
= Howto Move a Subversion Repository from One Server to Another =

Recently I had to move a subversion (svn) repository to another or lets say new server. I needed to upgrade the servers hardware and software, so a complete migration had to be done. After reading the Subversion book [1], I found that actually moving a subversion repository from one server to another, and preserve all your version/revision history is not a very difficult task to do! I was really surprised of that and I am sure after you finish reading this documentation you shall be too ;)

At work, we had more than one repository on the svn server, and for each project we had a separate repository. In the beginning I thought that its a complicated task to do, but found out that I am wrong.

The Subversion move/migrate from one server to another, shall be divided into 3 steps:

- Backup
- Create
- Import

Step 1: Backup

Backup all your old repositories, and this can be done by dumping the subversion repository to a dump file:

```
# svnadmin dump /path/to/repository > svn_repo_name.dump
```

The dump file contains all the revisions you have made to your svn repository, and thats why it will probably be quite large. Not to foget that it even includes files you may have deleted in a previous revision. Example:

```
# svnadmin dump /usr/local/svn > mysvn_repo.dump
```

Step 2: Create

In this step we need to create an empty repository. This can be done using the following command:

```
# svnadmin create /path/to/repository
```

Example:

```
# svnadmin create /usr/local/newsvn
```

Step 3: Import

Now copy the dump file from the old svn server to the new one, this is left to you. Use whatever technique to do so. After you finish copying the dump file from the old svn server, its time to import our old repository into the new repository we just created. This can be done using the following command:

```
# svnadmin load /path/to/repository < svn_repo_name.dump
```

Example:

```
# svnadmin load /usr/local/newsvn < mysvn_repo.dump
```

Note:

If you want to force subversion to use the same UUID for the new repository as the old repository. All you need to do is add the following to the load command:

```
# svnadmin load --force-uuid /usr/local/newsvn < mysvn_repo.dump
```

Now we have our repository working on our new server.

= APACHE SVN CONFIGURATION =

If you want to configure apache to work with svn and be able to access it from a remote location, then please read the apache configuration howto below.

Make sure you have the Apache configuration directives for serving Subversion repositories through Apache HTTP Server installed, if not install it:

- For Redhat/CentOS -

```
# yum install mod_dav_svn
```

- For Debian/Ubuntu -

```
# apt-get install libapache2-svn
```

I shall continue working with CentOS for this tutorial. There shall be simple changes such as file locations if you continue with Debian or Ubuntu, but that shall not be a hard task to do, right? :)

Now to configure our subversion to be accessed from the web, let add the directives needed. Open the subversion configuration file found in the Apache conf.d directory:

```
# vim /etc/httpd/conf.d/subversion.conf
```

Add the following (Please change svn repository paths to suite your needs):

```
<Location /svn>
    DAV svn
    SVNPath /usr/local/svn
    AuthzSVNAccessFile /usr/local/svn/conf/authz
    Satisfy Any
    Require valid-user
    AuthType Basic
    AuthName "Subversion repos"
    AuthUserFile /etc/httpd/htpasswd
</Location>
```

For security reasons and Access Controls to be used with the subversion, we used the "**AuthzSVNAccessFile**" option. This is to help us make acl's on our repositories and give specific access rights to specific groups. The file holding our ACL in the above example is "**/usr/local/svn/conf/authz**", you may use your own file location and name. Also, we are using here basic apache authentication, and we are storing user names and passwords in the file "**/etc/httpd/htpasswd**", you may use your own file location and name too. Save the "**subversion.conf**" file and exit.

Now lets add some users for web access. For the first time, we shall be using the **-c** option to create the file, after that when creating another user, we won't be needing this, because if you use it? you shall erase the current db and create a new one. Let's add a user called "**admin**", "**ali**", and "**mohamed**":

```
# htpasswd -cm /etc/httpd/htpasswd admin
# htpasswd -m /etc/httpd/htpasswd ali
# htpasswd -m /etc/httpd/htpasswd mohamed
```

Enter a password twice for each.

I shall suppose now that I want to create 3 groups, with the following names, permissions, and repository access level:

1- Administration Group

name: admin

perm: rw

repo: /

2- Staff Group

name: staff

perm: rw

repo: /files

3- Documentation Group

name: doc

perm: r

repo: /doc

Note:

r = read

w = write

What all that means?

This means that the Administration Group has full access on the svn repository, because we shall give it access on "/". Whereby the Staff Group has full access only to the "/files" location inside the repository. Finally, the Documentation Group has read only access to the "/doc" location inside the repository.

Now let's go and configure our acl's, so edit the acl file:

```
# vim /usr/local/svn/conf/authz
```

Add the following:

```
[groups]
```

```
admins = admin
```

```
staff = ali, mohamed
```

```
readers = mohamed
```

```
[/]
```

```
@admins = rw
```

```
[/files]
```

```
@staff = rw
```

```
[/doc]
```

```
@readers = r
```

Save and close the file. We are finished configuring the svn web access.

Now let's make sure our Apache goes up after a reboot:

```
# chkconfig httpd on
```

And finally let's restart the Apache Server:

```
# service httpd restart
```

Now for the user with the "admin" privileges, he/she can access the svn from the web using the following url (*a username/password shall be required*):

```
http://example.com/svn
```

For users in the staff group, the url is the following (*a username/password shall be required*):

```
http://example.com/svn/files
```

For users in the documentation group, the url is the following (*a(a username/password shall be required*):

```
http://example.com/svn/doc
```

References:

```
http://svnbook.red-bean.com
```

Done by:

Ali Al-Shemery