

[Gentoo Wiki](#)Search: A vertical advertisement for Trustico. At the top is the Trustico logo with the tagline 'global internet security'. Below it is a close-up image of a metal padlock. The text 'SSL CERTIFICATES' is in large, bold, yellow letters. Below that, 'FROM ONLY \$14.95' is written in large, bold, yellow letters. To the left of this text is a small image of an alarm clock. To the right, 'ISSUED WITHIN MINUTES' is written in bold, yellow letters. At the bottom, a red banner contains the text 'FULLY TRUSTED 256 BIT' in white, bold letters. The background of the ad is dark blue.

www.trustico.com
Ads by Goooooogle

A horizontal advertisement for Trustico. On the left is the Trustico logo with the tagline 'global internet security' and the website 'www.trustico.com'. To the right, the text 'TRUSTED SSL CERTIFICATES' is in large, bold, yellow letters. Below this, 'Fully Trusted 256 Bit' is written in white, italicized letters. To the right of the text is a red arrow pointing right with 'FROM ONLY' written inside it. To the right of the arrow is the price '\$14.95' in large, bold, yellow letters. On the far right is a small image of a metal padlock.

www.trustico.com

Feedback - Ads by Google

HOWTO_Apache2_with_subversion_SVN_and_DAV

[Apache2 Series](#)

Main

- [Installing Apache 2](#)
-

Modules

- [mod_ssl](#) (for encrypted https)
 - [mod_php](#) (PHP Support)
 - [mod_ftp](#) (FTP Support)
 - [mod_perl](#) (Perl Support)
 - [mod_python](#) (Python Support)
 - [mod_jk](#) (JSP & Servlet Support)
 - [mod_security](#) (intrusion detection)
 - [mod_vhost_alias](#) (Dynamic virtual hosting)
 - [mod_deflate](#) (deflate compression)
 - [mod_fcgid](#) and [mod_suexec](#) (Secure execution of external applications that support fastcgi)
-

Addons & Tunnels

- [Ruby on Rails](#)
-

Tips

- [Using cacert.org SSL certificates with Apache](#)
-

Configuring

- [Migrating to apache 2.0.54](#)
 - [Apache chroot: the mod_security way](#)
-

Other

- [Bandwidth Limiting](#)
- [HOWTO Apache2 with WordPress](#)
- [Subversion \(SVN & DAV\)](#)
- [PAM authentication with mod_authnz_external](#)

[edit](#)

Introduction

This doc serves to explain howto add subversion and DAV support to Apache2.

For information on how to use the subversion CLI client and server, see [Subversion](#).

Compile Apache without the worker MPM

There is [a suspected conflict between](#) large changeset check-ins and the Apache "worker" MPM. The recommended work-around is to compile Apache with the "prefork" MPM. To do so, first add or modify (in the case of an existing entry) a line in make.conf:

File: /etc/make.conf

```
APACHE2_MPMS="prefork"
```

Then rebuild and reinstall Apache: **emerge -aDNtuv apache**

Why not just emerge --ask ?

Furthermore, please ensure that subversion is merged with the **apache2** and without the **nowebdav** useflag.

Enabling DAV and SVN

Define the use of the DAV and SVN modules for apache2 at startup. For authentication, you'll also probably want SSL. Add this line beneath the existing APACHE2_OPTS line.

File: /etc/conf.d/apache2

```
APACHE2_OPTS="$APACHE2_OPTS -D DEFAULT_VHOST -D SVN -D SVN_AUTHZ
-D DAV -D DAV_FS -D SSL -D SSL_DEFAULT_VHOST"
```

Creating repositories for apache

Apache needs rw access to the repository directory. This can be achieved in two ways.

- Changing owner and/or group of the repository
 1. Change owner of the repository


```
chown apache:apache /var/svn/repos/test -R
```
- Adding apache to svnusers group
 1. Create group svnusers


```
groupadd svnusers
```
 2. Add apache to this group


```
usermod -G svnusers -a apache
```
 3. Change group of the repository


```
chgrp svnusers /var/svn/repos -R
```
 4. Give write access for the group


```
chmod g+w /var/svn/repos -R
```

Initial configuration

File `/etc/apache2/modules.d/47_mod_dav_svn.conf` is shipped with subversion (if apache2 use flag is enabled) and gives an almost working configuration.

```
<Location /svn>
```

```
DAV svn
```

instructs apache to handle all requests whose URL path part begins with /svn via DAV provider (Dav svn line).

If you want to support only one repository you set path for the repository using

```
SVNPath /var/svn/repos
```

However it is more convenient to have multiple repositories supported

```
SVNParentPath /var/svn
```

Now all items under SVNParentPath directive are treated as subversion repositories. If you want to see the list of repositories you should set the SVNListParentPath to on (NOTE: This directive is only available for subversion version 1.3 and higher)

```
SVNListParentPath on
```

Note: If using SVNListParentPath you might get 403 errors as described [here](#). A possible workaround is to change the location to include a trailing slash

```
<Location /svn/>
```

If you use the trailing slash, you might want to add

```
RedirectMatch ^(/svn)$ $1/
```

to your /etc/apache2/httpd.conf settings.

To prevent problems like [Resource cannot be created at the destination...](#), it may also be helpful to add:

```
SVNAutoVersioning On
```

Authentication

Basic

Ok, now we should have apache with subversion working. However anyone can access repository now and probably this is not what we expected.

Following lines in `/etc/apache2/modules.d/47_mod_dav_svn.conf` (old style configuration: `/etc/apache2/conf/modules.d/47_mod_dav_svn.conf`) enable authentication

File: /etc/apache2/modules.d/47_mod_dav_svn.conf

```
AuthType Basic
AuthName "Subversion repository"
AuthUserFile /var/svn/conf/svnusers
Require valid-user
```

AuthType Basic sends password almost in plaintext and is not secure. To prevent this, you need to enable access only through ssl using the directive

```
SSLRequireSSL
```

NOTE: You will have to have SSL enabled in order for this directive to work. To enable SSL, define the use of SSL for apache2 at startup as noted above.

ALTERNATIVE: You can also redirect HTTP access to HTTPS by adding the following lines:

File: /etc/apache2/modules.d/47_mod_dav_svn.conf

```
<Location /svn>
    [...]

    RewriteEngine On
    RewriteCond %{HTTPS} !on
    RewriteRule ^/(.*) https://%{SERVER_NAME}%{REQUEST_URI} [R]
```

OPTIONAL: In order to restrict access to only SSL it's not enough to add SSLRequireSSL to 47_mod_dav_svn.conf. Each configuration directive from 47_mod_dav_svn.conf must be moved inside the virtual host directive for XX_mod_ssl_default-vhost.conf. You also need to change the number of the file so that it loads **after** mod_dav (all the files in /etc/apache2/conf/modules.d are loaded alphabetic order).

SSL-ONLY: Instead of moving the content of 47_mod_dav_svn.conf to inside the SSL vhost, you can just include the file there. Since the default apache config loads modules.d/*.conf, rename: 47_mod_dav_svn.conf to: 47_mod_dav_svn.ssl-conf Then, inside of your SSL vhost, add this at the bottom (while still inside the </If...> tags of course):

```
# Load SSL only modules (like SVN)
Include /etc/apache2/modules.d/*.ssl-conf
```

READONLY: We may want authenticate users to allow them to commit in your subversion, and also allow any anonymous request have at least read rights. So for any write subversion method, a valid user will be required :

```
[...]
AuthType Basic
[...]
# For any operations other than these, require an authenticated user.
<LimitExcept GET PROPFIND OPTIONS REPORT>
    Require valid-user
</LimitExcept>
```

If you change to only SSL you need to add this line to XX_mod_ssl_default-vhost.conf in order to make it work. see (<http://www.mail-archive.com/dev@httpd.apache.org/msg14702.html>)

```
BrowserMatch "SVN" redirect-carefully
```

File /var/svn/conf/svnusers contains username and encrypted password pairs.

To add user and password do it with **htpasswd2** command.

First you need to create user-password storage file

```
htpasswd2 -c /var/svn/conf/svnusers newuser1
```

and then add other users

```
htpasswd2 /var/svn/conf/svnusers newuser2
```

PAM Based

If managing two separate password files is too tedious for you, try using `mod_auth_pam`, which uses PAM to handle authentication. If you haven't already, emerge *mod_auth_pam*.

Note: Be aware that by using `mod_auth_pam` has security implications; the 'apache' user will have read access to `/etc/shadow`. It's possible that if apache were compromised, `/etc/shadow` could be read, and a password-cracking program employed against it.

After emerging `mod_auth_pam`, it's necessary to make a few changes (as root) to your configuration files. For starters, apache needs to be able to read from `/etc/shadow`:

```
# groupadd shadow
# gpasswd -a apache shadow
# chgrp shadow /etc/shadow
# chmod g+r /etc/shadow
```

Note: Pay attention! If you have a webserver and you run it with the same apache these settings potentially let your users read the shadow passwords file and it is *really dangerous*.

Next, edit the following:

File: `/etc/apache2/conf/modules.d/47_mod_dav_svn.conf`

Within the `<location /svn>` tag, add:

```
<IfModule mod_auth_sys_group.c>
AuthPAM_Enabled on
AuthType Basic
AuthName "Subversion Repository"
SSLRequireSSL
Require group subversion
</IfModule>
```

Note: Unlike the 'basic' configuration above, make sure you don't have the line

```
AuthUserFile /var/svn/conf/svnusers
```

It is possible to fall back to the 'basic' authentication listed above should PAM somehow fail; look in `/etc/apache2/modules.d/10_mod_auth_pam.conf` for details.

Note: Authtype Basic sends password almost in plaintext and is not secure. To prevent this, you need to enable access only through ssl using the directive

```
SSLRequireSSL
```

You'll also want to activate `mod_auth_pam`:

File: `/etc/conf.d/apache2`

```
add

-D AUTH_PAM

to APACHE2_OPTS
```

Authorization

If basic authentication is not enough for you, you can enhance basic authentication by means of access control lists. Notice that you have done authentication already and are now going to allow/restrict access on a per user bases. This is done by a special file. The `SVNAccessFile`:

```
AuthzSVNAccessFile /var/svn/conf/svnpolicy
```

Above directive enables the ACL for accessing subversion repositories with apache.

You can define authorization rules in this file in form of `[repos-name:path]` sections and pairs of user names and access rights options, which can be `r`, `w` or empty. For example:

```
[test:/trunk]
testuser1 = rw
testuser2 =
* = r
```

allows `testuser1` to write to the repository (commit, copy, ...), `testuser2` has no access to repository, whereas any other user (wildcard `*`) can read repository (checkout).

In order for the `AuthzSVNAccessFile` directive to work, you will need to enable it in `/etc/conf.d/apache2`:

```
APACHE2_OPTS="... -D SVN_AUTHZ "
```

A few notes on setting access policies. Watch spacing - if you're using a single repository, or even if you're not, nothing in the brackets should have any spacing. For a single repository, you don't need to set the repository name - just the path relative to the `SVNPath` directive. For example:

```
[/trunk/bobsproject]
* = r
[/trunk/bobsproject/hidden]
* =
```

There can be spacing on the permissions files, just not in the brackets for the file directories.

Active Directory/LDAP

Follow these directions if you would like to get your SVN site to authenticate via LDAP or Active Directory.

In `/etc/conf.d/apache2` add `-D LDAP` and `-D AUTH_LDAP`

```
APACHE2_OPTS="-D SSL -D PHP4 -D DAV -D DAV_FS -D SVN -D DAV_SVN -D LDAP  
-D AUTHNZ_LDAP"
```

and here's my working apache config for Active Directory: (you'll have to modify your ldap search path for your directory)

```
<IfDefine SVN>  
<Location /svn/repo>  
DAV svn  
SVNPath /var/svn/repo  
AuthType Basic  
Options Indexes FollowSymLinks  
AllowOverride None  
order allow,deny  
allow from all  
AuthName "Authorize Me"  
AuthBasicProvider ldap  
AuthLDAPURL  
ldap://domain.com:389/  
OU=IT,OU=MainOffice,OU=Locations,OU=Corporate,DC=domainname,DC=com?samAc  
countName?sub?(objectCategory=person)  
AuthLDAPBindDN  
"CN=webuser,OU=Resources,OU=Corporate,OU=AOM,DC=domain,DC=com"  
AuthLDAPBindPassword xxxxxxxxxxxx  
Require valid-user  
</Location>  
</IfDefine>
```

To enable only specific user access:

```
<Location "/useraccess">  
AuthName "user permissions"  
require user larry bill sam  
</Location>
```

For group-level permissions, stick this in the directory block:

```
<Location "/group">  
AuthName "group permissions"  
require group cn=Group,cn=Users,dc=domainname,dc=com  
</Location>
```

If you're having problems building your LDAP queries, run this command on your Windows global catalog server:

```
ldifde -f mydomain.ldif -s 127.0.0.1
```

It will dump the entire ldap directory to a file.

If your Active Directory queries aren't working, you may have more than one tree in your directory. Apache

has trouble searching across a 'forest'. AD has a 'global catalog' that is an index of the entire directory, made for looking up objects. It doesn't have all the attributes, but is a good way to deal with multiple trees. The global catalog is set up by default on port 3268 (vs 389), use this port in AuthLDAPUrl and otherwise query the same way.

```
AuthLDAPURL "ldap://127.0.0.1:3268/DC=domain,DC=com?sAMAccountName?sub?
(objectClass=*)"
```

More information from Microsoft: <http://technet2.microsoft.com/windowsserver/en/library/24311c41-d2a1-4e72-a54f-150483fa885a1033.mspx>

Pretty formatting

You can browse latest revision your repository using your favourite browser. Unfortunately you will notice that it does not look very nice. To change this add

```
SVNIndexXSLT /svnindex.xsl
```

directive inside `<Location /svn>` tag and provide xml stylesheet for transformations. If you are not very familiar with xml, decompress the xsl and css files shipped with subversion in `/usr/share/doc/subversion-<VERSION>/` directory.

Some browsers (notably Opera) will just show a blank page instead of the XSL. You can get around this by preprocessing the XSL through xsltproc, rendering the XSL into html for non-XSL-enabled browsers. To do this, install xsltproc:

```
emerge libxslt
```

And then add the following to your apache2 config:

```
BrowserMatch "Opera" xsltfilter
```

```
ExtFilterDefine xslt mode=output enableenv=xsltfilter \
    intype=text/xml outtype=text/html \
    cmd="/usr/bin/xsltproc /var/www/svnindex.xsl -"
```

Additionally, add the following directives inside the `<Location /svn>` tag:

```
SetOutputFilter xslt
AddDefaultCharset utf-8
```

Resources

- [The Subversion Book](#)
- [mod_auth_ldap Apache Docs](#)

Retrieved from "<http://www.gentoo-wiki.info/Subversion/WebDAV>"

Category: [Apache2](#)

[Browse categories](#) > [Applications](#) > [Webserver](#) > [Apache2](#)

Last modified: Wed, 03 Sep 2008 23:52:00 +0000 **Hits:** 94,459

Created by [NickStallman.net](#), [His Dark Materials - The Golden Compass](#), [Luxury Homes Australia](#)

