Spacewalk for Oracle® Linux

Installation Guide for Release 2.2

ORACLE®

Oracle Legal Notices

About this document

This document describes how to install Spacewalk 2.2 servers and proxies.

Document generated on: 2019-11-25 (revision: 229)

Table of Contents

Preface

Spacewalk for Oracle® Linux: Installation Guide for Release 2.2 describes how to install Spacewalk 2.2 servers and proxies.

Audience

This document is written for system administrators who want to use Spacewalk to manage Oracle Linux systems. It is assumed that readers have a general understanding of the Linux operating system.

Related Documents

The documentation for this product is available at:

Spacewalk for Oracle® Linux Documentation

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Chapter 1 Installing and Upgrading Spacewalk Servers

This chapter describes how to install or upgrade a Spacewalk server.

1.1 Oracle Linux Requirements

Oracle supports Spacewalk servers that are running on Oracle Linux 6 (x86_64). Oracle recommends that you update Oracle Linux with the latest packages that are available on the Oracle Linux yum server.

> **Note**
>
> Do not register a Spacewalk server or client with ULN. If you register a system with a Spacewalk server without first unregistering the system from ULN, it will break `yum` on the system. You can register a Spacewalk server as a client of itself to receive updates.

You should install Oracle Linux 6 using the **Minimal** or **Basic Server** software set. If you select additional package groups during installation, remove the `jta` package before installing Spacewalk, as this package causes Spacewalk services to fail to start.

Install Spacewalk using only the packages provided by Oracle from the Oracle Linux yum server repository at https://yum.oracle.com.

No third-party package repositories are required to install Spacewalk on Oracle Linux 6. All the required packages are available in the Spacewalk repository on the Oracle Linux yum server.

1.2 Storage Requirements

A Spacewalk server should have a minimum of 8 GB of memory. If the Spacewalk server also runs the database that stores the Spacewalk repository, this memory requirement is *in addition to* what is required to run the database.

To preserve errata mapping, Spacewalk maintains all available versions of all available packages in each software channel that you configure. As a result, the storage requirements for a Spacewalk server can be significant, depending on the number of major versions and architectures that you chose to support. Typically, the Oracle Linux binary repositories require approximately 50 GB for each combination of Oracle Linux release and architecture. An extra 40 GB is required for source packages and 80 GB for Ksplice updates for each combination of Oracle Linux release and architecture.

> **Caution**
>
> Packages are never removed from Oracle Linux repositories, so the space required for each repository only ever increases. You should actively monitor the available disk space on the Spacewalk server.

A Spacewalk server stores the packages that it hosts under the `/var/satellite/redhat` directory hierarchy. You should plan how best to configure the `/var` file system before you install Spacewalk. For example, if you set up `/var` as an `ext4` file system on an LVM logical volume, you can expand the storage when required.

1.3 Database Requirements

You can use the following database solutions to store Spacewalk data:

• Oracle Database

The software described in this documentation is either no longer supported or is in extended support.
Oracle recommends that you upgrade to a current supported release.

Oracle Database Requirements

- Oracle Database Express Edition (Oracle Database XE)

- PostgreSQL

Oracle supports only Oracle Database for use with Spacewalk. For more information, see *Oracle® Linux 6: Licensing Information User Manual*.

Oracle Database XE and PostgreSQL are not supported and their use is not described further in this document.

Oracle does not provide any tools for migrating from an unsupported database. For information on setting up a PostgreSQL database for use with Spacewalk, see https://fedorahosted.org/spacewalk/wiki/PostgreSQLServerSetup.

1.3.1 Oracle Database Requirements

You must install an Oracle Database server and make this server available before you install Spacewalk.

The following Oracle Database releases are supported:

- Oracle Database 12c.

  You can download the software from Oracle Technology Network (OTN) at https://www.oracle.com/technetwork/database/enterprise-edition/downloads/index.html.

- Oracle Database 11gR2, release 11.2.0.3 or later.

  To obtain the correct Oracle Database 11gR2 release, you must download the software from My Oracle Support (MOS) at https://support.oracle.com.

Documentation for Oracle database is available at https://docs.oracle.com/en/database/database.html.

1.3.2 Oracle Database Configuration

Configure the database as follows:

- The database must use the `AL32UTF8` character set that supports Unicode.

  Note

  You can select the `AL32UTF8` character set if you select **Advanced install** in the Installation GUI but not if you select **Typical install**.

- The database must have a Spacewalk user. For example, you could create a user named `spacewalk` for Oracle Database 11gR2 or `c##spacewalk` for Oracle Database 12c.

- If several Spacewalk servers will share the same database server, you must create a separate Spacewalk user for each server. For example, if you intend to set up two Spacewalk servers that share the same database, you could create users named `spacewalk` and `spacewalk2` for Oracle Database 11gR2 or `c##spacewalk` and `c##spacewalk2` for Oracle Database 12c.

- The Spacewalk user must have the `CONNECT` and `RESOURCE` roles.

- The Spacewalk user must have the `ALTER SESSION`, `CREATE SYNONYM`, `CREATE TABLE`, `CREATE TRIGGER`, `CREATE VIEW` and `UNLIMITED TABLESPACE` system privileges.

To create a Spacewalk user:

1. Log in as a database administrator (typically, SYSDBA) on the database server.

```
$ sqlplus / as SYSDBA

SQL*Plus: Release 12.1.0.2.0 Production on Tue Jun 2 11:25:42 2015

Copyright (c) 1982, 2014, Oracle.  All rights reserved.


Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options

SQL>
```

2. Enter the following SQL*Plus commands to set up the Spacewalk user:

```
SQL> create user sw_user identified by sw_passwd;
SQL> grant connect,resource to sw_user;
SQL> grant alter session, create synonym, create table, create trigger, create view to sw_user;
SQL> grant unlimited tablespace to sw_user;
```

Replace sw_user and sw_passwd with the Spacewalk user name and password.

Repeat these steps for each Spacewalk user that you need to set up.

1.4 Networking Requirements

You must configure a fully-qualified domain name (FQDN) for the Spacewalk server. Spacewalk does not consider .local and .localdomain to be valid domain names. Spacewalk clients must be able to resolve the Spacewalk server's FQDN for both forward and reverse lookups in DNS. If these conditions are not met, neither certificate validation nor PXE booting work and clients cannot register with the Spacewalk server.

Verify that the host name returned by the hostname command and the value of HOSTNAME defined in /etc/sysconfig/network are identical and that this host name is consistent with the FQDN defined for the system in DNS for both forward and reverse lookups, for example:

```
# hostname
swksvr.mydom.com
# grep HOSTNAME /etc/sysconfig/network
HOSTNAME=swksvr.mydom.com
# host swksvr.mydom.com
swksvr.mydom.com has address 192.168.1.3
# host 192.168.1.3
3.1.168.192.in-addr.arpa domain name pointer swksvr.mydom.com.
```

Edit /etc/hosts and configure the actual IP address for the FQDN and host name and not the loopback address (127.0.0.1), for example:

```
127.0.0.1       localhost localhost.localdomain localhost4 localhost4.localdomain4
::1             localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.1.3     swksvr.mydom.com swksvr
```

The following table shows the network ports that a Spacewalk server uses, depending on its configuration.

Port/ Protocol	Direction	Purpose
69/udp	Inbound	TFTP (if PXE provisioning support is required)

Port/ Protocol	Direction	Purpose
80/tcp	Inbound and outbound	HTTP access
443/tcp	Inbound and outbound	HTTPS access
5222/tcp	Inbound	Push support to Spacewalk clients (if required)
5269/tcp	Inbound	Push support to Spacewalk proxies (if required)

If the Spacewalk server needs to connect though a web proxy, you can configure the web proxy in either of the following ways:

- Edit `/etc/rhn/rhn.conf` and enter the web proxy configuration settings for the `server.satellite.http_proxy`, `server.satellite.http_proxy_username`, and `server.satellite.http_proxy_password` parameters, for example:

```
server.satellite.http_proxy = webproxy.mydom.com:80
server.satellite.http_proxy_username = another
server.satellite.http_proxy_password = clydenw
```

- In the Spacewalk web interface:

  1. Select the **Admin** tab, then **Spacewalk Configuration**, and then **General**.

  2. Enter the web proxy configuration settings in the fields **HTTP proxy**, **HTTP proxy username**, **HTTP proxy password**, and **Confirm HTTP proxy password**, and then click **Update**.

Configure the Spacewalk server, proxies, and clients to use network time synchronization mechanism such as the Network Time Protocol (NTP) or Precision Time Protocol (PTP). Spacewalk requires that the system time on these systems are as consistent to within 120 seconds. For example, if the system times (corrected for time zone difference) of the server and a client differ by more than 120 seconds, authentication of the `osad` service on a client by the `jabberd` service on the server fails.

For more information, see *Oracle® Linux 6: Administrator's Guide*.

1.5 Installing a Spacewalk Server

To install the Spacewalk server software:

1. Install Oracle Instant Client release 11.2.0.3 or later:

   a. Download the following Instant Client packages. You can find out more about the Instant Client from:

      https://www.oracle.com/database/technologies/instant-client.html

      - Instant Client Package - Basic

      - Instant Client Package - SQL*Plus

b. Add the library path to `ldconfig`:

```
# echo /usr/lib/oracle/11.2/client64/lib > /etc/ld.so.conf.d/oracle-instantclient11.2.conf
# ldconfig
```

> **Note**
>
> The Spacewalk server configuration fails if the Instant Client is missing. Oracle recommends that you install the latest 11gR2 release of the Instant Client.

2. Ensure that the `jta` package is not installed and prevent it from being installed when you install Spacewalk.

   To check if the `jta` package is installed:

```
# yum list installed | grep jta
```

   To remove the `jta` package:

```
# yum remove jta
```

   To prevent the `jta` package from being installed, either disable the Oracle Linux 6 Add-ons channel (`[ol6_addons]`), or add the `jta` package to the `exclude` directive in the yum configuration file `/etc/yum.conf`, for example:

```
exclude=jta*
```

3. Configure the system firewall, for example:

```
# iptables -I INPUT -p udp -m udp --dport 69 -j ACCEPT
# iptables -I INPUT -p tcp -m state --state NEW -m tcp --dport 80 -j ACCEPT
# iptables -I INPUT -p tcp -m state --state NEW -m tcp --dport 443 -j ACCEPT
# iptables -I INPUT -p tcp -m state --state NEW -m tcp --dport 5222 -j ACCEPT
# iptables -I INPUT -p tcp -m state --state NEW -m tcp --dport 5269 -j ACCEPT
# iptables -I OUTPUT -p tcp -m state --state NEW -m tcp --dport 80 -j ACCEPT
# iptables -I OUTPUT -p tcp -m state --state NEW -m tcp --dport 443 -j ACCEPT
# service iptables save
```

4. Enable access to the Spacewalk Server 2.2 repository on the Oracle Linux yum server server at https://yum.oracle.com.

   Download the latest yum repository configuration file (https://yum.oracle.com/public-yum-ol6.repo) to the yum repositories directory (by default `/etc/yum.repos.d`) and enable the `ol6_spacewalk22_server` repository in that file.

   Alternatively, you can create a yum repository configuration file, for example `/etc/yum.repos.d/spacewalk22.repo`, with the following content:

```
[ol6_spacewalk22_server]
name=Spacewalk Server 2.2 for Oracle Linux 6 ($basearch)
baseurl=https://yum.oracle.com/repo/OracleLinux/OL6/spacewalk22/server/$basearch/
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle
gpgcheck=1
enabled=1
```

5. Install the Spacewalk server packages that are configured to use Oracle Database:

    ```
    # yum install spacewalk-oracle
    ```

    You can safely ignore any SELinux `restorecon` messages that are displayed when the packages are installed.

    Optionally, you can also install the `spacewalk-utils` and `spacecmd` packages if you want to use commands such as `spacecmd`, `spacewalk-common-channels`, `spacewalk-hostname-rename`, or `spacewalk-sync-setup`.

    ```
    # yum install spacewalk-utils spacecmd
    ```

    `spacecmd` allows you to administer Spacewalk from the command line. You can manage activation keys, configuration channels, Kickstarts, software channels, systems, and users.

    `spacewalk-common-channels` allows you to configure the software channels, the Oracle Linux yum server repositories, GPG keys, and activation keys for Oracle Linux from the command line.

    `spacewalk-hostname-rename` allows you to regenerate the SSL certificate if you change the system's host name.

    `spacewalk-sync-setup` allows you configure a master-slave relationship between two Spacewalk servers that you want to use in an Inter-Server Synchronization (ISS) configuration. See Chapter 4, *Configuring Inter-Server Synchronization*.

6. Configure Spacewalk to use the Oracle database by running `spacewalk-setup --disconnected --external-oracle`.

    The following example shows an interactive configuration.

    > **Note**
    >
    > The value that you specify for Organization Unit must be the FQDN of the server in DNS, which must be the same as the system's host name.
    >
    > `sw_user` and `sw_passwd` are the Spacewalk user name and password that you configured for Oracle Database.
    >
    > The value expected for the database service name (SID) is the global database name and not the value of `ORACLE_SID`.

    ```
    # spacewalk-setup --disconnected --external-oracle
     * Setting up SELinux..
     * Setting up Oracle environment.
     * Setting up database.
     ** Database: Setting up database connection for Oracle backend.
     Database service name (SID)? orcl.mydom.com
     Database hostname [localhost]? spacewalk-db.mydom.com
     Username? sw_user
     Password? sw_passwd
     Database: Testing database connection.
     ** Database: Populating database.
     *** Progress: #######################################################
     * Setting up users and groups.
     ** GPG: Initializing GPG and importing key.
     ** GPG: Creating /root/.gnupg directory
     You must enter an email address.
    ```

```
 Admin Email Address? my.email@mydom.com
 * Performing initial configuration.
 * Activating Spacewalk.
 ** Loading Spacewalk Certificate.
 ** Verifying certificate locally.
 ** Activating Spacewalk.
 * Enabling Monitoring.
 * Configuring apache SSL virtual host.
 Should setup configure apache's default ssl server for you (saves original ssl.conf) [Y]? y
 ** /etc/httpd/conf.d/ssl.conf has been backed up to ssl.conf-swsave
 * Configuring tomcat.
 ** /etc/sysconfig//tomcat6 has been backed up to tomcat6-swsave
 ** /etc/tomcat6//server.xml has been backed up to server.xml-swsave
 ** /etc/tomcat6//web.xml has been backed up to web.xml-swsave
 * Configuring jabberd.
 * Creating SSL certificates.
 CA certificate password? cert_passwd
 Re-enter CA certificate password? cert_passwd
 Organization? Oracle Demo
 Organization Unit [spacewalk.domain.com]? spacewalk.mydom.com
 Email Address [your.email@domain.com]? my.email@mydom.com
 City? Redwood Shores
 State? CA
 Country code (Examples: "US", "JP", "IN", or type "?" to see a list)? US
 ** SSL: Generating CA certificate.
 ** SSL: Deploying CA certificate.
 ** SSL: Generating server certificate.
 ** SSL: Storing SSL certificates.
 * Deploying configuration files.
 * Update configuration in database.
 * Setting up Cobbler..
 Processing /etc/cobbler/modules.conf
 `/etc/cobbler/modules.conf' -> `/etc/cobbler/modules.conf-swsave'
 Processing /etc/cobbler/settings `/etc/cobbler/settings' -> `/etc/cobbler/settings-swsave'
 cobblerd does not appear to be running/accessible
 Cobbler requires tftp and xinetd services be turned on for PXE provisioning functionality.
 Enable these services [Y]? y
 cobblerd does not appear to be running/accessible
 * Restarting services.
 Installation complete.
 Visit https://spacewalk.mydom.com to create the Spacewalk administrator account.
```

7. Verify that the Spacewalk services are running correctly.

```
# spacewalk-service status
router (pid 1556) is running...
sm (pid 1566) is running...
c2s (pid 1576) is running...
s2s (pid 1585) is running...
tomcat6 (pid 1499) is running...                        [  OK  ]
httpd (pid  1543) is running...
osa-dispatcher (pid  1620) is running...
rhn-search is running (1649).
cobblerd (pid 1766) is running...
RHN Taskomatic is running (1797).
```

The process IDs on your system are likely to be different from those shown in this example.

8. Point a browser at the Spacewalk server URL mentioned in the output and create the Spacewalk administrator account.

9. If you want to use third-party CA-signed SSL certificate instead of the self-signed SSL certificate, follow the procedure given in Chapter 3, *Replacing SSL Certificates on Spacewalk Servers or Spacewalk Proxies*.

> **Note**
>
> Oracle recommends that you replace the self-signed SSL certificate before registering any clients. Otherwise, you must log on separately to each existing client and configure it to use the new SSL certificate. You cannot do this from the Spacewalk server.

10. Oracle recommends that you register the Spacewalk server as a client of itself after you have set up the software channels. Include the Spacewalk Server 2.2 channel in the list of software channels to which the server is subscribed.

1.6 Upgrading a Spacewalk Server

To upgrade a Spacewalk 2.0 server to Spacewalk 2.2:

1. Back up the Spacewalk configuration files in the following directories:

   - `/etc/jabberd`

   - `/etc/rhn`

   - `/etc/sysconfig/rhn`

   - server SSL build directory, typically `/root/ssl-build`

   For example, you could use the `tar` command to create a backup:

   ```
   # tar -cvf preSWupgrade.tar /etc/jabberd /etc/rhn /etc/sysconfig/rhn /root/ssl-build
   ```

2. Back up the Spacewalk database.

   For more information on how to use the Recovery Manager (RMAN) to create a backup, see:

   - Getting Started with RMAN (Oracle Database 11gR2)

   - Getting Started with RMAN (Oracle Linux 12c)

3. Disable access to the Spacewalk Server 2.0 packages and enable access to the Spacewalk Server 2.2 packages.

   - If the Spacewalk server is registered as a client of itself:

     a. Create a Spacewalk Server 2.2 channel as a child of the Oracle Linux 6 base channel.

     b. Create a Spacewalk Server 2.2 repository that accesses the Spacewalk Server 2.2 channel on the Oracle Linux yum server (`https://yum.oracle.com/repo/OracleLinux/OL6/spacewalk22/server/x86_64/`), using the same GPG settings as for Oracle Linux 6.

     c. Associate the Spacewalk Server 2.2 repository with the Spacewalk Server 2.2 channel and synchronize the repository's packages from the Oracle Linux yum server.

     d. Change the channel subscription of the Spacewalk server in Spacewalk from the Spacewalk Server 2.0 channel to the Spacewalk Server 2.2 channel.

- If the Spacewalk server is configured to obtain Spacewalk Server 2.0 packages from the Oracle Linux yum server:

    a. Disable the Spacewalk Server 2.0 repository in the yum repository configuration file, or delete your Spacewalk Server 2.0 `.repo` file.

    b. Download the latest yum repository configuration file (https://yum.oracle.com/public-yum-ol6.repo) to the yum repositories directory (by default `/etc/yum.repos.d`) and enable the `ol6_spacewalk22_server` repository in that file.

    Alternatively, create a yum repository configuration file, for example `/etc/yum.repos.d/spacewalk22.repo`, with the following content:

    ```
    [ol6_spacewalk22_server]
    name=Spacewalk Server 2.2 for Oracle Linux 6 ($basearch)
    baseurl=https://yum.oracle.com/repo/OracleLinux/OL6/spacewalk22/server/$basearch/
    gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle
    gpgcheck=1
    enabled=1
    ```

4. Upgrade the Spacewalk packages.

    ```
    # yum upgrade
    ```

    You can safely ignore any SELinux `restorecon` messages that are displayed when the packages are installed.

5. Install the `rpmconf` package and use the `rpmconf` command to copy any customizations you want to preserve to the upgraded configuration files.

    ```
    # yum install rpmconf
    # rpmconf -a
    ```

6. Stop the Spacewalk services.

    ```
    # spacewalk-service stop
    Shutting down spacewalk services...
    Stopping RHN Taskomatic...
    Stopped RHN Taskomatic.
    Stopping cobbler daemon:                                   [  OK  ]
    Stopping rhn-search...
    Stopped rhn-search.
    Shutting down osa-dispatcher:                              [  OK  ]
    Stopping httpd:                                            [  OK  ]
    Stopping tomcat6:                                          [  OK  ]
    Terminating jabberd processes ...
    Stopping s2s:                                              [  OK  ]
    Stopping c2s:                                              [  OK  ]
    Stopping sm:                                               [  OK  ]
    Stopping router:                                           [  OK  ]
    Done.
    ```

7. Upgrade the Spacewalk database schema.

    a. On the Oracle Database server, log in as a database administrator (typically, `SYSDBA`) and verify that the database is running.

    ```
    $ sqlplus / as SYSDBA

    SQL*Plus: Release 12.1.0.2.0 Production on Tue Jun 2 11:25:42 2015
    ```

```
Copyright (c) 1982, 2014, Oracle.  All rights reserved.


Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options

SQL>
```

b. Grant the additional system privileges CREATE TABLE and CREATE TRIGGER to the Spacewalk database user.

```
SQL> grant create table, create trigger to sw_user;
```

c. On the Spacewalk server, enter the following command to upgrade the database schema.

```
# /usr/bin/spacewalk-schema-upgrade
```

The output of the spacewalk-schema-upgrade command confirms whether or not the schema upgrade was successful. In the event of a failure:

- Check the log files in the /var/log/spacewalk/schema-upgrade directory to establish the cause.

- Restore the database from the backup.

- Fix the cause of the problem, for example by extending the tablespaces if there is insufficient space.

- Upgrade the database schema.

8. Upgrade the Spacewalk configuration for the Oracle Database.

```
# spacewalk-setup --disconnected --external-oracle --upgrade
```

If you previously customized the Spacewalk installation, for example in the file /etc/rhn/rhn.conf, restore the customizations from your backup.

9. (Optional) Enable monitoring and the monitoring scout.

> **Note**
>
> Monitoring is a deprecated feature that will be removed in a future release.

- To enable monitoring without enabling the monitoring scout, enter:

```
# /usr/share/spacewalk/setup/upgrade/rhn-enable-monitoring.pl
```

- To enable monitoring and the monitoring scout, enter:

```
# /usr/share/spacewalk/setup/upgrade/rhn-enable-monitoring.pl --enable-scout
```

10. Restart the Spacewalk services.

```
# spacewalk-service start
Starting spacewalk services...
Initializing jabberd processes ...
Starting router:                                           [  OK  ]
```

```
Starting sm:                                                [  OK  ]
Starting c2s:                                               [  OK  ]
Starting s2s:                                               [  OK  ]
Starting tomcat6:                                           [  OK  ]
Waiting for tomcat to be ready ...
Starting httpd:                                             [  OK  ]
Starting osa-dispatcher:                                    [  OK  ]
Starting rhn-search...
Starting cobbler daemon:                                    [  OK  ]
Starting RHN Taskomatic...
Done.
```

1.7 Troubleshooting Spacewalk Server Problems

This section describes how to diagnose and fix problems that you might encounter on a Spacewalk server.

For information about other issues and workarounds, see *Spacewalk for Oracle® Linux: Release Notes for Release 2.2*.

Use the `spacewalk-service status` command to verify which Spacewalk services are not running correctly. If a service does not start correctly, this is usually due to incorrect configuration information being entered during installation.

Check the following log files for errors:

`/var/log/httpd/*`	`httpd` service log files.
`/var/log/nocpulse/*`	Monitoring log files (if enabled).
`/var/log/notification/*`	Monitoring notification log files (if enabled).
`/var/log/rhn_satellite_install.log`	Installation log file.
`/var/log/rhn/reposync/*`	Repository synchronization log files.
`/var/log/rhn/rhn_server_satellite.log`	Synchronization log file.
`/var/log/rhn/rhn_server_xmlrpc.log`	XML-RPC transaction log file.
`/var/log/rhn/rhn_taskomatic_daemon.log`	RHN Task Engine (Taskomatic) log messages.
`/var/log/yum.log`	Yum log file.

1.7.1 Managing Spacewalk Logging

Spacewalk generates large numbers of log messages, particularly under `/var/log/httpd`. To avoid running out of disk space, you might need to adjust the `logrotate` settings to implement more active rotation, compression, and archival of log files.

For more information, see *Oracle® Linux 6: Administrator's Guide*.

1.7.2 osa-dispatcher Reports a Certificate Verification Failure

If the `osa-dispatcher` service does not start, you might see an error such as the following if you attempt to start the service manually by using the `service osa-dispatcher start` command:

```
Starting osa-dispatcher: Spacewalk 10611 2015/05/26 17:11:22 +01:00: ('Traceback caught:',)
Spacewalk 10611 2015/05/26 17:11:22 +01:00: ('Traceback (most recent call last):\n
    File "/usr/share/rhn/osad/jabber_lib.py", line 631, in connect\n
    ssl.do_handshake()\nError: [(\'SSL routines\', \'SSL3_GET_SERVER_CERTIFICATE\',
    \'certificate verify failed\')]\n',)
                                                        [FAILED]
```

This error usually indicates that the system's host name does not match its FQDN in DNS or that you specified an incorrect FQDN as the name of the Organization Unit when you installed Spacewalk.

To regenerate the SSL certificate, you can use the `spacewalk-hostname-rename` command, which is available in the `spacewalk-utils` package.

> **Note**
>
> If the host name has changed, `spacewalk-hostname-rename` prompts you to enter the same certificate password as you used when you created the existing certificate. To verify that you know the correct password before running `spacewalk-hostname-rename`, use the following command, which returns the base64-encoded private key if the password is correct and an error otherwise:
>
> ```
> # openssl rsa -in /root/ssl-build/RHN-ORG-PRIVATE-SSL-KEY
> Enter pass phrase for /root/ssl-build/RHN-ORG-PRIVATE-SSL-KEY: cert_passwd
> writing RSA key
> -----BEGIN RSA PRIVATE KEY-----
> MIIEowIBAAKCAQEAvYQ/Enqo+T3WByuXiOQCpIh7eBFdqWt/e+cm3pfvGYBqi/1g
> ...
> +rEs1ZK2mCoofnPzg2o097oPj5v4IHYh8Bmlssbk9BHvQ2ZQckSA
> -----END RSA PRIVATE KEY-----
> ```

Run `spacewalk-hostname-rename`, specifying the IP address and FQDN of the server as arguments to the command, for example:

```
# spacewalk-hostname-rename IP_address --ssl-orgunit=FQDN
```

After regenerating the SSL certificate, stop the `jabberd` and `osa-dispatcher` services, clear the `jabberd` database, and restart the `jabberd` and `osa-dispatcher` services.

```
# service jabberd stop
# service osa-dispatcher stop
# rm -Rf /var/lib/jabberd/db/*
# service jabberd start
# service osa-dispatcher start
```

On every client registered to the server, verify that the value of `serverURL` in `/etc/sysconfig/rhn/up2date` is configured with the correct server host name or IP value, for example:

```
serverURL=http://swksvr.mydom.com/XMLRPC
```

1.7.3 osa-dispatcher Reports an Invalid Password

If the `osa-dispatcher` service starts correctly but later stops unexpectedly, you might see a log error such as the following:

```
Spacewalk 2316 2015/06/05 20:38:47 +01:00: ('Invalid password',)
```

This error can happen if the `jabberd` database contains invalid entries.

To clear the error, stop the `jabberd` and `osa-dispatcher` services, clear the `jabberd` database, and restart the `jabberd` and `osa-dispatcher` services.

```
# service jabberd stop
# service osa-dispatcher stop
# rm -Rf /var/lib/jabberd/db/*
# service jabberd start
# service osa-dispatcher start
```

Note

Oracle recommends that you clear the `jabberd` database at regular intervals to avoid problems with OSA.

1.7.4 tomcat6 Does Not Start

If the `tomcat6` service does not start, it is likely that the `jta` package has been installed instead of the `geronimo-jta-1.1-api` package. Remove the `jta` package, install the `geronimo-jta-1.1-api` package, and then restart the Spacewalk services:

```
# spacewalk-service restart
```

1.7.5 tomcat6 Runs Out of Memory

If you see messages such as `OutOfMemoryError: Java heap space` in the logs, increase the maximum amount of memory that is available to `tomcat6` for its heap:

1. Edit `/etc/sysconfig/tomcat6` and increase the argument to `-Xmx` in the `JAVA_OPTS` setting. For example, `-Xmx1024m` increases the maximum size of the heap to 1 GB.

2. Restart the Spacewalk services:

```
# spacewalk-service restart
```

Chapter 2 Installing and Upgrading Spacewalk Proxies

A Spacewalk proxy acts as an intermediary between Spacewalk clients and a Spacewalk server. The main purposes of a Spacewalk proxy are to mitigate the loading on the Spacewalk server and to reduce the download times for Spacewalk clients.

This chapter describes how to install or upgrade a Spacewalk proxy.

2.1 Spacewalk Proxy Requirements

Oracle supports Spacewalk proxies that are running on Oracle Linux 6 (x86_64). Oracle recommends that you update Oracle Linux with the latest packages that are available on the Oracle Linux yum server.

> **Note**
>
> Do not register a Spacewalk proxy with ULN. If you register a system with a Spacewalk server without first unregistering the system from ULN, it will break `yum` on the system. You can register a Spacewalk proxy as a client of itself or of a Spacewalk server to receive updates.

You can install Oracle Linux 6 using the **Minimal** or **Basic Server** software set. If you select additional package groups during installation, remove the `jta` package before installing Spacewalk, as this package causes Spacewalk services to fail to start.

Install Spacewalk using only the packages provided by Oracle from the Oracle yum server repository at https://yum.oracle.com.

No third-party package repositories are required to install Spacewalk on Oracle Linux 6. All the required packages are available in the Spacewalk repository on the Oracle Linux yum server.

2.2 Storage Requirements

A Spacewalk proxy should have a minimum of 4 GB of memory.

Typically, the proxy cache requires 10 GB of storage for each combination of Oracle Linux release and architecture.

By default, a Spacewalk proxy caches packages under the `/var/spool/squid` directory hierarchy and is limited to using up to 60% of the free space in the file system that contains this directory hierarchy.

2.3 Networking Requirements

You must configure a fully-qualified domain name (FQDN) for the Spacewalk proxy. Spacewalk does not consider `.local` and `.localdomain` to be valid domain names. Spacewalk clients must be able to resolve the Spacewalk proxy's FQDN for both forward and reverse lookups in DNS. If these conditions are not met, certificate validation and PXE booting do not work and clients cannot register with the Spacewalk server.

Verify that the host name returned by the `hostname` command and the value of `HOSTNAME` defined in `/etc/sysconfig/network` are identical and that this host name is consistent with the FQDN defined for the system in DNS for both forward and reverse lookups, for example:

```
# hostname
swkproxy.us.mydom.com
# grep HOSTNAME /etc/sysconfig/network
HOSTNAME=swkproxy.us.mydom.com
```

```
# host swkproxy.us.mydom.com
swkproxy.us.mydom.com has address 10.0.0.24
# host 10.0.0.24
24.0.0.10.in-addr.arpa domain name pointer swkproxy.us.mydom.com.
```

Edit `/etc/hosts` and configure the actual IP address for the FQDN and host name and not the loopback address (127.0.0.1), for example:

```
127.0.0.1        localhost localhost.localdomain localhost4 localhost4.localdomain4
::1              localhost localhost.localdomain localhost6 localhost6.localdomain6
10.0.0.24        swkproxy.us.mydom.com swkproxy
```

The following table shows the network ports that a Spacewalk proxy uses, depending on its configuration.

Port/ Protocol	Direction	Purpose
80/tcp	Inbound and outbound	HTTP access
443/tcp	Inbound and outbound	HTTPS access
4545/tcp	Outbound	Monitoring (if enabled)
5222/tcp	Inbound	Push support to Spacewalk clients (if required)
5269/tcp	Inbound	Push support to Spacewalk proxies (if required)

If the Spacewalk proxy needs to connect though a web proxy, you can configure the web proxy during installation.

Configure the Spacewalk server, proxies, and clients to use network time synchronization mechanism such as the Network Time Protocol (NTP) or Precision Time Protocol (PTP). Spacewalk requires that the system time on these systems are consistent to within 120 seconds.

2.4 Installing a Spacewalk Proxy

To install the Spacewalk proxy software:

1. Ensure that the `jta` package is not installed and prevent it from being installed when you install Spacewalk.

   To check if the `jta` package is installed:

   ```
   # yum list installed | grep jta
   ```

   To remove the `jta` package:

   ```
   # yum remove jta
   ```

   To prevent the `jta` package from being installed, either disable the Oracle Linux 6 Add-ons channel (`[ol6_addons]`), or add the `jta` package to the `exclude` directive in the yum configuration file `/etc/yum.conf`, for example:

   ```
   exclude=jta*
   ```

2. Configure the system firewall, for example:

```
# iptables -I INPUT -p tcp -m state --state NEW -m tcp --dport 80 -j ACCEPT
# iptables -I INPUT -p tcp -m state --state NEW -m tcp --dport 443 -j ACCEPT
# iptables -I INPUT -p tcp -m state --state NEW -m tcp --dport 5222 -j ACCEPT
# iptables -I INPUT -p tcp -m state --state NEW -m tcp --dport 5269 -j ACCEPT
# iptables -I OUTPUT -p tcp -m state --state NEW -m tcp --dport 80 -j ACCEPT
# iptables -I OUTPUT -p tcp -m state --state NEW -m tcp --dport 443 -j ACCEPT
# iptables -I OUTPUT -p tcp -m state --state NEW -m tcp --dport 4545 -j ACCEPT
# service iptables save
```

3. Register the system as a client of the Spacewalk server for which it will act as a Spacewalk proxy:

   a. Enable access to the Spacewalk Client 2.2 repository on the Oracle Linux yum server.

      Download the latest the yum repository configuration file from https://yum.oracle.com/ and save it
      to the yum repositories directory (by default `/etc/yum.repos.d`). Edit the configuration file and
      enable the `ol6_spacewalk22_client` repository.

      Alternatively, you can create a `/etc/yum.repos.d/spacewalk22-client.repo` file with the
      following content:

      ```
      [ol6_spacewalk22_client]
      name=Spacewalk Client 2.2 for Oracle Linux 6 ($basearch)
      baseurl=https://yum.oracle.com/repo/OracleLinux/OL6/spacewalk22/client/$basearch/
      gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle
      gpgcheck=1
      enabled=1
      ```

   b. Install the Spacewalk Client 2.2 software.

      ```
      # yum install rhn-client-tools rhn-check rhn-setup rhnsd m2crypto yum-rhn-plugin
      ```

      Note

      If you previously registered the system with ULN, this command unregisters
      the system from ULN.

   c. Register the system with the Spacewalk server by using the `rhnreg_ks` command.

      ```
      # rhnreg_ks --serverUrl=http://spacewalk_server/XMLRPC --activationkey=activation_key
      ```

      The registration process downloads the Spacewalk server's SSL certificate (`RHN-ORG-TRUSTED-
      SSL-CERT`) to the `/usr/share/rhn` directory and configures settings in `/etc/sysconfig/rhn/
      up2date`.

   d. Disable access to the Spacewalk Client repository in the yum repository configuration file or delete
      the Spacewalk Client `.repo` file.

4. If not already present on the Spacewalk server, create software channels for Spacewalk 2.2 Client and Spacewalk Server 2.2 and subscribe the Spacewalk proxy system to these channels:

   a. Create a Spacewalk Client 2.2 channel as a child of the Oracle Linux 6 base channel.

   b. Create a Spacewalk Client 2.2 repository that access the Spacewalk Client 2.2 channel on the Oracle Linux yum server (`https://yum.oracle.com/repo/OracleLinux/OL6/spacewalk22/client/x86_64/`), using the same GPG settings as for Oracle Linux 6.

   c. Associate the Spacewalk Client 2.2 repository with the Spacewalk Client 2.2 channel and synchronize the repository's packages from the Oracle Linux yum server.

   d. Create a Spacewalk Server 2.2 channel as a child of the Oracle Linux 6 base channel.

   e. Create a Spacewalk Server 2.2 repository that access the Spacewalk Server 2.2 channel on the Oracle Linux yum server (`https://yum.oracle.com/repo/OracleLinux/OL6/spacewalk22/server/x86_64/`), using the same GPG settings as for Oracle Linux 6.

   f. Associate the Spacewalk Server 2.2 repository with the Spacewalk Server 2.2 channel and synchronize the repository's packages from the Oracle yum server.

   g. Change the channel subscription of the Spacewalk server in Spacewalk from the Spacewalk Server 2.0 channel to the Spacewalk Server 2.2 channel.

   h. Subscribe the Spacewalk proxy to the Spacewalk Client 2.2 and Spacewalk Server 2.2 channels.

5. Install the `openssh-clients` and `rhn-client-tools` packages.

   ```
   # yum install openssh-clients rhn-client-tools
   ```

6. Create the directory `/root/ssl-build`.

   ```
   # mkdir /root/ssl-build
   ```

7. Install the Spacewalk proxy installer package.

   ```
   # yum install spacewalk-proxy-installer
   ```

8. Configure the Spacewalk proxy by running the `configure-proxy.sh` script.

   The following example shows an interactive configuration.

   > **Note**
   >
   > This example does not enable monitoring. Monitoring is a deprecated feature that will be removed in a future release.

   ```
   # configure_proxy.sh
   Proxy version to activate [2.2]: [Enter]
   Traceback email []: my.email@mydom.com
   Use SSL [Y/n]: Y
   HTTP Proxy []: [Enter]
   Regardless of whether you enabled SSL for the connection to the Spacewalk Parent
   Server, you will be prompted to generate an SSL certificate.
   This SSL certificate will allow client systems to connect to this Spacewalk Proxy
   securely. Refer to the Spacewalk Proxy Installation Guide for more information.
   Organization []: Oracle Demo
   Organization Unit [swkproxy.us.mydom.com]: [Enter]
   Common Name [swkproxy.us.mydom.com]: [Enter]
   ```

The software described in this documentation is either no longer supported or is in extended support.
Oracle recommends that you upgrade to a current supported release.

Installing a Spacewalk Proxy

```
City []: Redwood Shores
State []: CA
Country code []: US
Email [my.email@mydom.com]: [Enter]
Cname aliases (separated by space) []: [Enter]
Spacewalk Proxy successfully activated.
Loaded plugins: rhnplugin
This system is receiving updates from RHN Classic or Red Hat Satellite.
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package spacewalk-proxy-management.noarch 0:2.2.10-1.el6 will be installed
...
Transaction Summary
================================================================================
Install     42 Package(s)

Total download size: 13 M
Installed size: 32 M
Is this ok [y/N]: y
Downloading Packages:
(1/42): apr-1.3.9-5.el6_2.x86_64.rpm                        | 122 kB     00:00
...
You do not have monitoring installed.
Do you want to install monitoring scout?
Will run 'yum install spacewalk-proxy-monitoring'. [Y/n]: n
...
Using CA key at /root/ssl-build/RHN-ORG-PRIVATE-SSL-KEY.
Generating distributable RPM for CA public certificate:
Copying CA public certificate to /var/www/html/pub for distribution to clients:
Generating SSL key and public certificate:
CA password: cert_passwd
Installing SSL certificate for Apache and Jabberd:
Preparing packages for installation...
rhn-org-httpd-ssl-key-pair-swkproxy-1.0-1
Create and populate configuration channel rhn_proxy_config_1000010040? [Y/n]: Y
RHN username: []: swadmin
Password: swadmin_passwd
Using server name swksvr.mydom.com
Creating config channel rhn_proxy_config_1000010040
Config channel rhn_proxy_config_1000010040 created
Using server name swksvr.mydom.com
Pushing to channel rhn_proxy_config_1000010040:
Local file /etc/httpd/conf.d/ssl.conf -> remote file /etc/httpd/conf.d/ssl.conf
Local file /etc/rhn/rhn.conf -> remote file /etc/rhn/rhn.conf
Local file /etc/rhn/cluster.ini -> remote file /etc/rhn/cluster.ini
Local file /etc/squid/squid.conf -> remote file /etc/squid/squid.conf
Local file /etc/httpd/conf.d/cobbler-proxy.conf -> remote file
          /etc/httpd/conf.d/cobbler-proxy.conf
Local file /etc/httpd/conf/httpd.conf -> remote file /etc/httpd/conf/httpd.conf
Local file /etc/jabberd/c2s.xml -> remote file /etc/jabberd/c2s.xml
Local file /etc/jabberd/sm.xml -> remote file /etc/jabberd/sm.xml
Enabling Spacewalk Proxy.
Shutting down rhn-proxy...
Terminating jabberd processes ...
...
Done.
Starting rhn-proxy...
init_cache_dir /var/spool/squid... Starting squid: .[ OK ]
Starting httpd: [ OK ]
Initializing jabberd processes ...
Starting router: [ OK ]
Starting sm: [ OK ]
Starting c2s: [ OK ]
Starting s2s: [ OK ]
```

```
[ OK ]
Done.
There were some answers you had to enter manually.
Would you like to have written those into file
formatted as answers file? [Y/n]: Y
Writing proxy-answers.txt.NtM1Y
```

The RHN user name and password (`swadmin` and `swadmin_passwd`) are the administrator's user name and password for the Spacewalk server.

The information that you enter is recorded in a file named `proxy-answers.txt.UID`, where `UID` is a unique identifier. You can use this file to automate the configuration of a Spacewalk proxy, for example:

```
# configure-proxy.sh --non-interactive --answer-file=proxy-answers.txt.NtM1Y
```

9. If you want to use third-party CA-signed SSL certificate instead of the self-signed SSL certificate, follow the procedure given in Chapter 3, *Replacing SSL Certificates on Spacewalk Servers or Spacewalk Proxies*.

> **Note**
>
> Oracle recommends that you replace the self-signed SSL certificate before registering any clients. Otherwise, you must log on separately to each existing client and configure it to use the new SSL certificate. You cannot do this from the Spacewalk server.

10. To check that the Spacewalk proxy is running correctly, specify the URL of the proxy when registering a Spacewalk client, for example:

```
# rhnreg_ks --serverUrl=http://spacewalk_proxy/XMLRPC --activationkey=activation_key
```

After registering the client, subscribe it to software channels on the server and verify that you can update packages from the client.

2.5 Upgrading a Spacewalk Proxy

To upgrade a Spacewalk proxy:

1. Back up the CA key, SSL certificate, and openSSL configuration file in `/root/ssl-build`.

   For example, you could use the `tar` command to create a backup:

```
# cd /root/ssl-build
# tar -cvf /tmp/sslcerts.tar RHN-ORG-PRIVATE-SSL-KEY RHN-ORG-TRUSTED-SSL-CERT rhn-ca-openssl.cnf
```

2. Update the `openssh-clients` and `rhn-client-tools` packages.

```
# yum update openssh-clients rhn-client-tools
```

3. Extract the CA key, SSL certificate, and openSSL configuration file from your backup file to `/root/ssl-build`, for example:

```
# cd /root/ssl-build
# tar -xvf /tmp/sslcerts.tar
```

4. Update the Spacewalk proxy installer package.

```
# yum update spacewalk-proxy-installer
```

5. Configure the Spacewalk proxy by running the `configure-proxy.sh` script.

   The following example uses the `proxy-answers.txt.`*`UID`* file saved from a previous installation to perform the configuration:

   ```
   # configure-proxy.sh --non-interactive --answer-file=proxy-answers.txt.NtM1Y
   ```

2.6 Troubleshooting Spacewalk Proxy Installation Problems

This section describes how to diagnose and fix problems that you might encounter on a Spacewalk proxy.

For information about other issues and workarounds, see *Spacewalk for Oracle® Linux: Release Notes for Release 2.2*

Use the `rhn-proxy status` command to find out which Spacewalk proxy services are not running correctly. If a service does not start correctly, this is usually due to incorrect configuration information being entered during installation.

Verify that the file `/etc/sysconfig/rhn/systemid` is owned by `root:apache` and that its mode is set to 640.

Check the following log files for errors:

`/var/log/httpd/*`	`httpd` service log files.
`/var/log/rhn/` `rhn_proxy_broker.log`	Proxy brokering service log file.
`/var/log/rhn/` `rhn_proxy_redirect.log`	Proxy SSL redirection service log file.
`/var/log/squid/*`	Squid proxy log files.
`/var/log/yum.log`	Yum log file.

2.6.1 Clearing the Proxy Cache

If required to clear up package provisioning problems, you can clear the proxy cache:

1. Stop the `httpd` and `squid` services:

   ```
   # service httpd stop
   # service squid stop
   ```

2. Delete the contents of the cache:

   ```
   # rm -Rf /var/cache/rhn/*
   ```

3. Restart the `squid` and `httpd` services:

   ```
   # service squid start
   # service httpd start
   ```

Chapter 3 Replacing SSL Certificates on Spacewalk Servers or Spacewalk Proxies

When you install a Spacewalk server or Spacewalk proxy, you create a self-signed SSL certificate that you can use with Spacewalk clients. This section describes how to replace self-signed certificates or expired CA-signed certificates with certificates that have been signed by a Certificate Authority (CA).

You can use certificates for individual Spacewalk servers or Spacewalk proxies, or wildcard certificates for all Spacewalk servers or Spacewalk proxies in the domains that the wildcard certificates cover.

To replace the existing certificate on a Spacewalk server or Spacewalk proxy:

1. Create a backup of the system's existing SSL configuration, for example:

    ```
    # tar -cvf SSLconfig.tar \
      /etc/httpd/conf/ssl.* \
      /etc/pki/spacewalk/jabberd/server.pem \
      /root/ssl-build \
      /var/www/html/pub
    ```

2. Obtain a server certificate from a CA and install this certificate in the SSL build hierarchy on the system:

    a. Send the Certificate Signing Request (CSR) file `/root/ssl-build/swksvr/server.csr` to the CA.

    > **Note**
    >
    > *swksvr* is the name of the Spacewalk server or Spacewalk proxy that you used to set up the existing SSL configuration with the domain name removed.

    After validating your request, the CA returns a signed server certificate file.

    b. Create a backup of the signed server certificate file.

    c. If necessary, convert the certificate to PEM format.

    A PEM-format certificate file is a text file that contains a base64-encoded certificate section between begin and end markers, for example:

    ```
    -----BEGIN CERTIFICATE-----
    MIIF7DCCBNSgAwIBAgIQbsx6pacDIAm4zrz06VLUkTANBgkqhkiG9w0BAQUFADCB
    ...
    Rs/iGAZeqa6ogZpHFt4MKGwlJ7net4RYxh84HqTEy2Y=
    -----END CERTIFICATE-----
    ```

    A PEM-format certificate file usually has a file extension of `.crt` or `.pem`. However, binary DER-format certificate files are also sometimes given a `.crt` extension.

    A DER-format certificate file is a binary file that usually has a file extension of `.cer` or `.der`, but can also have the extension `.cert` or `.crt`.

    You can use the following command to test if a certificate file is in DER format:

    ```
    # openssl x509 -inform der -text -in certificate_file
    ```

    d. If a certificate file is in DER format, convert the file to a PEM-format certificate file, for example:

```
# openssl x509 -inform der -in server.cer -out server.pem
```

e. If a PEM-format certificate file was not generated on a UNIX or Linux system, it might contain
   `^M` carriage return characters. You can use either of the following commands to remove these
   characters:

```
# sed -i -e 's/\r//' server.pem
-or-
# dos2unix server.pem
```

The `dos2unix` command is available in the `dos2unix` package.

f. Copy the PEM-format server certificate file to `/root/ssl-build/swksvr/server.crt`,
   overwriting the original file.

```
# cp server.pem /root/ssl-build/swksvr/server.crt
```

3. Add the CA public certificate to the `/root/ssl-build` directory as the file `RHN-ORG-TRUSTED-SSL-CERT`, overwriting the original file:

   • If available, obtain the CA chain certificate from the CA that issued the server certificate. Copy this
     certificate file to `/root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT`, for example:

```
# cp ca_chain.pem /root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT
```

   • If the CA chain certificate is not available from the issuing CA, create the CA chain certificate
     yourself:

     a. Obtain the root CA public certificate and the intermediate CA public certificates from the issuing
        CA.

     b. Concatenate the chain of CA public certificate files starting with the public certificate file of the CA
        that issued your server certificate down to the public certificate file of the root CA to `/root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT`, for example:

```
# cat intermediate_ca.pem root_ca.pem > /root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT
```

        In this example, `intermediate_ca.pem` is the public certificate file of the intermediate CA that
        signed your server certificate, and `root_ca.pem` is the public certificate file of the root CA that
        signed the intermediate certificate.

        Note

        The order of the public certificates in a CA chain certificate file is critical.
        The CA chain certificate does not work if its component certificates are not
        in the correct order.

   • If a root CA signed your server certificate directly (which is unlikely nowadays), only the public
     certificate of the root CA is required. Copy the root CA public certificate file to `/root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT`, for example:

```
# cp root_ca.pem /root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT
```

4. Use the following command to validate the server certificate against the CA public certificate:

```
# openssl verify -CAfile /root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT \
  /root/ssl-build/swksvr/server.crt
```

24

```
/root/ssl-build/swksvr/server.crt: OK
```

If this command returns an error, verify that you have created `RHN-ORG-TRUSTED-SSL-CERT` correctly and that the date and time configured on the server are correct.

5. Store the CA public certificate in the Spacewalk database so that it is available for use in provisioning client systems.

```
# rhn-ssl-dbstore -v --ca-cert=/root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT
Public CA SSL certificate:  /root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT
```

If the command returns an error, enter the command again, specifying a higher level of debugging such as `-vvv` to gather more information about the problem.

6. Generate and install the web server SSL package:

   a. Generate the web server SSL package.

```
# rhn-ssl-tool --gen-server --rpm-only --dir /root/ssl-build

...working...

Generating web server's SSL key pair/set RPM:
    /root/ssl-build/swksvr/rhn-org-httpd-ssl-key-pair-swksvr-1.0-rev.src.rpm
    /root/ssl-build/swksvr/rhn-org-httpd-ssl-key-pair-swksvr-1.0-rev.noarch.rpm

The most current Spacewalk Proxy Server installation process against RHN hosted
requires the upload of an SSL tar archive that contains the CA SSL public
certificate and the web server's key set.

Generating the web server's SSL key set and CA SSL public certificate archive:
    /root/ssl-build/swksvr/rhn-org-httpd-ssl-archive-swksvr-1.0-rev.tar

Deploy the server's SSL key pair/set RPM:
    (NOTE: the Red Hat Satellite or Proxy installers may do this step for you.)
    The "noarch" RPM needs to be deployed to the machine working as a
    web server, or Red Hat Satellite, or Spacewalk Proxy.
    Presumably 'swksvr.mydom.com'.
```

   You can use `rpm` to list the files that the packages install.

```
# rpm -qlp /root/ssl-build/swksvr/rhn-org-httpd-ssl-key-pair-swksvr-1.0-rev.src.rpm
rhn-org-httpd-ssl-key-pair-swksvr-1.0.tar.gz
rhn-org-httpd-ssl-key-pair-swksvr.spec
# rpm -qlp /root/ssl-build/swksvr/rhn-org-httpd-ssl-key-pair-swksvr-1.0-rev.noarch.rpm
/etc/httpd/conf/ssl.crt/server.crt
/etc/httpd/conf/ssl.csr/server.csr
/etc/httpd/conf/ssl.key/server.key
/etc/pki/spacewalk/jabberd/server.pem
```

   b. Install the web server SSL `noarch` package.

```
# rpm -Uhv /root/ssl-build/swksvr/rhn-org-httpd-ssl-key-pair-swksvr-1.0-rev.noarch.rpm
Preparing...                ########################################### [100%]
   1:rhn-org-httpd-ssl-key-p########################################### [100%]
```

7. Generate the public CA certificate package and make both the package and the CA public certificate file available to clients:

   a. Generate the public CA certificate package.

```
# rhn-ssl-tool --gen-ca --dir=/root/ssl-build --rpm-only
```

```
...working...
Generating CA public certificate RPM:
    /root/ssl-build/rhn-org-trusted-ssl-cert-1.0-rev.src.rpm
    /root/ssl-build/rhn-org-trusted-ssl-cert-1.0-rev.noarch.rpm

Make the public CA certificate publicly available:
    (NOTE: the Red Hat Satellite or Proxy installers may do this step for you.)
    The "noarch" RPM and raw CA certificate can be made publicly accessible
    by copying it to the /var/www/html/pub directory of your Red Hat Satellite or
    Proxy server.
```

You can use `rpm` to list the files that the packages install.

```
# rpm -qlp /root/ssl-build/rhn-org-trusted-ssl-cert-1.0-rev.src.rpm
rhn-org-trusted-ssl-cert-1.0.tar.gz
rhn-org-trusted-ssl-cert.spec
# rpm -qlp /root/ssl-build/rhn-org-trusted-ssl-cert-1.0-rev.noarch.rpm
/usr/share/rhn/RHN-ORG-TRUSTED-SSL-CERT
```

b. If a Spacewalk server or Spacewalk proxy is also configured as a client, install the public CA certificate `noarch` package on this system.

```
# rpm -Uhv /root/ssl-build/pub/rhn-org-trusted-ssl-cert-1.0-rev.noarch.rpm
Preparing...                 ######################################## [100%]
   1:rhn-org-trusted-ssl-cer######################################## [100%]
```

The public CA certificate is installed as `/usr/share/rhn/RHN-ORG-TRUSTED-SSL-CERT`.

c. Copy the `rhn-org-trusted-ssl-cert-1.0-rev.noarch.rpm` package and CA public certificate file to `/var/www/html/pub` for access by clients.

```
# cp /root/ssl-build/rhn-org-trusted-ssl-cert-1.0-rev.noarch.rpm /var/www/html/pub
# cp /root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT /var/www/html/pub
```

> **Note**
>
> If you do not copy the updated `RHN-ORG-TRUSTED-SSL-CERT` to `/var/www/html/pub`, the `osa-dispatcher` service fails to start.
>
> To verify that the installed copies of `RHN-ORG-TRUSTED-SSL-CERT` are identical, compare their digest values, for example:
>
> ```
> # sha1sum /root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT \
>   /usr/share/rhn/RHN-ORG-TRUSTED-SSL-CERT \
>   /var/www/html/pub/RHN-ORG-TRUSTED-SSL-CERT
> 74380a372bfa55d8ab7579bf01502c874b8aae84
>                          /root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT
> 74380a372bfa55d8ab7579bf01502c874b8aae84
>                          /usr/share/rhn/RHN-ORG-TRUSTED-SSL-CERT
> 74380a372bfa55d8ab7579bf01502c874b8aae84
>                          /var/www/html/pub/RHN-ORG-TRUSTED-SSL-CERT
> ```

The `rhn-org-trusted-ssl-cert-1.0-rev.src.rpm` package is usually not made available to clients.

8. On a Spacewalk server, stop the Spacewalk services, clear the `jabberd` database, and restart Spacewalk.

```
# spacewalk-service stop
# rm -Rf /var/lib/jabberd/db/*
```

```
# spacewalk-service start
```

On a Spacewalk proxy, restart the Spacewalk proxy services:

```
# rhn-proxy restart
```

9. On the remaining Spacewalk clients, download and install the public CA certificate package, for example:

```
# wget https://swksvr.mydom.com/pub/rhn-org-trusted-ssl-cert-1.0-rev.noarch.rpm
--2015-06-05 15:15:44--  https://swksvr.mydom.com/pub/rhn-org-trusted-ssl-cert-1.0-rev.noarch.rpm
Resolving swksvr.mydom.com... 192.168.1.3
Connecting to swksvr.mydom.com|192.168.1.3|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 4840 (4.7K) [application/x-rpm]
Saving to: "rhn-org-trusted-ssl-cert-1.0-2.noarch.rpm"

100%[=======================================>] 4,840       --.-K/s   in 0s

2015-06-05 15:15:44 (57.5 MB/s) - "rhn-org-trusted-ssl-cert-1.0-rev.noarch.rpm" saved [4840/4840]

# rpm -Uhv /root/ssl-build/pub/rhn-org-trusted-ssl-cert-1.0-rev.noarch.rpm
Preparing...                ######################################### [100%]
   1:rhn-org-trusted-ssl-cer######################################### [100%]
```

Note

If you subsequently replace the server certificate because it is revoked or it expires, you do not need to update the public CA certificate on the clients unless you change the CA that signs the server certificate.

Chapter 4 Configuring Inter-Server Synchronization

You can configure Inter-Server Synchronization (ISS) to synchronize channel content, channel permissions, and organizational trust settings between Spacewalk servers. The configuration of local non-content settings for users and organizations is not affected. One Spacewalk server acts as a *master* to provide content to any number of *slave* Spacewalk servers. Typical use cases include:

• The content on the slave servers is regularly synchronized with the master server to obtain the latest maintenance releases.

• Content is developed and tested on the master server before distribution to the slave servers.

• Slave servers have local content that is additional to that synchronized from the master server.

You can configure master servers that are themselves slaves of a higher-level master Spacewalk server. The usual ISS topology is a tree-like hierarchy, where there is one top-level master server and each slave has only one master, rather than a directed graph, where there might be several top-level master servers and each slave can have more than one master. If a slave has more than one master, you can designate one to be the default master with which the slave synchronizes.

You can use the `spacewalk-sync-setup` command to set up the relationships between master and slave Spacewalk servers or you can use the Spacewalk web interface to configure each server independently.

4.1 Enabling or Disabling Support for Slave Synchronization

By default, a Spacewalk server is configured to be able to act as a master server. Any slave servers that you configure for the master server will be able to synchronize from it. The following steps are not usually necessary unless you want to disable this feature on a Spacewalk server that acts only as a slave server.

To enable or disable support for slave synchronization on a Spacewalk server:

1. To disable ISS support, edit `/etc/rhn/rhn.conf` and set the value of `disable_iss` to 1:

   ```
   disable_iss=1
   ```

   To enable ISS support, edit `/etc/rhn/rhn.conf` and set the value of `disable_iss` to 0:

   ```
   disable_iss=0
   ```

2. Restart the `httpd` service:

   ```
   # service httpd restart
   ```

4.2 Configuring Master and Slave Spacewalk Servers Using spacewalk-sync-setup

> **Note**
>
> The `spacewalk-sync-setup` utility is available in the `spacewalk-utils` package.

On either of the Spacewalk servers, run the `spacewalk-sync-setup` command:

```
# spacewalk-sync-setup --apply --create-templates \
  --ms=master_swksvr_FQDN --ml=master_swadm --mp=master_swadm_passwd \
```

The software described in this documentation is either no longer supported or is in extended support.
Oracle recommends that you upgrade to a current supported release.
Configuring a Master Spacewalk Server Using the Spacewalk Web Interface

```
  --ss=slave_swksvr_FQDN --sl=slave_swadm --sp=slave_swadm_passwd
```

master_swksvr_FQDN is the fully qualified domain name of the Spacewalk server that will act as the master server. *master_swadm* and *master_swadm_passwd* are the Spacewalk administrator's user name and password for that server.

slave_swksvr_FQDN is the fully qualified domain name of the Spacewalk server that will act as the slave server. *slave_swadm* and *slave_swadm_passwd* are the Spacewalk administrator's user name and password for that server.

The following example shows typical output from running this command:

```
# spacewalk-sync-setup --apply --create-templates \
  --ms=swksvr.mydom.com --ml=swadm --mp=swpass \
  --ss=swksvr2.mydom.com --sl=swadm2 --sp=swpass2
INFO: Connecting to swadmin@swksvr.mydom.com
INFO: Connecting to swadmin@swksvr2.mydom.com
INFO: Generating master-setup file /root/.spacewalk-sync-setup/master.txt
INFO: Generating slave-setup file /root/.spacewalk-sync-setup/slave.txt
INFO: About to wget master CA cert: [wget -q -O
/usr/share/rhn/swksvr.mydom.com_RHN-ORG-TRUSTED-SSL-CERT
http://swksvr.mydom.com/pub/RHN-ORG-TRUSTED-SSL-CERT]
INFO: Applying master-setup /root/.spacewalk-sync-setup/master.txt
INFO: Applying slave-setup /root/.spacewalk-sync-setup/slave.txt
```

A copy of the master's CA certificate is stored on the slave as `/usr/share/rhn/swksvr.mydom.com_RHN-ORG-TRUSTED-SSL-CERT`.

You can now map local organizations on the slave server to organizations that the master server exports. See Section 4.5, "Mapping a Local Organization to an Exported Organization Using the Spacewalk Web Interface".

4.3 Configuring a Master Spacewalk Server Using the Spacewalk Web Interface

To configure a master Spacewalk server:

1. Go to **Admin**, select **ISS Configuration**, and then select the **Master Setup** tab.

2. On the Known Slave Instances page, click **+ Add new slave**.

3. On the Edit Slave Details page, enter the fully qualified domain name of the slave server and select or deselect the check boxes that configure slave and organization synchronization. For example, you might want to allow the slave to synchronize from the master but not want to synchronize all organizations to the slave.

4. Click **Create**.

    The page refreshes to allow you to select which organizations can be exported. By default, no organizations are selected.

5. Select the organizations that you want to allow to be exported to the slave and click **Allow Orgs**.

4.4 Configuring a Slave Spacewalk Server Using the Spacewalk Web Interface

To configure a slave Spacewalk server:

1. In a browser tab, navigate to **http://*master_swksvr_FQDN*/pub**, where *master_swksvr_FQDN* is the fully qualified domain name of the master Spacewalk server, and download the CA certificate file `RHN-ORG-TRUSTED-SSL-CERT` as `RHN-ORG-TRUSTED-SSL-CERT-MASTER`.

   Alternatively, you can use `wget` from the command line, for example:

   ```
   # wget -q -O /usr/share/rhn/RHN-ORG-TRUSTED-SSL-CERT-MASTER \
     http://master_swksvr_FQDN/pub/RHN-ORG-TRUSTED-SSL-CERT
   ```

2. In the Spacewalk web interface, go to **Admin**, select **ISS Configuration** and then select the **Slave Setup** tab.

3. On the Known Master Instances page, click **+ Add new master**.

4. On the Details for new Master page, enter the fully qualified domain name of the master server and the absolute path name of the master's CA certificate that you downloaded (`RHN-ORG-TRUSTED-SSL-CERT-MASTER`), and select whether the master will be the default one with which the slave synchronizes.

5. Click **Add new master**.

   The page refreshes to display a Configure Master-to-Slave Org Mappings section that allows you to configure local names for the organizations that the master exports. When you synchronize content, access permissions that you have configured for channels on the master server propagate to the slave server. You can choose which organizations (and thereby, any associated channel permissions) to map to a slave server.

   If necessary, create the local organizations that you want to map to the organizations that the master server exports.

   To create a local organization:

   a. Go to **Admin**, select **Organizations** and click **+ create new organization**.

   b. On the Create New Organization page, enter the details for the organization, including its name and the login details of its administrator.

   > **Note**
   >
   > You must create a new user to act as the organization's administrator. The Spacewalk administrator cannot perform this role.

   c. Click **Create Organization**.

   d. To return to the Configure Master-to-Slave Org Mappings section for the master server instance, go to **Admin**, select **ISS Configuration**, select the **Slave Setup** tab, and click the name of the master server instance.

6. In the Configure Master-to-Slave Org Mappings section, select the local organizations that map to the organizations that the master server exports.

   For each exported organization in the Master Org Name column, use the pull-down list in the Matching Local Org column to select the local organization that should map to the export organization. If you do not want to import an organization, select **NOT MAPPED**.

7. Click **Update**.

The software described in this documentation is either no longer supported or is in extended support.
Oracle recommends that you upgrade to a current supported release.
Mapping a Local Organization to an Exported Organization Using the Spacewalk Web Interface

4.5 Mapping a Local Organization to an Exported Organization Using the Spacewalk Web Interface

To map local organizations on a slave server to organizations exported by a master server:

1. To view that organizations that a master server exports, go to **Admin**, select **ISS Configuration**, select the **Slave Setup** tab, and click the name of the master server instance.

2. Create the local organizations that you want to map to the organizations that the master server exports.

    To create a local organization:

    a. Go to **Admin**, select **Organizations** and click **+ create new organization**.

    b. On the Create New Organization page, enter the details for the organization, including its name and the login details of its administrator.

    > **Note**
    >
    > You must create a new user to act as the organization's administrator. The Spacewalk administrator cannot perform this role.

    c. Click **Create Organization**.

    d. To return to the Configure Master-to-Slave Org Mappings section for the master server instance, go to **Admin**, select **ISS Configuration**, select the **Slave Setup** tab, and click the name of the master server instance.

3. In the Configure Master-to-Slave Org Mappings section, select the local organizations that map to the organizations that the master server exports.

    For each exported organization in the Master Org Name column, use the pull-down list in the Matching Local Org column to select the local organization that should map to the export organization. If you do not want to import the organization, select **NOT MAPPED**.

4. Click **Update**.

4.6 Synchronizing Software Channels on a Slave Server

To synchronize a software channel, use the `satellite-sync` command on the slave server.

```
# satellite-sync [--iss-parent=master_swksvr_FQDN] [--orgid=N] -c channel_label
```

`channel_label` specifies the label of the software channel to synchronize from the master server.

The argument to the `--orgid` option specifies the ID of the organization on the master that provides the channel. If not specified, the Spacewalk Default Organization with ID 1 is assumed,

`master_swksvr_FQDN` is the fully qualified domain name of the master Spacewalk server. If not specified and the slave server has more than one master, the default master server is assumed.

The following example shows typical output from running this command to perform an initial synchronization of a software channel on a slave server from the Spacewalk Default Organization on the default master server:

```
# satellite-sync -c oraclelinux6-x86_64-patch
```

```
16:16:52 Spacewalk - live synchronization
16:16:52    url: https://swksvr.mydom.com
16:16:52    debug/output level: 1
16:16:52    db:  c##spacewalk2/<password>@//odbsvr.mydom.com/orcl.mydom.com
16:16:52
16:16:52 Retrieving / parsing orgs data
16:16:52 orgs data complete
16:16:52
16:16:52 Retrieving / parsing channel-families data
16:16:52 channel-families data complete
16:16:52
16:16:52 Retrieving / parsing product names data
16:16:52 product names data complete
16:16:52
16:16:52 Retrieving / parsing arches data
16:16:53 arches data complete
16:16:53
16:16:53 Retrieving / parsing additional arches data
16:16:53 additional arches data complete
16:16:53
16:16:53 Retrieving / parsing channel data
16:16:54    p = previously imported/synced channel
16:16:54    . = channel not yet imported/synced
16:16:54    base-channels:
16:16:54        NONE RELEVANT
16:16:54    oraclelinux6-x86_64:
16:16:54      . oraclelinux6-x86_64-patch    1367    full import from Fri Jul 10 13:02:52 2015
16:16:54
16:16:54 Channel data complete
16:16:54
16:16:54 Retrieving short package metadata (used for indexing)
16:16:54    Retrieving / parsing short package metadata: oraclelinux6-x86_64-patch (1367)
16:17:01 Diffing package metadata (what's missing locally?): oraclelinux6-x86_64-patch
         _____
Diffing:   ######################################## - complete
16:17:04
16:17:04 Downloading package metadata
16:17:04    Retrieving / parsing *relevant* package metadata: oraclelinux6-x86_64-patch (1357)
16:17:04    * WARNING: this may be a slow process.
           _____
Downloading:######################################## - complete
16:42:30
16:42:30 Downloading rpm packages
16:42:30    Fetching any missing RPMs: oraclelinux6-x86_64-patch (1357)
16:42:53    Total size: 5.31 GiB
16:47:53 Processing rpm packages complete
16:47:53
16:47:53 Importing package metadata
16:47:53    Importing *relevant* package metadata: oraclelinux6-x86_64-patch (1357)
           _____
Importing:  ######################################## - complete
18:06:44
18:06:44 Linking packages to channels
18:07:02
18:07:02 Downloading errata data
18:07:02    Retrieving / parsing errata data: oraclelinux6-x86_64-patch (216)
           _____
Downloading:######################################## - complete
18:07:06 Downloading errata data complete
18:07:06
18:07:06 Downloading kickstartable trees metadata
18:07:06    Retrieving / parsing kickstart data: oraclelinux6-x86_64-patch (NONE RELEVANT)
18:07:06
18:07:06 Downloading kickstartable trees files
18:07:06    Retrieving / parsing kickstart tree files: oraclelinux6-x86_64-patch (NONE RELEVANT)
```

```
18:07:06
18:07:06 Importing channel errata
18:07:13    Importing *relevant* errata: oraclelinux6-x86_64-patch (468)

           _____
Downloading:#################################### - complete
18:07:31    No new kickstartable tree to import
    Import complete:
        Begin time: Fri Jul 10 16:16:51 2015
        End time:   Fri Jul 10 18:07:31 2015
        Elapsed:    1 hours, 50 minutes, 40 seconds
```