



Red Hat Satellite 6.4

Upgrading and Updating Red Hat Satellite

Upgrading and updating Red Hat Satellite Server and Capsule Server

Red Hat Satellite 6.4 Upgrading and Updating Red Hat Satellite

Upgrading and updating Red Hat Satellite Server and Capsule Server

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

Legal Notice

Copyright © 2018 Red Hat, Inc.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

Abstract

This guide describes upgrading and updating Red Hat Satellite Server, Capsule Server, and hosts.

Table of Contents

CHAPTER 1. UPGRADE OVERVIEW	3
1.1. PREREQUISITES	3
1.2. UPGRADE PATHS	4
1.3. FOLLOWING THE PROGRESS OF THE UPGRADE	5
CHAPTER 2. CLONING SATELLITE SERVER	6
2.1. CLONING PROCESS OVERVIEW	6
2.2. PREREQUISITES	6
2.3. PULP DATA CONSIDERATIONS	7
2.4. CLONING SATELLITE SERVER	7
2.4.1. Preparing the source server for cloning	7
2.4.2. Cloning to the Target Server	8
CHAPTER 3. UPGRADING RED HAT SATELLITE	11
3.1. UPGRADING SATELLITE SERVER	11
3.1.1. Upgrading a Connected Satellite Server	12
3.1.2. Upgrading a Disconnected Satellite Server	15
3.2. UPGRADING CAPSULE SERVERS	19
3.3. UPGRADING SATELLITE CLIENTS	21
3.4. POST-UPGRADE TASKS	23
3.4.1. Upgrading Discovery	23
3.4.1.1. Upgrading Discovery on Satellite Server	23
3.4.1.2. Upgrading Discovery on Capsule Servers	24
3.4.1.3. Verifying Subnets have a Template Capsule	24
3.4.2. Upgrading virt-who	25
3.4.3. Removing the Previous Version of the Satellite Tools Repository	25
CHAPTER 4. UPGRADING PUPPET	26
4.1. UPGRADE PATH	26
4.2. TESTING PUPPET MODULES	27
4.3. PREPARING FOR THE UPGRADE	27
CHAPTER 5. UPDATING SATELLITE SERVER, CAPSULE SERVER, AND CONTENT HOSTS	29
5.1. UPDATING SATELLITE SERVER	29
5.2. UPDATING CAPSULE SERVER	30
5.3. UPDATING CONTENT HOSTS	31

CHAPTER 1. UPGRADE OVERVIEW

This chapter details the prerequisites and available upgrade paths to Red Hat Satellite 6.4. Review this information before upgrading your current Red Hat Satellite 6 installation.

In this guide, the terms update, upgrade, and migrate have the following meanings:

Upgrading

The process of advancing your Satellite Server and Capsule Server installations from a y-stream release to the next, for example Satellite 6.3 to Satellite 6.4.

Updating

The process of advancing your Satellite Server and Capsule Server installations from a z-stream release to the next, for example Satellite 6.3.0 to Satellite 6.3.1.

Migrating

The process of moving an existing Satellite installation to another Red Hat Enterprise Linux server.

For interactive upgrade instructions, you can also use the Red Hat Satellite Upgrade Helper on the Red Hat Customer Portal. This application provides you with an exact guide to match your current version number. You can find instructions that are specific to your upgrade path, as well as steps to prevent known issues. For more information, see [Satellite Upgrade Helper](#) on the customer portal.



IMPORTANT

The Red Hat Satellite Server and Capsule Server y-stream versions must match. For example, a 6.3 Satellite Server is not compatible with a 6.4 Capsule Server and a 6.4 Satellite Server is not compatible with a 6.3 Capsule Server. Mismatching Satellite Server and Capsule Server versions will result in the Capsule Server failing silently. However, a Capsule Server using one z-stream version older than the Satellite Server is supported. For example, a Satellite 6.3.1 Satellite Server is compatible with a 6.3.0 Capsule Server.

1.1. PREREQUISITES

Upgrading to Satellite 6.4 affects your entire Satellite infrastructure. Before proceeding, complete the following:

- Read the [Red Hat Satellite 6.4 Release Notes](#).
- Review this guide so that you are aware of the upgrade process and its impact.
- Plan your upgrade path. For more information, see [Section 1.2, “Upgrade Paths”](#).
- Red Hat Satellite 6.4 uses Puppet 5. Only Puppet 4 to Puppet 5 migrations are supported. You must upgrade to Puppet 4 prior to upgrading to Satellite 6.4. For information about upgrading to Puppet 4, see the [Upgrading Puppet](#) section in the *Satellite 6.3 Upgrading and Updating Red Hat Satellite* guide.
- Plan for the required Satellite downtime. The upgrade process requires that Satellite services be shut down for the duration.
- Plan for the storage requirements. The storage requirements for Red Hat Satellite 6.4 have changed from the previous version. For more information, see [Storage Requirements and Guidelines](#) in *Installing Satellite Server from a Connected Network*

- Back up your Satellite Server and all Capsule Servers. For more information, see [Backing Up and Restoring Satellite Server and Capsule Server](#) in the *Administering Red Hat Satellite 6.3* guide.



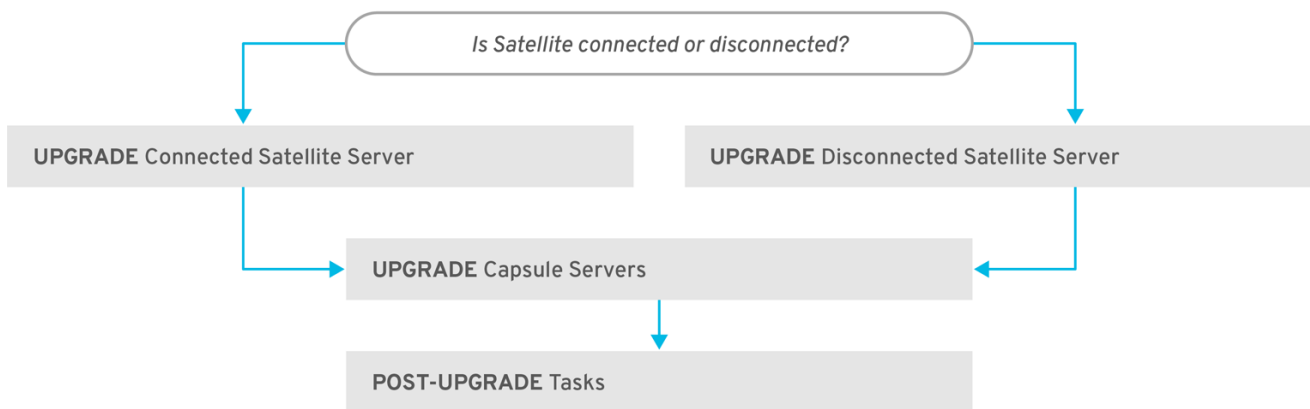
WARNING

If you customize configuration files, manually or use a tool such as Hiera, these customizations are overwritten when the installation script runs during upgrading or updating. You can use the `--noop` option with the `satellite-installer` script to test for changes. For more information, see the Red Hat Knowledgebase solution [How to use the noop option to check for changes in Satellite config files during an upgrade](#).

1.2. UPGRADE PATHS

You can upgrade to Red Hat Satellite 6.4 from Red Hat Satellite 6.3. Satellite Servers and Capsule Servers on earlier versions must first be upgraded to Satellite 6.3. For more details, see the [Satellite 6.3 Upgrading and Updating Red Hat Satellite](#) guide.

Figure 1.1. Overview of Satellite 6.4 Upgrade Paths



SATELLITE_461763_0718



WARNING

Upgrading from the Beta to GA version is not supported.

The high level steps in upgrading to Satellite 6.4 are as follows.

1. Clone your existing Satellite Servers. For more information, see [Chapter 2, Cloning Satellite Server](#).
2. Upgrade Satellite Server and all Capsule Servers to Satellite 6.4. For more information, see [Section 3.1, “Upgrading Satellite Server”](#).

3. Upgrade the Satellite tools on all Satellite clients. For more information, see [Section 3.3, “Upgrading Satellite Clients”](#).

Self-Registered Satellites

You cannot upgrade a self-registered Satellite. You must migrate a self-registered Satellite to the Red Hat Content Delivery Network (CDN) and then perform the upgrade. To migrate a self-registered Satellite to the CDN, see [Migrating Self-Registered Satellites](#) in the *Satellite 6.3 Upgrading and Updating Red Hat Satellite* guide.

1.3. FOLLOWING THE PROGRESS OF THE UPGRADE

Because of the lengthy upgrade time, use a utility such as **screen** to suspend and reattach a communication session. You can then check the upgrade progress without staying connected to the command shell continuously. For more information about using the screen command, see [How do I use the screen command?](#) article in the *Red Hat Knowledge Base*. You can also see the **screen** manual page for more information.

If you lose connection to the command shell where the upgrade command is running you can see the logs in `/var/log/foreman-installer/satellite.log` to check if the process completed successfully.

CHAPTER 2. CLONING SATELLITE SERVER

When you upgrade Satellite Server, you can optionally create a clone of your Satellite to ensure that you do not lose any data while you upgrade. After your upgrade is complete, you can then decommission the earlier version of Satellite Server.

Use the following procedures to clone your Satellite instances to preserve your environments in preparation for upgrade.

The Satellite clone tool does not support migrating a Capsule Server to Red Hat Enterprise Linux 7. Instead you must backup the existing Capsule Server, restore it on Red Hat Enterprise Linux 7, then reconfigure the Capsule Server.

Terminology

Ensure that you understand the following terms:

Source server: the server that you clone

Target server: the new server that you copy files to and clone the source server to.

2.1. CLONING PROCESS OVERVIEW

1. Back up the source server.
2. Clone the source server to the target server.
3. Power off the source server.
4. Update the network configuration on the target server to match the target server's IP address with its new host name.
5. Restart **goferd** in Content hosts and Capsules to refresh the connection.
6. Test the new target server.

2.2. PREREQUISITES

To clone Satellite Server, ensure that you have the following resources available:

- A minimal install of Red Hat Enterprise Linux 7 server to become the target server. Do not install Red Hat Enterprise Linux 7 software groups, or third-party applications. Ensure that your server complies with all the specifications of [Preparing your Environment for Installation](#) in *Installing Satellite Server from a Connected Network*.
- A backup from Satellite versions 6.1, 6.2, or 6.3 that you make using the **katello-backup** script. You can use a backup with or without Pulp data.
- A Satellite subscription for the target server.

Before you begin cloning, ensure the following conditions exist:

- The target server is on an isolated network. This avoids unwanted communication with Capsule Servers and hosts
- The target server has the capacity to store all your backup files from the source server.

2.3. PULP DATA CONSIDERATIONS

You can clone Satellite server without including Pulp data. However, for your cloned environment to work, you do require Pulp data. If the target server does not have Pulp data, it is not a fully working Satellite.

To transfer Pulp data to a target server, you have two options:

- Clone using backup with Pulp data
- Clone using backup without Pulp data and copy `/var/lib/pulp` manually from source server.

If your `pulp_data.tar` file is greater than 500 GB, or if you use a slow storage system, such as NFS, and your `pulp_data.tar` file is greater than 100 GB, do not include `pulp_data.tar` in the backup because this can cause memory errors during extraction. Copy the `pulp_data.tar` file from the source server to the target server.

To back up without Pulp data

Follow the steps in the procedure in [Section 2.4, “Cloning Satellite Server”](#) and replace the steps that involve cloning with Pulp data with the following steps:

1. Perform a backup with MongoDB and PostgreSQL databases active excluding the Pulp data:

```
# katello-service stop
# katello-service start --only mongod,postgresql
# katello-backup --skip-pulp-content /var/backup --assumeyes
```

2. Stop and disable all Satellite services

```
# katello-service stop
# for i in $(katello-service list | awk '{print $1}' | grep service); \
do systemctl disable $i ;done
```

3. Copy the Pulp data to the target server:

```
# rsync --archive --partial --progress --compress \
/var/lib/pulp target_server.example.com:/var/lib/pulp
```

Proceed to [Section 2.4.2, “Cloning to the Target Server”](#).

2.4. CLONING SATELLITE SERVER

Use the following procedures to clone Satellite Server. Note that because of the high volume of data that you must copy and transfer as part of these procedures, it can take a significant amount of time to complete.

2.4.1. Preparing the source server for cloning

On the source server, complete the following steps:

1. Verify the Pool ID of your Satellite subscription:

```
# subscription-manager list --consumed \
--matches 'Red Hat Satellite'|grep "Pool ID:"|awk '{print $3}'
```

2. Remove the Red Hat Satellite subscription.

```
# subscription-manager remove --serial=$(subscription-manager list \
--consumed \
--matches 'Red Hat Satellite'|grep "Serial:"|awk '{print $2}')
```

3. Determine the size of the Pulp data:

```
# du -sh /var/lib/pulp/
```

4. If you have less than 500 GB of Pulp data, perform a backup with MongoDB and PostgreSQL databases active including the Pulp data. If you have more than 500 GB of Pulp data, skip the following steps and complete the steps in [Section 2.3, “Pulp Data Considerations”](#) before you continue.

```
# katello-service stop
# katello-service start --only mongod,postgresql
# katello-backup /var/backup --assumeeyes
```

5. Stop and disable all Satellite services:

```
# katello-service stop
# for i in $(katello-service list| awk '{print $1}'|grep service); \
do systemctl disable $i ;done
```

Proceed to [Section 2.4.2, “Cloning to the Target Server”](#).

2.4.2. Cloning to the Target Server

To clone your server, complete the following steps on your target server:

1. The **satellite-clone** tool defaults to using `/backup/` as the backup folder. If you copy to a different folder, update the **backup_dir** variable in the `/etc/satellite-clone/satellite-clone-vars.yml` file.
2. Place the backup files in the `/backup` folder. You can either mount the shared storage or copy the following files: **candlepin.dump**, **config_files.tar.gz**, **foreman.dump**, and the **mongo_dump** folder.
3. Power off the source server.
4. Enter the following commands to register to the Customer Portal, attach subscriptions, and enable only the required subscriptions:

```
# subscription-manager register your_customer_portal_credentials
# subscription-manager attach --pool=pool_ID
# subscription-manager repos --disable=*
# subscription-manager repos \
--enable=rhel-7-server-rpms \
```

```
--enable=rhel-server-rhsc1-7-rpms \
--enable=rhel-7-server-satellite-maintenance-6-rpms \
--enable=rhel-7-server-satellite-6.3-rpms
```

5. Install the **satellite-clone** package

```
# yum install satellite-clone
```

After you install the **satellite-clone** tool, you can adjust any configuration to suit your own deployment in the `/etc/satellite-clone/satellite-clone-vars.yml` file.

6. Run the **satellite-clone** tool.

```
# satellite-clone
```

7. Reconfigure DHCP, DNS, TFTP and remote execution services. The cloning process disables these services on the target Satellite Server to avoid conflict with the source Satellite Server.
8. Reconfigure and enable DHCP, DNS, TFTP in the Satellite web UI. For more information, see [Configuring DNS, DHCP, and TFTP on Satellite Server](#) in *Installing Satellite Server from a Connected Network*.
9. Enable remote execution:

```
# satellite-installer --scenario satellite \
--enable-foreman-plugin-remote-execution \
--enable-foreman-proxy-plugin-remote-execution-ssh
```

10. Log on to the Satellite web UI, with the username **admin** and the password **changeme**. Immediately update the admin password to secure credentials.
11. Ensure that the correct organization is selected.
12. Navigate to **Content > Subscriptions**, then click **Manage Manifest**.
13. Click the **Refresh** button, then click **Close** to return to the list of subscriptions.
14. Verify that the available subscriptions are correct.
15. Follow the instructions in the `/usr/share/satellite-clone/logs/reassociate_capsules.txt` file to restore the associations between Capsules and their lifecycle environments.
16. Update your network configuration, for example, DNS, to match the target server's IP address with its new host name. The **satellite-clone** tool changes the hostname to the source server's hostname. If you want to change the hostname to something different, you can use the **satellite-change-hostname** tool. For more information, see [Renaming a Satellite or Capsule Server](#) in *Administering Red Hat Satellite*.
17. If the source server uses the **virt-who** daemon, install and configure it on the target server. Copy all the **virt-who** configuration files in the `/etc/virt-who.d/` directory from the source server to the same directory on the target server. For more information, see [Virt-who Installation and Configuration Overview](#) in the *Virtual instances Guide*.
18. Restart **goferd** on all registered content hosts and Capsules.

```
| systemctl restart goferd
```

After you perform an upgrade using the following chapters, you can safely decommission the source server.

CHAPTER 3. UPGRADING RED HAT SATELLITE



WARNING

If you have Satellite 6 installed in a high availability configuration, contact Red Hat Support before upgrading to Satellite 6.4.

Use this chapter to upgrade your existing Red Hat Satellite environment to Red Hat Satellite 6.4.

The chapter includes:

- [Section 3.1, “Upgrading Satellite Server”](#)
- [Section 3.2, “Upgrading Capsule Servers”](#)
- [Section 3.4, “Post-Upgrade Tasks”](#)

Before upgrading, see the [Section 1.1, “Prerequisites”](#).

3.1. UPGRADING SATELLITE SERVER

This section describes how to upgrade Satellite Server from 6.3 to 6.4. You can upgrade from any minor version of Red Hat Satellite Server 6.3.

Before You Begin

- Review and update your firewall configuration prior to upgrading your Satellite Server. For more information, see [Ports and Firewalls Requirements](#) in *Installing Satellite Server from a Connected Network*.
- Ensure that you do not delete the manifest from the Customer Portal or in the Satellite Web UI because this removes all the entitlements of your content hosts.
- Back up and remove all Foreman hooks before upgrading. Restore any hooks only after Satellite is known to be working after the upgrade is complete.
- If you have edited any of the default templates, back up the files either by cloning or exporting them. Cloning is the recommended method because that prevents them being overwritten in future updates or upgrades. To confirm if a template has been edited, you can view its **History** before you upgrade or view the changes in the audit log after an upgrade. In the web UI, Navigate to **Monitor > Audits** and search for the template to see a record of changes made. If you use the export method, restore your changes by comparing the exported template and the default template, manually applying your changes.
- If the `/root/.hammer/defaults.yml` file exists on your Satellite Server, create a back up of the file, and then delete `/root/.hammer/defaults.yml` before upgrading. Restore the file only after Satellite is known to be working after the upgrade is complete. You must apply this workaround until Red Hat [Bug 1632768](#) is resolved.

Capsule Considerations

- Ensure the Red Hat Satellite Capsule 6.4 repository is enabled in Satellite Server and synchronized. Until [Red Hat Bug 1305040](#) is resolved, allow time for the Capsules to synchronize these changes.
- If you use Content Views to control updates to a Capsule Server's base operating system, or for the Capsule Server repository, you must publish updated versions of those Content Views.



WARNING

If you implemented custom certificates, you must retain the content of both the `/root/ssl-build` directory and the directory in which you created any source files associated with your custom certificates.

Failure to retain these files during an upgrade causes the upgrade to fail. If these files have been deleted, they must be restored from a backup in order for the upgrade to proceed.

Configuring the BASH shell

The BASH shell stores the location of a binary in a hash table. During the upgrade, the location of the **katello-service** script is changed, but BASH does not register this change, and **foreman-maintain** fails if it calls the script after the change.

- Optional: Before the upgrade, users of the BASH shell can set the **checkhash** option temporarily to ensure **katello-service** works after the installer completes. Enter a command as follows in your BASH shell:

```
# shopt -s checkhash
```

- After a successful or failed upgrade, in all currently running BASH shells, enter the following command:

```
# hash -d katello-service 2> /dev/null
```

Upgrade Scenarios

- To upgrade a Satellite Server connected to the Red Hat Content Delivery Network, proceed to [Section 3.1.1, “Upgrading a Connected Satellite Server”](#).
- To upgrade a Satellite Server not connected to the Red Hat Content Delivery Network, proceed to [Section 3.1.2, “Upgrading a Disconnected Satellite Server”](#).

You cannot upgrade a self-registered Satellite. You must migrate a self-registered Satellite to the Red Hat Content Delivery Network (CDN) and then perform the upgrade. To migrate a self-registered Satellite to the CDN, see [Migrating Self-Registered Satellites](#) in the *Satellite 6.3 Upgrading and Updating Red Hat Satellite* guide.

3.1.1. Upgrading a Connected Satellite Server

Use this procedure for a Satellite Server connected to the Red Hat Content Delivery Network.

**WARNING**

If you customize configuration files, manually or using a tool such as Hiera, these changes are overwritten when the installation script runs during upgrading or updating. You can use the `--noop` option with the `satellite-installer` script to test for changes. For more information, see the Red Hat Knowledgebase solution [How to use the noop option to check for changes in Satellite config files during an upgrade](#).

Upgrade Satellite Server

1. Create a backup.
 - On a virtual machine, take a snapshot.
 - On a physical machine, create a backup.
For more information about backups, see [Backing Up and Restoring Satellite Server and Capsule Server](#) in the *Administering Red Hat Satellite 6.3* guide.
2. Back up the DNS and DHCP configuration files `/etc/zones.conf` and `/etc/dhcp/dhcpd.conf` as the installer only supports one domain or subnet, and therefore restoring changes from these backups might be required.
3. If you have made manual edits to DNS or DHCP configuration files and do not want to overwrite the changes, enter the following command:


```
# satellite-installer --foreman-proxy-dns-managed=false \
--foreman-proxy-dhcp-managed=false
```
4. In the Satellite web UI, navigate to **Hosts > Discovered hosts**. On the Discovered Hosts page, power off and then delete the discovered hosts. From the **Select an Organization** menu, select each organization in turn and repeat the process to power off and delete the discovered hosts. Make a note to reboot these hosts when the upgrade is complete.
5. In the Satellite web UI, navigate to **Content > Red Hat Subscriptions**, and then click **Manage Manifest**. In the Subscription Manifest pane, click the **Actions** tab, and then click **Refresh Manifest** to download the latest copy of the Subscription Manifest.
6. Configure the repositories:
 - a. In the Satellite web UI, navigate to **Content > Red Hat Repositories** and select the **RPMs** tab.
 - b. From the **PRODUCT** list, find and expand **Red Hat Enterprise Linux Server**.
 - c. From the **REPOSITORY SET** list, find and expand **Red Hat Satellite Tools 6.4 (for RHEL7 Server) (RPMs)**.
 - d. Select **Red Hat Satellite Tools 6.4 for RHEL 7 Server RPMs x86_64**.
7. Synchronize the newly enabled repositories:
 - a. In the Satellite web UI, navigate to **Content > Sync Status**.

b. Click the arrow next to the product to view available repositories.

c. Select the repositories for 6.4.

d. Click **Synchronize Now**.

If you get an error when trying to update a repository, ensure you do not delete the manifest from the Customer Portal or in the Satellite Web UI because this removes all the entitlements of your content hosts. Refresh the manifest and if the problem persists, raise a support request.

8. Update any pre-existing Content Views that utilize 6.3 version repositories with the new version for 6.4. Publish and promote updated versions of any Content Views that now have the new 6.4 version repositories.

9. Refresh your subscription:

```
# subscription-manager refresh
```

10. Enable the Satellite Maintenance and Red Hat Enterprise Linux Ansible repositories:

```
# subscription-manager repos \
--enable rhel-7-server-satellite-maintenance-6-rpms \
--enable rhel-7-server-ansible-2.6-rpms
```

11. Enter the following command to install **foreman-maintain** or to update it to the latest version:

```
# yum install rubygem-foreman_maintain
```

12. Check the available versions to confirm the version you want is listed:

```
# foreman-maintain upgrade list-versions
```

13. Use the health check option to determine if the system is ready for upgrade. When prompted, enter the hammer admin user credentials to configure **foreman-maintain** with hammer credentials. These changes are applied to the **/etc/foreman-maintain/foreman-maintain-hammer.yml** file.

```
# foreman-maintain upgrade check --target-version 6.4
```

Review the results and address any highlighted error conditions before performing the upgrade.

14. Because of the lengthy upgrade time, use a utility such as **screen** to suspend and reattach a communication session. You can then check the upgrade progress without staying connected to the command shell continuously. For more information about using the screen command, see [How do I use the screen command?](#) article in the *Red Hat Knowledge Base*.

If you lose connection to the command shell where the upgrade command is running you can see the logged messages in the **/var/log/foreman-installer/satellite.log** file to check if the process completed successfully.

15. Perform the upgrade:

```
# foreman-maintain upgrade run --target-version 6.4
```

16. If using a BASH shell, after a successful or failed upgrade, enter:

```
# hash -d foreman-maintain service 2> /dev/null
```

17. Check and restore any changes required to the DNS and DHCP configuration files using the backups that you make.
18. If you make changes in the previous step, restart Satellite services.

```
# foreman-maintain service restart
```

19. If you have the OpenSCAP plug-in installed, but do not have the default OpenSCAP content available, enter the following command:

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

3.1.2. Upgrading a Disconnected Satellite Server

Use this procedure for a Satellite Server not connected to the Red Hat Content Delivery Network.



WARNING

If you customize configuration files, manually or using a tool such as Hiera, these changes are overwritten when the installation script runs during upgrading or updating. You can use the **- -noop** option with the `satellite-installer` script to test for changes. For more information, see the Red Hat Knowledgebase solution [How to use the noop option to check for changes in Satellite config files during an upgrade](#).

Before You Begin

- Review and update your firewall configuration before upgrading your Satellite Server. For more information, see [Ports and Firewalls Requirements](#) in *Installing Satellite Server from a Disconnected Network*.
- Ensure that you do not delete the manifest from the Customer Portal or in the Satellite Web UI because this removes all the entitlements of your content hosts.
- Back up and remove all Foreman hooks before upgrading. Reinstate hooks only after Satellite is known to be working after the upgrade is complete.



WARNING

If you implemented custom certificates, you must retain the content of both the `/root/ssl-build` directory and the directory in which you created any source files associated with your custom certificates.

Failure to retain these files during an upgrade causes the upgrade to fail. If these files have been deleted, they must be restored from a backup in order for the upgrade to proceed.

Upgrade Disconnected Satellite Server

1. Create a backup.
 - On a virtual machine, take a snapshot.
 - On a physical machine, create a backup.
2. A pre-upgrade script is available to detect conflicts and list hosts which have duplicate entries in Satellite Server that can be unregistered and deleted after upgrade. In addition, it will detect hosts which are not assigned to an organization. If a host is listed under **Hosts > All hosts** without an organization association and if a content host with same name has an organization already associated with it then the content host will automatically be unregistered. This can be avoided by associating such hosts to an organization before upgrading.
Run the pre-upgrade check script to get a list of hosts that can be deleted after upgrading. If any unassociated hosts are found, associating them to an organization before upgrading is recommended.

```
# foreman-rake katello:upgrade_check
```

If the upgrade check reports a failure due to running tasks, then it is recommended that you wait for the tasks to complete. It is possible to cancel some tasks, but you should follow the guidance in the Red Hat Knowledgebase solution [How to manage paused tasks on Red Hat Satellite 6](#) to understand which tasks are safe to cancel and which are not safe to cancel.

3. Back up the DNS and DHCP configuration files `/etc/zones.conf` and `/etc/dhcp/dhcpd.conf` as the installer only supports one domain or subnet, and therefore restoring changes from these backups might be required.
4. If you have made manual edits to DNS or DHCP configuration files and do not want to overwrite the changes, run the installer script as follows:

```
# satellite-installer --foreman-proxy-dns-managed=false \
--foreman-proxy-dhcp-managed=false
```

5. In the Satellite web UI, navigate to **Hosts > Discovered hosts**. If there are discovered hosts available, turn them off and then delete all entries under the **Discovered hosts** page. Select all other organizations in turn using the organization setting menu and repeat this action as required. Reboot these hosts after the upgrade has completed.

6. Make sure all external Capsule Servers are assigned to an organization, otherwise they might get unregistered due to host-unification changes.

7. Disable the Puppet 4 repository:

```
# subscription-manager repos \
--disable=rhel-7-server-satellite-6.3-puppet4-rpms
```

8. Stop Satellite services.

```
# katello-service stop
```

9. Obtain the latest ISO file, mount it, and install the packages by following the procedures in the [Downloading and Installing from a Disconnected Network](#) section of the *Installing Satellite Server from a Disconnected Network*.
10. If you have custom Apache server configurations, they will be reverted to the installation defaults in the next step. If you want to see what will be changed **when you perform the upgrade**, you can enter the upgrade command with the **--noop** (no operation) option and review the changes that will be applied when you enter the upgrade command in the following step. If you choose not to do this test, skip to the next step now. Alternatively, proceed as follows:

- a. Add the following line to the `/etc/httpd/conf/httpd.conf` configuration file.

```
Include /etc/httpd/conf.modules.d/*.conf
```

- b. Restart the **httpd** service.

```
# systemctl restart httpd
```

- c. Start the **postgresql** and **mongod** database services.

```
# systemctl start postgresql
# systemctl start mongod
```

- d. Run the installer script with the **--noop** option as follows.

```
# satellite-installer --scenario satellite --upgrade --verbose --
noop
```

Review the `/var/log/foreman-installer/satellite.log` to see what changes **would be applied** if the **--noop** option was omitted. Look for the **+++** and **---** symbols indicating changes to configurations files. Because the above "no operation" option does not actually create the files, and some Puppet resources in the module expect them to be there, some failure messages are to be expected.

- e. Stop Satellite services.

```
# katello-service stop
```

11. Because of the lengthy upgrade time, use a utility such as **screen** to suspend and reattach a communication session. You can then check the upgrade progress without staying connected to the command shell continuously. For more information about using the `screen` command, see

[How do I use the screen command?](#) article in the *Red Hat Knowledge Base*.

If you lose connection to the command shell where the upgrade command is running you can see the logs in `/var/log/foreman-installer/satellite.log` to check if the process completed successfully.

12. Enter the following command to install **foreman-maintain** or to update it to the latest version:

```
# yum install rubygem-foreman_maintain
```

13. Check the available versions to confirm the version you want is listed:

```
# foreman-maintain upgrade list-versions
```

14. Use the health check option to determine if the system is ready for upgrade. When prompted, enter the hammer admin user credentials to configure **foreman-maintain** with hammer credentials. These changes are applied to the `/etc/foreman-maintain/foreman-maintain-hammer.yml` file.

```
# foreman-maintain upgrade check --target-version 6.4
```

Review the results and address any highlighted error conditions before performing the upgrade.

15. Perform the upgrade:

```
# foreman-maintain upgrade run --target-version 6.4
```

16. If using a BASH shell, after a successful or failed upgrade, enter:

```
# hash -d foreman-maintain service 2> /dev/null
```



WARNING

If you run the command from a directory containing a **config** subdirectory, you will encounter the following error:

```
ERROR: Scenario (config/satellite.yaml) was not
found, can not continue.
```

In such a case, change directory, for example to the **root** user's home directory, and run the command again.

17. Check and restore any changes required to the DNS and DHCP configuration files using the backups that you make.
18. If you make changes in the previous step, restart Satellite services.

```
# foreman-maintain service restart
```

19. If you have the OpenSCAP plug-in installed, but do not have the default OpenSCAP content available, enter the following command.

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

20. In the Satellite web UI, go to **Configure > Discovery Rules** and associate selected organizations and locations with discovery rules.

3.2. UPGRADING CAPSULE SERVERS

This section describes how to upgrade Capsule Servers from 6.3 to 6.4.

Before You Begin

- You must upgrade Satellite Server before you can upgrade any Capsule Servers.
- You must upgrade Capsule Server to Puppet 4. For more information, see the [Upgrading Puppet](#) chapter in the *Red Hat Satellite 6.3 Upgrading and Updating Guide*.
- Ensure the Red Hat Satellite Capsule 6.4 repository is enabled in Satellite Server and synchronized.
- If you use Content Views to control updates to a Capsule Server's base operating system, or for the Capsule Server repository, you must publish updated versions of those Content Views.
- Ensure the Capsule's base system is registered to the newly upgraded Satellite Server.
- Ensure the Capsule has the correct organization and location settings in the newly upgraded Satellite Server.
- Review and update your firewall configuration prior to upgrading your Capsule Server. For more information, see [Ports and Firewalls Requirements](#) in *Installing Capsule Server*.



WARNING

If you implemented custom certificates, you must retain the content of both the `/root/ssl-build` directory and the directory in which you created any source files associated with your custom certificates.

Failure to retain these files during an upgrade causes the upgrade to fail. If these files have been deleted, they must be restored from a backup in order for the upgrade to proceed.

Upgrading Capsule Servers

1. Create a backup.
 - On a virtual machine, take a snapshot.
 - On a physical machine, create a backup.

For information on backups, see [Backing Up and Restoring Satellite Server and Capsule Server](#) in the *Administering Red Hat Satellite 6.3* guide.

2. Back up the DNS and DHCP configuration files `/etc/zones.conf` and `/etc/dhcp/dhcpd.conf` as the installer only supports one domain or subnet, and therefore restoring changes from these backups might be required.
3. If you have made manual edits to DNS or DHCP configuration files and do not want to overwrite the changes, enter the following command:

```
# satellite-installer --foreman-proxy-dns-managed=false \
--foreman-proxy-dhcp-managed=false
```

4. Disable the repository for Red Hat Satellite 6.3.

```
# subscription-manager repos \
--disable rhel-7-server-satellite-capsule-6.3-rpms
```

5. Disable the Puppet 4 repository:

```
# subscription-manager repos \
--disable=rhel-7-server-satellite-capsule-6.3-puppet4-rpms
```

6. Enable the new repositories.

The Red Hat Software Collections repository provides a later version of Ruby required by some Red Hat Satellite features, including the Remote Execution feature. The Satellite tools repository provides **gofer** and **katello-agent** which provide communication services for managing Errata.

- Enter the following command:

```
# subscription-manager repos \
--enable rhel-7-server-satellite-capsule-6.4-rpms \
--enable rhel-server-rhsc1-7-rpms \
--enable rhel-7-server-satellite-tools-6.4-rpms \
--enable rhel-7-server-satellite-maintenance-6-rpms \
--enable rhel-7-server-ansible-2.6-rpms
```

7. In the Satellite web UI, go to **Hosts > Discovered hosts**. If there are discovered hosts available, power off the hosts and then delete all entries under the **Discovered hosts** page. Select all other organizations in turn using the organization setting menu and repeat this action as required. Reboot these hosts after the upgrade has completed.

8. Clear the repository cache.

```
# yum clean all
```

9. Stop Satellite services.

```
# katello-service stop
```

10. Update all packages.

```
# yum update
```


-
- 11. If you plan to use Capsule Server as a proxy for discovered hosts, install the Discovery plug-in.

```
# yum install rubygem-smart_proxy_discovery.noarch
```

- 12. On the Capsule Server, verify that the **foreman_url** setting is correct.

```
# grep foreman_url /etc/foreman-proxy/settings.yml
```

The fully qualified domain name of the Satellite Server should display.

- 13. Perform the upgrade by running the installer script with the **--upgrade** option:

```
# satellite-installer --scenario capsule --upgrade
```



WARNING

If you run the command from a directory containing a **config** subdirectory, you will encounter the following error:

```
ERROR: Scenario (config/capsule.yaml) was not found,
can not continue.
```

In such a case, change directory, for example to the **root** user's home directory, and run the command again.

- 14. Check and restore any changes required to the DNS and DHCP configuration files using the backups made earlier.
- 15. Upgrade the foreman-discovery package on Satellite Server and turn on the hosts that were shut down prior to the upgrade.

3.3. UPGRADING SATELLITE CLIENTS

Currently, the Satellite 6.3 version of **katello-agent** and other client libraries in the Satellite Tools repository are not formally tested or supported against Satellite 6.4.

Upgrade all clients to the new version of **katello-agent** as soon as possible so that your clients are fully compatible with Satellite Server. This requires changing the Satellite Tools repository from 6.3 to 6.4, which can be done manually or by installing the **satellite-6.4-tools-upgrade** package. This package only contains a post installation script to change the Satellite Tools repository version.

Prerequisites

- You must have upgraded Satellite Server.
- You must have enabled the new Satellite Tools repositories on the Satellite.

- You must have synchronized the new repositories in the Satellite.
- If you have not previously installed **katello-agent** on your clients, use the manual method.



WARNING

If you implemented custom certificates, you must retain the content of both the **/root/ssl-build** directory and the directory in which you created any source files associated with your custom certificates.

Failure to retain these files during an upgrade causes the upgrade to fail. If these files have been deleted, they must be restored from a backup in order for the upgrade to proceed.

Upgrade Satellite Clients Using the Bulk Repository Set UI:

1. In the Satellite web UI, navigate to **Hosts > Content Hosts** and select the Content Hosts that you want to upgrade.
2. From the **Select Action** list, select **Manage Repository Sets**.
3. From the **Repository Sets Management** list, select the **Red Hat Satellite Tools 6.3** check box.
4. From the **Select Action** list, select **Override to Disabled**, and click **Done**.
5. When the process completes, on the same set of hosts from the previous steps, select **Manage Repository Sets**.
6. From the **Repository Sets Management** list, select the **Red Hat Satellite Tools 6.4** check box.
7. From the **Select Action** list, select **Override to Enabled**, and click **Done**.
8. When the process completes, on the same set of hosts from the previous steps, select **Manage Repository Sets**.
9. From the **Select Action** list, select **Manage Packages**.
10. In the **Package** search field, enter **katello-agent**.
11. From the **Update** list, select your preferred update method.
12. Ensure that the update is complete, and then click **Done**.

Upgrade Satellite Clients Manually

1. Log into the client system.
2. Disable the repositories for the previous version of Satellite.

```
# subscription-manager repos \
--disable rhel-7-server-satellite-tools-6.3-rpms
```

3. Enable the Satellite tools repository for this version of Satellite.

```
# subscription-manager repos \
--enable=rhel-7-server-satellite-tools-6.4-rpms
```

4. Upgrade the following **Katello**, **Pulp**, and **qpidd** packages.

```
# yum upgrade katello-agent katello-host-tools katello-host-tools-
fact-plugin pulp-rpm-handlers qpidd-proton-c
```

5. Restart **goferd**.

```
# systemctl restart goferd
```

3.4. POST-UPGRADE TASKS

Some of the procedures in this section are optional. You can choose to perform only those procedures that are relevant to your installation.

If you use the PXE-based discovery process, then you must complete the discovery upgrade procedure on Satellite and on any Capsule Server with hosts that you want to be listed in Satellite on the **Hosts > Discovered hosts** page.

If you have scripts for backing up and restoring Satellite or Capsule, ensure that you update the scripts to replace **katello-backup** and **katello-restore** with **foreman-maintain backup** and **foreman-maintain restore**.

3.4.1. Upgrading Discovery

This section describes updating the PXELinux template and the boot image passed to hosts that use PXE booting to register themselves with Satellite Server.

From Satellite 6.4, provisioning templates now have a separate association with a subnet, and do not default to using the TFTP Capsule for that subnet. If you create subnets after the upgrade, you must specifically enable the Satellite or a Capsule to provide a proxy service for discovery templates and then configure all subnets with discovered hosts to use a specific *template Capsule*.

During the upgrade, for every subnet with a TFTP proxy enabled, the template Capsule is set to be the same as the TFTP Capsule. After the upgrade, check all subnets to verify this was set correctly.

These procedures are not required if you do not use PXE booting of hosts to enable Satellite to discover new hosts.

3.4.1.1. Upgrading Discovery on Satellite Server

1. Update the Discovery template in the Satellite web UI:
 - a. Navigate to **Hosts > Provisioning templates**.
 - b. On the **PXELinux global default** line, click **Clone**.
 - c. Enter a new name for the template in the **Name** field, for example **ACME PXE global default**.

- d. In the template editor field, change the line **ONTIMEOUT local** to **ONTIMEOUT discovery** and click **Submit**.
 - e. Navigate to **Administer > Settings**.
 - f. Locate **Global default PXELinux template** and click on its **Value**.
 - g. Select the name of the newly created template from the menu and click the tick button.
 - h. Navigate to **Hosts > Provisioning templates**.
 - i. Click **Build PXE Default**, then click **OK**.
2. In the Satellite web UI, go to **Configure > Discovery Rules** and associate selected organizations and locations with discovery rules.

3.4.1.2. Upgrading Discovery on Capsule Servers

1. Verify that the Foreman Discovery package is current on Satellite Server.

```
# yum upgrade tfm-rubygem-foreman_discovery
```

2. If an update occurred in the previous step, restart Satellite services.

```
# katello-service restart
```

3. Upgrade the Discovery image on the Satellite Capsule that is either connected to the provisioning network with discovered hosts or provides TFTP services for discovered hosts.

```
# yum upgrade foreman-discovery-image
```

4. On the same instance, install the package which provides the Proxy service, and then restart **foreman-proxy** service.

```
# yum install rubygem-smart_proxy_discovery  
# service foreman-proxy restart
```

5. In the Satellite web UI, go to **Infrastructure > Capsules** and verify that the relevant Capsule lists **Discovery** in the features column. Select **Refresh** from the **Actions** drop-down menu if necessary.
6. Go to **Infrastructure > Subnets** and for each subnet on which you want to use discovery:
 - a. Click the subnet name.
 - b. On the **Capsules** tab, ensure the **Discovery Capsule** is set to a Capsule you configured above.

3.4.1.3. Verifying Subnets have a Template Capsule

Ensure all subnets with discovered hosts have a template Capsule:

1. In the Satellite web UI, navigate to **Infrastructure > Subnets**.

2. Select the subnet you want to check.
3. On the **Capsules** tab, ensure a **Template Capsule** has been set for this subnet.

For more information about configuring subnets with template Capsules, see [Configuring Discovery Subnets](#) in the *Red Hat Satellite Managing Hosts* guide.

3.4.2. Upgrading virt-who

If virt-who is installed on Satellite Server or a Capsule Server, it will be upgraded when they are upgraded. No further action is required. If virt-who is installed elsewhere, it must be upgraded manually.

Before You Begin

If virt-who is installed on a host registered to Satellite Server or a Capsule Server, first upgrade the host to the latest packages available in the Satellite Tools repository. For information about upgrading hosts, see [Section 3.3, “Upgrading Satellite Clients”](#).

Upgrade virt-who Manually

1. Upgrade virt-who.

```
# yum upgrade virt-who
```

2. Restart the virt-who service so the new version is activated.

```
# systemctl restart virt-who.service
```

3.4.3. Removing the Previous Version of the Satellite Tools Repository

After completing the upgrade to Satellite 6.4, the Red Hat Satellite Tools 6.3 repository can be removed from Content Views and then disabled.

Disable Version 6.3 of the Satellite Tools Repository:

1. In the Satellite web UI, navigate to **Content > Red Hat Repositories**.
2. In the **Enabled Repositories** area, locate **Red Hat Satellite Tools 6.3 for RHEL 7 Server RPMs x86_64**.
3. Click the **Disable** icon to the right.

If the repository is still contained in a Content View then you cannot disable it. Packages from a disabled repository are removed automatically by a scheduled task.

CHAPTER 4. UPGRADING PUPPET

Red Hat Satellite 6.4 only supports Puppet 5. The migration path for Puppet 5 supports only Puppet 4 to Puppet 5 migrations. Therefore, if you are using a version of Puppet prior to version 4, you must upgrade to Puppet 4 prior to upgrading to Satellite 6.4. Satellite 6.3 includes an upgrade path to Puppet 4, for more information, see the [Upgrading Puppet](#) section in the *Satellite 6.3 Upgrading and Updating Red Hat Satellite* guide.

During the Satellite upgrade process, Puppet is upgraded from Puppet 4 to Puppet 5. It is not a separate process. This chapter describes the process of preparing for the upgrade from Puppet 4 to Puppet 5.

4.1. UPGRADE PATH

Before upgrading to Satellite 6.4, you must first review your Puppet modules, and make changes, to ensure they are compatible with Puppet 5. For information on the changes in Puppet 5, see the [Puppet 5.0 Release Notes](#) on the Puppet website.

Red Hat Satellite and Capsules do not have to be upgraded at the same time. You have the flexibility to upgrade Capsules depending on your progress with testing and upgrading Puppet modules. An alternative is to install new Capsules and move hosts to them after testing. You can use Satellite web UI, the Hammer CLI, or the **bootstrap** script to move hosts from a Puppet 4 Capsule to a Puppet 5 Capsule.

Cloning an existing Capsule to use for final testing of upgraded Puppet modules is also recommended. Cloning an existing Capsule can take a long time depending on the size of the repositories.

Using Parameters to Determine Puppet Versions

By default, Satellite 6.4 sets the global parameter **enable-puppet5**. Provisioning templates, such as Kickstart files, use **enable-puppet5** to provision new hosts to use the Puppet 5 agent if available. However, if the Puppet 4 agent is in a Content View attached to a host, this will be used in place of Puppet 5. If you want hosts to use Puppet 5, you must update Content Views associated with the hosts to contain the Puppet 5 agent.

If you must provision new hosts to use Puppet 4, you can override the global parameter at the host group level.

1. In the web UI, navigate to **Configure > Host Groups** and select the name of the host group you want to configure.
2. Click the **Parameters** tab, and in the **Host Group Parameters** area, click **Add Parameter**.
3. In the **Name** field, enter **enable-puppet4** and in the **Value** field, enter **true**.
4. Click **Submit** to save the changes.

If you set the parameter **enable-puppet4** in Satellite 6.3, it can now be safely removed.

1. In the web UI, navigate to **Configure > Host Groups** and select the name of the host group you want to configure.
2. Click the **Parameters** tab, and in the **Host Group Parameters** area, locate **enable-puppet4**.
3. In the **Actions** column, click **Remove**.
4. Click **Submit** to save the changes.

You can also use a hammer command to remove it. For example:

```
# hammer hostgroup delete-parameter --hostgroup my_hostgroup --name
enable-puppet4
```

4.2. TESTING PUPPET MODULES

Use the [Puppet 5.0 Release Notes](#) on the Puppet website and the following guidelines to test your Puppet modules before you begin upgrading:

- Test Puppet modules first on a workstation, separate from Satellite.
- Check the command syntax using the **puppet parser validate** command.
- Check for capital letters in the name of a class reference because this is no longer supported. Change class names to lower case; for example, change **Class[MyClass]** to **Class[myclass]**.
- If you have custom version comparison code in Ruby, note that the **Range#intersection** code to handle SemVer range matches has been removed.
- If you have custom Ruby code that makes use of the deprecated string formatting methods in the **TypeCalculator** class, change your code to use the replacement **TypeFormatter** class because the string formatting methods in the **TypeCalculator** have been removed. For more information, see the [Type mismatch describer should expand aliases](#) issue on the Puppet website.
- Previously, if a class was defined more than once their bodies were merged. A warning or error was displayed depending on the status of the **--strict** flag. Now, an error is always displayed except for the top scope class indicated by **' '** (also known as **'main'**).

4.3. PREPARING FOR THE UPGRADE

Use this section to prepare Hosts for the upgrade to Puppet 5. Satellite Server and Capsule Server are upgraded to Puppet 5 as part of their respective upgrade procedures.

Before You Begin

- Ensure you complete the upgrade to Satellite 6.4 for Satellite Server.
- Ensure you complete the upgrade for the Capsule Server with the hosts attached that you want to upgrade.
- Ensure you have reviewed your Puppet modules and upgraded them where required for Puppet 5.
- Ensure the associated Content Views have been updated to include Puppet 5 agent.
- Ensure there is no **enable-puppet4** parameter set for the host or host group to override the global **enable-puppet5** parameter.

Preparing a Host for Puppet 5

1. Disable the Puppet 4 repository:

- The Puppet 4 repositories take the following form for desktop, server, and workstation:

```
rhel-X-platform-satellite-tools-6.3-puppet4-rpms
```

Where *X* is the major release version, and *platform* is **desktop**, **server**, or **workstation**.

- The Puppet 4 repositories take the following form for alternative architectures:

```
rhel-X-for-architecture-satellite-tools-6.3-puppet4-rpms
```

Where *X* is the major release version, and *architecture* is for example **arm**, **power**, **hpc-node**, or **system-z**. Some repositories are for subscriptions with extended support. For example, Extended Update Support (EUS) is indicated by **eus** after the architecture.

For more information about types of extended support, see the Red Hat Knowledgebase article [Red Hat Enterprise Linux - Top Support Policies](#).

1. On the content host, search for the Puppet 4 RPM repository:

```
# subscription-manager repos --list-enabled |grep puppet4
```

2. Disable the Puppet 4 repository you find on the host, for example:

```
# subscription-manager repos \
--disable=rhel-7-server-satellite-tools-6.3-puppet4-rpms
```

3. Update all packages:

```
# yum update
```


CHAPTER 5. UPDATING SATELLITE SERVER, CAPSULE SERVER, AND CONTENT HOSTS

Use this chapter to update your existing Red Hat Satellite environment to a new minor version of Red Hat Satellite, for example, from 6.3.0 to 6.3.1. For more information, see [Chapter 5, Updating Satellite Server, Capsule Server, and Content Hosts](#).

This chapter includes update procedures for Satellite Server, Capsule Server, and Content Hosts.

Updating Between Minor Versions of Satellite

Updating is the process of migrating Satellite Server, Capsule Server, and Content Hosts to a new minor version. Updates typically patch security vulnerabilities and correct minor issues discovered after code is released. Generally speaking, updates require little time and are non-disruptive to your operating environment.

Before updating, check the [Red Hat Satellite Release Notes](#) for potential conflicts and backup your Satellite Server and all Capsule Servers. For more information, see [Backing Up and Restoring Satellite Server and Capsule Server](#) in the *Administering Red Hat Satellite* guide.

Follow these procedures to update between minor versions, for example, from 6.3.0 to 6.3.1.

5.1. UPDATING SATELLITE SERVER

Prerequisites

- Ensure that you have synchronized Satellite Server repositories for Satellite, Capsule, and Satellite Tools.
- Ensure each external Capsule and Content Host can be updated by promoting the updated repositories to all relevant Content Views.



WARNING

If you customize configuration files, manually or use a tool such as Hiera, these customizations are overwritten when the installation script runs during upgrading or updating. You can use the `--noop` option with the `satellite-installer` script to test for changes. For more information, see the Red Hat Knowledgebase solution [How to use the noop option to check for changes in Satellite config files during an upgrade](#).

Updating Satellite Server to the Next Minor Version

To Update Satellite Server:

1. Ensure the Satellite Maintenance repository is enabled:

```
# subscription-manager repos --enable \
  rhel-7-server-satellite-maintenance-6-rpms
```

2. Ensure **foreman-maintain** is installed and up to date:

```
# yum install rubygem-foreman_maintain
```

3. Check the available versions to confirm the next minor version is listed:

```
# foreman-maintain upgrade list-versions
```

4. Use the health check option to determine if the system is ready for upgrade. On first use of this command, **foreman-maintain** prompts you to enter the hammer admin user credentials and saves them in the `/etc/foreman-maintain/foreman-maintain-hammer.yml` file.

```
# foreman-maintain upgrade check --target-version 6.4.z
```

Review the results and address any highlighted error conditions before performing the upgrade.

5. Because of the lengthy update time, use a utility such as **screen** to suspend and reattach a communication session. You can then check the upgrade progress without staying connected to the command shell continuously. For more information about using the screen command, see [How do I use the screen command?](#) article in the *Red Hat Knowledge Base*.

If you lose connection to the command shell where the upgrade command is running, you can see the logged messages in the `/var/log/foreman-installer/satellite.log` file to check if the process completed successfully.

6. Perform the upgrade:

```
# foreman-maintain upgrade run --target-version 6.4.z
```

7. Check to see if any of the kernel packages were updated since you last rebooted the system:

```
# rpm -qa --last | grep kernel
```

8. If a kernel update occurred since you last rebooted the system, reboot the system:

```
# reboot
```

5.2. UPDATING CAPSULE SERVER

Updating Capsule Servers to the Next Minor Version

To Update a Capsule Server:

1. Check that only the correct repositories are enabled:

- a. List the enabled repositories:

```
# subscription-manager repos --list-enabled
```

- b. Ensure that you only have the following repositories enabled:

```
rhel-7-server-rpms
rhel-7-server-satellite-capsule-6.4-rpms
```

```
rhel-server-rhsc1-7-rpms
rhel-7-server-satellite-tools-6.4-rpms
rhel-7-server-satellite-maintenance-6-rpms
```

For more information about disabling and enabling repositories, see [Configuring Repositories](#) in *Installing Capsule Server*. The **rhel-7-server-satellite-tools-6.4-rpms** repository provides Katello Agent. For more information, see [Installing the katello Agent](#) in *Installing Capsule Server*. The Red Hat Software Collections repository is optional, it is required for the Remote Execution feature.

2. Stop Satellite services:

```
# katello-service stop
```

3. Update all packages:

```
# yum update
```

If a kernel update occurs, make a note to reboot **after** the upgrade is complete. Do **not** reboot at this point.

4. Perform the update by running the installer script with the **--upgrade** option.

```
# satellite-installer --scenario capsule --upgrade
```

5. If a kernel update occurred during the **yum update** step, reboot the system:

```
# reboot
```

6. If you did not reboot the system in the previous step, restart **goferd**:

```
# systemctl restart goferd
```

5.3. UPDATING CONTENT HOSTS

Updating Content Hosts to the Next Minor Version

To Update a Content Host, enter the following commands:

1. Update all packages:

```
# yum update
```

2. If a kernel update occurs, reboot the system:

```
# reboot
```

3. If you did not reboot the system in the previous step, restart **goferd**:

```
# systemctl restart goferd
```

