Red Hat Satellite 6.2-Beta Installation Guide

Installing Red Hat Satellite Server and Capsule Server

Red Hat Satellite Documentation Team

Red Hat Satellite 6.2-Beta Installation Guide

Installing Red Hat Satellite Server and Capsule Server

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

Legal Notice

Abstract

This guide describes how to install Red Hat Satellite Server and Capsule Server, perform initial configuration, and configure external services.

Table of Contents

CHAPTER 1. WHAT SATELLITE SERVER AND CAPSULE SERVER DO

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

Red Hat Satellite Capsule Server mirrors content from Red Hat Satellite Server to facilitate content federation across various geographical locations. Host systems can pull content from the Capsule Server and not from the central Satellite Server. The Capsule Server also provides localized services such as Puppet Master, DHCP, DNS, or TFTP. Capsule Servers assist you in scaling your Satellite environment as the number of your managed systems increases.

Capsule Servers decrease the load on the central server, increase redundancy, and reduce bandwidth usage.

CHAPTER 2. PREPARING YOUR ENVIRONMENT FOR INSTALLATION

Before you install Satellite Server or Capsule Server, you should ensure that your environment meets the requirements for installation.

> **Note**
>
> The Red Hat Satellite server and Capsule server versions must match. For example, a Satellite 6.1 server cannot run a 6.2 Capsule server and a Satellite 6.2 server cannot run a 6.1 Capsule server. Mismatching Satellite server and Capsule server versions results in the Capsule server failing silently.

If you have a large number of content hosts, see Large Deployment Considerations to ensure that your environment is set up appropriately.

For more information on scaling your Capsule Servers, see Capsule Server Scalability Considerations.

2.1. STORAGE REQUIREMENTS AND RECOMMENDATIONS

Ensure that your environment meets the minimum requirements before installing Satellite Server or Capsule Server.

Packages that are duplicated in different repositories are only stored once on the disk. Additional repositories containing duplicate packages will require less additional storage. The bulk of storage resides on the `/var/lib/mongodb` and `/var/lib/pulp` directories. These end points are not manually configurable. Make sure that storage is available on the `/var` file system to prevent storage issues.

Storage Requirements

The following table details recommended storage requirements for specific directories. These values are based on expected use case scenarios and may vary according to individual environments.

Table 2.1. Minimum Storage Requirements for Installation

Directory	Installation Size	Runtime Size with Red Hat Enterprise Linux 5/6/7 synchronized	Considerations

Directory	Installation Size	Runtime Size with Red Hat Enterprise Linux 5/6/7 synchronized	Considerations
/var/lib/pulp	1 MB	200 GB	- Will continue to grow as content is added to Satellite Server. Plan for expansion over time. - Symbolic links cannot be used.
/var/lib/mongodb	3.5 GB	25 GB	- Will continue to grow as content is added to Satellite Server. Plan for expansion over time. - Symbolic links cannot be used. - For improved performance, use solid state drives (SSD) rather than hard disk drives (HDD).
/var/log	10 MB	250 MB	None
/var/lib/pgsql	100 MB	2 GB	A minimum of 2 GB of available storage in **`/var/lib/pgsql`** with the ability to grow the partition containing this directory as data storage requirements grow.
/usr	3 GB	Not Applicable	None

Directory	Installation Size	Runtime Size with Red Hat Enterprise Linux 5/6/7 synchronized	Considerations
/opt/	500 MB (Connected Installations)	Not Applicable	Software collections are installed into the **`/opt/rh`** and **`/opt/theforman`** directories. Write and execute permissions by root are required for installation into to the **`/opt`** directory.
/opt/	2 GB (Disconnected Installations)	Not Applicable	- Software collections are installed into the **`/opt/rh`** and **`/opt/theforman`** directories. Write and execute permissions by root are required for installation into to the **`/opt`** directory. - A copy of the repositories used for installation is stored in this directory.

Storage Recommendations

- Because most Satellite Server data is stored within the /var directory, it is strongly recommended to mount /var on LVM storage, enabling the system to scale.

- Red Hat recommends the usage of high-bandwidth, low-latency storage for the /var/lib/pulp & /var/lib/mongodb file systems. As Red Hat Satellite has many operations that are IO intensive, usage of high latency, low-bandwidth storage could potentially have issues with performance degradation. Additionally, usage of NFS-backed storage may require additional configuration.

- The XFS file system is recommended for Red Hat Satellite 6. XFS is the default file system in Red Hat Enterprise Linux 7, which makes it the preferable base operating system. If you intend to use Red Hat Enterprise Linux 6 instead, contact your account team to learn about enabling XFS on this system. Also consider that long term support for Satellite 6 on Red Hat Enterprise Linux 6 has a shorter lifespan which may necessitate a migration from version 6 to 7 in the future. Red Hat Enterprise Linux 7 is highly recommended for new installations.

2.2. SUPPORTED OPERATING SYSTEMS

You can install the operating system from disc, local ISO image, kickstart, or any other method that Red Hat supports. Red Hat Satellite is only supported on the latest version of Red Hat Enterprise Linux 6 Server or 7 Server that is available when Satellite 6.2 is released. Previous versions of Red Hat Enterprise Linux including EUS or z-stream are not supported.

Table 2.2. Minimum Red Hat Enterprise Linux Version Requirements

Satellite 6 Version	Red hat Enterprise Linux 6	Red Hat Enterprise Linux 7	Notes
6.2-Beta	6.7	7.2	Not applicable
6.2	6.7	7.2	Not applicable

Red Hat Satellite Server requires Red Hat Enterprise Linux installations with the @Base package group with no other package-set modifications, and without third-party configurations or software that is not directly necessary for the direct operation of the server. This restriction includes hardening or other non-Red Hat security software. If such software is required in your infrastructure, install and verify a complete working Satellite Server first, then create a backup of the system before adding any non-Red Hat software.

It is recommended that the Satellite Server be a freshly provisioned system that serves no other function.

The following requirements apply to the networked base system:

- 64-bit architecture
- The latest version of Red Hat Enterprise Linux 6 Server or 7 Server
- A minimum of 2 CPU cores, 4 CPU cores are recommended
- A minimum of 12 GB memory, 16 GB of memory for each instance of Satellite. A minimum of 4 GB of swap space is recommended.
- A unique hostname, which can contain lower-case letters, numbers, dots (.) and hyphens (-)
- A current Red Hat Satellite subscription
- Administrative user (root) access
- Full forward and reverse DNS resolution using a fully-qualified domain name

If any of the following exist on the system, they must be removed before installation:

- Java virtual machines
- Puppet RPM files
- Additional yum repositories other than those explicitly required in this guide for installation

2.3. SUPPORTED BROWSERS

The following web browsers are fully supported:

- Firefox versions 22 and later
- Chrome versions 28 and later

The following web browsers are partially supported. The Satellite web UI interface will function correctly but certain design elements may not align as expected:

- Firefox version 38
- Chrome versions 27
- Internet Explorer versions 10 and 11

Note

The web UI and command-line interface for Satellite Server supports English, Portuguese, Simplified Chinese, Traditional Chinese, Korean, Japanese, Italian, Spanish, Russian, French, and German.

2.4. PORTS AND FIREWALLS REQUIREMENTS

Specific network ports must be open and free on the base operating system before continuing with the installation. Any managed host that is directly connected to the Satellite Server is a Client in this context. This includes the base system on which a Capsule Server is running.

Required ports can change based on your configuration.

Table 2.3. Ports for Browser-based User Interface Access to Satellite

Port	Protocol	Service	Required For
443	TCP	HTTPS	For Browser-based UI access to Satellite
80	TCP	HTTP	To enable redirection to HTTPS for web UI access to Satellite (Optional)

Table 2.4. Ports for Client to Satellite Communication

Port	Protocol	Service	Required For

Port	Protocol	Service	Required For
80	TCP	HTTP	Anaconda, yum, for obtaining Katello certificates, templates, and for downloading iPXE firmware
443	TCP	HTTPS	Subscription Management Services, yum, Telemetry Services, and for connection to the Katello Agent
5647	TCP	amqp	The Katello Agent to communicate with the Satellite’s Qpid dispatch router
8140	TCP	HTTPS	Puppet agent to Puppet master connections
9090	TCP	HTTPS	Sending SCAP reports to the Smart Proxy in the integrated Capsule and for the discovery image during provisioning

Table 2.5. Optional Network Ports

Port	Protocol	Service	Required For
53	TCP and UDP	DNS	Queries to the Satellite’s integrated DNS service
67, 68	UDP	DHCP	For Client provisioning from the integrated Capsule

Port	Protocol	Service	Required For
69	UDP	TFTP	Downloading PXE boot image files from the integrated Capsule
7911	TCP	DHCP	Capsule originated, for orchestration of DHCP records (local or external) If DHCP is provided by an external service, you must open the port on the external server.
5000	TCP	HTTP	Satellite originated, for compute resources in OpenStack or for running Docker containers
22, 16514	TCP	SSH, SSL/TLS	Satellite originated, for compute resources in libvirt
389, 636	TCP	LDAP, LDAPS	Satellite originated, for LDAP and secured LDAP authentication sources
5910 to 5930	TCP	SSL/TLS	Satellite originated, for NoVNC console in web UI to hypervisors

2.5. ENABLING CONNECTIONS FROM A CLIENT TO SATELLITE SERVER

You must configure the firewall on Satellite to enable incoming connections from a Client and to make these rules persistent during reboots. For more information on the ports used, see Ports and Firewalls Requirements.

Configuring the Firewall on Red Hat Enterprise Linux 6

1. Start and enable the iptables service on Satellite Server.

```
# iptables -A INPUT -m state --state NEW -p udp --dport 53 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 53 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p udp --dport 67 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p udp --dport 68 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p udp --dport 69 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 80 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 443 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 5647 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 8140 -j
ACCEPT \
&& iptables-save > /etc/sysconfig/iptables
```

2. Verify that the iptables service is started and enabled.

```
# service iptables start
# chkconfig iptables on
```

Configuring the Firewall on Red Hat Enterprise Linux 7

1. Add the RH-Satellite-6 service to the default zone on Satellite Server.

```
# firewall-cmd --add-service=RH-Satellite-6
```

2. Repeat the command adding the **`--permanent`** option to make the setting persistent.

```
# firewall-cmd --permanent --add-service=RH-Satellite-6
```

2.6. ENABLING CONNECTIONS FROM CAPSULE SERVER TO SATELLITE SERVER

You can enable incoming connections from Capsule Server to Satellite Server and make these rules persistent during reboots. If you do not use an external Capsule Server, you do not need to enable this connection.

For more information on the ports used, see Ports and Firewalls Requirements.

Ports 443 and 5647 are opened when you enable the connection from the client to Satellite Server, since Capsule Server is also a client of Satellite Server.

Configuring the Firewall on Red Hat Enterprise Linux 6

1. Configure iptables service.

```
# iptables -A INPUT -m state --state NEW -p tcp --dport 5646 -j
ACCEPT \
&& iptables-save > /etc/sysconfig/iptables
```

2. Start iptables service.

```
# service iptables restart
# chkconfig iptables on
```

Configuring the Firewall on Red Hat Enterprise Linux 7

1. Configure the firewall on Satellite Server.

```
# firewall-cmd --add-port="5646/tcp"
```

2. Repeat the command adding the **`--permanent`** option to make the settings persistent.

```
# firewall-cmd --permanent --add-port="5646/tcp"
```

2.7. ENABLING CONNECTIONS FROM SATELLITE SERVER AND CLIENTS TO A CAPSULE SERVER

You can enable incoming connections from Satellite Server and clients to Capsule Server and make these rules persistent during reboots. If you do not use an external Capsule Server, you do not need to enable this connection.

For more information on the ports used, see Ports and Firewalls Requirements.

Configuring the Firewall on Red Hat Enterprise Linux 6

1. Configure iptables service.

```
# iptables -A INPUT -m state --state NEW -p udp --dport 53 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 53 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p udp --dport 67 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p udp --dport 68 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p udp --dport 69 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 80 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 443 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 5647 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 8000 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 8140 -j
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 8443 -j
```

```
ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 9090 -j
ACCEPT \
&& iptables-save > /etc/sysconfig/iptables
```

2. Start iptables service.

```
# service iptables restart
# chkconfig iptables on
```

Configuring the Firewall on Red Hat Enterprise Linux 7

1. Configure the firewall on Capsule Server.

```
# firewall-cmd --add-port="53/udp" --add-port="53/tcp" \
 --add-port="67/udp" --add-port="68/udp" \
 --add-port="69/udp" --add-port="80/tcp" \
 --add-port="443/tcp" --add-port="5647/tcp" \
 --add-port="8000/tcp" --add-port="8140/tcp" \
 --add-port="8443/tcp" --add-port="9090/tcp"
```

2. Repeat the command adding the **`--permanent`** option to make the settings persistent.

```
# firewall-cmd --permanent --add-port="53/udp" --add-
port="53/tcp" \
 --add-port="67/udp" --add-port="68/udp" \
 --add-port="69/udp" --add-port="80/tcp" \
 --add-port="443/tcp" --add-port="5647/tcp" \
 --add-port="8000/tcp" --add-port="8140/tcp" \
 --add-port="8443/tcp" --add-port="9090/tcp"
```

2.8. VERIFYING DNS RESOLUTION

Verifying the full forward and reverse DNS resolution using a fully-qualified domain name enables you to prevent issues while installing Satellite.

1. Ensure that the hostname and localhost resolve correctly.

```
# ping -c1 localhost
# ping -c1 `hostname -f` # my_system.domain.com
```

2.9. SYNCHRONIZING TIME

Before installation, you must start and enable a time synchronizer on the host operating system to minimize the effects of time drift. If you do not set up ntp, installation and certificate setup could fail.

For more information, see the Red Hat Enterprise Linux 7 System Administrators Guide or the Red Hat Enterprise Linux 6 Deployment Guide.

Synchronizing Time on Red Hat Enterprise Linux 6

1. Install ntp.

   ```
   # yum install ntp
   ```

2. Verify that your NTP server is available.

   ```
   # ntpdate -q server_address
   ```

3. Set the system time.

   ```
   ntpdate server_address...
   ```

4. Start and enable the ntpd service.

   ```
   # chkconfig ntpd on
   ```

Synchronizing Time on Red Hat Enterprise Linux 7

1. Install chronyd.

   ```
   # yum install chrony
   ```

2. Start and enable the chrony service.

   ```
   # systemctl start chronyd
   # systemctl enable chronyd
   ```

2.10. CHANGING DEFAULT SELINUX PORTS

Red Hat Satellite 6 uses a set of predefined ports. Because Red Hat recommends that SELinux on Satellite 6 systems be set to permissive or enforcing, if you need to change the port for any service, you also need to change the associated SELinux port type to allow access to the resources. You only need to change these ports if you use non-standard ports.

For example, if you change the web UI ports (HTTP/HTTPS) to 8018/8019, you need to add these port numbers to the httpd_port_t SELinux port type.

This change is also required for target ports. For example, when Satellite 6 connects to an external source, like Red Hat Enterprise Virtualization Manager or OpenStack.

You only need to make changes to default port assignments once. Updating or upgrading Satellite has no effect on these assignments. Updating only adds default SELinux ports if no assignments exist.

Before You Begin

- SELinux must be enabled and running in permissive or enforcing mode before installing Satellite. For more information, see the *Red Hat Enterprise 6 Security-Enhanced Linux User Guide* or the *Red Hat Enterprise Linux 7 SELinux User's and Administrator's Guide*.

Changing default ports to user-specified ports

1. To change the port from the default port to a user-specified port, execute the commands using values that are relevant to your environment. These examples use port 99999 for demonstration purposes.

Default Port	SELinux Command
80, 443, 8443	semanage port -a -t http_port_t -p tcp 99999
8080	semanage port -a -t http_cache_port_t -p tcp 99999
8140	semanage port -a -t puppet_port_t -p tcp 99999
9090	semanage port -a -t websm_port_t -p tcp 99999
69	semanage port -a -t tftp_port_t -p udp 99999
53 (TCP)	semanage port -a -t dns_port_t -p tcp 99999
53 (UDP)	semanage port -a -t dns_port_t -p udp 99999
67, 68	semanage port -a -t dhcpd_port_t -p udp 99999
5671	semanage port -a -t amqp_port_t -p tcp 99999
8000	semanage port -a -t soundd_port_t -p tcp 99999
7911	semanage port -a -t dhcpd_port_t -p tcp 99999
5000 on Red Hat Enterprise Linux 6	semanage port -a -t commplex_port_t -p tcp 99999
5000 on Red Hat Enterprise Linux 7	semanage port -a -t commplex_main_port_t -p tcp 99999

Default Port	SELinux Command
22	semanage port -a -t ssh_port_t -p tcp 99999
16514 (libvirt)	semanage port -a -t virt_port_t -p tcp 99999
389, 636	semanage port -a -t ldap_port_t -p tcp 99999
5910 to 5930	semanage port -a -t vnc_port_t -p tcp 99999

2. Disassociate the previously used port number and port type.

```
# semanage port -d -t virt_port_t -p tcp 99999
```

2.11. INSTALLING THE SOS PACKAGE ON THE HOST OPERATING SYSTEM

Before you install Satellite Server, you should install the **sos** package on the host operating system. The **sos** package enables you to collect configuration and diagnostic information from a Red Hat Enterprise Linux system. You can also use it to provide the initial system analysis, which is required when opening a service request with Red Hat Technical Support. For more information on using sos, see What is a sosreport and how to create one in Red Hat Enterprise Linux 4.6 and later?.

1. Install the sos package.

   ```
   # yum install sos
   ```

CHAPTER 3. INSTALLING SATELLITE SERVER

There are two methods of installing Satellite Server, connected and disconnected. A connected installation enables you to obtain the packages necessary to install Satellite Server by installing them directly from the Red Hat Content Delivery Network (CDN). A disconnected installation enables you to download an ISO image of the packages from an external computer and copy it to the Satellite Server for installation.

For hosts that have network connectivity, Red Hat recommends installing the packages directly from the CDN. Using ISO images is only recommended for hosts in a disconnected environment because ISO images may not contain the latest updates.

To successfully install Satellite Server, you must have root access.

3.1. INSTALLING SATELLITE SERVER FROM A CONNECTED NETWORK

Installing Satellite Server from a connected network enables you to obtain packages and receive updates directly from the Red Hat Content Delivery Network.

3.1.1. Registering to Red Hat Subscription Management

Registering the host to Red Hat Subscription Management enables the host to subscribe to and consume content for any subscriptions available to the user. This includes content such as Red Hat Enterprise Linux, Red Hat Software Collections (RHSCL), and Red Hat Satellite.

1. Register your Satellite Server, using your user name and password.

   ```
   # subscription-manager register
   ```

   The command displays output similar to the following:

   ```
   # subscription-manager register
   Username: user_name
   Password:
   The system has been registered with ID: 541084ff2-44cab-4eb1-
   9fa1-7683431bcf9a
   ```

3.1.2. Identifying and Attaching the Satellite Subscription to the Host

After you have registered your host, you need to identify your Satellite subscription Pool ID. The Pool ID enables you to attach the required subscription to your host. The Satellite subscription provides access to the Satellite content, as well as Red Hat Enterprise Linux, Red Hat Software Collections (RHSCL), and Red Hat Satellite. This is the only subscription required.

1. To identify your Satellite subscription, run the following command:

   ```
   # subscription-manager list --all --available
   ```

   The outputs displays something similar to the following:

```
Subscription Name: Red Hat Satellite
Provides:          Oracle Java (for RHEL Server)
                   Red Hat Satellite 6 Beta
                   Red Hat Enterprise Linux Server
                   Red Hat Satellite
                   Red Hat Enterprise Linux Load Balancer (for
RHEL Server)
SKU:               MCT0370
Pool ID:           8a85f9874152663c0541943739717d11
Available:         3
Suggested:         1
Service Level:     Premium
Service Type:      L1-L3
Multi-Entitlement: No
Ends:              10/07/2014
System Type:       Physical
```

2. Make a note of the Pool ID so that you can attach it to your Satellite host. Your Pool ID will be different than the example provided.

3. To attach your subscription to your Satellite Server, run the following command, using your Pool ID:

```
# subscription-manager attach --pool=\
```

The outputs displays something similar to the following:

```
Successfully attached a subscription for: Red Hat Satellite
```

4. To verify that the subscriptions are successfully attached, run the following command:

```
# subscription-manager list --consumed
```

The outputs displays something similar to the following:

```
+-------------------------------------------+
   Consumed Subscriptions
+-------------------------------------------+
Subscription Name: Red Hat Satellite
Provides:          Red Hat Satellite
                   Red Hat Enterprise Linux Server
                   Red Hat Software Collections (for RHEL
Server)
                   Red Hat Satellite Beta
                   Red Hat Satellite 6 Beta
                   Red Hat Software Collections Beta (for RHEL
Server)
                   Red Hat Satellite Capsule Beta
                   Red Hat Enterprise Linux Load Balancer (for
RHEL Server)
                   Red Hat Satellite with Embedded Oracle
                   Red Hat Satellite Capsule
                   Red Hat Enterprise Linux High Availability
(for RHEL Server)
SKU:               MCT0370
```

```
Contract:          10293569
Account:           5361051
Serial:            1653856191250699363
Pool ID:           8a85f9874152663c0541943739717d11
Active:            True
Quantity Used:     1
Service Level:     Premium
Service Type:      L1-L3
Status Details:
Starts:            10/08/2013
Ends:              10/07/2014
System Type:       Physical
```

3.1.3. Configuring Repositories

1. Disable all existing repositories.

   ```
   # subscription-manager repos --disable "*"
   ```

2. Enable the Red Hat Satellite and Red Hat Enterprise Linux and Red Hat Software Collections repositories.

   Ensure the Red Hat Enterprise Linux repository matches the specific version you are using.

   a. If you are using Red Hat Enterprise Linux 6, run this command.

      ```
      # subscription-manager repos --enable=rhel-6-server-rpms \
      --enable=rhel-server-rhscl-6-rpms \ --enable=rhel-server-6-
      satellite-6-beta-rpms
      ```

   b. If you are using Red Hat Enterprise Linux 7, run this command.

      ```
      # subscription-manager repos --enable=rhel-7-server-rpms \
      --enable=rhel-server-rhscl-7-rpms \ --enable=rhel-server-7-
      satellite-6-beta-rpms
      ```

      Note

      If you are using a different version of Red Hat Enterprise Linux, change the repository based on your specific version.

3. Clear out any metadata left from any non-Red Hat yum repositories.

   ```
   # yum clean all
   ```

4. Verify that the repositories have been enabled.

   ```
   # yum repolist enabled
   ```

   The following output displays:

```
Loaded plugins: product-id, subscription-manager
repo id                                        repo name
status
!rhel-7-server-rpms/x86_64                     Red Hat
Enterprise Linux 7 Server (RPMs)
9,889
!rhel-server-7-satellite-6-beta-rpms/x86_64    Red Hat Satellite
6.2 Beta (for RHEL 7 Server) (RPMs)                         545
!rhel-server-rhscl-7-rpms/x86_64               Red Hat Software
Collections RPMs for Red Hat Enterprise Linux 7 Server     4,279
repolist: 14,713
```

3.1.4. Installing the Satellite Server Packages

You must install the Satellite Server packages and then perform the initial configuration of Satellite Server, including configuring customer server certificates, setting your user name, password, and the default organization and location.

1. Install the installation package.

```
# yum install satellite
```

2. Go to Performing the Initial Configuration to run the installer program and perform the initial configuration of your Satellite Server.

3.2. DOWNLOADING AND INSTALLING FROM A DISCONNECTED NETWORK

When the intended host for the Red Hat Satellite Server is in a disconnected environment, you can install the Satellite Server using an ISO image. This method is not recommended for any other situation as ISO images might not contain the latest updates to Satellite. By installing Satellite with an ISO image, you might be installing an older version of Satellite. Older versions might be missing bug fixes and functionality.

Before You Begin

- A copy of the repositories used in the installation are stored in the **/opt/** directory. Ensure you have a minimum of 2GB of space for this file system and directory.

3.2.1. Downloading the Binary DVD Images

1. Go to Red Hat Customer Portal and log in.

2. Click **DOWNLOADS**.

3. Select **Red Hat Enterprise Linux**.

4. Ensure that you have the correct product and version for your environment.

   - **Product Variant** is set to **Red Hat Enterprise Linux Server**.

   - **Version** is set to the latest minor version of the product you plan to use as the base system.

- **Architecture** is set to the 64 bit version.

5. On the **Product Software** tab, download the Binary DVD image for the latest Red Hat Enterprise Linux Server version.

6. Click **DOWNLOADS** and select **Red Hat Satellite**.

7. Ensure that you have the correct product and version for your environment.

   - **Product Variant** is set to **Red Hat Satellite**.
   - **Version** is set to the latest minor version of the product you plan to use as the base system.
   - **Architecture** is set to the 64 bit version.

8. On the **Product Software** tab, download the Binary DVD image for the latest Red Hat Satellite version.

9. Copy the ISO files to the Satellite base system or other accessible storage device.

   ```
   scp localfile username@hostname:remotefile
   ```

3.2.2. Configuring the Base System with Offline Repositories

1. Create a directory to serve as the mount point for the ISO file corresponding to the base system's version.

   ```
   # mkdir /media/rhelX-server
   ```

2. Mount the ISO image to the mount point.

   ```
   mount -o loop rhelX-Server-DVD.iso /media/rhelX-server
   ```

   The following example shows mounting the ISO using Red Hat Enterprise Linux 7.2:

   ```
   # mount -o loop RHEL-7.2-20151030.0-Server-x86_64-dvd1.iso
   /media/rhel7-server
   mount: /dev/loop0 is write-protected, mounting read-only
   ```

3. Copy the ISO file's repository data file.

   ```
   # cp /media/rhelX-server/media.repo /etc/yum.repos.d/rhelX-
   server.repo
   ```

4. Edit the repository data file and add the **`baseurl`** directive.

   ```
   baseurl=file:///media/rhelX-server/
   ```

   The following example shows the repository data file using Red Hat Enterprise Linux 7.2:

   ```
   # vi /etc/yum.repos.d/rhel7-server.repo
   [InstallMedia]
   name=Red Hat Enterprise Linux 7.2
   ```

```
mediaid=1446216863.790260
metadata_expire=-1
gpgcheck=0
cost=500
baseurl=file:///media/rhel7-server/
enabled=1
```

5. Create a directory to serve as the mount point for the ISO file of the Satellite Server.

```
# mkdir /media/sat6
```

6. Mount the ISO image to the mount point.

```
mount -o loop sat6-DVD.iso /media/sat6
```

   The following example shows mounting the ISO using Red Hat Enterprise Linux 7.2:

```
# mount -o loop satellite-6.2-beta-rhel-7-x86_64-dvd.iso
/media/sat6
mount: /dev/loop1 is write-protected, mounting read-only
```

7. Verify that the repositories have been configured.

```
# yum repolist
Loaded plugins: product-id, search-disabled-repos, subscription-
manager
This system is not registered to Red Hat Subscription Management.
You can use subscription-manager to register.
repo id              repo name                     status
InstallMedia         Red Hat Enterprise Linux 7.2  4,620
katello-local        katello-local                 405
scl-local            scl-local                     2,362
repolist: 7,387
```

3.2.3. Installing from the Offline Repositories

1. Import the Red Hat GPG keys.

```
# rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
```

2. Ensure the base system is up to date with the Binary DVD image.

```
# yum update
```

3. Change to the directory where the Satellite ISO is mounted.

```
# cd /media/sat6/
```

4. Run the installer script in the mounted directory.

```
# ./install_packages
 This script will install the foreman packages on the current
machine.
```

```
   - Ensuring we are in an expected directory.
   - Copying installation files.
   - Creating a Repository File
   - Creating RHSCL Repository File
   - Checking to see if Foreman is already installed.
   - Importing the gpg key.
   - Foreman is not yet installed, installing it.
   - Installation repository will remain configured for future
package installs.
   - Installation media can now be safely unmounted.

Install is complete. Please run foreman-installer.
```

5. Unmount the ISO files.

```
# unmount /media/sat6
# unmount /media/rhelX-server
```

3.3. PERFORMING THE INITIAL CONFIGURATION

Initial configuration of your Satellite Server includes configuring a custom server certificate and either manually configuring Satellite or automatically configuring Satellite using an answer file.

- Manual Configuration - Satellite Server has default initial configuration options that prepare the server for use. You can override these settings depending on your environment’s requirements. You can run the command as often as needed to configure any necessary options.

- Automatic Configuration - You can automate most of the installation and configuration by using an answer file.

3.3.1. Configuring Satellite Server with a Custom Server Certificate

Red Hat Satellite 6 provides a default SSL certificate to enable encrypted communications between the Satellite Server, Capsule Servers, and all hosts. You can replace the default certificate with custom certificates if you want to do so. For example, your company might have a security policy stating that SSL certificates must be obtained from a specific Certificate Authority.

To replace the default SSL certificate you must obtain custom SSL certificates for the Satellite Server and all external Capsule Servers (if any), then install them on their respective hosts.

Note

Obtain custom SSL certificates for the Satellite Server and all external Capsule Servers **before** starting this procedure.

To use a custom certificate on Satellite Server, complete these steps:

1. Section 3.3.1.1, “Obtain an SSL Certificate for the Satellite Server”

2. Section 3.3.1.2, “Validate the Satellite Server’s SSL Certificate”

3. Section 3.3.1.3, “Run the Satellite Installer with Custom Certificate Parameters”

4. Section 3.3.1.4, “Install the New Certificate on all Hosts Connected to the Satellite Server”

If you have external Capsule Servers, you must also complete the steps in Section 4.5.1, “Configuring Capsule Server with a Custom Server Certificate”.

3.3.1.1. Obtain an SSL Certificate for the Satellite Server

Note

If you already have a custom SSL Certificate for the Satellite Server, skip this procedure.

1. Create a directory to contain all the source certificate files, accessible to only the **`root`** user.

   In these examples, the directory is **`/root/sat_cert`**.

   ```
   # mkdir /root/sat_cert
   # cd /root/sat_cert
   ```

2. Create a private key with which to sign the Certificate Signing Request (CSR).

   Note

   If you already have a private key for the Satellite Server, skip this step.

   ```
   # openssl genrsa -out /root/sat_cert/satellite_cert_key.pem 2048
   ```

3. Create a Certificate Signing Request (CSR)

   A Certificate Signing Request is a text file containing details of the server for which you are requesting a certificate. For this command, you provide the private key (output by the previous step), answer some questions about the Satellite Server, and the Certificate Signing Request is created.

   Note

   The certificate’s Common Name (CN) must match the fully-qualified domain name (FQDN) of the server on which it is used. If you are requesting a certificate for a Satellite Server, this is the FQDN of the Satellite Server. If you are requesting a certificate for a Capsule Server, this is the FQDN of the Capsule Server.

   To confirm a server’s FQDN, run the following command on that server: **`hostname -f`**.

   ```
   # openssl req -new \
     -key /root/sat_cert/satellite_cert_key.pem \ 1
     -out /root/sat_cert/satellite_cert_csr.pem   2
   ```

1

Satellite Server’s private key, used to sign the certificate

2

Certificate Signing Request file

Example Certificate Signing Request session

```
You are about to be asked to enter information that will be
incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished
Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

Country Name (2 letter code) [XX]:AU
State or Province Name (full name) []:Queensland
Locality Name (eg, city) [Default City]:Brisbane
Organization Name (eg, company) [Default Company Ltd]:Example
Organizational Unit Name (eg, section) []:Sales
Common Name (eg, your name or your server's hostname)
[]:satellite.example.com
Email Address []:example@example.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:password
An optional company name []:Example
```

4. Send the certificate request to the Certificate Authority.

   When you submit the request, be sure to specify the lifespan of the certificate. The method for sending the certificate request varies, so consult the Certificate Authority for the preferred method. In response to the request you can expect to receive a Certificate Authority bundle, and a signed certificate, in separate files.

3.3.1.2. Validate the Satellite Server’s SSL Certificate

Run the katello-certs-check command with the required parameters as per the following example. The command validates the input files required for custom certificates and outputs the commands necessary to install them on the Satellite Server, all Capsule Servers, and hosts under management with Satellite.

1. Validate the custom SSL certificate input files. Change the files' names to match your files.

```
# katello-certs-check \
   -c /root/sat_cert/satellite_cert.pem \      1
   -k /root/sat_cert/satellite_cert_key.pem \  2
```

```
    -r /root/sat_cert/satellite_cert_csr.pem \    3
    -b /root/sat_cert/ca_cert_bundle.pem          4
```

1

Certificate file for the Satellite Server, signed by your Certificate Authority

2

Satellite Server’s private key, used to sign the certificate

3

Certificate signing request file for the Satellite Server

4

Certificate Authority bundle

Example output of `katello-certs-check`

```
Validating the certificate subject=
/C=AU/ST=Queensland/L=Brisbane/O=Example/OU=Sales/CN=satellite.example.
com/emailAddress=example@example.com
Check private key matches the certificate: [OK]
Check ca bundle verifies the cert file: [OK]

Validation succeeded.

To install the Satellite main server with the custom certificates, run:

    satellite-installer --scenario satellite\
                        --certs-server-cert
"/root/sat_cert/satellite_cert.pem"\
                        --certs-server-cert-req
"/root/sat_cert/satellite_cert_csr.pem"\
                        --certs-server-key
"/root/sat_cert/satellite_cert_key.pem"\
                        --certs-server-ca-cert
"/root/sat_cert/ca_cert_bundle.pem"

To update the certificates on a currently running Satellite
installation, run:

    satellite-installer --scenario satellite\
                        --certs-server-cert
"/root/sat_cert/satellite_cert.pem"\
                        --certs-server-cert-req
"/root/sat_cert/satellite_cert_csr.pem"\
                        --certs-server-key
```

```
"/root/sat_cert/satellite_cert_key.pem"\
                        --certs-server-ca-cert
"/root/sat_cert/ca_cert_bundle.pem"\
                        --certs-update-server --certs-update-server-
ca

To use them inside a $CAPSULE, run this command INSTEAD:

    capsule-certs-generate --capsule-fqdn ""\
                           --certs-tar  "/root/certs.tar"\
                           --server-cert
"/root/sat_cert/satellite_cert.pem"\
                           --server-cert-req
"/root/sat_cert/satellite_cert_csr.pem"\
                           --server-key
"/root/sat_cert/satellite_cert_key.pem"\
                           --server-ca-cert
"/root/sat_cert/ca_cert_bundle.pem"\
                           --certs-update-server
```

3.3.1.3. Run the Satellite Installer with Custom Certificate Parameters

Now that you have created an SSL certificate and verified it is valid for use with Red Hat Satellite 6, the next step is to install the custom SSL certificate on the Satellite Server and all its hosts.

There is a minor variation to this step, depending on whether or not the Satellite Server is already installed. If it is **already** installed, the existing certificates must be *updated* with those in the certificates archive.

The commands in this section are output by the **katello-certs-check** command, as detailed in Section 3.3.1.2, “Validate the Satellite Server’s SSL Certificate”, and can be copied and pasted into a terminal.

1. Run the **satellite-installer**, depending on your situation.

   a. If Satellite is already installed, run the following command on the Satellite Server.

   ```
   # satellite-installer --scenario satellite\
      --certs-server-cert "/root/sat_cert/satellite_cert.pem"\
      --certs-server-cert-req
   "/root/sat_cert/satellite_cert_csr.pem"\
      --certs-server-key
   "/root/sat_cert/satellite_cert_key.pem"\
      --certs-server-ca-cert
   "/root/sat_cert/ca_cert_bundle.pem"\
      --certs-update-server --certs-update-server-ca
   ```

   Important parameters in this command include **--certs-update-server** and **--certs-update-server-ca**, which specify that the server’s SSL certificate and certificate authority are to be updated. For an explanation of all the installer’s parameters, run the command **satellite-installer --help**.

   b. If Satellite is **not** already installed, run the following command on the Satellite Server.

```
# satellite-installer --scenario satellite\
  --certs-server-cert "/root/sat_cert/satellite_cert.pem"\
  --certs-server-cert-req
"/root/sat_cert/satellite_cert_csr.pem"\
  --certs-server-key
"/root/sat_cert/satellite_cert_key.pem"\
  --certs-server-ca-cert
"/root/sat_cert/ca_cert_bundle.pem"
```

2. Verify the certificate has been successfully installed on the Satellite Server before installing it on hosts. On a computer with network access to the Satellite Server, start a web browser, navigate to the URL **`https://`*`satellite.example.com`*** and view the certificate's details.

3.3.1.4. Install the New Certificate on all Hosts Connected to the Satellite Server

Now that the custom SSL certificate has been installed on the Satellite Server, it must also be installed on every host registered to the Satellite Server. Run the following command on all applicable hosts.

```
# yum -y localinstall http://satellite.example.com/pub/katello-ca-
consumer-latest.noarch.rpm
```

3.3.2. Performing the Initial Configuration Manually

The initial configuration creates an organization, location, username, and password. After the initial configuration, you can create additional organizations and locations. You can rename the default organization or location and you can delete the default organization, but you cannot delete the default location.

1. Manually configure Satellite Server.

   If you do not specify any values, the default values are used. Use the 'foreman-installer --help' command to display the available options and any default values.

   ```
   # foreman-installer --scenario katello \
   --foreman-initial-organization "initial_organization_name" \
   --foreman-initial-location "initial_location_name" \
   --foreman-admin-username admin-username \
   --foreman-admin-password admin-password
   ```

   When the script completes successfully, the following output is displayed:

   ```
   Installing             Done
      [100%] [........................................]
      Success!
      * Satellite is running at https://satellite.example.com
          Default credentials are 'admin / changeme'
      * Capsule is running at https://satellite.example.com:9090
      * To install additional capsule on separate machine continue
   by running:

      capsule-certs-generate --capsule-fqdn "$CAPSULE" --certs-tar
   ```

```
"~/$CAPSULE-certs.tar"

   The full log is at /var/log/foreman-installer/foreman-
installer.log
```

3.3.3. Configuring Red Hat Satellite with an Answer File

You can use answer files to automate installations with customized options. The initial answer file is sparsely populated and after you run **`foreman-installer`** the first time, the answer file is populated with the standard parameter values for installation.

You should use the FQDN instead of the IP address where possible in case of network changes.

1. Copy the default answer file located at **`/etc/foreman-installer/scenarios.d/katello-answers.yaml`** to a location on your local file system.

   ```
   # cp /etc/foreman-installer/scenarios.d/katello-answers.yaml
   /etc/foreman-installer/scenarios.d/my-answer-file.yaml
   ```

2. To view all of the configurable options, run the **`foreman-installer --help`** command.

3. Open your copy of the answer file, edit the values to suit your environment, and save the file.

4. Open the **`/etc/foreman-installer/scenarios.d/katello-answers.yaml`** file and edit the answer file entry to point to your custom answer file.

   ```
   :answer_file: /etc/foreman-installer/scenarios.d/my-answer-
   file.yaml
   ```

5. Run the **`foreman-installer`** command.

   ```
   # foreman-installer --scenario katello \
   ```

3.4. PERFORMING ADDITIONAL CONFIGURATION

3.4.1. Configuring Satellite Server with HTTP Proxy

If your network uses an HTTP Proxy, you can enable it. Use the FQDN instead of the IP address where possible in case of network changes.

1. Verify that the 'http_proxy', 'https_proxy', and 'no_proxy' variables are not set.

   ```
   # export http_proxy=""
   # export https_proxy=$http_proxy
   # export no_proxy=$http_proxy
   ```

2. Run 'foreman-installer' with the HTTP proxy options.

   ```
   # foreman-installer --scenario katello \
   --katello-proxy-url=http://myproxy.example.com \
   ```

```
--katello-proxy-port=8080 \
--katello-proxy-username=proxy_username \
--katello-proxy-password=proxy_password
```

3. Verify that Satellite Server can connect to the Red Hat Content Delivery Network (CDN) and can synchronize its repositories.

   a. On the network gateway and the HTTP Proxy, enable TCP for the following host names:

Hostname	Port	Protocol
subscription.rhn.redhat.com	443	HTTPS
cdn.redhat.com	443	HTTPS
*.akamaiedge.net	443	HTTPS
cert-api.redhat.access.com (if using Red Hat Insights)	443	HTTPS
api.redhat.access.com (if using Red Hat Insights)	443	HTTPS

   For the IP addresses currently used by subscription.rhn.redhat.com, see the Knowledgebase solution What is the IP address range for 'subscription.rhn.redhat.com' on the Red Hat customer portal.

   For a list of IP addresses used by the Red Hat CDN (cdn.redhat.com), see the Knowledgebase article Public CIDR Lists for Red Hat on the Red Hat customer portal.

   b. On Satellite Server, complete the following details in the **`/etc/rhsm/rhsm.conf`** file:

```
# an http proxy server to use (enter server FQDN)
proxy_hostname = http_proxy.example.com

# port for http proxy server
proxy_port = 3128

# user name for authenticating to an http proxy, if needed
proxy_user =

# password for basic http proxy auth, if needed
proxy_password =
```

3.4.2. Configuring DNS, DHCP, and TFTP on Satellite Server

You can configure DNS, DHCP, and TFTP on Satellite Server.

If you want to configure external services, see Configuring Satellite Server with External Services for more information.

To view a complete list of configurable options, run the **foreman-installer --help** command.

Before You Begin

- Contact your network administrator to ensure that you have the correct settings.
- You should have the following information available:
  - DHCP IP address ranges
  - DHCP gateway IP address
  - DHCP nameserver IP address
  - DNS information
  - TFTP server name
- Use the FQDN instead of the IP address where possible in case of network changes.

Note

The information in the task is an example. You should use the information relevant to your own environment.

Configure DNS, DHCP, and TFTP on Satellite Server

If you have already created an admin user, do not include the --**foreman-admin-username** and **--foreman-admin-password** options.

If you do not specify the administrator user name and password, a default user **admin** is created and the password is automatically generated. You can view the user name and password in the command output. You can also retrieve the information from the **admin_password** parameter in the **/etc/katello-installer/answers.katello-installer.yaml** file.

1. Run **foreman-installer** with the options appropriate for your environment.

```
# foreman-installer --scenario katello \
--foreman-admin-username admin-username \
--foreman-admin-password admin-password \
--foreman-proxy-dns true \
--foreman-proxy-dns-interface eth0 \
--foreman-proxy-dns-zone example.com \
--foreman-proxy-dns-forwarders 172.17.13.1 \
--foreman-proxy-dns-reverse 13.17.172.in-addr.arpa \
--foreman-proxy-dhcp true \
--foreman-proxy-dhcp-interface eth0 \
--foreman-proxy-dhcp-range "172.17.13.100 172.17.13.150" \
--foreman-proxy-dhcp-gateway 172.17.13.1 \
--foreman-proxy-dhcp-nameservers 172.17.13.2 \
--foreman-proxy-tftp true \
--foreman-proxy-tftp-servername $(hostname) \
--capsule-puppet true \
--foreman-proxy-puppetca true
```

The status of the installation is displayed.

```
Success!
  * Satellite is running at https://satellite.example.com
      Default credentials are 'admin:*******'
  * Capsule is running at https://satellite.example.com:9090
  * To install additional capsule on separate machine continue by
running:"

      capsule-certs-generate --capsule-fqdn "$CAPSULE" --certs-
tar "~/$CAPSULE-certs.tar"

  The full log is at /var/log/foreman-installer/foreman-
installer.log
```

Note

Any changes to the settings require running **foreman-installer** again. You can run the script multiple times and it updates all configuration files with the changed values.

CHAPTER 4. INSTALLING CAPSULE SERVER

Before you install Capsule Server, you should ensure that your environment meets the requirements for installation. Capsule Server has the same requirements for installation as Satellite Server. For more information, see Preparing Your Environment for Installation.

4.1. REGISTERING CAPSULE SERVER TO SATELLITE SERVER

Before You Begin

- The Satellite Server must have a manifest installed with the appropriate repositories for the organization you intend to subscribe to. The manifest must contain repositories for the Capsule's base system as well as any clients connected to the Capsule.
- The Satellite Server's base system must be able to resolve the host name of the Capsule Server's base system and vice versa.
- You must have a Satellite Server user name and password. For more information, see the Red Hat Satellite 6.2-Beta Server Administration Guide.

Register Capsule Server to Satellite Server

1. Install the Satellite Server's CA certificate in the Capsule Server.

```
# rpm -Uvh http://satellite.example.com/pub/katello-ca-consumer-
latest.noarch.rpm
```

2. Register the Capsule server with your organization.

```
# subscription-manager register --org organization_name
```

4.2. IDENTIFYING AND ATTACHING THE CAPSULE SERVER SUBSCRIPTION

After you have registered the Capsule Server, you need to identify your Capsule Server subscription Pool ID. The Pool ID enables you to attach the required subscription to your Capsule Server. The Capsule Server subscription provides access to the Capsule Server content, as well as Red Hat Enterprise Linux, Red Hat Software Collections (RHSCL), and Red Hat Satellite. This is the only subscription required.

1. Identify your Capsule Server subscription.

```
# subscription-manager list --all --available
```

The command displays output similar to the following:

```
+
+-------------------------------------------+
    Available Subscriptions
+-------------------------------------------+

Subscription Name: Red Hat Satellite Capsule Server
```

```
Provides:           Red Hat Satellite Proxy
                    Red Hat Satellite Capsule
                    Red Hat Software Collections (for RHEL
Server)
                    Red Hat Satellite Capsule
                    Red Hat Enterprise Linux Server
                    Red Hat Enterprise Linux High Availability
(for RHEL Server)
                    Red Hat Software Collections (for RHEL
Server)
                    Red Hat Enterprise Linux Load Balancer (for
RHEL Server)
SKU:                MCT0369
Pool ID:            9e4cc4e9b9fb407583035861bb6be501
Available:          3
Suggested:          1
Service Level:      Premium
Service Type:       L1-L3
Multi-Entitlement: No
Ends:               10/07/2022
System Type:        Physical
```

2. Make a note of the Pool ID so that you can attach it to your Satellite host. Your Pool ID will be different than the example provided.

3. Attach your subscription to your Capsule Server, using your Pool ID:

   ```
   # subscription-manager attach --
   pool=Red_Hat_Satellite_Capsule_Pool_Id
   ```

   The outputs displays something similar to the following:

   ```
   Successfully attached a subscription for: Red Hat Capsule Server
   ```

4. To verify that the subscriptions are successfully attached, run the following command:

   ```
   # subscription-manager list --consumed
   ```

4.3. CONFIGURING REPOSITORIES

1. Disable all existing repositories.

   ```
   # subscription-manager repos --disable ''*''
   ```

2. Enable the Red Hat Satellite and Red Hat Enterprise Linux and Red Hat Software Collections repositories.

   Ensure the Red Hat Enterprise Linux repository matches the specific version you are using.

If you are using…	Run this command…
Red Hat Enterprise Linux 6	subscription-manager repos --enable rhel-6-server-rpms \ --enable rhel-6-server-satellite-capsule-6-beta-rpms
Red Hat Enterprise Linux 7	subscription-manager repos --enable rhel-7-server-rpms \ --enable rhel-7-server-satellite-capsule-6-beta-rpms

Note

If you are using a different version of Red Hat Enterprise Linux, change the repository based on your specific version.

3. Clear out any metadata left from any non-Red Hat yum repositories.

```
# yum clean all
```

4. Verify that the repositories have been enabled.

```
# yum repolist enabled
```

The following output displays:

```
Loaded plugins: langpacks, product-id, subscription-manager
repo id                                                        repo
name                                                           status
!rhel-7-server-rpms/7Server/x86_64                             Red Hat
Enterprise Linux 7 Server (RPMs)                             7,617
!rhel-7-server-satellite-capsule-6-beta-rpms/x86_64            Red Hat
Satellite Capsule 6.2 Beta(for RHEL 7 Server) (RPMs)           176
repolist: 7,793
```

4.4. INSTALLING CAPSULE SERVER

1. Install the installation package.

```
# yum install satellite-capsule
```

4.5. PERFORMING INITIAL CONFIGURATION OF CAPSULE SERVER

Initial configuration of Capsule Server enables you to create and install a custom server certificate.

4.5.1. Configuring Capsule Server with a Custom Server Certificate

Red Hat Satellite 6 comes with default SSL certificates to enable encrypted communications between the Satellite Server, Capsule Servers, and all hosts. You can replace the default certificates with custom certificates if required. For example, your company's security policy dictates that SSL certificates must be obtained from a specific Certificate Authority.

Prerequisites

- Satellite Server configured with a custom certificate. For more information, see Section 3.3.1, "Configuring Satellite Server with a Custom Server Certificate".
- Capsule Server installed and registered to the Satellite Server.

To use a custom certificate on each Capsule Server, complete these steps:

1. Section 4.5.1.1, "Obtain an SSL Certificate for the Capsule Server"
2. Section 4.5.1.2, "Create the Capsule Server's certificates archive file"
3. Section 4.5.1.3, "Run the Capsule Installer with Custom Certificate Parameters"
4. Section 4.5.1.4, "Install the Capsule Server's New Certificate on all Hosts"

4.5.1.1. Obtain an SSL Certificate for the Capsule Server

Note

- If you already have a custom SSL Certificate for the Capsule Server, skip this procedure.
- Do **not** use the Satellite Server's certificate on any of the Capsule Servers as each server's certificate is unique.

Obtain a custom SSL certificate for the Capsule Server, following the instructions in Section 3.3.1.1, "Obtain an SSL Certificate for the Satellite Server". The process is identical for a Capsule Server, but to keep files separate, give each output file a different prefix. For example, if you have a Capsule Server named **`capsule_apac`**, you might use a prefix of *capsule_apac* instead of *satellite*. For example, the file containing the Capsule Server's private key would be named *capsule_apac_cert_key.pem*.

4.5.1.2. Create the Capsule Server's certificates archive file

The Capsule Server's installer requires the server's certificates be provided in an archive file. To create this file, you use the **`capsule-certs-generate`** command.

1. Generate the Capsule Server's certificates archive file

   The **`capsule-certs-generate`** command must be run once for every external Capsule Server. In the examples here, **`capsule.example.com`** is the example FQDN and **`capsule_certs.tar`** the example archive file's name. Replace these with values appropriate to your environment, taking care not to overwrite an existing certificates archive file. For example, if you have Capsule Servers named **`capsule1`** and **`capsule2`**, you could name the certificates archive files **`capsule1_certs.tar`** and **`capsule2_certs.tar`**.

The **`capsule-certs-generate`** command, with all required parameters and their values, is output by the **`katello-certs-check`** command, as detailed in Section 3.3.1.2, “Validate the Satellite Server’s SSL Certificate”. Copy and paste this command into a terminal, but before running the command, edit the values for **`--capsule-fqdn`** and **`--certs-tar`** to suit your environment.

a. If the Capsule Server is already installed, run the following command on the Satellite Server. The key parameter in this command is **`--certs-update-server`**, which indicates that the existing certificates are to be updated.

```
# capsule-certs-generate --capsule-fqdn
"capsule.example.com"\
  --certs-tar  "/root/sat_cert/capsule_certs.tar"\
  --server-cert "/root/sat_cert/satellite_cert.pem"\
  --server-cert-req
"/root/sat_cert/satellite_cert_csr.pem"\
  --server-key "/root/sat_cert/satellite_cert_key.pem"\
  --server-ca-cert "/root/sat_cert/ca_cert_bundle.pem"\
  --certs-update-server
```

b. If the Capsule Server is **not** already installed, run the following command on the Satellite Server.

```
# capsule-certs-generate --capsule-fqdn
"capsule.example.com"\
  --certs-tar  "/root/sat_cert/capsule_certs.tar"\
  --server-cert "/root/sat_cert/satellite_cert.pem"\
  --server-cert-req
"/root/sat_cert/satellite_cert_csr.pem"\
  --server-key "/root/sat_cert/satellite_cert_key.pem"\
  --server-ca-cert "/root/sat_cert/ca_cert_bundle.pem"
```

Example extract of output from `capsule-certs-generate`

```
# satellite-installer --scenario capsule\
                    --capsule-parent-fqdn
"satellite.example.com"\
                    --foreman-proxy-register-in-foreman
"true"\
                    --foreman-proxy-foreman-base-url
"https://satellite.example.com"\
                    --foreman-proxy-trusted-hosts
"satellite.example.com"\
                    --foreman-proxy-trusted-hosts
"capsule.example.com"\
                    --foreman-proxy-oauth-consumer-key
"FeQsbASvCjvvaqE6duKH6SoYZWg4jwjg"\
                    --foreman-proxy-oauth-consumer-secret
"7UhPXFPDBongvdTbNixbsWR5WFZsKEgF"\
                    --capsule-pulp-oauth-secret
"VpQ9587tVmYeuY4Du6VitmZpZE5vy9ac"\
                    --capsule-certs-tar
"/root/sat_cert/capsule_certs.tar"
```

2. On the Satellite Server, copy the certificates archive file to the Capsule Server.

   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/sat_cert/capsule_certs.tar root@capsule.example.com:~
   ```

4.5.1.3. Run the Capsule Installer with Custom Certificate Parameters

The **satellite-installer** command, with all required parameters and their values, is output by the **capsule-certs-generate** command, as detailed in Section 4.5.1.2, "Create the Capsule Server's certificates archive file". Copy and paste this command into a terminal, but before running the command, edit the value for **--capsule-certs-tar** to match the location of the certificates archive file. If you used the command in Section 4.5.1.2, "Create the Capsule Server's certificates archive file", the file is in the **root** user's directory.

If you want additional features enabled on the Capsule Server, you can add these to the **satellite-installer** command, or run the **satellite-installer** command with the additional parameters later. For an explanation of all the installer's parameters, run the command **satellite-installer --help**.

Note

The **satellite-installer** command, as output by the **capsule-certs-generate** command, is unique to each Capsule Server. Do **not** use the same command on more than one Capsule Server.

```
# satellite-installer --scenario capsule\
                    --capsule-parent-fqdn
"satellite.example.com"\
                    --foreman-proxy-register-in-foreman
"true"\
                    --foreman-proxy-foreman-base-url
"https://satellite.example.com"\
                    --foreman-proxy-trusted-hosts
"satellite.example.com"\
                    --foreman-proxy-trusted-hosts
"capsule.example.com"\
                    --foreman-proxy-oauth-consumer-key
"FeQsbASvCjvvaqE6duKH6SoYZWg4jwjg"\
                    --foreman-proxy-oauth-consumer-secret
"7UhPXFPDBongvdTbNixbsWR5WFZsKEgF"\
                    --capsule-pulp-oauth-secret
"VpQ9587tVmYeuY4Du6VitmZpZE5vy9ac"\
                    --capsule-certs-tar
"/root/sat_cert/capsule_certs.tar"
```

4.5.1.4. Install the Capsule Server's New Certificate on all Hosts

4.5.1.4. Install the Capsule Server's New Certificate on all Hosts

Hosts which connect to an external Capsule server require that server's custom certificate. Run the following command on all the Capsule Server's hosts.

Note

In the following command, be sure to use the Capsule Server's host name, **not** that of the Satellite Server.

```
# yum -y localinstall http://capsule.example.com/pub/katello-ca-
consumer-latest.noarch.rpm
```

4.6. PERFORMING ADDITIONAL CONFIGURATION ON CAPSULE SERVER

4.6.1. Adding Life Cycle Environments to Capsule Servers

If your Capsule Server has content functionality enabled, you must add an environment. Adding an environment enables Capsule Server to synchronize content from Satellite Server and provide content to host systems.

Capsule Server is configured using Satellite Server's Hammer CLI. You must execute all commands on Satellite Server.

1. Log in to the Hammer CLI as root.
2. Display a list of all Capsule Servers and note the ID.

   ```
   # hammer capsule list
   ```

3. Using the ID, verify the details of your Capsule Server.

   ```
   # hammer capsule info --id capsule_id_number
   ```

4. Verify the life cycle environments available and note the environment ID.

   ```
   # hammer capsule content available-lifecycle-environments --id
   capsule_id_number
   ```

   Available life cycle environments are available for Capsule Server, but not currently attached.

5. Add the life cycle environment to your Capsule Server.

   ```
   # hammer capsule content add-lifecycle-environment --id
   capsule_id_number --environment-id environment_id_number
   ```

6. Repeat for each life cycle environment you want to add to Capsule Server.
7. To synchronize all content from your Satellite Server environment to Capsule Server, run the following command:

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

8. To synchronize a specific life cycle environment from your Satellite Server to Capsule Server, run the following command:

```
# hammer capsule content synchronize --id
external_capsule_id_number --environment-id environment_id_number
```

4.6.2. Enabling Power Management on Managed Hosts

When you enable the baseboard management controller (BMC) module on the Capsule Server, you can use power management commands on managed hosts using the intelligent platform management interface (IPMI) or a similar protocol.

The BMC service on the satellite Capsule Server enables you to perform a range of power management tasks. The underlying protocol for this feature is IPMI; also referred to as the BMC function. IPMI uses a special network interface on the managed hardware that is connected to a dedicated processor that runs independently of the host’s CPUs. In many instances the BMC functionality is built into chassis-based systems as part of chassis management (a dedicated module in the chassis).

For more information on the BMC service, see the *Red Hat Satellite 6.2-Beta Host Configuration Guide*.

Before You Begin

- All managed hosts must have a network interface, with type **BMC**. Satellite uses this NIC to pass the appropriate credentials to the host.

Enable Power Management on Managed Hosts

1. Run Foreman installer with the options to enable BMC.

```
# foreman-installer --scenario capsule\
--foreman-proxy-bmc "true"\
--foreman-proxy-bmc-default-provider "freeipmi"
```

4.6.3. Configuring DNS and DHCP on Capsule Server

You can configure DNS, DHCP, and TFTP on Capsule Server.

You can also configure Capsule Server to use external DNS and DHCP services. See Configuring Satellite Server with External Services for more information.

To view a complete list of configurable options, run the **foreman-installer --help** command.

Before You Begin

- You must have the correct network name (**dns-interface**)for the DNS server.
- You must have the correct interface name (**dhcp-interface**) for the DHCP server.

Configure DNS, DHCP, and TFTP on Capsule Server

1. Run capsule installer with the options applicable to your environment.

   The following example shows full provisioning services:

```
# foreman-installer --scenario capsule\
--tftp=true\
--foreman-oauth-key    "your_organization_key"\
--foreman-oauth-secret "your_organization_secret"\
--certs-tar            "~/capsule.example.com-certs.tar"\
--templates=true\
--dhcp=true\
--dhcp-gateway=192.168.122.1\
--dhcp-nameservers=192.168.122.1\
--dhcp-range="192.168.122.100 192.168.122.200"\
--dhcp-interface=eth0\
--dns=true\
--dns-forwarders=8.8.8.8\
--dns-interface=eth0\
--dns-zone=example.com

# foreman-installer --scenario capsule\
--foreman-admin-username admin-username \
--foreman-admin-password admin-password \
--foreman-proxy-dns true \
--foreman-proxy-dns-interface eth0 \
--foreman-proxy-dns-zone example.com \
--foreman-proxy-dns-forwarders 172.17.13.1 \
--foreman-proxy-dns-reverse 13.17.172.in-addr.arpa \
--foreman-proxy-dhcp true \
--foreman-proxy-dhcp-interface eth0 \
--foreman-proxy-dhcp-range "172.17.13.100 172.17.13.150" \
--foreman-proxy-dhcp-gateway 172.17.13.1 \
--foreman-proxy-dhcp-nameservers 172.17.13.2 \
--foreman-proxy-tftp true \
--foreman-proxy-tftp-servername $(hostname) \
--capsule-puppet true \
--foreman-proxy-puppetca true
```

CHAPTER 5. CONFIGURING EXTERNAL SERVICES

Some environments have existing DNS, DHCP, and TFTP services and do not need to use the Satellite Server to provide these services. If you want to use external servers to provide DNS, DHCP, or TFTP, you can configure them for use with Satellite Server.

5.1. CONFIGURING SATELLITE WITH EXTERNAL DNS

You can configure Satellite to use an external server to provide DNS service. The recommended and tested version of Red Hat Enterprise Linux Server is 7.1.

1. Deploy a Red Hat Enterprise Linux Server and install the ISC DNS Service.

   ```
   # yum install bind bind-utils
   ```

2. Create the configuration for the domain.

   The following example configures a domain **virtual.lan** as one subnet 192.168.38.0/24, a security key named **foreman**, and sets forwarders to Google’s public DNS addresses (8.8.8.8 and 8.8.4.4).

   ```
   # cat /etc/named.conf
   include "/etc/rndc.key";

   controls  {
       inet 192.168.38.2 port 953 allow { 192.168.38.1;
   192.168.38.2; } keys { "capsule"; };
   };

   options  {
       directory "/var/named";
       forwarders { 8.8.8.8; 8.8.4.4; };
   };

   include "/etc/named.rfc1912.zones";

   zone "38.168.192.in-addr.arpa" IN {
       type master;
       file "dynamic/38.168.192-rev";
       update-policy {
           grant "capsule" zonesub ANY;
       };
   };

   zone "virtual.lan" IN {
       type master;
       file "dynamic/virtual.lan";
       update-policy {
           grant "capsule" zonesub ANY;
       };
   };
   ```

   The **inet** line must be entered as one line in the configuration file.

3. Create a key file.

   ```
   # ddns-confgen -k capsule
   ```

   This command can take a long time to complete.

4. Copy and paste the output from the key section into a separate file called **`/etc/rndc.key`**.

   ```
   # cat /etc/rndc.key
   key "capsule" {
           algorithm hmac-sha256;
           secret "GeBbgGoLedEAAwNQPtPh3zP56MJbkwM84UJDtaUS9mw=";
   };
   ```

   Important

   This is the key used to change DNS server configuration. Only the root user should read and write to it.

5. Create zone files.

   ```
   # cat /var/named/dynamic/virtual.lan
   $ORIGIN .
   $TTL 10800      ; 3 hours
   virtual.lan             IN SOA  service.virtual.lan.
   root.virtual.lan. (
                                   9          ; serial
                                   86400      ; refresh (1 day)
                                   3600       ; retry (1 hour)
                                   604800     ; expire (1 week)
                                   3600       ; minimum (1 hour)
                                   )
                           NS      service.virtual.lan.
   $ORIGIN virtual.lan.
   $TTL 86400      ; 1 day
   capsule                 A       192.168.38.1
   service                 A       192.168.38.2
   ```

6. Create the reverse zone file.

   ```
   # cat /var/named/dynamic/38.168.192-rev
   $ORIGIN .
   $TTL 10800      ; 3 hours
   38.168.192.in-addr.arpa IN SOA  service.virtual.lan.
   root.38.168.192.in-addr.arpa. (
                                   4          ; serial
                                   86400      ; refresh (1 day)
                                   3600       ; retry (1 hour)
                                   604800     ; expire (1 week)
                                   3600       ; minimum (1 hour)
                                   )
                           NS      service.virtual.lan.
   ```

```
$ORIGIN 38.168.192.in-addr.arpa.
$TTL 86400      ; 1 day
1                PTR     capsule.virtual.lan.
2                PTR     service.virtual.lan.
```

There should be no extra non-ASCII characters.

5.2. VERIFYING AND STARTING THE DNS SERVICE

1. Validate the syntax.

   ```
   # named-checkconf -z /etc/named.conf
   ```

2. Start the server.

If you are using…	Run this command…
Red Hat Enterprise Linux 7	# systemctl restart named
Red Hat Enterprise Linux 6	# service named restart

3. Add a new host.

   The following uses the example host 192.168.38/2. You should change this to suit your environment.

   ```
   # echo -e "server 192.168.38.2\n \
   update add aaa.virtual.lan 3600 IN A 192.168.38.10\n \
   send\n" | nsupdate -k /etc/rndc.key
   ```

4. Test that the DNS service can resolve the new host.

   ```
   # nslookup aaa.virtual.lan 192.168.38.2
   ```

5. If necessary, delete the new entry.

   ```
   # echo -e "server 192.168.38.2\n \
   update delete aaa.virtual.lan 3600 IN A 192.168.38.10\n \
   send\n" | nsupdate -k /etc/rndc.key
   ```

6. Configure the firewall for external access to the DNS service (UDP and TCP on port 53).

   a. For Satellite Server running Red Hat Enterprise Linux 7, configure the firewall:

      ```
      # firewall-cmd --add-port="53/udp" --add-port="53/tcp" \
      && firewall-cmd --permanent --add-port="53/udp" --add-
      port="53/tcp"
      ```

b. For Satellite Server running Red Hat Enterprise Linux 6, start and enable the iptables service:

```
# iptables -A INPUT -m state --state NEW -p udp --dport 53
-j ACCEPT \
&& iptables -A INPUT -m state --state NEW -p tcp --dport 53
-j ACCEPT \
&& iptables-save > /etc/sysconfig/iptables
```

c. Verify that the iptables service is started and enabled.

```
# service iptables start
# chkconfig iptables on
```

5.3. CONFIGURING CAPSULE SERVER WITH EXTERNAL DNS

1. On the Red Hat Enterprise Linux Server, install the ISC DNS Service.

```
# yum install bind bind-utils
```

Ensure that the **nsupdate** utility was installed. The Capsule uses the **nsupdate** utility to update DNS records on the remote server.

2. Copy the **/etc/rndc.key** file from the services server to the Capsule Server.

```
scp localfile username@hostname:remotefile
```

3. Verify that the key file has the correct owner, permissions, and SELinux label.

```
# ls /etc/rndc.key -Zla
-rw-r-----. root named system_u:object_r:dnssec_t:s0
/etc/rndc.key
```

4. Test that the **nsupdate** utility by adding a host remotely.

```
# echo -e "server 192.168.38.2\n \
update add aaa.virtual.lan 3600 IN A 192.168.38.10\n \
send\n" | nsupdate -k /etc/rndc.key
# nslookup aaa.virtual.lan 192.168.38.2
# echo -e "server 192.168.38.2\n \
update delete aaa.virtual.lan 3600 IN A 192.168.38.10\n \
send\n" | nsupdate -k /etc/rndc.key
```

5. Edit the **/etc/foreman-proxy/settings.d/dns.yml** file.

a. Enable the smart-proxy module setting provider to **nsupdate**.

b. Add the IP address to the DNS server.

c. Set the default time to **live** for records created by the Capsule Server.

```
---
```

```
:enabled: true
:dns_provider: nsupdate
:dns_key: /etc/rndc.key
:dns_server: 192.168.38.2
:dns_ttl: 86400
```

Since the configuration file uses YAML format, the three dash characters are required.

6. Restart foreman-proxy service.

If you are using…	Run this command…
Red Hat Enterprise Linux 7	systemctl restart foreman-proxy
Red Hat Enterprise Linux 6	service foreman-proxy restart

7. Log in to the Satellite Server web UI.

8. Go to **Infrastructure** > **Capsules**. Locate the appropriate Capsule Server and select **Refresh features** from the drop-down list. The DNS feature should appear.

9. Associate the DNS service with the appropriate subnets and domain.

5.4. CONFIGURING SATELLITE SERVER WITH EXTERNAL DHCP

1. Deploy a Red Hat Enterprise Linux Server and install the ISC DHCP Service.

```
# yum install dhcp
```

2. Generate a security token in an empty directory.

```
# dnssec-keygen -a HMAC-MD5 -b 512 -n HOST omapi_key
```

3. For testing or proof-of-concept deployments, run an insecure non-blocking device command.

```
# dnssec-keygen -r /dev/urandom -a HMAC-MD5 -b 512 -n HOST
omapi_key
```

This creates a key pair in two files in the current directory.

4. Copy the secret hash from the key.

```
# cat Komapi_key.+*.private |grep ^Key|cut -d ' ' -f2
```

5. Edit the **dhcpd** configuration file for all of the subnets and add the key.

```
# cat /etc/dhcp/dhcpd.conf
```

```
default-lease-time 604800;
max-lease-time 2592000;
log-facility local7;

subnet 192.168.38.0 netmask 255.255.255.0 {
 range 192.168.38.10 192.168.38.100;
 option routers 192.168.38.1;
 option subnet-mask 255.255.255.0;
 option domain-search "virtual.lan";
 option domain-name "virtual.lan";
 option domain-name-servers 8.8.8.8;
}

omapi-port 7911;
key omapi_key {
 algorithm HMAC-MD5;
 secret "jNSE5YI3H1A8Oj/tkV4...A2ZOHb6zv315CkNAY7DMYYCj48Umw==";
};
omapi-key omapi_key;
```

6. Delete the two key files from the directory where you created them.

7. Define each subnet on the Satellite Server.

   It is recommended to set up a lease range and reservation range separately to prevent conflicts. For example, the lease range is 192.168.38.10 to 192.168.38.100 so the reservation range (defined in the Satellite web UI) is 192.168.38.101 to 192.168.38.250. Do not set DHCP Capsule for the defined Subnet yet.

   ISC DHCP listens only on interfaces that match defined subnets. In this example, the server has an interface that routes to 192.168.38.0 subnet directly.

8. Configure the firewall for external access to the DHCP server.

   a. For Satellite Server running Red Hat Enterprise Linux 7, configure the firewall.

      ```
      # firewall-cmd --add-service dhcp \
      && firewall-cmd --permanent --add-service dhcp
      ```

   b. For Satellite Server running Red Hat Enterprise Linux 6, start and enable the iptables service.

      ```
      # iptables -A INPUT -m state --state NEW -p tcp --dport 67
      -j ACCEPT \
      && iptables-save > /etc/sysconfig/iptables
      ```

   c. Verify that the iptables service is started and enabled.

      ```
      # service iptables start
      # chkconfig iptables on
      ```

9. Determine the UID and GID numbers of the foreman-proxy user on the Capsule Server. Create the same user and group with the same IDs on the DHCP server.

   ```
   # groupadd -g 990 foreman-proxy
   # useradd -u 992 -g 990 -s /sbin/nologin foreman-proxy
   ```

10. To make the configuration files readable, restore the read and execute flags.

    ```
    # chmod o+rx /etc/dhcp/
    # chmod o+r /etc/dhcp/dhcpd.conf
    # chattr +i /etc/dhcp/ /etc/dhcp/dhcpd.conf
    ```

11. Start the DHCP service.

If you are using…	Run this command…
Red Hat Enterprise Linux 7	systemctl start dhcpd
Red Hat Enterprise Linux 6	service dhcpd start

12. Export the DHCP configuration and leases files using NFS.

    ```
    # yum install nfs-utils
    # systemctl enable rpcbind nfs-server
    # systemctl start rpcbind nfs-server nfs-lock nfs-idmapd
    ```

13. Create the DHCP configuration and leases files to be exported using NFS.

    ```
    # mkdir -p /exports/var/lib/dhcpd /exports/etc/dhcp
    ```

14. Add the newly created mount point to /etc/fstab file.

    ```
    /var/lib/dhcpd /exports/var/lib/dhcpd none bind,auto 0 0
    /etc/dhcp /exports/etc/dhcp none bind,auto 0 0
    ```

15. Mount the file systems in /etc/fstab.

    ```
    # mount -a
    ```

16. Ensure the following lines are present in /etc/exports:

    ```
    /exports
    192.168.38.1(rw,async,no_root_squash,fsid=0,no_subtree_check)
    ```

    ```
    /exports/etc/dhcp
    192.168.38.1(ro,async,no_root_squash,no_subtree_check,nohide)
    ```

    ```
    /exports/var/lib/dhcpd
    192.168.38.1(ro,async,no_root_squash,no_subtree_check,nohide)
    ```

17. Reload the NFS server.

    ```
    # exportfs -rva
    ```

18. Configure the firewall for the DHCP omapi port 7911 for the Capsule Server.

    - On Red Hat Enterprise Linux 7, run the following command:

      ```
      # firewall-cmd --add-port="7911/tcp" \
      && firewall-cmd --permanent --add-port="7911/tcp"
      ```

    - On Red Hat Enterprise Linux 6, run the following command:

      ```
      # iptables -A INPUT -m state --state NEW -p tcp --dport 7911 -
      j ACCEPT \
      && iptables-save > /etc/sysconfig/iptables
      ```

19. Ensure that the iptables service is started and enabled.

    ```
    # service iptables start
    # chkconfig iptables on
    ```

20. If required, configure the firewall for external access to NFS.

    Clients are configured using NFSv3.

    a. On Red Hat Enterprise Linux 7, use the **firewalld** daemon’s NFS service to configure the firewall.

    ```
    #  firewall-cmd --zone public --add-service mountd \
    && firewall-cmd --zone public --add-service rpc-bind \
    && firewall-cmd --zone public --add-service nfs \
    && firewall-cmd --permanent --zone public --add-service
    mountd \
    && firewall-cmd --permanent --zone public --add-service
    rpc-bind \
    && firewall-cmd --permanent --zone public --add-service nfs
    ```

    b. On Red Hat Enterprise Linux 6, configure the ports for NFSv3 in the **/etc/sysconfig/nfs** file.

    ```
    LOCKD_TCPPORT=32803
    LOCKD_UDPPORT=32769
    MOUNTD_PORT=892
    RQUOTAD_PORT=875
    STATD_PORT=662
    STATD_OUTGOING_PORT=2020
    ```

    c. Restart the service.

    ```
    # service nfs restart
    ```

    d. Add rules to the **/etc/sysconfig/iptables** file.

    ```
    # iptables -A INPUT -s 192.168.1.0/24 -m state --state NEW
    -p udp --dport 111 -j ACCEPT \
    && iptables -A INPUT -s 192.168.1.0/24 -m state --state NEW
    -p tcp --dport 111 -j ACCEPT \
    ```

```
&& iptables -A INPUT -s 192.168.1.0/24 -m state --state NEW
-p udp --dport 2049 -j ACCEPT \
&& iptables -A INPUT -s 192.168.1.0/24 -m state --state NEW
-p tcp --dport 2049 -j ACCEPT \
&& iptables -A INPUT -s 192.168.1.0/24 -m state --state NEW
-p tcp --dport 32803 -j ACCEPT \
&& iptables -A INPUT -s 192.168.1.0/24 -m state --state NEW
-p udp --dport 32769 -j ACCEPT \
&& iptables -A INPUT -s 192.168.1.0/24 -m state --state NEW
-p udp --dport 892 -j ACCEPT \
&& iptables -A INPUT -s 192.168.1.0/24 -m state --state NEW
-p tcp --dport 892 -j ACCEPT \
&& iptables -A INPUT -s 192.168.1.0/24 -m state --state NEW
-p udp --dport 875 -j ACCEPT \
&& iptables -A INPUT -s 192.168.1.0/24 -m state --state NEW
-p tcp --dport 875 -j ACCEPT \
&& iptables -A INPUT -s 192.168.1.0/24 -m state --state NEW
-p udp --dport 662 -j ACCEPT \
&& iptables -A INPUT -s 192.168.1.0/24 -m state --state NEW
-p tcp --dport 662 -j ACCEPT \
&& iptables-save > /etc/sysconfig/iptables
```

e. Restart the firewall.

```
# service iptables restart
```

For more information on using NFSv3 behind a firewall on Red Hat Enterprise Linux 6, see the *Red Hat Enterprise Linux 6 Storage Administration Guide* and the section called "Running NFS Behind a Firewall" in the *Red Hat Enterprise Linux 6 Security Guide*.

5.5. CONFIGURING CAPSULE SERVER WITH EXTERNAL DHCP

1. Install the NFS client.

```
# yum install nfs-utils
```

2. Create the DHCP directories for NFS.

```
# mkdir -p /mnt/nfs/etc/dhcp /mnt/nfs/var/lib/dhcpd
```

3. Change the file owner.

```
# chown -R foreman-proxy /mnt/nfs
```

4. Verify communication with the NFS server and RPC communication paths.

```
# showmount -e 192.168.38.2
# rpcinfo -p 192.168.38.2
```

5. Add the following lines to the **/etc/fstab** file:

```
192.168.38.2:/exports/etc/dhcp /mnt/nfs/etc/dhcp nfs
ro,vers=3,auto,nosharecache,context="system_u:object_r:dhcp_etc_t
:s0" 0 0
```

```
192.168.38.2:/exports/var/lib/dhcpd /mnt/nfs/var/lib/dhcpd nfs
ro,vers=3,auto,nosharecache,context="system_u:object_r:dhcpd_stat
e_t:s0" 0 0
```

6. Mount the file systems on **`/etc/fstab`**.

```
# mount -a
```

7. Read the relevant files.

```
# su foreman-proxy -s /bin/bash
bash-4.2$ cat /mnt/nfs/etc/dhcp/dhcpd.conf
bash-4.2$ cat /mnt/nfs/var/lib/dhcpd/dhcpd.leases
bash-4.2$ exit
```

8. On the Capsule Server, edit **`/etc/foreman-proxy/settings.d/dhcp.yml`**.

   Since the configuration file uses YAML format, the three dash characters are required.

```
---
:enabled: true
:dhcp_vendor: isc
:dhcp_config: /mnt/nfs/etc/dhcp/dhcpd.conf
:dhcp_leases: /mnt/nfs/var/lib/dhcpd/dhcpd.leases
:dhcp_key_name: omapi_key
:dhcp_key_secret:
jNSE5YI3H1A8Oj/tkV4...A2ZOHb6zv315CkNAY7DMYYCj48Umw==
:dhcp_server: dhcp.example.com
```

9. Restart the proxy.

If you are using…	Run this command…
Red Hat Enterprise Linux 7	systemctl restart foreman-proxy
Red Hat Enterprise Linux 6	service foreman-proxy restart

10. Log in to the Satellite Server web UI.

11. Go to **Infrastructure** > **Capsules**. Locate the appropriate Capsule Server and select **Refresh features** from the drop-down list. The DHCP feature should appear.

12. Associate the DHCP service with the appropriate subnets and domain.

5.6. CONFIGURING SATELLITE SERVER WITH EXTERNAL TFTP

Before You Begin

- You should have already configured the firewall for external access to NFS. To configure the firewall, see Configuring Satellite Server with External DHCP.

Configure Satellite Server with External TFTP

1. Install and enable the TFTP server.

   ```
   # yum install tftp-server syslinux
   ```

   a. On Red Hat Enterprise 7, enable and activate the **`tftp.socket`** unit.

      ```
      # systemctl enable tftp.socket
      # systemctl start tftp.socket
      ```

   b. On Red Hat Enterprise Linux 6, enable and start the **`xinetd`** service.

      ```
      # service xinetd enable
      # service xinetd start
      ```

2. Configure the PXELinux environment.

   ```
   # mkdir -p /var/lib/tftpboot/{boot,pxelinux.cfg}
   # chown foreman-proxy /var/lib/tftpboot/{boot,pxelinux.cfg}
   # cp /usr/share/syslinux/{pxelinux.0,menu.c32,chain.c32}
   /var/lib/tftpboot/
   ```

3. Create the TFTP directory to be exported using NFS.

   ```
   # mkdir -p /exports/var/lib/tftpboot
   ```

4. Add the newly created mount point to the /etc/fstab file.

   ```
   /var/lib/tftpboot /exports/var/lib/tftpboot none bind,auto 0 0
   ```

5. Mount the file systems in **`/etc/fstab`**.

   ```
   # mount -a
   ```

6. Ensure the following lines are present in **`/etc/exports`**:

   ```
   /exports
   192.168.38.1(rw,async,no_root_squash,fsid=0,no_subtree_check)
   ```

   ```
   /exports/var/lib/tftpboot
   192.168.38.1(rw,async,no_root_squash,no_subtree_check,nohide)
   ```

   The first line is common to the DHCP configuration and therefore should already be present if the previous procedure was completed on this system.

7. Reload the NFS server.

```
# exportfs -rva
```

5.6.1. Configuring the Firewall for External Access to TFTP

Configuring the Firewall for External Access to the TFTP Service Using Red Hat Enterprise Linux 7

1. Configure the firewall (UDP on port 69).

   ```
   # firewall-cmd --add-port="69/udp" \
   && firewall-cmd --permanent --add-port="69/udp"
   ```

Configuring the Firewall for External Access to the TFTP Service Using Red Hat Enterprise Linux 7

1. Configure the firewall.

   ```
   # iptables -A INPUT -m state --state NEW -p tcp --dport 69 -j
   ACCEPT \
   && iptables-save > /etc/sysconfig/iptables
   ```

2. Ensure the iptables service is started and enabled.

   ```
   # service iptables start
   # chkconfig iptables on
   ```

3. Restore SELinux file contexts.

   ```
   # restorecon -RvF /var/lib/tftpboot/
   ```

5.7. CONFIGURING CAPSULE SERVER WITH EXTERNAL TFTP

1. Create the TFTP directory to prepare for NFS.

   ```
   # mkdir -p /mnt/nfs/var/lib/tftpboot
   ```

2. Add the newly created mount point to the **/etc/fstab** file.

   ```
   /mnt/nfs/var/lib/tftpboot /exports/mnt/nfs/var/lib/tftpboot none
   bind,auto 0 0
   ```

3. Mount the file systems in **/etc/fstab**.

   ```
   # mount -a
   ```

4. Add the following line in the **/etc/fstab** file:

```
192.168.38.2:/exports/var/lib/tftpboot /mnt/nfs/var/lib/tftpboot
nfs
rw,vers=3,auto,nosharecache,context="system_u:object_r:tftpdir_rw
_t:s0" 0 0
```

5. To enable TFTP support in foreman-proxy, edit the **`/usr/share/foreman-proxy/config/settings.d/tftp.yml`** file.

    ```
    :enabled: true
    :tftproot: /mnt/nfs/var/lib/tftpboot
    ```

6. If the TFTP service is running on a different server than the DHCP service, update the **`tftp_servername`** setting with the FQDN or IP address of that server.

    ```
    capsule-installer --tftp-servername=new_FQDN
    ```

    This updates all configuration files with the new value.

7. Log in to the Satellite Server web UI.

8. Go to **Infrastructure** > **Capsules**. Locate the appropriate Capsule Server and select **Refresh features** from the drop-down list. The TFTP feature should appear.

9. Associate the TFTP service with the appropriate subnets and domain.

CHAPTER 6. UPGRADING SATELLITE SERVER AND CAPSULE SERVER

The Satellite Server and Capsule Servers are upgraded independently. Upgrade the Satellite server first, and then upgrade any Capsule Servers. Satellite 6.1 Capsule Servers are not compatible with Satellite 6.2, and must be upgraded before attempting to synchronize any repositories. You must also manually upgrade Satellite clients to the new version of katello-agent after upgrading the Satellite Server and Capsule Servers.

Important

The Red Hat Satellite Server and Capsule Server versions must match. For example, a Satellite 6.1 Satellite Server cannot run a 6.2 Capsule Server and a Satellite 6.2 Server cannot run a 6.1 Capsule Server. Mismatching Satellite Server and Capsule Server versions will result in the Capsule Server failing silently.

Supported upgrade paths for Satellite 6.2 Beta:

- Perform a fresh installation of the Satellite 6.2 Satellite Server and Capsule Server on Red Hat Enterprise Linux 7

6.1. UPGRADING THE SATELLITE SERVER

6.1.1. Upgrading the Subscription Manifest

Before You Begin

Before you can upgrade Satellite Server, the Red Hat Satellite 6.2 Tools repository must be available in the subscription manifest. You must remove any repositories that are no longer required.

Upgrade the Subscription Manifest

1. Go to Red Hat Customer Portal and click **SUBSCRIPTIONS**.
2. In the Subscription Management Applications section, click **Satellite**.
3. Select the system with which the manifest is associated and click **Attach a subscription**.
4. For each subscription that you want to attach, select the check box for that subscription and specify the quantity of subscriptions to attach.
5. Click **Attach Selected**.

   It can take several minutes for all the subscriptions to attach. You can refresh the screen until you receive confirmation that the subscriptions are attached.
6. Click **Download Manifest** to generate an archive in `.zip` format.
7. Save the manifest file to a known location.
8. Upload the updated manifest to the Red Hat Satellite Server.

   a. Log in to the Satellite server.
   b. Select the organization that you want to associate with the subscription manifest.
   c. Click **Content** > **Red Hat Subscriptions** > **Manage Manifest**.
   d. Click **Actions** > **Browse**.

9. Select the manifest file to upload, and click **Upload**.

6.1.2. Upgrading Satellite Server

You should upgrade Satellite Server before upgrading any Capsule Servers. Before upgrading, you should be using the latest version of Satellite 6.1.

- You should have upgraded the subscription manifest.

Upgrade Satellite Server

1. Create a backup of all relevant databases.
   - On a virtual machine, take a snapshot.
   - On a physical machine, create a backup.
2. Update the operating system.

```
# yum update
```

3. If you have new packages, update Satellite Server. If you do not have any new packages, you can skip this step.

```
# foreman-installer --scenario katello --upgrade
```

4. Disable the repositories for the previous version of Satellite.

If you are upgrading from…	Run this command…
Satellite 6.1 on Red Hat Enterprise Linux 6	subscription-manager repos --disable rhel-server-6-satellite-6-beta-rpms
Satellite 6.1 on Red Hat Enterprise Linux 7	subscription-manager repos --disable rhel-server-7-satellite-6-beta-rpms

5. Enable the new repositories.
   - On Red Hat Enterprise Linux 6, run the following command:

```
# subscription-manager repos --enable rhel-6-server-satellite-
6-beta-rpms
```

- On Red Hat Enterprise Linux 7, run the following command:

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

6. If there are discovered hosts available, turn them off and delete all entries under the **`Discovered hosts`** page.

7. Stop services.

   ```
   katello-service stop
   ```

8. Confirm services have stopped.

   ```
   # katello-service status
   mongod is stopped
   qdrouterd is stopped
   qpidd is stopped
   celery init v10.0.
   Using configuration: /etc/default/pulp_workers,
   /etc/default/pulp_celerybeat
   pulp_celerybeat is stopped.
   celery init v10.0.
   Using config script: /etc/default/pulp_resource_manager
   node resource_manager is stopped...
   foreman-proxy is stopped
   tomcat6 is stopped                                   [  OK  ]
   ```

9. Restart the Mongo database.

   ```
   service-wait mongod start
   ```

   This database is required for upgrading the Pulp database.

10. Clear the repository cache.

    ```
    yum clean all
    ```

11. Update all packages.

    ```
    # yum update
    ```

12. If you want to see the changes that are applied when the upgrade occurs, run the installer with the **`--noop`** option.

    ```
    # foreman-installer --scenario katello --upgrade --noop
    ```

13. If you have made manual edits to DNS or DHCP configuration files and do not want the changes overwritten, run the following command:

    ```
    # foreman-installer --scenario katello --upgrade --capsule-dns-
    managed=false --capsule-dhcp-managed=false
    ```

14. Restart all services.

```
# katello-service restart
```

6.1.3. Removing Redundant Firewall Rules

Red Hat Satellite 6.2 does not use Elasticsearch and therefore firewall rules related to Elasticsearch can be removed. These are the lines with destination port 9200.

Removing Redundant Firewall Rules on Red Hat Enterprise Linux 6

1. List the firewall rules.

   ```
   # iptables -nL --line-numbers
   ```

2. Identify the following lines and remove them. Note that the chain name is OUTPUT and the line numbers might differ.

   ```
   Chain OUTPUT (policy ACCEPT)
   num  target   prot opt source         destination
   1    ACCEPT   tcp  --  0.0.0.0/0      0.0.0.0/0      tcp
   dpt:9200 owner UID match 496
   2    ACCEPT   tcp  --  0.0.0.0/0      0.0.0.0/0      tcp
   dpt:9200 owner UID match 0
   3    DROP     tcp  --  0.0.0.0/0      0.0.0.0/0      tcp
   dpt:9200
   ```

3. Remove the iptables rules.

   ```
   iptables -D <chain-name> <line-number>
   ```

   For example, to remove line 1 from the above output, enter a command as follows:

   ```
   # iptables -D OUTPUT 1
   ```

4. After removing the lines, save the changes.

   ```
   # service iptables save
   ```

5. Make sure the iptables service is started and enabled.

   ```
   # service iptables start
   # chkconfig iptables on
   ```

Removing Redundant Firewall Rules on Red Hat Enterprise Linux 7

1. List the IPv4 direct rules.

   ```
   # firewall-cmd --direct --get-rules ipv4 filter OUTPUT
   ```

2. List the IPv6 direct rules.

   ```
   # firewall-cmd --direct --get-rules ipv6 filter OUTPUT
   ```

3. Identify and remove the following lines, for both IPv4 and IPv6. Note that the chain name is OUTPUT and the first number is the priority.

```
0 -o lo -p tcp -m tcp --dport 9200 -m owner --uid-owner foreman -
j ACCEPT
0 -o lo -p tcp -m tcp --dport 9200 -m owner --uid-owner root -j
ACCEPT
1 -o lo -p tcp -m tcp --dport 9200 -j DROP
```

4. Remove the firewalld direct rules.

```
firewall-cmd --direct --remove-rule <inet_family> filter
<chain_name> rule
```

   Where <inet_family> is IPv4 or IPv6.

   For example, to remove the IPv4 lines above:

```
# firewall-cmd --direct --remove-rule ipv4 filter OUTPUT 0 -o lo
-p tcp -m tcp --dport 9200 -m owner --uid-owner foreman -j ACCEPT
\
&& firewall-cmd --direct --remove-rule ipv4 filter OUTPUT 0 -o lo
-p tcp -m tcp --dport 9200 -m owner --uid-owner root -j ACCEPT \
&& firewall-cmd --direct --remove-rule ipv4 filter OUTPUT 1 -o lo
-p tcp -m tcp --dport 9200 -j DROP
```

5. Repeat the commands for IPv6.

6. Ensure the firewall service is enabled and started.

```
# systemctl enable firewalld
# systemctl start firewalld
```

6.2. UPGRADING DISCONNECTED SATELLITE SERVER

6.2.1. Upgrading to Satellite Server 6.1

Before upgrading to Satellite Server 6.2 Beta, you must first upgrade to the latest minor version of Satellite 6.1. A direct upgrade to Satellite Server 6.2 Beta is not supported.

1. Download the Satellite Server 6.1.x ISO files, configure a local repository, and install the packages. For more information, see Downloading and Installing from a Disconnected Network.

2. Update the system.

```
# yum update
```

3. Restart all services.

```
# katello-service restart
```

6.2.2. Upgrading To Disconnected Satellite Server version 6.2 Beta

Before You Begin

- You should have upgraded to the latest minor release of Red Hat Satellite Server 6.1. Direct upgrade to Satellite Server 6.2 Beta is not supported.

Upgrade Disconnected Satellite Server

1. If there are discovered hosts available, turn them off and delete all entries under the **Discovered hosts** page.

2. Stop services.

   ```
   # katello-service stop
   ```

3. Confirm services have stopped.

   ```
   # katello-service status
   ```

   The following is displayed:

   ```
   mongod is stopped
   qdrouterd is stopped
   qpidd is stopped
   celery init v10.0.
   Using configuration: /etc/default/pulp_workers,
   /etc/default/pulp_celerybeat
   pulp_celerybeat is stopped.
   celery init v10.0.
   Using config script: /etc/default/pulp_resource_manager
   node resource_manager is stopped...
   foreman-proxy is stopped
   tomcat6 is stopped                                        [  OK  ]
   output truncated
   ```

4. Restart the Mongo database.

   This database is required for upgrading the Pulp database.

   ```
   # service-wait mongod start
   ```

5. Obtain the ISO file, mount it, and install the packages. For more information, see Downloading and Installing from a Disconnected Network.

   After executing successfully, the following message displays:

   ```
   Upgrade is complete. Please backup your data and run foreman-
   installer.
   ```

6. Create a backup of all relevant databases. For more information, see *Red Hat Satellite 6.2-Beta Server Administration Guide*.

7. Run the installer with the **`--upgrade`** option.

```
# foreman-installer --scenario katello --upgrade
```

a. (Optional) To view installer change log, run the installer with the **--noop** option.

```
# foreman-installer --scenario katello --noop
```

b. Upgrading overrides manual edits to DNS and DHCP configuration files. To avoid this, run the installer with the **--capsule-dns-managed=false** and **--capsule-dhcp-managed=false** options.

```
# foreman-installer --scenario katello --upgrade --capsule-dns-managed=false --capsule-dhcp-managed=false
```

The **foreman-installer** utility backs up files that it changes and logs them. If you need to restore the old file, you can run the following command:

```
# puppet filebucket -l restore /etc/dhcp/dhcpd.conf
622d9820b8e764ab124367c68f5fa3a1
```

8. Restart all services.

```
# katello-service restart
```

9. Update the Discovery template in the web UI.

a. Go to **Hosts** > **Provisioning templates**.

b. Select **PXELinux global default**.

c. In the **Template editor** dialog box, edit the **PXELinux global default** template discovery menu entry by inserting the following text at the end of the template.

```
LABEL discovery
MENU LABEL Satellite 6 Discovery
MENU DEFAULT
KERNEL boot/fdi-image-rhel_7-vmlinuz
APPEND initrd=boot/fdi-image-rhel_7-img rootflags=loop
root=live:/fdi.iso rootfstype=auto ro rd.live.image
acpi=force rd.luks=0 rd.md=0 rd.dm=0 rd.lvm=0 rd.bootif=0
rd.neednet=0 nomodeset
proxy.url=https://SATELLITE_CAPSULE_URL:9090
proxy.type=proxy
IPAPPEND 2
```

- The **proxy.type** option can be either **proxy** or **foreman**. For **proxy**, all communication goes through the Capsule. For **foreman**, the communication goes directly to Satellite Server.
- The **proxy.url** specifies the URL of the Satellite Capsule or Server. Both HTTP and HTTPS protocols are supported.

6.3. UPGRADING CAPSULE SERVERS

Before You Begin

- You should have upgraded to the latest minor release of Red Hat Satellite Server 6.1. Direct upgrade from earlier minor versions is not supported.

Upgrading Capsule Servers

1. Update the operating system.

   ```
   # yum update
   ```

2. Disable the repositories for the previous version of Satellite.

If you are upgrading from…	Run this command…
Satellite 6.1 on Red Hat Enterprise Linux 7	subscription-manager repos --disable rhel-server-7-satellite-capsule-6-beta-rpms
Satellite 6.1 on Red Hat Enterprise Linux 6	subscription-manager repos --disable rhel-server-6-satellite-capsule-6-beta-rpms

3. Enable the new repositories.

   - On Red Hat Enterprise Linux 7, run the following command:

     ```
     # subscription-manager repos --enable rhel-7-server-satellite-capsule-6-beta-rpms
     ```

   - On Red Hat Enterprise Linux 6, run the following command:

     ```
     # subscription-manager repos --enable rhel-6-server-satellite-capsule-6-beta-rpms
     ```

4. If there are discovered hosts available, turn them off and delete all entries under the **`Discovered hosts`** page.

5. Stop services to prevent dependency errors during the database migration.

   ```
   # for i in qpidd pulp_workers pulp_celerybeat
   pulp_resource_manager httpd; do service $i stop; done
   ```

6. Clear the repository cache and update all packages.

   ```
   # yum clean all
   # yum update
   ```

7. On the Satellite Server, generate an archive with new certificates.

   ```
   # capsule-certs-generate --capsule-fqdn "mycapsule.example.com" --certs-tar "mycapsule.example.com-certs.tar"
   ```

–

You should replace mycapsule.example.com with the fully qualified domain name of the Capsule Server.

8. Copy the archive file to the Capsule Server.

```
# scp mycapsule.example.com-certs.tar mycapsule.example.com
```

9. 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
```

10. 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.

11. Restart the foreman-proxy component on the Capsule Server.

```
# service foreman-proxy restart
```

12. Run the installer with the **`--upgrade`** option.

```
# foreman-installer --scenario capsule --upgrade\
                    --certs-tar mycapsule.example.com-certs.tar\
                    --certs-update-all --regenerate --deploy
```

13. If you have made manual edits to DNS and DHCP configuration, run the installer with the following options:

```
# foreman-installer --scenario capsule --upgrade --dns-
managed=false --dhcp-managed=false
```

14. Upgrade the foreman-discovery package on Satellite Server and turn on the hosts that were shut down prior to the upgrade.

6.4. UPGRADING DISCOVERY ON CAPSULE SERVERS

1. Verify that all relevant packages are current on the Satellite Server.

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

2. Restart the Satellite Server, if applicable.

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 web UI, go to **Infrastructure** > **Capsules** and verify that the relevant proxy lists the Discovery feature. Click **Refresh features** if necessary.

6. Go to **Infrastructure** > **Subnets** and select the required Smart Proxy for each subnet on which you want to use discovery, and verify that it is connected to the Discovery Proxy.

Update the Discovery template in the web UI.

a. Go to **Hosts** > **Provisioning Templates**.

b. Select **PXELinux global default**.

c. In the **Template editor** dialog box, edit the **PXELinux global default** template discovery menu entry by inserting the following text at the end of the template.

```
LABEL discovery
MENU LABEL Satellite 6 Discovery
MENU DEFAULT
KERNEL boot/fdi-image-rhel_7-vmlinuz
APPEND initrd=boot/fdi-image-rhel_7-img rootflags=loop
root=live:/fdi.iso rootfstype=auto ro rd.live.image acpi=force
rd.luks=0 rd.md=0 rd.dm=0 rd.lvm=0 rd.bootif=0 rd.neednet=0
nomodeset proxy.url=https://SATELLITE_CAPSULE_URL:9090
proxy.type=proxy
IPAPPEND 2
```

- The **`proxy.type`** option can be either **`proxy`** or **`foreman`**. For **`proxy`**, all communication goes through the Capsule. For **`foreman`**, the communication goes directly to Satellite Server.

- The **`proxy.url`** specifies the URL of the Satellite Capsule or Server. Both HTTP and HTTPS protocols are supported.

- You can omit the proxy.url option and determine the Capsule DNS name from its SRV record. This is useful if there are multiple discovery subnets. For more information, see the *Red Hat Satellite 6.2-Beta Host Configuration Guide*.

6.5. UPGRADING SATELLITE CLIENTS

You must manually upgrade to the new version of **`katello-agent`** so that your client is compatible with Satellite Server.

Before You Begin

- You must have upgraded your Satellite Server.

Upgrade Satellite Clients

1. Log in to the client system and enable the Satellite tools repository.

   - On Red Hat Enterprise Linux 7, run the following command:

```
# subscription-manager repos --enable=rhel-7-server-satellite-tools-6-beta-rpms
```

- On Red Hat Enterprise Linux 6, run the following command:

```
# subscription-manager repos --enable=rhel-6-server-satellite-tools-6-beta-rpms
```

2. Synchronize the repository.

```
# hammer repository synchronize --id ID
```

3. Upgrade the package.

```
# yum upgrade katello-agent
```

CHAPTER 7. UNINSTALLING SATELLITE SERVER AND CAPSULE SERVER

If you no longer need Satellite Server or Capsule Server, you can uninstall them.

7.1. UNINSTALLING SATELLITE SERVER

Uninstalling Satellite Server and Capsule Server erases all applications used on the target system. If you use any applications or application data for purposes other than Satellite Server, you should back up the information before the removal process.

Before you Begin

The uninstall script issues two warnings, requiring confirmation before removing all packages and configuration files in the system.

> **Warning**
>
> This script will erase many packages and config files. Important packages such as the following will be removed:

- httpd (apache)
- mongodb
- tomcat6
- puppet
- ruby
- rubygems
- All Katello and Foreman Packages

Uninstall Satellite Server

1. Uninstall Satellite Server.

```
# katello-remove
```

```
Once these packages and configuration files are removed there is
no going back.
If you use this system for anything other than Katello and
Foreman you probably
do not want to execute this script.
Read the source for a list of what is removed.  Are you
sure(Y/N)? y
ARE YOU SURE?: This script permanently deletes data and
```

```
configuration.
Read the source for a list of what is removed.  Type [remove] to
continue? remove
Shutting down Katello services...
```

7.2. UNINSTALLING CAPSULE SERVERS

Uninstalling Capsule Server erases all applications used on the target system. If you use any applications or application data for purposes other than Satellite Server, you should back up the information before the removal process.

Before you Begin

The uninstall script issues two warnings, requiring confirmation before removing all packages and configuration files in the system.

> **Warning**
>
> This script erases packages and config files. Important packages such as the following will be removed:

- httpd (apache)
- mongodb
- tomcat6
- puppet
- ruby
- rubygems
- All Katello and Foreman Packages

Uninstall Capsule Server

1. Uninstall Capsule Server.

   ```
   $ capsule-remove
   ```

   The following message displays.

   ```
   Once these packages and configuration files are removed there is
   no going back.
   If you use this system for anything other than Katello and
   Foreman you probably
   do not want to execute this script.
   Read the source for a list of what is removed.  Are you
   sure(Y/N)? y
   ARE YOU SURE?: This script permanently deletes data and
   ```

```
configuration.
Read the source for a list of what is removed.  Type [remove] to
continue? remove
Shutting down Katello services...
```

CHAPTER 8. WHERE TO FIND MORE INFORMATION

At the end of the initial installation and setup, you can perform additional configuration and set up your Satellite environment. You can use the following Satellite documentation resources to assist you:

- *Hammer CLI Guide*
- *Server Administration Guide*
- *Host Configuration Guide*
- *Content Management Guide*
- *Puppet Guide*
- *Virtual Instances Guide*

APPENDIX A. LARGE DEPLOYMENT CONSIDERATIONS

With more than 225 content hosts, the qpidd message broker can reach several system-level limits, resulting in Satellite's failure to operate. To avoid this, one or more of these limits must be increased before deploying a large number of content hosts.

You can use the following table to determine which values you need to change based on the number of content hosts you deploy.

Table A.1. Large Deployment Limits

Number of Content Hosts	Client Connections	File Descriptors	Parallel Asynchronous I/O Operations	Concurrent Locks	Memory Map Areas
More than 225	✔				
More than 500	✔	✔			
More than 1900	✔	✔	✔		
More than 30,000	✔	✔	✔	✔	
More than 32,900	✔	✔	✔	✔	✔

Increasing the Maximum Number of Client Connections

1. Calculate the new value for the maximum number of client connections.

   ```
   (number_of_content_hosts x 2) + 100
   ```

   For example, a deployment with 300 content hosts requires at least 700 connections.

2. Use the calculated value in **`/etc/qpid/qpidd.conf`**.

   ```
   max-connections=value
   ```

Increasing the Maximum Number of File Descriptors

1. Calculate the new value for the maximum number of file descriptors.

```
(number_of_content_hosts x 4) + 500
```

For example, a deployment with 600 content hosts requires 2900 file descriptors.

2. Update the appropriate configuration file with the new value.

   a. On Red Hat Enterprise Linux 6, add the following line to **/etc/security/limits.conf**:

   ```
   qpidd x nofile value
   ```

   b. On Red Hat Enterprise Linux 7, create the **/etc/systemd/system/qpidd.service.d/qpidd.conf** file and insert the following text to it:

   ```
   [Service]
   LimitNOFILE=value
   ```

   To apply changes to the unit, run the following commands:

   ```
   systemctl daemon-reload
   systemctl restart qpidd.service
   ```

Increasing the Maximum Number of Parallel Asynchronous I/O Operations

1. Calculate the new value for the maximum number of parallel asynchronous I/O operations.

   ```
   33 x number_of_content_hosts
   ```

2. Use the calculated value in **/etc/sysctl.conf**.

   ```
   fs.aio-max-nr=value
   ```

3. Reload the setting.

   ```
   # sysctl -p
   ```

Increasing the Maximum Number of Concurrent Locks

1. Locate the directory in which the **exchanges.db** file is stored.

   ```
   # find /var/lib/qpidd -name exchanges.db
   /var/lib/qpidd/qls/dat/exchanges.db
   ```

2. To increase the limit of concurrent locks, create a configuration file called **DB_CONFIG** in the directory where the **exchanges.db** file is stored.

   This file must be owned and readable by the qpidd user.

3. Add the following content to the **DB_CONFIG** file:

```
set_lk_max_locks 10000
set_lk_max_objects 10000
```

Increasing the Maximum Number of Memory Map Areas

With more than 32,900 content hosts, qpidd reaches the kernel limit of maximum number of memory map areas per process. This occurs only on Red Hat Enterprise Linux 7.

1. Increase the limit by adding the following line to **`/etc/sysctl.conf`**:

   ```
   vm.max_map_count = 655300
   ```

2. Reload the setting.

   ```
   # sysctl -p
   ```

3. Restart qpidd to apply the changes.

If you are using…	Run this command…
Red Hat Enterprise Linux 7	systemctl restart qpidd
Red Hat Enterprise Linux 6	service qpidd restart

APPENDIX B. CAPSULE SERVER SCALABILITY CONSIDERATIONS

The maximum number of Capsule Servers that the Satellite Server can support has no fixed limit. The tested limit is 14 Capsule Servers with 2 vCPUs on a Satellite Server with Red Hat Enterprise Linux 6.6 and 7 hosts. However, scalability is highly variable, especially when managing Puppet clients.

Capsule Server scalability when managing Puppet clients depends on the number of CPUs, the run-interval distribution, and the number of Puppet managed resources. The Capsule Server has a limitation of 100 concurrent Puppet agents running at any single point in time. Running more than 100 concurrent Puppet agents results in a 503 HTTP error.

For example, assuming that Puppet agent runs are evenly distributed with less than 100 concurrent Puppet agents running at any single point during a run-interval, a Capsule Server with 4 CPUs has a maximum of 1250-1600 Puppet clients with a moderate workload of 10 Puppet classes assigned to each Puppet client. Depending on the number of Puppet clients required, the Satellite installation can scale out the number of Capsule Servers to support them.

If you want to scale your Capsule Server when managing Puppet clients, the following assumptions are made:

- There are no external Puppet clients reporting directly to the Satellite 6 integrated Capsule.
- All other Puppet clients report directly to an external Capsule.
- There is an evenly distributed run-interval of all Puppet agents.

Note

Deviating from the even distribution increases the risk of filling the passenger request queue. The limit of 100 concurrent requests applies.

The following table describes the scalability limits using the recommended 4 CPUs with Red Hat Enterprise Linux 7.

Table B.1. Puppet Scalability Using 4 CPUs with Red Hat Enterprise Linux 7 (Recommended)

Puppet Managed Resources per Host	Run-Interval Distribution
1	1700-1450
10	1500-1250
20	850-700

The following table describes the scalability limits using the minimum 2 CPUs with Red Hat Enterprise Linux 7.

Table B.2. Puppet Scalability Using 2 CPUs with with Red Hat Enterprise Linux 7

Puppet Managed Resources per Host	Run-Interval Distribution
1	3000-2500
10	2400-2000
20	1700-1400

The following table describes the scalability limits using the recommended 4 CPUs with Red Hat Enterprise Linux 6.

Table B.3. Puppet Scalability Using 4 CPUs with Red Hat Enterprise Linux 6 (Recommended)

Puppet Managed Resources per Host	Run-Interval Distribution
1	2250-1875
10	1600-1250
20	700-560

The following table describes the scalability limits using the minimum 2 CPUs.

Table B.4. Puppet Scalability Using 2 CPUs with Red Hat Enterprise Linux 6

Puppet Managed Resources per Host	Run-Interval Distribution
1	Not tested
10	1020-860

Puppet Managed Resources per Host	Run-Interval Distribution
20	375-330