

Introduction to Space Walk



Space Walk

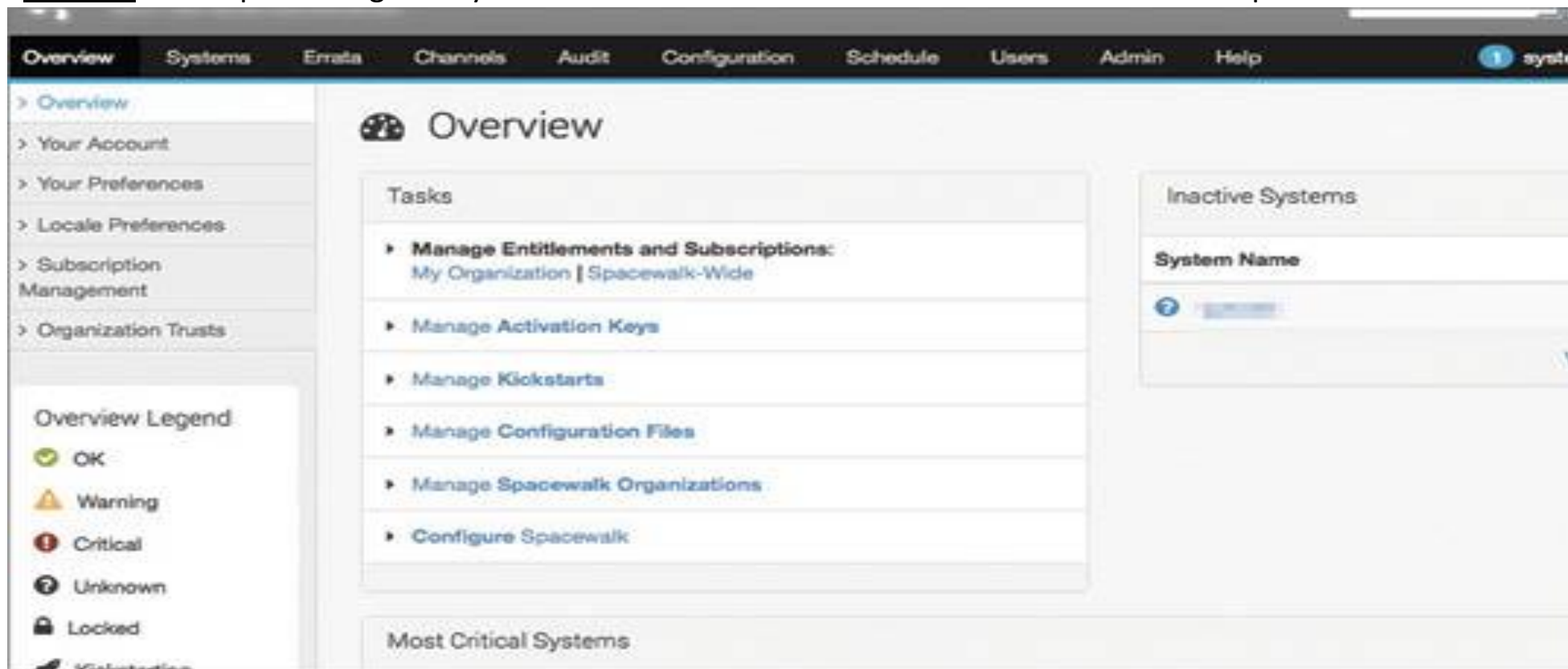
- Spacewalk is an open source Linux server management software, a version of Red Hat® Satellite server. Red Hat is one of the big sponsors of Spacewalk and it is also community supported. It can support Red Hat, SUSE and Oracle Linux agents. Prior to version 2.3, Solaris agents were also supported. Spacewalk has been developed in a combination of Python, Perl and, in recent versions, Java too.

Inventory your systems (hardware and software information)

- Install and update software on your systems
- Collect and distribute your custom software packages into manageable groups
- Provision (kickstart) your systems
- Manage and deploy configuration files to your systems
- Provision virtual guests
- Start/stop/configure virtual guests
- Distribute content across multiple geographical sites in an efficient manner

Overview

Spacewalk is a project where you can make use of the latest free & open source management technologies. Changes will occur frequently in the Spacewalk project to fulfill this mission. However, if your needs require a level of stability and support, you may decide that a Red Hat Satellite subscription is right for you. Please refer to the table below for some direct comparisons:



Overview

- The Red Hat Satellite server, including the web interface and back-end, as well as Red Hat Satellite Proxy Server and all clients that connect to Satellite are now open source. Satellite can now run on PostgreSQL (default) or optional as an External Database, Oracle Database.
- Spacewalk can run on Oracle or PostgreSQL.
- Both Satellite and Spacewalk are completely open source
- Spacewalk as a project officially became an open source, community driven project in June 2008, and it is the upstream for the Red Hat Satellite & Satellite Proxy products.

History and development

- Red Hat formally released Spacewalk as open source(GPLv2) in June 2008
- Red Hat continues to sponsor and support Spacewalk as the upstream Red Hat Satellite 5. however that participation is anticipated to diminish as Red Hat Satellite 5 enters the final phases of its lifecycle.
- Satellite 5.3 was the first version to be based on upstream Spacewalk code
- Spacewalk is not and can never be upstream for Red Hat Satellite 6 released in September 2014^{[11][12]} due to it being a ground up rebuild with a different toolset.
- The Spacewalk project can continue to grow and flourish providing the community continue to find it a useful tool and a willing to support it.

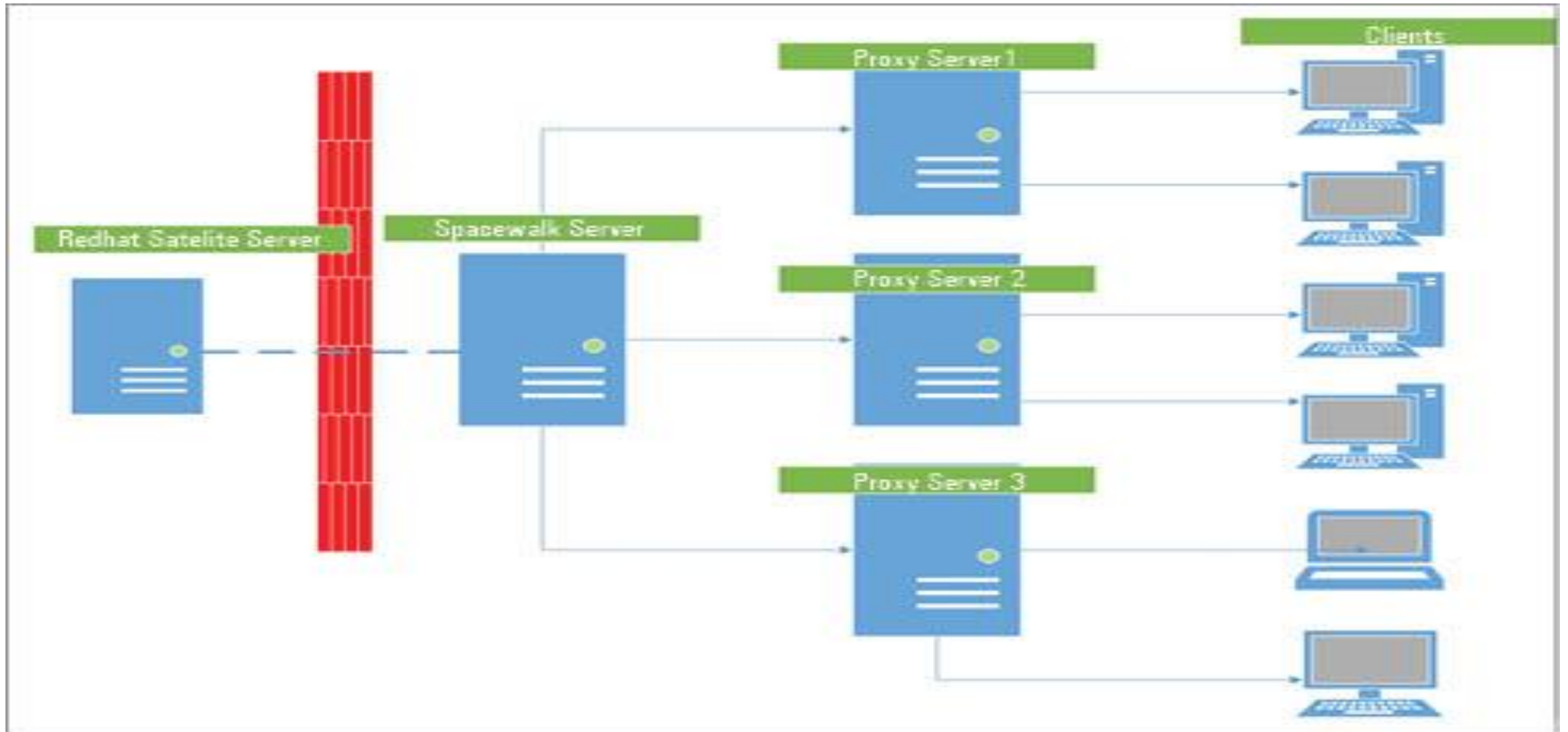
Releases of Space Walk

Release	Release Date	Features
Spacewalk 2.6	29-Nov-16	See Release Notes
Spacewalk 2.5	8-Jun-16	See Release Notes
Spacewalk 2.4	7-Oct-15	See Release Notes
Spacewalk 0.4	15-Jan-09	Release Announcement
Spacewalk 0.3	7-Nov-08	Release Announcement
Spacewalk 0.2	16-Sep-08	Release Announcement
Spacewalk	17-Jun-08	Initial Release Announcemen

Difference between Satellite and Spacewalk

	Satellite	Spacewalk
Primary Benefits	Stable and supported.	The latest technology released early and often.
Feature selection and Integration	Red Hat	Red Hat and developer community
Development Model	Open source	Open source
Architectures	x86_64 & s390x	x86_64
Managed Systems	Red Hat Enterprise Linux	Fedora, CentOS, SLE and Debian
Red Hat Support Options	Many, including 24x7 premium with unlimited incidents	None (community supported)
Content Stream	Direct via Red Hat Network	Manual Import
Release Interval	9-12 months	2-5 months
Testers	Red Hat	Community
Maintenance and updates	Available via Red Hat Network	Community-driven
Where to Buy	Contact Red Hat Sales, and partners	Free download
Price	Annual subscription, multiple offerings	Free download

Spacewalk Architecture



Spacewalk Installation Instructions

- **Prerequisites**

- Outbound open ports 80, 443
- Inbound open ports 80, 443, 5222 (only if you want to push actions to client machines) and 5269 (only for push actions to a Spacewalk Proxy), 69 udp if you want to use tftp
- Storage for database: 250 KiB per client system + 500 KiB per channel + 230 KiB per package in channel (i.e. 1.1GiB for channel with 5000 packages)
- Storage for packages (default /var/satellite): Depends on what you're storing; Red Hat recommend 6GB per channel for their channels
- 2GB RAM minimum, 4GB recommended
- Make sure your underlying OS is fully up-to-date.
- If you use LDAP as a central identity service and wish to pull user and group information from it, see [SpacewalkWithLDAP](#)
- In the following steps we assume you have a default, vanilla installation of your operating system, without any customized setup of yum repositories, user management, security, etc.

Setting up Spacewalk repo

RPM downloads of the project are available through yum repositories at

- <http://yum.spacewalkproject.org/> - Binary RPMs

To use this repository easily, install spacewalk-repo package with commands below

Red Hat Enterprise Linux 6, Scientific Linux 6, CentOS 6

```
#rpm -Uvh http://yum.spacewalkproject.org/2.5/RHEL/6/x86\_64/spacewalk-repo-2.5-3.el6.noarch.rpm
```

Red Hat Enterprise Linux 7, Scientific Linux 7, CentOS 7

```
#rpm -Uvh http://yum.spacewalkproject.org/2.5/RHEL/7/x86\_64/spacewalk-repo-2.5-3.el7.noarch.rpm
```

- **Red Hat Optional Server (Red Hat Enterprise Linux)**
- When using Red Hat Enterprise Linux 6 or 7, make sure you are subscribed to the appropriate Red Hat Optional Server channel:
- *Red Hat Optional Server 6 , OR*
- *Red Hat Optional Server 7*

Installing PostgreSQL

Spacewalk uses database server to store its primary data. It supports either PostgreSQL (version 8.4 and higher) or Oracle RDBMS (version 10g or higher).

- PostgreSQL server, set up by Spacewalk (embedded) To configure Postgre-sql on a dedicated server, it's better to use a standalone server.
- To install on a standalone server, please follow the steps shown below:

```
#yum install postgresql-server
```

```
#yum install postgresql-pltcl
```

```
#chkconfigpostgresql on =>To make the service running while next reboot
```

```
#service postgresqlinitdb =>Toconfig DB
```

```
#service postgresql start => To start the service
```

To install the Spacewalk-setup-PostgreSQL package, type:

```
# yum install spacewalk-setup-PostgreSQL
```

INSTALLING SPACE WALK

We now need to configure Spacewalk. If you are using the default database as Postgre-sql from the local server, run the following command:

```
#spacewalk-setup --disconnected
```

If you want to use Oracle (either on the local server or another server), use the following command:

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

If you want to use Postgre-sql from another server, use the command shown below:

```
#spacewalk-setup --disconnected --external-postgresql
```

INSTALLING SPACE WALK

```
#spacewalk-setup --disconnected --answer-file=<answer file>
```

```
admin-email = root@localhost.com
```

```
ssl-set-org = spacewalk.org
```

```
ssl-set-org-unit = spacewalk
```

```
ssl-set-city = chennai
```

```
ssl-set-state = Tamilnadu
```

```
ssl-set-country = India
```

```
ssl-password = walker2
```

```
ssl-set-email = root@localhost
```

```
ssl-config-sslhost = Y
```

```
db-backend=oracle
```

```
db-name=spaceschema
```

```
db-user=spaceuser
```

```
db-password=password1
```

```
db-host=localhost
```

```
db-port=1521
```

```
enable-tftp=Y
```

Restart the service by issuing the following command:

```
#!/usr/sbin/spacewalk-service restart
```

Configuring the firewall

Spacewalk needs various inbound ports to be accessible. Use `system-config-firewall` or edit `/etc/sysconfig/iptables`, adding the ports needed -- 80 and 443.

On a system with `firewalld` use `firewall-cmd --add-service=http ; firewall-cmd --add-service=https`.

Add port 5222 if you want to push actions to client machines and 5269 for push actions to a Spacewalk Proxy, 69 udp if you want to use tftp.

Firewall ports used in Spacewalk include:

- 69: TFTP (PXE provisioning)
- 80: Spacewalk Web interface
- 443: Spacewalk Web interface (SSL)
- 4545: Spacewalk monitoring
- 5222: If you plan to push actions to client systems
- 5269: If you push actions to a Spacewalk proxy server
- 9055: Oracle XE Web access

Configuring Spacewalk

Your Spacewalk server should have a resolvable fully-qualified domain name (FQDN) such as 'hostname.domain.com'.

If the installer complains that the hostname is not the FQDN, do not use the `--skip-fqdn-test` flag to skip !

The setup requires that the database account has a password.

Note: Please don't use * '#' * (number sign/pound/hash) and * '@' * in your database password otherwise installation will fail.

Once the Spacewalk RPM is installed you need to configure the application.

If you wish to use the default PostgreSQL database, and have installed

spacewalk-setup-postgresql, just run

spacewalk-setup

Configuring Spacewalk

If you are planning to use your own database setup (either locally or on a separate machine), then run

```
#spacewalk-setup --external-oracle
```

if you are using an Oracle database, OR

```
#spacewalk-setup --external-postgresql
```

Should the spacewalk-setup fail, check the error it gives you and also investigate the logs in /var/log/rhn, as well as the logs from your database server, Apache server and tomcat

Spacewalk consists of several services. Each of them has its own init.d script to stop/start/restart. If you want manage all spacewalk services at once use

```
#!/usr/sbin/spacewalk-service [stop|start|restart].
```


Space walk Proxy setup

To boost the performance of patching or deploying a package, we need the help of proxy servers. Shown below are the steps to create a proxy server. Make sure the Spacewalk repos, as discussed earlier, have been set up and are available.

Install the following packages to enable the proxy server. This will install many dependency packages and, at the end, install the GPG key (answer `y` to the question).

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

Configure the Spacewalk-proxy, as follows:

```
# configure-proxy.sh
```

If you are facing any issues regarding the proxy, always refer to the log files below for more information:

*/var/log/httpd/** – Apache logs

*/var/log/squid/** – Squid logs

/var/log/rhn/rhn_proxy_broker.log – Proxy Broker Server

/var/log/rhn/rhn_proxy_redirect.log – SSL Redirect Server

Installing Client packages

Install Spacewalk client packages, as follows:

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

3. Install the CA certificate from the Spacewalk site to enable SSL communication, as follows:

```
# rpm -Uvhhttp://YourSpacewalk.example.com/pub/rhn-org-trusted-ssl-cert-1.0-1.noarch.rpm
```

4. To register the client with Spacewalk, we will use the activation key which we created at the end of the installation:

```
rhnreg_ks serverUrl=https://<myspacew
```

```
alkserver>/XMLRPC --sslCACert=/usr/share/rhn/RHN-ORG-TRUSTED-SSL-CERT --activationkey=<key-with-  
channel>
```

Pushing packages to clients

We can configure the Spacewalk server in such a way that it allows pushing the packages or runs a command on remote clients.

This feature is serviced by OSA group services (osad, osa-dispatcher and jabberd). The osa-dispatcher will check the database periodically for any actions that are pending or available.

1. To set up OSA services on the server, follow the steps given below:

```
#yum install osa-dispatcher ==> install OSA dispatcher
```

```
#chkconfigosa-dispatcher on
```

```
#service osa-dispatcher start
```

Pushing packages to clients

To set up OSA services on the client, follow these steps:

d. Install osad:

```
#yum install osad
```

Download the trusted cert to the client:

```
#cd /usr/share/rhn/
```

```
#wget http://spacewalk.example.org/pub/RHN-ORG-TRUSTED-SSL-CERT
```

Modify the `/etc/sysconfig/rhn/osad.conf` file:

```
osa_ssl_cert = /usr/share/rhn/RHN-ORG-TRUSTED-SSL-CERT
```

Start osad:

```
#service osad start
```

Now you can start enjoying patching on multiple servers with a single click.

Spacewalk Upgrade Instructions

- These are upgrade instructions for upgrading Spacewalk 2.5 to Spacewalk 2.6
- These upgrade instruction apply to Spacewalk installations meeting the following criteria:
- Spacewalk 2.5 running on Red Hat Enterprise Linux/CentOS/Scientific Linux 6/7 Server, or Fedora 22/23.
- Your Spacewalk uses one of Oracle 10g (including XE) / Oracle 11g / PostgreSQL 8.4+ as a database backend.
- In most cases it's possible to perform Package upgrade and Schema upgrade steps from any previous version to the latest one directly (e.g. from 1.6 to 2.6). Make sure you have a valid backup in case anything will go wrong.

Database and configuration backup

- For existing configuration files, create a backup of everything under `/etc/sysconfig/rhn` `/etc/rhn` and `/etc/jabberd`
- Backup your SSL build directory, ordinarily `/root/ssl-build`
- *BACKUP YOUR DATABASE.* For instructions on how to create a backup of your existing Spacewalk database consult either Oracle / PostgreSQL documentation or contact your DBA

Upgrade of Spacewalk configuration

use spacewalk-setup to upgrade Spacewalk configuration.

If you are using the default database setup, run:

```
# spacewalk-setup --upgrade
```

If you are using an external database, then run

```
# spacewalk-setup --external-$DB --upgrade
```

where DB will be either oracle or postgresql, depending on what your external database is.

Restore some of the custom values you might have set previously in `/etc/rhn/rhn.conf` from the backup of your configuration files, such as:

```
debug = 3
```

```
pam_auth_service = rhn-satellite
```

Restart Spacewalk

If you are running Spacewalk on Fedora run following command before starting Spacewalk services

```
systemctl daemon-reload
```

Then, start all Spacewalk services:

```
# /usr/sbin/spacewalk-service start
```


spacewalk-service(8) - Linux man page

- Spacewalk server consist from several services. While each of such service has its own init.d script and you can start/stop each service separately, you usually want to control all services together. This script does exactly this.

Options -- start|stop|status|reload|restart Behave exactly as service for every Spacewalk service.

- enable|disable-- Behave exactly as chkconfig on/off for every Spacewalk service.
- List all Spacewalk services and if it is enabled or disabled.

--no-wait-for-tomcat

By default, spacewalk-service wait until tomcat is able to answer requests. If you specify this option, spacewalk-service will continue immediately when tomcat service will start. --exclude SERVICE

- It will not start/stop SERVICE .
- --level LEVEL
- Pass LEVEL to chkconfig when you specify option enable.