1 CVS Howto

1.1 Description

A short description how to use CVS to access the mod_perl and related projects source distributions (also applies to the documentation project).

Just as cvs access to the Apache development tree, the mod_perl code pulled from cvs is not guaranteed to do anything, especially not compile or work. But, that's exactly why we are using cvs, so everyone has access the latest version and can help see to it that mod_perl does compile and work on all platforms, with the various versions and configurations of Perl and Apache. Patches are always welcome and simply testing the latest snapshots is just as, if not more helpful.

It's recommended to subscribe to the modperl-cvs list, which is the place cvs commit logs and diffs are mailed to; at least if you're going to work on the code.

1.2 Anonymous CVS

You can get cvs here: http://www.cvshome.org/

1.2.1 Checking Out

To checkout a fresh copy run the following commands from the directory you want the sources to stay in:

```
% cvs -d ":pserver:anoncvs@cvs.apache.org:/home/cvspublic" login
```

(use the password "anoncvs")

```
% cvs -d ":pserver:anoncvs@cvs.apache.org:/home/cvspublic" co modperl
```

After cvs has finished downloading the files you will find a new directory called *modperl* in the current working directory.

1.2.2 keeping your copy up to date

To keep your local copy in sync with the repository, do

```
% cvs update -dP
```

in the top directory of the project. You should run update evertime before you start working on the project.

1.2.3 Sending Patches

To send a patch, first run:

```
% cvs diff -u
```

in the top directory of the project. The output of diff will be sent to STDOUT, so it might be better to redirect the output to a file:

```
% cvs diff -u > patch
```

If you added files or directories to the project, do a diff against */dev/null*:

```
% diff -u /dev/null newdir/newfilename
```

When this patch is applied, the new dir and the new file will be automatically created.

On Windows-based systems, you can do

```
% diff -u NUL newdir/newfilename
```

instead.

Then send your patch to the maintainer of the project, or the appropriate mailing list.

1.2.4 Useful Default Parameters

Here are some recommended parameters. Save them in *~/.cvsrc*, so you don't have to type them everytime you use cvs.

```
cvs -z9
update -dP
diff -u
```

- *cvs -z9* sets the compression level to 9 (the highest value) to speed up downloading.

- *update -d* automatically creates directories that are missing in your local copy because they where added to the repository after your initial checkout or your last update.

- *update -P* automatically deletes empty directories.

- *diff -u* to use the unified output format so that your changes can be easily merged back into the repository.

1.3 cvsup

`cvsup` has come out of the FreeBSD group. It's a client/server beast that offers an efficient way to sync collections of files over the net, and it is very CVS aware, allowing synchronisation of repositories or checked out files using the cvs deltas to bring the client side files up to date with minimal data transfer.

For a FreeBSD cvsup client see http://www.freebsd.org/cgi/ports.cgi?query=cvsup&stype=all

Others (SunOS, alpha.osf, linux, Solaris2.4, HPAA 10.2, irix): ftp://ftp.postgresql.org/pub/CVSup/

Here's a config file for the client (cvsup) to sync modperl sources.

```
*default tag=.
# comment out the above if you want the raw cvs files

*default host=cvs.apache.org
*default prefix=/path/on/this/machine/to/install/
# a subdir for modperl will appear here ^^^

*default base=/path/on/this/machine/where/cvsup/will/keep/status/info
# you'll never need to look in the 'base' dir.

*default release=cvs delete use-rel-suffix compress

modperl
#apache-1.3
#apache-docs
#modperl-2.0
#httpd-2.0
#modperl-docs
#make your picks above by uncommenting the entries
```

1.4 Getting CVS snapshots

In case you can't get anonymous CVS access to work (or don't want to), there is another possibility: at http://cvs.apache.org/snapshots/ , there are snapshots of the interesting CVS repositories you might want to download. These snapshots are extracted from CVS every 6 hours, so might not contain the *latest* changes, but you'll get pretty close. The file names contain the date and time, which you can also see in the directory listing. So just grab the latest one by date, which will get you the latest version.

See the list of interesting repositories below to find out which snapshots you might want to download.

1.5 Inspecting the CVS server with ViewCVS

ViewCVS is installed on the Apache CVS server. You can reach it at http://cvs.apache.org/viewcvs.cgi/.

From there you can browse the list of available projects, look at the files contained in those projects, their logs, and do colored diffs between versions. This is information you can get from your CVS client (through `cvs log`, `cvs diff` and friends), but the web interface makes it much easier to get a good overview of the different files.

1.6 CVS+SSH access for mod_perl committers

mod_perl committers need to use CVS over SSH. Normal SSH authentications mechanisms apply; you can use public key, password, etc. Refer to your ssh client's manpage.

1.6.1 Getting the Client

If you don't have SSH already installed, you can get it from *http://www.openssh.org/*. If your platform is not supported, this site provides pointers to other implementations. For example for Windows, you might want to look at `Putty`, http://www.chiark.greenend.org.uk/~sgtatham/putty/, especially the `plink` client which runs from the command line. Also see `http://www.ssh.com/`.

1.6.2 Setting Up the Environment

Set your `CVS_RSH` environment variable to `ssh` (if your SSH program has a different name, such as *plink*, use that instead). For example if you are using Bourne-style shell:

```
% export CVS_RSH=ssh
```

You should add this your *.bashrc* or similar file, so that it's set on startup. For Windows, add the line

```
set CVS_RSH=ssh
```

to *autoexec.bat*, or set the environment variable through the System section of the *Control Panel* on NT systems.

The CVS Root is different when using SSH. It's:

```
username@cvs.apache.org:/home/cvs
```

where *username* is your username on the *cvs.apache.org* machine.

1.6.3 Working with CVS

For example let's say you want to work with the *modperl-docs* cvs repository.

To check out the repository do:

```
% cvs -d username@cvs.apache.org:/home/cvs checkout modperl-docs
```

or the shortcut:

```
% cvs -d username@cvs.apache.org:/home/cvs co modperl-docs
```

If it's a first time you ssh to cvs.apache.org, it will ask if you want to allow the host *cvs.apache.org*. Answer *yes*. Then you are asked for your password; type it in. Now you will get a freshly checked out copy of the *modperl-docs* repository.

If you get permission problems, most likely your Unix group wasn't adjusted. Contact the person who gave you the cvs access.

To bring your repository's copy up to date, run:

```
% cvs update -dP
```

or the shortcut:

```
% cvs up -dP
```

The -d option builds directories (if any were added since last update), like checkout does. The -P option prunes empty directories if any. You can put these and other handy options into the rc file. For example with openssh and ssh clients, this is *~/.cvsrc*. Usually we have the following in this file.

```
cvs -z9
update -dP
diff -u
```

The first line tells to use the best compression level when communicating with the server. The last line will do a unified *diff* when cvs diff is used.

If you have done some changes, which weren't committed, it's possible that while trying to merge the differences the client will report about collisions which happens when you've happened to change something that was changed and committed by somebody else. You will have to resolve the conflicts by manual editing of the files in question.

Normally, most changes should go through peer review first. It might be a good idea to discuss the intricacies of a change on the appropriate mailing list before committing anything. Then, to commit:

```
% cvs commit filename(s)
```

or the shortcut:

```
% cvs ci filename(s)
```

But first run cvs update to avoid any problems with out of date versions of files. If you get any conflicts because of it, these must be changed before doing cvs commit, which will incorporate any changes into the repository. To commit only a single file, do:

```
% cvs commit path/to/file
```

If a file or a directory is not under cvs control you have to add it to the cvs first and then commit it:

```
% cvs add path/to/file
% cvs ci path/to/file
```

Usually cvs recognizes binary files by their extensions (e.g. images), but if you commit a file with some unusual extension tell your cvs client that it's a binary file with -kb option:

```
% cvs add -kb path/to/binary/file
```

Then, to add it permanently, you will have to commit it.

1.6.4 A Special Note to modperl-docs Committers

One **very** important note before (ab)using your powers: the mod_perl documentation project makes use of an automatic build system. This means that any changes committed will be periodically rebuilt to create the new site (so you don't have to login and do the manual update/rebuild). This is great, but a side-effect of this is that if someone commits anything that doesn't work, the build will fail, and might break parts of the site.

To avoid these problems, please make sure to run `bin/build` on your working copy to test *before* committing. Also, make sure to run `cvs update` to check that you have added all files to the repository; it's easy to forget adding the files you have created, and `bin/build` will work fine your side, but will fail for others because of the missing files the build depends on.

1.6.5 Avoiding Typing in the Password

After awhile you will get tired of typing the password for every cvs operation that you do. You can avoid that using the public key authentication and the ssh authentication agent. Refer to your ssh client's manpage for more information. For example for the `openssh` and `ssh` clients, the `ssh-add` utility can be used to enter the password once for your private key and then it'll do the public key authentication for you every time you work with cvs over ssh (as long as `ssh-agent` is running). Refer to the relevant manpage for more info (*ssh-agent* in this case).

1.7 mod_perl and Related Projects on cvs.apache.org

- **modperl**

  sources for mod_perl 1.0, for use with apache-1.3

- **apache-1.3**

  the Apache 1.3 HTTP Server

- **modperl-2.0**

  the new version of mod_perl, for use with httpd-2.0. See the install docs for more information about downloading the 2.0 components and installing them.

- **httpd-2.0**

  the new Apache 2.0 HTTP Server

- **apr**

  needed for modperl-2.0

- **apr-util**

needed for modperl-2.0

- **modperl-docs**

the mod_perl documentation (i.e. this site). See the documentation download for information on how to download, build and submit patches to the documentation.

Or see http://cvs.apache.org/viewcvs.cgi/ for a list of all projects.

1.8 See also

- http://httpd.apache.org/dev/anoncvs.txt

For a basic introduction to Anonymous CVS on the Apache CVS server.

- http://cvsbook.red-bean.com/

Open Source Development with CVS is a book published by Coriolis Inc. as part of the Coriolis OpenPress series. Chapters 2, 4, 6, 8, 9, and 10 -- comprising a complete introduction, tutorial and reference to CVS -- are being released free under the terms of the GNU General Public License.

- http://www.cvshome.org/docs/manual/

Version Management with CVS by Per Cederqvist et al is the "official" manual for CVS. Commonly known as "the Cederqvist," the manual covers repositories, branches, and file maintenance, and includes reference material for both CVS users and CVS repository administrators.

1.9 Maintainers

Maintainer is the person(s) you should contact with updates, corrections and patches.

- the documentation mailing list

1.10 Authors

- Thomas Klausner <domm (at) zsi.at>

- Doug MacEachern

- Per Einar Ellefsen <per.einar (at) skynet.be>

Only the major authors are listed above. For contributors see the Changes file.

Table of Contents: