**Purpose of this howto**

This howto will illustrate a way to install and configure Subversion and websvn on a Debian server with the following features:

* multiple repository [Subversion](http://subversion.tigris.org/)
* access to the repositories via WebDAV (http, https) and ssh
* Linux system account access control and/or Apache level access control
* a secured [websvn](http://websvn.tigris.org/) (php web application for easy code browsing)
* configured syntax coloring in websvn with [gnu enscript](http://www.codento.com/people/mtr/genscript/)

I will not specifically configure *inetd* with *svnserve* in this howto. Rest assured that Subversion will be totally functional without it. You can copy/paste most of the howto to get it working.

Packages that are assumed to already be installed

This howto assumes PHP and apache2 are installed and configured. Configuring apache2 with SSL is optional.

Setting up Subversion

**Subversion packages**

As root you can enter the following commands to install the packages required for our Subversion setup:

*# apt-get update  
# apt-get install subversion  
# apt-get install libapache2-svn*

The package libapache2-svn will install the subversion WebDAV apache module.

**Creating and populating repositories**

To work with in this howto we'll create two repos:

*# mkdir /var/svn-repos/  
# svnadmin create --fs-type fsfs /var/svn-repos/project\_zen   
# svnadmin create --fs-type fsfs /var/svn-repos/project\_wombat*

The repository directories need the proper permissions for apache and the other users. I'll make a group and add users to it (don't just copy/paste here). The apache user won't be put in the group because I find it less secure.

*# groupadd subversion  
# addgroup john subversion  
# addgroup bert subversion  
# addgroup you subversion  
...  
# chown -R www-data:subversion /var/svn-repos/\*  
# chmod -R 770 /var/svn-repos/\**

Let's set up easy ssh connectivity, on a user machine enter the following commands:

*$ mkdir ~/.ssh/  
$ cd ~/.ssh/  
$ ssh-keygen -t dsa  
$ cat ~/.ssh/id\_dsa.pub | ssh you@example.com "cat - >> ~/.ssh/authorized\_keys"*

The server *example.com* is the server we installed Subversion on. For easy ssh use you can chose not to use a passphrase with your key or use an agent to keep authenticated. Otherwise each transaction between the user machine and Subversion will require the user to enter a password (very inconvenient). Using an agent can be done like this:

*$ ssh-agent  
$ ssh-add  
$ ssh you@example.com*

All should be set now to use the a repository. You may test it like this, it shows an import and a checkout:

*$ mkdir ~/TEMP/  
$ echo "testing svn" > ~/TEMP/testing.txt  
$ svn import -m "importing test over ssh+svn" ~/TEMP/ svn+ssh://example.com/var/svn-repos/project\_zen/trunk  
$ svn co svn+ssh://example.com/var/svn-repos/project\_zen/trunk testcheckout*

As a result the testing.txt file should be in a directory called *testcheckout*. On the serverside you can check the repositories with *svnlook*.

*# svnlook tree /var/svn-repos/project\_zen/*

**Configuring Subversion WebDAV**

Normally the apache mod will be enabled by default, to ensure this is true enter the following commands:

*# a2enmod dav  
# a2enmod dav\_svn*

Configuration is done in the file */etc/apache2/mods-available/dav\_svn.conf*, but first we'll make an access file.

*# htpasswd2 -c /etc/apache2/dav\_svn.passwd you  
# htpasswd2 /etc/apache2/dav\_svn.passwd john  
# htpasswd2 /etc/apache2/dav\_svn.passwd sten  
...*

This is the content my */etc/apache2/mods-available/dav\_svn.conf* file:

<Location /svn\_zen>  
 DAV svn  
 SVNPath /var/svn-repos/project\_zen  
 AuthType Basic  
 AuthName "Subversion Repository"  
 AuthUserFile /etc/apache2/dav\_svn.passwd  
 Require valid-user  
 SSLRequireSSL  
 </Location>  
   
 <Location /svn\_wombat>  
 DAV svn  
 SVNPath /var/svn-repos/project\_wombat  
 AuthType Basic  
 AuthName "Subversion Repository"  
 AuthUserFile /etc/apache2/dav\_svn.passwd  
 Require valid-user  
 SSLRequireSSL  
 </Location>

You can uncomment the *SSLRequireSSL* file if you don't want to use SSL, but then you need to use *http* and not *https* in the commands that follow. Apache should be restarted and we can test from a user machine. We'll import the same testfile in the wombat project.

*# /etc/init.s/apache2 restart  
$ svn import -m "testing over https" https://example.com/svn\_wombat ~/TEMP/*

Using a webbrowser you can visit your URL *https://example.com/svn\_wombat* and see what was just committed. This is a basic on-line view on the repository, but using a web font-end like websvn will offer a better repository browsing experience.

Setting up websvn

**Required packages**

To get rolling with websvn we'll need to install the following packages, both will show you configuration screens (explained in the next paragraph):

*# apt-get install enscript  
# apt-get install websvn*

Enscript isn't mandatory but we'll need it for syntax coloring in websvn.

**Configuration**

**Enscript** will ask for paper size, this might seem awkward but that's because enscript is also used for converting ASCII files to PostScript. We need it for it's syntax coloring features.

**Websvn** will first ask for which kind of server to configure, go ahead and just press enter.



The next screens ask for a parent repository folder (*/var/svn-repos/* in this case) and specific repository folders, this will determine which repositories will show up in websvn. We will only enter a parent repository, all repositories created in this folder will show up in websvn for users to browse. If you want to show only specific repositories enter their full paths in the second screen and leave the parent path blank.

As a result the file */etc/websvn/svn\_deb\_conf.inc* will be written. You can rerun debian package configuration screens with *dpkg-reconfigure*. Further websvn configuration is done in the file */etc/websvn/config.inc*. This is the content of my file with some extension mappings for the syntax coloring.

<?php  
 // --- LOOK AND FEEL ---  
 //  
 // Uncomment ONLY the display file that you want.   
 $config->setTemplatePath("$locwebsvnreal/templates/Standard/");  
 // $config->setTemplatePath("$locwebsvnreal/templates/BlueGrey/");  
 // $config->setTemplatePath("$locwebsvnreal/templates/Zinn/");  
 // $contentType[".c"] = "plain/text"; // Create a new association  
 // $contentType[".doc"] = "plain/text"; // Modify an existing one  
 unset($contentType[".sh"]); // Remove a default association -> .sh is regarded as a binary file by default, needs to be unset  
 // --- 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:  
 //  
 // php is default correctly colourized  
 $extEnscript[".java"] = "java";  
 $extEnscript[".pl"] = "perl";  
 $extEnscript[".py"] = "python";  
 $extEnscript[".sql"] = "sql";  
 $extEnscript[".java"] = "java";  
 $extEnscript[".html"] = "html";  
 $extEnscript[".xml"] = "html";  
 $extEnscript[".thtml"] = "html";  
 $extEnscript[".tpl"] = "html";  
 $extEnscript[".sh"] = "bash";  
 // --- MISCELLANOUS ---  
 // Uncomment 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).  
 // $config->findRepository("myrep")->expandTabsBy(3); // Expand Tabs by 3 for repository 'myrep'  
 ?>  
 <?php  
 if ( file\_exists("/etc/websvn/svn\_deb\_conf.inc") ) {  
 include("/etc/websvn/svn\_deb\_conf.inc");  
 }  
 ?>

Next up is configuring the apache virtualhost for websvn.  
Example using SSL:

<VirtualHost \*:443>  
 ServerAdmin webmaster@example.com  
 ServerName svn.example.com  
 DocumentRoot /var/www/websvn/  
 <Location />  
 Options FollowSymLinks   
 order allow,deny  
 allow from all  
 AuthType Basic  
 AuthName "Subversion Repository"  
 Require valid-user  
 AuthUserFile /etc/apache2/dav\_svn.passwd   
 <IfModule mod\_php4.c>  
 php\_flag magic\_quotes\_gpc Off  
 php\_flag track\_vars On  
 </IfModule>  
 </Location>  
 SSLEngine on  
 SSLCertificateFile /etc/apache2/ssl/apache.pem  
 </VirtualHost>

Example without SSL:

<VirtualHost \*:80>  
 ServerAdmin webmaster@example.com  
 ServerName svn.example.com  
 DocumentRoot /var/www/websvn/  
   
 <Location />  
 Options FollowSymLinks   
 AllowOverride None  
 order allow,deny  
 allow from all  
 AuthType Basic  
 AuthName "Subversion Repository"  
 Require valid-user  
 AuthUserFile /etc/apache2/dav\_svn.passwd   
 <IfModule mod\_php4.c>  
 php\_flag magic\_quotes\_gpc Off  
 php\_flag track\_vars On  
 </IfModule>  
 </Location>  
 </VirtualHost>

Restart apache and have a look at the result at your *https://svn.example.com/*.

Useful Subversion references

**Getting more information**

* [official subversion site](http://subversion.tigris.org/)
* [Version Control with Subversion](http://svnbook.red-bean.com/), free on-line book
* [websvn](http://websvn.tigris.org/)

**Subversion clients**

* [Subclipse](http://subclipse.tigris.org/)
* [RapidSVN](http://rapidsvn.tigris.org/)
* [kdesvn](http://www.alwins-world.de/programs/kdesvn/)
* [Zigversion](http://zigversion.com/) (Mac OS X)
* [Quicksilver plugin](http://docs.blacktree.com/quicksilver/plug-ins/subversion) (Mac OS X)
* [svnX](http://www.lachoseinteractive.net/en/community/subversion/svnx/features/) (Mac OS X)
* [TortoiseSVN](http://tortoisesvn.tigris.org/) (Windows)

I hope you find this howto useful. This isn't a perfect setup, but hopefully it will help you in using Subversion. Please feel free to add comments or corrections.