Abstract

This guide describes how to install Red Hat Satellite from a connected network, perform initial configuration, and configure external services.
# Table of Contents

**CHAPTER 1. PREPARING YOUR ENVIRONMENT FOR INSTALLATION** ........................................... 4
  1.1. SYSTEM REQUIREMENTS ........................................... 4
  1.2. STORAGE REQUIREMENTS ........................................... 5
    1.2.1. Red Hat Enterprise Linux 7 ........................................... 5
  1.3. STORAGE GUIDELINES ........................................... 5
  1.4. SUPPORTED OPERATING SYSTEMS ........................................... 7
  1.5. SUPPORTED BROWSERS ........................................... 7
  1.6. PORTS AND FIREWALLS REQUIREMENTS ........................................... 7
  1.7. ENABLING CONNECTIONS FROM A CLIENT TO SATELLITE SERVER ........................................... 12
  1.8. VERIFYING FIREWALL SETTINGS ........................................... 13
  1.9. VERIFYING DNS RESOLUTION ........................................... 13

**CHAPTER 2. PREPARING YOUR ENVIRONMENT FOR SATELLITE INSTALLATION IN AN IPV6 NETWORK** ........ 15
  2.1. LIMITATIONS OF SATELLITE INSTALLATION IN AN IPV6 NETWORK ........................................... 15
  2.2. REQUIREMENTS FOR SATELLITE INSTALLATION IN AN IPV6 NETWORK ........................................... 15

**CHAPTER 3. INSTALLING SATELLITE SERVER** ................................................................. 16
  3.1. REGISTERING TO RED HAT SUBSCRIPTION MANAGEMENT ........................................... 16
  3.2. ATTACHING THE SATELLITE INFRASTRUCTURE SUBSCRIPTION ........................................... 16
  3.3. CONFIGURING REPOSITORIES ........................................... 18
    3.3.1. Red Hat Enterprise Linux 7 ........................................... 18
  3.4. INSTALLING SATELLITE SERVER PACKAGES ........................................... 18
    3.4.1. Red Hat Enterprise Linux 7 ........................................... 18
  3.5. SYNCHRONIZING THE SYSTEM CLOCK WITH CHRONYD ........................................... 19
  3.6. INSTALLING THE SOS PACKAGE ON THE BASE OPERATING SYSTEM ........................................... 19
  3.7. CONFIGURING SATELLITE SERVER ........................................... 19
    3.7.1. Configuring Satellite ........................................... 20
  3.8. IMPORTING A SUBSCRIPTION MANIFEST INTO SATELLITE SERVER ........................................... 21

**CHAPTER 4. PERFORMING ADDITIONAL CONFIGURATION ON SATELLITE SERVER** ........................... 22
  4.1. USING RED HAT INSIGHTS WITH SATELLITE SERVER ........................................... 22
  4.2. DISABLING REGISTRATION TO RED HAT INSIGHTS ........................................... 22
  4.3. ENABLING THE SATELLITE TOOLS 6.10 REPOSITORY ........................................... 23
  4.4. SYNCHRONIZING THE SATELLITE TOOLS 6.10 REPOSITORY ........................................... 23
  4.5. CONFIGURING SATELLITE FOR UEFI HTTP BOOT PROVISIONING IN AN IPV6 NETWORK ........................................... 24
  4.6. CONFIGURING SATELLITE SERVER WITH AN HTTP PROXY ........................................... 24
    4.6.1. Adding a Default HTTP Proxy to Satellite ........................................... 25
    4.6.2. Configuring the HTTP Proxy to Connect to Red Hat CDN ........................................... 25
    4.6.3. Configuring SELinux to Ensure Access to Satellite on Custom Ports ........................................... 26
    4.6.4. Using an HTTP Proxy for all Satellite HTTP Requests ........................................... 27
    4.6.5. Excluding Hosts from Receiving Proxied Requests ........................................... 27
    4.6.6. Resetting the HTTP Proxy ........................................... 28
  4.7. ENABLING POWER MANAGEMENT ON MANAGED HOSTS ........................................... 28
  4.8. CONFIGURING DNS, DHCP, AND TFTP ON SATELLITE SERVER ........................................... 28
  4.9. DISABLING DNS, DHCP, AND TFTP FOR UNMANAGED NETWORKS ........................................... 29
  4.10. CONFIGURING SATELLITE SERVER FOR OUTGOING EMAILS ........................................... 30
  4.11. CONFIGURING AN ALTERNATE CNAME FOR SATELLITE ........................................... 32
    4.11.1. Configuring Satellite with an Alternate CNAME ........................................... 32
    4.11.2. Configuring Hosts to Use an Alternate Satellite CNAME for Content Management ........................................... 33
  4.12. CONFIGURING SATELLITE SERVER WITH A CUSTOM SSL CERTIFICATE ........................................... 33
    4.12.1. Creating a Custom SSL Certificate for Satellite Server ........................................... 34
    4.12.2. Deploying a Custom SSL Certificate to Satellite Server ........................................... 35
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.12.3. Deploying a Custom SSL Certificate to Hosts</td>
<td>37</td>
</tr>
<tr>
<td>4.13. USING EXTERNAL DATABASES WITH SATELLITE</td>
<td>37</td>
</tr>
<tr>
<td>4.13.1. PostgreSQL as an External Database Considerations</td>
<td>37</td>
</tr>
<tr>
<td>4.13.2. Preparing a Host for External Databases</td>
<td>38</td>
</tr>
<tr>
<td>4.13.3. Installing PostgreSQL</td>
<td>38</td>
</tr>
<tr>
<td>4.13.4. Configuring Satellite to use External Databases</td>
<td>40</td>
</tr>
<tr>
<td>4.14. TUNING SATELLITE SERVER WITH PREDEFINED PROFILES</td>
<td>40</td>
</tr>
<tr>
<td>CHAPTER 5. CONFIGURING SATELLITE SERVER WITH EXTERNAL SERVICES</td>
<td>43</td>
</tr>
<tr>
<td>5.1. CONFIGURING SATELLITE SERVER WITH EXTERNAL DNS</td>
<td>43</td>
</tr>
<tr>
<td>5.2. CONFIGURING SATELLITE SERVER WITH EXTERNAL DHCP</td>
<td>44</td>
</tr>
<tr>
<td>5.2.1. Configuring an External DHCP Server to Use with Satellite Server</td>
<td>44</td>
</tr>
<tr>
<td>5.2.2. Configuring Satellite Server with an External DHCP Server</td>
<td>47</td>
</tr>
<tr>
<td>5.3. CONFIGURING SATELLITE SERVER WITH EXTERNAL TFTP</td>
<td>48</td>
</tr>
<tr>
<td>5.4. CONFIGURING SATELLITE SERVER WITH EXTERNAL IDM DNS</td>
<td>49</td>
</tr>
<tr>
<td>5.4.1. Configuring Dynamic DNS Update with GSS-TSIG Authentication</td>
<td>50</td>
</tr>
<tr>
<td>5.4.2. Configuring Dynamic DNS Update with TSIG Authentication</td>
<td>53</td>
</tr>
<tr>
<td>5.4.3. Reverting to Internal DNS Service</td>
<td>55</td>
</tr>
<tr>
<td>APPENDIX A. APPLYING CUSTOM CONFIGURATION TO RED HAT SATELLITE</td>
<td>57</td>
</tr>
<tr>
<td>APPENDIX B. RESTORING MANUAL CHANGES OVERWRITTEN BY A PUPPET RUN</td>
<td>58</td>
</tr>
</tbody>
</table>
CHAPTER 1. PREPARING YOUR ENVIRONMENT FOR INSTALLATION

Before you install Satellite, ensure that your environment meets the following requirements.

1.1. SYSTEM REQUIREMENTS

The following requirements apply to the networked base operating system:

- x86_64 architecture
- The latest version of Red Hat Enterprise Linux 7 Server
- 4-core 2.0 GHz CPU at a minimum
- A minimum of 20 GB RAM is required for Satellite Server to function. In addition, a minimum of 4 GB RAM of swap space is also recommended. Satellite running with less RAM than the minimum value might not operate correctly.
- A unique host name, which can contain lower-case letters, numbers, dots (.) and hyphens (-)
- A current Red Hat Satellite subscription
- Administrative user (root) access
- A system umask of 0022
- Full forward and reverse DNS resolution using a fully-qualified domain name

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

Satellite Server must be installed on a freshly provisioned system that serves no other function except to run Satellite Server. The freshly provisioned system must not have the following users provided by external identity providers to avoid conflicts with the local users that Satellite Server creates:

- apache
- foreman
- foreman-proxy
- postgres
- pulp
- puppet
- puppetserver
- qdrouterd
- qpidd
- tomcat

Certified hypervisors
Satellite Server is fully supported on both physical systems and virtual machines that run on hypervisors that are supported to run Red Hat Enterprise Linux. For more information about certified hypervisors, see Which hypervisors are certified to run Red Hat Enterprise Linux?.

**SELinux Mode**

SELinux must be enabled, either in enforcing or permissive mode. Installation with disabled SELinux is not supported.

**FIPS Mode**

You can install Satellite Server on a Red Hat Enterprise Linux system that is operating in FIPS mode. For more information, see Enabling FIPS Mode in the Red Hat Enterprise Linux Security Guide.

### 1.2. STORAGE REQUIREMENTS

- **Red Hat Enterprise Linux 7**

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

The runtime size was measured with Red Hat Enterprise Linux 6, 7, and 8 repositories synchronized.

#### 1.2.1. Red Hat Enterprise Linux 7

<table>
<thead>
<tr>
<th>Directory</th>
<th>Installation Size</th>
<th>Runtime Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>/var/log/</td>
<td>10 MB</td>
<td>10 GB</td>
</tr>
<tr>
<td>/var/opt/rh/rh-postgresql12/lib/pgsql</td>
<td>100 MB</td>
<td>20 GB</td>
</tr>
<tr>
<td>/usr</td>
<td>3 GB</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>/opt</td>
<td>3 GB</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>/opt/puppetlabs</td>
<td>500 MB</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>/var/lib/pulp/</td>
<td>1 MB</td>
<td>300 GB</td>
</tr>
<tr>
<td>/var/lib/qpidd/</td>
<td>25 MB</td>
<td>Not Applicable</td>
</tr>
</tbody>
</table>

### 1.3. STORAGE GUIDELINES

Consider the following guidelines when installing Satellite Server to increase efficiency.

- If you mount the /tmp directory as a separate file system, you must use the `exec` mount option in the /etc/fstab file. If /tmp is already mounted with the `noexec` option, you must change the option to `exec` and re-mount the file system. This is a requirement for the puppetserver service to work.
Because most Satellite Server data is stored in the /var directory, mounting /var on LVM storage can help the system to scale.

The /var/lib/qpidd/ directory uses slightly more than 2 MB per Content Host managed by the goferd service. For example, 10,000 Content Hosts require 20 GB of disk space in /var/lib/qpidd/.

Use high-bandwidth, low-latency storage for the /var/lib/pulp/ directories. As Red Hat Satellite has many operations that are I/O intensive, using high latency, low-bandwidth storage causes performance degradation. Ensure your installation has a speed in the range 60 - 80 Megabytes per second.

You can use the fio tool to get this data. See the Red Hat Knowledgebase solution Impact of Disk Speed on Satellite Operations for more information on using the fio tool.

File System Guidelines

- Do not use the GFS2 file system as the input-output latency is too high.

Log File Storage

Log files are written to /var/log/messages/, /var/log/httpd/, and /var/lib/foreman-proxy/openscap/content/. You can manage the size of these files using logrotate. For more information, see Log Rotation in the Red Hat Enterprise Linux 7 System Administrator’s Guide.

The exact amount of storage you require for log messages depends on your installation and setup.

SELinux Considerations for NFS Mount

When the /var/lib/pulp directory is mounted using an NFS share, SELinux blocks the synchronization process. To avoid this, specify the SELinux context of the /var/lib/pulp directory in the file system table by adding the following lines to /etc/fstab:

```
nfs.example.com:/nfsshare  /var/lib/pulp  nfs  context="system_u:object_r:var_lib_t:s0"  1 2
```

If NFS share is already mounted, remount it using the above configuration and enter the following command:

```
# restorecon -R /var/lib/pulp
```

Duplicated Packages

Packages that are duplicated in different repositories are only stored once on the disk. Additional repositories containing duplicate packages require less additional storage. The bulk of storage resides in the /var/lib/pulp/ directory. These end points are not manually configurable. Ensure that storage is available on the /var file system to prevent storage problems.

Software Collections

Software collections are installed in the /opt/rh/ and /opt/theforeman/ directories.

Write and execute permissions by the root user are required for installation to the /opt directory.

Symbolic links

You cannot use symbolic links for /var/lib/pulp/.
Synchronized RHEL ISO

If you plan to synchronize RHEL content ISOs to Satellite, note that all minor versions of Red Hat Enterprise Linux also synchronize. You must plan to have adequate storage on your Satellite to manage this.

1.4. SUPPORTED OPERATING SYSTEMS

You can install the operating system from a disc, local ISO image, kickstart, or any other method that Red Hat supports. Red Hat Satellite Server is supported only on the latest versions of Red Hat Enterprise Linux 7 Server that is available at the time when Satellite Server 6.10 is installed. Previous versions of Red Hat Enterprise Linux including EUS or z-stream are not supported.

The following operating systems are supported by the installer, have packages, and are tested for deploying Satellite:

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Architecture</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red Hat Enterprise Linux 7</td>
<td>x86_64 only</td>
<td></td>
</tr>
</tbody>
</table>

Before you install Satellite, apply all operating system updates if possible.

Red Hat Satellite Server requires a Red Hat Enterprise Linux installation with the @Base package group with no other package-set modifications, and without third-party configurations or software not directly necessary for the direct operation of the server. This restriction includes hardening and other non-Red Hat security software. If you require such software 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.

Install Satellite Server on a freshly provisioned system.

Red Hat does not support using the system for anything other than running Satellite Server.

1.5. SUPPORTED BROWSERS

Satellite supports recent versions of Firefox and Google Chrome browsers.

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

1.6. PORTS AND FIREWALLS REQUIREMENTS

For the components of Satellite architecture to communicate, ensure that the required network ports are open and free on the base operating system. You must also ensure that the required network ports are open on any network-based firewalls.

Use this information to configure any network-based firewalls. Note that some cloud solutions must be specifically configured to allow communications between machines because they isolate machines similarly to network-based firewalls. If you use an application-based firewall, ensure that the application-based firewall permits all applications that are listed in the tables and known to your firewall. If possible, disable the application checking and allow open port communication based on the protocol.
Integrated Capsule

Satellite Server has an integrated Capsule and any host that is directly connected to Satellite Server is a Client of Satellite in the context of this section. This includes the base operating system on which Capsule Server is running.

Clients of Capsule

Hosts which are clients of Capsules, other than Satellite’s integrated Capsule, do not need access to Satellite Server. For more information on Satellite Topology, see Capsule Networking in Planning for Red Hat Satellite 6.

Required ports can change based on your configuration.

The following tables indicate the destination port and the direction of network traffic:

Table 1.3. Satellite Server incoming traffic

<table>
<thead>
<tr>
<th>Destination Port</th>
<th>Protocol</th>
<th>Service</th>
<th>Source</th>
<th>Required For</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>53</td>
<td>TCP and UDP</td>
<td>DNS</td>
<td>DNS Servers and clients</td>
<td>Name resolution</td>
<td>DNS (optional)</td>
</tr>
<tr>
<td>67</td>
<td>UDP</td>
<td>DHCP</td>
<td>Client</td>
<td>Dynamic IP</td>
<td>DHCP (optional)</td>
</tr>
<tr>
<td>69</td>
<td>UDP</td>
<td>TFTP</td>
<td>Client</td>
<td>TFTP Server</td>
<td>TFTP Server (optional)</td>
</tr>
<tr>
<td>443</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Capsule</td>
<td>Red Hat Satellite API</td>
<td>Communication from Capsule</td>
</tr>
<tr>
<td>443, 80</td>
<td>TCP</td>
<td>HTTPS, HTTP</td>
<td>Client</td>
<td>Content Retrieval</td>
<td>Content</td>
</tr>
<tr>
<td>443, 80</td>
<td>TCP</td>
<td>HTTPS, HTTP</td>
<td>Capsule</td>
<td>Content Retrieval</td>
<td>Content</td>
</tr>
<tr>
<td>443, 80</td>
<td>TCP</td>
<td>HTTPS, HTTP</td>
<td>Client</td>
<td>Content Host Registration</td>
<td>Capsule CA RPM installation</td>
</tr>
<tr>
<td>443</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Red Hat Satellite</td>
<td>Content Mirroring</td>
<td>Management</td>
</tr>
<tr>
<td>443</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Red Hat Satellite</td>
<td>Capsule API</td>
<td>Smart Proxy functionality</td>
</tr>
<tr>
<td>5646</td>
<td>TCP</td>
<td>AMQP</td>
<td>Capsule</td>
<td>Katello agent</td>
<td>Forward message to Qpid dispatch router on Satellite (optional)</td>
</tr>
<tr>
<td>Destination Port</td>
<td>Protocol</td>
<td>Service</td>
<td>Destination</td>
<td>Required For</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>----------</td>
<td>-------------</td>
<td>--------------</td>
<td>-----------------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>5910 - 5930</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Browsers</td>
<td>Compute Resource’s virtual console</td>
<td></td>
</tr>
<tr>
<td>8000</td>
<td>TCP</td>
<td>HTTP</td>
<td>Client</td>
<td>Provisioning templates</td>
<td>Template retrieval for client installers, iPXE or UEFI HTTP Boot</td>
</tr>
<tr>
<td>8000</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Client</td>
<td>PXE Boot</td>
<td>Installation</td>
</tr>
<tr>
<td>8140</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Client</td>
<td>Puppet agent</td>
<td>Client updates (optional)</td>
</tr>
<tr>
<td>8443</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Client</td>
<td>Content Host registration</td>
<td>Initiation, Uploading facts, Sending installed packages and traces</td>
</tr>
<tr>
<td>9090</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Client</td>
<td>OpenSCAP</td>
<td>Configure Client</td>
</tr>
<tr>
<td>9090</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Discovered Node</td>
<td>Discovery</td>
<td>Host discovery and provisioning</td>
</tr>
<tr>
<td>9090</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Red Hat Satellite</td>
<td>Capsule API</td>
<td>Capsule functionality</td>
</tr>
</tbody>
</table>

Any managed host that is directly connected to Satellite Server is a client in this context because it is a client of the integrated Capsule. This includes the base operating system on which a Capsule Server is running.

A DHCP Capsule performs ICMP ping or TCP echo connection attempts to hosts in subnets with DHCP IPAM set to find out if an IP address considered for use is free. This behavior can be turned off using `satellite-installer --foreman-proxy-dhcp-ping-free-ip=false`.

**Table 1.4. Satellite Server outgoing traffic**

<table>
<thead>
<tr>
<th>Destination Port</th>
<th>Protocol</th>
<th>Service</th>
<th>Destination</th>
<th>Required For</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICMP</td>
<td>ping</td>
<td></td>
<td>Client</td>
<td>DHCP</td>
<td>Free IP checking (optional)</td>
</tr>
<tr>
<td>7</td>
<td>TCP</td>
<td>echo</td>
<td>Client</td>
<td>DHCP</td>
<td>Free IP checking (optional)</td>
</tr>
<tr>
<td>Port</td>
<td>Protocol</td>
<td>Service</td>
<td>Target host</td>
<td>Remote execution</td>
<td>Comment</td>
</tr>
<tr>
<td>------</td>
<td>----------</td>
<td>---------</td>
<td>-------------</td>
<td>------------------</td>
<td>---------</td>
</tr>
<tr>
<td>22</td>
<td>TCP</td>
<td>SSH</td>
<td>Target host</td>
<td>Remote execution</td>
<td>Run jobs</td>
</tr>
<tr>
<td>22, 16514</td>
<td>TCP</td>
<td>SSH, SSH/TLS</td>
<td>Compute Resource</td>
<td>Satellite originated communications, for compute resources in libvirt</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>TCP and UDP</td>
<td>DNS</td>
<td>DNS Server on the Internet</td>
<td>DNS Server</td>
<td>Resolve DNS records (optional)</td>
</tr>
<tr>
<td>53</td>
<td>TCP and UDP</td>
<td>DNS</td>
<td>DNS Server</td>
<td>Capsule DNS</td>
<td>Validation of DNS conflicts (optional)</td>
</tr>
<tr>
<td>53</td>
<td>TCP and UDP</td>
<td>DNS</td>
<td>DNS Server</td>
<td>Orchestration</td>
<td>Validation of DNS conflicts</td>
</tr>
<tr>
<td>68</td>
<td>UDP</td>
<td>DHCP</td>
<td>Client</td>
<td>Dynamic IP</td>
<td>DHCP (optional)</td>
</tr>
<tr>
<td>80</td>
<td>TCP</td>
<td>HTTP</td>
<td>Remote repository</td>
<td>Content Sync</td>
<td>Remote yum repository</td>
</tr>
<tr>
<td>389, 636</td>
<td>TCP</td>
<td>LDAP, LDAPS</td>
<td>External LDAP Server</td>
<td>LDAP</td>
<td>LDAP authentication, necessary only if external authentication is enabled. The port can be customized when LDAPAuthSource is defined</td>
</tr>
<tr>
<td>443</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Satellite</td>
<td>Capsule</td>
<td>Capsule Configuration management Template retrieval OpenSCAP Remote Execution result upload</td>
</tr>
<tr>
<td>443</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Amazon EC2, Azure, Google GCE</td>
<td>Compute resources</td>
<td>Virtual machine interactions (query/create/destroy) (optional)</td>
</tr>
<tr>
<td>Port</td>
<td>Protocol</td>
<td>Protocol Type</td>
<td>Address/Identifier</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>----------</td>
<td>---------------</td>
<td>--------------------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>443</td>
<td>TCP</td>
<td>HTTPS</td>
<td>cloud.redhat.com</td>
<td>Red Hat Cloud plugin API calls</td>
<td></td>
</tr>
<tr>
<td>443</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Red Hat Portal</td>
<td>SOS report</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Assisting support cases (optional)</td>
<td></td>
</tr>
<tr>
<td>443</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Red Hat CDN</td>
<td>Content Sync</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Red Hat CDN</td>
<td></td>
</tr>
<tr>
<td>443</td>
<td>TCP</td>
<td>HTTPS</td>
<td>cert-api.access.redhat.com</td>
<td>Telemetry data upload and report</td>
<td></td>
</tr>
<tr>
<td>443</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Capsule</td>
<td>Content mirroring</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Initiation</td>
<td></td>
</tr>
<tr>
<td>443</td>
<td>TCP</td>
<td>HTTPS</td>
<td>Infoblox DHCP Server</td>
<td>DHCP management</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>When using Infoblox for DHCP, management of the DHCP leases (optional)</td>
<td></td>
</tr>
<tr>
<td>623</td>
<td>TCP</td>
<td></td>
<td>Client</td>
<td>Power management</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>BMC On/Off/Cycle/Status</td>
<td></td>
</tr>
<tr>
<td>5000</td>
<td>TCP</td>
<td>HTTPS</td>
<td>OpenStack Compute Resource</td>
<td>Compute resources</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Virtual machine interactions (query/create/destroy) (optional)</td>
<td></td>
</tr>
<tr>
<td>5646</td>
<td>TCP</td>
<td>AMQP</td>
<td>Satellite Server</td>
<td>Katello agent</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Forward message to Qpid dispatch router on Capsule (optional)</td>
<td></td>
</tr>
<tr>
<td>5671</td>
<td>TCP</td>
<td>AMQP</td>
<td>Qpid</td>
<td>Remote install</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Send install command to client</td>
<td></td>
</tr>
<tr>
<td>5671</td>
<td>TCP</td>
<td>AMQP</td>
<td>Dispatch router (hub)</td>
<td>Remote install</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Forward message to dispatch router on Satellite</td>
<td></td>
</tr>
<tr>
<td>5671</td>
<td>TCP</td>
<td>AMQP</td>
<td>Satellite Server</td>
<td>Remote install for Katello agent</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Send install command to client</td>
<td></td>
</tr>
<tr>
<td>5671</td>
<td>TCP</td>
<td>AMQP</td>
<td>Satellite Server</td>
<td>Remote install for Katello agent</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Forward message to dispatch router on Satellite</td>
<td></td>
</tr>
</tbody>
</table>
1.7. ENABLING CONNECTIONS FROM A CLIENT TO SATELLITE SERVER

Capsules and Content Hosts that are clients of a Satellite Server’s internal Capsule require access through Satellite’s host-based firewall and any network-based firewalls.

Use this procedure to configure the host-based firewall on the Red Hat Enterprise Linux 7 system that Satellite is installed on, to enable incoming connections from Clients, and to make the configuration persistent across system reboots. For more information on the ports used, see Ports and Firewalls Requirements.

Procedure

1. To open the ports for client to Satellite communication, enter the following command on the base operating system that you want to install Satellite on:

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

2. Make the changes persistent:
# firewall-cmd --runtime-to-permanent

## 1.8. VERIFYING FIREWALL SETTINGS

Use this procedure to verify your changes to the firewall settings.

**Procedure**

1. Enter the following command:

   ```
   # firewall-cmd --list-all
   ```

   For more information, see [Getting Started with firewalld](https://access.redhat.com/docs/en-us/red_hat_enterprise_linux/7/html/security_guide/chapter-10) in the *Red Hat Enterprise Linux 7 Security Guide*.

## 1.9. VERIFYING DNS RESOLUTION

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

**Procedure**

1. Ensure that the host name and local host resolve correctly:

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

   Successful name resolution results in output similar to the following:

   ```
   # ping -c1 localhost
   PING localhost (127.0.0.1) 56(84) bytes of data.
   64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.043 ms
   --- localhost ping statistics ---
   1 packets transmitted, 1 received, 0% packet loss, time 0ms
   rtt min/avg/max/mdev = 0.043/0.043/0.043/0.000 ms
   
   # ping -c1 `hostname -f`
   PING hostname.gateway (XX.XX.XX.XX) 56(84) bytes of data.
   64 bytes from hostname.gateway (XX.XX.XX.XX): icmp_seq=1 ttl=64 time=0.019 ms
   --- localhost.gateway ping statistics ---
   1 packets transmitted, 1 received, 0% packet loss, time 0ms
   rtt min/avg/max/mdev = 0.019/0.019/0.019/0.000 ms
   ```

2. To avoid discrepancies with static and transient host names, set all the host names on the system by entering the following command:

   ```
   # hostnamectl set-hostname name
   ```

   For more information, see the [Configuring Host Names Using hostnamectl](https://access.redhat.com/docs/en-us/red_hat_enterprise_linux/7/html/networking_guide/chapter-12) in the *Red Hat Enterprise Linux 7 Networking Guide*. 
WARNING

Name resolution is critical to the operation of Satellite 6. If Satellite cannot properly resolve its fully qualified domain name, tasks such as content management, subscription management, and provisioning will fail.
CHAPTER 2. PREPARING YOUR ENVIRONMENT FOR SATELLITE INSTALLATION IN AN IPV6 NETWORK

You can install and use Satellite in an IPv6 network. Before installing Satellite in an IPv6 network, view the limitations and ensure that you meet the requirements.

To provision hosts in an IPv6 network, after installing Satellite, you must also configure Satellite for the UEFI HTTP boot provisioning. For more information, see Configuring Satellite for UEFI HTTP Boot Provisioning in an IPv6 Network.

2.1. LIMITATIONS OF SATELLITE INSTALLATION IN AN IPV6 NETWORK

Satellite installation in an IPv6 network has the following limitations:

• You can install Satellite and Capsules in IPv6-only systems, dual-stack installation is not supported.

• Although Satellite provisioning templates include IPv6 support for PXE and HTTP (iPXE) provisioning, the only tested and certified provisioning workflow is the UEFI HTTP Boot provisioning. This limitation only relates to users who plan to use Satellite to provision hosts.

2.2. REQUIREMENTS FOR SATELLITE INSTALLATION IN AN IPV6 NETWORK

Before installing Satellite in an IPv6 network, ensure that you meet the following requirements:

• If you plan to provision hosts from Satellite or Capsules, you must install Satellite and Capsules on Red Hat Enterprise Linux version 7.9 or higher because these versions include the latest version of the `grub2` package.

• You must deploy an external DHCP IPv6 server as a separate unmanaged service to bootstrap clients into GRUB2, which then configures IPv6 networking either using DHCPv6 or assigning static IPv6 address. This is required because the DHCP server in Red Hat Enterprise Linux (ISC DHCP) does not provide an integration API for managing IPv6 records, therefore the Capsule DHCP plug-in that provides DHCP management is limited to IPv4 subnets.

• You must deploy an external IPv4 HTTP proxy server. This is required because Satellite distributes content only over IPv4 networks, therefore you must use an IPv4 proxy to redirect that content to hosts in your IPv6 network.

• You must configure Satellite to use this IPv4 HTTP proxy server as the default proxy. For more information, see Adding a Default HTTP Proxy to Satellite.
CHAPTER 3. INSTALLING SATELLITE SERVER

When you install Satellite Server from a connected network, you can obtain packages and receive updates directly from the Red Hat Content Delivery Network.

NOTE
You cannot register Satellite Server to itself.

Use the following procedures to install Satellite Server, perform the initial configuration, and import subscription manifests. For more information on subscription manifests, see Managing Subscriptions in the Content Management Guide.

Note that the Satellite 6 installation script is based on Puppet, which means that if you run the installation script more than once, it might overwrite any manual configuration changes. To avoid this and determine which future changes apply, use the --noop argument when you run the installation script. This argument ensures that no actual changes are made. Potential changes are written to /var/log/foreman-installer/satellite.log.

Files are always backed up and so you can revert any unwanted changes. For example, in the foreman-installer logs, you can see an entry similar to the following about Filebucket:

```
/Stage[main]/Dhcp/File[/etc/dhcp/dhcpd.conf]: Filebucketed /etc/dhcp/dhcpd.conf to puppet with sum 622d9820b8e764ab124367c68f5fa3a1
```

You can restore the previous file as follows:

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

3.1. 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.

Procedure

- Register your system with the Red Hat Content Delivery Network, entering your Customer Portal user name and password when prompted:

```
# 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
```

3.2. ATTACHING THE SATELLITE INFRASTRUCTURE SUBSCRIPTION
After you have registered Satellite Server, you must identify your subscription Pool ID and attach an available subscription. The Red Hat Satellite Infrastructure subscription provides access to the Red Hat Satellite, Red Hat Enterprise Linux, and Red Hat Software Collections (RHSCCL) content. This is the only subscription required.

Red Hat Satellite Infrastructure is included with all subscriptions that include Smart Management. For more information, see the Red Hat Knowledgebase solution Satellite Infrastructure Subscriptions MCT3718 MCT3719.

Subscriptions are classified as available if they are not already attached to a system. If you are unable to find an available Satellite subscription, see the Red Hat Knowledgebase solution How do I figure out which subscriptions have been consumed by clients registered under Red Hat Subscription Manager? to run a script to see if your subscription is being consumed by another system.

Procedure

1. Identify the Pool ID of the Satellite Infrastructure subscription:

   ```
   # subscription-manager list --all --available --matches 'Red Hat Satellite Infrastructure Subscription'
   ``

   The command displays output similar to the following:

   | Subscription Name: Red Hat Satellite Infrastructure Subscription |
   | Provides: Red Hat Satellite |
   | Red Hat Software Collections (for RHEL Server) |
   | Red Hat CodeReady Linux Builder for x86_64 |
   | Red Hat Ansible Engine |
   | Red Hat Enterprise Linux Load Balancer (for RHEL Server) |
   | Red Hat |
   | Red Hat Software Collections (for RHEL Server) |
   | Red Hat Enterprise Linux Server |
   | Red Hat Satellite Capsule |
   | Red Hat Enterprise Linux for x86_64 |
   | Red Hat Enterprise Linux High Availability for x86_64 |
   | Red Hat Satellite |
   | Red Hat Satellite 5 Managed DB |
   | Red Hat Satellite 6 |
   | Red Hat Discovery |
   | SKU: MCT3719 |
   | Contract: 11878983 |
   | Pool ID: 8a85f9996b92c370169ee998cf03b8 |
   | Provides Management: No |
   | Available: 1 |
   | Suggested: 1 |
   | Service Level: Premium |
   | Service Type: L1-L3 |
   | Subscription Type: Standard |
   | Ends: 03/04/2020 |
   | System Type: Physical |

2. Make a note of the subscription Pool ID. Your subscription Pool ID is different from the example provided.
3. Attach the Satellite Infrastructure subscription to the base operating system that your Satellite Server is running on:

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

The command displays output similar to the following:

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

4. Optional: Verify that the Satellite Infrastructure subscription is attached:

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

### 3.3. CONFIGURING REPOSITORIES

Use this procedure to enable the repositories that are required to install Satellite Server. Choose from the available list which operating system and version you are installing on:

- Red Hat Enterprise Linux 7

#### 3.3.1. Red Hat Enterprise Linux 7

1. Disable all repositories:

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

2. Enable the following repositories:

```
# subscription-manager repos --enable=rhel-7-server-rpms \
--enable=rhel-7-server-satellite-6.10-rpms \
--enable=rhel-7-server-satellite-maintenance-6-rpms \
--enable=rhel-server-rhscl-7-rpms \
--enable=rhel-7-server-ansible-2.9-rpms
```

**NOTE**

If you are installing Satellite Server as a virtual machine hosted on Red Hat Virtualization, you must also enable the Red Hat Common repository, and install Red Hat Virtualization guest agents and drivers. For more information, see Installing the Guest Agents and Drivers on Red Hat Enterprise Linux in the Virtual Machine Management Guide for more information.

### 3.4. INSTALLING SATELLITE SERVER PACKAGES

- Red Hat Enterprise Linux 7

#### 3.4.1. Red Hat Enterprise Linux 7

**Procedure**

1. Update all packages:

```
- 
```
# yum update

2. Install Satellite Server packages:

# yum install satellite

## 3.5. SYNCHRONIZING THE SYSTEM CLOCK WITH CHRONYD

To minimize the effects of time drift, you must synchronize the system clock on the base operating system on which you want to install Satellite Server with Network Time Protocol (NTP) servers. If the base operating system clock is configured incorrectly, certificate verification might fail.

For more information about the `chrony` suite, see Configuring NTP Using the `chrony` Suite in the Red Hat Enterprise Linux 7 System Administrator's Guide.

**Procedure**

1. Install the `chrony` package:

   # yum install chrony

2. Start and enable the `chronyd` service:

   # systemctl start chronyd
   # systemctl enable chronyd

## 3.6. INSTALLING THE SOS PACKAGE ON THE BASE OPERATING SYSTEM

Install the `sos` package on the base operating system so that you can 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 the Knowledgebase solution What is a sosreport and how to create one in Red Hat Enterprise Linux 4.6 and later? on the Red Hat Customer Portal.

**Procedure**

1. Install the `sos` package:

   # yum install sos

## 3.7. CONFIGURING SATELLITE SERVER

Install Satellite Server using the `satellite-installer` installation script.

This method is performed by running the installation script with one or more command options. The command options override the corresponding default initial configuration options and are recorded in the Satellite answer file. You can run the script as often as needed to configure any necessary options.
NOTE

Depending on the options that you use when running the Satellite installer, the configuration can take several minutes to complete.

3.7.1. Configuring Satellite

This initial configuration procedure creates an organization, location, user name, and password. After the initial configuration, you can create additional organizations and locations if required. The initial configuration also installs PostgreSQL databases on the same server.

The installation process can take tens of minutes to complete. If you are connecting remotely to the system, use a utility such as `screen` or `tmux` that allows suspending and reattaching a communication session so that you can check the installation progress in case you become disconnected from the remote system. The Red Hat Knowledgebase article How to use the screen command describes installing `screen`; alternately see the `screen` manual page for more information. If you lose connection to the shell where the installation command is running, see the log at `/var/log/foreman-installer/satellite.log` to determine if the process completed successfully.

Considerations

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

- Specify a meaningful value for the option: `--foreman-initial-organization`. This can be your company name. An internal label that matches the value is also created and cannot be changed afterwards. If you do not specify a value, an organization called `Default Organization` with the label `Default_Organization` is created. You can rename the organization name but not the label.

- Remote Execution is the primary method of managing packages on Content Hosts. If you want to use the deprecated Katello Agent instead of Remote Execution SSH, use the `--foreman-proxy-content-enable-katello-agent=true` option to enable it. The same option should be given on any Capsule Server as well as Satellite Server.

- By default, all configuration files configured by the installer are managed by Puppet. When `satellite-installer` runs, it overwrites any manual changes to the Puppet managed files with the initial values. By default, Satellite Server is installed with the Puppet agent running as a service. If required, you can disable Puppet agent on Satellite Server using the `--puppet-runmode=none` option.

- If you want to manage DNS files and DHCP files manually, use the `--foreman-proxy-dns-managed=false` and `--foreman-proxy-dhcp-managed=false` options so that Puppet does not manage the files related to the respective services. For more information on how to apply custom configuration on other services, see Applying Custom Configuration to Satellite.

Procedure

1. Enter the following command with any additional options that you want to use:

```bash
# satellite-installer --scenario satellite \ 
--foreman-initial-organization "initial_organization_name" \ 
--foreman-initial-location "initial_location_name" \ 
--foreman-initial-admin-username admin_user_name \ 
--foreman-initial-admin-password admin_password
```
The script displays its progress and writes logs to /var/log/foreman-installer/satellite-installer--scenario satellite.log.

3.8. IMPORTING A SUBSCRIPTION MANIFEST INTO SATELLITE SERVER

Use the following procedure to import a Subscription Manifest into Satellite Server.

Prerequisites

- You must have a Subscription Manifest file exported from the Customer Portal. For more information, see Using Manifests in the Using Red Hat Subscription Management guide.

Procedure

1. In the Satellite web UI, ensure the context is set to the organization you want to use.
2. Navigate to Content > Subscriptions and click Manage Manifest.
3. In the Manage Manifest window, click Browse.
4. Navigate to the location that contains the Subscription Manifest file, then click Open. If the Manage Manifest window does not close automatically, click Close to return to the Subscriptions window.

CLI procedure

1. Copy the Subscription Manifest file from your client to Satellite Server:

   ```sh
ds cp ~/.manifest_file.zip root@satellite.example.com:~/.manifest_file.zip
   ``

2. Log in to Satellite Server as the root user and import the Subscription Manifest file:

   ```sh
   # hammer subscription upload
   --file ~/.manifest_file.zip
   --organization "organization_name"
   ```
CHAPTER 4. PERFORMING ADDITIONAL CONFIGURATION ON SATELLITE SERVER

4.1. USING RED HAT INSIGHTS WITH SATELLITE SERVER

You can use Red Hat Insights to diagnose systems and downtime related to security exploits, performance degradation and stability failures. You can use the dashboard to quickly identify key risks to stability, security, and performance. You can sort by category, view details of the impact and resolution, and then determine what systems are affected.

Note that you do not require a Red Hat Insights entitlement in your subscription manifest. For more information about Satellite and Red Hat Insights, see Red Hat Insights on Satellite Red Hat Enterprise Linux (RHEL).

To maintain your Satellite Server, and improve your ability to monitor and diagnose problems you might have with Satellite, install Red Hat Insights on Satellite Server and register Satellite Server with Red Hat Insights.

Scheduling insights-client

Note that you can change the default schedule for running insights-client by configuring insights-client.timer on Satellite. For more information, see Changing the insights-client schedule in the Client Configuration Guide for Red Hat Insights.

Procedure

1. To install Red Hat Insights on Satellite Server, enter the following command:

   # satellite-maintain packages install insights-client

2. To register Satellite Server with Red Hat Insights, enter the following command:

   # satellite-installer --register-with-insights

4.2. DISABLING REGISTRATION TO RED HAT INSIGHTS

After you install or upgrade Satellite, you can choose to unregister or register Red Hat Insights as needed. For example, if you need to use Satellite in a disconnected environment, you can unregister insights-client from Satellite Server.

Prerequisites

1. You have registered Satellite to Red Hat Customer Portal.

Procedure

1. Optional: To unregister Red Hat Insights from Satellite Server, enter the following command:

   # insights-client --unregister

2. Optional: To register Satellite Server with Red Hat Insights, enter the following command:
4.3. ENABLING THE SATELLITE TOOLS 6.10 REPOSITORY

The Satellite Tools 6.10 repository provides the katello-agent, katello-host-tools, and puppet packages for clients registered to Satellite Server.

To use the CLI instead of the web UI, see the CLI procedure.

Procedure

1. In the Satellite web UI, navigate to Content > Red Hat Repositories.
2. Use the Search field to enter the following repository name: Satellite Tools 6.10 (for RHEL 7 Server) (RPMs).
3. In the Available Repositories pane, click on Satellite Tools 6.10 (for RHEL 7 Server) (RPMs) to expand the repository set.
   If the Satellite Tools 6.10 items are not visible, it may be because they are not included in the Subscription Manifest obtained from the Customer Portal. To correct that, log in to the Customer Portal, add these repositories, download the Subscription Manifest and import it into Satellite.
4. For the x86_64 entry, click the Enable icon to enable the repository.

Enable the Satellite Tools 6.10 repository for every supported major version of Red Hat Enterprise Linux running on your hosts. After enabling a Red Hat repository, a Product for this repository is automatically created.

CLI procedure

- Enable the Satellite Tools 6.10 repository using the hammer repository-set enable command:
  ```bash
  # hammer repository-set enable --organization "initial_organization_name"
  --product 'Red Hat Enterprise Linux Server'
  --basearch='x86_64'
  --name 'Red Hat Satellite Tools 6.10 (for RHEL 7 Server) (RPMs)'
  ```

4.4. SYNCHRONIZING THE SATELLITE TOOLS 6.10 REPOSITORY

Use this section to synchronize the Satellite Tools 6.10 repository from the Red Hat Content Delivery Network (CDN) to your Satellite. This repository provides the katello-agent, katello-host-tools, and puppet packages for clients registered to Satellite Server.

Procedure

1. In the Satellite web UI, navigate to Content > Sync Status.
   A list of product repositories available for synchronization is displayed.
2. Click the arrow next to the Red Hat Enterprise Linux Server product to view available content.
3. Select Satellite Tools 6.10 (for RHEL 7 Server) RPMs x86_64
4. Click Synchronize Now.
CLI procedure

- Synchronize your Satellite Tools 6.10 repository using the `hammer repository synchronize` command:

  ```sh
  # hammer repository synchronize --organization "initial_organization_name"
  --product 'Red Hat Enterprise Linux Server'
  --name 'Red Hat Satellite Tools 6.10 for RHEL 7 Server RPMs x86_64'
  --async
  ```

4.5. CONFIGURING SATELLITE FOR UEFI HTTP BOOT PROVISIONING IN AN IPV6 NETWORK

Use this procedure to configure Satellite to provision hosts in an IPv6 network with UEFI HTTP Boot provisioning.

Prerequisites

- Ensure that your clients can access DHCP and HTTP servers.
- Ensure that the UDP ports 67 and 68 are accessible by clients so clients can send DHCP requests and receive DHCP offers.
- Ensure that the TCP port 8000 is open for clients to download files and Kickstart templates from Satellite and Capsules.
- Ensure that the host provisioning interface subnet has an HTTP Boot Capsule, and Templates Capsule set. For more information, see Adding a Subnet to Satellite Server in the Provisioning Guide.
- Navigate to Administer > Settings > Provisioning and ensure that the Token duration setting is not set to 0. Satellite cannot identify clients that are booting from the network by a remote IPv6 address because of unmanaged DHCPv6 service, therefore provisioning tokens must be enabled.

Procedure

1. You must disable DHCP management in the installer or not use it.
2. For all IPv6 subnets created in Satellite, set the DHCP Capsule to blank.
3. Optional: If the host and the DHCP server are separated by a router, configure the DHCP relay agent and point to the DHCP server.
4. On Satellite or Capsule from which you provision, update the `grub2-efi` package to the latest version:

   ```sh
   # satellite-maintain packages install grub2-efi
   ```
5. Synchronize the Red Hat Enterprise Linux 8 kickstart repository.

4.6. CONFIGURING SATELLITE SERVER WITH AN HTTP PROXY

Use the following procedures to configure Satellite with an HTTP proxy.
4.6.1. Adding a Default HTTP Proxy to Satellite

If your network uses an HTTP Proxy, you can configure Satellite Server to use an HTTP proxy for requests to the Red Hat Content Delivery Network (CDN) or another content source. Use the FQDN instead of the IP address where possible to avoid losing connectivity because of network changes.

The following procedure configures a proxy only for downloading content for Satellite. To use the CLI instead of the web UI, see the CLI procedure.

Procedure

1. In the Satellite web UI, navigate to Infrastructure > HTTP Proxies.
2. Click New HTTP Proxy.
3. In the Name field, enter the name for the HTTP proxy.
4. In the Url field, enter the URL of the HTTP proxy in the following format: \https://proxy.example.com:8080.
5. Optional: If authentication is required, in the Username field, enter the username to authenticate with.
6. Optional: If authentication is required, in the Password field, enter the password to authenticate with.
7. To test connection to the proxy, click the Test Connection button.
8. Click Submit.
9. Navigate to Administer > Settings, and click the Content tab.
10. Set the Default HTTP Proxy setting to the created HTTP proxy.

CLI procedure

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

```bash
# unset http_proxy
# unset https_proxy
# unset no_proxy
```

2. Add an HTTP proxy entry to Satellite:

```bash
# hammer http-proxy create --name=myproxy
  --url=http://myproxy.example.com:8080
  --username=proxy_username
  --password=proxy_password
```

3. Configure Satellite to use this HTTP proxy by default:

```bash
# hammer settings set --name=content_default_http_proxy --value=myproxy
```

4.6.2. Configuring the HTTP Proxy to Connect to Red Hat CDN
Verify that Satellite can connect to the Red Hat CDN and can synchronize its repositories.

**Procedure**

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

<table>
<thead>
<tr>
<th>Host name</th>
<th>Port</th>
<th>Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>subscription.rhsm.redhat.com</td>
<td>443</td>
<td>HTTPS</td>
</tr>
<tr>
<td>cdn.redhat.com</td>
<td>443</td>
<td>HTTPS</td>
</tr>
<tr>
<td>*.akamaiedge.net</td>
<td>443</td>
<td>HTTPS</td>
</tr>
<tr>
<td>cert-api.access.redhat.com (if using Red Hat Insights)</td>
<td>443</td>
<td>HTTPS</td>
</tr>
<tr>
<td>api.access.redhat.com (if using Red Hat Insights)</td>
<td>443</td>
<td>HTTPS</td>
</tr>
</tbody>
</table>

Satellite Server uses SSL to communicate with the Red Hat CDN securely. Use of an SSL interception proxy interferes with this communication. These hosts must be whitelisted on the proxy.

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](https://www.redhat.com) on the Red Hat Customer Portal.

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

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

# port for http proxy server
proxy_port = 8080

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

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

4.6.3. Configuring SELinux to Ensure Access to Satellite on Custom Ports

SELinux ensures access of Red Hat Satellite 6 and Red Hat Subscription Manager only to specific ports. In the case of the HTTP cache, the TCP ports are 8080, 8118, 8123, and 10001 - 10010. If you use a port that does not have SELinux type `http_cache_port_t`, complete the following steps.

**Procedure**

1. On Satellite, to verify the ports that are permitted by SELinux for the HTTP cache, enter a command as follows:
To configure SELinux to permit a port for the HTTP cache, for example 8088, enter a command as follows:

```
# semanage port -a -t http_cache_port_t -p tcp 8088
```

4.6.4. Using an HTTP Proxy for all Satellite HTTP Requests

If your Satellite Server must remain behind a firewall that blocks HTTP and HTTPS, you can configure a proxy for communication with external systems, including compute resources.

Note that if you are using compute resources for provisioning, and you want to use a different HTTP proxy with the compute resources, the proxy that you set for all Satellite communication takes precedence over the proxies that you set for compute resources.

**Procedure**

1. In the Satellite web UI, navigate to **Administer > Settings**.
2. In the **HTTP(S) proxy** row, select the adjacent **Value** column and enter the proxy URL.
3. Click the tick icon to save your changes.

**CLI procedure**

- Enter the following command:

```
# hammer settings set --name=http_proxy --value=Proxy_URL
```

4.6.5. Excluding Hosts from Receiving Proxied Requests

If you use an HTTP Proxy for all Satellite HTTP or HTTPS requests, you can prevent certain hosts from communicating through the proxy.

**Procedure**

1. In the Satellite web UI, navigate to **Administer > Settings**.
2. In the **HTTP(S) proxy except hosts** row, select the adjacent **Value** column and enter the names of one or more hosts that you want to exclude from proxy requests.
3. Click the tick icon to save your changes.

**CLI procedure**

- Enter the following command:

```
# hammer settings set --name=http_proxy_except_list --value=[hostname1.hostname2...]
```
4.6.6. Resetting the HTTP Proxy

If you want to reset the current HTTP proxy setting, unset the **Default HTTP Proxy** setting.

**Procedure**

1. Navigate to **Administer** > **Settings**, and click the **Content** tab.
2. Set the **Default HTTP Proxy** setting to **no global default**

**CLI procedure**

- Set the **content_default_http_proxy** setting to an empty string:

  ```bash
  # hammer settings set --name=content_default_http_proxy --value=""
  ```

4.7. ENABLING POWER MANAGEMENT ON MANAGED HOSTS

To perform power management tasks on managed hosts using the intelligent platform management interface (IPMI) or a similar protocol, you must enable the baseboard management controller (BMC) module on Satellite Server.

**Prerequisites**

- All managed hosts must have a network interface of BMC type. Satellite Server uses this NIC to pass the appropriate credentials to the host. For more information, see [Adding a Baseboard Management Controller (BMC) Interface](Managing_Hosts) in *Managing Hosts*.

**Procedure**

- To enable BMC, enter the following command:

  ```bash
  # satellite-installer --foreman-proxy-bmc "true" \
  --foreman-proxy-bmc-default-provider "freeipmi"
  ```

4.8. CONFIGURING DNS, DHCP, AND TFTP ON SATELLITE SERVER

To configure the DNS, DHCP, and TFTP services on Satellite Server, use the `satellite-installer` command with the options appropriate for your environment. To view a complete list of configurable options, enter the `satellite-installer --scenario satellite --help` command.

Any changes to the settings require entering the `satellite-installer` command again. You can enter the command multiple times and each time it updates all configuration files with the changed values.

To use external DNS, DHCP, and TFTP services instead, see [Chapter 5, Configuring Satellite Server with External Services](#).

**Adding Multihomed DHCP details**

If you want to use Multihomed DHCP, you must inform the installer.

**Prerequisites**

- Ensure that the following information is available to you:
CHAPTER 4. PERFORMING ADDITIONAL CONFIGURATION ON SATELLITE SERVER

- 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.
- Contact your network administrator to ensure that you have the correct settings.

Procedure

- Enter the **satellite-installer** command with the options appropriate for your environment. The following example shows configuring full provisioning services:

```bash
# satellite-installer --scenario satellite
  --foreman-proxy-dns true
  --foreman-proxy-dns-managed true
  --foreman-proxy-dns-interface eth0
  --foreman-proxy-dns-zone example.com
  --foreman-proxy-dns-reverse 2.0.192.in-addr.arpa
  --foreman-proxy-dhcp true
  --foreman-proxy-dhcp-managed true
  --foreman-proxy-dhcp-interface eth0
  --foreman-proxy-dhcp-additional-interfaces eth1
  --foreman-proxy-dhcp-additional-interfaces eth2
  --foreman-proxy-dhcp-range "192.0.2.100 192.0.2.150"
  --foreman-proxy-dhcp-gateway 192.0.2.1
  --foreman-proxy-dhcp-nameservers 192.0.2.2
  --foreman-proxy-tftp true
  --foreman-proxy-tftp-managed true
  --foreman-proxy-tftp-servername 192.0.2.3
```

You can monitor the progress of the **satellite-installer** command displayed in your prompt. You can view the logs in `/var/log/foreman-installer/satellite.log`. You can view the settings used, including the `initial_admin_password` parameter, in the `/etc/foreman-installer/scenarios.d/satellite-answers.yaml` file.

For more information about configuring DHCP, DNS, and TFTP services, see the Configuring Network Services section in the Provisioning Guide.

### 4.9. DISABLING DNS, DHCP, AND TFTP FOR UNMANAGED NETWORKS

If you want to manage TFTP, DHCP, and DNS services manually, you must prevent Satellite from maintaining these services on the operating system and disable orchestration to avoid DHCP and DNS validation errors. However, Satellite does not remove the back-end services on the operating system.

Procedure

1. On Satellite Server, enter the following command:
2. In the Satellite web UI, navigate to Infrastructure > Subnets and select a subnet.

3. Click the Capsules tab and clear the DHCP Capsule, TFTP Capsule, and Reverse DNS Capsule fields.

4. Navigate to Infrastructure > Domains and select a domain.

5. Clear the DNS Capsule field.

6. Optional: If you use a DHCP service supplied by a third party, configure your DHCP server to pass the following options:
   
   Option 66: IP address of Satellite or Capsule
   Option 67: /pxelinux.0

   For more information about DHCP options, see RFC 2132.

   **NOTE**

   Satellite 6 does not perform orchestration when a Capsule is not set for a given subnet and domain. When enabling or disabling Capsule associations, orchestration commands for existing hosts can fail if the expected records and configuration files are not present. When associating a Capsule to turn orchestration on, make sure the required DHCP and DNS records as well as the TFTP files are in place for the existing Satellite hosts in order to prevent host deletion failures in the future.

### 4.10. CONFIGURING SATELLITE SERVER FOR OUTGOING EMAILS

To send email messages from Satellite Server, you can use either an SMTP server, or the `sendmail` command.

#### Prerequisites

- Some SMTP servers with anti-spam protection or grey-listing features are known to cause problems. To setup outgoing email with such a service either install and configure a vanilla SMTP service on Satellite Server for relay or use the `sendmail` command instead.

#### Procedure

1. In the Satellite web UI, navigate to Administer → Settings.

2. Click the Email tab and set the configuration options to match your preferred delivery method. The changes have an immediate effect.
   
   a. The following example shows the configuration options for using an SMTP server:

   **Table 4.1. Using an SMTP server as a delivery method**
Name | Example value
--- | ---
Delivery method | SMTP
SMTP address | smtp.example.com
SMTP authentication | login
SMTP HELO/EHLO domain | example.com
SMTP password | password
SMTP port | 25
SMTP username | user@example.com

The SMTP username and SMTP password specify the login credentials for the SMTP server.

b. The following example uses gmail.com as an SMTP server:

<table>
<thead>
<tr>
<th>Name</th>
<th>Example value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delivery method</td>
<td>SMTP</td>
</tr>
<tr>
<td>SMTP address</td>
<td>smtp.gmail.com</td>
</tr>
<tr>
<td>SMTP authentication</td>
<td>plain</td>
</tr>
<tr>
<td>SMTP HELO/EHLO domain</td>
<td>smtp.gmail.com</td>
</tr>
<tr>
<td>SMTP enable StartTLS auto</td>
<td>Yes</td>
</tr>
<tr>
<td>SMTP password</td>
<td>password</td>
</tr>
<tr>
<td>SMTP port</td>
<td>587</td>
</tr>
<tr>
<td>SMTP username</td>
<td><a href="mailto:user@gmail.com">user@gmail.com</a></td>
</tr>
</tbody>
</table>

c. The following example uses the sendmail command as a delivery method:

<table>
<thead>
<tr>
<th>Name</th>
<th>Example value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delivery method</td>
<td>sendmail</td>
</tr>
<tr>
<td>SMTP address</td>
<td></td>
</tr>
</tbody>
</table>
The **Sendmail arguments** specify the options passed to the `sendmail` command. The default value is `-i -t`. For more information see the `sendmail(1)` man page.

3. If you decide to send email using an SMTP server which uses TLS authentication, also perform one of the following steps:
   - Mark the CA certificate of the SMTP server as trusted. To do so, execute the following commands on Satellite Server:
     ```bash
     # cp mailca.crt /etc/pki/ca-trust/source/anchors/
     # update-ca-trust enable
     # update-ca-trust
     ```
     Where `mailca.crt` is the CA certificate of the SMTP server.
   - Alternatively, in the web UI, set the **SMTP enable StartTLS auto** option to **No**.

4. Click **Test email** to send a test message to the user’s email address to confirm the configuration is working. If a message fails to send, the web UI displays an error. See the log at `/var/log/foreman/production.log` for further details.

   **NOTE**
   For information on configuring email notifications for individual users or user groups, see Configuring Email Notifications in Administering Red Hat Satellite.

### 4.11. Configuring an Alternate CNAME for Satellite

You can configure an alternate CNAME for Satellite. This might be useful if you want to deploy the Satellite web interface on a different domain name than the one that is used by client systems to connect to Satellite. You must plan the alternate CNAME configuration in advance prior to installing Capsules and registering hosts to Satellite to avoid redeploying new certificates to hosts.

#### 4.11.1. Configuring Satellite with an Alternate CNAME

Use this procedure to configure Satellite with an alternate CNAME. Note that the procedures for users of a default Satellite certificate and custom certificate differ.

**For Default Satellite Certificate Users**

- If you have installed Satellite with a default Satellite certificate and want to configure Satellite with an alternate CNAME, enter the following command on Satellite to generate a new default Satellite SSL certificate with an additional CNAME.
  ```bash
  # satellite-installer --certs-cname alternate_fqdn --certs-update-server
  ```
If you have not installed Satellite, you can add the `--certs-cname alternate_fqdn` option to the `satellite-installer` command to install Satellite with an alternate CNAME.

For Custom Certificate Users
If you use Satellite with a custom certificate, when creating a custom certificate, include the alternate CNAME records to the custom certificate. For more information, see Creating a Custom SSL Certificate for Satellite Server.

4.11.2. Configuring Hosts to Use an Alternate Satellite CNAME for Content Management
If Satellite is configured with an alternate CNAME, you can configure hosts to use the alternate Satellite CNAME for content management. To do this, you must point hosts to the alternate Satellite CNAME prior to registering the hosts to Satellite. You can do this using the bootstrap script or manually.

Configuring Hosts with the bootstrap Script
On the host, run the bootstrap script with the `--server alternate_fqdn.example.com` option to register the host to the alternate Satellite CNAME:

```
# ./bootstrap.py --server alternate_fqdn.example.com
```

Configuring Hosts Manually
On the host, edit the `/etc/rhsm/rhsm.conf` file to update `hostname` and `baseurl` settings to point to the alternate host name, for example:

```
[server]
# Server hostname:
hostname = alternate_fqdn.example.com

[rhsm]
# Content base URL:
baseurl=https://alternate_fqdn.example.com/pulp/content/
```

Now you can register the host with the `subscription-manager`.

4.12. CONFIGURING SATELLITE SERVER WITH A CUSTOM SSL CERTIFICATE
By default, Red Hat Satellite 6 uses a self-signed SSL certificate to enable encrypted communications between Satellite Server, external Capsule Servers, and all hosts. If you cannot use a Satellite self-signed certificate, you can configure Satellite Server to use an SSL certificate signed by an external Certificate Authority.

To configure your Satellite Server with a custom certificate, complete the following procedures:

1. **Section 4.12.1, “Creating a Custom SSL Certificate for Satellite Server”**
2. **Section 4.12.2, “Deploying a Custom SSL Certificate to Satellite Server”**
3. **Section 4.12.3, “Deploying a Custom SSL Certificate to Hosts”**
4. If you have external Capsule Servers registered to Satellite Server, you must configure them with custom SSL certificates. The same Certificate Authority must sign certificates for Satellite Server and Capsule Server. For more information, see Configuring Capsule Server with a Custom SSL Certificate in Installing Capsule Server.


Use this procedure to create a custom SSL certificate for Satellite Server. If you already have a custom SSL certificate for Satellite Server, skip this procedure.

When you configure Satellite Server with custom certificates, note the following considerations:

- You must use the Privacy-Enhanced Mail (PEM) encoding for the SSL certificates.
- You cannot use the same certificate for both Satellite Server and Capsule Server.
- The same Certificate Authority must sign certificates for Satellite Server and Capsule Server.

Procedure

1. To store all the source certificate files, create a directory that is accessible only to the root user.

   ```
   # mkdir /root/satellite_cert
   ```

2. Create a private key with which to sign the Certificate Signing Request (CSR). Note that the private key must be unencrypted. If you use a password-protected private key, remove the private key password.

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

   ```
   # openssl genrsa -out /root/satellite_cert/satellite_cert_key.pem 4096
   ```

3. Create the `/root/satellite_cert/openssl.cnf` configuration file for the Certificate Signing Request (CSR) and include the following content:

   ```
   [ req ]
   req_extensions = v3_req
   distinguished_name = req_distinguished_name
   x509_extensions = usr_cert
   prompt = no

   [ req_distinguished_name ]
   C = Country Name (2 letter code)
   ST = State or Province Name (full name)
   L = Locality Name (eg, city)
   O = Organization Name (eg, company)
   OU = The division of your organization handling the certificate
   CN = satellite.example.com

   [ v3_req ]
   basicConstraints = CA:FALSE
   keyUsage = digitalSignature, nonRepudiation, keyEncipherment, dataEncipherment
   extendedKeyUsage = serverAuth, clientAuth, codeSigning, emailProtection
   subjectAltName = @alt_names
   ```
In the [usr_cert] section, enter information about your organization.

Set the certificate’s Common Name CN to match the fully qualified domain name (FQDN) of your Satellite Server. To confirm a FQDN, on that Satellite Server, enter the hostname -f command. This is required to ensure that the katello-certs-check command validates the certificate correctly.

Set the Subject Alternative Name (SAN) DNS.1 to match the fully qualified domain name (FQDN) of your server.

4. Generate the Certificate Signing Request (CSR):

```
# openssl req -new \\
  -key /root/satellite_cert/satellite_cert_key.pem 1 \\
  -config /root/satellite_cert/openssl.cnf 2 \\
  -out /root/satellite_cert/satellite_cert_csr.pem 3
```

1. Path to the private key.
2. Path to the configuration file.
3. Path to the CSR to generate.

5. Send the certificate signing request to the Certificate Authority. The same Certificate Authority must sign certificates for Satellite Server and Capsule Server.

When you submit the request, 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.

### 4.12.2. Deploying a Custom SSL Certificate to Satellite Server

Use this procedure to configure your Satellite Server to use a custom SSL certificate signed by a Certificate Authority. The katello-certs-check command validates the input certificate files and returns the commands necessary to deploy a custom SSL certificate to Satellite Server.

Procedure
1. Validate the custom SSL certificate input files. Note that for the `katello-certs-check` command to work correctly, Common Name (CN) in the certificate must match the FQDN of Satellite Server.

```
# katello-certs-check \
- c /root/satellite_cert/satellite_cert.pem \   1
 - k /root/satellite_cert/satellite_cert_key.pem \  2
 - b /root/satellite_cert/ca_cert_bundle.pem       3
```

1. Path to Satellite Server certificate file that is signed by a Certificate Authority.
2. Path to the private key that was used to sign Satellite Server certificate.
3. Path to the Certificate Authority bundle.

If the command is successful, it returns two `satellite-installer` commands, one of which you must use to deploy a certificate to Satellite Server.

**Example output of `katello-certs-check`**

```
Validation succeeded.
To install the Red Hat Satellite Server with the custom certificates, run:
satellite-installer --scenario satellite \ 
 --certs-server-cert "/root/satellite_cert/satellite_cert.pem" \ 
 --certs-server-key "/root/satellite_cert/satellite_cert_key.pem" \ 
 --certs-server-ca-cert "/root/satellite_cert/ca_cert_bundle.pem"
```

```
To update the certificates on a currently running Red Hat Satellite installation, run:
satellite-installer --scenario satellite \ 
 --certs-server-cert "/root/satellite_cert/satellite_cert.pem" \ 
 --certs-server-key "/root/satellite_cert/satellite_cert_key.pem" \ 
 --certs-server-ca-cert "/root/satellite_cert/ca_cert_bundle.pem" \ 
 --certs-update-server --certs-update-server-ca
```

2. From the output of the `katello-certs-check` command, depending on your requirements, enter the `satellite-installer` command that installs a new Satellite with custom SSL certificates or updates certificates on a currently running Satellite.

If you are unsure which command to run, you can verify that Satellite is installed by checking if the file `/etc/foreman-installer/scenarios.d/.installed` exists. If the file exists, run the second `satellite-installer` command that updates certificates.

**IMPORTANT**

Do not delete the certificate archive file after you deploy the certificate. It is required, for example, when upgrading Satellite Server.

3. On a computer with network access to Satellite Server, navigate to the following URL:

  `https://satellite.example.com`

4. In your browser, view the certificate details to verify the deployed certificate.
4.12.3. Deploying a Custom SSL Certificate to Hosts

After you configure Satellite Server to use a custom SSL certificate, you must also install the `katello-ca-consumer` package on every host that is registered to this Satellite Server.

Procedure

- On each host, install the `katello-ca-consumer` package:

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

4.13. USING EXTERNAL DATABASES WITH SATELLITE

As part of the installation process for Red Hat Satellite, the `satellite-installer` command installs PostgreSQL databases on the same server as Satellite. In certain Satellite deployments, using external databases instead of the default local databases can help with the server load.

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

To create and use external databases for Satellite, you must complete the following procedures:

1. Section 4.13.2, "Preparing a Host for External Databases". Prepare a Red Hat Enterprise Linux 7 server to host the external databases.
3. Section 4.13.4, "Configuring Satellite to use External Databases". Edit the parameters of `satellite-installer` to point to the new databases, and run `satellite-installer`.

4.13.1. PostgreSQL as an External Database Considerations

Foreman, Katello, and Candlepin use the PostgreSQL database. If you want to use PostgreSQL as an external database, the following information can help you decide if this option is right for your Satellite configuration. Satellite supports PostgreSQL version 12.1.

Advantages of External PostgreSQL:

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

Disadvantages of External PostgreSQL

- Increase in deployment complexity that can make troubleshooting more difficult
- The external PostgreSQL server is an additional system to patch and maintain
If either Satellite or the PostgreSQL database server suffers a hardware or storage failure, Satellite is not operational.

If there is latency between the Satellite server and database server, performance can suffer.

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

### 4.13.2. Preparing a Host for External Databases

Install a freshly provisioned system with the latest Red Hat Enterprise Linux 7 server to host the external databases.

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

**Prerequisites**

- The Red Hat Enterprise Linux 7 server must meet Satellite’s [Storage Requirements](#).

**Procedure**

1. Use the instructions in *Attaching the Satellite Infrastructure Subscription* to attach a Satellite subscription to your server.

2. Disable all repositories and enable only the following repositories:

```
# subscription-manager repos --disable "*"
# subscription-manager repos --enable=rhel-server-rhscl-7-rpms
    --enable=rhel-7-server-rpms
    --enable=rhel-7-server-satellite-6.10-rpms
```

### 4.13.3. Installing PostgreSQL

You can install only the same version of PostgreSQL that is installed with the *satellite-installer* tool during an internal database installation. You can install PostgreSQL using Red Hat Enterprise Linux Server 7 repositories or from an external source, as long as the version is supported. Satellite supports PostgreSQL version 12.1.

**Procedure**

1. To install PostgreSQL, enter the following command:

```
# yum install rh-postgresql12-postgresql-server
    rh-postgresql12-syspaths
    rh-postgresql12-postgresql-evr
```

2. To initialize PostgreSQL, enter the following command:

```
# postgresql-setup initdb
```
3. Edit the `/var/opt/rh/rh-postgresql12/lib/pgsql/data/postgresql.conf` file:
   ```bash
   # vi /var/opt/rh/rh-postgresql12/lib/pgsql/data/postgresql.conf
   ``

4. Remove the `#` and edit to listen to inbound connections:
   ```bash
   listen_addresses = '*'
   ``

5. Edit the `/var/opt/rh/rh-postgresql12/lib/pgsql/data/pg_hba.conf` file:
   ```bash
   # vi /var/opt/rh/rh-postgresql12/lib/pgsql/data/pg_hba.conf
   ``

6. Add the following line to the file:
   ```bash
   host all all Satellite_ip/24 md5
   ``

7. To start, and enable PostgreSQL service, enter the following commands:
   ```bash
   # systemctl start postgresql
   # systemctl enable postgresql
   ``

8. Open the `postgresql` port on the external PostgreSQL server:
   ```bash
   # firewall-cmd --add-service=postgresql
   # firewall-cmd --runtime-to-permanent
   ``

9. Switch to the `postgres` user and start the PostgreSQL client:
   ```bash
   $ su - postgres -c psql
   ``

10. Create three databases and dedicated roles: one for Satellite, one for Candlepin, and one for Pulp:
    ```bash
        CREATE USER "foreman" WITH PASSWORD 'Foreman_Password';
        CREATE USER "candlepin" WITH PASSWORD 'Candlepin_Password';
        CREATE USER "pulp" WITH PASSWORD 'Pulpcore_Password';
        CREATE DATABASE foreman OWNER foreman;
        CREATE DATABASE candlepin OWNER candlepin;
        CREATE DATABASE pulpcore OWNER pulp;
    ```

11. Exit the `postgres` user:
    ```bash
        # \q
    ```

12. From Satellite Server, test that you can access the database. If the connection succeeds, the commands return 1.
    ```bash
        # PGPASSWORD='Foreman_Password' psql -h postgres.example.com -p 5432 -U foreman -d foreman -c "SELECT 1 as ping"
        # PGPASSWORD='Candlepin_Password' psql -h postgres.example.com -p 5432 -U
    ```
candlepin -d candlepin -c "SELECT 1 as ping"
# PGPassword='Pulpcore_Password' psql -h postgres.example.com -p 5432 -U pulp -d pulpcore -c "SELECT 1 as ping"

4.13.4. Configuring Satellite to use External Databases

Use the `satellite-installer` command to configure Satellite to connect to an external PostgreSQL database.

Prerequisites

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

Procedure

1. To configure the external databases for Satellite, enter the following command:

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

   To enable the Secure Sockets Layer (SSL) protocol for these external databases, add the following options:

   ```bash
   --foreman-db-sslmode verify-full \
   --foreman-db-root-cert <path_to_CA> \
   --katello-candlepin-db-ssl true \
   --katello-candlepin-db-ssl-verify true \
   --foreman-proxy-content-pulpcore-postgresql-ssl true \
   --foreman-proxy-content-pulpcore-postgresql-ssl-root-ca <path_to_CA>
   ```

4.14. TUNING SATELLITE SERVER WITH PREDEFINED PROFILES

If your Satellite deployment includes more than 5000 hosts, you can use predefined tuning profiles to improve performance of Satellite.

Note that you cannot use tuning profiles on Capsules.

You can choose one of the profiles depending on the number of hosts your Satellite manages and available hardware resources.

The tuning profiles are available in the `/usr/share/foreman-installer/config/foreman.hiera/tuning/sizes` directory.
When you run the `satellite-installer` command with the `--tuning` option, deployment configuration settings are applied to Satellite in the following order:

1. The default tuning profile defined in the `/usr/share/foreman-installer/config/foreman.hiera/tuning/common.yaml` file
2. The tuning profile that you want to apply to your deployment and is defined in the `/usr/share/foreman-installer/config/foreman.hiera/tuning/sizes/` directory
3. Optional: If you have configured a `/etc/foreman-installer/custom-hiera.yaml` file, Satellite applies these configuration settings.

Note that the configuration settings that are defined in the `/etc/foreman-installer/custom-hiera.yaml` file override the configuration settings that are defined in the tuning profiles.

Therefore, before applying a tuning profile, you must compare the configuration settings that are defined in the default tuning profile in `/usr/share/foreman-installer/config/foreman.hiera/tuning/common.yaml`, the tuning profile that you want to apply and your `/etc/foreman-installer/custom-hiera.yaml` file, and remove any duplicated configuration from the `/etc/foreman-installer/custom-hiera.yaml` file.

### default
- Number of managed hosts: 0-5000
- RAM: 20G
- Number of CPU cores: 4

### medium
- Number of managed hosts: 5001-10000
- RAM: 32G
- Number of CPU cores: 8

### large
- Number of managed hosts: 10001-20000
- RAM: 64G
- Number of CPU cores: 16

### extra-large
- Number of managed hosts: 20001-60000
- RAM: 128G
- Number of CPU cores: 32

### extra-extra-large
- Number of managed hosts: 60000+
- RAM: 256G
- Number of CPU cores: 48+
1. Optional: If you have configured the `custom-hiera.yaml` file on Satellite Server, back up the `/etc/foreman-installer/custom-hiera.yaml` file to `custom-hiera.original`. You can use the backup file to restore the `/etc/foreman-installer/custom-hiera.yaml` file to its original state if it becomes corrupted:

   ```bash
   # cp /etc/foreman-installer/custom-hiera.yaml \\
   /etc/foreman-installer/custom-hiera.original
   ```

2. Optional: If you have configured the `custom-hiera.yaml` file on Satellite Server, review the definitions of the default tuning profile in `/usr/share/foreman-installer/config/foreman.hiera/tuning/common.yaml` and the tuning profile that you want to apply in `/usr/share/foreman-installer/config/foreman.hiera/tuning/sizes/`. Compare the configuration entries against the entries in your `/etc/foreman-installer/custom-hiera.yaml` file and remove any duplicated configuration settings in your `/etc/foreman-installer/custom-hiera.yaml` file.

3. Enter the `satellite-installer` command with the `--tuning` option for the profile that you want to apply. For example, to apply the medium tuning profile settings, enter the following command:

   ```bash
   # satellite-installer --tuning medium
   ```
CHAPTER 5. CONFIGURING SATELLITE SERVER WITH EXTERNAL SERVICES

If you do not want to configure the DNS, DHCP, and TFTP services on Satellite Server, use this section to configure your Satellite Server to work with external DNS, DHCP and TFTP services.

5.1. CONFIGURING SATELLITE SERVER WITH EXTERNAL DNS

You can configure Satellite Server with external DNS. Satellite Server uses the `nsupdate` utility to update DNS records on the remote server.

To make any changes persistent, you must enter the `satellite-installer` command with the options appropriate for your environment.

Prerequisites

- You must have a configured external DNS server.

Procedure

1. Unlock packages to enable installation of new packages:
   ```bash
   # satellite-maintain packages unlock
   ```

2. Install BIND and the utility packages:
   ```bash
   # yum install bind bind-utils
   ```

3. Lock the packages:
   ```bash
   # satellite-maintain packages lock
   ```

4. Copy the `/etc/rndc.key` file from the external DNS server to Satellite Server:
   ```bash
   # scp root@dns.example.com:/etc/rndc.key /etc/rndc.key
   ```

5. Configure the ownership, permissions, and SELinux context:
   ```bash
   # restorecon -v /etc/rndc.key
   # chown -v root:named /etc/rndc.key
   # chmod -v 640 /etc/rndc.key
   ```

6. To test the `nsupdate` utility, add a host remotely:
   ```bash
   # echo -e "server DNS_IP_Address\n   update add aaa.virtual.lan 3600 IN A Host_IP_Address\n   send\n" | nsupdate -k /etc/rndc.key
   # nslookup aaa.virtual.lan DNS_IP_Address
   # echo -e "server DNS_IP_Address\n   update delete aaa.virtual.lan 3600 IN A Host_IP_Address\n   send\n" | nsupdate -k /etc/rndc.key
   ```
7. Assign the foreman-proxy user to the named group manually. Normally, satellite-installer ensures that the foreman-proxy user belongs to the named UNIX group, however, in this scenario Satellite does not manage users and groups, therefore you need to assign the foreman-proxy user to the named group manually.

```
# usermod -a -G named foreman-proxy
```

8. Enter the satellite-installer command to make the following persistent changes to the /etc/foreman-proxy/settings.d/dns.yml file:

```
# satellite-installer --foreman-proxy-dns=true \  
  --foreman-proxy-dns-managed=false \  
  --foreman-proxy-dns-provider=nsupdate \  
  --foreman-proxy-dns-server="DNS_IP_Address" \  
  --foreman-proxy-keyfile=/etc/rndc.key \  
  --foreman-proxy-dns-ttl=86400
```

9. Restart the foreman-proxy service:

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

10. Log in to Satellite Server web UI.

11. Navigate to Infrastructure > Capsules, locate the Satellite Server, and from the list in the Actions column, select Refresh.

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

### 5.2. CONFIGURING SATELLITE SERVER WITH EXTERNAL DHCP

To configure Satellite Server with external DHCP, you must complete the following procedures:

1. Section 5.2.1, “Configuring an External DHCP Server to Use with Satellite Server”

2. Section 5.2.2, “Configuring Satellite Server with an External DHCP Server”

#### 5.2.1. Configuring an External DHCP Server to Use with Satellite Server

To configure an external DHCP server to use with Satellite Server, on a Red Hat Enterprise Linux server, you must install the ISC DHCP Service and Berkeley Internet Name Domain (BIND) packages. You must also share the DHCP configuration and lease files with Satellite Server. The example in this procedure uses the distributed Network File System (NFS) protocol to share the DHCP configuration and lease files.

**NOTE**

If you use dnsmasq as an external DHCP server, enable the dhcp-no-override setting. This is required because Satellite creates configuration files on the TFTP server under the grub2/ subdirectory. If the dhcp-no-override setting is disabled, clients fetch the bootloader and its configuration from the root directory, which might cause an error.
1. On a Red Hat Enterprise Linux Server server, install the ISC DHCP Service and Berkeley Internet Name Domain (BIND) packages:

   ```
   # yum install dhcp bind
   ```

2. Generate a security token:

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

   As a result, a key pair that consists of two files is created in the current directory.

3. Copy the secret hash from the key:

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

4. Edit the `dhcpd` configuration file for all of the subnets and add the key. The following is an example:

   ```
   # 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;
   ```

   Note that the `option routers` value is the Satellite or Capsule IP address that you want to use with an external DHCP service.

5. Delete the two key files from the directory that they were created in.

6. On Satellite Server, define each subnet. Do not set DHCP Capsule for the defined Subnet yet. To prevent conflicts, set up the lease and reservation ranges separately. For example, if the lease range is 192.168.38.10 to 192.168.38.100, in the Satellite web UI define the reservation range as 192.168.38.101 to 192.168.38.250.

7. Configure the firewall for external access to the DHCP server:

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

8. On Satellite Server, determine the UID and GID of the `foreman` user:
On the DHCP server, create the `foreman` user and group with the same IDs as determined in a previous step:

```
# groupadd -g 990 foreman
# useradd -u 993 -g 990 -s /sbin/nologin foreman
```

To ensure that the configuration files are accessible, 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
```

Start the DHCP service:

```
# systemctl start dhcpd
```

Export the DHCP configuration and lease files using NFS:

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

Create directories for the DHCP configuration and lease files that you want to export using NFS:

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

To create mount points for the created directories, add the following line to the `/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
```

Mount the file systems in `/etc/fstab`:

```
# mount -a
```

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

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

Note that the IP address that you enter is the Satellite or Capsule IP address that you want to use with an external DHCP service.
17. Reload the NFS server:

   # exportfs -rva

18. Configure the firewall for the DHCP omapi port 7911:

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

19. Optional: Configure the firewall for external access to NFS. Clients are configured using NFSv3.

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

### 5.2.2. Configuring Satellite Server with an External DHCP Server

You can configure Satellite Server with an external DHCP server.

**Prerequisite**

- Ensure that you have configured an external DHCP server and that you have shared the DHCP configuration and lease files with Satellite Server. For more information, see Section 5.2.1, “Configuring an External DHCP Server to Use with Satellite Server”.

**Procedure**

1. Install the **nfs-utils** utility:

   # 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 the Remote Procedure Call (RPC) communication paths:

   # showmount -e DHCP_Server_FQDN
   # rpcinfo -p DHCP_Server_FQDN

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

   ```
   DHCP_Server_FQDN:/exports/etc/dhcp /mnt/nfs/etc/dhcp nfs
   ro,vers=3,auto,nosharecache,context="system_u:object_r:dhcp_etc_t:s0" 0 0
   ```
6. Mount the file systems on `/etc/fstab`:

   ```bash
   # mount -a
   ```

7. To verify that the `foreman-proxy` user can access the files that are shared over the network, display the DHCP configuration and lease files:

   ```bash
   # 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. Enter the `satellite-installer` command to make the following persistent changes to the `/etc/foreman-proxy/settings.d/dhcp.yml` file:

   ```bash
   # satellite-installer --foreman-proxy-dhcp=true \
   --foreman-proxy-dhcp-provider=remote_isc \
   --foreman-proxy-plugin-dhcp-remote-isc-dhcp-config /mnt/nfs/etc/dhcp/dhcpd.conf \
   --foreman-proxy-plugin-dhcp-remote-isc-dhcp-leases /mnt/nfs/var/lib/dhcpd/dhcpd.leases \
   --foreman-proxy-plugin-dhcp-remote-isc-key-name=omapi_key \
   --foreman-proxy-plugin-dhcp-remote-isc-key-secret=jNSE5YI3H1A8Oj/tkV4...A2ZOHb6zv315CkNAY7DMYYCj48Umw== \
   --foreman-proxy-plugin-dhcp-remote-isc-omapi-port=7911 \
   --enable-foreman-proxy-plugin-dhcp-remote-isc \
   --foreman-proxy-dhcp-server=DHCP_Server_FQDN
   ```

9. Restart the `foreman-proxy` service:

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

10. Log in to Satellite Server web UI.

11. Navigate to `Infrastructure > Capsules`, locate the Satellite Server, and from the list in the `Actions` column, select `Refresh`.

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

### 5.3. CONFIGURING SATELLITE SERVER WITH EXTERNAL TFTP

You can configure Satellite Server with external TFTP services.

**Procedure**

1. Create the TFTP directory for NFS:

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

2. In the `/etc/fstab` file, add the following line:
3. Mount the file systems in `/etc/fstab`:

```
# mount -a
```

4. Enter the `satellite-installer` command to make the following persistent changes to the `/etc/foreman-proxy/settings.d/tftp.yml` file:

```
# satellite-installer --foreman-proxy-tftp=true \ 
--foreman-proxy-tftp-root /mnt/nfs/var/lib/tftpboot
```

5. 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 the server that the TFTP service is running on:

```
# satellite-installer --foreman-proxy-tftp-servername=TFTP_Server_FQDN
```

6. Log in to Satellite Server web UI.

7. Navigate to **Infrastructure > Capsules**, locate the Satellite Server, and from the list in the **Actions** column, select **Refresh**.

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

### 5.4. CONFIGURING SATELLITE SERVER WITH EXTERNAL IDM DNS

When Satellite Server adds a DNS record for a host, it first determines which Capsule is providing DNS for that domain. It then communicates with the Capsule that is configured to provide DNS service for your deployment and adds the record. The hosts are not involved in this process. Therefore, you must install and configure the IdM client on the Satellite or Capsule that is currently configured to provide a DNS service for the domain you want to manage using the IdM server.

Satellite Server can be configured to use a Red Hat Identity Management (IdM) server to provide DNS service. For more information about Red Hat Identity Management, see the *Linux Domain Identity, Authentication, and Policy Guide*.

To configure Satellite Server to use a Red Hat Identity Management (IdM) server to provide DNS service, use one of the following procedures:

- Section 5.4.1, “Configuring Dynamic DNS Update with GSS-TSIG Authentication”
- Section 5.4.2, “Configuring Dynamic DNS Update with TSIG Authentication”

To revert to internal DNS service, use the following procedure:

- Section 5.4.3, “Reverting to Internal DNS Service”
NOTE

You are not required to use Satellite Server to manage DNS. When you are using the realm enrollment feature of Satellite, where provisioned hosts are enrolled automatically to IdM, the `ipa-client-install` script creates DNS records for the client. Configuring Satellite Server with external IdM DNS and realm enrollment are mutually exclusive. For more information about configuring realm enrollment, see External Authentication for Provisioned Hosts in Administering Red Hat Satellite.

5.4.1. Configuring Dynamic DNS Update with GSS-TSIG Authentication

You can configure the IdM server to use the generic security service algorithm for secret key transaction (GSS-TSIG) technology defined in RFC3645. To configure the IdM server to use the GSS-TSIG technology, you must install the IdM client on the Satellite Server base operating system.

Prerequisites

- You must ensure the IdM server is deployed and the host-based firewall is configured correctly. For more information, see Port Requirements in the Linux Domain Identity, Authentication, and Policy Guide.
- You must contact the IdM server administrator to ensure that you obtain an account on the IdM server with permissions to create zones on the IdM server.
- You must confirm whether Satellite Server or Capsule Server is configured to provide DNS service for your deployment.
- You must configure DNS, DHCP and TFTP services on the base operating system of either the Satellite or Capsule that is managing the DNS service for your deployment.
- You must create a backup of the answer file. You can use the backup to restore the answer file to its original state if it becomes corrupted. For more information, see Configuring Satellite Server.

Procedure

To configure dynamic DNS update with GSS-TSIG authentication, complete the following steps:

Creating a Kerberos Principal on the IdM Server

1. Obtain a Kerberos ticket for the account obtained from the IdM administrator:

   ```
   # kinit idm_user
   ```

2. Create a new Kerberos principal for Satellite Server to use to authenticate on the IdM server.

   ```
   # ipa service-add satellite.example.com
   ```

Installing and Configuring the IdM Client

1. On the base operating system of either the Satellite or Capsule that is managing the DNS service for your deployment, install the `ipa-client` package:

   ```
   # satellite-maintain packages install ipa-client
   ```
2. Configure the IdM client by running the installation script and following the on-screen prompts:

   # ipa-client-install

3. Obtain a Kerberos ticket:

   # kinit admin

4. Remove any preexisting **keytab**:

   # rm /etc/foreman-proxy/dns.keytab

5. Obtain the **keytab** for this system:

   # ipa-getkeytab -p capsule/satellite.example.com@EXAMPLE.COM
   -s idm1.example.com -k /etc/foreman-proxy/dns.keytab

   **NOTE**

   When adding a keytab to a standby system with the same host name as the original system in service, add the `r` option to prevent generating new credentials and rendering the credentials on the original system invalid.

6. For the **dns.keytab** file, set the group and owner to **foreman-proxy**:

   # chown foreman-proxy:foreman-proxy /etc/foreman-proxy/dns.keytab

7. Optional: To verify that the **keytab** file is valid, enter the following command:

   # kinit -kt /etc/foreman-proxy/dns.keytab \
   capsule/satellite.example.com@EXAMPLE.COM

**Configuring DNS Zones in the IdM web UI**

1. Create and configure the zone that you want to manage:

   a. Navigate to **Network Services > DNS > DNS Zones**.

   b. Select **Add** and enter the zone name. For example, **example.com**.

   c. Click **Add and Edit**

   d. Click the Settings tab and in the **BIND update policy** box, add the following to the semi-colon separated list:

      grant capsule/047 satellite.example.com@EXAMPLE.COM wildcard * ANY;

   e. Set **Dynamic update** to **True**.

   f. Enable **Allow PTR sync**.

   g. Click **Save** to save the changes.
2. Create and configure the reverse zone:
   a. Navigate to **Network Services > DNS > DNS Zones.**
   b. Click **Add.**
   c. Select **Reverse zone IP network** and add the network address in CIDR format to enable reverse lookups.
   d. Click **Add and Edit**
   e. Click the **Settings** tab and in the **BIND update policy** box, add the following to the semi-colon separated list:
      
      ```
      grant capsule\047satellite.example.com@EXAMPLE.COM wildcard * ANY;
      ```
   f. Set **Dynamic update** to **True.**
   g. Click **Save** to save the changes.

**Configuring the Satellite or Capsule Server that Manages the DNS Service for the Domain**

1. Use the **satellite-installer** command to configure the Satellite or Capsule that manages the DNS Service for the domain:
   - On Satellite, enter the following command:
     ```
     satellite-installer --scenario satellite \
     --foreman-proxy-dns=true \
     --foreman-proxy-dns-managed=true \
     --foreman-proxy-dns-provider=nsupdate_gss \
     --foreman-proxy-dns-server="idm1.example.com" \
     --foreman-proxy-dns-tsig-principal="capsule/satellite.example.com@EXAMPLE.COM" \
     --foreman-proxy-dns-tsig-keytab=/etc/foreman-proxy/dns.keytab \
     --foreman-proxy-dns-reverse="55.168.192.in-addr.arpa" \
     --foreman-proxy-dns-zone=example.com \
     --foreman-proxy-dns-ttl=86400
     ```
   - On Capsule, enter the following command:
     ```
     satellite-installer --scenario capsule \
     --foreman-proxy-dns=true \
     --foreman-proxy-dns-managed=true \
     --foreman-proxy-dns-provider=nsupdate_gss \
     --foreman-proxy-dns-server="idm1.example.com" \
     --foreman-proxy-dns-tsig-principal="capsule/satellite.example.com@EXAMPLE.COM" \
     --foreman-proxy-dns-tsig-keytab=/etc/foreman-proxy/dns.keytab \
     --foreman-proxy-dns-reverse="55.168.192.in-addr.arpa" \
     --foreman-proxy-dns-zone=example.com \
     --foreman-proxy-dns-ttl=86400
     ```
   2. Restart the Satellite or Capsule’s Proxy Service.
      ```
      # systemctl restart foreman-proxy
      ```
After you run the `satellite-installer` command to make any changes to your Capsule configuration, you must update the configuration of each affected Capsule in the Satellite web UI.

### Updating the Configuration in the Satellite web UI

1. Navigate to **Infrastructure > Capsules**, locate the Satellite Server, and from the list in the **Actions** column, select **Refresh**.

2. Configure the domain:
   a. Navigate to **Infrastructure > Domains** and select the domain name.
   b. In the **Domain** tab, ensure **DNS Capsule** is set to the Capsule where the subnet is connected.

3. Configure the subnet:
   a. Navigate to **Infrastructure > Subnets** and select the subnet name.
   b. In the **Subnet** tab, set **IPAM** to **None**.
   c. In the **Domains** tab, select the domain that you want to manage using the IdM server.
   d. In the **Capsules** tab, ensure **Reverse DNS Capsule** is set to the Capsule where the subnet is connected.
   e. Click **Submit** to save the changes.

### 5.4.2. Configuring Dynamic DNS Update with TSIG Authentication

You can configure an IdM server to use the secret key transaction authentication for DNS (TSIG) technology that uses the `rndc.key` key file for authentication. The TSIG protocol is defined in [RFC2845](https://tools.ietf.org/html/rfc2845).

**Prerequisites**

- You must ensure the IdM server is deployed and the host-based firewall is configured correctly. For more information, see **Port Requirements** in the *Linux Domain Identity, Authentication, and Policy Guide*.

- You must obtain **root** user access on the IdM server.

- You must confirm whether Satellite Server or Capsule Server is configured to provide DNS service for your deployment.

- You must configure DNS, DHCP and TFTP services on the base operating system of either the Satellite or Capsule that is managing the DNS service for your deployment.

- You must create a backup of the answer file. You can use the backup to restore the answer file to its original state if it becomes corrupted. For more information, see **Configuring Satellite Server**.

**Procedure**

To configure dynamic DNS update with TSIG authentication, complete the following steps:

**Enabling External Updates to the DNS Zone in the IdM Server**
1. On the IdM Server, add the following to the top of the `/etc/named.conf` file:

```plaintext
########################################################################
include "/etc/rndc.key";
controls {
  inet _IdM_Server_IP_Address_ port 953 allow {_Satellite_IP_Address_;} keys { "rndc-key"; }
};
########################################################################
```

2. Reload the named service to make the changes take effect:

```plaintext
# systemctl reload named
```

3. In the IdM web UI, navigate to Network Services > DNS > DNS Zones and click the name of the zone. In the Settings tab, apply the following changes:
   a. Add the following in the BIND update policy box:

   ```plaintext
   grant "rndc-key" zonesub ANY;
   ```
   b. Set Dynamic update to True.
   c. Click Update to save the changes.

4. Copy the `/etc/rndc.key` file from the IdM server to the base operating system of your Satellite Server. Enter the following command:

   ```plaintext
   # scp /etc/rndc.key root@satellite.example.com:/etc/rndc.key
   ```

5. To set the correct ownership, permissions, and SELinux context for the `rndc.key` file, enter the following command:

   ```plaintext
   # restorecon -v /etc/rndc.key
   # chown -v root:named /etc/rndc.key
   # chmod -v 640 /etc/rndc.key
   ```

6. Assign the foreman-proxy user to the named group manually. Normally, satellite-installer ensures that the foreman-proxy user belongs to the named UNIX group, however, in this scenario Satellite does not manage users and groups, therefore you need to assign the foreman-proxy user to the named group manually.

   ```plaintext
   # usermod -a -G named foreman-proxy
   ```

7. On Satellite Server, enter the following satellite-installer command to configure Satellite to use the external DNS server:

   ```plaintext
   # satellite-installer --scenario satellite \
   --foreman-proxy-dns=true \ 
   --foreman-proxy-dns-managed=false \ 
   --foreman-proxy-dns-provider=nsupdate \ 
   ```
Testing External Updates to the DNS Zone in the IdM Server

1. Ensure that the key in the `/etc/rndc.key` file on Satellite Server is the same key file that is used on the IdM server:

   ```bash
   key "rndc-key" {
     algorithm hmac-md5;
     secret "secret-key==";
   };
   ```

2. On Satellite Server, create a test DNS entry for a host. For example, host `test.example.com` with an A record of `192.168.25.20` on the IdM server at `192.168.25.1`.

   ```bash
   # echo -e "server 192.168.25.1\n   update add test.example.com 3600 IN A 192.168.25.20\n   send\n" | nsupdate -k /etc/rndc.key
   ```

3. On Satellite Server, test the DNS entry:

   ```bash
   # nslookup test.example.com 192.168.25.1
   Server: 192.168.25.1
   Address: 192.168.25.1#53
   
   Name: test.example.com
   Address: 192.168.25.20
   ```

4. To view the entry in the IdM web UI, navigate to Network Services > DNS > DNS Zones. Click the name of the zone and search for the host by name.

5. If resolved successfully, remove the test DNS entry:

   ```bash
   # echo -e "server 192.168.25.1\n   update delete test.example.com 3600 IN A 192.168.25.20\n   send\n" | nsupdate -k /etc/rndc.key
   ```

6. Confirm that the DNS entry was removed:

   ```bash
   # nslookup test.example.com 192.168.25.1
   ```

   The above `nslookup` command fails and returns the `SERVFAIL` error message if the record was successfully deleted.

5.4.3. Reverting to Internal DNS Service

You can revert to using Satellite Server and Capsule Server as your DNS providers. You can use a backup of the answer file that was created before configuring external DNS, or you can create a backup of the answer file. For more information about answer files, see Configuring Satellite Server.

Procedure
On the Satellite or Capsule Server that you want to configure to manage DNS service for the domain, complete the following steps:

**Configuring Satellite or Capsule as a DNS Server**

- If you have created a backup of the answer file before configuring external DNS, restore the answer file and then enter the `satellite-installer` command:

  ```bash
  # satellite-installer
  ``

- If you do not have a suitable backup of the answer file, create a backup of the answer file now. To configure Satellite or Capsule as DNS server without using an answer file, enter the following `satellite-installer` command on Satellite and each affected Capsule:

  ```bash
  # satellite-installer \
  --foreman-proxy-dns=true \
  --foreman-proxy-dns-managed=true \
  --foreman-proxy-dns-provider=nsupdate \
  --foreman-proxy-dns-server="127.0.0.1" \
  --foreman-proxy-dns-tsig-principal="foremanproxy/satellite.example.com@EXAMPLE.COM" \
  --foreman-proxy-dns-tsig-keytab=/etc/foreman-proxy/dns.keytab
  ``

For more information, see Configuring DNS, DHCP, and TFTP on Capsule Server.

After you run the `satellite-installer` command to make any changes to your Capsule configuration, you must update the configuration of each affected Capsule in the Satellite web UI.

**Updating the Configuration in the Satellite web UI**

1. Navigate to **Infrastructure > Capsules**.

2. For each Capsule that you want to update, from the **Actions** list, select **Refresh**.

3. Configure the domain:
   a. Navigate to **Infrastructure > Domains** and click the domain name that you want to configure.
   b. In the **Domain** tab, set **DNS Capsule** to the Capsule where the subnet is connected.

4. Configure the subnet:
   a. Navigate to **Infrastructure > Subnets** and select the subnet name.
   b. In the **Subnet** tab, set **IPAM** to **DHCP** or **Internal DB**.
   c. In the **Domains** tab, select the domain that you want to manage using Satellite or Capsule.
   d. In the **Capsules** tab, set **Reverse DNS Capsule** to the Capsule where the subnet is connected.
   e. Click **Submit** to save the changes.
APPENDIX A. APPLYING CUSTOM CONFIGURATION TO RED HAT SATELLITE

When you install and configure Satellite for the first time using `satellite-installer`, you can specify that the DNS and DHCP configuration files are not to be managed by Puppet using the installer flags `--foreman-proxy-dns-managed=false` and `--foreman-proxy-dhcp-managed=false`. If these flags are not specified during the initial installer run, rerunning of the installer overwrites all manual changes, for example, rerun for upgrade purposes. If changes are overwritten, you must run the restore procedure to restore the manual changes. For more information, see Restoring Manual Changes Overwritten by a Puppet Run.

To view all installer flags available for custom configuration, run `satellite-installer --scenario satellite --full-help`. Some Puppet classes are not exposed to the Satellite installer. To manage them manually and prevent the installer from overwriting their values, specify the configuration values by adding entries to configuration file `/etc/foreman-installer/custom-hiera.yaml`. This configuration file is in YAML format, consisting of one entry per line in the format of `<puppet class>::<parameter name>: <value>`. Configuration values specified in this file persist across installer reruns.

Common examples include:

- For Apache, to set the ServerTokens directive to only return the Product name:
  ```yaml
  apache::server_tokens: Prod
  ```

- To turn off the Apache server signature entirely:
  ```yaml
  apache::server_signature: Off
  ```

The Puppet modules for the Satellite installer are stored under `/usr/share/foreman-installer/modules`. Check the `.pp` files (for example: `moduleName/manifests/example.pp`) to look up the classes, parameters, and values. Alternatively, use the `grep` command to do keyword searches.

Setting some values may have unintended consequences that affect the performance or functionality of Red Hat Satellite. Consider the impact of the changes before you apply them, and test the changes in a non-production environment first. If you do not have a non-production Satellite environment, run the Satellite installer with the `--noop` and `--verbose` options. If your changes cause problems, remove the offending lines from `custom-hiera.yaml` and rerun the Satellite installer. If you have any specific questions about whether a particular value is safe to alter, contact Red Hat support.
APPENDIX B. RESTORING MANUAL CHANGES OVERWRITTEN BY A PUPPET RUN

If your manual configuration has been overwritten by a Puppet run, you can restore the files to the previous state. The following example shows you how to restore a DHCP configuration file overwritten by a Puppet run.

Procedure

1. Copy the file you intend to restore. This allows you to compare the files to check for any mandatory changes required by the upgrade. This is not common for DNS or DHCP services.

   ```bash
   # cp /etc/dhcp/dhcpd.conf /etc/dhcp/dhcpd.backup
   ```

2. Check the log files to note down the md5sum of the overwritten file. For example:

   ```bash
   # journalctl -xe
   ...
   /Stage[main]/Dhcp/File[/etc/dhcp/dhcpd.conf]: Filebucketed /etc/dhcp/dhcpd.conf to puppet with sum 622d9820b8e764ab124367c68f5fa3a1
   ...
   ```

3. Restore the overwritten file:

   ```bash
   # puppet filebucket restore --local --bucket /var/lib/puppet/clientbucket /etc/dhcp/dhcpd.conf 622d9820b8e764ab124367c68f5fa3a1
   ```

4. Compare the backup file and the restored file, and edit the restored file to include any mandatory changes required by the upgrade.