

Red Hat Network 3.7

UNIX Support Guide



Red Hat Network 3.7: UNIX Support Guide

Copyright © 2005 by Red Hat, Inc.



Red Hat, Inc.

1801 Varsity Drive
Raleigh NC 27606-2072 USA
Phone: +1 919 754 3700
Phone: 888 733 4281
Fax: +1 919 754 3701
PO Box 13588
Research Triangle Park NC 27709 USA

UNIX-support(EN)-3.7-RHI (2005-03-02T18:17)

Copyright © 2005 by Red Hat, Inc. This material may be distributed only subject to the terms and conditions set forth in the Open Publication License, V1.0 or later (the latest version is presently available at <http://www.opencontent.org/openpub/>).

Distribution of substantively modified versions of this document is prohibited without the explicit permission of the copyright holder.

Distribution of the work or derivative of the work in any standard (paper) book form for commercial purposes is prohibited unless prior permission is obtained from the copyright holder.

Red Hat and the Red Hat "Shadow Man" logo are registered trademarks of Red Hat, Inc. in the United States and other countries.

All other trademarks referenced herein are the property of their respective owners.

The GPG fingerprint of the security@redhat.com key is:

CA 20 86 86 2B D6 9D FC 65 F6 EC C4 21 91 80 CD DB 42 A6 0E

Table of Contents

1. Introduction	1
1.1. Supported UNIX variants	1
1.2. Prerequisites	1
1.3. Included features	1
1.4. Differences in functionality	2
1.5. Excluded features	2
2. Client System Preparation	3
2.1. Installing Red Hat packages	3
2.2. Deploying client SSL certificates	4
2.3. Configuring the clients	4
3. Registration and Updates	7
3.1. Registering systems	7
3.2. Obtaining updates	7
3.2.1. Updating through the website	7
3.2.2. Updating at the command line	8
4. Remote Commands	11
4.1. Enabling commands	11
4.2. Issuing commands	11
Index	13

This document identifies differences in Red Hat Network functionality when used to manage UNIX-based systems. RHN offers UNIX support to help customers migrate from UNIX to Linux. Because of the limited scope of this task, the features offered for UNIX are limited. Review the subsequent sections to determine what UNIX variants are supported, which RHN features are included, excluded, and modified, and the prerequisites for managing a UNIX system with RHN.

1.1. Supported UNIX variants

The following UNIX variants and versions are supported by Red Hat Network:

- Solaris 2.5.1
- Solaris 2.6
- Solaris 2.7
- Solaris 2.8
- Solaris 2.9

1.2. Prerequisites

These items are needed to obtain UNIX Support:

- RHN Satellite Server 3.7 or later
- A Satellite certificate with non-Linux entitlements
- Client RHN entitlements for non-Linux service
- RHN-modified packages for UNIX, including **Python** and the **Red Hat Update Agent** - Refer to Section 2.1 *Installing Red Hat packages* for the complete list

1.3. Included features

The following features are included in the UNIX Support service level as they exist within RHN:

- A completely new Provisioning feature called *Remote Command* that enables users to schedule root-level commands on any managed client through the Satellite's website, if the client allows this action
- All Management-level functionality, such as system grouping, package profile comparison, and use of the System Set Manager to administer multiple systems at once
- The `rhnc_check` program, which checks in with the Satellite and performs any actions scheduled from the server
- The **Red Hat Network Service Daemon** (`rhnsd`), which triggers `rhnc_check` according to a configurable interval
- The **Red Hat Network Configuration Client** (`rhncfg-client`), which executes all configuration actions scheduled from the Satellite

- The **Red Hat Network Configuration Manager** (`rhncfg-client`), which allows command line administration of RHN configuration channels

1.4. Differences in functionality

The following RHN features work differently in a UNIX environment:

- The **Red Hat Update Agent for UNIX** offers a much smaller set of options than its Linux counterpart and relies upon the operating system's native toolset for package installation, rather than `RPM` - Refer to Section 3.2.2 *Updating at the command line* for the precise list of options
- Package installation now takes answer files into account for Solaris
- The **RHN Push** application has been similarly modified to upload native UNIX file types, including packages, patches, and Patch Clusters
- The **Channels** category of the RHN website has been augmented to accommodate the storage and installation of native UNIX file types, including the deployment of answer files with package actions

1.5. Excluded features

The following RHN offerings are deliberately excluded from the UNIX Support service level:

- All Provisioning-level functionality, such as kickstarting and package rollback, with the exception of configuration file management
- All Errata-related options, since the concept of Errata Updates is not understood in UNIX
- Source files for packages

Client System Preparation

Before your UNIX-based systems benefit from Red Hat Network, they must be prepared for connection. First, you need to install special packages that do not accompany the base operating system. Next, you need to deploy the SSL certificates required for a secure connection. Finally, you must reconfigure the client applications to connect to the RHN Satellite Server. Once finished, your systems will be ready to begin receiving RHN updates.

2.1. Installing Red Hat packages

This section walks you through the process of getting RHN-required packages installed on top of your base operating system. These packages, which do not come with standard installations of UNIX, are prerequisites to using Red Hat Network.

Of the utmost importance is the **Red Hat Update Agent for UNIX** (RHATu2d), which provides the link between your clients systems and the Red Hat Network service. The UNIX-specific version of the **Red Hat Update Agent** is limited in functionality compared to its Linux counterpart but still enables system registration and facilitates package installs and updates. Refer to Chapter 3 *Registration and Updates* for a full description of the tool's options.

In addition, you will need to install several other packages that have been modified by Red Hat to enable RHN management of your UNIX systems. Refer to Table 2-1 for this list, which also identifies the original packages from which they were derived.

UNIX Derivative	Original Package
RHATgnupg	gnupg
RHATlgcc	libgcc
RHATossl	openssl
RHATpssl	pyopenssl
RHATpyotk	python optik
RHATpythn	python
RHATrhnl	rhnlb
RHATsnprt	snprintf
RHATu2d	up2date
RHATzlib	zlib

Table 2-1. Red Hat Packages for UNIX

Follow these steps to install the required packages on your UNIX systems:

1. Log into the website of the RHN Satellite Server.
2. Click the **Channels** tab in the top menu, then **Software Channels** in the left menu, followed by the name of the channel associated with your UNIX variant.
3. Click the **Packages** tab, then the name of the each Red Hat package required for UNIX. Obtain the URL and use **wget** to download each file to a temporary directory on the client system, or

simply click the **Download** link near the bottom of the **Package Details** page.

4. Use the UNIX variant's native installation tool to then install each package. For instance, use `pkgadd` on Solaris machines at the command line, like so:


```
pkgadd -n -d /path/to/RHATu2d.package RHATu2d
```
5. If asked for what package you want, enter **A11**.
6. Answer any queries regarding permissions/ownership and repeat the installation command for each required package.

This will install the packages in the RHN-specific path for your UNIX variant. In the case of Solaris, this is `/opt/redhat/rhn/solaris`.

2.2. Deploying client SSL certificates

To ensure secure data transfer, Red Hat strongly recommends the use of SSL. The RHN Satellite Server eases implementation of SSL by generating the necessary certificates during its installation. The server-side certificate is automatically installed on the Satellite itself, while the client certificate is placed within a compressed archive, or tarball, in the `/pub` directory of the Satellite's Web server.

To install the certificate, follow these steps for each client:

1. Download the SSL tarball from the `/var/www/html/pub` directory of the RHN Satellite Server onto the client system. It will be named something like, "rhn-org-trusted-ssl-cert-*version*.tar.gz"
2. Decompress the tarball on the client using the `tar` command, like so:


```
tar -zxvf rhn-org-trusted-ssl-cert-version.tar.gz
```
3. Move the resulting client SSL certificate to the RHN-specific directory for your UNIX variant. For Solaris, this can be accomplished with a command similar to:


```
mv /path/to/RHN-ORG-TRUSTED-SSL-CERT /opt/redhat/rhn/solaris/usr/share/rhn/
```

When finished, the new client certificate will be installed in the appropriate directory for your UNIX system. If you have a large number of systems to prepare for RHN management, you may script this entire process.

Now you must reconfigure RHN client applications to refer to the newly installed SSL certificate. Refer to Section 2.3 *Configuring the clients* for instructions.

2.3. Configuring the clients

The final step in preparing your client systems for Red Hat Network is to reconfigure their RHN applications to use the new SSL certificate and obtain updates from the RHN Satellite Server. Both of these changes can be made by editing the configuration file of the **Red Hat Update Agent**, which provides registration and update functionality.

Follow these steps on each client system:

1. As root, go to the RHN-specific directory for the UNIX variant. For instance, on Solaris, this is `/opt/redhat/rhn/solaris`. Take note of its contents and that of its subdirectories.
2. Drill down to the `rhn` configuration directory for the system. For Solaris, the full path is `/opt/redhat/rhn/solaris/etc/sysconfig/rhn`.
3. Open the `up2date` configuration file in a text editor.
4. Find the `serverURL` entry and set its value to the fully qualified domain name (FQDN) of your RHN Satellite Server, like so:

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

5. Ensure the application refers to the RHN Satellite Server even when SSL is turned off by also setting the `noSSLServerURL` value to the Satellite, as in:

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

6. With the `up2date` configuration file still open, find the `sslCACert` entry and set its value to the name and location of the SSL certificate described in Section 2.2 *Deploying client SSL certificates*, such as:

```
sslCACert[comment]=The CA cert used to verify the ssl server
sslCACert=/opt/redhat/rhn/solaris/usr/share/rhn/RHN-ORG-TRUSTED-SSL-CERT
```

7. Finally, set the shell environment variables on the client to accommodate RHN-specific paths, commands, and libraries. For instance, for `BASH` on Solaris, add this to the `.bashrc` file:

```
export PATH=$PATH:/opt/redhat/rhn/solaris/bin:/opt/redhat/rhn/solaris/usr/sbin/
export LD_LIBRARY_PATH=/opt/redhat/rhn/solaris/lib
```

You should not need to include a `PYTHONPATH` since the bin knows where to find its site packages.

Your client systems are now ready for registration with Red Hat Network and management by your Satellite.

Registration and Updates

Now that you've installed RHN-specific packages, implemented SSL, and reconfigured your client systems to connect to the RHN Satellite Server, you're ready to begin registering systems and obtaining updates.

3.1. Registering systems

This section describes the RHN registration process for UNIX systems. You must use activation keys to accomplish this. These keys allow you to predetermine settings within RHN, such as base channels and system groups, and apply those automatically to systems during their registration.

Since activation key generation and use is covered extensively in other guides, this section focuses on differences when applying them to UNIX variants. Refer to the *Activation Keys* and *Registering with Activation Keys* sections of the *RHN Reference Guide* for full descriptions of this process.

Remember your system will connect to the Satellite and not the central RHN Servers, assuming you made the configuration changes required in Section 2.3 *Configuring the clients*. To register UNIX systems with your RHN Satellite Server, accomplish the following tasks in this order:

1. Log into the Satellite's version of the RHN website and click the **Systems** tab in the top navigation bar followed by **Activation Keys** in the left navigation bar. Then click the **create new key** link at the top-right corner of the page.
2. When creating the activation key on the following page, ensure you select a base channel corresponding to your particular UNIX variant (such as Solaris) and an entitlement level not associated with Linux systems; in this case, Non-Linux will do.
3. After creating the key, click its name in the **Activation Keys** list to enhance its RHN settings by associating software and configuration channels and system groups.
4. Open a terminal on the client system to be registered and switch user to root.
5. Enter the activation key command (`rhnreg_ks`) followed by the key character string, which may be copied directly from the **Activation Keys** list in the website. The command will look like:

```
rhncp_ks --activationkey=25ad3d099e69ff1a3c6dc60d7479c0f6
```
6. Go back to the website, click the name of the activation key, and ensure the new system appears within the **Activated Systems** tab.

3.2. Obtaining updates

Package updates in UNIX are handled much differently than in Linux. For instance, Solaris relies upon Patch Clusters to update multiple packages at once, while Red Hat operating systems use Errata Updates to associate upgrades with specific packages. In addition, Solaris uses answer files to predetermine package installations, something Linux doesn't understand, while Red Hat offers the conversely foreign concept of source packages. For this reason, this section seeks to highlight differences in using RHN tools on UNIX systems.

3.2.1. Updating through the website

Despite inherent differences, such as the lack of Errata, the channel and package management interfaces within the RHN website on the Satellite work largely the same for UNIX systems. All software channels designed to serve UNIX variants can be constructed almost exactly as the custom channels described in the *RHN Channel Management Guide*. The most significant difference is expectedly the architecture. When creating a UNIX software channel, ensure you select the base channel architecture appropriate for the systems to be served.

Further, Red Hat recommends you break down your packages into base and child channels depending on their nature. For instance, in Solaris, installation packages should go in the Solaris base channel, while patches and Patch Clusters should go in a child channel of the Solaris base channel. Extra installation packages can go in a separate Extras child channel.

RHN treats patches similarly to packages; they are listed and installed in the same way and with the same interface as normal packages. Patches are 'numbered' by Solaris, and will have names like "patch-solaris-108434". The version of a Solaris patch is extracted from the original Solaris metadata, and the release is always 1.

Patch Clusters are bundles of patches that are installed as a unit. RHN does not store the list of patches associated with a Patch Cluster. RHN has no way of telling that a Patch Cluster is installed on a system since Solaris doesn't track this information. So Patch Clusters always appear to be installable on a system and never appear in the system's installed packages list, even after they are installed. Patch Cluster names look like "patch-cluster-solaris-7_Recommended". The version is a datestring, such as "20040206", and the release is always 1.

To install packages or patches on an individual system, click the name of the system in the **Systems** category normally, select the packages from the Upgrade or Install lists of the **Packages** tab, and click **Install/Upgrade Selected Packages**. If a package (and not a patch), select the packages, and click **Confirm** to provide an answer file. Enter the full path and filename to the answer file and click **Upload File**.

To run a remote command while installing the package, click **Run Remote Command** rather than **Confirm**. Refer to Chapter 4 *Remote Commands* for instructions.

To install packages or patches on multiple systems at once, select the systems and click **System Set Manager** in the left navigation bar. Then, in the **Packages** tab, select the packages from the Upgrade or Install lists and click **Install/Upgrade Packages**. To complete the action, schedule the updates.

3.2.2. Updating at the command line

Like the website, command line use of the **Red Hat Update Agent** is affected by the limitations of UNIX package management. That said, most core functions can still be accomplished through the `up2date` command. The most significant difference is the absence of all options regarding source files. Refer to Table 3-1 for the precise list of options available for UNIX systems.

The command line version of the **Red Hat Update Agent** accepts the following arguments on UNIX systems:

Argument	Description
<code>--channel=channel</code>	Specify which channels to update from using channel labels.
<code>--get</code>	Fetch the package specified without resolving dependencies.
<code>--hardware</code>	Update this system's hardware profile on RHN.
<code>-h, --help</code>	Show this help message and exit.
<code>--installall</code>	Install all available packages. Use with <code>--channel</code> .

Argument	Description
<code>-l, --list</code>	List the latest versions of all packages installed.
<code>-p, --packages</code>	Update packages associated with this System Profile.
<code>--proxy=<i>proxy URL</i></code>	Specify an HTTP proxy to use.
<code>--proxyUser=<i>proxy user ID</i></code>	Specify a username to use with an authenticated HTTP proxy.
<code>--proxyPassword=<i>proxy password</i></code>	Specify a password to use with an authenticated HTTP proxy.
<code>--serverUrl=<i>server URL</i></code>	Specify what server to use.
<code>--showall</code>	List all packages available for download.
<code>--show-available</code>	List all the packages available that are not currently installed.
<code>--show-channels</code>	Show the channel names along with the package names where appropriate.
<code>--show-orphans</code>	List all the packages currently installed that are not in channels the system is subscribed to.
<code>-v, --verbose</code>	Show additional output.
<code>--version</code>	Show program version information.

Table 3-1. Update Agent Command Line Arguments

Remote Commands

With UNIX support, RHN offers the flexibility of issuing remote commands on client systems through the Satellite's RHN website. This feature allows you to run virtually any (compatible) application or script on any system in your domain without ever having to open a terminal.

4.1. Enabling commands

With the flexibility this tool offers comes great risk and the responsibility to mitigate that risk. For all practical purposes, this feature grants a root BASH prompt to anyone with administrative access to the system on the website.

This can be controlled, however, through the same config-enable mechanism used to determine which systems can have their configuration files managed by Red Hat Network. Refer to the **Configuration** tab description within the *System Details* section of the *RHN Provisioning Reference Guide* for details.

In short, you must create a directory and file on the UNIX system that tell RHN it is acceptable to run remote commands on the machine. The directory must be named `script`, the file must be named `run`, and both must be located in the `/etc/sysconfig/rhn/allowed-actions/` directory specific to your UNIX variant.

For instance, in Solaris, issue this command to create the directory:

```
mkdir -p /opt/redhat/rhn/solaris/etc/sysconfig/rhn/allowed-actions/script
```

To create the requisite file in Solaris, issue this command:

```
touch /opt/redhat/rhn/solaris/etc/sysconfig/rhn/allowed-actions/script/run
```

4.2. Issuing commands

You may schedule a remote command in a variety of ways: on an individual system, on multiple systems at once, and to accompany a package action.

To run a remote command on an individual system by itself, open the **System Details** page, click the **Remote Command** subtab, and establish the settings for the command. You may identify a specific user, group, and timeout period, as well as the script itself. Select and date and time to begin attempting the command, and click **Schedule Remote Command**.

Similarly, you may issue a remote command on multiple systems at once through the **System Set Manager**. Select the systems, go to the **System Set Manager**, click the **Misc** tab, and scroll down to the **Remote Command** section. From there you may run a remote command on the selected systems at once.

To run a remote command with a package action, schedule the action through the **Packages** tab of the **System Details** page and click **Run Remote Command** while confirming the action. Use the radio buttons at the top to determine whether the command should run before or after the package action, establish the settings for the command, and click **Schedule Package Install/Upgrade**.

Note that installing multiple packages that have different remote commands requires scheduling the installs separately or combining the commands into a single script.

Index

C

- client applications
 - redirecting, 4
- client systems
 - configuring, 4
 - registering, 7
 - updating, 7

R

- Red Hat packages
 - for UNIX, 3
 - installing, 3
- Red Hat Update Agent
 - UNIX Command Line Arguments, 8
- remote commands
 - enabling, 11
 - issuing, 11

S

- SSL
 - setting up, 4
- SSL certificates
 - deploying, 4

U

- UNIX variants
 - (See supported)
- updating
 - via command line, 8
 - via website, 8

