

# 6.x Subversion

## Setup

```
sudo apt-get install subversion
```

Web-Dav apache module is contained in the libapache2-svn

```
sudo apt-get install libapache2-svn
```

To check if Web-Dav is enabled run the 2 following commands

```
sudo a2enmod dav
sudo a2enmod dav_svn
```

Before creating repositories users and a group must be defined to have access to create or it will not work so we add the group subversion and the user www-data which is what apache uses,

```
sudo mkdir /opt/svn-repos
sudo addgroup --gid 3001 subversion
sudo chown -R www-data:subversion svn-repos
```

Now our model we do not want Subversion users tied to the operating system. So we are going to rely on our catch all server admin id, serveradmin,

```
sudo usermod -a -G subversion serveradmin
```

## Creating Repositories

First go to or create the directory you want to create the repositories in for this example we've created the svn-repos directory in /opt/,

The svnadmin command allows you to create svn repositories and allows you to perform several maintenance operations on the repositories.

Prior to Subversion 1.2, the default was to use Berkeley DB; the default is now FSFS. You can explicitly choose the filesystem type using the --fs-type argument, which accepts as a parameter either fsfs or bdb.

```
su - serveradmin
svnadmin create --fs-type fsfs /opt/svn-data/hamsters
```

## Setting Permissions on the Repository for WebDAV

After the repository is created, in order for the WebDAV process to work with the repository you need to change the permissions,

```
cd /opt/svn-data
sudo chown -R www-data:subversion hamsters
sudo chmod -R g+rws hamsters
```

## Backup and Restore Commands

Checks the number of revisions in a repository

```
svnlook info hamsters
```

Dumps all revisions of the repository into a file in this case hamstersdumpfile

```
svnadmin dump /opt/svn-repos/hamsters > hamstersdumpfile
```

Dumps a revision # that you decide. The other command dumps a range of revisions that you choose.

```
svnadmin dump hamsters -r # > hamsters.rev.#.dumpfile
svnadmin dump hamsters -r 50:100 > hamsters.revs.50-100.dumpfile
```

Loads repository from dump file.

```
svnadmin load /opt/svn-repos/gerbils < hamstersdumpfile
```

## Setting Permissions on the Repository for WEBDAV

Similar to creating a repository permissions must also be set,

```
cd /opt/svn-data
sudo chown -R www-data:subversion gerbils
sudo chmod -R g+rws gerbils
```

## Universally Unique Identifiers

Universally Unique Identifiers or UUID verifies the identity of a repository. Usually a unique UUID for each repository is desired except for situations between two repositories that is under migration of data. Using the same UUID will ensure a perfect replica of the repository being replaced and users connecting to it won't see a different repository.

First check the old repository UUID.

```
svnlook uuid /opt/svn-data/hamsters
```

As an example the output should look something like

693708d0-288a-4d90-aadb-1524a6fdb94

Then replace the new repository uuid...

```
svnadmin setuuid /opt/svn-data/hamsters \
```

...

Still to write is an entry on UUID when restoring from a shadow repository.

There should be instructions here... if using a shadow strategy, to record the UUID of the main repository in case you lose it.

## References

<http://svnbook.red-bean.com/en/1.4/svn.intro.whatish.html>

[http://www.howtoforge.com/debian\\_subversion\\_websvn](http://www.howtoforge.com/debian_subversion_websvn) - Debian instructions more straight forward

<http://confluence.atlassian.com/display/CROWD/Integrating+Crowd+with+Subversion> - Integrating with Crowd

<http://svnbook.red-bean.com/nightly/en/svn-book.html#svn.reposadmin.maint.migrate> - Indepth explanation on migrating repositories