
Chapter 1. Upgrading Evergreen from 1.4 to 1.6

Abstract

This Chapter will explain the step-by-step process of upgrading Evergreen from version 1.4 to 1.6, including steps to upgrade OpenSRF. Before upgrading, it is important to carefully plan an upgrade strategy to minimise system downtime and service interruptions. All of the steps in this chapter are to be completed from the command line.

1. Stop Evergreen and back up data.
 1. As *root*, stop the Apache web server.
 2. As the *opensrf* user, stop all Evergreen and OpenSRF services:

```
osrf_ctl.sh -l -a stop_all
```
 3. Back up of the */openils* directory.
 4. Back up the evergreen database.
2. Upgrade OpenSRF to 1.2
 1. As the *opensrf* user, download and extract the source files for OpenSRF 1.2:

```
wget \
  http://open-ils.org/downloads/OpenSRF-1.2.2.tar.gz
tar xzf OpenSRF-1.2.2.tar.gz
```

A new directory *OpenSRF-1.2.2* will be created

Note

For the latest edition of OpenSRF, check the Evergreen download page at <http://www.open-ils.org/downloads.php>.

2. As the root user, install the software prerequisites using the automatic prerequisite installer.

```
aptitude install make
cd /home/opensrf/OpenSRF-1.2.2
```

Replace *[distribution]* below with the following value for your distribution:
 - *debian-etch* for Debian Etch (4.0)
 - *debian-lenny* for Debian Lenny (5.0)
 - *ubuntu-hardy* for Ubuntu Hardy Heron (8.04)
 - *ubuntu-intrepid* for Ubuntu Intrepid Ibex (8.10)
 - *ubuntu-jaunty* for Ubuntu Jaunty Jackalope (9.04)

- *ubuntu-karmic* for Ubuntu Karmic Koala (9.10)

```
make -f src/extras/Makefile.install [distribution]
```

This will install a number of packages required by OpenSRF on your system, including some Perl modules from CPAN. You can say “no” to the initial CPAN configuration prompt to allow it to automatically configure itself to download and install Perl modules from CPAN. The CPAN installer will ask you a number of times whether it should install prerequisite modules - say “yes”.

3. As the `opensrf` user, configure and compile OpenSRF:

Note

You can include the `--enable-python` and `--enable-java` configure options if you want to include support for Python and Java, respectively.

```
cd /home/opensrf/OpenSRF-1.2.2
./configure --prefix=/openils --sysconfdir=/openils/conf
make
```

4. As the root user, return to your OpenSRF build directory and install OpenSRF:

```
cd /home/opensrf/OpenSRF-1.2.2
make install
```

5. As the root user, change the ownership of the installed files to the `opensrf` user:

```
chown -R opensrf:opensrf /openils
```

6. Restart and Test OpenSRF

```
osrf_ctl.sh -l -a start_all
/openils/bin/srfsh
srfsh# request opensrf.math add 2 2
```

You should see output like:

```
Received Data: 4
-----
Request Completed Successfully
Request Time in seconds: 0.007519
-----
srfsh#
```

If test *Completed Successfully* move onto next section. If not, refer to troubleshooting section of this documentation.

3. Upgrade Evergreen to 1.6

1. As the *opensrf* user, download and extract Evergreen 1.6

```
wget \
http://evergreen-ils.org/downloads/Evergreen-ILS-1.6.0.3.tar.gz
tar xzf Evergreen-ILS-1.6.0.3.tar.gz
```

Note

For the latest edition of Evergreen check the Evergreen download page at <http://www.open-ils.org/downloads.php>

2. As the *root* user, install the prerequisites:

```
cd /home/opensrf/Evergreen-ILS-1.6.0.3
```

On the next command, replace *[distribution]* with one of these values for your distribution of Debian or Ubuntu:

- *debian-etch* for Debian Etch (4.0)
- *debian-lenny* for Debian Lenny (5.0)
- *ubuntu-hardy* for Ubuntu Hardy Heron (8.04)
- *ubuntu-intrepid* for Ubuntu Intrepid Ibex (8.10)
- *ubuntu-jaunty* for Ubuntu Jaunty Jackalope (9.04)
- *ubuntu-karmic* for Ubuntu Karmic Koala (9.10)

```
make -f Open-ILS/src/extras/Makefile.install [distribution]
```

3. As the *opensrf* user, configure and compile Evergreen:

```
cd /home/opensrf/Evergreen-ILS-1.6.0.3
./configure --prefix=/openils --sysconfdir=/openils/conf
make
```

4. As the *root* user, install Evergreen:

```
make STAFF_CLIENT_BUILD_ID=rel_1_6_0_3 install
```

5. Change to the Evergreen installation directory:

```
cd /home/opensrf/Evergreen-ILS-1.6.0.3
```

6. As the *root* user, build *live-db-setup.pl* for the cgi-bin bootstrapping scripts and *offline-config.pl* for the offline staff client data uploader:

```
perl Open-ILS/src/support-scripts/eg_db_config.pl \  
--create-bootstrap --create-offline --user evergreen \  
--password evergreen --hostname localhost --port 5432 \  
--database evergreen
```

7. As the *root* user, change all files to be owned by the *opensrf* user and group:

```
chown -R opensrf:opensrf /openils
```

8. Update the Evergreen database

```
psql -U evergreen -h localhost \  
-f Open-ILS/src/sql/Pg/1.4.0.5-1.6.0.0-upgrade-db.sql evergreen  
  
psql -U evergreen -h localhost \  
-f Open-ILS/src/sql/Pg/1.6.0.0-1.6.0.1-upgrade-db.sql evergreen  
  
psql -U evergreen -h localhost \  
-f Open-ILS/src/sql/Pg/1.6.0.1-1.6.0.2-upgrade-db.sql evergreen  
  
psql -U evergreen -h localhost \  
-f Open-ILS/src/sql/Pg/1.6.0.2-1.6.0.3-upgrade-db.sql evergreen  
  
psql -U evergreen -h localhost -f 1.6.0-mmbxs-cleanup.sql \  
evergreen
```

9. As the *opensrf* user, copy */openils/conf/oils_web.xml.example* to */openils/conf/oils_web.xml* (needed for acquisitions templates).

```
cp /openils/conf/oils_web.xml.example \  
/openils/conf/oils_web.xml
```

10. Update *opensrf_core.xml* and *opensrf.xml* by copying the new example files (*/openils/conf/opensrf_core.xml.example* and */openils/conf/opensrf.xml*).

```
cp /openils/conf/opensrf_core.xml.example \  
/openils/conf/opensrf_core.xml  
  
cp /openils/conf/opensrf.xml.example /openils/conf/opensrf.xml
```

11. Update *opensrf.xml* with the database connection info:

```
perl Open-ILS/src/support-scripts/eg_db_config.pl \  
--update-config --service all --user evergreen \  
--password evergreen --hostname localhost --port 5432 \  
--database evergreen
```

```
--database evergreen
```

12. Update `/etc/apache2/startup.pl` by copying the example from `Open-ILS/examples/apache/startup.pl`.
 13. Update `/etc/apache2/eg_vhost.conf` by copying the example from `Open-ILS/examples/apache/eg_vhost.conf`.
 14. Update `/etc/apache2/sites-available/eg.conf` by copying the example from `Open-ILS/examples/apache/eg.conf`.
 15. Recover customizations you have made to the Apache configuration files. For example, if you purchased an SSL certificate, you will need to edit `eg.conf` to point to the appropriate SSL certificate files.
4. Restart Evergreen and Test
 1. As the `opensrf` user, start all Evergreen and OpenSRF services:

```
osrf_ctl.sh -l -a start_all
```
 2. As the `opensrf` user, run `autogen` to refresh the static organizational data files:

```
cd /openils/bin  
./autogen.sh -c /openils/conf/opensrf_core.xml -u
```
 3. Start `srfsh` and try logging in using your Evergreen username and password:

```
/openils/bin/srfsh  
srfsh% login username password
```
 4. Start the Apache web server.

Troubleshooting:

If you encounter errors, refer to the troubleshooting section of this documentation for tips on finding solutions and seeking further assistance from the Evergreen community.