

SVNManager Installation

Documentation

M. Verkerk
Department of Public Health
Erasmus MC – University Medical Center

July 2005

Preface

Version control in the context of this document is all about keeping track of previous versions of electronic documents. We have chosen to use the Subversion software package in order to keep track of our documents.

Subversion stores these documents in a so called repository. These repositories can reside on an Internet server. Therefore, there is some administration needed on these servers regarding the repositories. SVNManager is a web based application to administrate the repository part of Subversion repository servers.

This document will explain the installation of SVNManager. I assume that you're familiar with version control and Subversion. If not, please read the introduction of the 'SVNManager Usage Documentation' or any other documents referred in the literature chapter of this document.

Kind regards,

M. Verkerk

This document is licensed under the Creative Commons Attribution License. To view a copy of this license, see:

http://creativecommons.org/licenses/by/2.0/

Contents

Preface....2
Contents....3
Introduction....4
Chapter 1: SVNManager Requirements....5
- Introduction....5
- General Requirements....5
  - Server Hardware....5
  - Operating System....5
- Software Package Requirements....5
  - Subversion....5
  - Apache 2....5
    - SVN Modules....6
    - PHP Configuration....6
  - PHP....6
  - PEAR....7
  - Prado....7
  - MySQL....7

Chapter 2: SVNManager Installation....8
- Introduction....8
- Installation....8
  - Preparation....8
  - Configuration....8
    - Directory Structure....8
    - Apache Configuration....8
    - Database Configuration....10
    - SVNManager configuration....11
      - Shell commands....11
      - Subversion locations....11
      - SMTP Server address....11
      - Data Source Name of database back-end....11
      - Temporary Administrator Account....12
- Final Steps....12
  - Restarting Apache....12
  - First step....12
  - Finally: Test Repository Access....12

References....13
Literature....13

Introduction

The first chapter summarizes the requirements of SVNManager and will refer to other available documentation regarding installation of these requirements.

The second and final chapter will explain step by step the work that is needed in order to get SVNManager installed and running.

Chapter 1: SVNManager Requirements

Introduction

This chapter summarize the requirements of SVNManager and refer to other available documentation regarding installation of these requirements.

General Requirements

Server Hardware

The Apache 2.0 webserver is the determinant for the hardware requirements for a repository server running SVNManager. It is most likely that you use the same Apache web server for repository access as for accessing SVNManager. According to the Apache 2.0 manual, you need a system with a fast enough processor, fast enough hard drives, which should be determined by confirming an acceptable latency with access to either the SVNManager web application or the repositories, that both communicate over http. Further, it is advised to calculate the maximum amount of simultaneously allowed clients. This can be done by dividing your system's totally available memory by the amount of memory an average Apache process takes. A program such as 'top' can be used to determine this. Apache can be restricted to allow a maximum number of simultaneous clients with the MaxClients setting.

Apache 2.0 documentation:

http://httpd.apache.org/docs-2.0/
http://httpd.apache.org/docs-2.0/misc/perf-tuning.html#hardware

Operating System

There isn't really a restriction to the operation system on which you can run Subversion / SVNManager, since the needed software should compile or work on either a Windows system newer than Windows 98, or any of the available Unixes. Since Subversion is still in development and regularly updated, you could do yourself a favour by choosing a binary supported operating system, such as: Red Hat, Fedora, Debian, SuSE, FreeBSD, OpenBSD, Solaris, Mac OS X, Windows NT, 2000, XP and 2003, or IBM i5/OS (OS/400).

Software Package Requirements

Subversion

First of all, you should install the newest version of Subversion on your system. You need both the binary tools, such as 'svn' and 'svnadmin', and the Apache 2 module. Subversion repositories can be accessed remotely in two ways. First, by means of the 'svnserve' stand alone server and / or by means of a Apache 2 module, named 'mod_dav_svn', probably with the use of an extra module, named 'authz_svn_module'. SVNManager is only compatible with the Apache 2 module, since it uses the more advanced access rights mechanisms of the Apache 2 method.

Apache 2

There are a few notes on installing Apache 2 for use with SVN:

First, in order to run Subversion with Apache you need Apache version 2. It will not work with the ubiquitously used Apache 1.3. If you got a multi purpose web server and you persistently need Apache 1.3 for other purposes, you could decide to install Apache 1.3 and Apache 2 next to each other. The consequence is that one of the Apache's needs to listen to another port than the standard port 80 and therefore the (Subversion) clients accessing the servers need to specify a slightly different URL, including the atypical port number.

Second, SVNManager is a PHP application and needs version 5 of PHP, on which I will come back later. There is a note in the PHP manual that you should not use a threaded Multi-Processing Module (MPM), since it could cause instability, especially on Unix platforms. Please refer to the PHP and Apache 2 documentation for more information.

The Apache configuration file should include the following aspects for Subversion and SVNManager:

SVN Modules

These modules are part of the SVN distribution.

```
LoadModule dav_svn_module     modules/mod_dav_svn.so
LoadModule authz_svn_module   modules/mod_authz_svn.so
```

PHP Configuration

These modules are part of the PHP distribution, described later.

```
LoadModule php5_module        modules/libphp5.so
AddType application/x-httpd-php .php .phtml
DirectoryIndex index.php
```

Notes:

Especially with remotely accessed (extra-net) repositories containing valuable data, it is recommend to configure secure SSL connectivity.

There are different authentication / authorisation possibilities for Apache 2 and Subversion. SVNManager currently needs authentication / authorisation by means of the functionality of the 'mod_authz_svn' module.

There is some more Apache configuration needed, but that will be explained in the next chapter.

Please refer to the following documents for detailed configuration information:

Apache 2.0 documentation:
http://httpd.apache.org/docs-2.0/

PHP Manual
http://www.php.net/manual/en/

Subversion book:
http://svnbook.red-bean.com/

PHP

SVNManager is a PHP program, to be more specific a PHP 5 program. To run SVNManager, PHP 5 needs to be compiled or installed with at least the following functionality:

- Apache 2 support
- Database support, at least one of:
  - mysqli for MySQL support
  - sqlite, which is compiled by default, but not always installed by default (Fedora Core 4?)
- PHP Pear

If you need to compile PHP 5 manually, the configure string should look like:

```
./configure –prefix=/usr/local –with-apxs2=/usr/bin/apxs –with-
mysqli=/usr/bin/mysql_config --with-mysql=/usr --with-zlib
```

Of course you need to specify your preferred paths.

Please refer to the following document for more information:

PHP Manual
http://www.php.net/manual/en/

PEAR

SVNManager uses the PHP Pear module 'VersionControl_SVN' for a small aspect of its functionality. Therefore you need to install this module:

```
pear install --alldeps VersionControl_SVN-alpha
```

This will install the currently alpha version of the VersionControl_SVN package, including needed depending packages.

More information available at:

PHP Pear site
http://pear.php.net

VersionControl_SVN site
http://pear.php.net/package/VersionControl_SVN

Prado

At the moment, Prado is included in the SVNManager package and therefore needs no extra attention.

More information about Prado:

Prado Homepage

http://www.xisc.com/

MySQL

SVNManager uses a database to store the repository, group and user data. It is possible to use either a MySQL or SQLite database. The advantages of a SQLite database is the ease of installation, since it will be automatically created on starting SVNManager for the first time. MySQL could be more convenient because of the ease to alter the database by means of the phpMyAdmin utility.

In order to use MySQL as database back-end you need to take the following steps:

1. Install MySQL
2. Create a user with password for SVNManager
3. Create a database
4. Setup the database with the 'createtables.sql' script (this will ask for the user's password):
   ```
   mysql -u username -p databasename < createtables.sql
   ```
5. Enter the Data Source Name of the database in the config.php, including the server's address, username, password and database name, e.g.:
   ```
   mysql://username:password@localhost/svnmanager"
   ```

Chapter 2: SVNManager Installation

Introduction

This chapter will explain step by step the work that is needed in order to get SVNManager installed and running.

Installation

Preparation

Before you install SVNManager make sure that you have installed all of the software requirements. Further, you should get newest version of SVNManager (currently 0.35), which you can download at the following location:

https://sourceforge.net/projects/svnmanager/

Configuration

Directory Structure

Create a repository directory at a convenient location, such as the apache data directory, e.g. '/srv/www', '/var/www/' or on a windows machine 'C:\www\'.

Do NOT put this directory into the folder of your www-root, e.g. '/srv/www/htdocs'!!

Repository directory, e.g.:
Unix
```
mkdir /srv/www/repos
```

Windows
```
mkdir C:\www\repos
```

This folder is destined for the repositories and the Subversion data files.

Important: make sure that the repository folder is readable and writeable for the user that runs the Apache server, e.g. 'apache', 'wwwrun' on unix systems.
```
chown apache.apache repos
```

Unpack the contents of the svnmanager tar ball or zip file into the same location where you made the 'repos' folder and rename the it to 'svnmanager'.

Apache Configuration

The Apache configuration needs to be altered in order to give the Subversion clients access to the repositories. This is done by means of the SVNParentPath directive inside of a Location directive.

Put the following in your Apache 2 configuration file:

Unix

```
1. <Location /repos>
2.     DAV svn
3.     SVNParentPath /srv/www/repos/
4.     AuthType Basic
5.     AuthName "Subversion realm"
6.     AuthUserFile /srv/www/svn_passwdfile
7.     AuthzSVNAccessFile /srv/www/svn_accessfile
```

```
8.     Require valid-user
9. </Location>
```

Windows

```
1. <Location /repos>
2.     DAV svn
3.     SVNParentPath C:\www\repos
4.     AuthType Basic
5.     AuthName "Subversion realm"
6.     AuthUserFile C:\www\svn_passwdfile
7.     AuthzSVNAccessFile C:\www\svn_accessfile
8.     Require valid-user
9. </Location>
```

Explanation:

1. The relative location of the repositories from the internet, which is translated to the following URL: http://servername.tld/repos/repositoryname
2. Enable WebDAV with the svn file system
3. The location on the file system where the SVN repositories are going to resided (the directory you just made)
4. The Authentication type
5. Name of the realm
6. The user file (created with Apache's 'htpassword' command, by SVNManager)
7. The SVN access file (created with SVNManager)
8. Set the requirement criteria

Important: I emphasize again that it is not recommendable to locate these files and folders into your 'htdocs' main folder!!

Further, the location of the SVNManager application itself has to be specified as well:

Unix
```
Alias /svnmanager/ "/var/www/svnmanager/html/"
```

Windows
```
Alias /svnmanager/ "C:\www\svnmanager\"
```

If desired, you can restrict access to this location by adding a Directory directive:

Unix
```
<Directory "/var/www/svnmanager/" >
        AllowOverride None
        Order deny,allow
        Deny from all
        Allow from domain.tld
</Directory>
```

Windows
```
<Directory "C:\www\svnmanager\" >
        AllowOverride None
        Order deny,allow
        Deny from all
        Allow from domain.tld
</Directory>
```

The previous will only accept connections from hosts with a domain name ending with 'domain.tld'.

Database Configuration

If you want to use a MySQL database back-end, you need to create and set-up a database and and probably an user account for SVNManager to access it:

Create a database:

```
mysqladmin create svnmanager
```

Create database structure (with createtables.sql from svnmanager package):
```
mysql -u root -p svnmanager < createtables.sql
```

Notes:
This will prompt for the root password!
If there is no password, omit the -p
If you want to use another user, replace 'root' with the user name.

Create user to access the database:
Start to mysql client:
```
mysql -u root -p
```
Enter at mysql prompt:
```
GRANT ALL PRIVILEGES ON svnmanager.* TO 'svnmanager'@'localhost'
IDENTIFIED BY 'secret-password'
```

Notes:
If there is no password, omit the -p
If you want to use another user, replace 'root' with the user name.
If you use a remote mysql database, replace 'localhost'.

SVNManager configuration

The final steps regard the configuration of the SVNManager application. This is done by editing the 'config.php' file inside the 'svnmanager' folder. If you use a Windows machine, you should copy the 'config.php.win' template file over 'config.php'.

Shell commands

Enter the location of the following shell-commands. These commands are invoked by SVNManager.

Unix

```
$htpassword_cmd         =     "/usr/bin/htpasswd";
$svn_cmd                =     "/usr/bin/svn";
$svnadmin_cmd           =     "/usr/bin/svnadmin";
```

Windows

```
$htpassword_cmd         =     "c:\\Progra~1\\Apache~1\\Apache2\\bin\\htpasswd";
$svn_cmd                =     "c:\\PROGRA~1\\Subversion\\bin\\svn";
$svnadmin_cmd           =     "c:\\PROGRA~1\\Subversion\\bin\\svnadmin";
```

Subversion locations

These are the locations of the files and directories that we previously made and specified in the Apache configuration.

Unix

```
$svn_repos_loc                =     "/srv/www/repos";
$svn_passwd_file              =     "/srv/www/svn_passwdfile";
$svn_access_file              =     "/srv/www/svn_accessfile";
```

Windows

```
$svn_repos_loc                =     "c:\\www\repos";
$svn_passwd_file              =     "c:\\www\\svn_passwdfile";
$svn_access_file              =     "c:\\www\\svn_accessfile";
```

SMTP Server address

Both Unix and Windows

```
$smtp_server                        =     "smtp.domain.org";
```

Data Source Name of database back-end

As mentioned before, there are two (tested) databases that can be used as back-end, either SQLite or MySQL. The following line specifies the type and location of the desired database.

For both Unix and Windows:

SQLite:

```
$dsn              =     "sqlite://svnmanager.db";
```

This will create a database at the first start of SVNManager, named svnmanager.db, inside the svnmanager folder.

MySQL:

```
$dsn              =     "mysql://svnmanager:password@localhost/svnmanager";
```

Temporary Administrator Account

When there are no administrator accounts specified in the database back-end, this account will allow an administrator to login and create one, probably mostly done the first time accessing SVNManager.

Both Unix and Windows

```
//Administrator account
$admin_name                   =     "admin";
$admin_temp_password          =     "admin";
```

Final Steps

Restarting Apache

After setting up the required software and configuring Apache and SVNManager you should restart Apache in order to make sure that the changed configuration is loaded.

First step

The next thing to do is to browse to SVNManager location and see if it works. The first thing to do is to create an administrator account by either inviting yourself to the system or entering the user's parameters manually. Don't play around to long with the configured administrator account if you haven't change the password!

Finally: Test Repository Access

Create a repository and try to access it with a Subversion client.

For more information on graphical clients, see:

TortoiseSVN Homepage

http://tortoisesvn.tigris.org

Subversion Manual

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

References

[SVN]

Subversion homepage

http://subversion.tigris.org

[OSS]

The Open Source Initiative

http://www.opensource.org/

The free software Foundation

http://www.fsf.org

[CVS]

CVS Homepage

https://www.cvshome.org/

Literature

SVN General Tutorial, M. Verkerk, Erasmus MC, Department of Public Health

SVN Programming Tutorial, M. Verkerk, Erasmus MC, Department of Public Health

SVNManager Usage Documentation, M. Verkerk, Erasmus MC, Department of Public Health

Version Control with Subversion (freely available on-line), Ben-Collins-Sussman et al, O'Reilly Media Inc

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