Red Hat Satellite 5.6

Client Configuration Guide

Configuring, registering, and updating your Red Hat Enterprise Linux clients with Red Hat Satellite

Edition 1

Last Updated: 2017-09-26

Red Hat Satellite 5.6 Client Configuration Guide

Configuring, registering, and updating your Red Hat Enterprise Linux clients with Red Hat Satellite Edition 1

John Ha
Red Hat Engineering Content Services

Lana Brindley
Red Hat Engineering Content Services

Daniel Macpherson
Red Hat Engineering Content Services

Athene Chan
Red Hat Engineering Content Services

David O'Brien
Red Hat Engineering Content Services

Legal Notice

Abstract

This guide covers how to properly configure Red Hat Enterprise Linux systems to register with and download updates from Red Hat Satellite. It covers how to register systems, how to deploy the latest packages, and other topics related to server and client synchronization. For further information, see the Red Hat Satellite Getting Started Guide and the Red Hat Satellite Installation Guide.

Table of Contents

PREFACE

Red Hat Network (https://access.redhat.com/home) provides system-level support and management of Red Hat systems and networks. It brings together the tools, services, and information repositories needed to maximize the reliability, security, and performance of Red Hat systems. To use Red Hat Network, system administrators register software and hardware profiles, known as *System Profiles,* of their client systems with Red Hat Network. When a client system requests package updates, only the applicable packages for the client are returned.

Red Hat Satellite allows organizations to use the benefits of Red Hat Network without having to provide public Internet access to their servers or other client systems. System profiles are stored locally on the Satellite server. The Satellite website is served from a local web server and is only accessible to systems that can reach the Satellite server. All package management tasks, including errata updates, are performed through the Satellite server.

Red Hat Satellite provides a solution for organizations that require absolute control over and privacy of the maintenance and package deployment of their servers. It allows Red Hat Network customers the greatest flexibility and power in keeping systems secure and updated. Modules can be added to the Satellite server to provide extra functionality.

This document provides guidance on how to configure your Red Hat Enterprise Linux systems to stay up to date using Red Hat Satellite.

CHAPTER 1. INTRODUCTION

This guide is designed to help users of Red Hat Satellite and Red Hat Satellite Proxy to configure their client systems.

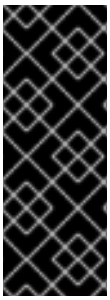
By default, all Red Hat Network client applications are configured to communicate with central Red Hat Network servers. When clients connect to a Red Hat Satellite or Red Hat Satellite Proxy instead, the default settings change. This document is intended to assist by offering mass reconfiguration steps which will help large enterprise environments, containing hundreds or thousands of systems, address the default setting changes.

Due to the complexity of this undertaking, customers can use a pre-populated script that automates many of the tasks necessary to access their Satellite or Satellite Proxy server; see the *Getting Started Guide* for details. Red Hat believes that understanding the implications of these changes is helpful and therefore describes the manual steps for reconfiguration in the opening chapters. Use your best judgement in determining the ideal solution for your organization.

Although many of the commands provided within this guide can be applied as they appear, it is impossible to predict all potential network configurations adopted by customers. Therefore, Red Hat encourages you to use these commands as references that must take into account your organization's individual settings.

CHAPTER 2. CONFIGURING CLIENT APPLICATIONS

In order to use most enterprise-class features of Red Hat Network, such as registering with Red Hat Satellite, configuration of the latest client applications is required. Obtaining these applications before the client has registered with Red Hat Network can be difficult. This paradox is especially problematic for customers migrating large numbers of older systems to Red Hat Network. This chapter identifies techniques to resolve this dilemma.

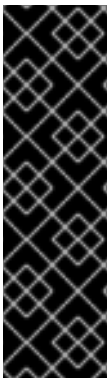


IMPORTANT

Red Hat strongly recommends that clients connected to a Red Hat Proxy Server or Red Hat Satellite Server be running the latest update of Red Hat Enterprise Linux to ensure proper connectivity.

Additionally, if client firewalls are configured, ports 80 and 443 should be open for proper functionality with Red Hat Network.

Not every customer must connect securely to a Red Hat Satellite or Red Hat Proxy within their organization, nor build and deploy a GPG key for custom packages, but every customer who uses these products must reconfigure the **Red Hat Update Agent (up2date)** and possibly the **Red Hat Network Registration Client (rhn_register)** to redirect it from Red Hat Network to their Satellite or Proxy.



IMPORTANT

By default, the `yum` command on Red Hat Enterprise Linux 5 and 6 uses SSL for communication with remote repositories. Consequently, you should ensure that firewalls allow connections over port 443.

To bypass SSL, change the value of `serverURL` from `https` to `http` in the `/etc/sysconfig/rhn/up2date` file. Similarly, to use Red Hat Network's Monitoring feature and probes requiring the Red Hat Network Monitoring Daemon, client systems must allow connections on port 4545 (or port 22, if it is using `sshd` instead).

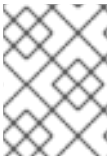
By default, `rhn_register` refers to the main Red Hat Network servers. You need to reconfigure client systems to see Red Hat Satellite or Red Hat Proxy.

The latest versions of **Red Hat Update Agent** can be configured to accommodate several Red Hat Satellite servers, thereby providing failover protection in case the primary server is inaccessible. See Section 2.5, “Implementing Server Failover” for instructions on enabling this feature.

The following sections describe different methods of configuring the client systems to access your Red Hat Satellite or Proxy. See the Red Hat Satellite *Getting Started Guide* for information about scripting configuration commands.

2.1. DEPLOYING THE LATEST RED HAT NETWORK CLIENT PACKAGES

The **Package Updater** (`pup`), `yum`, the yum Red Hat Network Plugin (`yum-rhn-plugin`) and the **Red Hat Network Registration Client** (`rhn_register`) on Red Hat Enterprise Linux 5 and 6 are prerequisites for using much of Red Hat Network's enterprise functionality. It is crucial to install them on client systems before attempting to use Red Hat Proxy or Red Hat Satellite in your environment.



NOTE

Systems running Red Hat Enterprise Linux 5 or 6 must be registered as part of the **firstboot** process after installation or by using the **rhn_register** command.

2.1.1. The Package Updater Applet

Red Hat Enterprise Linux 5 and later feature a running program on the graphical desktop panel that periodically checks for updates from the Red Hat Network or Satellite server and alerts users when updates are available.

Figure 2.1. Package Updater Applet

The Package Updater Applet stays in the notification tray of the desktop panel and periodically checks for updates. The applet also facilitates various package maintenance tasks; click the notification icon and choose from the following actions:

- Refresh: check Red Hat Network or Satellite for new updates.
- View Updates: launch the Package Updater application and display any available updates in more detail, and configure the updates to specifications.
- Apply Updates: download and install all updated packages.
- Quit: close the applet.

2.2. REGISTERING CLIENTS WITH RED HAT SATELLITE SERVER

The following procedure describes how to use the **rhn_register** command to register a system with Red Hat Satellite. Ensure you replace the example host names and domain names with those that apply to your configuration.

Procedure 2.1. To Use rhn_register to Register a System with Red Hat Satellite:

1. Change into the **/usr/share/rhn/** directory and download the SSL certificate to the client:

```
# cd /usr/share/rhn/
# wget http://satellite.example.com/pub/RHN-ORG-TRUSTED-SSL-CERT
```

2. Edit the **/etc/sysconfig/rhn/up2date** file and ensure that it contains the following entries:

```
serverURL=https://satellite.example.com/XMLRPC
noSSLServerURL=http://satellite.example.com/XMLRPC
sslCACert=/usr/share/rhn/RHN-ORG-TRUSTED-SSL-CERT
```

3. Use the **rhn_register** command to register the machine:

```
# rhn_register
```

2.3. USING ACTIVATION KEYS TO REGISTER CLIENTS WITH RED HAT SATELLITE

Red Hat recommends using activation keys for registering and configuring client systems that access Red Hat Proxy or Red Hat Satellite. You can use activation keys to register, entitle, and subscribe multiple systems in a single operation. See the relevant section in the Red Hat Satellite *Getting Started Guide* for more information about activation keys.

Procedure 2.2. To Use Activation Keys to Register a System with Red Hat Satellite:

1. Generate an activation key. (See "Using Activation Keys" in the Red Hat Satellite *Getting Started Guide*.)
2. Import custom GPG keys.
3. Download and install the SSL Certificate RPM from the **/pub/** directory of the Red Hat Proxy or Red Hat Satellite. For example (update the URL to suit your environment):

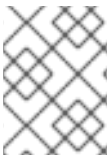
```
# rpm -Uvh http://satellite.example.com/pub/rhn-org-trusted-ssl-
cert-1.0-1.noarch.rpm
```

4. Register the system with the Red Hat Proxy or Red Hat Satellite:

```
# rhnreg_ks --activationkey mykey --serverUrl
https://satellite.example.com/XMLRPC
```

Alternatively, most of the above steps can be combined in a shell script that includes the following lines:

```
wget -0 - http://satellite.example.com/pub/bootstrap.sh | bash
&& rhnreg_ks --activation-key my_key --serverUrl
https://satellite.example.com/XMLRPC
```



NOTE

This command has been split into multiple lines for print and PDF purposes but should be typed as one line at a shell prompt.

The bootstrap script, generated at installation and available for both Red Hat Satellite Server and Red Hat Proxy Server, is such a script. The script and the Red Hat Network Bootstrap that generates it are discussed in detail in the *Getting Started Guide*.

2.4. UPDATING THE CONFIGURATION FILES MANUALLY

As an alternative to the GUI interface described in the previous section, users may also reconfigure the **Red Hat Update Agent** by editing the application's configuration file.

To configure the **Update Agent** on client systems that connect to Red Hat Proxy or Satellite, edit the values of the **serverURL** and **noSSLServerURL** settings in the **/etc/sysconfig/rhn/up2date** configuration file (as root). Replace the default Red Hat Network URL with the fully qualified domain name (FQDN) of the Proxy or Satellite. For example:

```
serverURL[comment]=Remote server URL
serverURL=https://your_primary.your_domain.com/XMLRPC

noSSLServerURL[comment]=Remote server URL without SSL
noSSLServerURL=http://your_primary.your_domain.com/XMLRPC
```

WARNING

The **httpProxy** setting in **/etc/sysconfig/rhn/up2date** does *not* refer to the Red Hat Proxy. It is used to configure an optional HTTP proxy for the client. With a Red Hat Proxy in place, the **httpProxy** setting must be blank (not set to any value).

2.5. IMPLEMENTING SERVER FAILOVER

Procedure 2.3. To Implement Server Failover:

1. Ensure that you are running Red Hat Enterprise Linux 5 or 6, or for Red Hat Enterprise Linux 3 or 4, the latest version of **up2date**.

2. Manually add the secondary servers to the **serverURL** and **noSSLServerURL** settings in the **/etc/sysconfig/rhn/up2date** configuration file (as root).

3. Add the fully qualified domain names (FQDN) of Red Hat Proxy or Red Hat Satellite immediately after the primary server, separated by a semicolon (;). Your client will attempt to connect to these servers in the order provided here. Include as many servers as necessary. For example:

```
serverURL[comment]=Remote server URL
serverURL=https://satellite.example.com/XMLRPC;
https://your_secondary.your_domain.com/XMLRPC;

noSSLServerURL[comment]=Remote server URL without SSL
noSSLServerURL=http://satellite.example.com/XMLRPC;
http://your_secondary.your_domain.com/XMLRPC;
```

CHAPTER 3. REGISTERING RED HAT SYSTEMS WITH RED HAT NETWORK

Red Hat Enterprise Linux features an application called **rhn_register**. This application works with the **yum**-based Red Hat Network Hosted and Red Hat Satellite client called **Package Updater** (or **pup**) that replaces **up2date**.

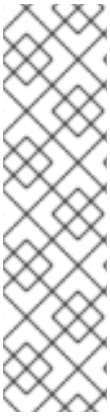
The **rhn_register** application normally runs as part of the **firstboot** configuration process just after installation. The first time a newly-installed Red Hat Enterprise Linux 5 or 6 system is booted, the **firstboot** command runs **rhn_register** to register the system with Red Hat Network.

You also need to use the **rhn_register** command in the following circumstances:

- You skipped the registration process during the initial installation
- You are reinstalling the system
- You are moving the system to a new account

3.1. USING THE GRAPHICAL INTERFACE TO REGISTER WITH RED HAT NETWORK

This section describes how to use the graphical version of the Package Updater to register your system with Red Hat Network.



NOTE

If the system has not yet been registered, the **/etc/sysconfig/rhn/systemid** file should not exist. In this case, when you run the **Package Updater** command, it triggers the **rhn_register** command.

If this file does exist, you might see a warning that the system is already registered, and that if you continue, it might produce duplicate entries on Red Hat Network. If you are certain you want to reregister with the possibility of duplicating the system on Red Hat Network, continue with the registration. If not, use **rhnreg_ks** and activation keys instead.

Procedure 3.1. To Use the GUI to Register with Red Hat Network:

1. On the main panel, click **System** → **Administration** → **Red Hat Network Registration** and enter the root password when prompted.

   The **Registering for Software Updates** page summarizes the steps involved in the registration process. To learn more about the benefits of Hosted and Satellite, click **Why Should I Connect to Red Hat Network.** Otherwise, click **Forward** to continue.

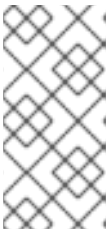
2. Use the **Choose an Update Location** page to select the source of your software updates - either Red Hat Network Hosted, or Satellite Server or Proxy Server. For Satellite or Proxy, select the associated radio button and enter the URL of your Satellite or Proxy into the **Red Hat Network Location** field.

   If you connect to the internet through an HTTP Proxy, click **Advanced Network Configuration** and enter the details for your HTTP proxy. If your proxy requires

authentication, enter the user name and password here, and then click **Close** to return to the **Choose an Update Location** page. Click **Forward** to continue.

3. Use the **Enter Your Account Information** page to enter your Red Hat Network login information. If you do not have an account and your organization has one, ask the Organization Administrator to create an account for you. Otherwise, you might not be associated with your organization or its resources. Click **Forward** to continue.

4. Use the **Create Your System Profile** page to select a profile name for the system you are registering. The default profile name is the system's host name, but you can change it to any valid profile name. You can also select whether to report hardware and package information to Red Hat Network. It is recommended that you report this information because it allows Red Hat Network to automatically subscribe your system to the base and child channels most appropriate to your system. You can click **View Hardware Profile** or **View Package Profile** to inspect the information that **rhn_register** uploads to Red Hat Network or Satellite in this step.



NOTE

This automatic registration does not automatically subscribe your system to optional child channels, such as the Red Hat Network Tools channel. If you want to register a system and automatically subscribe it to a set of channels, consider using a kickstart profile or **rhnreg_ks** and activation keys.

5. Click **Forward** to display the **Review System Subscription Details** page, which displays the base and child channel information to which your system has been subscribed. Review the channels, and then click **Forward** to continue.

6. The **Finish Setting Up Software Updates** page indicates that you have successfully registered a Red Hat Enterprise Linux system with Red Hat Network. A "package" icon appears in the upper right corner of your desktop when updates are available. Click the icon to apply available updates. Click **Finish** to exit the wizard.

NOTE

If you do not have any entitlements available for this system, this final page indicates that the registration has failed. This does not mean that the system profile has not been stored with Red Hat Network, only that you will not receive automatic updates without manual intervention. You can always log in to the Red Hat Network or Satellite Web interface and either purchase additional entitlements or get an entitlement from your Satellite administrator. Click **Exit software update setup** to exit the wizard.

Reinstalling the System

If you have already registered the system and the **/etc/sysconfig/rhn/systemid** file exists on the system, use a reactivation key. On the Satellite server, navigate to the system profile's **Details** → **Reactivation** page, which provides a means to create a reactivation key. Use this key with the **rhnreg_ks** to reregister the system without creating a duplicate entry in Red Hat Satellite.

3.1.1. Command-line Version of rhn_register

There is a command-line version of **rhn_register** that allows you to register your system for access to Red Hat Network or Red Hat Satellite without using a graphical desktop environment.

Type **rhn_register** on the command line to start the text-based version of **rhn_register.** If you are in shell terminal window and want to use the text-based version, type **rhn_register --nox** to prevent opening the graphical client.

The text-based version of **rhn_register** has the same configuration screens as the graphical version. Use the arrow keys on the keyboard to move left, right, up, or down and to highlight selections. Press the **Spacebar** key to select an option. Press the **Tab** key to move through different navigational elements such as text boxes, check boxes, and radio buttons.

CHAPTER 4. SSL INFRASTRUCTURE

For Red Hat Satellite customers, security concerns are of the utmost importance. One of the strengths of Red Hat Satellite is its ability to process every single request over Secure Sockets Layer, or SSL. To maintain this level of security, customers installing Red Hat Satellite within their infrastructures must generate custom SSL keys and certificates.

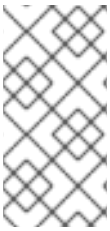
Manual creation and deployment of SSL keys and certificates can be quite involved. Both the Red Hat Proxy Server and the Red Hat Satellite Server allows users to build their own SSL keys and certificates based on their own private Certificate Authority (CA) during installation. In addition, a separate command line utility, the **Red Hat Satellite SSL Maintenance Tool** , exists for this purpose. Regardless, these keys and certificates must then be deployed to all systems within the managed infrastructure. In many cases, deployment of these SSL keys and certificates is automated. This chapter describes efficient methods for conducting all of these tasks.

NOTE

This chapter does not explain SSL in depth. The Red Hat Satellite **SSL Maintenance Tool** was designed to hide much of the complexity involved in setting up and maintaining the public-key infrastructure (PKI). For more information, see the relevant sections of the Red Hat Enterprise Linux *Deployment Guide*.

4.1. A BRIEF INTRODUCTION TO SSL

Secure Sockets Layer (SSL) is a protocol that enables client-server applications to pass information securely. SSL uses a system of public and private key pairs to encrypt communication passed between clients and servers. Public certificates can be left accessible, while private keys must be secured. It's the mathematical relationship (a digital signature) between a private key and its paired public certificate that makes this system work. Through this relationship, a connection of trust is established.



NOTE

SSL private keys and public certificates will be discussed throughout this document. Both can be referred to as keys, one public and one private. However, when discussing SSL, it is the convention to refer to the public half of an SSL key pair (or key set) as the SSL public certificate.

An organization's SSL infrastructure is generally made up of the following SSL keys and certificates:

- Certificate Authority (CA) SSL private key and public certificate: only one set per organization generally generated. The public certificate is digitally signed by its private key. The public certificate is distributed to every system.

- Web server SSL private key and public certificate: one set per application server. The public certificate is digitally signed by both its private key and the CA SSL private key. It is often referred to as a Web server's key *set*; this is because there is an intermediary SSL certificate request that is generated. The details of what this is used for are not important to this discussion. All three are deployed to a Red Hat Satellite Server.

The following is a scenario to help visualize the concept: An organization with one Red Hat Satellite Server and five Red Hat Proxy Servers will need to generate one CA SSL key pair and six Web server SSL key sets. A CA SSL public certificate is distributed to all systems and used by all clients to establish a connection to their respective upstream servers. Each server has its own SSL key set that is specifically tied to that server's host name and generated using its own SSL private key and the CA

SSL private key in combination. This establishes a digitally verifiable association between the Web server's SSL public certificate and the CA SSL key pair and server's private key. The Web server's key set cannot be shared with other web servers.

IMPORTANT

The most critical portion of this system is the CA SSL key pair. From that private key and public certificate an administrator can regenerate any Web server's SSL key set. This CA SSL key pair must be secured. It is highly recommended that once the entire Red Hat Satellite infrastructure of servers is set up and running, archive the SSL build directory generated by this tool and/or the installers onto separate media, write down the CA password, and secure the media and password in a safe place.

4.2. THE RED HAT SATELLITE SSL MAINTENANCE TOOL

Red Hat Satellite provides a command line tool to ease the management of the organization's secure infrastructure: the **Red Hat Satellite SSL Tool** , commonly known by its command **`rhn-ssl-tool`**. This tool is available as part of the **`spacewalk-certs-tools`** package. This package can be found within the software channels for the latest Red Hat Proxy Server and Red Hat Satellite Server (as well as the Red Hat Satellite Server ISO). The **Red Hat Satellite SSL Tool** enables organizations to generate their own Certificate Authority SSL key pair, as well as Web server SSL key sets (sometimes called *key pairs*).

This tool is only a build tool. It generates all of the SSL keys and certificates that are required. It also packages the files in RPM format for quick distribution and installation on all client machines. It does not deploy them. That is left to the administrator, or in many cases, automated by the Red Hat Satellite Server.

NOTE

The **`spacewalk-certs-tools`**, which contains **`rhn-ssl-tool`**, can be installed and run on any current Red Hat Enterprise Linux system with minimal requirements. This is offered as a convenience for administrators who want to manage their SSL infrastructure from their workstation or another system other than their Satellite or Proxy servers.

The **Red Hat Satellite SSL Tool** is required in the following situations:

- When updating the Certificate Authority (CA) public certificate.
- When installing a Red Hat Proxy Server 3.6 or later that connects to the central Red Hat Satellite Servers as its top-level service. The hosted service, for security reasons, cannot be a repository for the CA SSL key and certificate, which is private to the organization.
- When reconfiguring the Satellite or Proxy infrastructure to use SSL where it previously did not.
- When adding multiple Red Hat Satellite Servers to the Red Hat Satellite infrastructure. Consult with a Red Hat representative for instructions regarding this.

The **Red Hat Satellite SSL Tool** is *not* required in the following situations:

- During installation of a Red Hat Satellite Server. All SSL settings are configured during the installation process. The SSL keys and certificate are built and deployed automatically.
- During installation of a Red Hat Proxy Server 3.6 or later if connected to a Red Hat

Satellite Server 3.6 or later as its top-level service. The Red Hat Satellite Server contains all of the SSL information needed to configure, build and deploy the Red Hat Proxy Server's SSL keys and certificates.

The installation procedures for both the Red Hat Satellite Server and the Red Hat Proxy Server ensure the CA SSL public certificate is deployed to the **/pub** directory of each server. This public certificate is used by the client systems to connect to the Red Hat Satellite Server. See Section 4.3, "Deploying the CA SSL Public Certificate to Clients" for more information.

In summary, if the organization's Satellite or Proxy infrastructure deploys the latest version of Red Hat Satellite Server as its top-level service, there should be little need to use the **Red Hat Satellite SSL Tool**.

4.2.1. Generating SSL Certificates

The primary benefits of using the **Red Hat Satellite SSL Maintenance Tool** are security, flexibility, and portability. Security is achieved through the creation of distinct Web server SSL keys and certificates for each Red Hat Satellite server, all signed by a single Certificate Authority SSL key pair created by the organization. Flexibility is supplied by the tool's ability to work on any machine that has the spacewalk-certs-tools package installed. Portability exists in a build structure that can be stored anywhere for safe keeping and then installed whenever the need arises.

If the organization infrastructure's top-level Server is the most current Red Hat Satellite Server, the most that may be required is to restore the **ssl-build** tree from an archive to the **/root** directory and utilize the configuration tools provided within the Red Hat Satellite Server's website.

To make the best use of the **Red Hat Satellite SSL Maintenance Tool** , complete the following high-level tasks in the following order. See the remaining sections for the required details:

1. Install the spacewalk-certs-tools package on a system within the organization, perhaps but not necessarily the Red Hat Satellite Server or Red Hat Proxy Server.

2. Create a single Certificate Authority SSL key pair for the organization and install the resulting RPM or public certificate on all client systems. See Section 4.2.3, "Generating the Certificate Authority SSL Key Pair" for more information.

3. Create a Web server SSL key set for each of the Proxy and Satellite servers to be deployed and install the resulting RPM files on the Red Hat Satellite servers.

4. Restart the **httpd** service:

```
# service httpd restart
```

5. Back up the SSL *build tree* - consisting of the primary build directory and all subdirectories and files - to removable media, such as a CD or DVD. (Disk space requirements are insignificant.)

6. Verify and then store that archive in a safe location, such as the one described for backups in the *Additional Requirements* sections of either the Proxy or Satellite installation guide.

7. Record and secure the CA password for future use.

8. Delete the build tree from the build system for security purposes, but only after the entire Satellite infrastructure is in place and configured.

NOTE

When additional Web server SSL key sets are needed, restore the build tree on a system running the **Red Hat Satellite SSL Maintenance Tool** and repeat steps 3 through 7.

4.2.2. Red Hat Satellite SSL Maintenance Tool Options

The **Red Hat Satellite SSL Maintenance Tool** offers numerous command line options for generating Certificate Authority SSL key pair and managing your server SSL certificates and keys. The following command-line help options are available:

- **`rhn-ssl-tool --help`:** for general help.
- **`rhn-ssl-tool --gen-ca --help`:** for Certificate Authority help.
- **`rhn-ssl-tool --gen-server --help`:** for Web server help.

See the manual page (**`man rhn-ssl-tool`**) for more information.

4.2.3. Generating the Certificate Authority SSL Key Pair

Before creating the SSL key set required by the Web server, generate a Certificate Authority (CA) SSL key pair. A CA SSL public certificate is distributed to client systems of the Satellite or Proxy. The **Red Hat Satellite SSL Maintenance Tool** allows you to generate a CA SSL key pair if needed and reuse it for all subsequent Red Hat Satellite server deployments.

The build process automatically creates the key pair and public RPM for distribution to clients. All CA components are created in the build directory specified at the command line, typically **`/root/ssl-build`** (or **`/etc/sysconfig/rhn/ssl`** for older Satellite and Proxy servers). To generate a CA SSL key pair, run the following command.

IMPORTANT

Replace the example values with those appropriate for your organization.

```
# rhn-ssl-tool --gen-ca \
  --password=MY_CA_PASSWORD \
  --dir="/root/ssl-build" \
  --set-state="North Carolina" \
  --set-city="Raleigh" \
  --set-org="Example Inc." \
  --set-org-unit="SSL CA Unit"
```

This command generates the following relevant files in the specified build directory:

- **`RHN-ORG-PRIVATE-SSL-KEY`:** the CA SSL private key.
- **`RHN-ORG-TRUSTED-SSL-CERT`:** the CA SSL public certificate.
- **`rhn-org-trusted-ssl-cert-`*`VER-REL`*`.noarch.rpm`:** the RPM prepared for distribution to client systems.

This file contains the CA SSL public certificate (above) and installs it as **/usr/share/rhn/RHN-ORG-TRUSTED-SSL-CERT**

- **rhn-ca-openssl.cnf:** the SSL CA configuration file.
- **latest.txt:** lists the latest versions of the relevant files.

When this process is complete, distribute the RPM file to the client systems. See Section 4.3, "Deploying the CA SSL Public Certificate to Clients" for more information.

4.2.4. Generating Web Server SSL Key Sets

At this point, a CA SSL key pair should already be generated. However there is a likelihood of generating web server SSL key sets more frequently, especially if more than one Proxy or Satellite is deployed. A distinct set of SSL keys and certificates must be generated and installed for every distinct Satellite or Proxy server host name. The value for **--set-hostname** is therefore different for each server.

The server certificate build process works in a similar fashion to CA SSL key pair generation, with one exception: All server components are saved in subdirectories of the build directory. These subdirectories reflect the build system's machine name, such as **/root/ssl-build/MACHINE_NAME**. To generate a server certificate, run the following command.

IMPORTANT

Replace the example values with those appropriate for your organization.

The following is a single command. Ensure you enter it all on one line.

```
# rhn-ssl-tool --gen-server \
  --password=MY_CA_PASSWORD \
  --dir="/root/ssl-build" \
  --set-state="North Carolina" \
  --set-city="Raleigh"
  --set-org="Example Inc." \
  --set-org-unit="IS/IT" \
  --set-email="admin@example.com" \
  --set-hostname="rhnbox1.example.com"
```

This command generates the following relevant files in a machine-specific subdirectory of the build directory:

- **server.key:** the Web server's SSL private server key.
- **server.csr:** the Web server's SSL certificate request.
- **server.crt:** the web server's SSL public certificate.
- **rhn-org-httpd-ssl-key-pair-*MACHINE_NAME-VER-REL*.noarch.rpm:** the RPM prepared for distribution to Satellite and Proxy Servers. Its associated **src.rpm** file is also generated.

  This RPM file contains the **server.key, server.csr,** and **server.crt** files. These files are installed in the following directories:

- `/etc/httpd/conf/ssl.key/server.key`
- `/etc/httpd/conf/ssl.csr/server.csr`
- `/etc/httpd/conf/ssl.crt/server.crt`

- **`rhn-server-openssl.cnf`:** the Web server's SSL configuration file.
- **`latest.txt`:** lists the latest versions of the relevant files.

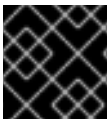
When this process is complete, distribute and install the RPM file on its respective Satellite or Proxy Server, and then restart the **`httpd`** service.

```
# service httpd restart
```

4.3. DEPLOYING THE CA SSL PUBLIC CERTIFICATE TO CLIENTS

The Red Hat Satellite Proxy Server and Red Hat Satellite Server installation processes generate a CA SSL public certificate and package it in an RPM file. These installation processes make the certificate and RPM file publicly available by placing a copy of one or both into the **`/var/www/html/pub/`** directory of the Satellite or Proxy Server.

You can use your web browser to inspect the contents of this directory: http://proxy-or-sat.example.com/pub/. You can use the **`wget`** or **`curl`** commands to download the CA SSL public certificate to a client system.



IMPORTANT

Confirm the name of the certificate or RPM file before running any of these commands.

```
# curl -O http://proxy-or-sat.example.com/pub/RHN-ORG-TRUSTED-SSL-CERT
# wget http://proxy-or-sat.example.com/pub/RHN-ORG-TRUSTED-SSL-CERT
```

Alternatively, if the CA SSL public certificate RPM file exists in the **`/pub`** directory, you can use the **`rpm`** command to install the package. For example:

```
# rpm -Uvh http://proxy-or-sat.example.com/pub/rhn-org-trusted-ssl-
cert-VER-REL.noarch.rpm
```

4.4. CONFIGURING CLIENT SYSTEMS TO USE CERTIFICATES

After you have deployed the RPM file or the certificate to a client system, you need to edit the configuration files of the **Red Hat Update Agent** and the **Red Hat Satellite Registration Client** (if necessary) to use the new CA SSL public certificate file. You also need to update the configuration so that it connects to the appropriate Red Hat Proxy Server or Red Hat Satellite Server. The generally accepted location for that CA SSL public certificate is in the **`/usr/share/rhn`** directory.

The Red Hat Proxy Server and Red Hat Satellite Server both have **Red Hat Satellite Bootstrap** installed by default, which can greatly reduce these repetitive steps and simplify the process of registering and configuring client systems. See the *Red Hat Satellite Getting Started Guide*for details.

CHAPTER 5. REPORTING SOFTWARE FAILURES

You can take advantage of Red Hat Satellite's software failure reporting capabilities and the Automatic Bug Reporting Tool (ABRT) to extend the overall reporting functionality of your systems. This extended functionality allows your clients to automatically report software failures captured by ABRT to the Satellite server, and also to process the captured failures in a centralized fashion. You can use either the webUI or the API to process these failure reports.

5.1. INSTALLING SOFTWARE FAILURE REPORTING TOOLS

The following procedure describes how to install Red Hat Satellite tools for ABRT on clients.

Procedure 5.1. To Use the Software Failure Reporting Functionality:

1. Log into your client system as the **root** user.
2. Install the spacewalk-abrt package on your client systems. This package installs the abrt package as a dependency.

```
# yum install spacewalk-abrt
```

NOTE

Neither the abrt nor spacewalk-abrt packages are available for Red Hat Enterprise Linux 5.

3. Run the **rhn-profile-sync** command to update the information stored on the Satellite server about this client system.

```
# rhn-profile-sync
```

5.2. USING SOFTWARE FAILURE REPORTING TOOLS

The spacewalk-abrt package has two important components:

- The configuration file for ABRT: **/etc/libreport/events.d/spacewalk.conf**
- The **spacewalk-abrt** utility: **/usr/bin/spacewalk-abrt**

The configuration file instructs the **abrt** daemon to use the **/usr/bin/spacewalk-abrt** utility to automatically report every software failure that occurs on the system to your Satellite server. This is a fully automated process and ordinarily does not require any human intervention.

Use the Red Hat Satellite Web UI to view software failure reports from clients. For more information, see the *Red Hat Satellite User Guide*.

5.3. MANUALLY REPORTING SOFTWARE FAILURES

Use the **spacewalk-abrt** utility to manually report software failures to your Satellite server. The following procedure shows how to perform a manually send a software failure report.

Procedure 5.2. To manually report software failures

1. Use the ***abrt-cli list*** parameter to display a list of existing failure reports.

```
# abrt-cli list

@0
Directory: /var/tmp/abrt/ccpp-2013-02-28-15:48:50-8820
count: 2
executable: /usr/bin/python2.7
package: python-2.7.3-13.fc16
time: Thu 28 Feb 2013 03:48:50 PM CET
uid: 0


@1
Directory: /var/tmp/abrt/oops-2013-02-27-14:16:03-8107-1
count: 3
package: kernel
time: Wed 27 Feb 2013 02:16:03 PM CET
```

2. After you have identified the failure that you want to report, use the **--report** option to send the report to the Satellite server.

```
# spacewalk-abrt --report /var/tmp/abrt/ccpp-2013-02-28-15:48:50-
8820
```

3. To manually report all of the software failures that have occurred on your system, use the **--sync** option:

```
# spacewalk-abrt --sync
```

5.4. CREATING SOFTWARE FAILURES FOR TESTING

You can force a software failure in order to verify that your reporting configuration is working properly. The following example demonstrates using the **kill** command to send a signal **11** argument (segmentation fault) to an example process:

```
# abrt-cli list
# sleep 600 &
[1] 17564
# kill -11 17564
#
[1]+  Segmentation fault      (core dumped) sleep 600
#
# abrt-cli list
@0
Directory:      /var/spool/abrt/ccpp-2013-05-14-04:56:17-17564
count:          1
executable:     /bin/sleep
package:        coreutils-8.4-19.el6
time:           Tue 14 May 2013 04:56:17 EDT
uid:            0
#
```

APPENDIX A. REVISION HISTORY

Revision 3-21.402 **Thu Jul 13 2017** **Russell Dickenson**
BZ#1359321 - Added rhn-profile-sync command to the Client Configuration Guide.

Revision 3-21.401 **Thu Aug 20 2015** **Dan Macpherson**
Mass publication of all Satellite 5.6 books

Revision 3-21.400 **2013-10-31** **Rüdiger Landmann**
Rebuild with publican 4.0.0

Revision 3-21 **Fri Sep 27 2013** **Dan Macpherson**
Final version of documentation suite

Revision 3-20 **Tue Sep 10 2013** **Dan Macpherson**
Revised Subtitle, Abstract and Preface for all Guides

Revision 3-19 **Mon Sep 2 2013** **Dan Macpherson**
Removing content relating to certain RPMs being located on /pub/ as per BZ#998336

Revision 3-18 **Thu Aug 29 2013** **Dan Macpherson**
First implementation of QE Review feedback

Revision 3-17 **Tue Aug 20 2013** **Athene Chan**
BZ#998333, 998336 Quality Assurance feedback incorporated into book.

Revision 3-16 **Mon Jul 29 2013** **Dan Macpherson**
Restructuring Software Failure chapter based upon tech review feedback

Revision 3-15 **Sun Jul 28 2013** **Dan Macpherson**
Second implementation of tech review feedback

Revision 3-14 **Wed Jul 24 2013** **Dan Macpherson**
Corrections for BZ#987245

Revision 3-13 **Tue Jul 23 2013** **Dan Macpherson**
First implementation of tech review feedback

Revision 3-12 **Fri Jul 19 2013** **Dan Macpherson**
Typo correction

Revision 3-11 **Fri Jul 12 2013** **Dan Macpherson**
Typo correction

Revision 3-10 **Fri Jul 12 2013** **Dan Macpherson**
Final beta updates

Revision 3-8 **Fri Jul 12 2013** **Dan Macpherson**
Update to Beta docs

Revision 3-6 **Fri Jul 12 2013** **David O'Brien**

Update section on CAs.
Add section on new reports.
Add section on using ABRT.
Update section about using rhn_register.
Remove chapter on Bootstraps and Scripting.
Remove redundant tables that reproduce man pages.

Revision 3-5 **Wed Sept 19 2012** **Dan Macpherson**
Final packaging for 5.5

Revision 3-4 **Fri Aug 10 2012** **Athene Chan**
Staging for review

Revision 3-0 **Tue Jun 28 2012** **Athene Chan**
Prepared for Red Hat Satellite Server 5.5 publication
Technical Review Edits
BZ#837703 Custom GPG Key note added

Revision 2-2 **Mon Aug 15 2011** **Lana Brindley**
Folded z-stream release into y-stream

Revision 2-1 **Wed Jun 15 2011** **Lana Brindley**
Prepared for publication

Revision 2-0 **Fri May 7 2011** **Lana Brindley**
Prepared for translation

Revision 1-8 **Mon Feb 7 2011** **Lana Brindley**
BZ#662876 - Certificates

Revision 1-7 **Tue Feb 1 2011** **Lana Brindley**
BZ#636703 - Latest Clients