database maintenance. This includes backups, upgrades, and database tuning. You must have your own database administrator to support and maintain external databases.

To migrate from the default internal databases to external databases, you must complete the following procedures:

1. [Section 3.2, “Preparing a Host for External Databases”](#) . Prepare a Red Hat Enterprise Linux 8 server to host the external databases.
2. [Section 3.3, “Installing PostgreSQL”](#) . Prepare PostgreSQL with databases for Satellite, Pulp and Candlepin with dedicated users owning them.
3. [Section 3.4, “Migrating to External Databases”](#) . Edit the parameters of **satellite-installer** to point to the new databases, and run **satellite-installer**.

3.1. POSTGRESQL AS AN EXTERNAL DATABASE CONSIDERATIONS

Foreman, Katello, and Candlepin use the PostgreSQL database. If you want to use PostgreSQL as an external database, the following information can help you decide if this option is right for your Satellite configuration. Satellite supports PostgreSQL version 12.

Advantages of External PostgreSQL

- Increase in free memory and free CPU on Satellite
- Flexibility to set **shared_buffers** on the PostgreSQL database to a high number without the risk of interfering with other services on Satellite
- Flexibility to tune the PostgreSQL server’s system without adversely affecting Satellite operations

Disadvantages of External PostgreSQL

- Increase in deployment complexity that can make troubleshooting more difficult
- The external PostgreSQL server is an additional system to patch and maintain

- If either Satellite or the PostgreSQL database server suffers a hardware or storage failure, Satellite is not operational
- If there is latency between the Satellite server and database server, performance can suffer

If you suspect that the PostgreSQL database on your Satellite is causing performance problems, use the information in [Satellite 6: How to enable postgres query logging to detect slow running queries](#) to determine if you have slow queries. Queries that take longer than one second are typically caused by performance issues with large installations, and moving to an external database might not help. If you have slow queries, contact Red Hat Support.

3.2. PREPARING A HOST FOR EXTERNAL DATABASES

Install a freshly provisioned system with the latest Red Hat Enterprise Linux 8 to host the external databases.

Subscriptions for Red Hat Enterprise Linux do not provide the correct service level agreement for using Satellite with external databases. You must also attach a Satellite subscription to the base operating system that you want to use for the external databases.

Prerequisite

- The prepared host must meet Satellite's [Storage Requirements](#).

Procedure

1. Use the instructions in [Attaching the Satellite Infrastructure Subscription](#) to attach a Satellite subscription to your server.
2. Disable all repositories and enable only the following repositories:

```
# subscription-manager repos --disable '*'
# subscription-manager repos \
--enable=satellite-6.12-for-rhel-8-x86_64-rpms \
--enable=rhel-8-for-x86_64-baseos-rpms \
--enable=rhel-8-for-x86_64-appstream-rpms
```

3. Enable the following modules:

```
# dnf module enable satellite:el8
```



NOTE

Enablement of the module **satellite:el8** warns about a conflict with **postgresql:10** and **ruby:2.5** as these modules are set to the default module versions on Red Hat Enterprise Linux 8. The module **satellite:el8** has a dependency for the modules **postgresql:12** and **ruby:2.7** that will be enabled with the **satellite:el8** module. These warnings do not cause installation process failure, hence can be ignored safely. For more information about modules and lifecycle streams on Red Hat Enterprise Linux 8, see [Red Hat Enterprise Linux Application Streams Life Cycle](#).

3.3. INSTALLING POSTGRESQL

You can install only the same version of PostgreSQL that is installed with the **satellite-installer** tool during an internal database installation. Satellite supports PostgreSQL version 12.

Procedure

1. To install PostgreSQL, enter the following command:

```
# dnf install postgresql-server postgresql-evr
```

2. To initialize PostgreSQL, enter the following command:

```
# postgresql-setup initdb
```

3. Edit the **/var/lib/pgsql/data/postgresql.conf** file:

```
# vi /var/lib/pgsql/data/postgresql.conf
```

Note that the default configuration of external PostgreSQL needs to be adjusted to work with Satellite. The base recommended external database configuration adjustments are as follows:

- `checkpoint_completion_target`: 0.9
- `max_connections`: 500
- `shared_buffers`: 512MB
- `work_mem`: 4MB

4. Remove the **#** and edit to listen to inbound connections:

```
listen_addresses = '*'
```

5. Edit the **/var/lib/pgsql/data/pg_hba.conf** file:

```
# vi /var/lib/pgsql/data/pg_hba.conf
```

6. Add the following line to the file:

```
host all all Satellite_ip/32 md5
```

7. To start, and enable PostgreSQL service, enter the following commands:

```
# systemctl enable --now postgresql
```

8. Open the **postgresql** port on the external PostgreSQL server:

```
# firewall-cmd --add-service=postgresql  
# firewall-cmd --runtime-to-permanent
```

9. Switch to the **postgres** user and start the PostgreSQL client:

```
$ su - postgres -c psql
```

10. Create three databases and dedicated roles: one for Satellite, one for Candlepin, and one for Pulp:

```
CREATE USER "foreman" WITH PASSWORD 'Foreman_Password';
CREATE USER "candlepin" WITH PASSWORD 'Candlepin_Password';
CREATE USER "pulp" WITH PASSWORD 'Pulpcore_Password';
CREATE DATABASE foreman OWNER foreman;
CREATE DATABASE candlepin OWNER candlepin;
CREATE DATABASE pulpcore OWNER pulp;
```

11. Exit the **postgres** user:

```
# \q
```

12. From Satellite Server, test that you can access the database. If the connection succeeds, the commands return **1**.

```
# PGPASSWORD='Foreman_Password' psql -h postgres.example.com -p 5432 -U foreman
-d foreman -c "SELECT 1 as ping"
# PGPASSWORD='Candlepin_Password' psql -h postgres.example.com -p 5432 -U
candlepin -d candlepin -c "SELECT 1 as ping"
# PGPASSWORD='Pulpcore_Password' psql -h postgres.example.com -p 5432 -U pulp -d
pulpcore -c "SELECT 1 as ping"
```

3.4. MIGRATING TO EXTERNAL DATABASES

Back up and transfer existing data, then use the **satellite-installer** command to configure Satellite to connect to an external PostgreSQL database server.

Prerequisite

- You have installed and configured a PostgreSQL server on a Red Hat Enterprise Linux server.

Procedure

1. On Satellite Server, stop Satellite services:

```
# satellite-maintain service stop
```

2. Start the **PostgreSQL** services:

```
# systemctl start postgresql
```

3. Back up the internal databases:

```
# satellite-maintain backup online --skip-pulp-content --preserve-directory -y
/var/migration_backup
```

4. Transfer the data to the new external databases:

```
PGPASSWORD='Foreman_Password' pg_restore -h postgres.example.com -U foreman -d
foreman < /var/migration_backup/foreman.dump
```

```
PGPASSWORD='Candlepin_Password' pg_restore -h postgres.example.com -U candlepin -  
d candlepin < /var/migration_backup/candlepin.dump  
PGPASSWORD='Pulpcore_Password' pg_restore -h postgres.example.com -U pulp -d  
pulpcore < /var/migration_backup/pulpcore.dump
```

5. Use the **satellite-installer** command to update Satellite to point to the new databases:

```
satellite-installer --scenario satellite \  
  --foreman-db-host postgres.example.com \  
  --foreman-db-password Foreman_Password \  
  --foreman-db-database foreman \  
  --foreman-db-manage false \  
  --foreman-db-username foreman \  
  --katello-candlepin-db-host postgres.example.com \  
  --katello-candlepin-db-name candlepin \  
  --katello-candlepin-db-password Candlepin_Password \  
  --katello-candlepin-manage-db false \  
  --katello-candlepin-db-user candlepin \  
  --foreman-proxy-content-pulpcore-manage-postgresql false \  
  --foreman-proxy-content-pulpcore-postgresql-host postgres.example.com \  
  --foreman-proxy-content-pulpcore-postgresql-db-name pulpcore \  
  --foreman-proxy-content-pulpcore-postgresql-password Pulpcore_Password \  
  --foreman-proxy-content-pulpcore-postgresql-user pulp
```

CHAPTER 4. MANAGING SATELLITE WITH ANSIBLE COLLECTIONS

Satellite Ansible Collections is a set of Ansible modules that interact with the Satellite API. You can use Satellite Ansible Collections to manage and automate many aspects of Satellite.

4.1. INSTALLING THE SATELLITE ANSIBLE MODULES

Use this procedure to install the Satellite Ansible modules.

Procedure

- Install the package using the following command:

```
# satellite-maintain packages install ansible-collection-redhat-satellite
```

4.2. VIEWING THE SATELLITE ANSIBLE MODULES

You can view the installed Satellite Ansible modules by listing the content of the following directory:

```
# ls /usr/share/ansible/collections/ansible_collections/redhat/satellite/plugins/modules/
```



NOTE

At the time of writing, the **ansible-doc -l** command does not list collections yet.

Alternatively, you can also see the complete list of Satellite Ansible modules and other related information at <https://console.redhat.com/ansible/automation-hub/redhat/satellite/docs>.

All modules are in the **redhat.satellite** namespace and can be referred to in the format **redhat.satellite._module_name_**. For example, to display information about the **activation_key** module, enter the following command:

```
$ ansible-doc redhat.satellite.activation_key
```

CHAPTER 5. MANAGING ORGANIZATIONS

Organizations divide Red Hat Satellite resources into logical groups based on ownership, purpose, content, security level, or other divisions. You can create and manage multiple organizations through Red Hat Satellite, then divide and assign your Red Hat subscriptions to each individual organization. This provides a method of managing the content of several individual organizations under one management system. Here are some examples of organization management:

Single Organization

A small business with a simple system administration chain. In this case, you can create a single organization for the business and assign content to it.

Multiple Organizations

A large company that owns several smaller business units. For example, a company with separate system administration and software development groups. In this case, you can create organizations for the company and each of the business units it owns. This keeps the system infrastructure for each separate. You can then assign content to each organization based on its needs.

External Organizations

A company that manages external systems for other organizations. For example, a company offering cloud computing and web hosting resources to customers. In this case, you can create an organization for the company's own system infrastructure and then an organization for each external business. You can then assign content to each organization where necessary.

A default installation of Red Hat Satellite has a default organization called **Default Organization**.

New Users

If a new user is not assigned a default organization, their access is limited. To grant systems rights to users, assign them to a default organization. The next time the user logs on to Satellite, the user's account has the correct system rights.

5.1. CREATING AN ORGANIZATION

Use this procedure to create an organization. To use the CLI instead of the Satellite web UI, see [CLI procedure](#).

Procedure

1. In the Satellite web UI, navigate to **Administer > Organizations**.
2. Click **New Organization**.
3. In the **Name** field, enter a name for the organization.
4. In the **Label** field, enter a unique identifier for the organization. This is used for creating and mapping certain assets, such as directories for content storage. Use letters, numbers, underscores, and dashes, but no spaces.
5. Optional: in the **Description** field, enter a description for the organization.
6. Click **Submit**.
7. If you have hosts with no organization assigned, select the hosts that you want to add to the organization, then click **Proceed to Edit**

8. In the **Edit** page, assign the infrastructure resources that you want to add to the organization. This includes networking resources, installation media, kickstart templates, and other parameters. You can return to this page at any time by navigating to **Administer > Organizations** and then selecting an organization to edit.
9. Click **Submit**.

CLI procedure

1. To create an organization, enter the following command:

```
# hammer organization create \  
--name "My_Organization" \  
--label "My_Organization_Label" \  
--description "My_Organization_Description"
```

2. Optional: To edit an organization, enter the **hammer organization update** command. For example, the following command assigns a compute resource to the organization:

```
# hammer organization update \  
--name "My_Organization" \  
--compute-resource-ids 1
```

5.2. SETTING THE ORGANIZATION CONTEXT

An organization context defines the organization to use for a host and its associated resources.

Procedure

The organization menu is the first menu item in the menu bar, on the upper left of the Satellite web UI. If you have not selected a current organization, the menu says **Any Organization**. Click the **Any Organization** button and select the organization to use.

CLI procedure

While using the CLI, include either **--organization "My_Organization"** or **--organization-label "My_Organization_Label"** as an option. For example:

```
# hammer subscription list \  
--organization "My_Organization"
```

This command outputs subscriptions allocated for the *My_Organization*.

5.3. CREATING AN ORGANIZATION DEBUG CERTIFICATE

If you require a debug certificate for your organization, use the following procedure.

Procedure

1. In the Satellite web UI, navigate to **Administer > Organizations**.
2. Select an organization that you want to generate a debug certificate for.
3. Click **Generate and Download**.

4. Save the certificate file in a secure location.

Debug Certificates for Provisioning Templates

Debug Certificates are automatically generated for provisioning template downloads if they do not already exist in the organization for which they are being downloaded.

5.4. BROWSING REPOSITORY CONTENT USING AN ORGANIZATION DEBUG CERTIFICATE

You can view an organization's repository content using a web browser or using the API if you have a debug certificate for that organization.

Prerequisite

- You created and downloaded an organization certificate. For more information, see [Section 5.3, "Creating an Organization Debug Certificate"](#).

Procedure

1. Split the private and public keys from the certificate into two files.
 - a. Open the X.509 certificate, for example, for the default organization:


```
$ vi 'Default Organization-key-cert.pem'
```
 - b. Copy the contents of the file from **-----BEGIN RSA PRIVATE KEY-----** to **-----END RSA PRIVATE KEY-----**, into a **key.pem** file.
 - c. Copy the contents of the file from **-----BEGIN CERTIFICATE-----** to **-----END CERTIFICATE-----**, into a **cert.pem** file.
2. To use a browser, you must first convert the X.509 certificate to a format your browser supports and then import the certificate.

For Firefox Users

1. Convert the certificate into the PKCS12 format using the following command:

```
$ openssl pkcs12 -keypbe PBE-SHA1-3DES -certpbe PBE-SHA1-3DES -export -in cert.pem
-inkey key.pem -out My_Organization_Label.pfx -name My_Organization
```

2. In the Firefox browser, navigate to **Edit > Preferences > Advanced Tab**.
3. Select **View Certificates** and click the **Your Certificates** tab.
4. Click **Import** and select the **.pfx** file to load.
5. Enter the following URL in the address bar to browse the accessible paths for all the repositories and check their contents:

```
https://satellite.example.com/pulp/content/
```

For CURL Users

- To use the organization debug certificate with CURL, enter the following command:

```
$ curl -k --cert cert.pem --key key.pem \
https://satellite.example.com/pulp/content/My_Organization_Label/Library/content/dist/rhel/server/7/7Server/x86_64/os/
```

Ensure that the paths to **cert.pem** and **key.pem** are the correct absolute paths otherwise the command fails silently. Pulp uses the organization label, therefore, you must enter the organization label into the URL.

5.5. DELETING AN ORGANIZATION

You can delete an organization if the organization is not associated with any life cycle environments or host groups. If there are any life cycle environments or host groups associated with the organization you are about to delete, remove them by navigating to **Administer > Organizations** and clicking the relevant organization. Do not delete the default organization created during installation because the default organization is a placeholder for any unassociated hosts in the Satellite environment. There must be at least one organization in the environment at any given time.

Procedure

1. In the Satellite web UI, navigate to **Administer > Organizations**.
2. From the list to the right of the name of the organization you want to delete, select **Delete**.
3. Click **OK** to delete the organization.

CLI procedure

1. Enter the following command to retrieve the ID of the organization that you want to delete:

```
# hammer organization list
```

From the output, note the ID of the organization that you want to delete.

2. Enter the following command to delete an organization:

```
# hammer organization delete --id Organization_ID
```


CHAPTER 6. MANAGING LOCATIONS

Locations function similar to organizations: they provide a method to group resources and assign hosts. Organizations and locations have the following conceptual differences:

- Locations are based on physical or geographical settings.
- Locations have a hierarchical structure.

6.1. CREATING A LOCATION

Use this procedure to create a location so that you can manage your hosts and resources by location. To use the CLI instead of the Satellite web UI, see the [CLI procedure](#).

Procedure

1. In the Satellite web UI, navigate to **Administer > Locations**.
2. Click **New Location**.
3. Optional: from the **Parent** list, select a parent location. This creates a location hierarchy.
4. In the **Name** field, enter a name for the location.
5. Optional: in the **Description** field, enter a description for the location.
6. Click **Submit**.
7. If you have hosts with no location assigned, add any hosts that you want to assign to the new location, then click **Proceed to Edit**.
8. Assign any infrastructure resources that you want to add to the location. This includes networking resources, installation media, kickstart templates, and other parameters. You can return to this page at any time by navigating to **Administer > Locations** and then selecting a location to edit.
9. Click **Submit** to save your changes.

CLI procedure

- Enter the following command to create a location:

```
# hammer location create \  
--description "My_Location_Description" \  
--name "My_Location" \  
--parent-id "My_Location_Parent_ID"
```

6.2. CREATING MULTIPLE LOCATIONS

The following example Bash script creates three locations – London, Munich, Boston – and assigns them to the Example Organization.

```
ORG="Example Organization"  
LOCATIONS="London Munich Boston"
```

```

for LOC in ${LOCATIONS}
do
  hammer location create --name "${LOC}"
  hammer location add-organization --name "${LOC}" --organization "${ORG}"
done

```

6.3. SETTING THE LOCATION CONTEXT

A location context defines the location to use for a host and its associated resources.

Procedure

The location menu is the second menu item in the menu bar, on the upper left of the Satellite web UI. If you have not selected a current location, the menu displays **Any Location**. Click **Any location** and select the location to use.

CLI procedure

While using the CLI, include either `--location "My_Location"` or `--location-id "My_Location_ID"` as an option. For example:

```

# hammer subscription list \
--location "My_Location"

```

This command outputs subscriptions allocated for the *My_Location*.

6.4. DELETING A LOCATION

You can delete a location if the location is not associated with any life cycle environments or host groups. If there are any life cycle environments or host groups associated with the location you are about to delete, remove them by navigating to **Administer > Locations** and clicking the relevant location. Do not delete the default location created during installation because the default location is a placeholder for any unassociated hosts in the Satellite environment. There must be at least one location in the environment at any given time.

Procedure

1. In the Satellite web UI, navigate to **Administer > Locations**.
2. Select **Delete** from the list to the right of the name of the location you want to delete.
3. Click **OK** to delete the location.

CLI procedure

1. Enter the following command to retrieve the ID of the location that you want to delete:

```
# hammer location list
```

From the output, note the ID of the location that you want to delete.

2. Enter the following command to delete the location:

■ # hammer location delete --id *Location ID*

CHAPTER 7. MANAGING USERS AND ROLES

A User defines a set of details for individuals using the system. Users can be associated with organizations and environments, so that when they create new entities, the default settings are automatically used. Users can also have one or more *roles* attached, which grants them rights to view and manage organizations and environments. See [Section 7.1, “Managing Users”](#) for more information on working with users.

You can manage permissions of several users at once by organizing them into user groups. User groups themselves can be further grouped to create a hierarchy of permissions. For more information on creating user groups, see [Section 7.3, “Creating and Managing User Groups”](#).

Roles define a set of permissions and access levels. Each role contains one or more *permission filters* that specify the actions allowed for the role. Actions are grouped according to the *Resource type*. Once a role has been created, users and user groups can be associated with that role. This way, you can assign the same set of permissions to large groups of users. Satellite provides a set of predefined roles and also enables creating custom roles and permission filters as described in [Section 7.4, “Creating and Managing Roles”](#).

7.1. MANAGING USERS

As an administrator, you can create, modify and remove Satellite users. You can also configure access permissions for a user or a group of users by assigning them different *roles*.

7.1.1. Creating a User

Use this procedure to create a user. To use the CLI instead of the Satellite web UI, see the [CLI procedure](#).

Procedure

1. In the Satellite web UI, navigate to **Administer** > **Users**.
2. Click **Create User**.
3. In the **Login** field, enter a username for the user.
4. In the **Firstname** and **Lastname** fields, enter the real first name and last name of the user.
5. In the **Mail** field, enter the user’s email address.
6. In the **Description** field, add a description of the new user.
7. Select a specific language for the user from the **Language** list.
8. Select a timezone for the user from the **Timezone** list.
By default, Satellite Server uses the language and timezone settings of the user’s browser.
9. Set a password for the user:
 - a. From the **Authorized by** list, select the source by which the user is authenticated.
 - **INTERNAL**: to enable the user to be managed inside Satellite Server.
 - **EXTERNAL**: to configure external authentication as described in [Configuring External Authentication](#) in *Installing Satellite Server in a Connected Network Environment*.

- b. Enter an initial password for the user in the **Password** field and the **Verify** field.
10. Click **Submit** to create the user.

CLI procedure

- To create a user, enter the following command:

```
# hammer user create \
--auth-source-id My_Authentication_Source \
--login My_User_Name \
--mail My_User_Mail \
--organization-ids My_Organization_ID_1,My_Organization_ID_2 \
--password My_User_Password
```

The **--auth-source-id 1** setting means that the user is authenticated internally, you can specify an external authentication source as an alternative. Add the **--admin** option to grant administrator privileges to the user. Specifying organization IDs is not required, you can modify the user details later using the **update** subcommand.

For more information about user related subcommands, enter **hammer user --help**.

7.1.2. Assigning Roles to a User

Use this procedure to assign roles to a user. To use the CLI instead of the Satellite web UI, see the [CLI procedure](#).

Procedure

1. In the Satellite web UI, navigate to **Administer > Users**.
2. Click the **username** of the user to be assigned one or more roles.



NOTE

If a user account is not listed, check that you are currently viewing the correct organization. To list all the users in Satellite, click **Default Organization** and then **Any Organization**.

3. Click the **Locations** tab, and select a location if none is assigned.
4. Click the **Organizations** tab, and check that an organization is assigned.
5. Click the **Roles** tab to display the list of available roles.
6. Select the roles to assign from the **Roles** list.
To grant all the available permissions, select the **Admin** checkbox.
7. Click **Submit**.

To view the roles assigned to a user, click the **Roles** tab; the assigned roles are listed under **Selected items**. To remove an assigned role, click the role name in **Selected items**.

CLI procedure

- To assign roles to a user, enter the following command:

```
# hammer user add-role --id user_id --role role_name
```

7.1.3. Impersonating a Different User Account

Administrators can impersonate other authenticated users for testing and troubleshooting purposes by temporarily logging on to the Satellite web UI as a different user. When impersonating another user, the administrator has permissions to access exactly what the impersonated user can access in the system, including the same menus.

Audits are created to record the actions that the administrator performs while impersonating another user. However, all actions that an administrator performs while impersonating another user are recorded as having been performed by the impersonated user.

Prerequisites

- Ensure that you are logged on to the Satellite web UI as a user with administrator privileges for Satellite.

Procedure

1. In the Satellite web UI, navigate to **Administer** > **Users**.
2. To the right of the user that you want to impersonate, from the list in the **Actions** column, select **Impersonate**.

When you want to stop the impersonation session, in the upper right of the main menu, click the impersonation icon.

7.1.4. Creating an API-Only User

You can create users that can interact only with the Satellite API.

Prerequisite

- You have created a user and assigned roles to them. Note that this user must be authorized internally. For more information, see [Creating a User](#) and [Assigning Roles to a User](#).

Procedure

1. Log in to your Satellite as admin.
2. Navigate to **Administer** > **Users** and select a user.
3. On the **User** tab, set a password. Do not save or communicate this password with others. You can create pseudo-random strings on your console:

```
# openssl rand -hex 32
```

4. On the **Personal Access Tokens** tab, click **Add Personal Access Token**
5. Enter a name for the token.

6. Optional: Set an expiration date.
7. Click **Submit** to create the API token.
8. Copy the API token.

7.2. MANAGING SSH KEYS

Adding SSH keys to a user allows deployment of SSH keys during provisioning. For information on deploying SSH keys during provisioning, see [Deploying SSH Keys during Provisioning](#) in *Provisioning Hosts*.

For information on SSH keys and SSH key creation, see [Using SSH-based Authentication](#) in *Red Hat Enterprise Linux 8 Configuring basic system settings*.

7.2.1. Managing SSH Keys for a User

Use this procedure to add or remove SSH keys for a user. To use the CLI instead of the Satellite web UI, see the [CLI procedure](#).

Prerequisites

- Ensure that you are logged in to the Satellite web UI as an Admin user of Red Hat Satellite or a user with the `create_ssh_key` permission enabled for adding SSH key and `destroy_ssh_key` permission for removing a key.

Procedure

1. In the Satellite web UI, navigate to **Administer** > **Users**.
2. From the **Username** column, click on the username of the required user.
3. Click on the **SSH Keys** tab.
 - To Add SSH key
 - i. Prepare the content of the public SSH key in a clipboard.
 - ii. Click **Add SSH Key**.
 - iii. In the **Key** field, paste the public SSH key content from the clipboard.
 - iv. In the **Name** field, enter a name for the SSH key.
 - v. Click **Submit**.
 - To Remove SSH key
 - i. Click **Delete** on the row of the SSH key to be deleted.
 - ii. Click **OK** in the confirmation prompt.

CLI procedure

To add an SSH key to a user, you must specify either the path to the public SSH key file, or the content of the public SSH key copied to the clipboard.

- If you have the public SSH key file, enter the following command:

```
# hammer user ssh-keys add \
--user-id user_id \
--name key_name \
--key-file ~/.ssh/id_rsa.pub
```

- If you have the content of the public SSH key, enter the following command:

```
# hammer user ssh-keys add \
--user-id user_id \
--name key_name \
--key ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNtYAAABBBHHS2KmNyIYa27Qaa7
EHp+2I99ucGStx4P77e03ZvE3yVRJEFikpoP3MjtYYfle8k 1/46MTIZo9CPTX4CYUHeN8=
host@user
```

To delete an SSH key from a user, enter the following command:

```
# hammer user ssh-keys delete --id key_id --user-id user_id
```

To view an SSH key attached to a user, enter the following command:

```
# hammer user ssh-keys info --id key_id --user-id user_id
```

To list SSH keys attached to a user, enter the following command:

```
# hammer user ssh-keys list --user-id user_id
```

7.3. CREATING AND MANAGING USER GROUPS

7.3.1. User Groups

With Satellite, you can assign permissions to groups of users. You can also create user groups as collections of other user groups. If using an external authentication source, you can map Satellite user groups to external user groups as described in [Configuring External User Groups](#) in *Installing Satellite Server in a Connected Network Environment*.

User groups are defined in an organizational context, meaning that you must select an organization before you can access user groups.

7.3.2. Creating a User Group

Use this procedure to create a user group.

Procedure

1. In the Satellite web UI, navigate to **Administer > User Groups**.
2. Click **Create User group**.
3. On the **User Group** tab, specify the name of the new user group and select group members:

- Select the previously created user groups from the **User Groups** list.
 - Select users from the **Users** list.
4. On the **Roles** tab, select the roles you want to assign to the user group. Alternatively, select the **Admin** checkbox to assign all available permissions.
 5. Click **Submit**.

CLI procedure

- To create a user group, enter the following command:

```
# hammer user-group create \  
--name My_User_Group_Name \  
--role-ids My_Role_ID_1,My_Role_ID_2 \  
--user-ids My_User_ID_1,My_User_ID_2
```

7.3.3. Removing a User Group

Use the Satellite web UI to remove a user group.

Procedure

1. In the Satellite web UI, navigate to **Administer > User Groups**.
2. Click **Delete** to the right of the user group you want to delete.
3. In the alert box that appears, click **OK** to delete a user group.

7.4. CREATING AND MANAGING ROLES

Satellite provides a set of predefined roles with permissions sufficient for standard tasks, as listed in [Section 7.5, “Predefined Roles Available in Satellite”](#). It is also possible to configure custom roles, and assign one or more permission filters to them. Permission filters define the actions allowed for a certain resource type. Certain Satellite plug-ins create roles automatically.

7.4.1. Creating a Role

Use this procedure to create a role.

Procedure

1. In the Satellite web UI, navigate to **Administer > Roles**.
2. Click **Create Role**.
3. Provide a **Name** for the role.
4. Click **Submit** to save your new role.

CLI procedure

- To create a role, enter the following command:

```
# hammer role create --name My_Role_Name
```

To serve its purpose, a role must contain permissions. After creating a role, proceed to [Section 7.4.3, “Adding Permissions to a Role”](#).

7.4.2. Cloning a Role

Use the Satellite web UI to clone a role.

Procedure

1. In the Satellite web UI, navigate to **Administer** > **Roles** and select **Clone** from the drop-down menu to the right of the required role.
2. Provide a **Name** for the role.
3. Click **Submit** to clone the role.
4. Click the name of the cloned role and navigate to **Filters**.
5. Edit the permissions as required.
6. Click **Submit** to save your new role.

7.4.3. Adding Permissions to a Role

Use this procedure to add permissions to a role. To use the CLI instead of the Satellite web UI, see the [CLI procedure](#).

Procedure

1. In the Satellite web UI, navigate to **Administer** > **Roles**.
2. Select **Add Filter** from the drop-down list to the right of the required role.
3. Select the **Resource type** from the drop-down list. The *(Miscellaneous)* group gathers permissions that are not associated with any resource group.
4. Click the permissions you want to select from the **Permission** list.
5. Depending on the **Resource type** selected, you can select or deselect the **Unlimited** and **Override** checkbox. The **Unlimited** checkbox is selected by default, which means that the permission is applied on all resources of the selected type. When you disable the **Unlimited** checkbox, the **Search** field activates. In this field you can specify further filtering with use of the Satellite search syntax. For more information, see [Section 7.6, “Granular Permission Filtering”](#). When you enable the **Override** checkbox, you can add additional locations and organizations to allow the role to access the resource type in the additional locations and organizations; you can also remove an already associated location and organization from the resource type to restrict access.
6. Click **Next**.
7. Click **Submit** to save changes.

CLI procedure

1. List all available permissions:

```
# hammer filter available-permissions
```

2. Add permissions to a role:

```
# hammer filter create \
--permission-ids My_Permission_ID_1,My_Permission_ID_2\
--role My_Role_Name
```

For more information about roles and permissions parameters, enter the **hammer role --help** and **hammer filter --help** commands.

7.4.4. Viewing Permissions of a Role

Use the Satellite web UI to view the permissions of a role.

Procedure

1. In the Satellite web UI, navigate to **Administer** > **Roles**.
2. Click **Filters** to the right of the required role to get to the **Filters** page.

The **Filters** page contains a table of permissions assigned to a role grouped by the resource type. It is also possible to generate a complete table of permissions and actions that you can use on your Satellite system. For more information, see [Section 7.4.5, "Creating a Complete Permission Table"](#).

7.4.5. Creating a Complete Permission Table

Use the Satellite CLI to create a permission table.

Procedure

1. Ensure that the required packages are installed. Execute the following command on Satellite Server:

```
# satellite-maintain packages install foreman-console
```

2. Start the Satellite console with the following command:

```
# foreman-rake console
```

Insert the following code into the console:

```
f = File.open('/tmp/table.html', 'w')

result = Foreman::AccessControl.permissions [a,b] a.security_block <=>
b.security_block}.collect do |p|
  actions = p.actions.collect { |a| "<li>#{a}</li>" }
  "<tr><td>#{p.name}</td><td><ul>#{actions.join("</ul>#{p.resource_type}</td>
</tr>"
```

```
end.join("\n")
```

```
f.write(result)
```

The above syntax creates a table of permissions and saves it to the **/tmp/table.html** file.

3. Press **Ctrl + D** to exit the Satellite console. Insert the following text at the first line of **/tmp/table.html**:

```
<table border="1"><tr><td>Permission name</td><td>Actions</td><td>Resource type</td></tr>
```

Append the following text at the end of **/tmp/table.html**:

```
</table>
```

4. Open **/tmp/table.html** in a web browser to view the table.

7.4.6. Removing a Role

Use the Satellite web UI to remove a role.

Procedure

1. In the Satellite web UI, navigate to **Administer > Roles**.
2. Select **Delete** from the drop-down list to the right of the role to be deleted.
3. In an alert box that appears, click **OK** to delete the role.

7.5. PREDEFINED ROLES AVAILABLE IN SATELLITE

Role	Permissions Provided by Role ^[a]
Access Insights Admin	Add and edit Insights rules.
Access Insights Viewer	View Insight reports.
Ansible Roles Manager	Play roles on hosts and host groups. View, destroy, and import Ansible roles. View, edit, create, destroy, and import Ansible variables.
Ansible Tower Inventory Reader	View facts, hosts, and host groups.
Bookmarks manager	Create, edit, and delete bookmarks.
Boot disk access	Download the boot disk.

Role	Permissions Provided by Role ^[a]
Compliance manager	View, create, edit, and destroy SCAP content files, compliance policies, and tailoring files. View compliance reports.
Compliance viewer	View compliance reports.
Create ARF report	Create compliance reports.
Default role	The set of permissions that every user is granted, irrespective of any other roles.
Discovery Manager	View, provision, edit, and destroy discovered hosts and manage discovery rules.
Discovery Reader	View hosts and discovery rules.
Edit hosts	View, create, edit, destroy, and build hosts.
Edit partition tables	View, create, edit and destroy partition tables.
Manager	A role similar to administrator, but does not have permissions to edit global settings. In the Satellite web UI, global settings can be found under Administer > Settings .
Organization admin	An administrator role defined per organization. The role has no visibility into resources in other organizations.
Red Hat Access Logs	View the log viewer and the logs.
Remote Execution Manager	Control which roles have permission to run infrastructure jobs.
Remote Execution User	Run remote execution jobs against hosts.
Site manager	A restrained version of the Manager role.
System admin	<ul style="list-style-type: none"> ● Edit global settings in Administer > Settings. ● View, create, edit and destroy users, user groups, and roles. ● View, create, edit, destroy, and assign organizations and locations but not view resources within them. <p>Users with this role can create users and assign all roles to them. Therefore, ensure to give this role only to trusted users.</p>
Tasks manager	View and edit Satellite tasks.

Role	Permissions Provided by Role ^[a]
Tasks reader	A role that can only view Satellite tasks.
Viewer	A passive role that provides the ability to view the configuration of every element of the Satellite structure, logs, reports, and statistics.
View hosts	A role that can only view hosts.
Virt-who Manager	A role with full virt-who permissions.
Virt-who Reporter	Upload reports generated by virt-who to Satellite. It can be used if you configure virt-who manually and require a user role that has limited virt-who permissions.
Virt-who Viewer	View virt-who configurations. Users with this role can deploy virt-who instances using existing virt-who configurations.

[a] The exact set of allowed actions associated with predefined roles can be viewed by the privileged user as described in [Section 7.4.4, "Viewing Permissions of a Role"](#)

7.6. GRANULAR PERMISSION FILTERING

As mentioned in [Section 7.4.3, "Adding Permissions to a Role"](#), Red Hat Satellite provides the ability to limit the configured user permissions to selected instances of a resource type. These granular filters are queries to the Satellite database and are supported by the majority of resource types.

7.6.1. Creating a Granular Permission Filter

Use this procedure to create a granular filter. To use the CLI instead of the Satellite web UI, see the [CLI procedure](#).

Satellite does not apply search conditions to create actions. For example, limiting the `create_locations` action with `name = "Default Location"` expression in the search field does not prevent the user from assigning a custom name to the newly created location.

Procedure

Specify a query in the **Search** field on the **Edit Filter** page. Deselect the **Unlimited** checkbox for the field to be active. Queries have the following form:

field_name operator value

- *field_name* marks the field to be queried. The range of available field names depends on the resource type. For example, the *Partition Table* resource type offers *family*, *layout*, and *name* as query parameters.
- *operator* specifies the type of comparison between *field_name* and *value*. See [Section 7.6.3, "Supported Operators for Granular Search"](#) for an overview of applicable operators.

- *value* is the value used for filtering. This can be for example a name of an organization. Two types of wildcard characters are supported: underscore (_) provides single character replacement, while percent sign (%) replaces zero or more characters.

For most resource types, the **Search** field provides a drop-down list suggesting the available parameters. This list appears after placing the cursor in the search field. For many resource types, you can combine queries using logical operators such as *and*, *not* and *has* operators.

CLI procedure

- To create a granular filter, enter the **hammer filter create** command with the **--search** option to limit permission filters, for example:

```
# hammer filter create \
--permission-ids 91 \
--search "name ~ ccv*" \
--role qa-user
```

This command adds to the **qa-user** role a permission to view, create, edit, and destroy Content Views that only applies to Content Views with name starting with **ccv**.

7.6.2. Examples of Using Granular Permission Filters

As an administrator, you can allow selected users to make changes in a certain part of the environment path. The following filter allows you to work with content while it is in the development stage of the application life cycle, but the content becomes inaccessible once is pushed to production.

7.6.2.1. Applying Permissions for the Host Resource Type

The following query applies any permissions specified for the Host resource type only to hosts in the group named host-editors.

```
hostgroup = host-editors
```

The following query returns records where the name matches *XXXX*, *Yyyy*, or *zzzz* example strings:

```
name ^ (XXXX, Yyyy, zzzz)
```

You can also limit permissions to a selected environment. To do so, specify the environment name in the **Search** field, for example:

```
Dev
```

You can limit user permissions to a certain organization or location with the use of the granular permission filter in the **Search** field. However, some resource types provide a GUI alternative, an **Override** checkbox that provides the **Locations** and **Organizations** tabs. On these tabs, you can select from the list of available organizations and locations. For more information, see [Section 7.6.2.2, "Creating an Organization Specific Manager Role"](#).

7.6.2.2. Creating an Organization Specific Manager Role

Use the Satellite web UI to create an administrative role restricted to a single organization named *org-1*.

Procedure

1. In the Satellite web UI, navigate to **Administer** > **Roles**.
2. Clone the existing **Organization admin** role. Select **Clone** from the drop-down list next to the **Filters** button. You are then prompted to insert a name for the cloned role, for example `org-1 admin`.
3. Click the desired locations and organizations to associate them with the role.
4. Click **Submit** to create the role.
5. Click `org-1 admin`, and click **Filters** to view all associated filters. The default filters work for most use cases. However, you can optionally click **Edit** to change the properties for each filter. For some filters, you can enable the **Override** option if you want the role to be able to access resources in additional locations and organizations. For example, by selecting the **Domain** resource type, the **Override** option, and then additional locations and organizations using the **Locations** and **Organizations** tabs, you allow this role to access domains in the additional locations and organizations that is not associated with this role. You can also click **New filter** to associate new filters with this role.

7.6.3. Supported Operators for Granular Search

Table 7.1. Logical Operators

Operator	Description
and	Combines search criteria.
not	Negates an expression.
has	Object must have a specified property.

Table 7.2. Symbolic Operators

Operator	Description
=	<i>Is equal to.</i> An equality comparison that is case-sensitive for text fields.
!=	<i>Is not equal to.</i> An inversion of the = operator.
~	<i>Like.</i> A case-insensitive occurrence search for text fields.
!~	<i>Not like.</i> An inversion of the ~ operator.
^	<i>In.</i> An equality comparison that is case-sensitive search for text fields. This generates a different SQL query to the <i>Is equal to</i> comparison, and is more efficient for multiple value comparison.
!^	<i>Not in.</i> An inversion of the ^ operator.

>, >=	<i>Greater than, greater than or equal to</i> Supported for numerical fields only.
<, <=	<i>Less than, less than or equal to</i> Supported for numerical fields only.

CHAPTER 8. CONFIGURING EMAIL NOTIFICATIONS

Email notifications are created by Satellite Server periodically or after completion of certain events. The periodic notifications can be sent daily, weekly or monthly.

For the events that trigger a notification and notification types, see [Section 8.1, “Notification Types”](#).

Users do not receive any email notifications by default. An administrator can configure users to receive notifications based on criteria such as the type of notification, and frequency.



IMPORTANT

Satellite Server does not enable outgoing emails by default, therefore you must review your email configuration. For more information, see [Configuring Satellite Server for Outgoing Emails](#) in *Installing Satellite Server in a Connected Network Environment*.

8.1. NOTIFICATION TYPES

The following are the notifications created by Satellite:

- **Audit summary:** A summary of all activity audited by Satellite Server.
- **Host built:** A notification sent when a host is built.
- **Host errata advisory:** A summary of applicable and installable errata for hosts managed by the user.
- **OpenSCAP policy summary:** A summary of OpenSCAP policy reports and their results.
- **Promote errata:** A notification sent only after a Content View promotion. It contains a summary of errata applicable and installable to hosts registered to the promoted Content View. This allows a user to monitor what updates have been applied to which hosts.
- **Puppet error state:** A notification sent after a host reports an error related to Puppet.
- **Puppet summary:** A summary of Puppet reports.
- **Sync errata:** A notification sent only after synchronizing a repository. It contains a summary of new errata introduced by the synchronization.

8.2. CONFIGURING EMAIL NOTIFICATION PREFERENCES

You can configure Satellite to send email messages to individual users registered to Satellite. Satellite sends the email to the email address that has been added to the account, if present. Users can edit the email address by clicking on their name in the top-right of the Satellite web UI and selecting **My account**.

Configure email notifications for a user from the Satellite web UI.



NOTE

If you want to send email notifications to a group email address instead of an individual email address, create a user account with the group email address and minimal Satellite permissions, then subscribe the user account to the desired notification types.

Procedure

1. In the Satellite web UI, navigate to **Administer > Users**.
2. Click the **Username** of the user you want to edit.
3. On the **User** tab, verify the value of the **Mail** field. Email notifications will be sent to the address in this field.
4. On the **Email Preferences** tab, select **Mail Enabled**.
5. Select the notifications you want the user to receive using the drop-down menus next to the notification types.

**NOTE**

The **Audit Summary** notification can be filtered by entering the required query in the **Mail Query** text box.

6. Click **Submit**.
The user will start receiving the notification emails.

8.3. TESTING EMAIL DELIVERY

To verify the delivery of emails, send a test email to a user. If the email gets delivered, the settings are correct.

Procedure

1. In the Satellite web UI, navigate to **Administer > Users**.
2. Click on the username.
3. On the **Email Preferences** tab, click **Test email**.
A test email message is sent immediately to the user's email address.

If the email is delivered, the verification is complete. Otherwise, you must perform the following diagnostic steps:

- a. Verify the user's email address.
- b. Verify Satellite Server's email configuration.
- c. Examine firewall and mail server logs.

8.4. TESTING EMAIL NOTIFICATIONS

To verify that users are correctly subscribed to notifications, trigger the notifications manually.

Procedure

- To trigger the notifications, execute the following command:

```
# foreman-rake reports:_My_Frequency_
```

Replace *My_Frequency* with one of the following:

- daily
- weekly
- monthly

This triggers all notifications scheduled for the specified frequency for all the subscribed users. If every subscribed user receives the notifications, the verification succeeds.



NOTE

Sending manually triggered notifications to individual users is currently not supported.

8.5. CHANGING EMAIL NOTIFICATION SETTINGS FOR A HOST

Satellite can send event notifications for a host to the host's registered owner. You can configure Satellite to send email notifications either to an individual user or a user group. When set to a user group, all group members who are subscribed to the email type receive a message.

To view the notification status for a host, navigate to **Hosts > All Hosts** and click the host you want to view. In the host details page, click the **Additional Information** tab, you can view the email notification status.

Receiving email notifications for a host can be useful, but also overwhelming if you are expecting to receive frequent errors, for example, because of a known issue or error you are working around. To change the email notification settings for a host, complete the following steps.

Procedure

1. In the Satellite web UI, navigate to **Hosts > All Hosts**, and select the host with the notification setting you want to change.
2. Select the host's checkbox, and from the **Select Action** list, select **Enable Notifications** or **Disable Notifications**, depending on what you want.

CHAPTER 9. MANAGING SECURITY COMPLIANCE

Security compliance management is the ongoing process of defining security policies, auditing for compliance with those policies and resolving instances of non-compliance. Any non-compliance is managed according to the organization's configuration management policies. Security policies range in scope from host-specific to industry-wide, therefore, flexibility in their definition is required.

9.1. SECURITY CONTENT AUTOMATION PROTOCOL

Satellite uses the Security Content Automation Protocol (SCAP) to define security configuration policies. For example, a security policy might specify that for hosts running Red Hat Enterprise Linux, login via SSH is not permitted for the **root** account. With Satellite, you can schedule compliance auditing and reporting on all managed hosts. For more information about SCAP, see [Red Hat Enterprise Linux 8 Security hardening](#).

9.1.1. SCAP Content

SCAP content is a datastream format containing the configuration and security baseline against which hosts are checked. Checklists are described in the extensible checklist configuration description format (XCCDF) and vulnerabilities in the open vulnerability and assessment language (OVAL). Checklist items, also known as rules express the desired configuration of a system item. For example, you may specify that no one can log in to a host over SSH using the **root** user account. Rules can be grouped into one or more profiles, allowing multiple profiles to share a rule. SCAP content consists of both rules and profiles.

You can either create SCAP content or obtain it from a vendor. Supported profiles are provided for Red Hat Enterprise Linux in the **scap-security-guide** package. The creation of SCAP content is outside the scope of this guide, but see [Red Hat Enterprise Linux 8 Security hardening](#) for information on how to download, deploy, modify, and create your own content.

The default SCAP content provided with the OpenSCAP components of Satellite depends on the version of Red Hat Enterprise Linux. On Red Hat Enterprise Linux 7, content for both Red Hat Enterprise Linux 6 and Red Hat Enterprise Linux 7 is installed.

9.1.2. XCCDF Profile

An XCCDF profile is a checklist against which a host or host group is evaluated. Profiles are created to verify compliance with an industry standard or custom standard.

The profiles provided with Satellite are obtained from the [OpenSCAP project](#).

9.1.3. Listing Available XCCDF Profiles

In the Satellite web UI, list the available XCCDF profiles.

Procedure

- In the Satellite web UI, navigate to **Hosts > SCAP contents**.

9.2. INSTALLING THE OPENSAP PLUG-IN

You can install and enable the OpenSCAP plug-in to generate OpenSCAP compliance reports. The OpenSCAP plug-in consists of the main OpenSCAP plug-in itself, the OpenSCAP smart proxy plug-in, and the OpenSCAP Hammer CLI plug-in.

Procedure

1. Install the OpenSCAP plug-in on your Satellite Server:

```
# satellite-installer --enable-foreman-plugin-openscap --enable-foreman-proxy-plugin-openscap
```

2. Install the OpenSCAP plug-in on any Capsule Servers:

```
# satellite-installer --enable-foreman-proxy-plugin-openscap
```

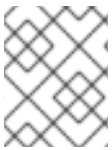
3. Install the OpenSCAP plug-in Puppet module:

```
# dnf install puppet-foreman_scap_client
```

4. In the Satellite web UI, navigate to **Configure > Puppet Classes**
5. Click **Import environments from satellite.example.com**.
You can use Puppet to install and configure the OpenSCAP plug-in on your Satellite Server and Capsules.

9.3. CONFIGURING SCAP CONTENT

9.3.1. Importing OpenSCAP Puppet Modules



NOTE

If you do not use Puppet to configure OpenSCAP auditing on hosts, you can skip this procedure.

To audit hosts with OpenSCAP, you must first import a Puppet environment. The Puppet environment contains the Puppet classes you must assign to each host to deploy the OpenSCAP configuration.

You must associate each host that you want to audit with the Puppet environment in the Satellite web UI.

Procedure

1. In the Satellite web UI, navigate to **Configure > Environments**.
2. Click **Import environments from satellite.example.com**.
3. Select the Puppet environment checkbox associated with the host you want to audit.
If no Puppet environment exists, select the **production** environment checkbox. The Puppet classes that you require for OpenSCAP are in the **production** environment by default.
4. Click **Update**.

9.3.2. Loading the Default OpenSCAP Content

In the CLI, load the default OpenSCAP content using one of the following methods.

Procedure

- Use the Hammer command:

```
# hammer scap-content bulk-upload --type default
```

- (Deprecated) Use the **foreman-rake** command:

```
# foreman-rake foreman_openscap:bulk_upload:default
```

9.3.3. Extra SCAP Content

You can upload extra SCAP content into Satellite Server, either content created by yourself or obtained elsewhere. SCAP content must be imported into Satellite Server before being applied in a policy.

For example, the **scap-security-guide** RPM package available in the Red Hat Enterprise Linux repositories includes a profile for the Payment Card Industry Data Security Standard (PCI-DSS) version 3. You can upload this content into a Satellite Server even if it is not running Red Hat Enterprise Linux as the content is not specific to an operating system version.

9.3.4. Uploading Extra SCAP Content

In the Satellite web UI, upload the extra SCAP content. To use the CLI instead of the Satellite web UI, see the [CLI procedure](#).

Procedure

1. In the Satellite web UI, navigate to **Hosts > SCAP contents** and click **New SCAP Content**.
2. Enter a title in the **Title** text box.
Example: **RHEL 7.2 SCAP Content**.
3. Click **Choose file**, navigate to the location containing the SCAP content file and select **Open**.
4. Click **Submit**.

If the SCAP content file is loaded successfully, a message similar to **Successfully created RHEL 7.2 SCAP Content** is shown and the list of **SCAP Contents** includes the new title.

CLI procedure

1. To upload SCAP content to your Satellite Server, enter the following command:

```
# hammer scap-content bulk-upload \
--directory /usr/share/xml/scap/ssg/content/ \
--location "_My_Location_" \
--organization "_My_Organization_" \
--type directory
```

SCAP content in **/usr/share/xml/scap/ssg/content/** is part of the **scap-security-guide** package.

9.4. MANAGING COMPLIANCE POLICIES

A scheduled audit, also known as a *compliance policy*, is a scheduled task that checks the specified hosts for compliance against an XCCDF profile. The schedule for scans is specified by Satellite Server and the scans are performed on the host. When a scan completes, an *Asset Reporting File* (ARF) is generated in XML format and uploaded to Satellite Server. You can see the results of the scan in the compliance policy dashboard. No changes are made to the scanned host by the compliance policy. The SCAP content includes several profiles with associated rules but policies are not included by default.

9.4.1. Creating a Compliance Policy

With Satellite, you can create a compliance policy to scan your content hosts to ensure that the hosts remain compliant to your security requirements.

You can use either Puppet or Ansible to deploy the compliance policy to your hosts. Note that Puppet runs by default every 30 minutes. If you assign a new policy, the next Puppet run synchronizes the policy to the host. However Ansible does not perform scheduled runs. To add a new policy, you must run Ansible role manually or using remote execution. For more information about remote execution, see [Configuring and Setting up Remote Jobs](#) in *Managing Hosts*.

Prerequisites

Before you begin, you must decide whether you want to use a Puppet or Ansible deployment.

- For Puppet deployment, ensure that each host that you want to audit is associated with a Puppet environment. For more information, see [Section 9.3.1, "Importing OpenSCAP Puppet Modules"](#).
- For Ansible deployment, ensure that you import the **theforeman.foreman_scap_client** Ansible role. For more information about importing Ansible roles, see [Getting Started with Ansible in Satellite](#) in *Managing Configurations Using Ansible Integration in Red Hat Satellite*.

Procedure

1. In the Satellite web UI, navigate to **Hosts > Policies**, and select whether you want a manual, Ansible, or Puppet deployment.
2. Enter a name for this policy, a description (optional), then click **Next**.
3. Select the SCAP Content and XCCDF Profile to be applied, then click **Next**.
Note that the openSCAP plugin does not detect if a SCAP content role has no content, which means that the **Default XCCDF Profile** might return an empty report.
4. Specify the scheduled time when the policy is to be applied, then click **Next**.
Select **Weekly**, **Monthly**, or **Custom** from the **Period** list.
 - If you select **Weekly**, also select the desired day of the week from the **Weekday** list.
 - If you select **Monthly**, also specify the desired day of the month in the **Day of month** field.
 - If you select **Custom**, enter a valid Cron expression in the **Cron line** field.
The **Custom** option allows for greater flexibility in the policy's schedule than either the **Weekly** or **Monthly** options.
5. Select the locations to which the policy is to be applied, then click **Next**.
6. Select the organizations to which the policy is to be applied, then click **Next**.
7. Select the host groups to which the policy is to be applied, then click **Submit**.

When the Puppet agent runs on the hosts which belong to the selected host group, or hosts to which the policy has been applied, the OpenSCAP client will be installed and a Cron job added with the policy's specified schedule. The **SCAP Content** tab provides the name of the SCAP content file which will be distributed to the directory `/var/lib/openscap/content/` on all target hosts.

9.4.2. Viewing a Compliance Policy

You can preview the rules which will be applied by specific OpenSCAP content and profile combination. This is useful when planning policies.

Procedure

1. In the Satellite web UI, navigate to **Hosts > Policies**.
2. Click **Show Guide**.

9.4.3. Editing a Compliance Policy

In the Satellite web UI, you can edit compliance policies.

Procedure

1. In the Satellite web UI, navigate to **Hosts > Policies**.
2. From the drop-down list to the right of the policy's name, select **Edit**.
3. Edit the necessary attributes.
4. Click **Submit**.

An edited policy is applied to the host when its Puppet agent next checks with Satellite Server for updates. By default, this occurs every 30 minutes.

9.4.4. Deleting a Compliance Policy

In the Satellite web UI, you can delete existing compliance policies.

Procedure

1. In the Satellite web UI, navigate to **Hosts > Policies**.
2. From the drop-down list to the right of the policy's name, select **Delete**.
3. Click **OK** in the confirmation message.

9.5. CUSTOMIZING OPENSAP POLICIES

You can customize existing OpenSCAP policies using Tailoring Files without forking or rewriting the policy. You can assign a Tailoring File to a policy when creating or updating a policy.

You can create a Tailoring File using the [SCAP Workbench](#). For more information on using the SCAP Workbench tool, see [Customizing SCAP Security Guide for your use-case](#).

9.5.1. Uploading a Tailoring File

In the Satellite web UI, you can upload a Tailoring file.

Procedure

1. In the Satellite web UI, navigate to **Hosts > Compliance – Tailoring Files** and click **New Tailoring File**.
2. Enter a name in the **Name** text box.
3. Click **Choose File**, navigate to the location containing the SCAP DataStream Tailoring File and select **Open**.
4. Click **Submit** to upload the chosen Tailoring File.

9.5.2. Assigning a Tailoring File to a Policy

In the Satellite web UI, assign a Tailoring file to a policy.

Procedure

1. In the Satellite web UI, navigate to **Hosts > Compliance – Policies**.
2. Click **New Policy**, or **New Compliance Policy** if there are existing Compliance Policies.
3. Enter a name in the **Name** text box, and click **Next**.
4. Select a **Scap content** from the dropdown menu.
5. Select a **XCCDF Profile** from the dropdown menu.
6. Select a **Tailoring File** from the dropdown menu.
7. Select a **XCCDF Profile in Tailoring File** from the dropdown menu.
It is important to select the XCCDF Profile because Tailoring Files are able to contain multiple XCCDF Profiles.
8. Click **Next**.
9. Select a **Period** from the dropdown menu.
10. Select a **Weekday** from the dropdown menu, and click **Next**.
11. Select a **Location** to move it to the **Selected Items** window, and click **Next**.
12. Select an **Organization** to move it to the **Selected Items** window, and click **Next**.
13. Select a **Hostgroup** to move it to the **Selected Items** window, and click **Submit**.

9.6. CONFIGURING A HOST GROUP FOR OPENSCAP

Use this procedure to configure all the OpenSCAP requirements for a host group.

Prerequisites

- Enable OpenSCAP on Capsule. For more information, see [Enabling OpenSCAP on External Capsules](#) in *Installing Capsule Server*.

- Assign an OpenSCAP Capsule.
- Assign a Puppet environment that contains the Puppet classes to deploy the OpenSCAP policies.
- Assign the **foreman_scap_client** and **foreman_scap_client::params** Puppet classes.
- Assign any compliance policies that you want to add.

For information about creating and administering hosts, see the [Managing Hosts](#) guide.

Procedure

1. In the Satellite web UI, navigate to **Configure > Host Groups**, and either create a host group or click the host group that you want to configure for OpenSCAP reporting.
2. From the **Puppet Environment** list, select the Puppet environment that contains the **foreman_scap_client** and **foreman_scap_client::params** Puppet classes.
3. From the **OpenSCAP Capsule** list, select the Capsule with OpenSCAP enabled that you want to use.
4. Click the **Puppet Classes** tab, and add the **foreman_scap_client** and **foreman_scap_client::params** Puppet classes.
5. Click **Submit** to save your changes.
6. In the Satellite web UI, navigate to **Hosts > Policies**.
7. Select the policy that you want to assign to the host group.
8. Click the **Host Groups** tab.
9. From the **Host Groups** list, select as many host groups as you want to assign to this policy.
10. Click **Submit** to save your changes.

CHAPTER 10. RUNNING OPENS CAP SCANS

Procedure

1. In the Satellite web UI, navigate to **Hosts > All Hosts**
2. Select one or multiple hosts.
3. Click on **Run OpenSCAP scan**
Alternatively, [schedule a remote job](#) to scan one or multiple hosts.

10.1. CONFIGURING A HOST FOR OPENS CAP

Use this procedure to configure all the OpenSCAP requirements for a host.

Prerequisites

- Enable OpenSCAP on Capsule. For more information, see [Enabling OpenSCAP on External Capsules](#) in *Installing Capsule Server*.
- Assign an OpenSCAP Capsule.
- Assign a Puppet environment that contains the Puppet classes to deploy the OpenSCAP policies.
- Assign the **foreman_scap_client** and **foreman_scap_client::params** Puppet classes.
- Assign any compliance policies that you want to add.

For information about creating and administering hosts, see the [Managing Hosts](#) guide.

Procedure

1. In the Satellite web UI, navigate to **Hosts > All Hosts**, and select **Edit** on the host you want to configure for OpenSCAP reporting.
2. From the **Puppet Environment** list, select the Puppet environment that contains the **foreman_scap_client** and **foreman_scap_client::params** Puppet classes.
3. From the **OpenSCAP Capsule** list, select the Capsule with OpenSCAP enabled that you want to use.
4. Click the **Puppet Classes** tab, and add the **foreman_scap_client** and **foreman_scap_client::params** Puppet classes.
5. To add a compliance policy, navigate to one of the following locations:
6. In the Satellite web UI, navigate to **Hosts > All Hosts**.
7. Select the host or hosts to which you want to add the policy.
8. Click **Select Action**.
9. Select **Assign Compliance Policy** from the list.

10. In the Policy window, select the policy that you want from the list of available policies and click **Submit**.

10.2. MONITORING COMPLIANCE

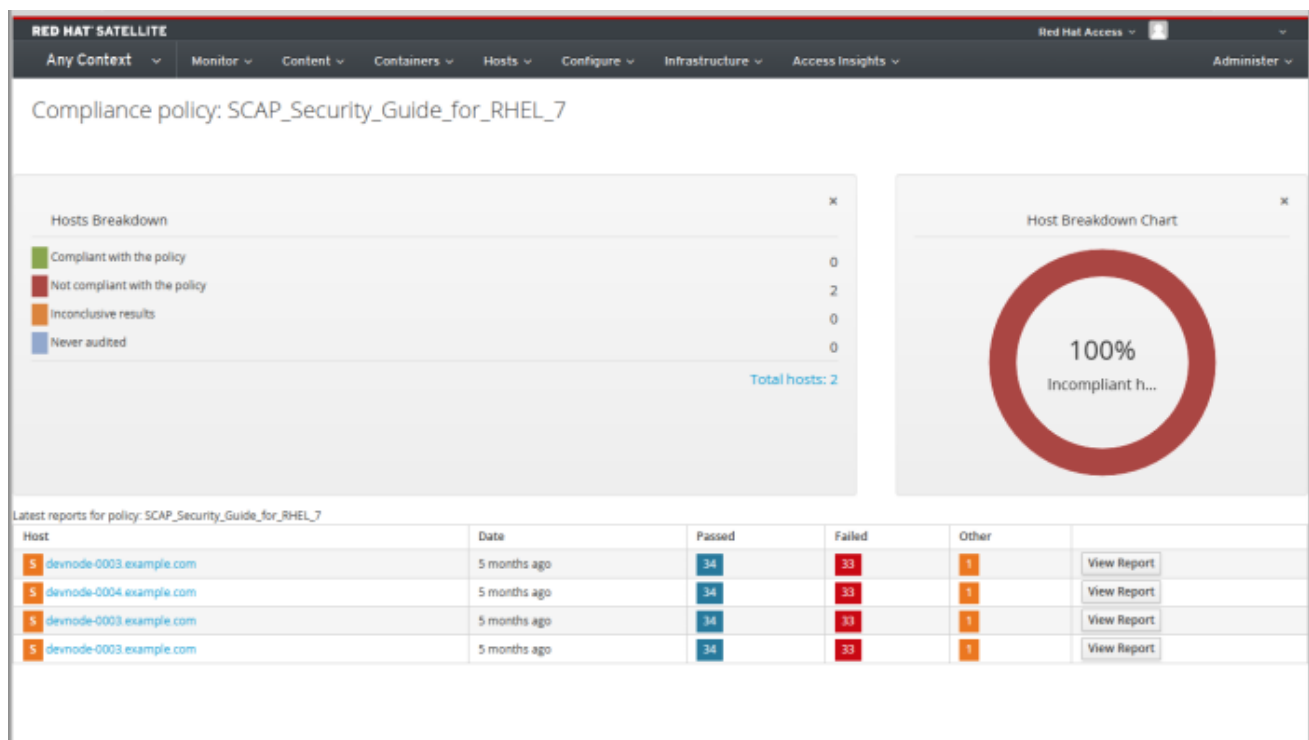
Satellite enables centralized compliance monitoring and management. A compliance dashboard provides an overview of compliance of hosts and the ability to view details for each host within the scope of that policy. Compliance reports provide a detailed analysis of compliance of each host with the applicable policy. With this information, you can evaluate the risks presented by each host and manage the resources required to bring hosts into compliance.

Common objectives when monitoring compliance using SCAP include the following:

- Verifying policy compliance.
- Detecting changes in compliance.

10.2.1. Compliance Policy Dashboard

The compliance policy dashboard provides a statistical summary of compliance of hosts and the ability to view details for each host within the scope of that policy. For all hosts which were evaluated as non-compliant, the **Failed** statistic provides a useful metric for prioritizing compliance effort. The hosts detected as **Never audited** should also be a priority, since their status is unknown.



10.2.2. Viewing the Compliance Policy Dashboard

Use the Satellite web UI to verify policy compliance with the compliance policy dashboard.

Procedure

1. In the Satellite web UI, navigate to **Hosts > Policies**.
2. Click the required policy name. The dashboard provides the following information:

- A ring chart illustrating a high-level view of compliance of hosts with the policy.
- A statistical breakdown of compliance of hosts with the policy, in a tabular format.
- Links to the latest policy report for each host.

10.2.3. Compliance Email Notifications

Satellite Server sends an OpenSCAP Summary email to all users who subscribe to the **Openscap policy summary** email notifications. For more information on subscribing to email notifications, see [Section 8.2, "Configuring Email Notification Preferences"](#). Each time a policy is run, Satellite checks the results against the previous run, noting any changes between them. The email is sent according to the frequency requested by each subscriber, providing a summary of each policy and its most recent result.

An **OpenSCAP Summary** email message contains the following information:

- Details of the time period it covers.
- Totals for all hosts by status: changed, compliant, and noncompliant.
- A tabular breakdown of each host and the result of its latest policy, including totals of the rules that passed, failed, changed, or where results were unknown.

10.2.4. Compliance Reports

A compliance report is the output of a policy run against a host. Each report includes the total number of rules passed or failed per policy. By default, reports are listed in descending date order.

In the Satellite web UI, navigate to **Hosts > Reports** to list all compliance reports.

A compliance report consists of the following areas:

- Introduction
- Evaluation Characteristics
- Compliance and Scoring
- Rule Overview

Evaluation Characteristics

The Evaluation Characteristics area provides details about an evaluation against a specific profile, including the host that was evaluated, the profile used in the evaluation, and when the evaluation started and finished. For reference, the IPv4, IPv6, and MAC addresses of the host are also listed.

Name	Description	Example
Target machine	The fully-qualified domain name (FQDN) of the evaluated host.	test-system.example.com
Benchmark URL	The URL of the SCAP content against which the host was evaluated.	/var/lib/openscap/content/1fbdc87d24db51ca184419a2b6f

Name	Description	Example
Benchmark ID	The identifier of the benchmark against which the host was evaluated. A benchmark is a set of profiles	xccdf_org.ssgproject.content_benchmark_RHEL_7
Profile ID	The identifier of the profile against which the host was evaluated.	xccdf_org.ssgproject_content_profile_rht-ccp
Started at	The date and time at which the evaluation started, in ISO 8601 format.	2015-09-12T14:40:02
Finished at	The date and time at which the evaluation finished, in ISO 8601 format.	2015-09-12T14:40:05
Performed by	The local account name under which the evaluation was performed on the host.	root

Compliance and Scoring

The Compliance and Scoring area provides an overview of whether or not the host is in compliance with the profile rules, a breakdown of compliance failures by severity, and an overall compliance score as a percentage. If compliance with a rule was not checked, this is categorized in the **Rule results** field as **Other**.

Rule Overview

The Rule Overview area provides details about every rule and the compliance result, with the rules presented in a hierarchical layout.

Select or clear the checkboxes to narrow the list of rules included in the compliance report. For example, if the focus of your review is any non-compliance, clear the **pass** and **informational** checkboxes.

To search all rules, enter a criterion in the **Search** field. The search is dynamically applied as you type. The **Search** field only accepts a single plain-text search term and it is applied as a case-insensitive search. When you perform a search, only those rules whose descriptions match the search criterion will be listed. To remove the search filter, delete the search criterion.

For an explanation of each result, hover the cursor over the status shown in the **Result** column.

10.2.5. Examining Compliance Failures of Hosts

Use the Satellite web UI to determine why a host failed compliance on a rule.

Procedure

1. In the Satellite web UI, navigate to **Hosts > Reports** to list all compliance reports.
2. Click **View Report** in the row of the specific host to view the details of an individual report.
3. Click on the rule's title to see further details:

- A description of the rule with instructions for bringing the host into compliance if available.
- The rationale for the rule.
- In some cases, a remediation script.



WARNING

Do not implement any of the recommended remedial actions or scripts without first testing them in a non-production environment.

10.2.6. Searching Compliance Reports

Use the Compliance Reports search field to filter the list of available reports on any given subset of hosts.

Procedure

- To apply a filter, enter the search query in the **Search** field and click **Search**. The search query is case insensitive.

Search Use Cases

- The following search query finds all compliance reports for which more than five rules failed:

```
failed > 5
```

- The following search query finds all compliance reports created after January 1, YYYY, for hosts with host names that contain the **prod-** group of characters:

```
host ~ prod- AND date > "Jan 1, YYYY"
```

- The following search query finds all reports generated by the **rhel7_audit** compliance policy from an hour ago:

```
"1 hour ago" AND compliance_policy = date = "1 hour ago" AND compliance_policy = rhel7_audit
```

- The following search query finds reports that pass an XCCDF rule:

```
xccdf_rule_passed = xccdf_org.ssgproject.content_rule_firefox_preferences-auto-download_actions
```

- The following search query finds reports that fail an XCCDF rule:

```
xccdf_rule_failed = xccdf_org.ssgproject.content_rule_firefox_preferences-auto-download_actions
```


- The following search query finds reports that have a result different than fail or pass for an XCCDF rule:

```
xccdf_rule_others = xccdf_org.ssgproject.content_rule_firefox_preferences-auto-download_actions
```

Additional Information

- To see a list of available search parameters, click the empty **Search** field.
- You can create complex queries with the following logical operators: **and**, **not** and **has**. For more information about logical operators, see [Section 7.6.3, "Supported Operators for Granular Search"](#).
- You cannot use regular expressions in a search query. However, you can use multiple fields in a single search expression. For more information about all available search operators, see [Section 7.6.3, "Supported Operators for Granular Search"](#).
- You can bookmark a search to reuse the same search query. For more information, see [Section 18.3.1, "Creating Bookmarks"](#).

10.2.7. Deleting a Compliance Report

You can delete compliance reports on your Satellite.

Procedure

1. In the Satellite web UI, navigate to **Hosts > Reports**.
2. In the Compliance Reports window, identify the policy that you want to delete and, on the right of the policy's name, select **Delete**.
3. Click **OK**.

10.2.8. Deleting Multiple Compliance Reports

You can delete multiple compliance policies simultaneously. However, in the Satellite web UI, compliance policies are paginated, so you must delete one page of reports at a time. If you want to delete all OpenSCAP reports, use the script in the [Deleting OpenSCAP Reports](#) section of *API Guide*.

Procedure

1. In the Satellite web UI, navigate to **Hosts > Reports**.
2. In the Compliance Reports window, select the compliance reports that you want to delete.
3. In the upper right of the list, select **Delete reports**.
4. Repeat these steps for as many pages as you want to delete.

10.3. SPECIFICATIONS SUPPORTED BY OPENSAP

The following specifications are supported by OpenSCAP:

Title	Description	Version
XCCDF	The Extensible Configuration Checklist Description Format	1.2
OVAL	Open Vulnerability and Assessment Language	5.11
-	Asset Identification	1.1
ARF	Asset Reporting Format	1.1
CCE	Common Configuration Enumeration	5.0
CPE	Common Platform Enumeration	2.3
CVE	Common Vulnerabilities and Exposures	-
CVSS	Common Vulnerability Scoring System	2.0

CHAPTER 11. BACKING UP SATELLITE SERVER AND CAPSULE SERVER

You can back up your Satellite deployment to ensure the continuity of your Red Hat Satellite deployment and associated data in the event of a disaster. If your deployment uses custom configurations, you must consider how to handle these custom configurations when you plan your backup and disaster recovery policy.

To create a backup of your Satellite Server or Capsule Server and all associated data, use the **satellite-maintain backup** command. Backing up to a separate storage device on a separate system is highly recommended.

Satellite services are unavailable during the backup. Therefore, you must ensure that no other tasks are scheduled by other administrators. You can schedule a backup using **cron**. For more information, see the [Section 11.5, “Example of a Weekly Full Backup Followed by Daily Incremental Backups”](#) .

During offline or snapshot backups, the services are inactive and Satellite is in a maintenance mode. All the traffic from outside on port 443 is rejected by a firewall to ensure there are no modifications triggered.

A backup contains sensitive information from the **/root/ssl-build** directory. For example, it can contain hostnames, ssh keys, request files and SSL certificates. You must encrypt or move the backup to a secure location to minimize the risk of damage or unauthorized access to the hosts.

Conventional Backup Methods

You can also use conventional backup methods. For more information, see [Recovering and restoring a system](#) in *Red Hat Enterprise Linux 8 Configuring basic system settings* .



NOTE

If you plan to use the **satellite-maintain backup** command to create a backup, do not stop Satellite services.

- When creating a snapshot or conventional backup, you must stop all services as follows:

```
# satellite-maintain service stop
```

- Start the services after creating a snapshot or conventional backup:

```
# satellite-maintain service start
```

11.1. ESTIMATING THE SIZE OF A BACKUP

The full backup creates uncompressed archives of PostgreSQL and Pulp database files, and Satellite configuration files. Compression occurs after the archives are created to decrease the time when Satellite services are unavailable.

A full backup requires space to store the following data:

- Uncompressed Satellite database and configuration files
- Compressed Satellite database and configuration files

- An extra 20% of the total estimated space to ensure a reliable backup

Procedure

1. Enter the **du** command to estimate the size of uncompressed directories containing Satellite database and configuration files:

```
# du -sh /var/lib/pgsql/data /var/lib/pulp
100G /var/lib/pgsql/data
100G /var/lib/pulp

# du -csh /var/lib/qpidd /var/lib/tftpboot /etc /root/ssl-build \
/var/www/html/pub /opt/puppetlabs
886M /var/lib/qpidd
16M /var/lib/tftpboot
37M /etc
900K /root/ssl-build
100K /var/www/html/pub
2M /opt/puppetlabs
942M total
```

2. Calculate how much space is required to store the compressed data. The following table describes the compression ratio of all data items included in the backup:

Table 11.1. Backup Data Compression Ratio

Data type	Directory	Ratio	Example results
PostgreSQL database files	/var/lib/pgsql/data	80 – 85%	100 GB → 20 GB
Pulp RPM files	/var/lib/pulp	(not compressed)	100 GB
Configuration files	/var/lib/qpidd /var/lib/tftpboot /etc /root/ssl-build /var/www/html/pub /opt/puppetlabs	85%	942 MB → 141 MB

In this example, the compressed backup data occupies 120 GB in total.

3. To calculate the amount of available space you require to store a backup, calculate the sum of the estimated values of compressed and uncompressed backup data, and add an extra 20% to ensure a reliable backup. This example requires 201 GB plus 120 GB for the uncompressed and compressed backup data, 321 GB in total. With 64 GB of extra space, 385 GB must be allocated for the backup location.

11.2. PERFORMING A FULL BACKUP OF SATELLITE SERVER OR CAPSULE SERVER

Red Hat Satellite uses the **satellite-maintain backup** command to make backups.

There are three main methods of backing up Satellite Server:

- Offline backup
 - Online backup
 - Snapshot backups
- For more information about each of these methods, you can view the usage statements for each backup method.

Offline backups

```
# satellite-maintain backup offline --help
```

Online backups

```
# satellite-maintain backup online --help
```

Snapshots backups

```
# satellite-maintain backup snapshot --help
```

Directory creation

The **satellite-maintain backup** command creates a time-stamped subdirectory in the backup directory that you specify. The **satellite-maintain backup** command does not overwrite backups, therefore you must select the correct directory or subdirectory when restoring from a backup or an incremental backup. The **satellite-maintain backup** command stops and restarts services as required.

When you run the **satellite-maintain backup offline** command, the following default backup directories are created:

- **satellite-backup** on Satellite
- **foreman-proxy-backup** on Capsule

If you want to set a custom directory name, add the **--preserve-directory** option and add a directory name. The backup is then stored in the directory you provide in the command line. If you use the **--preserve-directory** option, no data is removed if the backup fails.

Note that if you use a local PostgreSQL database, the **postgres** user requires write access to the backup directory.

Remote databases

You can use the **satellite-maintain backup** command to back up remote databases.

You can use both online and offline methods to back up remote databases, but if you use offline methods, such as snapshot, the **satellite-maintain backup** command performs a database dump.

Prerequisites

- Ensure that your backup location has sufficient available disk space to store the backup. For more information, see [Section 11.1, “Estimating the Size of a Backup”](#).



WARNING

Request other users of Satellite Server or Capsule Server to save any changes and warn them that Satellite services are unavailable for the duration of the backup. Ensure no other tasks are scheduled for the same time as the backup.

Procedure

- On Satellite Server, enter the following command:

```
# satellite-maintain backup offline /var/satellite-backup
```

- On Capsule Server, enter the following command:

```
# satellite-maintain backup offline /var/foreman-proxy-backup
```

11.3. PERFORMING A BACKUP WITHOUT PULP CONTENT

You can perform an offline backup that excludes the contents of the Pulp directory. The backup without Pulp content is useful for debugging purposes and is only intended to provide access to configuration files without backing up the Pulp database. You cannot restore from a directory that does not contain Pulp content.



WARNING

Request other users of Satellite Server or Capsule Server to save any changes and warn them that Satellite services are unavailable for the duration of the backup. Ensure no other tasks are scheduled for the same time as the backup.

Prerequisites

- Ensure that your backup location has sufficient available disk space to store the backup. For more information, see [Section 11.1, “Estimating the Size of a Backup”](#).

Procedure

- To perform an offline backup without Pulp content, enter the following command:

```
# satellite-maintain backup offline --skip-pulp-content /var/backup_directory
```

11.4. PERFORMING AN INCREMENTAL BACKUP

Use this procedure to perform an offline backup of any changes since a previous backup.

To perform incremental backups, you must perform a full backup as a reference to create the first incremental backup of a sequence. Keep the most recent full backup and a complete sequence of incremental backups to restore from.



WARNING

Request other users of Satellite Server or Capsule Server to save any changes and warn them that Satellite services are unavailable for the duration of the backup. Ensure no other tasks are scheduled for the same time as the backup.

Prerequisites

- Ensure that your backup location has sufficient available disk space to store the backup. For more information, see [Section 11.1, “Estimating the Size of a Backup”](#).

Procedure

1. To perform a full offline backup, enter the following command:

```
# satellite-maintain backup offline /var/backup_directory
```

2. To create a directory within your backup directory to store the first incremental backup, enter the **satellite-maintain backup** command with the **--incremental** option:

```
# satellite-maintain backup offline --incremental /var/backup_directory/full_backup
/var/backup_directory
```

3. To create the second incremental backup, enter the **satellite-maintain backup** command with the **--incremental** option and include the path to the first incremental backup to indicate the starting point for the next increment. This creates a directory for the second incremental backup in your backup directory:

```
# satellite-maintain backup offline --incremental
/var/backup_directory/first_incremental_backup /var/backup_directory
```

4. Optional: If you want to point to a different version of the backup, and make a series of increments with that version of the backup as the starting point, you can do this at any time. For example, if you want to make a new incremental backup from the full backup rather than the first or second incremental backup, point to the full backup directory:

```
# satellite-maintain backup offline --incremental /var/backup_directory/full_backup
/var/backup_directory
```

11.5. EXAMPLE OF A WEEKLY FULL BACKUP FOLLOWED BY DAILY INCREMENTAL BACKUPS

The following script performs a full backup on a Sunday followed by incremental backups for each of the following days. A new subdirectory is created for each day that an incremental backup is performed. The script requires a daily cron job.

```
#!/bin/bash -e
PATH=/sbin:/bin:/usr/sbin:/usr/bin
DESTINATION=/var/backup_directory
if [[ $(date +%w) == 0 ]]; then
    satellite-maintain backup offline --assumeeyes $DESTINATION
else
    LAST=$(ls -td -- $DESTINATION/* | head -n 1)
    satellite-maintain backup offline --assumeeyes --incremental "$LAST" $DESTINATION
fi
exit 0
```

Note that the **satellite-maintain backup** command requires **/sbin** and **/usr/sbin** directories to be in **PATH** and the **--assumeeyes** option is used to skip the confirmation prompt.

11.6. PERFORMING AN ONLINE BACKUP

Perform an online backup only for debugging purposes.

Risks Associated with Online Backups

When performing an online backup, if there are procedures affecting the Pulp database, the Pulp part of the backup procedure repeats until it is no longer being altered. Because the backup of the Pulp database is the most time consuming part of backing up Satellite, if you make a change that alters the Pulp database during this time, the backup procedure keeps restarting.

For production environments, use the snapshot method. For more information, see [Section 11.7, "Performing a Snapshot Backup"](#). If you want to use the online backup method in production, proceed with caution and ensure that no modifications occur during the backup.



WARNING

Request other users of Satellite Server or Capsule Server to save any changes and warn them that Satellite services are unavailable for the duration of the backup. Ensure no other tasks are scheduled for the same time as the backup.

Prerequisites

- Ensure that your backup location has sufficient available disk space to store the backup. For more information, see [Section 11.1, "Estimating the Size of a Backup"](#).

Procedure

- To perform an online backup, enter the following command:


```
# satellite-maintain backup online /var/backup_directory
```

11.7. PERFORMING A SNAPSHOT BACKUP

You can perform a snapshot backup that uses Logical Volume Manager (LVM) snapshots of the Pulp, and PostgreSQL directories. Creating a backup from LVM snapshots mitigates the risk of an inconsistent backup.

The snapshot backup method is faster than a full offline backup and therefore reduces Satellite downtime.

To view the usage statement, enter the following command:

```
satellite-maintain backup snapshot -h
```



WARNING

Request other Satellite Server or Capsule Server users to save any changes and warn them that Satellite services are unavailable for the duration of the backup. Ensure no other tasks are scheduled for the same time as the backup.

Prerequisites

- The system uses LVM for the directories that you snapshot: **/var/lib/pulp/**, and **/var/lib/pgsql**.
- The free disk space in the relevant volume group (VG) is three times the size of the snapshot. More precisely, the VG must have enough space unreserved by the member logical volumes (LVs) to accommodate new snapshots. In addition, one of the LVs must have enough free space for the backup directory.
- The target backup directory is on a different LV than the directories that you snapshot.

Procedure

- To perform a snapshot backup, enter the **satellite-maintain backup snapshot** command:

```
# satellite-maintain backup snapshot /var/backup_directory
```

The **satellite-maintain backup snapshot** command creates snapshots when the services are active, and stops all services which can impact the backup. This makes the maintenance window shorter. After the successful snapshot, all services are restarted and LVM snapshots are removed.

11.8. WHITE-LISTING AND SKIPPING STEPS WHEN PERFORMING BACKUPS

A backup using the **satellite-maintain backup** command proceeds in a sequence of steps. To skip part of the backup add the **--whitelist** option to the command and add the step label that you want to omit.

Procedure

- To display a list of available step labels, enter the following command:

```
# satellite-maintain advanced procedure run -h
```

- To skip a step of the backup, enter the **satellite-maintain backup** command with the **--whitelist** option. For example:

```
# satellite-maintain backup online --whitelist backup-metadata -y /var/backup_directory
```

CHAPTER 12. RESTORING SATELLITE SERVER OR CAPSULE SERVER FROM A BACKUP

You can restore Satellite Server or Capsule Server from the backup data that you create as part of [Chapter 11, Backing Up Satellite Server and Capsule Server](#). This process outlines how to restore the backup on the same server that generated the backup, and all data covered by the backup is deleted on the target system. If the original system is unavailable, provision a system with the same configuration settings and host name.

12.1. RESTORING FROM A FULL BACKUP

Use this procedure to restore Red Hat Satellite or Capsule Server from a full backup. When the restore process completes, all processes are online, and all databases and system configuration revert to the state at the time of the backup.

Prerequisites

- Ensure that you are restoring to the correct instance. The Red Hat Satellite instance must have the same host name, configuration, and be the same minor version (X.Y) as the original system.
- Ensure that you have an existing target directory. The target directory is read from the configuration files contained within the archive.
- Ensure that you have enough space to store this data on the base system of Satellite Server or Capsule Server as well as enough space after the restoration to contain all the data in the **/etc/** and **/var/** directories contained within the backup.

To check the space used by a directory, enter the following command:

```
# du -sh /var/backup_directory
```

To check for free space, enter the following command:

```
# df -h /var/backup_directory
```

Add the **--total** option to get a total of the results from more than one directory.

- Ensure that all SELinux contexts are correct. Enter the following command to restore the correct SELinux contexts:

```
# restorecon -Rv /
```

Procedure

1. Choose the appropriate method to install Satellite or Capsule:
 - To install Satellite Server from a connected network, follow the procedures in [Installing Satellite Server in a Connected Network Environment](#).
 - To install Satellite Server from a disconnected network, follow the procedures in [Installing Satellite Server in a Disconnected Network Environment](#).
 - To install a Capsule Server, follow the procedures in [Installing Capsule Server](#).

2. Copy the backup data to Satellite Server's local file system. Use `/var/` or `/var/tmp/`.
3. Run the restoration script.

```
# satellite-maintain restore /var/backup_directory
```

Where `backup_directory` is the time-stamped directory or subdirectory containing the backed-up data.

The restore process can take a long time to complete, because of the amount of data to copy.

Additional Resources

- For troubleshooting, you can check `/var/log/foreman/production.log` and `/var/log/messages`.

12.2. RESTORING FROM INCREMENTAL BACKUPS

Use this procedure to restore Satellite or Capsule Server from incremental backups. If you have multiple branches of incremental backups, select your full backup and each incremental backup for the *branch* you want to restore, in chronological order.

When the restore process completes, all processes are online, and all databases and system configuration revert to the state at the time of the backup.

Procedure

1. Restore the last full backup using the instructions in [Section 12.1, "Restoring from a Full Backup"](#).
2. Remove the full backup data from Satellite Server's local file system, for example, `/var/` or `/var/tmp/`.
3. Copy the incremental backup data to Satellite Server's local file system, for example, `/var/` or `/var/tmp/`.
4. Restore the incremental backups in the same sequence that they are made:

```
# satellite-maintain restore -i /var/backup_directory/FIRST_INCREMENTAL
# satellite-maintain restore -i /var/backup_directory/SECOND_INCREMENTAL
```

If you created the backup using the `satellite-maintain backup` command, you do not need to use `-i` option in the command.

Additional Resources

- For troubleshooting, you can check `/var/log/foreman/production.log` and `/var/log/messages`.

12.3. BACKUP AND RESTORE CAPSULE SERVER USING A VIRTUAL MACHINE SNAPSHOT

If your Capsule Server is a virtual machine, you can restore it from a snapshot. Creating weekly snapshots to restore from is recommended. In the event of failure, you can install, or configure a new Capsule Server, and then synchronize the database content from Satellite Server.

If required, deploy a new Capsule Server, ensuring the host name is the same as before, and then install

the Capsule certificates. You may still have them on Satellite Server, the package name ends in -certs.tar, alternately create new ones. Follow the procedures in [Installing Capsule Server](#) until you can confirm, in the Satellite web UI, that Capsule Server is connected to Satellite Server. Then use the procedure [Section 12.3.1, "Synchronizing an External Capsule"](#) to synchronize from Satellite.

12.3.1. Synchronizing an External Capsule

Synchronize an external Capsule with Satellite.

Procedure

1. To synchronize an external Capsule, select the relevant organization and location in the Satellite web UI, or choose **Any Organization** and **Any Location**.
2. In the Satellite web UI, navigate to **Infrastructure > Capsules** and click the name of the Capsule to synchronize.
3. On the **Overview** tab, select **Synchronize**.

CHAPTER 13. RENAMING SATELLITE SERVER OR CAPSULE SERVER

To rename Satellite Server or Capsule Server, you must use the **satellite-change-hostname** script.

If you rename Satellite Server, you must reregister all Satellite clients and configure each Capsule Server to point them to the new Satellite host name. If you use custom SSL certificates, you must regenerate them with the new host name. If you use virt-who, you must update the virt-who configuration files with the new host name.

If you rename Capsule Server, you must reregister all Capsule clients and update the Capsule host name in the Satellite web UI. If you use custom SSL certificates, you must regenerate them with the new host name.



WARNING

The renaming process shuts down all Satellite Server services on the host being renamed. When the renaming is complete, all services are restarted.

13.1. RENAMING SATELLITE SERVER

The host name of Satellite Server is used by Satellite Server components, all Capsule Servers, and hosts registered to it for communication. This procedure ensures that you update all references to the new host name.

If you use external authentication, you must reconfigure Satellite Server for external authentication after you run the **satellite-change-hostname** script. The **satellite-change-hostname** script breaks external authentication for Satellite Server. For more information about configuring external authentication, see [Configuring External Authentication](#) in *Installing Satellite Server in a Connected Network Environment*.

If you use virt-who, you must update the virt-who configuration files with the new host name after you run the **satellite-change-hostname** script. For more information, see [Modifying a virt-who Configuration](#) in *Configuring Virtual Machine Subscriptions in Red Hat Satellite*.

Prerequisites

- Both the **hostname** and **hostname -f** commands must return the FQDN of Satellite Server or the **satellite-change-hostname** script will fail to complete. If the **hostname** command returns the shortname of Satellite Server instead of the FQDN, use **hostnamectl set-hostname *old_fqdn*** to set the old FQDN correctly before attempting to use the **satellite-change-hostname** script.
- Perform a backup of Satellite Server before changing a host name. If the renaming process is not successful, you must restore it from a backup. For more information, see [Chapter 11, Backing Up Satellite Server and Capsule Server](#).
- Optional: If Satellite Server has a custom SSL certificate installed, a new certificate must be obtained for the host's new name. For more information, see [Configuring Satellite Server with a Custom SSL Certificate](#) in *Installing Satellite Server in a Connected Network Environment*.

Procedure

1. On Satellite Server, choose the appropriate method to run the **satellite-change-hostname** script, providing the new host name and Satellite credentials:

- If your Satellite Server is installed with default self-signed SSL certificates, enter the following command:

```
# satellite-change-hostname new-satellite \
--username admin \
--password password
```

- If your Satellite Server is installed with custom SSL certificates:

```
# satellite-change-hostname new-satellite \
--username admin \
--password password \
--custom-cert "/root/ownca/test.com/test.com.crt" \
--custom-key "/root/ownca/test.com/test.com.key"
```

2. Optional: If you have created a custom SSL certificate for the new Satellite Server host name, run the Satellite installation script to install the certificate. For more information about installing a custom SSL certificate, see [Deploying a Custom SSL Certificate to Satellite Server](#) in *Installing Satellite Server in a Connected Network Environment* .
3. Reregister all Satellite hosts. For more information, see [Registering Hosts](#) in *Managing Hosts*.
4. On all Capsule Servers, run the Satellite installation script to update references to the new host name:

```
# satellite-installer \
--foreman-proxy-foreman-base-url https://new-satellite.example.com \
--foreman-proxy-trusted-hosts new-satellite.example.com \
--puppet-server-foreman-url https://new-satellite.example.com
```

5. On Satellite Server, list all Capsule Servers:

```
# hammer capsule list
```

6. On Satellite Server, synchronize content for each Capsule Server:

```
# hammer capsule content synchronize \
--id capsule_id_number
```

13.2. RENAMING CAPSULE SERVER

The host name of Capsule Server is referenced by Satellite Server components, and all hosts registered to it. This procedure ensures that you update all references to the new host name.



NOTE

- Both the **hostname** and **hostname -f** commands must return the FQDN of Capsule Server or the **satellite-change-hostname** script will fail to complete.
- If the **hostname** command returns the shortname of Capsule Server instead of the FQDN, use **hostnamectl set-hostname *old_fqdn*** to set the old FQDN correctly before attempting to use the **satellite-change-hostname** script.

Prerequisites

- Backup Capsule Server. The **satellite-change-hostname** script makes irreversible changes to Capsule Server. If the renaming process is not successful, you must restore it from a backup. Perform a backup before changing a host name. For more information, see [Chapter 11, Backing Up Satellite Server and Capsule Server](#).

Procedure

1. On Satellite Server, generate a new certificates archive file for Capsule Server.

- If you are using the default SSL certificate, enter the following command:

```
# capsule-certs-generate \  
--foreman-proxy-fqdn new-capsule.example.com \  
--certs-tar /root/new-capsule.example.com-certs.tar
```

Ensure that you enter the full path to the **.tar** file.

- If you are using a custom SSL certificate, create a new SSL certificate for Capsule Server. For more information, see [Configuring Capsule Server with a Custom SSL Certificate](#) in *Installing Capsule Server*.
2. On Satellite Server, copy the certificates archive file to Capsule Server, providing the **root** user's password when prompted. In this example the archive file is copied to the **root** user's home directory, but you may prefer to copy it elsewhere.

```
# scp /root/new-capsule.example.com-certs.tar root@capsule.example.com:
```

3. On Capsule Server, run the **satellite-change-hostname** script and provide the host's new name, Satellite credentials, and certificates archive filename.

```
# satellite-change-hostname new-capsule --username admin \  
--password password \  
--certs-tar /root/new-capsule.example.com-certs.tar
```

Ensure that you enter the full path to the **.tar** file.

4. Optional: If you have created a custom certificate for Capsule Server, on Capsule Server, to deploy the certificate, enter the **satellite-installer** command that the **capsule-certs-generate** command returns. For more information, see [Deploying a Custom SSL Certificate to Capsule Server](#) in *Installing Capsule Server*.
5. On all Capsule clients, enter the following commands to reinstall the bootstrap RPM, reregister clients, and refresh their subscriptions.

You can use remote execution feature to perform this step. For more information, see [Configuring and Setting up Remote Jobs](#) in *Managing Hosts*.

```
# dnf remove katello-ca-consumer*

# dnf install http://new-capsule.example.com/pub/katello-ca-consumer-latest.noarch.rpm

# subscription-manager register --org="My_Organization" \
--environment="Library" \
--force

# subscription-manager refresh
```

6. In the Satellite web UI, navigate to **Infrastructure** > **Capsules**.
7. Locate Capsule Server in the list, and click **Edit** to the right of it.
8. Edit the **Name** and **URL** fields to match Capsule Server's new host name, then click **Submit**.
9. On your DNS server, add a record for Capsule Server's new host name, and delete the record for the previous host name.

CHAPTER 14. MAINTAINING SATELLITE SERVER

This chapter provides information on how to maintain a Satellite Server, including information on how to work with audit records, how to clean unused tasks, and how to recover Pulp from a full disk.

14.1. DELETING AUDIT RECORDS

Audit records are created automatically in Satellite. You can use the **foreman-rake audits:expire** command to remove audits at any time. You can also use a cron job to schedule audit record deletions at the set interval that you want.

By default, using the **foreman-rake audits:expire** command removes audit records that are older than 90 days. You can specify the number of days to keep the audit records by adding the **days** option and add the number of days.

For example, if you want to delete audit records that are older than seven days, enter the following command:

```
# foreman-rake audits:expire days=7
```

14.2. ANONYMIZING AUDIT RECORDS

You can use the **foreman-rake audits:anonymize** command to remove any user account or IP information while maintaining the audit records in the database. You can also use a cron job to schedule anonymizing the audit records at the set interval that you want.

By default, using the **foreman-rake audits:anonymize** command anonymizes audit records that are older than 90 days. You can specify the number of days to keep the audit records by adding the **days** option and add the number of days.

For example, if you want to anonymize audit records that are older than seven days, enter the following command:

```
# foreman-rake audits:anonymize days=7
```

14.3. DELETING REPORT RECORDS

Report records are created automatically in Satellite. You can use the **foreman-rake reports:expire** command to remove reports at any time. You can also use a cron job to schedule report record deletions at the set interval that you want.

By default, using the **foreman-rake reports:expire** command removes report records that are older than 90 days. You can specify the number of days to keep the report records by adding the **days** option and add the number of days.

For example, if you want to delete report records that are older than seven days, enter the following command:

```
# foreman-rake reports:expire days=7
```

14.4. CONFIGURING THE CLEANING UNUSED TASKS FEATURE

Satellite performs regular cleaning to reduce disc space in the database and limit the rate of disk growth. As a result, Satellite backup completes faster and overall performance is higher.

By default, Satellite executes a cron job that cleans tasks every day at 19:45. Satellite removes the following tasks during the cleaning:

- Tasks that have run successfully and are older than thirty days
- All tasks that are older than a year

You can configure the cleaning unused tasks feature using these options:

- To configure the time at which Satellite runs the cron job, set the **--foreman-plugin-tasks-cron-line** parameter to the time you want in cron format. For example, to schedule the cron job to run every day at 15:00, enter the following command:

```
# satellite-installer --foreman-plugin-tasks-cron-line "00 15 * * *"
```

- To configure the period after which Satellite deletes the tasks, edit the **:rules:** section in the **/etc/foreman/plugins/foreman-tasks.yaml** file.
- To disable regular task cleanup on Satellite, enter the following command:

```
# satellite-installer --foreman-plugin-tasks-automatic-cleanup false
```

- To reenabale regular task cleanup on Satellite, enter the following command:

```
# satellite-installer --foreman-plugin-tasks-automatic-cleanup true
```

14.5. DELETING TASK RECORDS

Task records are created automatically in Satellite. You can use the **foreman-rake foreman_tasks:cleanup** command to remove tasks at any time. You can also use a cron job to schedule Task record deletions at the set interval that you want.

For example, if you want to delete task records from successful repository synchronizations, enter the following command:

```
# foreman-rake foreman_tasks:cleanup TASK_SEARCH='label = Actions::Katello::Repository::Sync' STATES='stopped'
```

14.6. DELETING A TASK BY ID

You can delete tasks by ID, for example if you have submitted confidential data by mistake.

Procedure

1. Connect to your Satellite Server using SSH:

```
# ssh root@satellite.example.com
```

2. Optional: View the task:

```
# hammer task info --id My_Task_ID
```

3. Delete the task:

```
# foreman-rake foreman_tasks:cleanup TASK_SEARCH="id=My_Task_ID"
```

4. Optional: Ensure the task has been removed from Satellite Server:

```
# hammer task info --id My_Task_ID
```

Note that because the task is deleted, this command returns a non-zero exit code.

14.7. RECOVERING FROM A FULL DISK

The following procedure describes how to resolve the situation when a logical volume (LV) with the Pulp database on it has no free space.

Procedure

1. Let running Pulp tasks finish but do not trigger any new ones as they can fail due to the full disk.
2. Ensure that the LV with the `/var/lib/pulp` directory on it has sufficient free space. Here are some ways to achieve that:

- a. Remove orphaned content:

```
# foreman-rake katello:delete_orphaned_content RAILS_ENV=production
```

This is run weekly so it will not free much space.

- b. Change the download policy from **Immediate** to **On Demand** for as many repositories as possible and remove already downloaded packages. See the Red Hat Knowledgebase solution [How to change syncing policy for Repositories on Satellite from "Immediate" to "On-Demand"](#) on the Red Hat Customer Portal for instructions.
- c. Grow the file system on the LV with the `/var/lib/pulp` directory on it. For more information, see [Growing a logical volume and file system](#) in *Red Hat Enterprise Linux 8 Configuring and managing logical volumes*.

**NOTE**

If you use an untypical file system (other than for example ext3, ext4, or xfs), you might need to unmount the file system so that it is not in use. In that case, complete the following steps:

1. Stop Satellite services:

```
# satellite-maintain service stop
```

2. Grow the file system on the LV.

3. Start Satellite services:

```
# satellite-maintain service start
```

3. If some Pulp tasks failed due to the full disk, run them again.

14.8. MANAGING PACKAGES ON THE BASE OPERATING SYSTEM OF SATELLITE SERVER OR CAPSULE SERVER

To install and update packages on the Satellite Server or Capsule Server base operating system, you must enter the **satellite-maintain packages** command. Satellite prevents users from installing and updating packages with **yum** because **yum** might also update the packages related to Satellite Server or Capsule Server and result in system inconsistency.

**IMPORTANT**

The **satellite-maintain packages** command restarts some services on the operating system where you run it because it runs the **satellite-installer** command after installing packages.

Procedure

- To install packages on Satellite Server or Capsule Server, enter the following command:

```
# satellite-maintain packages install package_1 package_2
```

- To update specific packages on Satellite Server or Capsule Server, enter the following command:

```
# satellite-maintain packages update package_1 package_2
```

- To update all packages on Satellite Server or Capsule Server, enter the following command:

```
# satellite-maintain packages update
```

Using yum to Check for Package Updates

If you want to check for updates using **yum**, enter the command to install and update packages manually and then you can use **yum** to check for updates:

```
# satellite-maintain packages unlock  
# yum check update  
# satellite-maintain packages lock
```

Updating packages individually can lead to package inconsistencies in Satellite Server or Capsule Server. For more information about updating packages in Satellite Server, see [Updating Satellite Server](#).

Enabling yum for Satellite Server or Capsule Server Package Management

If you want to install and update packages on your system using **yum** directly and control the stability of the system yourself, enter the following command:

```
# satellite-maintain packages unlock
```

Restoring Package Management to the Default Settings

If you want to restore the default settings and enable Satellite Server or Capsule Server to prevent users from installing and updating packages with **yum** and ensure the stability of the system, enter the following command:

```
# satellite-maintain packages lock
```

14.9. RECLAIMING POSTGRESQL SPACE

The PostgreSQL database can use a large amount of disk space especially in heavily loaded deployments. Use this procedure to reclaim some of this disk space on Satellite.

Procedure

1. Stop all services, except for the **postgresql** service:

```
# satellite-maintain service stop --exclude postgresql
```

2. Switch to the **postgres** user and reclaim space on the database:

```
# su - postgres -c 'vacuumdb --full --all'
```

3. Start the other services when the vacuum completes:

```
# satellite-maintain service start
```

CHAPTER 15. LOGGING AND REPORTING PROBLEMS

This chapter provides information on how to log and report problems in Satellite, including information on relevant log files, how to enable debug logging, how to open a support case and attach the relevant log tar files, and how to access support cases within the Satellite web UI.

You can use the log files and other information described in this chapter to do your own troubleshooting, or you can capture these and many more files, as well as diagnostic and configuration information, to send to Red Hat Support if you need further assistance.

For more information about Satellite logging settings, use **satellite-installer** with the **--full-help** option:

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

15.1. ENABLING DEBUG LOGGING

Debug logging provides the most detailed log information and can help with troubleshooting issues that can arise with Satellite and its components. In the Satellite CLI, enable debug logging to log detailed debugging information for Satellite.

Procedure

1. To enable debug logging, enter the following command:

```
# satellite-installer --foreman-logging-level debug
```

2. After you complete debugging, reset the logging level to the default value:

```
# satellite-installer --reset-foreman-logging-level
```

15.2. INCREASING THE LOGGING LEVELS TO HELP WITH DEBUGGING

By default, Satellite comes with **:INFO** level logging enabled. You can increase or decrease the log levels on your Satellite.

Enabling debug level logging on all components

```
# hammer admin logging --all --level-debug
# satellite-maintain service restart
```

Enabling debug level logging for a specific component

```
# hammer admin logging --components "Component" --level-debug
```

Reverting debug level logging to INFO

```
# hammer admin logging --all --level-production
# satellite-maintain service restart
```

Listing all components and changed configuration files

```
# hammer admin logging --list
-----|-----|-----
COMPONENT | AUTO-DETECTED BY EXISTENCE OF | DESTINATIONS
-----|-----|-----
dhcpd    | /etc/dhcp/dhcpd.conf           | syslog /var/log/dhcpd-debug.log
postgres| /var/lib/pgsql/data/postgresql.conf | syslog /var/lib/pgsql/data/pg_log/
proxy    | /etc/foreman-proxy/settings.yml | /var/log/foreman-proxy/proxy.log
qpidd    | /etc/qpidd/qpidd.conf         | syslog
rails    | /etc/foreman/settings.yml      | /var/log/foreman/production.log
tomcat   | /etc/candlepin/candlepin.conf  | /var/log/candlepin/ /var/log/tomcat/
virt-who | /etc/sysconfig/virt-who       | syslog
-----|-----|-----
```

15.2.1. Increasing the Logging Level For Hammer

You can find the log for Hammer in `~/.hammer/log/hammer.log`. Edit `/etc/hammer/cli_config.yml` and set the `:log_level:`:

```
:log_level: 'debug'
```

15.2.2. Increasing the Logging Level On Capsule

You can find the log for Capsule in `/var/log/foreman-proxy/proxy.log`. Uncomment the **DEBUG** line in `/etc/foreman-proxy/settings.yml`:

```
:log_level: DEBUG
```

Ensure to restart the **foreman-proxy** service afterwards:

```
# systemctl restart foreman-proxy
```

CAUTION

Running the installer will revert this change back.

15.2.3. Increasing the Logging Level For Candlepin

You can find the log for Candlepin in `/var/log/candlepin/candlepin.log`. Errors are also logged to a separate file for easier debugging `/var/log/candlepin/error.log`.

Extend `/etc/candlepin/candlepin.conf`:

```
log4j.logger.org.candlepin=DEBUG
```

Ensure to restart the **tomcat** service afterwards:

```
# systemctl restart tomcat
```

If the candlepin log files are too verbose, you can decrease the default debug level:


```
log4j.logger.org.candlepin.resource.ConsumerResource=WARN
log4j.logger.org.candlepin.resource.HypervisorResource=WARN
```

15.2.4. Increasing the Logging Level On Satellite

You can find the log for Satellite in `/var/log/foreman/production.log`.

Satellite stores logs for Apache in:

- `/var/log/httpd/foreman_error.log`
- `/var/log/httpd/foreman_access.log`
- `/var/log/httpd/foreman_ssl_error.log`
- `/var/log/httpd/foreman_ssl_access.log`

Procedure

1. Set the logging level in `/etc/foreman/settings.yaml`:

```
:logging:
  :production:
    :type: file
    :layout: pattern
    :level: debug
```

2. Enable selected loggers in `/etc/foreman/settings.yaml`:

```
:loggers:
  :ldap:
    :enabled: true
  :permissions:
    :enabled: true
  :sql:
    :enabled: true
```

Note that to see logging from some area, debug logging has to be set.

3. Restart Satellite services:

```
# satellite-maintain service restart
```

You can find the complete list of loggers with their default values in `/usr/share/foreman/config/application.rb` in the `Foreman::Logging.add_loggers` command.

15.2.5. Increasing the Logging Level For Qpid Dispatch Router

Qpid logs to syslog and can be viewed in `/var/log/messages` or with `journalctl`. Enable debug logging in `/etc/qpid-dispatch/qdrouterd.conf`:

```
enable: debug+
```

Ensure to restart the Qpid Dispatch Router afterwards:

```
# systemctl restart qdrouterd
```

CAUTION

Running the installer will revert this change back.

15.2.6. Increasing the Logging Level For Qpid Broker

Qpid logs to syslog and can be viewed in `/var/log/messages` or with `journalctl`. Set the log level in `/etc/qpid/qpidd.conf`:

```
log-enable=debug+
```

Ensure to restart the Qpid Broker afterwards:

```
# systemctl restart qpidd
```

CAUTION

Running the installer will revert this change.

15.2.7. Increasing the Logging Level For Redis

You can find the log for Redis in `/var/log/redis/redis.log`. Set the log level in `/etc/opt/rh/rh-redis5/redis.conf`:

```
loglevel debug
```

Ensure to restart the Redis service afterwards:

```
# systemctl restart rh-redis5-redis
```

15.2.8. Increasing the Logging Level For Postgres

You can find the log for Postgres in `/var/lib/pgsql/data/log`. Uncomment the `log_statement` in `/var/lib/pgsql/data/postgresql.conf`:

```
log_statement = 'all'
```

Ensure to restart Satellite services afterwards:

```
# satellite-maintain service restart
```

CAUTION

Based on the size of your Satellite installation, this can cause disk space to fill up very quickly. Only turn this on if absolutely needed.

For more debug log settings, refer to the [Postgresql documentation](#).

15.2.9. Increasing the Logging Level For Satellite Installer

You can find the log files in `/var/log/foreman-installer/`. To increase the log level of the Satellite Installer during an install:

```
# satellite-installer --verbose-log-level debug
```

15.2.10. Increasing the Logging Level For Pulp

By default, Pulp logs to syslog and can be viewed in `/var/log/messages` or with `journalctl`. Add the following config to the `/etc/pulp/settings.py` file:

```
LOGGING = {"dynaconf_merge": True, "loggers": {"": {'handlers': ['console'], 'level': 'DEBUG'}}
```

Ensure to restart the Pulp services afterwards:

```
# systemctl restart \
pulpcore-api \
pulpcore-content \
pulpcore-resource-manager \
pulpcore-worker@1 \
pulpcore-worker@2 \
rh-redis5-redis
```

15.2.11. Increasing the Logging Level For Puppet Agent

You can increase the logging level for Puppet agent on your Satellite Server.

Procedure

1. Add the following line to the `[agent]` block in the `/etc/puppetlabs/puppet/puppet.conf` file:

```
[agent]
log_level = debug
```

You can find the logs in `/var/log/puppetlabs/puppet/`

15.2.12. Increasing the Logging Level For Puppet Server

You can increase the logging level for Puppet server on your Satellite Server.

Procedure

1. Add the following line to the `[master]` block in `/etc/puppetlabs/puppet/puppet.conf` file:

```
[master]
log_level = debug
```

2. Restart the Puppet server:

```
# satellite-maintain service restart --only puppetserver
```

You can find the logs in `/var/log/puppetlabs/puppetserver/`.

15.3. RETRIEVING THE STATUS OF SERVICES

Procedure

1. In the Satellite web UI, navigate to **Administer > About**
2. On the **Smart Proxies** tab, you can view the status of all Capsules.
3. On the **Compute Resources** tab, you can view the status of attached compute resource providers.
4. In the **Backend System Status** table, you can view the status of all back-end services.

CLI procedure

- Run **hammer ping** to get information from the database and Satellite services:

```
# hammer ping
```

- Use **satellite-maintain** to check the status of the services running in systemd:

```
# satellite-maintain service status
```

- Use **satellite-maintain** to perform a health check:

```
$ satellite-maintain health check
```

15.4. RESTARTING SERVICES

Satellite uses a set of back-end services to perform tasks. If you experience an issue with your Satellite, check the status of Satellite services.

Procedure

- Use **satellite-maintain** to restart Satellite services:

```
# satellite-maintain service restart
```

TIP

Run **foreman-maintain --help** for more information.

15.5. ENABLING INDIVIDUAL LOGGERS

You can enable individual loggers for selective logging. Satellite uses the following loggers:

app

Logs web requests and all general application messages. Default value: true.

audit

Logs additional fact statistics, numbers of added, updated, and removed facts. Default value: true.

ldap

Logs high level LDAP queries and LDAP operations. Default value: false.

permissions

Logs queries to user roles, filters, and permissions when loading pages. Default value: false.

sql

Logs SQL queries made through Rails ActiveRecord. Default value: false.

Procedure

1. Enable the individual loggers that you want. For example, to enable **sql** and **ldap** loggers, enter the following command:

```
# satellite-installer \
--foreman-loggers ldap:true \
--foreman-loggers sql:true
```

2. Optional: To reset loggers to their default values, enter the following command:

```
# satellite-installer --reset-foreman-loggers
```

15.6. CONFIGURING LOGGING TO JOURNAL

You can configure Satellite to manage logging with Journal. Journal then forwards log messages to **rsyslog** and **rsyslog** writes the log messages to `/var/log/messages`. Note that after this change the log messages do not appear in `/var/log/foreman/production.log` or `/var/log/foreman-proxy.log` any more.

For more information about Journal, see [Viewing logs using the command line](#) in the *Red Hat Enterprise Linux 8 Configuring Basic System Settings Guide*.

Procedure

1. Enter the following **satellite-installer** command to configure logging to **journald**:

```
# satellite-installer \
--foreman-logging-layout pattern \
--foreman-logging-level info \
--foreman-logging-type journald \
--foreman-proxy-log JOURNAL
```

15.7. LOG FILE DIRECTORIES PROVIDED BY SATELLITE

Red Hat Satellite provides system information in the form of notifications and log files.

Table 15.1. Log File Directories for Reporting and Troubleshooting

Log File Directories	Description of Log File Content
/var/log/candlepin	Subscription management
/var/log/foreman-installer	Installer
/var/log/foreman-maintain	Foreman maintain
/var/log/foreman-proxy	Foreman proxy
/var/log/foreman	Foreman
/var/log/httpd	Apache HTTP server
/var/log/messages	Various other log messages
/var/log/puppetlabs/puppet	Configuration management
/var/log/rhsm	Subscription management
/var/log/tomcat	Candlepin webservice logs

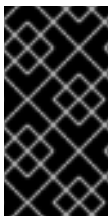
You can also use the **foreman-tail** command to follow many of the log files related to Satellite. You can run **foreman-tail -l** to list the processes and services that it follows.

15.8. UTILITIES FOR COLLECTING LOG INFORMATION

You can collect information from log files to troubleshoot Satellite.

sosreport

The **sosreport** command collects configuration and diagnostic information from a Linux system, such as the running Kernel version, loaded modules, running services, and system and service configuration files. This output is stored in a tar file located at **/var/tmp/sosreport-XXX-20171002230919.tar.xz**. For more information, run **sosreport --help** or see [What is a sosreport and how can I create one?](#)



IMPORTANT

The collection process removes security information such as passwords, tokens, and keys while collecting information. However, the tar files can still contain sensitive information about the Satellite Server. Red Hat recommends that you send this information directly to the intended recipient and not to a public target.

CHAPTER 16. MONITORING RESOURCES

The following chapter details how to configure monitoring and reporting for managed systems. This includes host configuration, Content Views, compliance, subscriptions, registered hosts, promotions, and synchronization.

16.1. USING THE RED HAT SATELLITE CONTENT DASHBOARD








The Red Hat Satellite content dashboard contains various widgets which provide an overview of the host configuration, Content Views, compliance reports, subscriptions and hosts currently registered, promotions and synchronization, and a list of the latest notifications.

In the Satellite web UI, navigate to **Monitor > Dashboard** to access the content dashboard. The dashboard can be rearranged by clicking on a widget and dragging it to a different position. The following widgets are available:

Host Configuration Status

An overview of the configuration states and the number of hosts associated with it during the last reporting interval. The following table shows the descriptions of the possible configuration states.

Table 16.1. Host Configuration States

Icon	State	Description
	Hosts that had performed modifications without error	Host that successfully performed modifications during the last reporting interval.
	Hosts in error state	Hosts on which an error was detected during the last reporting interval.
	Good host reports in the last 35 minutes	Hosts without error that did not perform any modifications in the last 35 minutes.
	Hosts that had pending changes	Hosts on which some resources would be applied but Puppet was configured to run in the noop mode.
	Out of sync hosts	Hosts that were not synchronized and the report was not received during the last reporting interval.
	Hosts with no reports	Hosts for which no reports were collected during the last reporting interval.
	Hosts with alerts disabled	Hosts which are not being monitored.

Click the particular configuration status to view hosts associated with it.

Host Configuration Chart

A pie chart shows the proportion of the configuration status and the percentage of all hosts associated with it.

Latest Events

A list of messages produced by hosts including administration information, product and subscription changes, and any errors.

Monitor this section for global notifications sent to all users and to detect any unusual activity or errors.

Run Distribution (last 30 minutes)

A graph shows the distribution of the running Puppet agents during the last puppet interval which is 30 minutes by default. In this case, each column represents a number of reports received from clients during 3 minutes.

New Hosts

A list of the recently created hosts. Click the host for more details.

Task Status

A summary of all current tasks, grouped by their state and result. Click the number to see the list of corresponding tasks.

Latest Warning/Error Tasks

A list of the latest tasks that have been stopped due to a warning or error. Click a task to see more details.

Discovered Hosts

A list of all bare-metal hosts detected on the provisioning network by the Discovery plug-in.

Latest Errata

A list of all errata available for hosts registered to Satellite.

Content Views

A list of all Content Views in Satellite and their publish status.




Sync Overview

An overview of all products or repositories enabled in Satellite and their synchronization status. All products that are in the queue for synchronization, are unsynchronized or have been previously synchronized are listed in this section.

Host Subscription Status

An overview of the subscriptions currently consumed by the hosts registered to Satellite. A subscription is a purchased certificate that unlocks access to software, upgrades, and security fixes for hosts. The following table shows the possible states of subscriptions.

Table 16.2. Host Subscription States

Icon	State	Description
	Invalid	Hosts that have products installed, but are not correctly subscribed. These hosts need attention immediately.
	Partial	Hosts that have a subscription and a valid entitlement, but are not using their full entitlements. These hosts should be monitored to ensure they are configured as expected.
	Valid	Hosts that have a valid entitlement and are using their full entitlements.

Click the subscription type to view hosts associated with subscriptions of the selected type.

Subscription Status

An overview of the current subscription totals that shows the number of active subscriptions, the number of subscriptions that expire in the next 120 days, and the number of subscriptions that have recently expired.

Host Collections

A list of all host collections in Satellite and their status, including the number of content hosts in each host collection.

Virt-who Configuration Status

An overview of the status of reports received from the **virt-who** daemon running on hosts in the environment. The following table shows the possible states.

Table 16.3. Virt-who Configuration States

State	Description
No Reports	No report has been received because either an error occurred during the virt-who configuration deployment, or the configuration has not been deployed yet, or virt-who cannot connect to Satellite during the scheduled interval.
No Change	No report has been received because hypervisor did not detect any changes on the virtual machines, or virt-who failed to upload the reports during the scheduled interval. If you added a virtual machine but the configuration is in the No Change state, check that virt-who is running.
OK	The report has been received without any errors during the scheduled interval.
Total Configurations	A total number of virt-who configurations.

Click the configuration status to see all configurations in this state.

The widget also lists the three latest configurations in the **No Change** state under **Latest Configurations Without Change**.

Latest Compliance Reports

A list of the latest compliance reports. Each compliance report shows a number of rules passed (P), failed (F), or othered (O). Click the host for the detailed compliance report. Click the policy for more details on that policy.

Compliance Reports Breakdown

A pie chart shows the distribution of compliance reports according to their status.

Red Hat Insights Actions

Red Hat Insights is a tool embedded in Satellite that checks the environment and suggests actions you can take. The actions are divided into 4 categories: Availability, Stability, Performance, and Security.

Red Hat Insights Risk Summary

A table shows the distribution of the actions according to the risk levels. Risk level represents how critical the action is and how likely it is to cause an actual issue. The possible risk levels are: Low, Medium, High, and Critical.

**NOTE**

It is not possible to change the date format displayed in the Satellite web UI.

16.1.1. Managing Tasks

Red Hat Satellite keeps a complete log of all planned or performed tasks, such as repositories synchronised, errata applied, and Content Views published. To review the log, navigate to **Monitor > Tasks**.

In the Task window, you can search for specific tasks, view their status, details, and elapsed time since they started. You can also cancel and resume one or more tasks.

The tasks are managed using the Dynflow engine. Remote tasks have a timeout which can be adjusted as needed.

To Adjust Timeout Settings:

1. In the Satellite web UI, navigate to **Administer > Settings**.
2. Enter `_%_timeout` in the search box and click **Search**. The search should return four settings, including a description.
3. In the **Value** column, click the icon next to a number to edit it.
4. Enter the desired value in seconds, and click **Save**.

**NOTE**

Adjusting the `_%_finish_timeout` values might help in case of low bandwidth. Adjusting the `_%_accept_timeout` values might help in case of high latency.

When a task is initialized, any back-end service that will be used in the task, such as Candlepin or Pulp, will be checked for correct functioning. If the check fails, you will receive an error similar to the following one:

There was an issue with the backend service candlepin: Connection refused – connect(2).

If the back-end service checking feature turns out to be causing any trouble, it can be disabled as follows.

To Disable Checking for Services:

1. In the Satellite web UI, navigate to **Administer > Settings**.
2. Enter `check_services_before_actions` in the search box and click **Search**.
3. In the **Value** column, click the icon to edit the value.
4. From the drop-down menu, select **false**.
5. Click **Save**.

16.2. CONFIGURING RSS NOTIFICATIONS

To view Satellite event notification alerts, click the **Notifications** icon in the upper right of the screen.

By default, the Notifications area displays RSS feed events published in the [Red Hat Satellite Blog](#).

The feed is refreshed every 12 hours and the Notifications area is updated whenever new events become available.

You can configure the RSS feed notifications by changing the URL feed. The supported feed format is RSS 2.0 and Atom. For an example of the RSS 2.0 feed structure, see the [Red Hat Satellite Blog feed](#). For an example of the Atom feed structure, see the [Foreman blog feed](#).

To Configure RSS Feed Notifications:

1. In the Satellite web UI, navigate to **Administer** > **Settings** and select the **Notifications** tab.
2. In the RSS URL row, click the edit icon in the **Value** column and type the required URL.
3. In the RSS enable row, click the edit icon in the **Value** column to enable or disable this feature.

16.3. MONITORING SATELLITE SERVER

Audit records list the changes made by all users on Satellite. This information can be used for maintenance and troubleshooting.

Procedure

1. In the Satellite web UI, navigate to **Monitor** > **Audits** to view the audit records.
2. To obtain a list of all the audit attributes, use the following command:

```
# foreman-rake audits:list_attributes
```

16.4. MONITORING CAPSULE SERVER

The following section shows how to use the Satellite web UI to find Capsule information valuable for maintenance and troubleshooting.

16.4.1. Viewing General Capsule Information

In the Satellite web UI, navigate to **Infrastructure** > **Capsules** to view a table of Capsule Servers registered to Satellite Server. The information contained in the table answers the following questions:

Is Capsule Server running?

This is indicated by a green icon in the **Status** column. A red icon indicates an inactive Capsule, use the **service foreman-proxy restart** command on Capsule Server to activate it.

What services are enabled on Capsule Server?

In the **Features** column you can verify if Capsule for example provides a DHCP service or acts as a Pulp mirror. Capsule features can be enabled during installation or configured in addition. For more information, see [Installing Capsule Server](#).

What organizations and locations is Capsule Server assigned to?

A Capsule Server can be assigned to multiple organizations and locations, but only Capsules belonging to the currently selected organization are displayed. To list all Capsules, select **Any Organization** from the context menu in the top left corner.

After changing the Capsule configuration, select **Refresh** from the drop-down menu in the **Actions** column to ensure the Capsule table is up to date.

Click the Capsule name to view further details. At the **Overview** tab, you can find the same information as in the Capsule table. In addition, you can answer to the following questions:

Which hosts are managed by Capsule Server?

The number of associated hosts is displayed next to the **Hosts managed** label. Click the number to view the details of associated hosts.

How much storage space is available on Capsule Server?

The amount of storage space occupied by the Pulp content in **/var/lib/pulp** is displayed. Also the remaining storage space available on the Capsule can be ascertained.

16.4.2. Monitoring Services

In the Satellite web UI, navigate to **Infrastructure > Capsules** and click the name of the selected Capsule. At the **Services** tab, you can find basic information on Capsule services, such as the list of DNS domains, or the number of Pulp workers. The appearance of the page depends on what services are enabled on Capsule Server. Services providing more detailed status information can have dedicated tabs at the Capsule page. For more information, see [Section 16.4.3, "Monitoring Puppet"](#).

16.4.3. Monitoring Puppet

In the Satellite web UI, navigate to **Infrastructure > Capsules** and click the name of the selected Capsule. At the **Puppet** tab you can find the following:

- A summary of Puppet events, an overview of latest Puppet runs, and the synchronization status of associated hosts at the **General** sub-tab.
- A list of Puppet environments at the **Environments** sub-tab.

At the **Puppet CA** tab you can find the following:

- A certificate status overview and the number of autosign entries at the **General** sub-tab.
- A table of CA certificates associated with the Capsule at the **Certificates** sub-tab. Here you can inspect the certificate expiry data, or cancel the certificate by clicking **Revoke**.
- A list of autosign entries at the **Autosign entries** sub-tab. Here you can create an entry by clicking **New** or delete one by clicking **Delete**.

CHAPTER 17. USING WEBHOOKS

A webhook is a way for a web page or web application to provide other applications with information in real time. Webhooks are only triggered after an event occurs. The request usually contains details of the event. An event triggers callbacks, such as sending an e-mail confirming a host has been provisioned. Webhooks enable you to define a call to an external API based on Satellite internal event using a fire-and-forget message exchange pattern. The application sending the request does not wait for the response, or ignores it.

Payload of a webhook is created from webhook templates. Webhook templates use the same ERB syntax as Provisioning templates. Available variables:

- **@event_name**: Name of an event.
- **@webhook_id**: Unique event ID.
- **@payload**: Payload data, different for each event type. To access individual fields, use **@payload[:key_name]** Ruby hash syntax.
- **@payload[:object]**: Database object for events triggered by database actions (create, update, delete). Not available for custom events.
- **@payload[:context]**: Additional information as hash like request and session UUID, remote IP address, user, organization and location.

Because webhooks use HTTP, no new infrastructure needs be added to existing web services.

The typical use case for webhooks in Satellite is making a call to a monitoring system when a host is created or deleted.

Webhooks are useful where the action you want to perform in the external system can be achieved through its API. Where it is necessary to run additional commands or edit files, the shellhooks plugin for Capsules is available. The shellhooks plugin enables you to define a shell script on the Capsule that can be executed through the API.

You can use webhooks successfully without installing the shellhooks plugin.

For a list of available events, see [Available webhook events](#).

17.1. MIGRATING TO WEBHOOKS

The legacy **foreman_hooks** plugin provided full access to model objects that the webhooks plugin does not intentionally provide.

The scope of what is available is limited by the safemode and all objects and macros are both subject to an API stability promise and are fully documented.

The number of events triggered by webhooks is substantially fewer than with **foreman_hooks**.

Webhooks are processed asynchronously so there is minimal risk of tampering with internals of the system. It is not possible to migrate from **foreman_hooks** without creating payloads for each individual webhook script. However, the webhooks plugin comes with several example payload templates. You can also use the example payloads with shellhooks to simplify migration.

Both script and payload templates must be customized to achieve similar results.

17.2. INSTALLING WEBHOOKS

Use the following procedure to install webhooks. After installing webhooks, you can configure Satellite Server to send webhook requests.

Procedure

- Install webhooks using the following command:

```
# satellite-installer --enable-foreman-plugin-webhooks
```

- Optionally, you can install the CLI plugin using the following command:

```
# satellite-installer --enable-foreman-cli-plugin-webhooks
```

17.3. CREATING A WEBHOOK TEMPLATE

Use the following procedure to create a webhook template in the Satellite web UI.

Procedure

1. In the Satellite web UI, navigate to **Administer > Webhooks Templates**.
2. Click **Clone an existing template** or **Create Template**.
3. Enter a name for the template.
4. Use the editor to make changes to the template payload.
A webhook HTTP payload must be created using Satellite template syntax. The webhook template can use a special variable called **@object** that can represent the main object of the event.

For more information, see [Template Writing Reference](#) in *Managing Hosts* and for available template macros and methods, visit [/templates_doc](#) on Satellite Server.

5. Optional: Enter the description and audit comment.
6. Assign organizations and locations.
7. Click **Submit**.

17.4. CREATING A WEBHOOK

You can customize events, payloads, HTTP authentication, content type, and headers through the Satellite web UI.

Use the following procedure to create a webhook in the Satellite web UI.

Procedure

1. In the Satellite web UI, click **Administer > Webhooks**.
2. Click **Create Webhook**.

3. Click **Subscribe to** to select an event.
4. Enter a name.
5. Enter a target URL. Webhooks make HTTP requests to pre-configured URLs. The target URL can be a dynamic URL. When using the shellhooks plugin, the URL should be in the form **https://capsule.example.com:9090/shellhook/my_script**.
6. Click **Template** to select a template.
7. Enter an HTTP method.
8. Check the **Enabled** flag if you want to create an active webhook.
9. Click the **Credentials** tab.
10. Optional: If HTTP authentication is required, enter the username and password.
11. Select **Verify SSL** if the server certificate should be verified against the system certificate store or Satellite CA.
12. Select **Proxy Authorization** when using shellhooks, otherwise clear this box.
13. On the **Additional** tab, enter the HTTP Content Type. For example, **application/json**, **application/xml** or **text/plain** on the payload you define. The application does not attempt to convert the content to match the specified content type.
14. Optional: Provide HTTP headers as JSON. ERB is also allowed.

When configuring webhooks with endpoints with non-standard HTTP or HTTPS ports, an SELinux port must be assigned, see [Configuring SELinux to Ensure Access to Satellite on Custom Ports](#) in *Installing Satellite Server in a Connected Network Environment*.

17.5. AVAILABLE WEBHOOK EVENTS

The following table contains a list of webhook events that are available from the Satellite web UI. **Action** events trigger webhooks only on **success**, so if an action fails, a webhook is not triggered.

For more information about payload, go to **Administer > About > Support > Templates DSL**. A list of available types is provided in the following table. Some events are marked as **custom**, in that case, the payload is an object object but a Ruby hash (key-value data structure) so syntax is different.

Event name	Description	Payload
Actions Katello Content View Promote Succeeded	A Content View was successfully promoted.	Actions::Katello::ContentView: :Promote
Actions Katello Content View Publish Succeeded	A repository was successfully synchronized.	Actions::Katello::ContentView: :Publish

Event name	Description	Payload
Actions Remote Execution Run Host Job Succeeded	A generic remote execution job succeeded for a host. This event is emitted for all Remote Execution jobs, when complete.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Katello Errata Install Succeeded	Install errata using the Katello interface.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Katello Group Install Succeeded	Install package group using the Katello interface.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Katello Package Install Succeeded	Install package using the Katello interface.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Katello Group Remove	Remove package group using the Katello interface.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Katello Package Remove Succeeded	Remove package using the Katello interface.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Katello Service Restart Succeeded	Restart Services using the Katello interface.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Katello Group Update Succeeded	Update package group using the Katello interface.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Katello Package Update Succeeded	Update package using the Katello interface.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Foreman OpenSCAP Run Scans Succeeded	Run OpenSCAP scan.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Ansible Run Host Succeeded	Runs an Ansible playbook containing all the roles defined for a host.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Ansible Run Capsule Upgrade Succeeded	Upgrade Capsules on given Capsule server hosts.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Ansible Configure Cloud Connector Succeeded	Configure Cloud Connector on given hosts.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Ansible Run Insights Plan Succeeded	Runs a given maintenance plan from Red Hat Access Insights given an ID.	Actions::RemoteExecution::RunHostJob

Event name	Description	Payload
Actions Remote Execution Run Host Job Ansible Run Playbook Succeeded	Run an Ansible playbook against given hosts.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Ansible Enable Web Console Succeeded	Run an Ansible playbook to enable the web console on given hosts.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Puppet Run Host Succeeded	Perform a single Puppet run.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Katello Module Stream Action Succeeded	Perform a module stream action using the Katello interface.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Leapp Pre-upgrade Succeeded	Upgradeability check for RHEL 7 host.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Leapp Remediation Plan Succeeded	Run Remediation plan with Leapp.	Actions::RemoteExecution::RunHostJob
Actions Remote Execution Run Host Job Leapp Upgrade Succeeded	Run Leapp upgrade job for RHEL 7 host.	Actions::RemoteExecution::RunHostJob
Build Entered	A host entered the build mode.	Custom event: @payload[:id] (host id), @payload[:hostname] (host name).
Build Exited	A host build mode was canceled, either it was successfully provisioned or the user canceled the build manually.	Custom event: @payload[:id] (host id), @payload[:hostname] (host name).
Content View Created/Updated/Destroyed	Common database operations on a Content View.	Katello::ContentView
Domain Created/Updated/Destroyed	Common database operations on a domain.	Domain
Host Created/Updated/Destroyed	Common database operations on a host.	Host
Hostgroup Created/Updated/Destroyed	Common database operations on a hostgroup.	Hostgroup

Event name	Description	Payload
Model Created/Updated/Destroyed	Common database operations on a model.	Model
Status Changed	Global host status of a host changed.	Custom event: @payload[:id] (host id), @payload[:hostname] , @payload[:global_status] (hash)
Subnet Created/Updated/Destroyed	Common database operations on a subnet.	Subnet
Template Render Performed	A report template was rendered.	Template
User Created/Updated/Destroyed	Common database operations on a user.	User

17.6. SHELLHOOKS

With webhooks, one Satellite event can only be mapped to one API call. For advanced integrations, where a single shell script can contain multiple commands, you can install a Capsule shellhooks plugin that exposes executables using a REST HTTP API.

A webhook can then be configured to reach out to a Capsule API to run a predefined shellhook, which, for example, can contain commands or edit files.

Scripts must be placed in **/var/lib/foreman-proxy/shellhooks** as executables with only alphanumeric characters and underscores in the name.

The HTTPS payload is passed using standard input, optional command line arguments can be provided using `X-Shellhook-Arg-1` to `N`.

The HTTP method must be POST. An example URL would be:
https://capsule.example.com:9090/shellhook/my_script.

You must enable **Proxy Authorization** for each webhook that is connected to a shellhook, to enable it to authorize a call.

Standard output and error are redirected to the Capsule log as messages with debug or warning levels respectively.

There is no return value from shellhook HTTPS calls.

17.7. INSTALLING THE SHELLHOOKS PLUGIN

Optionally, you can install and enable the shellhooks plugin on each Capsule used for shellhooks, using the following command:

```
# satellite-installer --enable-foreman-proxy-plugin-shellhooks
```

17.8. USING SHELLHOOK ARGUMENTS

Procedure

To pass arguments into a shellhook script, create the following HTTP headers:

```
{  
  "X-Shellhook-Arg-1": "<%= @object.content_view_version_id %>",  
  "X-Shellhook-Arg-2": "<%= @object.content_view_name %>"  
}
```

Ensure the content renders to a valid JSON. Also, only pass safe fields like database ID, name, or labels which do not include new lines or quote characters.

CHAPTER 18. SEARCHING AND BOOKMARKING

Satellite features powerful search functionality on most pages of the Satellite web UI. It enables you to search all kinds of resources that Satellite manages. Searches accept both free text and syntax-based queries, which can be built using extensive input prediction. Search queries can be saved as bookmarks for future reuse.

18.1. BUILDING SEARCH QUERIES

As you start typing a search query, a list of valid options to complete the current part of the query appears. You can either select an option from the list and keep building the query using the prediction, or continue typing. To learn how free text is interpreted by the search engine, see [Section 18.2, “Using Free Text Search”](#).

18.1.1. Query Syntax

parameter operator value

Available fields, resources to search, and the way the query is interpreted all depend on context, that is, the page where you perform the search. For example, the field "hostgroup" on the Hosts page is equivalent to the field "name" on the Host Groups page. The field type also determines available operators and accepted values.

For a list of all operators, see [Operators](#). For descriptions of value formats, see [Values](#).

18.1.2. Query Operators

All operators that can be used between *parameter* and *value* are listed in the following table. Other symbols and special characters that might appear in a prediction-built query, such as colons, do not have special meaning and are treated as free text.

Table 18.1. Comparison Operators Accepted by Search

Operator	Short Name	Description	Example
=	EQUALS	Accepts numerical, temporal, or text values. For text, exact case sensitive matches are returned.	hostgroup = RHEL7
!=	NOT EQUALS		
~	LIKE	Accepts text or temporal values. Returns case insensitive matches. Accepts the following wildcards: _ for a single character, % or * for any number of characters including zero. If no wildcard is specified, the string is treated as if surrounded by wildcards: %rhel7%	hostgroup ~ rhel%
!~	NOT LIKE		

Operator	Short Name	Description	Example
>	GREATER THAN	Accepts numerical or temporal values. For temporal values, the operator > is interpreted as "later than", and < as "earlier than". Both operators can be combined with EQUALS: >= <=	registered_at > 10-January-2017 The search will return hosts that have been registered after the given date, that is, between 10th January 2017 and now.
<	LESS THAN		registered_at <= Yesterday The search will return hosts that have been registered yesterday or earlier.
^	IN	Compares an expression against a list of values, as in SQL. Returns matches that contain or not contain the values, respectively.	release_version !^ 7
!^	NOT IN		
HAS or set?		Returns values that are present or not present, respectively.	has hostgroup or set? hostgroup On the Puppet Classes page, the search will return classes that are assigned to at least one host group.
NOT HAS or null?			not has hostgroup or null? hostgroup On the Dashboard with an overview of hosts, the search will return all hosts that have no assigned host group.

Simple queries that follow the described syntax can be combined into more complex ones using logical operators AND, OR, and NOT. Alternative notations of the operators are also accepted:

Table 18.2. Logical Operators Accepted by Search

Operator	Alternative Notations			Example
and	&	&&	<whitespace>	class = motd AND environment ~ production
or				errata_status = errata_needed errata_status = security_needed
not	-	!		hostgroup ~ rhel7 not status.failed

18.1.3. Query Values

Text Values

Text containing whitespaces must be enclosed in quotes. A whitespace is otherwise interpreted as the AND operator.

Examples:

hostgroup = "Web servers"

The search will return hosts with assigned host group named "Web servers".

hostgroup = Web servers

The search will return hosts in the host group Web with any field matching %servers%.

Temporal Values

Many date and time formats are accepted, including the following:

- "10 January 2017"
- "10 Jan 2017"
- 10-January-2017
- 10/January/2017
- "January 10, 2017"
- Today, Yesterday, and the like.



WARNING

Avoid ambiguous date formats, such as 02/10/2017 or 10-02-2017.

18.2. USING FREE TEXT SEARCH

When you enter free text, it will be searched for across multiple fields. For example, if you type "64", the search will return all hosts that have that number in their name, IP address, MAC address, and architecture.



NOTE

Multi-word queries must be enclosed in quotes, otherwise the whitespace is interpreted as the AND operator.

Because of searching across all fields, free text search results are not very accurate and searching can be slow, especially on a large number of hosts. For this reason, we recommend that you avoid free text and use more specific, syntax-based queries whenever possible.

18.3. MANAGING BOOKMARKS

You can save search queries as bookmarks for reuse. You can also delete or modify a bookmark.

Bookmarks appear only on the page on which they were created. On some pages, there are default bookmarks available for the common searches, for example, all **active** or **disabled** hosts.

18.3.1. Creating Bookmarks

This section details how to save a search query as a bookmark. You must save the search query on the relevant page to create a bookmark for that page, for example, saving a host related search query on the Hosts page.

Procedure

1. In the Satellite web UI, navigate to the page where you want to create a bookmark.
2. In the **Search** field, enter the search query you want to save.
3. Select the arrow to the right of the **Search** button and then select **Bookmark this search**.
4. In the **Name** field, enter a name for the new bookmark.
5. In the **Search query** field, ensure your search query is correct.
6. Ensure the **Public** checkbox is set correctly:
 - Select the **Public** checkbox to set the bookmark as public and visible to all users.
 - Clear the **Public** checkbox to set the bookmark as private and only visible to the user who created it.
7. Click **Submit**.

To confirm the creation, either select the arrow to the right of the **Search** button to display the list of bookmarks, or navigate to **Administer > Bookmarks** and then check the **Bookmarks** list for the name of the bookmark.

18.3.2. Deleting Bookmarks

You can delete bookmarks on the Bookmarks page.

Procedure

1. In the Satellite web UI, navigate to **Administer > Bookmarks**.
2. On the Bookmarks page, click **Delete** for the Bookmark you want to delete.
3. When the confirmation window opens, click **OK** to confirm the deletion.

To confirm the deletion, check the **Bookmarks** list for the name of the bookmark.

APPENDIX A. ADMINISTRATION SETTINGS

This section contains information about settings that you can edit in the Satellite web UI by navigating to **Administer > Settings**.

A.1. GENERAL SETTINGS

Setting	Default Value	Description
Administrator email address		The default administrator email address
Satellite URL		URL where your Satellite instance is reachable. See also Provisioning > Unattended URL .
Entries per page	20	Number of records shown per page in Satellite
Fix DB cache	No	Satellite maintains a cache of permissions and roles. When set to Yes , Satellite recreates this cache on the next restart.
DB pending seed	No	Should the foreman-rake db:seed be executed on the next run of the installer modules?
Capsule request timeout	60	Open and read timeout for HTTP requests from Satellite to Capsule (in seconds).
Login page footer text		Text to be shown in the login-page footer.
Show host power status	Yes	Show power status on the host index page. This feature calls to compute resource providers which may lead to decreased performance on the host listing page.
HTTP(S) proxy		Set a proxy for outgoing HTTP(S) connections from the Satellite product. System-wide proxies must be configured at the operating system level.
HTTP(S) proxy except hosts	[]	Set hostnames to which requests are not to be proxied. Requests to the local host are excluded by default.

Setting	Default Value	Description
Show Experimental Labs	No	Whether or not to show a menu to access experimental lab features (requires reload of page).
Append domain names to the host	Yes	If set to Yes , Satellite appends domain names when new hosts are provisioned.
Out of sync interval	30	Managed hosts report periodically, and if the time between reports exceeds this duration in minutes, hosts are considered out of sync. You can override this on your hosts by adding the outofsync_interval parameter, per host, at Hosts > All hosts > \$host > Edit > Parameters > Add Parameter .
Satellite UUID		Satellite instance ID. Uniquely identifies a Satellite instance.
Default language		The UI for new users uses this language.
Default timezone		The timezone to use for new users.
Instance title		The instance title is shown on the top navigation bar (requires a page reload).
Saved audits interval		Duration in days to preserve audit data. Leave empty to disable the audits cleanup.
New host details UI	Yes	Satellite loads the new UI for host details.

A.2. SATELLITE TASK SETTINGS

Setting	Default Value	Description
Sync task timeout	120	Number of seconds to wait for a synchronous task to finish before an exception is raised.
Enable dynflow console	Yes	Enable the dynflow console (/foreman_tasks/dynflow) for debugging.
Require auth for dynflow console	Yes	The user must be authenticated as having administrative rights before accessing the dynflow console.

Setting	Default Value	Description
Capsule action retry count	4	Number of attempts permitted to start a task on the Capsule before failing.
Capsule action retry interval	15	Time in seconds between retries.
Allow Capsule batch tasks	Yes	Enable batch triggering of tasks on the Capsule.
Capsule tasks batch size	100	Number of tasks included in one request to the Capsule if foreman_tasks_proxy_batch_trigger is enabled.
Tasks troubleshooting URL		URL pointing to the task troubleshooting documentation. It should contain a %{label} placeholder that is replaced with a normalized task label (restricted to only alphanumeric characters)). A %{version} placeholder is also available.
Polling intervals multiplier	1	Polling multiplier used to multiply the default polling intervals. You can use this to prevent polling too frequently for long running tasks.

A.3. TEMPLATE SYNC SETTINGS

Setting	Default Value	Description
Associate	New	Associate templates with OS, organization and location.
Branch		Default branch in Git repo.
Commit message	Templates export made by a Satellite user	Custom commit message for exported templates.
Dirname	/	The directory within the Git repo containing the templates.
Filter		Import or export of names matching this regex. Case-insensitive. Snippets are not filtered.
Force import	No	If set to Yes , locked templates are overwritten during an import.

Setting	Default Value	Description
Lock templates	Keep, do not lock new	How to handle lock for imported templates.
Metadata export mode	Refresh	Default metadata export mode. Possible options: refresh re-renders metadata. keep keeps existing metadata. remove exports the template without metadata.
Negate	No	Negate the filter for import or export.
Prefix		A string added as a prefix to imported templates.
Repo		Target path from where to import or export templates. Different protocols can be used, for example: /tmp/dir git://example.com https://example.com ssh://example.com When exporting to /tmp , note that production deployments may be configured to use private tmp .
Verbosity	No	Choose verbosity for Rake task importing templates.

A.4. DISCOVERED SETTINGS

Setting	Default Value	Description
Discovery location		Indicates the default location to place discovered hosts in.
Discovery organization		Indicates the default organization to which discovered hosts are added.

Setting	Default Value	Description
Interface fact	discovery_bootif	Fact name to use for primary interface detection.
Create bond interfaces	No	Automatically create a bond interface if another interface is detected on the same VLAN using LLDP.
Clean all facts	No	Clean all reported facts (except discovery facts) during provisioning.
Hostname facts	discovery_bootif	List of facts to use for the hostname (comma separated, first wins).
Auto provisioning	No	Use the provisioning rules to automatically provision newly discovered hosts.
Reboot	Yes	Automatically reboot or kexec discovered hosts during provisioning.
Hostname prefix	mac	The default prefix to use for the hostname. Must start with a letter.
Fact columns		Extra factor columns to show in host lists (comma separated).
Highlighted facts		Regex to organize facts for highlights section – e.g. ^(abc cde)\$.
Storage facts		Regex to organize facts for the storage section.
Software facts		Regex to organize facts for the software section.
Hardware facts		Regex to organize facts for the hardware section.
Network facts		Regex to organize facts for the network section.
IPMI facts		Regex to organize facts for the <i>Intelligent Platform Management Interface</i> (IPMI) section.

Setting	Default Value	Description
Lock PXE	No	Automatically generate a <i>Preboot Execution Environment</i> (PXE) configuration to pin a newly discovered host to discovery.
Locked PXELinux template name	pxelinux_discovery	PXELinux template to be used when pinning a host to discovery.
Locked PXEGrub template name	pxegrub_discovery	PXEGrub template to be used when pinning a host to discovery.
Locked PXEGrub2 template name	pxegrub2_discovery	PXEGrub2 template to be used when pinning a host to discovery.
Force DNS	Yes	Force the creation of DNS entries when provisioning a discovered host.
Error on existing NIC	No	Do not permit to discover an existing managed host matching the MAC of a provisioning <i>Network Interface Card</i> (NIC) (errors out early).
Type of name generator	Fact + prefix	Discovery hostname naming pattern.
Prefer IPv6	No	Prefer IPv6 to IPv4 when calling discovered nodes.

A.5. BOOT DISK SETTINGS

Setting	Default Value	Description
iPXE directory	/usr/share/ipxe	Path to directory containing iPXE images.
ISOLINUX directory	/usr/share/syslinux	Path to directory containing ISOLINUX images.
SYSLINUX directory	/usr/share/syslinux	Path to directory containing SYSLINUX images.
Grub2 directory	/var/lib/tftpboot/grub2	Path to directory containing grubx64.efi and shimx64.efi .
Host image template	Boot disk iPXE - host	iPXE template to use for host-specific boot disks.

Setting	Default Value	Description
Generic image template	Boot disk iPXE - generic host	iPXE template to use for generic host boot disks.
Generic Grub2 EFI image template	Boot disk Grub2 EFI - generic host	Grub2 template to use for generic <i>Extensible Firmware Interface</i> (EFI) host boot disks.
ISO generation command	genisoimage	Command to generate ISO image, use genisoimage or mkisofs .
Installation media caching	Yes	Installation media files are cached for full host images.
Allowed bootdisk types	[generic, host, full_host, subnet]	List of permitted bootdisk types. Leave blank to disable it.

A.6. RED HAT CLOUD SETTINGS

Setting	Default Value	Description
Automatic inventory upload	Yes	Enable automatic upload of your host inventory to the Red Hat cloud.
Synchronize recommendations Automatically	No	Enable automatic synchronization of Insights recommendations from the Red Hat cloud.
Obfuscate host names	No	Obfuscate hostnames sent to the Red Hat cloud.
Obfuscate host ipv4 addresses	No	Obfuscate IPv4 addresses sent to the Red Hat cloud.
Red Hat Cloud token	*****	Authentication token sent to the Red Hat cloud services. Used to authenticate requests to cloud APIs.
Exclude installed Packages	No	Exclude installed packages from upload to the Red Hat cloud.
Include parameters in insights-client reports	No	Include parameter tags from Satellite during import.

A.7. CONTENT SETTINGS

Setting	Default Value	Description
Default HTTP Proxy		Default HTTP Proxy for syncing content.
CDN SSL version		SSL version used to communicate with the CDN.
Default synced OS provisioning template	Kickstart default	Default provisioning template for operating systems created from synced content.
Default synced OS finish template	Kickstart default finish	Default finish template for new operating systems created from synced content.
Default synced OS user-data	Kickstart default user data	Default user data for new operating systems created from synced content.
Default synced OS PXELinux template	Kickstart default PXELinux	Default PXELinux template for new operating systems created from synced content.
Default synced OS PXEGrub template	Kickstart default PXEGrub	Default PXEGrub template for new operating systems created from synced content.
Default synced OS PXEGrub2 template	Kickstart default PXEGrub2	Default PXEGrub2 template for new operating systems created from synced content.
Default synced OS iPXE template	Kickstart default iPXE	Default iPXE template for new operating systems created from synced content.
Default synced OS partition table	Kickstart default	Default partitioning table for new operating systems created from synced content.
Default synced OS kexec template	Discovery Red Hat kexec	Default kexec template for new operating systems created from synced content.
Default synced OS Atomic template	Atomic Kickstart default	Default provisioning template for new atomic operating systems created from synced content.
Manifest refresh timeout	1200	Timeout when refreshing a manifest (in seconds).
Accept action timeout	20	Time in seconds to wait for a host to pick up a remote action.

Setting	Default Value	Description
Finish action timeout	3600	Time in seconds to wait for a host to finish a remote action.
Subscription connection enabled	Yes	Can communicate with the Red Hat Portal for subscriptions.
Installable errata from Content View	No	Calculate errata host status based only on errata in a host's Content View and Lifecycle Environment.
Restrict Composite Content View promotion	No	If this is enabled, a composite content view cannot be published or promoted, unless the content view versions that it includes exist in the target environment.
Check services before actions	Yes	Check the status of backend services such as pulp and candlepin before performing actions?
Batch size to sync repositories in	100	How many repositories should be synced concurrently on a Capsule. A smaller number may lead to longer sync times. A larger number will increase dynflow load.
Sync Capsules after Content View promotion	Yes	Whether or not to auto sync Capsules after a Content View promotion.
Default Custom Repository download policy	immediate	Default download policy for custom repositories. Either immediate or on_demand .
Default Red Hat Repository download policy	on_demand	Default download policy for enabled Red Hat repositories. Either immediate or on_demand .
Default Capsule download policy	on_demand	Default download policy for Capsule syncs. Either inherit , immediate , or on_demand .
Pulp export destination filepath	/var/lib/pulp/katello-export	On-disk location for exported repositories.
Pulp 3 export destination filepath	/var/lib/pulp/exports	On-disk location for Pulp 3 exported repositories.
Pulp client key	/etc/pki/katello/private/pulp-client.key	Path for SSL key used for Pulp server authentication.

Setting	Default Value	Description
Pulp client cert	<code>/etc/pki/katello/certs/pulp-client.crt</code>	Path for SSL certificate used for Pulp server authentication.
Sync Connection Timeout	300	Total timeout in seconds for connections when syncing.
Use remote execution by default	No	If enabled, remote execution is used instead of katello-agent for remote actions.
Delete Host upon unregister	No	When unregistering a host using subscription-manager, also delete the host record. Managed resources linked to the host such as virtual machines and DNS records might also be deleted.
Subscription manager name registration fact		When registering a host using subscription-manager, force use the specified fact for the host name (in the form of fact.fact).
Subscription manager name registration fact strict matching	No	If this is enabled, and register_hostname_fact is set and provided, registration looks for a new host by name only using that fact, and skips all hostname matching.
Default Location subscribed hosts	Default Location	Default location where new subscribed hosts are stored after registration.
Expire soon days	120	The number of days remaining in a subscription before you are reminded about renewing it.
Content View Dependency Solving Default	No	The default dependency solving value for new content views.
Host Duplicate DMI UUIDs	[]	If hosts fail to register because of duplicate <i>Desktop Management Interface</i> (DMI) UUIDs, add their comma-separated values here. Subsequent registrations generate a unique DMI UUID for the affected hosts.
Host Profile Assume	Yes	Enable new host registrations to assume registered profiles with matching hostname as long as the registering DMI UUID is not used by another host.

Setting	Default Value	Description
Host Profile Can Change In Build	No	Enable host registrations to bypass Host Profile Assume as long as the host is in build mode.
Host Can Re-Register Only In Build	No	Enable hosts to re-register only when they are in build mode.
Host Tasks Workers Pool Size	5	Number of workers in the pool to handle the execution of host-related tasks. When set to 0, the default queue is used. Restart of the dynflowd/foreman-tasks service is required.
Applicability Batch Size	50	Number of host applicability calculations to process per task.
Autosearch	Yes	For pages that support it, automatically perform the search while typing in search input.
Autosearch delay	500	If Autosearch is enabled, delay in milliseconds before executing searches while typing.
Pulp bulk load size	2000	The number of items fetched from a single paged Pulp API call.
Upload profiles without Dynflow	Yes	Enable Katello to update host installed packages, enabled repositories, and module inventory directly instead of wrapped in Dynflow tasks (try turning off if Puma processes are using too much memory).
Orphaned Content Protection Time	1440	Time in minutes to consider orphan content as orphaned.
Prefer registered through Capsule for remote execution	No	Prefer using a proxy to which a host is registered when using remote execution.
Allow deleting repositories in published content views	Yes	Enable removal of repositories that the user has previously published in one or more Content View versions.

A.8. AUTHENTICATION SETTINGS

Setting	Default Value	Description
OAuth active	Yes	Satellite will use OAuth for API authorization.
OAuth consumer key	*****	OAuth consumer key.
OAuth consumer secret	*****	OAuth consumer secret.
OAuth map users	No	Satellite maps users by username in the request-header. If this is disabled, OAuth requests have administrator rights.
Failed login attempts limit	30	Satellite blocks user logins from an incoming IP address for 5 minutes after the specified number of failed login attempts. Set to 0 to disable brute force protection.
Restrict registered Capsules	Yes	Only known Capsules can access features that use Capsule authentication.
Require SSL for capsules	Yes	Client SSL certificates are used to identify Capsules (:require_ssl should also be enabled).
Trusted hosts	[]	List of hostnames, IPv4, IPv6 addresses or subnets to be trusted in addition to Capsules for access to fact/report importers and ENC output.
SSL certificate	/etc/foreman/client_cert.pem	SSL Certificate path that Satellite uses to communicate with its proxies.
SSL CA file	/etc/foreman/proxy_ca.pem	SSL CA file path that Satellite uses to communicate with its proxies.
SSL private key	/etc/foreman/client_key.pem	SSL Private Key path that Satellite uses to communicate with its proxies.
SSL client DN env	HTTP_SSL_CLIENT_S_DN	Environment variable containing the subject DN from a client SSL certificate.
SSL client verify env	HTTP_SSL_CLIENT_VERIFY	Environment variable containing the verification status of a client SSL certificate.
SSL client cert env	HTTP_SSL_CLIENT_CERT	Environment variable containing a client's SSL certificate.

Setting	Default Value	Description
Server CA file		SSL CA file path used in templates to verify the connection to Satellite.
Websockets SSL key	etc/pki/katello/private/katello-apache.key	Private key file path that Satellite uses to encrypt websockets.
Websockets SSL certificate	/etc/pki/katello/certs/katello-apache.crt	Certificate path that Satellite uses to encrypt websockets.
Websockets encryption	Yes	VNC/SPICE websocket proxy console access encryption (websockets_ssl_key/cert setting required).
Login delegation logout URL		Redirect your users to this URL on logout. Enable Authorize login delegation also.
Authorize login delegation auth source user autocreate	External	Name of the external authentication source where unknown externally authenticated users (see Authorize login delegation) are created. Empty means no autocreation.
Authorize login delegation	No	Authorize login delegation with REMOTE_USER HTTP header.
Authorize login delegation API	No	Authorize login delegation with REMOTE_USER HTTP header for API calls too.
Idle timeout	60	Log out idle users after the specified number of minutes.
BCrypt password cost	9	Cost value of bcrypt password hash function for internal auth-sources (4 – 30). A higher value is safer but verification is slower, particularly for stateless API calls and UI logins. A password change is needed to affect existing passwords.
BMC credentials access	Yes	Permits access to BMC interface passwords through ENC YAML output and in templates.

Setting	Default Value	Description
OIDC JWKs URL		OpenID Connect <i>JSON Web Key Set</i> (JWKS) URL. Typically https://keycloak.example.com/auth/realms/<realm name>/protocol/openid-connect/certs when using Keycloak as an OpenID provider.
OIDC Audience	[]	Name of the OpenID Connect Audience that is being used for authentication. In the case of Keycloak this is the Client ID.
OIDC Issuer		The issuer claim identifies the principal that issued the <i>JSON Web tokens</i> (JWT), which exists at a /.well-known/openid-configuration in case of most of the OpenID providers.
OIDC Algorithm		The algorithm used to encode the JWT in the OpenID provider.

A.9. EMAIL SETTINGS

Setting	Default Value	Description
Email reply address		Email reply address for emails that Satellite is sending.
Email subject prefix		Prefix to add to all outgoing email.
Send welcome email	No	Send a welcome email including username and URL to new users.
Delivery method	Sendmail	Method used to deliver email.
SMTP enable StartTLS auto	Yes	SMTP automatically enables StartTLS.
SMTP OpenSSL verify mode	Default verification mode	When using TLS, you can set how OpenSSL checks the certificate.
SMTP address		SMTP address to connect to.
SMTP port	25	SMTP port to connect to.
SMTP HELO/EHLO domain		HELO/EHLO domain.

Setting	Default Value	Description
SMTP username		Username to use to authenticate, if required.
SMTP password	*****	Password to use to authenticate, if required.
SMTP authentication	none	Specify authentication type, if required.
Sendmail arguments	-i	Specify additional options to sendmail. Only used when the delivery method is set to sendmail.
Sendmail location	/usr/sbin/sendmail	The location of the sendmail executable. Only used when the delivery method is set to sendmail.

A.10. NOTIFICATIONS SETTINGS

Setting	Default Value	Description
RSS enable	Yes	Pull RSS notifications.
RSS URL	https://www.redhat.com/en/rss/blog/channel/red-hat-satellite	URL from which to fetch RSS notifications.

A.11. PROVISIONING SETTINGS

Setting	Default Value	Description
Host owner		Default owner on provisioned hosts, if empty Satellite uses the current user.
Root password	*****	Default encrypted root password on provisioned hosts.
Unattended URL		URL that hosts retrieve templates from during the build. When it starts with https, unattended, or userdata, controllers cannot be accessed using HTTP.

Setting	Default Value	Description
Safemode rendering	Yes	<p>Enables safe mode rendering of provisioning templates. The default and recommended option Yes denies access to variables and any object that is not listed in Satellite.</p> <p>When set to No, any object may be accessed by a user with permission to use templating features, either by editing templates, parameters or smart variables. This permits users full remote code execution on Satellite Server, effectively disabling all authorization. This is not a safe option, especially in larger companies.</p>
Access unattended without build	No	Enable access to unattended URLs without build mode being used.
Query local nameservers	No	Satellite queries the locally configured resolver instead of the SOA/NS authorities.
Installation token lifetime	360	Time in minutes that installation tokens should be valid for. Set to 0 to disable the token.
SSH timeout	120	Time in seconds before SSH provisioning times out.
Libvirt default console address	0.0.0.0	The IP address that should be used for the console listen address when provisioning new virtual machines using libvirt.
Update IP from built request	No	Satellite updates the host IP with the IP that made the build request.
Use short name for VMs	No	Satellite uses the short hostname instead of the FQDN for creating new virtual machines.
DNS timeout	[5, 10, 15, 20]	List of timeouts (in seconds) for DNS lookup attempts such as the dns_lookup macro and DNS record conflict validation.
Clean up failed deployment	Yes	Satellite deletes the virtual machine if the provisioning script ends with a non-zero exit code.

Setting	Default Value	Description
Type of name generator	Random-based	<p>Specifies the method used to generate a hostname when creating a new host.</p> <p>The default Random-based option generates a unique random hostname which you can but do not have to use. This is useful for users who create many hosts and do not know how to name them.</p> <p>The MAC-based option is for bare-metal hosts only. If you delete a host and create it later on, it receives the same hostname based on the MAC address. This can be useful for users who recycle servers and want them to always get the same hostname.</p> <p>The Off option disables the name generator function and leaves the hostname field blank.</p>
Default PXE global template entry		Default PXE menu item in a global template – local, discovery or custom, use blank for template default.
Default PXE local template entry		Default PXE menu item in local template – local, local_chain_hd0 , or custom, use blank for template default.
iPXE intermediate script	iPXE intermediate script	Intermediate iPXE script for unattended installations.
Destroy associated VM on host delete	No	Destroy associated VM on host delete. When enabled, VMs linked to hosts are deleted on Compute Resource. When disabled, VMs are unlinked when the host is deleted, meaning they remain on Compute Resource and can be re-associated or imported back to Satellite again. This does not automatically power off the VM
Maximum structured facts	100	Maximum number of keys in structured subtree, statistics stored in satellite::dropped_subtree_facts .
Default Global registration template	Global Registration	Global Registration template.

Setting	Default Value	Description
Default 'Host initial configuration' template	Linux host_init_config default	Default 'Host initial configuration' template, automatically assigned when a new operating system is created.
Global default PXEGrub2 template	PXEGrub2 global default	Global default PXEGrub2 template. This template is deployed to all configured TFTP servers. It is not affected by upgrades.
Global default PXELinux template	PXELinux global default	Global default PXELinux template. This template is deployed to all configured TFTP servers. It is not affected by upgrades.
Global default PXEGrub template	PXEGrub global default	Global default PXEGrub template. This template is deployed to all configured TFTP servers. It is not affected by upgrades.
Global default iPXE template	iPXE global default	Global default iPXE template. This template is deployed to all configured TFTP servers. It is not affected by upgrades.
Local boot PXEGrub2 template	PXEGrub2 default local boot	Template that is selected as PXEGrub2 default for local boot.
Local boot PXELinux template	PXELinux default local boot	Template that is selected as PXELinux default for local boot.
Local boot PXEGrub template	PXEGrub default local boot	Template that is selected as PXEGrub default for local boot.
Local boot iPXE template	iPXE default local boot	Template that is selected as iPXE default for local boot.
Manage PuppetCA	Yes	Satellite automates certificate signing upon provision of a new host.
Use UUID for certificates	No	Satellite uses random UUIDs for certificate signing instead of hostnames.
Show unsupported provisioning templates	No	Show unsupported provisioning templates. When enabled, all the available templates are shown. When disabled, only Red Hat supported templates are shown.

A.12. FACTS SETTINGS

Setting	Default Value	Description
Create new host when facts are uploaded	Yes	Satellite creates the host when new facts are received.
Location fact	satellite_location	Hosts created after a Puppet run are placed in the location specified by this fact.
Organization fact	satellite_organization	Hosts created after a Puppet run are placed in the organization specified by this fact. The content of this fact should be the full label of the organization.
Default location	Default Location	Hosts created after a Puppet run that did not send a location fact are placed in this location.
Default organization	Default Organization	Hosts created after a Puppet run that did not send an organization fact are placed in this organization.
Update hostgroup from facts	Yes	Satellite updates a host's hostgroup from its facts.
Ignore facts for operating system	No	Stop updating operating system from facts.
Ignore facts for domain	No	Stop updating domain values from facts.
Update subnets from facts	None	Satellite updates a host's subnet from its facts.
Ignore interfaces facts for provisioning	No	Stop updating IP and MAC address values from facts (affects all interfaces).
Ignore interfaces with matching identifier	[lo, en*v*, usb*, vnet*, macvtap*, vdsmdummy, veth*, tap*, qbr*, qvb*, qvo*, qr-*, qg-*, vlinuxbr*, vovsbr*, br-int]	Skip creating or updating host network interfaces objects with identifiers matching these values from incoming facts. You can use a * wildcard to match identifiers with indexes, e.g. macvtap* . The ignored interface raw facts are still stored in the database, see the Exclude pattern setting for more details.

Setting	Default Value	Description
Exclude pattern for facts stored in satellite	[lo, en*v*, usb*, vnet*, macvtap*, ;vdsmdummy;, veth*, tap*, qbr*, qvb*, qvo*, qr-*, qg-*, vlinuxbr*, vovsbr*, br-int, load_averages::*, memory::swap::available*, memory::swap::capacity, memory::swap::used*, memory::system::available*, memory::system::capacity, memory::system::used*, memoryfree, memoryfree_mb, swapfree, swapfree_mb, uptime_hours, uptime_days]	Exclude pattern for all types of imported facts (Puppet, Ansible, rhsm). Those facts are not stored in the satellite database. You can use a * wildcard to match names with indexes, e.g. ignore* filters out ignore, ignore123 as well as a::ignore or even a::ignore123::b.

A.13. CONFIGURATION MANAGEMENT SETTINGS

Setting	Default Value	Description
Create new host when report is uploaded	Yes	Satellite creates the host when a report is received.
Matchers inheritance	Yes	Satellite matchers are inherited by children when evaluating smart class parameters for hostgroups, organizations, and locations.
Default parameters lookup path	[fqdn, hostgroup, os, domain]	Satellite evaluates host smart class parameters in this order by default.
Interpolate ERB in parameters	Yes	Satellite parses ERB in parameters value in the ENC output.
Always show configuration status	No	All hosts show a configuration status even when a Puppet Capsule is not assigned.

A.14. REMOTE EXECUTION SETTINGS

Setting	Default Value	Description
Fallback to Any Capsule	No	Search the host for any proxy with Remote Execution. This is useful when the host has no subnet or the subnet does not have an execution proxy.
Enable Global Capsule	Yes	Search for Remote Execution proxy outside of the proxies assigned to the host. The search is limited to the host's organization and location.
SSH User	root	Default user to use for SSH. You can override per host by setting the remote_execution_ssh_user parameter.
Effective User	root	Default user to use for executing the script. If the user differs from the SSH user, su or sudo is used to switch the user.
Effective User Method	sudo	The command used to switch to the effective user. One of [sudo, dzdo, su]
Effective user password	*****	Effective user password. See Effective User .
Sync Job Templates	Yes	Whether to sync templates from disk when running db:seed .
SSH Port	22	Port to use for SSH communication. Default port 22. You can override per host by setting the remote_execution_ssh_port parameter.
Connect by IP	No	Whether the IP addresses on host interfaces are preferred over the FQDN. It is useful when the DNS is not resolving the FQDNs properly. You can override this per host by setting the remote_execution_connect_by_ip parameter. For dual-stacked hosts, consider the remote_execution_connect_by_ip_prefer_ipv6 setting.

Setting	Default Value	Description
Prefer IPv6 over IPv4	No	When connecting using an IP address, are IPv6 addresses preferred? If no IPv6 address is set, it falls back to IPv4 automatically. You can override this per host by setting the remote_execution_connect_by_ip_prefer_ipv6 parameter. By default and for compatibility, IPv4 is preferred over IPv6.
Default SSH password	*****	Default password to use for SSH. You can override per host by setting the remote_execution_ssh_password parameter.
Default SSH key passphrase	*****	Default key passphrase to use for SSH. You can override per host by setting the remote_execution_ssh_key_passphrase parameter.
Workers pool size	5	Number of workers in the pool to handle the execution of the remote execution jobs. Restart of the dynflowd/satellite-tasks service is required.
Cleanup working directories	Yes	Whether working directories are removed after task completion. You can override this per host by setting the remote_execution_cleanup_working_dirs parameter.
Cockpit URL		Where to find the Cockpit instance for the Web Console button. By default, no button is shown.
Form Job Template	Run Command - SSH Default	Choose a job template that is pre-selected in job invocation form.
Job Invocation Report Template	Jobs - Invocation report template	Select a report template used for generating a report for a particular remote execution job.

A.15. ANSIBLE SETTINGS

Setting	Default Value	Description
---------	---------------	-------------

Setting	Default Value	Description
Private Key Path		Use this to supply a path to an SSH Private Key that Ansible uses instead of a password. Override with the ansible_ssh_private_key_file host parameter.
Connection type	ssh	Use this connection type by default when running Ansible playbooks. You can override this on hosts by adding the ansible_connection parameter.
WinRM cert Validation	validate	Enable or disable WinRM server certificate validation when running Ansible playbooks. You can override this on hosts by adding the ansible_winrm_server_cert_validation parameter.
Default verbosity level	Disabled	Satellite adds this level of verbosity for additional debugging output when running Ansible playbooks.
Post-provision timeout	360	Timeout (in seconds) to set when Satellite triggers an Ansible roles task playbook after a host is fully provisioned. Set this to the maximum time you expect a host to take until it is ready after a reboot.
Ansible report timeout	30	Timeout (in minutes) when hosts should have reported.
Ansible out of sync disabled	No	Disable host configuration status turning to out of sync for Ansible after a report does not arrive within the configured interval.
Default Ansible inventory report template	Ansible - Ansible Inventory	Satellite uses this template to schedule the report with Ansible inventory.
Ansible roles to ignore	[]	The roles to exclude when importing roles from Capsule. The expected input is comma separated values and you can use * wildcard metacharacters. For example: foo*, *b*, *bar .

Setting	Default Value	Description
Capsule tasks batch size for Ansible		Number of tasks which should be sent to the Capsule in one request if satellite_tasks_proxy_batch_trigger is enabled. If set, it overrides satellite_tasks_proxy_batch_size setting for Ansible jobs.