

Subversion (SVN) is a version control system. It allows users to keep track of changes made over time to any type of electronic data. Typical uses are versioning source code, web pages or design documents.

To complete the "How to use SVN (Subversion)", here comes a simple installation guide for Linux Debian servers. Most of the configuration should be the same for other Linux distributions but there could be some differences in the paths.

Configurations are different if you are using Apache 1 or Apache 2 and you won't be able to have the same functionalities and access methods.

Web server administrators are slow to move to Apache2 and most of us wish to keep Apache 1.3 on our production server. It is possible to use both Apache at the same time.

When running both versions of apache, Apache 1.3 is the front-end server providing SSL encryption (https) and proxy server. Apache 2 becomes a dedicated SVN server listening on a specific port.

Contents

- 1 Installation
 - ◆ 1.1 Apache 1.3
 - ◆ 1.2 Apache 2.2
 - ◆ 1.3 Subversion
 - ◆ 1.4 Instructions
- 2 svnserv
 - ◆ 2.1 Configuration
 - ◆ 2.2 Warning
 - ◆ 2.3 Goody
 - ◆ 2.4 Limitations
- 3 apache 1.3
 - ◆ 3.1 WebSVN installation
 - ◆ 3.2 Safe mode
- 4 Apache 1.3 + Apache 2.2
 - ◆ 4.1 Generate SSL certificate
 - ◆ 4.2 Virtual server for https://svn.myserver.ltd on Apache 1.3
 - ◆ 4.3 Apache 2 configuration
 - ◇ 4.3.1 Multiple repository structure proposal
 - ◇ 4.3.2 Single repository structure
- 5 Apache 2.2
- 6 Conclusion
- 7 External Links

Installation

Apache 1.3

Server

```
apt-get install apache
```

SSL module

```
apt-get install libapache-mod-ssl
```

Apache 2.2

```
apt-get install apache2 libapache2-svn
```

Subversion

```
apt-get install subversion subversion-tools
```

Instructions

Create a directory **/home/svn** or elsewhere but stick. If you haven't created any repositories create one:

```
svnadmin create /home/svn/test
```

svnserve

Svnserve is a svn server shipped in the SVN package. Use it if you want to install a svn server without apache 2.

Configuration

Check the options with:

```
svnserve --help
```

You can just run the daemon by giving the parent repositories parent path (`--root`), the port to listen to on the server (`--listen-port`).

Subversion_(SVN)_server_installation_guide

```
svnserve -d --root /home/svn --listen-port 1234
```

You should now be able to browse test

```
svn://youserver.ltd:1234/test
```

To manage the authentication for user or limit writing rights to authenticated users only, you have to modify:

- /home/svn/test/conf/passwd:

```
### This file is an example password file for svnserve.
### Its format is similar to that of svnserve.conf. As shown in the
### example below it contains one section labeled [users].
### The name and password for each user follow, one account per line.
```

```
[users]
myusername=mypassword
```

- /home/svn/test/conf/svnserve.conf:

```
### These options control access to the repository for unauthenticated
### and authenticated users. Valid values are "write", "read",
### and "none". The sample settings below are the defaults.
anon-access = read
auth-access = write
### The password-db option controls the location of the password
### database file. Unless you specify a path starting with a /,
### the file's location is relative to the conf directory.
### Uncomment the line below to use the default password file.
password-db = passwd
### The authz-db option controls the location of the authorization
### rules for path-based access control. Unless you specify a path
### starting with a /, the file's location is relative to the conf
### directory. If you don't specify an authz-db, no path-based access
### control is done.
### Uncomment the line below to use the default authorization file.
# authz-db = authz
### This option specifies the authentication realm of the repository.
### If two repositories have the same authentication realm, they should
### have the same password database, and vice versa. The default realm
### is repository's uuid.
realm = test Repository
```

Now only **myusername** will be able to write into the repository.

Warning

You will have to restart the daemon by hand every time you restart the server, remember the port, give again the root...

Goody

Here is a little script that makes life much easier.

- create the file `/etc/init.d/svnserve` and put the following content:

```
#!/bin/sh

unset LC_CTYPE

case "$1" in
  start)
    echo "Starting svnserve"
    # Ensure that we run from a readable working dir, and that we do not
    # lock filesystems when being run from such a location.
    cd /
    start-stop-daemon --start --quiet --background --make-pidfile \
      --pidfile /var/run/svnserve.pid --exec /usr/bin/svnserve \
      --chuid ${SVNSERVE_USER:-svnserve}:${SVNSERVE_GROUP:-svnserve} -- \
      --foreground --daemon ${SVNSERVE_OPTS:---root=/home/svn} --listen-port 1007
    exit $?
    ;;
  stop)
    echo "Stopping svnserve"
    start-stop-daemon --stop --quiet --pidfile /var/run/svnserve.pid
    exit $?
    ;;
  *)
    echo 'Usage $0 start|stop'
    ;;
esac

exit 1
```

- You will need to create a **svnserve** user and group !!!
- To allow svn viewers, trac or other to read the repository without any troubles (safe mode) you'll need to change the owner and group of **/home/svn**:

```
chown -R /home/svn
```

Now to start/stop the daemon, you do:

```
/etc/init.d/svnserve start
```

```
/etc/init.d/svnserve stop
```

This is much more comfortable to me.

Limitations

There are few points that make svnserve not suitable enough for my server which has many virtual hosts:

Subversion_(SVN)_server_installation_guide

- With svnserve you can set only one parent path for the repositories for the entire web server. It's not good enough if you have multiple web sites on the same server since they will, no matter what the domain name is, point to the same parent path. **i.e:** server1.ltd:1234 == server2.ltd:1234 == anotherserver1.ltd:1234...

Having many web sites with different project it keeps me away.

- Not being able to browse the repository via HTTP makes it not possible to set SSL encryption (https). So most of the data can be seen. The only solution is to use ssh:

```
ssh+svn://youruser@yourserver.ltd:1234/test
```

But it also means that you need valid users on the server. You better trust the people to whom you give access to your server.

apache 1.3

With svn and DAV installed on your web server, you should be able to browse the repository via a normal web browser. I've tried for a long time to set it up with Apache 1.3 but it seems to be impossible.

WebSVN installation

One alternative to at least browse the repository is to install a svn browser. I'm using [WebSVN](#) but there are a few others.

If your web root directory is `/var/www`, then:

```
cd /var/www
wget http://websvn.tigris.org/files/documents/1380/34308/websvn-2.0rc4.tar.gz
tar -xvzf websvn-2.0rc4.tar.gz
mv websvn-2.0rc4 websvn
cd websvn/include
cp distconfig.inc config.inc
nano config.inc
```

In `config.inc`, you define the repository(ies) you'd like to be browsed. Here is a sample I use:

```
<?php
# vim:et:ts=3:sts=3:sw=3:

// WebSVN - Subversion repository viewing via the web using PHP
// Copyright (C) 2004 Tim Armes
//
// This program is free software; you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation; either version 2 of the License, or
// (at your option) any later version.
//
// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
```

Subversion_(SVN)_server_installation_guide

```
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with this program; if not, write to the Free Software
// Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
//
// --
//
// config.inc
//
// Configuration parameters

// --- FOLLOW THE INSTRUCTIONS BELOW TO CONFIGURE YOUR SETUP ---

// --- PLATFORM CONFIGURATION ---

// Uncomment the next line if your running a windows server
//
// $config->setServerIsWindows();

// Configure these lines if your commands aren't on your path.
//
// $config->setSVNCommandPath("Path/to/svn and svnlook/ e.g.
// c:\\program files\\subversion\\bin");
// $config->setDiffPath("Path/to/diff/command/");

// For syntax colouring, if option enabled...
// $config->setEnscriptPath("Path/to/enscript/command/");
// $config->setSedPath("Path/to/sed/command/");

// For delivered tarballs, if option enabled...
// $config->setTarPath("Path/to/tar/command/");

// For delivered GZIP'd files and tarballs, if option enabled...
// $config->setGzipPath("Path/to/gzip/command/");

// --- REPOSITORY SETUP ---

// There are 2 methods for defining the repositories available on the system.
// Either you list them by hand, in which case you can give each one the name of
// your choice, or you use the parent path function, in which case the name of
// the directory is used as the repository name.
//
// In all cases, you may optionally supply a group name to the repositories.
// This is useful in the
// case that you need to separate your projects. Grouped Repositories are
// referred to using the // convention GroupName.RepositoryName
//
// Performance is much better on local repositories (e.g. accessed by file:///).
// However, you can also provide an interface onto a remote repository. In this
// case you should supply the username and password needed to access it.
//
// To configure the repositories by hand, copy the appropriate line below,
// uncomment it and
// replace the name and URL of your repository.

// Local repositories (without and with optional group):
//
// $config->addRepository("NameToDisplay", "URL to repository
// (e.g. file:///c:/svn/proj)");
// $config->addRepository("NameToDisplay", "URL to repository
// (e.g. file:///c:/svn/proj)", "group");
```

Subversion_(SVN)_server_installation_guide

```
//
// Remote repositories (without and with optional group):
//
// $config->addRepository("NameToDisplay", "URL (e.g. http://path/to/rep)",
// NULL, "username", "password");
// $config->addRepository("NameToDisplay", "URL (e.g. http://path/to/rep)",
// "group", "username", "password");
//
// To use the parent path method, uncomment the newt line and and replace the
// path with your one. You can call the function several times if you have
// several parent paths. Note that in this case the path is a filesystem path
//
// $config->parentPath("Path/to/parent (e.g. c:\\svn)");

# ALL REPOSITORIES IN /home/svn/ WILL BE BROWSABLE
$config->parentPath("/home/svn/");

// --- LOOK AND FEEL ---
//
// Uncomment ONLY the template file that you want.

// $config->setTemplatePath("$locwebsvnreal/templates/Standard/");
$config->setTemplatePath("$locwebsvnreal/templates/BlueGrey/");
// $config->setTemplatePath("$locwebsvnreal/templates/Zinn/");

// You may also specify a per repository template file by uncommenting and
// changing the following
// line as necessary. Use the convention "groupname.myrep" if your repository
// is in a group.

// $config->setTemplatePath("$locwebsvnreal/templates/Standard/", "myrep");
// Access file for myrep

// The index page containing the projects may either be displayed as a flat view
// (the default),
// where grouped repositories are displayed as "GroupName.RepName" or as a tree
// view.
// In the case of a tree view, you may choose whether the entire tree is open by
// default.

// $config->useTreeIndex(false); // Tree index, closed by default
$config->useTreeIndex(true); // Tree index, open by default

// By default, WebSVN displays a tree view onto the current directory. You can
// however
// choose to display a flat view of the current directory only, which may make
// the display
// load faster. Uncomment this line if you want that.

// $config->useFlatView();

// --- LANGUAGE SETUP ---
// --- LANGUAGE SETUP ---

// WebSVN uses the iconv module to convert messages from your system's character
// set to the UTF-8 output encoding. If you find that your log messages aren't
// displayed correctly then you'll need to change the value here.
//
// You may also specify the character encoding of the repository contents if
```

Subversion_(SVN)_server_installation_guide

```
// different from the system encoding. This is typically the case for windows
// users, whereby the command line returns, for example, CP850 encoded strings,
// whereas the source files are encoded as iso-8859-1 by Windows based text
// editors. When display text file, WebSVN will convert them from the content
// encoding to the output encoding (UTF-8).
//
// WebSVN does its best to automate all this, so only use the following if it
// doesn't work
// "out of the box". Uncomment and change one of the examples below.
//
// $config->setInputEncoding("CP850"); // Encoding of result returned by svn
// command line, etc.
// $config->setContentEncoding("iso-8859-1"); // Content encoding of all your
// repositories

// You may also specify a content encoding on a per repository basis. Uncomment
// and copy this

// line as necessary.
//
// $config->setContentEncoding("iso-8859-1", "MyEnc");

// Note for Windows users: To enable iconv you'll need to enable the extension
// in your php.ini file
// AND copy iconv.dll (not php_iconv.dll) to your Windows system folder. In most
// cases the correct
// encoding is set when you call $config->setServerIsWindows();.

// Note for *nix users. You'll need to have iconv compiled into your binary.
// The default input and output encodings are taken from your locale
// informations. Override these if they aren't correct.

// Uncomment the default language. If you want English then don't do anything
// here.
//
// include 'languages/catalan.inc';
// include 'languages/danish.inc';
// include 'languages/dutch.inc';
// include 'languages/finnish.inc';
// include 'languages/french.inc';
// include 'languages/german.inc';
// include 'languages/japanese.inc';
// include 'languages/korean.inc';
// include 'languages/norwegian.inc';
// include 'languages/polish.inc';
// include 'languages/portuguese.inc';
// include 'languages/russian.inc';
// include 'languages/schinese.inc';
// include 'languages/slovenian.inc';
// include 'languages/spanish.inc';
// include 'languages/swedish.inc';
// include 'languages/tchinese.inc';
// include 'languages/turkish.inc';

// --- MULTIVIEWS ---

// Uncomment this line if you want to use MultiView to access the repository by,
// for example:
//
// http://servername/wsvn/rename/path/in/repository
// Note: The websvn directory will need to have Multiviews turned on in Apache,
// and you'll need to configure
```

Subversion_(SVN)_server_installation_guide

```
//      wsvn.php

// $config->useMultiViews();

// --- ACCESS RIGHTS ---

// Uncomment this line if you want to use your Subversion access file to control
// access rights via WebSVN.  For this to work, you'll need to set up the same
// Apache based authentication
// to the WebSVN (or wsvn) directory as you have for Subversion itself.  More
// information can be found in install.txt

// $config->useAuthenticationFile("/path/to/accessfile"); // Global access file

// You may also specify a per repository access file by uncommenting and copying
// the following line as necessary.  Use the convention "groupname.myrep" if
// your repository is in a group.

// $config->useAuthenticationFile("/path/to/accessfile", "myrep"); // Access
// file for myrep

// --- FILE CONTENT ---
//
// You may wish certain file types to be GZIP'd and delivered to the user when
// clicked upon.
// This is useful for binary files and the like that don't display well in a
// browser window!
// Copy, uncomment and modify this line for each extension to which this rule
// should apply. (Don't forget the . before the extension.  You don't need an
// index between the []'s).
// If you'd rather that the files were delivered uncompressed with the
// associated MIME type,
// then read below.
//
// $zipped[] = ".dll";

// Subversion controlled files have an svn:mime-type property that can
// be set on a file indicating its mime type.  By default binary files
// are set to the generic application/octet-stream, and other files
// don't have it set at all.  WebSVN also has a built-in list of
// associations from file extension to MIME content type. (You can
// view this list in setup.inc).
//
// Determining the content-type: By default, if the svn:mime-type
// property exists and is different from application/octet-stream, it
// is used.  Otherwise, if the built-in list has a contentType entry
// for the extension of the file, that is used.  Otherwise, if the
// svn:mime-type property exists has the generic binary value of
// application/octet-stream, the file will be served as a binary
// file.  Otherwise, the file will be brought up as ASCII text in the
// browser window (although this text may optionally be coloured.
// See below).
//
// Uncomment this if you want to ignore any svn:mime-type property on your
// files.
//
// $config->ignoreSvnMimeTypes();
//
// Uncomment this if you want skip WebSVN's custom mime-type handling
//
// $config->ignoreWebSVNContentTypes();
//
```

Subversion_(SVN)_server_installation_guide

```
// Following the examples below, you can add new associations, modify
// the default ones or even delete them entirely (to show them in
// ASCII via WebSVN).

// $contentType[".c"] = "plain/text"; // Create a new association
// $contentType[".doc"] = "plain/text"; // Modify an existing one
// unset($contentType[".m"]) // Remove a default association

// --- TARBALLS ---

// You need tar and gzip installed on your system. Set the paths above if
// necessary
//
// Uncomment the line below to offer a tarball download option across all your
// repositories.
//
$config->allowDownload();
//
// To change the global option for individual repositories, uncomment and
// replicate
// the required line below (replacing 'myrep' for the name of the repository to
// be changed).
// Use the convention "groupname.myrep" if your repository is in a group.

// $config->allowDownload("myrep"); // Specifically allow downloading for
// 'myrep'
// $config->disallowDownload("myrep"); // Specifically disallow downloading for
// 'myrep'

// You can also choose the minimum directory level from which you'll allow
// downloading.
// A value of zero will allow downloading from the root. 1 will allow
// downloading of directories
// in the root, etc.
//
// If your project is arranged with trunk, tags and branches at the root level,
// then a value of 2 would allow the downloading of directories within
// branches/tags while disallowing the download of the entire branches or tags
// directories. This would also stop downloading of the trunk, but see after
// for path exceptions.
//
// Change the line below to set the download level across all your repositories.

$config->setMinDownloadLevel(2);

// To change the level for individual repositories, uncomment and replicate
// the required line below (replacing 'myrep' for the name of the repository to
// be changed).
// Use the convention "groupname.myrep" if your repository is in a group.

// $config->setMinDownloadLevel(2, "myrep");

// Finally, you may add or remove certain directories (and their contents)
// either globally or on a per repository basis. Uncomment and copy the
// following lines as necessary. Note that these are searched in the order
// than you give them until a match is made (with the exception that all the per
// repository exceptions are tested before the global ones). This means that
// you must disallow /a/b/c/ before you allow /a/b/ otherwise the allowed match
// on /a/b/ will stop any further searching, thereby allowing downloads on
// /a/b/c/.

// Global exceptions possibilities:
```

Subversion_(SVN)_server_installation_guide

```
//
// $config->addAllowedDownloadException("/path/to/allowed/directory/");
// $config->addDisAllowedDownloadException("/path/to/disallowed/directory/");
//
// Per repository exception possibilities:
// Use the convention "groupname.myrep" if your repository is in a group.
//
// $config->addAllowedDownloadException("/path/to/allowed/directory/", "myrep");
// $config->addDisAllowedDownloadException("/path/to/disallowed/directory/",
// "myrep");

// --- COLOURISATION ---

// Uncomment this line if you want to use Enscript to colourise your file
// listings
//
// You'll need Enscript version 1.6 or higher AND Sed installed to use this
// feature.
// Set the path above.
//
$config->useEnscript();

// Enscript need to be told what the contents of a file are so that it can be
// colourised correctly. WebSVN includes a predefined list of mappings from
// file extension to Enscript file type (viewable in setup.inc).
//
// Here you should add and other extensions not already listed or redefine the
// default ones. eg:
//
// $extEnscript[".pas"] = "pascal";
//
// Note that extensions are case sensitive.

// --- RSSFEED ---

// Uncomment this line if you wish to hide the RSS feed links across all
// repositories
//
// $config->hideRSS();
//
// To change the global option for individual repositories, uncomment and
// replicate the required line below (replacing 'myrep' for the name of the
// repository to be changed).
// Use the convention "groupname.myrep" if your repository is in a group.

// $config->hideRSS("myrep"); // Specifically hide RSS links for 'myrep'
// $config->showRSS("myrep"); // Specifically show RSS links for 'myrep'

// --- BUGTRAQ ---

// Uncomment this line if you wish to use bugtraq: properties to show links to
// your BugTracker
// from the log messages.
//
// $config->useBugtraqProperties();
//
// To change the global option for individual repositories, uncomment and
// replicate the required line below (replacing 'myrep' for the name of the
// repository to be changed).
// Use the convention "groupname.myrep" if your repository is in a group.

// $config->useBugtraqProperties("myrep"); // Specifically use bugtraq
```

Subversion_(SVN)_server_installation_guide

```
// properties for 'myrep'
// $config->ignoreBugtraqProperties("myrep"); // Specifically ignore bugtraq
// properties for 'myrep'

// --- MISCELLANEOUS ---
// Comment out this if you don't have the right to use it. Be warned that you
// may need it however!
set_time_limit(0);

// Comment this line to turn off caching of repo information. This will slow
// down your browsing.
$config->setCachingOn();

// Number of spaces to expand tabs to in diff/listing view across all
// repositories

$config->expandTabsBy(8);

// To change the global option for individual repositories, uncomment and
// replicate the required line below (replacing 'myrep' for the name of the
// repository to be changed).
// Use the convention "groupname.myrep" if your repository is in a group.

// $config->expandTabsBy(3, "myrep"); // Expand Tabs by 3 for repository 'myrep'
?>
```

With this configuration I chose to have the source coloration and to give the possibility to the user to download tarballs.

You'll need to install Enscript:

```
apt-get install enscript
```

Safe mode

If your server has safe mode enabled globally, you'll need to disable it by adding in **httpd.conf** or your virtual host:

```
...
<Directory /var/www/websvn>
    php_admin_flag safe_mode off
</Directory>
...
```

Apache 1.3 + Apache 2.2

In this configuration both Apaches are running at the same time. Apache 2.2 becomes a svn dedicated server which listens on a different port than Apache 1.3.

Apache 1.3 will provide the SSL encryption and will be used as a proxy to get the data from Apache 2.

In this configuration Apache 2 will listen only on port **1234**.

Safe mode

Generate SSL certificate

If you are not ready to buy SSL certificates you can still generate them. Let's say that we want our **test** repository to be accessible via <https://svn.myserver.ltd/>

For the following steps you'll need to give a passphrase that you **have to remember!**

- First we generate a key

```
openssl genrsa -des3 -rand file1:file2:file3:file4:file5 -out svn.myserver.ltd.key 1024
```

- Now we remove the triple-DES encryption. It's a security break but if you leave the encryption Apache will ask you the passphrase for every certificate every time you restart it. It is strongly advised to let only **read** right (**400**) to the root user. After, if anybody manages to read it, let say that you have other problems than that

```
openssl rsa -in svn.myserver.ltd.key -out svn.myserver.ltd.pem
```

- Next step is to create the Certificate Signing Request (CRS)

```
openssl req -new -key svn.myserver.ltd.key -out svn.myserver.ltd.csr
```

Fill as below:

```
Using configuration from /usr/local/ssl/openssl.cnf
Enter PEM pass phrase:Enter pass phrase here
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:FI
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:Helsinki
Organization Name (eg, company) [Internet Widgits Pty Ltd]: Myserver.ltd, Inc.
Organizational Unit Name (eg, section) []:.
Common Name (eg, YOUR name) []:svn.myserver.ltd
Email Address []:webmaster@ myserver.ltd

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

Subversion_(SVN)_server_installation_guide

All those information will be displayed if you check the certificate on your web browser.

- Finally we generate a Self-Signed Certificate for 360 days:

```
openssl x509 -req -days 360 -in svn.myserver.ltd.csr -signkey svn.myserver.ltd.key -out svn.myserver.ltd.crt
```

To keep your certificates organized, place them in their dedicated directories:

```
/etc/apache/ssl.crt/svn.myserver.ltd.crt
/etc/apache/ssl.csr/svn.myserver.ltd.csr
/etc/apache/ssl.key/svn.myserver.ltd.key
/etc/apache/ssl.key/svn.myserver.ltd.pem
```

Virtual server for <https://svn.myserver.ltd> on Apache 1.3

Add the following entry in your virtual host configuration file.

It can be in

```
/etc/apache/httpd.conf
```

With my config I put all the virtual hosts in

```
/etc/apache/conf.d/
```

and the directory path is added to httpd.conf. All files in this directory will be included to the apache configuration. For now let's create:

```
/etc/apache/conf.d/ssl.conf
```

```
### https://svn.myserver.ltd ###
<VirtualHost xxx.xxx.xxx.xxx:443>
    ServerName svn.myserver.ltd
    DocumentRoot /home/svn/test/html
    ServerAdmin admin@myserver.ltd

    # SSL
    SSLEngine on
    SSLCipherSuite ALL:!ADH:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL:+SSLv3

    SSLCertificateKeyFile /etc/apache/ssl.key/svn.myserver.ltd.pem
    SSLCertificateFile /etc/apache/ssl.crt/svn.myserver.ltd.crt

    # SSL Protocol Adjustments:
    SetEnvIf User-Agent ".*MSIE.*" nokeepalive ssl-unclean-shutdown downgrade-1.0 force-response-1.0

    ProxyPass / http://svn.myserver.ltd:1234 /
    ProxyPassReverse / http://svn.myserver.ltd:1234/

    CustomLog /var/log/apache/wssl_access.svn.myserver.ltd.log combined
    ErrorLog /var/log/apache/wssl_error.svn.myserver.ltd.log
```

Subversion_(SVN)_server_installation_guide

```
TransferLog /var/log/apache/wssl_access.svn.myserver.ltd.log
</VirtualHost>
### <https://svn.myserver.ltd > ###
```

Because you added SSL certificates, you will need to make a complete restart of Apache 1.3:

```
/etc/init.d/apache stop
/etc/init.d/apache force-reload
/etc/init.d/apache restart
```

Apache 2 configuration

To make Apache 2 listening on port **1234**, modify

```
/etc/apache2/ports.conf
```

Be sure to have **/etc/apache2/httpd.conf** even empty. For some reason, one of my servers didn't have it and it took a while to figure out why apache did not start.

All virtual host are located in

```
/etc/apache2/sites-enabled/
```

We can create

```
/etc/apache2/sites-enabled/myserverltd
```

that will contain all the virtual hosts for **myserver.ltd** on Apache2.

Add one of the following configurations:

Multiple repository structure proposal

```
### svn.myserver.ltd ###
<VirtualHost *>
    ServerAdmin admin@myserver.ltd
    DocumentRoot /home/svn/test/html
    ServerName svn.myserver.ltd

    <Directory />
        Options FollowSymLinks Indexes
        AllowOverride All
    </Directory>

    # Public repositories
    <Location /public>
        Options FollowSymLinks Indexes
        DAV svn
```

Subversion_(SVN)_server_installation_guide

```
SVNParentPath /home/svn/test/public
AuthName "Test SVN"
AuthUserFile /home/svn/test/svn.myserverltd.htpasswd
order allow,deny
allow from all
<LimitExcept GET PROPFIND OPTIONS REPORT>
    Require valid-user
</LimitExcept>
</Location>

# Private repository
<Location /private>
    Options FollowSymLinks Indexes
    DAV svn
    SVNParentPath /home/svn/test/private
    AuthName "Test SVN"
    AuthUserFile /home/svn/test/svn.myserverltd.htpasswd
    Require valid-user
</Location>

# Logs:
LogLevel warn
ErrorLog /var/log/apache2/error.svn.myserver.ltd.log
CustomLog /var/log/apache2/access.svn.myserver.ltd.log combined
ServerSignature On
</VirtualHost>
###< svn.myserver.ltd >###
```

To restart apache 2

```
/etc/init.d/apache2 restart
```

With this configuration example, test is a directory which has the following structure:

```
test
|_ html
|_ public
|_ private
```

- In **html** directory, you put the index page of your choice. A SVN browser for example. If you follow the previous installation instruction then just install mod php for apache2

```
apt-get install libapache2-mod-phpX
```

- In **public** directory, you can put all the repositories that anybody can read but need restriction on writing. You have to give the path of the proper authentication file for that **AuthUserFile** **/home/svn/test/svn.myserverltd.htpasswd**

You can access the repositories like <https://svn.myserver.ltd/public/repository1>. If you get **forbidden** when trying to get <https://svn.myserver.ltd/public/> it is not a configuration problem! Apache just won't list all repositories.

Subversion_(SVN)_server_installation_guide

- In **private** directory, you put all the projects that need authentication for read and write.
AuthUserFile /home/svn/test/svn.myserverltd.htpasswd

You can access the repositories like <https://svn.myserver.ltd/private/repository1>. Once again, if you get **forbidden** when trying to get <https://svn.myserver.ltd/private/> it is not a configuration problem! Apache just won't list all repositories.

If you don't have the authentication file then:

- Create the file

```
htpasswd -c /home/svn/test/svn.myserverltd.htpasswd myuser
```

- Add users

```
htpasswd /home/svn/test/svn.myserverltd.htpasswd myuser2
```

- More options

```
htpasswd --help
```

Now, you just need to decide in which directory your repository belongs the best.

Goody

Here is a script that adds repository with the basic archive structure (trunk, tags, branches).

- Create **/home/svn/test/makerep**
- Add:

```
#!/bin/sh
#
# svn projects creation
#
dir='private/';
serv="https://svn.myserver.ltd/"
cwd="/home/svn/test/"

echo -n "Private or public projet ? [P/c]";
read param;

case $param in
  c|C)   echo " Public Site";dir='public/';;
  p|P)   echo " Private Site";dir='private/';;
  *)     echo " Private Site";;
esac
```

Subversion_(SVN)_server_installation_guide

```
echo -n "repository name :"  
read projet;  
  
site="$serv$dir$projet";  
  
while true; do  
    echo -n "Confirm : $site ? (y/n)";  
    read yn;  
    case $yn in  
        [nN]* ) echo "Cancel";echo;exit;;  
        [yY]* ) echo;break;;  
    esac  
done  
  
echo " svnadmin create $cwd$dir$projet --fs-type fsfs";  
svnadmin create $cwd$dir$projet  
echo " chown -R www-data:www-data $cwd$dir$projet";  
chown -R www-data:www-data $cwd$dir$projet  
echo " svn mkdir $site/trunk $site/branches $site/tags -m \"Subversion archive folder creation.\""  
svn mkdir $site/trunk $site/branches $site/tags -m "Subversion archive folder creation."  
  
echo "  
#####  
Finished!  
Remember to add users...  
#####
```

- Make sure to add the execute mod

```
chmod +x /home/svn/makerep
```

- Then run it to add repositories in /home/svn/test

```
./home/svn/test/makerep
```

Single repository structure

If you want <https://svn.myserver.ltd> to point directly on the **test repository** then remove all <Location /*> from the previous configuration and add

```
# Public repositories  
<Location />  
    Options FollowSymLinks Indexes  
    DAV svn  
    SVNPath /home/svn/test  
    AuthName "Test SVN"  
    AuthUserFile /home/svn/test/conf/svn.myserverltd.htpasswd  
    order allow,deny  
    allow from all  
    <LimitExcept GET PROPFIND OPTIONS REPORT>  
        Require valid-user  
    </LimitExcept>  
</Location>
```

You can access the **test** repository like <https://svn.myserver.ltd/>

At this point you should be able to browse the repositories

- ◇ <https://svn.myserver.ltd/>
- ◇ <https://svn.myserver.ltd/public/repository1>
- ◇ <https://svn.myserver.ltd/private/repository1>
- ◇ ...

via a normal web browser.

Apache 2.2

If Apache 2.2 is your main server, then:

- modify **/etc/apache2/ports.conf** to listen on port 80 and 443

```
Listen 80
Listen 443
```

- Make sure to have SSL module installed and enabled.
- Combine the previous configurations and forget the proxy stuff.
- Add all your virtual hosts in **/etc/apache2/sites-enabled/**
- Restart

You should be able to browse the repositories

- ◇ <https://svn.myserver.ltd/>
- ◇ <https://svn.myserver.ltd/public/repository1>
- ◇ <https://svn.myserver.ltd/private/repository1>
- ◇ ...

via a normal web browser.

Conclusion

There might be other configurations or better advices. Please complete the document.

I hope this will help you setting up your own svn server to centralize and manage your projects efficiently.

External Links

- [SVN official site](#)
- [SVN on wikipedia](#)